MeMAD

Methods for Managing Audiovisual Data

memad.eu
info@memad.eu

Twitter – @memadproject
LinkedIn – MeMAD Project

MeMAD Deliverable

D2.2 Implementations of methods adapted to enhanced human inputs

Grant agreement number	780069
Action acronym	MeMAD
Action title	Methods for Managing Audiovisual Data: Combining Automatic Efficiency with Human Accuracy
Funding scheme	H2020–ICT–2016–2017/H2020–ICT–2017–1
Version date of the Annex I against which the assessment will be made	8.5.2019
Start date of the project	1.1.2018
Due date of the deliverable	31.12.2019
Actual date of submission	20.8.2020
Lead beneficiary for the deliverable	Aalto University
Dissemination level of the deliverable	Public

Action coordinator's scientific representative
Prof. Mikko Kurimo
AALTO–KORKEAKOULUSÄÄTIÖ, Aalto University School of Electrical Engineering, Department of Signal Processing and Acoustics
mikko.kurimo@aalto.fi

MeMAD project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 780069. This document has been produced by theMeMAD project. The content in this document represents the views of the authors, and the European Commission has no liability in respect of the content.

Responses to the reviewer comments

We want to thank the scientific reviewers of the MeMAD project for pointing out the weaknesses of Deliverable D2.2 *Implementations of methods adapted to enhanced human inputs* in the project's second intermediate review. Based on the identified shortcomings and other recommendations, we have been able to improve the deliverable in a number of ways including:

1. We have added comparisons to state of the art methods and results when possible.
2. We have emphasised and detailed the collaboration of the partners.
3. We have added a specific Discussion section for self-critical reflection where the applicability of the methods has been discussed .

We answer to the detailed criticism as follows (the reviewer comments are numbered and in *italics*, whereas our responses are alphabetised and in regular font):

1. *D2.2 describes the "Implementations of methods adapted to enhanced human inputs". In addition to the main description it includes summaries of Master Theses and scientific papers. The deliverable was submitted in time. There are various typos and grammar issues.*
    a. We have updated and improved the text and paid special attention to correcting typos and grammar issues and harmonising the style of writing.

2. *All in all, only parts of the deliverable are of sufficient quality. The actual findings and research results fall short when compared to the corresponding state of the art. A strong point of this deliverable is the joint code and service repository.*
    a. We have added new evaluations and comparisons so that the visual domain and audio domain parts of the project are supported with comparison to the state of the art and we show good performances on a broad set of tasks.
    b. For the face recognition task, our innovation is in the application of SOTA methods which combine MTCNN [14] and FaceNet [15] and their productisation via an API. We also evaluate the performance of the detection and recognition of known faces on our own broadcast video data which are typically long videos (see Table 1).
    c. The facial gender classification method is compared to the SOTA models with good results in three benchmark tasks as reported in Table 3.
    d. The performance of our video captioning library has been continuously improving and is keeping up with the development of the state of the art as depicted in Figure 4.
    e. Speech recognition is evaluated on a new and highly challenging YLE dataset gathered as a part of the MeMAD project. Along the evaluations, we observe that our proposed models outperform the baseline models (see Table 5) and our Lingsoft ASR software outperforms the Google ASR (see Table 4) in Finnish and Swedish.

f. We have added a new Discussion section where we emphasise the importance of applying the methods to the partners' own media assets in addition to standard benchmark datasets.

3. *The approach for face recognition only works for people who are considered celebrities, i.e., for people for whom a Google image search produces at least 50 photos. It remains unclear how this approach can be generalised, if at all.*
    a. First, we used the word "celebrities" following a media archive descriptions centric definition, where celebrities are notable people in the similar vein that the notability criteria of Wikipedia[1]. In the revised version of the deliverable we have rephrased some passages accordingly to avoid confusion.
    b. Second, our method has also been extended to recognize people who are rarely seen in visual archives. Our system needs a number of sample images depicting a person (without necessarily naming this person). It is now possible to bypass the crawling stage (that relies on the Google image search engine) and to extract few photos of an unseen person within the video itself.
    c. Consequently, we believe our method can be easily applied to recognize people who are rarely seen in visual archives or to recognize people within a single video.
    d. Following this, we performed two experiments and we proposed a new ground truth dataset in Section 3.1 with results reported in Table 2.

4. *In Section 3.3.1 ("Paragraph-length image captioning") it is mentioned that: "It can be seen that the generated captions read and match the actual image content quite well." While certainly true for this specific example, this statement contradicts the findings of D5.2, which are much more pessimistic when it comes to the overall quality of machine-generated video descriptions.*
    a. The tone of the sentence has been changed to be less optimistic. D5.2 actually studied single-sentence captions, but many of the findings for single-sentence captions apply also to the multi-sentence captions. The latter, however, have not been analysed with an effort equal to that of analysing the single-sentence captions in D5.2.

5. *The ASR results reported in Section 4.2 have a very high word error rate (25.8 and 41.2) while modern approaches typically have a WER of approx. 4-5. The results reported in Table 4 are a bit better but still in the range of 17-20.*
    a. Our experiments reported in Tables 4 and 5 are conducted on a challenging conversational multispeaker television broadcast YLE test set. This dataset contains highly challenging samples and it is gathered as the first domain specific dataset in Finnish and Swedish materials. To the best of our knowledge, there is no other similar benchmark data that exists in the literature. We present a detailed discussion on the properties of the proposed dataset and challenges in

[1] https://en.wikipedia.org/wiki/Wikipedia:Notability

Section 4.2.1. Moreover, the word error rates mean different things in different languages; therefore, a direct comparison of word error rates appearing in other languages is not possible. Instead, we report the performances of our models and the baseline model on our dataset (see Table 5). The results show that our models have significant improvements over baselines both in Finnish and Swedish. We also conduct additional experiments on the same dataset, and we compare the performances of the Lingsoft ASR with the commercial Google ASR. As reported in Table 4, our system achieves good results compared to the Google ASR on our challenging YLE dataset. This dataset is currently available for the MEMAD consortium for evaluation purposes, but we are working towards opening the evaluation dataset for wider use.

b. The TED-LIUM benchmark set-up for Table 6 (was Table 4 in the previous version referred by the reviewers) was chosen here, because the training and test data are all public and the domain of the data is not far from typical TV broadcast material. However, the size of the training data is not huge and thus the conventional hybrid DNN-HMMs are still much better than all the end-to-end systems that typically need thousands of hours to become comparable in performance. Unfortunately, there are no such huge public training data with a compatible domain. The focus of this piece of research work was not to improve the current state-of-the-art in English ASR, but to pave the way to develop better methods for the future multimodal models that will most likely require end-to-end training. For that perspective, this experiment to compare methods that embed speaker information in ASR is indicative despite the size of the models and the training data.

6. *With regard to the multimodal approaches (Section 5), no real progress has been made but rather preliminary steps. Section 5.3 mentions that the approach for "person re-identification and re-referencing" is not fully automatic, which begs the question if/how this approach can be embedded into the prototype.*
    a. Many of the results reported have indeed been preliminary steps, but the work has been continued and will be reported in full in the forthcoming Deliverables D2.3 and D6.9.
    b. The person re-identification and re-referencing approach cannot be applied to all programs, but it can be applied to a substantial subset of them, for example to series programs and news broadcasts where at least some of the persons appear repeatedly. We assume that in such cases it will be sufficient to annotate only some appearances of the reoccurring persons.
    c. The prototype will have some functionality to support person identification with human effort for the processing of programs where this functionality is needed.

7. *The success of the cooperation among all partners should have been reflected in a comprehensive and convincing report. Unfortunately, D2.2 does not match the quality of*

other deliverables. It leaves many questions open about the efficiency (or effort) granted to have MeMAD work together as a team to deliver outstanding results.

a. Throughout the project and within WP2, the project partners have collaborated strongly in many parts, but the original version of the deliverable failed to emphasise this sufficiently. We have updated and revised our deliverable to clarify these collaborations in detail.
b. The data provided by INA, YLE and SURREY is used for evaluating a number of tools developed by EURECOM, AALTO and Lingsoft on various tasks such as face recognition, gender classification, speech recognition and video captioning.
c. The face recognition tool has been improved through a joint effort of EURECOM and AALTO for enabling the recognition of both “celebrities” as well as non-named but recurrent persons in videos.
d. EURECOM and AALTO joined the TRECVID competition individually in 2019 and this gave the groups a chance to compare their systems against each other in a competition. They also joined their effort in the MediaEval 2019 competition.
e. In addition to collaborations conducted solely in visual-based and audio-based approaches, almost all project partners have collaborated in the work on person re-identification and re-referencing described in Section 5.3.

8. *All in all, the reviewers miss some honest self-criticism of how the work was undertaken and a clear path towards making the tools useful and performing against similar approaches. We also would have expected the authors to elaborate on the conclusions of the results of doing the joint approaches and how that will be impacting future actions.*
    a. We have added a new Discussion section where we discuss the relation of our results with respect to the state of the art and the importance of applying them on the MeMAD project’s own video collections.
    b. The collaboration of the project partners has been detailed in many parts of the revised deliverable. The importance of the collaboration for the obtained results and for the completion of the project has been emphasised in the Discussion section.
    c. In the Discussion section we deliberate that even if some of the ambitious goals of the project may likely not be met, we have already gained extremely valuable insights into the applicability of each of the analysis components alone and in combination with others. Some of these applicability issues can still be resolved during the MeMAD project, whereas others will remain to be solved in the future by the multimedia research community as a whole.

MeMAD

Methods for Managing Audiovisual Data

memad.eu
info@memad.eu

Twitter – @memadproject
LinkedIn – MeMAD Project

MeMAD Deliverable

D2.2 Implementations of methods adapted to enhanced human inputs

Grant agreement number	780069
Action acronym	MeMAD
Action title	Methods for Managing Audiovisual Data: Combining Automatic Efficiency with Human Accuracy
Funding scheme	H2020–ICT–2016–2017/H2020–ICT–2017–1
Version date of the Annex I against which the assessment will be made	8.5.2019
Start date of the project	1.1.2018
Due date of the deliverable	31.12.2019
Actual date of submission	20.8.2020
Lead beneficiary for the deliverable	Aalto University
Dissemination level of the deliverable	Public

Action coordinator's scientific representative
Prof. Mikko Kurimo
AALTO–KORKEAKOULUSÄÄTIÖ, Aalto University School of Electrical Engineering, Department of Signal Processing and Acoustics
mikko.kurimo@aalto.fi

MeMAD project has received funding from the European Union's Horizon 2020 research and innovation programme under grant agreement No 780069. This document has been produced by theMeMAD project. The content in this document represents the views of the authors, and the European Commission has no liability in respect of the content.

Authors in alphabetical order		
Name	Beneficiary	e-mail
David Doukhan	INA	ddoukhan@ina.fr
Danny Francis	EURECOM	danny.francis@eurecom.fr
Ismail Harrando	EURECOM	ismail.herrando@eurecom.fr
Benoit Huet	EURECOM	benoit.huet@eurecom.fr
Tuomas Kaseva	AALTO	tuomas.kaseva@aalto.fi
Mikko Kurimo	AALTO	mikko.kurimo@aalto.fi
Jorma Laaksonen	AALTO	jorma.laaksonen@aalto.fi
Tiina Lindh-Knuutila	LLS	tiina.lindh-knuutila@lingsoft.fi
Pasquale Lisena	EURECOM	pasquale.lisena@eurecom.fr
Selen Pehlivan Tort	AALTO	selen.pehlivantort@aalto.fi
Alison Reboud	EURECOM	alison.reboud@eurecom.fr
Aku Rouhe	AALTO	aku.rouhe@aalto.fi
Raphaël Troncy	EURECOM	raphael.troncy@eurecom.fr
Anja Virkkunen	AALTO	anja.virkkunen@aalto.fi

Internal reviewers in alphabetical order		
Name	Beneficiary	e-mail
Maija Hirvonen	UH	maija.hirvonen@helsinki.fi
Maarit Koponen	UH	maarit.koponen@helsinki.fi

Abstract

This deliverable describes the second development iteration of the joint collection of libraries and tools for multimodal content analysis from AALTO, EURECOM, INA, Lingsoft, LLS and Limecraft. Based on the methods' primary input domain, they have been grouped as *visual* (facial person recognition, facial gender classification and video description), *auditory* (speech and gender segmentation, speech recognition and speaker identification and diarisation) and *multimodal* (audio-enhanced captioning, visual–auditory gender classification, person re-identification and multimodal speech recognition) approaches in this report. Special attention has been on methods that combine different modalities and bring human knowledge as input to the learning system. As part of this deliverable, the existing open source components gathered into a joint software collection of tools and libraries have been updated and new components have been added. This deliverable also summarises in an appendix the dissemination activities related to the research work in MeMAD's Work Package WP2 during its second year. Finally, the abstracts of five academic theses together with full texts of ten scientific publications appear at the end of the report. These appendices describe the technological advances related to the software components of MeMAD Task T2.2 in further detail.

Contents

1 Introduction

This deliverable describes the second development iteration of the joint collection of libraries and tools for multimodal content analysis from AALTO, EURECOM, INA, Lingsoft, LLS and Limecraft. The majority of the tools have their roots in the research and development work of the parties before the MeMAD project, but they have been developed further during the first two years of the project. In addition, the partners have just created tools specifically for the project, and most importantly these tools combine the unimodal outputs from the different parties' systems into multimodal approaches.

Following the aim of the MeMAD project, special attention has been on methods that bring human knowledge as input to the learning system. The tools and libraries described in the current document are needed in the continuation of Work Packages WP2, WP3 and WP5, and also in Task T6.2 *Prototype implementation*.

In the next section, the role of Task T2.2 and the requirements for this deliverable according to the MeMAD project's *Description of Action* are first revisited. Then, we briefly describe the visual, auditory and multimodal approaches in Sections 3, 4 and 5 , respectively, followed by a discussion on the strengths and shortcomings of the approaches in Section 6. Finally, in Section 7, we include an updated summary of the components and the open source collection of software that form the primary contents of Deliverable D2.2.

Appendix A summarises the dissemination activities related to Work Package WP2 during the second year of the MeMAD project. At the end of this report, in Appendix B, we have included a set of theses and scientific publications or their drafts that describe the technological advances made in the project.

2 The role of Task T2.2 in the MeMAD project

The aim of MeMAD Work Package WP2 *Automatic multimodal content analysis* is to develop the tools and libraries that AALTO, EURECOM, INA, Lingsoft, LLS and Limecraft have previously created for multimodal analysis, description and indexing of audio and video content. These tools include speech recognition, speaker recognition and diarisation as well as visual and audio description techniques in both uni- and multimodal domains.

Task T2.2 *Using human input to multimodal content analysis* has been the next step in developing these automatic multimodal content analysis tools. The work in this task has been carried out in close collaboration with work in Work Package WP5 and its tasks T5.1 *Multimodal annotation of described video* and T5.2 *Key characteristics of human and machine video description* with regard to the human needs and ways of describing multimodal content to humans. The work of T2.2 contributes to a better understanding of the theoretical concepts relating to multimodal content analysis and to knowledge of what aspects of multimedia content can be captured by automatically extracted features and from existing metadata. This understanding will be needed in remaining work of Work Package WP2 and in Task T6.2 *Prototype implementation*.

This report accompanies the software components stored in a GitHub repository with brief descriptions and evaluations of their use. The address of the GitHub repository is:

`https://github.com/MeMAD-project/mmca`

3 Visual domain

For the description of the visual content of media, several technical components are needed. These have been developed in tight collaboration between the project partners, making use of each partner's special expertise. In this section we report the development of the previously existing and newly created tools and libraries for different types of visual analysis. We start in Section 3.1 by describing EURECOM's new facial person recognition tools that replace their earlier tools introduced in MeMAD Deliverable D2.1. Next, in Section 3.2 we introduce INA's novel facial gender classification library. Finally, Section 3.3 describes the developments in AALTO's and EURECOM's approaches to image and video content description with single-sentence and paragraph-based captioning.

3.1 Facial person recognition

People are undoubtedly an important cue when watching a video. Knowing who appears in a video, when, where and with whom, can reveal interesting patterns of relationships among characters of a movie or a news program. Such person-related annotations are useful for facilitating multimedia search and re-use of video content.

Figure 1 shows an example in which both Emilie Tran Nguyen and Markus Preiss are successfully recognised in a political debate program broadcasted by YLE.

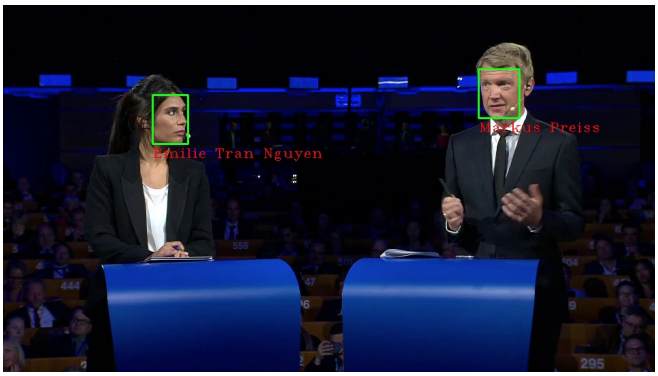


Figure 1: An example of face recognition for broadcast video material.

Related work. During the last decade there has been substantial progress in the methods for automatic recognition of individuals. The recognition process generally consists of two steps. First, faces need to be detected in a video, i.e. which region of the frame may contain a face. Second, those faces can be recognised, i.e. to whom a face belongs.

The Viola-Jones algorithm [11] for face detection and the Local Binary Pattern (LBP) features [12] for the clustering and recognition of faces were the most famous methods until the advent of deep learning and convolutional neural networks (CNN). Nowadays, two main approaches are used for detecting faces in video and both use CNNs. One implementation is available in the Dlib library [13] and provides good performance for frontal images, but it requires an additional alignment step before the face recognition step can be performed. The recent Multi-task Cascaded Convolutional Networks (MTCNN) [14] approach provides even better performance using an image pyramid approach and using face landmarks detection for re-aligning the detected faces to the frontal orientation.

After locating the position and orientation of the faces in the video frames, the face recognition process can be performed. There are several strategies available in the literature for

face recognition. Currently, the most practical approach is to perform face comparison using a transformation space in which similar faces are mapped close together, and to use this representation to identify individuals. Such embeddings, computed on large collections of faces have been made available to the research community, such as the popular FaceNet [15].

In [16], MTCNN and FaceNet are used in combination and tested with eight public face datasets, reaching a recognition accuracy close to 100% and surpassing other methods. These results have been confirmed in several surveys [17, 18] and in recent works [19]. In addition, MTCNN has been recognised to be very fast while having good performance [20].

Our approach. Given the almost perfect performance of the MTCNN + FaceNet face recognition setups, our work focuses on setting up a system built upon these technologies which is suitable for the context and data of the MeMAD project. In this perspective, our contribution does not consist in a new state-of-the-art performance in face recognition, but of the application of face recognition to MeMAD broadcast videos provided by INA and YLE. The MeMAD Face Recognition library, mainly developed by EURECOM with contributions from AALTO and other MeMAD partners, is made of the following modules:

- A crawler which, given a person's name, automatically downloads a set of k photos using Google's image search engine that will be used for training a particular face model. Among the results, the images not containing any face or containing more than one face are discarded. In our experiments, we have typically used $k = 50$.
- A module for extracting face embeddings, where photos or video frames are converted to greyscale, cropped and resized to obtain images containing only a face. The module uses the MTCNN algorithm [14] for face detection. A pretrained FaceNet [15] model with Inception ResNet v1 architecture trained on the VGGFace2 dataset [21] is applied for extracting visual features or embeddings of the faces.
- A clustering module where the face embeddings from one or more video programs are clustered for finding sample facial images of the persons frequently appearing in the footage. The module uses the simple agglomerative complete-linkage clustering available in the SciPy Python library. The sample images and their corresponding clusters can be labeled with human effort in cases when the crawler module cannot be used to obtain representative samples of the individuals appearing in the videos.
- A classifier training module that uses the face embeddings of the known individuals to train a multi-class SVM classifier for the recognition of these persons' faces.
- A recognition module which takes a video as input and extracts frames from it with a preset skipping distance d. For each extracted frame, faces are detected using the MTCNN algorithm and their FaceNet embeddings are computed. The SVM classifier then decides if the face matches any one among the training classes with a given confidence threshold t. In our experiments, we have used $d = 50$ and $t = 0.6$.
- Finally, we integrated a tracking algorithm as the last module. Simple Online and Realtime Tracking (SORT) is an object tracking algorithm which can track multiple objects in realtime [22] and its implementation is inspired by the code from Linzaer[1]. The algorithm uses the MTCNN bounding box detection and tracks the bounding boxes across frames. We introduced this module to increase the robustness of the library towards recognition errors in individual frames for getting more consistent person identifications.

[1] `https://github.com/Linzaer/Face-Track-Detect-Extract`

Program	Duration	# faces detected	# errors	Avg. confidence
A-Studio	41:58	9	3	0.80
Eurovaalit 2019, Part 2	1:35:41	1036	11	0.79
Eurovaalit 2019, Part 3	29:18	33	2	0.78

Table 1: Results of the face recognition on three programs: A-Studio broadcasted on 27/05/2019 at 21:00 on YLE TV1; Eurovaalit 2019: Kuka johtaa Eurooppaa? Parts 2 and 3, broadcasted on 15/05/2019 at 21:55 and 23:30 on YLE TV1.
The MeMAD Knowledge Graph identifiers of those parts are respectively `http://data.memad.eu/yle/a-studio/8a3a9588e0f58e1e40bfd30198274cb0ce27984e`, `http://data.memad.eu/yle/eurovaalit-2019-kuka-johtaa-eurooppaa/d9d05488b35db559cdef35bac95f518ee0dda76a` and `http://data.memad.eu/yle/eurovaalit-2019-kuka-johtaa-eurooppaa/0460c1b7d735e3fc796aa2829811aa1ae5dc9fa8`

In order to make the software publicly usable, we wrapped it with a Flask server and made it available as a service[2]. The service includes two output formats: a custom JSON format and a serialization format in RDF using the Turtle syntax, so that the results can be directly integrated in the MeMAD Knowledge Graph. The latter uses the Media Fragment URI syntax with *npt* in seconds for identifying temporal fragments and *xywh* for identifying the bounding box rectangle encompassing the face in the frame. A light cache system that enables to serve pre-computed results is also provided.

Evaluation. In the absence of a large and rigorously annotated ground truth for MeMAD broadcast videos, we performed two experiments for the evaluation of the system: a qualitative analysis on three videos and a quantitative analysis on a small ground truth dataset created for this purpose.

a. Qualitative analysis. We run an experiment using face models of the following nine people: Manfred Weber, Frans Timmermans, Jan Zahradil, Margrethe Vestager, Ska Keller, Nico Cué, Emilie Tran Nguyen, Markus Preiss, and Annastiina Heikkilä. These people were selected as they were key persons in the 2019 European Election and some of them were also frequently present during the 2014 European Election. These two events have been covered by both INA and YLE in the datasets they have provided to the MeMAD consortium. Hence, we were confident that those persons are likely to be shown in news programs and political debates that were broadcasted during the 2019 election period. For each detected person, we manually assessed whether the correct person was recognised or not. This enables us to evaluate the precision of the system but not the recall.

The results are presented in Table 1 together with the average confidence of all face recognition in each program. While the precision of the recognition varies, we observe it is very good for the Part 2 of the Eurovaalit (European Election) 2019 program.

b. Quantitative analysis. We developed a face recognition ground truth dataset from the INA videos which are part of the MeMAD video corpus. A list of six people (Nathanaël de Rincquesen, Elise Lucet, Sophie Le Saint, Laurent Delahousse, Sophie Gastrin, Marie Drucker) that are frequently present in the MeMaD Knowledge Graph's video segments was selected. For each of the 483 segments of duration n frames, we extracted the frames at positions $n/4$, $n/2$ and $3n/4$, which led to 355 video segments including one or more faces. From this set, we manually annotated a ground truth of 100 video segments, among which 55 segments

[2] `http://facerec.eurecom.fr/`

featured one of the six known people and 45 segments did not include any of the specified people.

A face recognition model trained on those six people was then evaluated on this ground truth dataset. We varied the confidence threshold under which we considered the face not matched as shown in Figure 2, and found $t = 0.6$ to be the optimal value with respect to the F-score. The details of each person class are reported in Table 2.

We made the following observations:

- The library generally fails to detect people when they are in the background and their faces are therefore relatively small.
- When faces are perfectly aligned in frontal orientation, they are easier to detect.
- We did not encounter cases where one known person was confused with another known person.
- Most errors occurred when an unknown face was recognised as one of the known people.

Our implementation relying on multiple off-the-shelf Python components has been made available at `https://github.com/D2KLab/Face-Celebrity-Recognition/`.

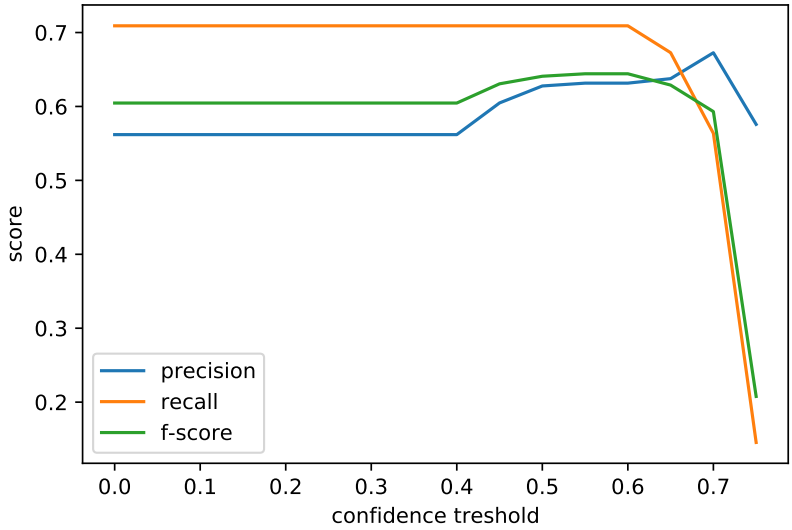


Figure 2: Precision, recall and F-score of the Face Recognition system on different confidence thresholds.

Person	Precision	Recall	F-score	Support
Le Saint, Sophie	0.67	0.80	0.73	10
Delahousse, Laurent	0.43	0.60	0.50	5
Lucet, Elise	0.71	0.50	0.59	10
Gastrin, Sophie	0.86	0.60	0.71	10
Rincquesen, Nathanaël de	0.43	0.80	0.55	10
Drucker, Marie	0.60	0.90	0.72	10
– unknown –	0.52	0.38	0.44	45
average excluding unknown	0.63	0.71	0.64	55

Table 2: Precision, recall and F-score for each class and aggregate results. The support column reports the number of segments involving the person.

3.2 Facial gender classification

Face gender information is a robust low-level descriptor which can be used to describe audiovisual content. It can be used to facilitate gender balance considerations during document search procedures, to monitor the gender balance of audiovisual productions, and to combine gender presence in media with user preferences.

The work realised at INA within the MeMAD project is fully described in Zohra Rezgui's Master's Thesis report [23] available online[3]. A face detection and tracking pipeline was realised, which makes it possible to detect faces, lower the computation time associated with face detection through tracking, and average gender classification predictions over tracked faces associated with the same character.

Three face datasets were used to train and evaluate the performance of face gender detection models realised in the work of MeMAD Task T2.2:

Labeled Faces in the Wild (LFW): A collection of 13,000 photos from the internet [24].

YouTube Faces Database (YTF): A collection of 621,126 frames obtained from YouTube videos. The images are organised based on the character identity (1595) and recording session (3425) [25]. All characters found in YTF are also present in LFW.

Adience: A collection of 26,580 photos obtained from the social media service Flickr, corresponding to 2284 distinct characters [26].

Since YTF was the only dataset obtained from video streams, it was considered as the most representative dataset with respect to the MeMAD use cases. During experimentation, we defined a subset of the YTF and LFW characters for training, and a different subset of characters for testing. This was done in order to avoid having the same character in the training and test sets, including the case of cross-corpora evaluation.

The first set of experiments was aimed at measuring the impact of the width of the face bounding boxes for the task of gender classification. We found that using larger bounding boxes benefited the gender classification task. Some facial features which are undesirable for identity detection (such as hair) located near the boundary of face bounding boxes provide better performance for the task of gender description. This finding was consistent with other studies found in the literature [27]. As a result, for optimal performance, it seems one should use different bounding box sizes and neural representations for the identity and the gender classification tasks. Best results were obtained in the gender classification task by using pretrained VGGFace neural models, combined with a linear SVM.

Table 3 presents a comparison of our implementation to two open source frameworks and results published in the literature. The face gender classification results obtained using INA's implementation were better than those obtained with available open source implementations for YTF and LFW. This is overall a positive outcome and shows that the implementation is close to the state of the art. Exhaustive comparison to results published in the literature was difficult, since most studies found were based on the use of a single corpus, which may lead to systems over-fitting to a given dataset (see Table 3 and appendices of [23]).

Lastly, we reproduced a study which involved fine-tuning of neural face representations originally aimed for face identification to the task of gender classification [31]. We obtained similar improvements on the dataset used for training, but these improvements resulted in a decrease of performance on datasets that were not used for training. Consequently, we considered this approach to over-fit with a particular dataset.

[3] https://www.researchgate.net/publication/337635267_Rapport_de_stage_Detection_et_classification_de_visages_pour_la_description_de_l'egalite_femme-homme_dans_les_archives_televisuelles

	YTF	LFW	Adience
INA's implementations			
trained on YTF	95.37	96.97	80.33
trained on Adience	89.45	92.00	80.26
Open source implementations			
CVLib	52.79	80.54	58.55
Scanner	88.05	93.28	95.15
Results published in research papers			
[28]	–	97.31	–
[29]	–	–	67.10
[30]	–	91.75	83.06

Table 3: Comparison of INA's face gender classification results to open source implementations and results found in research papers.

3.3 Single and multi-sentence video description

Image and video description entails automatically generating a short text or caption that describes the visual contents using only the image or video itself as the input. At AALTO and UH, the development of visual captioning techniques has been continued in three parallel tracks. First, we have studied and developed further neural network architectures for generating paragraph-length image captions (Section 3.3.1). Second, we have implemented and studied reinforcement learning based optimization methods for training image and video captioning models (Section 3.3.2). Third, we have addressed the task of visual storytelling where series of images are described with textual narratives as output (Section 3.3.3). In EURECOM, the development of video captioning has been based on utilizing curriculum learning as a method for enhancing the training of an embedding based captioning model (Section 3.3.4).

In addition to the research directly related to captioning, researchers at AALTO have contributed to the wider research area of image and video description with three conference papers on dense captioning [8], indoor scene recognition [9] and human-object interaction detection [10]. These works are attached in Appendix B of this report.

3.3.1 Paragraph-length image captioning

Arturs Polis' Master's Thesis [32] explores three variations of the encoder-decoder neural network architectures for generating paragraph-length image captions. These include both *flat* and *hierarchical* models, where the former consist of a single level of one recurrent neural network (RNN) and the latter of multiple levels of RNNs in a hierarchy. The benefit of using a hierarchy comes from the top-level RNN that is able to separately keep track of the sentence context.

The flat model studied was based on the original *Show and Tell* [33] architecture, the basic hierarchical model was the architecture proposed by Krause et al. [34], and the last one was the *Diverse-Coherent* hierarchical model from [35]. The coherent model extends the basic hierarchical model by adding a mechanism for allowing the gradients to flow directly from the last word of the previous generated sentence to the next sentence via a special coherence vector.

The experiments were carried out using image-level MS COCO [36] captions, region-based Visual Genome [37] captions, and paragraph-level captions from the recently introduced Stanford-Paragraph dataset [34]. DenseCap [38] and ResNet-152 [39] were used as visual features. The results of the experiments showed that with some modifications to the baseline flat model one could obtain results that exceeded the earlier reported flat paragraph captioning scores. The fluency of the output from the hierarchical-coherent model seemed to be

Figure 3: A video frame from the movie *500 Days of Summer*. The caption generated by AALTO's original captioning model is: *"A group of people standing around a kitchen counter."* The coherent hierarchical paragraph captioning model generated: *"People are sitting around a table in a restaurant. They are all dressed nicely. One of the women is wearing a dark shirt and pants. The other man is wearing a light green shirt with a short sleeve shirt. The glasses are on the table and filled with glass and filled wine. Most of the bottles are almost empty."*

somewhat higher than that of its flat counterpart, but this was not clearly captured in scores produced by the standard automatic metrics, such as METEOR [40] and CIDEr [41].

An example of the hierarchical-coherent paragraph captioning model being applied to movie content is shown in Figure 3 together with a caption generated with an earlier single-sentence model trained with MS COCO and TGIF [42] data. It can be seen that the generated captions read well and match the actual image content in this particular case quite well. Unfortunately that is not the case for all shots as the model often generates inaccurate captions similarly to our original single-sentence models.

The full details of the methods studied, experiments and their results are documented in Arturs Polis' Master's Thesis [32] available online[4]. Its abstract has additionally been attached to this report in Appendix B. All the methods studied for paragraph-length image captioning have been implemented and integrated in the *DeepCaption* Python library. In addition, the best-performing models studied for both standalone DeepCaption and PicSOM-integrated use setups have been made available.

3.3.2 Deep reinforcement video captioning

Héctor Laria Mantecón's Master's Thesis [43] studies the use of reinforcement learning for training image and video captioning models. Deep reinforcement learning based techniques have already been found to be very competitive training methods in many application areas of machine learning. In video captioning, for example in the TRECVID 2018 evaluation, the best results have been obtained with reinforcement learning [44].

Traditional cross-entropy loss based training for captioning models causes two major problems. First, the traditional approach inherently presents *exposure bias* because the model is only exposed to human-written descriptions, not to its own outputs, which causes an incremental error in test time. Second, the ultimate objective is not directly optimised because the scoring metrics cannot be used in the procedure as they are non-differentiable. New applications of reinforcement learning algorithms, such as self-critical training, overcome the exposure bias because they directly optimise non-differentiable sequence-based test metrics, such as CIDEr [41] and METEOR [40].

[4] https://helda.helsinki.fi/bitstream/handle/10138/304686/arturs_polis_thesis_final.pdf

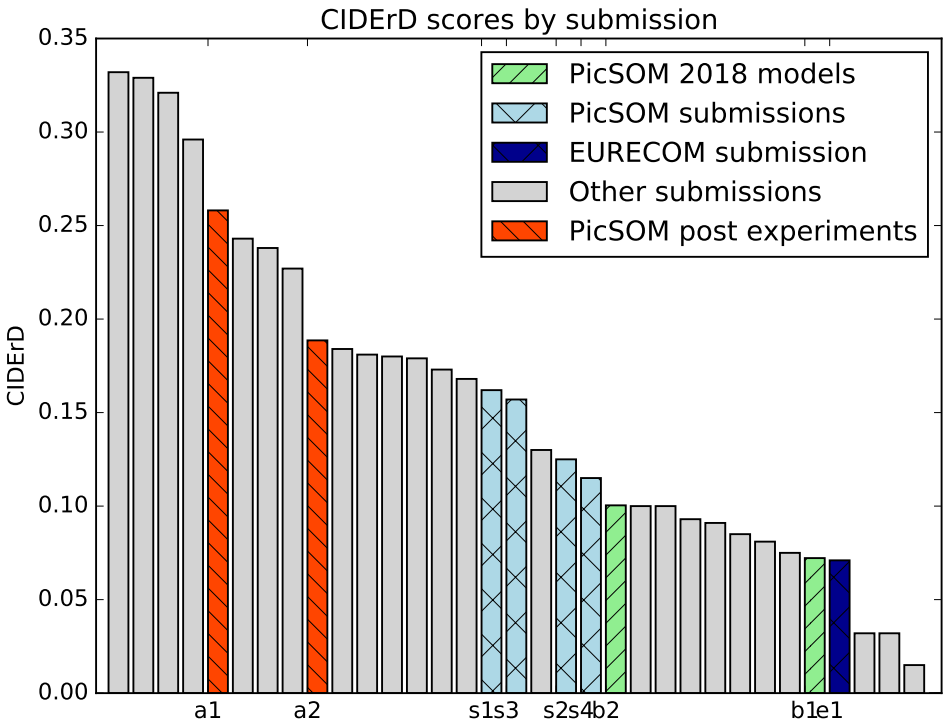


Figure 4: The performance of the PicSOM and EURECOM teams in TRECVID 2019 VTT task (blue bars) compared to the PicSOM team's best submission in 2018 ("b1") and best post 2018 workshop result ("b2"). Red results are PicSOM team's developments after TRECVID 2019. Higher CIDErD score is better.

The self-critical reinforcement learning approach studied in the thesis was applied when Aalto University's PicSOM team participated in the TRECVID 2019 Video to Text Description (VTT) task [1]. Thanks to self-critical training, we made substantial progress compared to both the submission of year 2018 and to the post-workshop experiments reported in our previous workshop paper. However, the other teams had likewise improved the performance of their approaches, and compared to the best level of performance obtained by the participating research groups, we were still behind the lead.

Figure 4 shows how the four submissions by the PicSOM team ("s1" to "s4") are clearly better in CIDErD scores than the results obtained with the best model of 2018 ("b1") and the best model studied after the 2018 workshop ("b2"). The submissions "s1" and "s4" differ in the fact that the former used also self-critical reinforcement learning, whereas the latter only used cross-entropy loss. Submission "s2" and "s3" differ from "s1" in their different selection of visual features used. The best result was obtained when both MS COCO and TGIF datasets and both video and still image features were used in training the captioning model. However, the benefit of using video features was found to be minor. In Figure 4, the red bars indicate two results obtained after the TRECVID 2019 workshop: "a2" with enhanced self-critical training and "a1" with the VATEX video captioning dataset [45] used as additional training data. We can see that with these two improvements the PicSOM team's captioning results approach the state of the art.

The full details of the methods studied, experiments and their results are documented in Héctor Larian Mantecón's Master's Thesis [43] available online[5]. Its abstract has additionally been attached to this report in Appendix B. The self-critical reinforcement learning technique has been integrated in the *DeepCaption* Python library, and is available in MeMAD's GitHub repository together with pretrained models for PicSOM-integrated use of the library.

[5] https://aaltodoc.aalto.fi/bitstream/handle/123456789/39942/master_Laria_Mantec%c3%b3n_H%c3%a9ctor_2019.pdf

3.3.3 Visual storytelling

The topic of Aditya Surikuchi's Master's Thesis [46] is *visual storytelling*. Given a sequence of images as input, the visual storytelling task is about building a model that can generate a coherent textual narrative as output. An image sequence would typically be a group of images portraying an event or an episode, and the output story could be up to fifty words long, with an average of ten words per image in the input sequence. This task has gained popularity as a research topic since the publication of the VIST dataset [47] and the first Visual Storytelling Challenge[6] in 2018.

Five visual storytelling models found in the research literature were either implemented from scratch or obtained as open source implementation. In addition, two models developed by the author were studied. The best results were obtained with the author's modified AREL model [48] which used Generative Adversarial Networks (GAN) [49] objective in its training.

Figure 5 displays one example of visual stories generated by the AREL GAN model. It can be seen that the five-sentence story generated for the five images in the sequence is able to describe the visual contents of the images and produce fluent narrative quite well. There is, however, unnecessary repetition in the story, but some of it can also be associated with the repetition in the visual contents of the images.

The full details of the methods studied, experiments and their results are documented in Aditya Surikuchi's Master's Thesis [46] available online[7]. Its abstract has additionally been attached to this report in Appendix B. The code for various visual storytelling models studied in the thesis has been made available in MeMAD's GitHub repository together with pretrained storytelling models.

Figure 5: A visual story generated by the AREL GAN model: *"There were a lot of people at the convention today. Everyone was there to support the event. The speaker gave a speech about the students. The speaker gave a speech. After the presentation, the speaker gave a speech to the audience."* [46]

3.3.4 Embedding-based captioning

After participating in the matching task of TRECVID VTT in 2018, EURECOM participated in the description generation task in TRECVID 2019. The method was developed by Danny Francis in his PhD Thesis [50] on image and video captioning. EURECOM submitted an embedding-based captioning run, using *curriculum learning* instead of a simple gradient descent to train the model.

The idea behind curriculum learning is to present data during the training in an ascending order of difficulty: the first epochs are based on easy samples, and after each epoch, more difficult samples are added to the training data. We computed a difficulty score for a given sample composed of a video and the corresponding caption as follows: the caption is translated into a list of indices so that the larger the index value is, the less frequent is the corresponding word. The difficulty score of the sample is then the maximum index value of its caption. Once the samples were scored, we trained the model by starting with an easy subset of the training data and added more complex samples after each epoch. Video features were extracted with the

[6] http://www.visionandlanguage.net/workshop2018/index.html#challenge
[7] https://aaltodoc.aalto.fi/bitstream/handle/123456789/41756/master_Surikuchi_Aditya_2019.pdf

I3D neural network [51], input to a fully-connected layer and then processed by a GRU [52] to generate the captions. Cross-entropy loss was used for training the model.

As seen in Figure 4, the performance of the EURECOM run ("e1") is below average. However, multiple ways to improve this performance can be explored, such as different scoring methods or finer curriculum learning algorithms. Moreover, more complex embedding-based approaches such as the spatio-temporal one developed in [7] and presented in Danny Francis' PhD Thesis [50] could be employed in future work.

4 Auditory domain

For the description of the audible content of media, several technological tools have been made available. The methods can be divided into detection and recognition of speech, speaker and language identification and recognition of other audio events, such as noise or music. In speech recognition the efforts in the first year of MeMAD were focused on accurate automatic transcription of relatively good quality broadcast speech that has a large vocabulary and multiple genres. Now in the second year, the focus has been on the end-to-end approach, which means that instead of separately trained acoustic, lexical and language models, there is only a single model. This is an important step towards the integration of multiple modalities into a single model. The speaker identification and diarisation provides useful information for both the video segmentation and speech transcripts. Here we have also focused on developing a deep neural network (DNN) based system where speakers are represented as embeddings. We have also shown that the same speaker embedding developed for speaker verification is an effective tool for adapting the speech recognizer for new speakers. During the second year of MeMAD, the audio event classifier was not developed further as we wanted to see first how it integrates with the visual features (see Section 5.1).

4.1 Speech and gender segmentation

INA has improved their *inaSpeechSegmenter* software to take also noise sound events into account, whereas the older implementations used to take only speech and music into account. The MUSAN database was used to gather examples of noise sound events [53].

In the work for MeMAD Task T2.2, *inaSpeechSegmenter* was used to process about one million hours of INA's audiovisual content, and to describe the evolution of French audiovisual landscape from the gender perspective. These analyses resulted in publications which met a very positive reception and visibility in the French media [3, 4].

4.2 Speech recognition

4.2.1 Creation of a domain-specific broadcast media evaluation set for speech recognition and diarisation for Finnish and Swedish

In 2019, Lingsoft has produced detailed transcripts of a challenging conversational multispeaker broadcast media dataset (YLE MeMAD broadcast dataset) to be used as a gold standard evaluation dataset of automatic speech recognition (ASR) and diarisation both in Finnish and Swedish in the MeMAD domain. The content is intentionally selected to be diverse to include multispeaker discussion programs, election panel discussions, interviews in different environments (moving car, outside) to measure the speech recognition result in actual use cases. Often the speakers are talking over each other and there is background noise and music, which all make the speech recognition task more difficult. The programs also contain a variety of topics. In the Finnish set, the topics range from a magazine program devoted to

consumer issues in which e-bikes and driving school are discussed, to talk show type current events discussion programs and European Parliament election debates. In the Swedish set, there are also current events talk shows, European Parliament election debates, and a craft and cooking show, each with a variety of different topics.

To our knowledge, such a domain specific material has not been available previously for Finnish or Swedish. Previously AALTO and Lingsoft have used an older YLE news broadcast dataset, which is generally easier: For example, there is only one speaker speaking at a time who articulates well (the news anchor) etc.

The main language of the Swedish programming in the test set is Finnish Swedish, the variant of Swedish spoken in Finland. This poses an additional challenge to the Swedish speech recognition models, developed with Swedish spoken in Sweden, as there is little training material available in the Finnish Swedish dialect. This test dataset also includes short sentences in other languages, such as Finnish or English.

The length of the Finnish test set is approximately 5 hours (7 different program items) and the test set contains 1345 sentences in total. The test data have been split into separate audio files at speaker changes, by the transcribers of the test sets. The length of the Swedish test set is approximately 5.5 hours (9 different program items) it contains 1466 sentences in the test set.

This evaluation dataset is available for the consortium for evaluation purposes. We are working towards opening the data for wider use: The annotations (e.g, time codes and annotations for speaker changes, music or background noise) can be more freely shared, but sharing the audiovisual material requires the permission from all the rights-holders (not limited to YLE), which requires a considerable effort. While sharing the annotations is more straightforward, they only have very limited use for the evaluation of ASR without the accompanying audio data. The MeMAD project has obtained a special permission from the all rights-holders of the audiovisual material for their use in the MeMAD project.

4.2.2 Finnish and Swedish ASR results on the challenging YLE test set

Lingsoft continues to provide its ASR in Finnish and Swedish for the use in the both via Lingsoft Speech Service API and integrated to the Limecraft Flow platform for the prototype. As the test set is currently limited only for the use of the MeMAD consortium, and no benchmarks as such exist, we tested the Lingsoft speech recognition against the commercially available Google ASR, which is generally thought to be of good quality. Table 4 summarises the comparison results, where the Lingsoft speech recognition shows clear improvements over Google ASR.

In the Finnish test set, the Lingsoft model generates an empty output for 23 sentences, whereas the Google benchmark model returns an empty output for 304 sentences. In addition, the Google recognizer did not recognise files that were longer than one minute, even when following the Google API instructions, hence there is a difference in the number of sentences tested between the Google ASR benchmark model and the Lingsoft recognizer. The difference should not affect the relative number of recognition errors, though. In Swedish, the number of sentences not recognised at all is 98 for Lingsoft and 520 for Google. For normalization purposes, numbers have been not spelled out but written as digits. shrunk (writing “6” instead

Recognizer	Number of sentences	Number of words	Word error rate
Lingsoft/Finnish	1323	36871	**25.6**
Google/Finnish	1041	33018	40.0
Lingsoft/Swedish	1368	48900	**41.2**
Google/Swedish	946	37530	54.6

Table 4: Word error rate results for benchmarking Lingsoft ASR (no RNNLM) with Google ASR.

of “six”, for example) in both speech recognition results, as Google’s speech recognition does not support the expanded form. In addition, punctuation and capitalization have been ignored in the comparison.

The ongoing development in 2019 of the Lingsoft Finnish ASR has included the accommodation of state-of-the-art neural network based acoustic modelling, improvements in the position dependency of phonemes, and recurrent neural network language modeling (RNNLM) [54] to rescore the first-pass decoded lattices.

The baseline Finnish and Swedish ASR systems have been evaluated against the improved systems with the YLE data described in more detail above. Word error rates (WERs) of the systems and development between the baseline and the current version are presented in Table 5. During 2019, small improvements have been achieved on top of major development in 2018. As with the results presented earlier, punctuation and case sensitivity have been ignored when computing the error rates.

Language	**Baseline**	**Improved (2019)**	**Improved + RNNLM**
Finnish	31.3	25.8	24.4
Swedish	56.0	41.2	–

Table 5: Lingsoft ASR: word error rate improvements in 2019

4.2.3 Speaker-aware training for end-to-end speech recognition

In recent years, a paradigm of speech recognition using a single model, which is optimised end-to-end, has become viable. Particularly encoder-decoder models with attention [55, 56] have been successful, although it seems that on standard academic benchmarks, conventional HMM-DNN architectures still prevail [57]. AALTO has also followed this line of research as it offers an attractively simple training scheme and joint optimisation of the whole system. Furthermore, in the context of creating multimodal models, end-to-end speech recognition models are a necessary starting point when building larger multimodal end-to-end systems. For example, they share a lot of similarities with machine translation architectures, making end-to-end speech translation models feasible.

Specifically, AALTO has worked on speaker-aware training of end-to-end speech recognition. In speaker-aware training, speaker embeddings are appended to the input features, and the model learns to use this information to adjust to, and thus be more robust to, differences between speakers. This work leveraged AALTO’s parallel work in speaker identification and diarisation.

The experiments that were conducted show that separately optimised speaker embeddings (so-called i-vectors and x-vectors) currently outperform a previously proposed [58] fully end-to-end sequence summary network method. Table 6 shows a subset of the results on the TED-LIUM corpus [59], which consists of TED-talks and is the basis for a lot of recent work in speech translation. Additionally, speaker embedding models trained on the large VoxCeleb [60, 61] corpora were used. These datasets are freely available. As the VoxCeleb training performs better than simply training embedding models on the fixed dataset, AALTO proposes speaker-aware training as a viable strategy to incorporate untranscribed data into the end-to-end ASR paradigm. A conference article describing the experiments was published in the ICASSP 2020 conference proceedings [62], and also included as Appendix B.6. The implementation of the experiments is available online[8].

[8] https://github.com/Gastron/espnet-old-speaker-aware

The TED-LIUM benchmark set-up was chosen here because the training and test data are all public and the domain of the data is not far from typical TV broadcast material. However, the size of the training data is not huge and thus the conventional hybrid DNN-HMMs are still much better than all the end-to-end systems that typically need thousands of hours to become comparable in performance. Unfortunately, there are no such huge public training data with a compatible domain. The focus of this piece of research work was not to improve the current state-of-the-art, but to pave the way to develop better methods for the future multimodal models that will most likely require end-to-end training. For that perspective, this experiment to compare methods that embed speaker information in ASR is indicative despite the size of the models and the training data.

	TED-LIUM	Test		Dev	
		No LM	+LM	No LM	+LM
Fixed	Baseline	21.7	18.6	22.6	20.0
	SeqSum [58]	21.1	–	21.7	–
	i-vector$_{100}$	**20.9**	**17.9**	**21.4**	**18.9**
	x-vector$_{256}$	21.5	18.4	23.0	20.0
+VoxCeleb	i-vector$_{200\text{-LDA}}$	20.2	17.4	**20.7**	18.2
	i-vector$_{400}$	20.4	**17.2**	21.0	18.3
	x-vector$_{200\text{-LDA}}$	20.9	17.4	21.6	18.6
	x-vector$_{512}$	**20.1**	**17.2**	20.9	**18.1**
	thin-ResNet$_{512}$	20.7	**17.2**	21.0	18.3

Table 6: WER results of AALTO's speech recognition experiments. SeqSum refers to the sequence summary network approach of Delcroix et al. The embedding methods denote the dimensionality and whether LDA was used, in the subscript. The +VoxCeleb section presents results with the pretrained VoxCeleb embeddings. The Fixed section presents results with embeddings trained on the fixed ASR data.

4.3 Speaker identification and diarisation

The task of speaker identification and diarisation is to divide the recordings into single-speaker segments and recognise the speakers. AALTO has continued to develop their deep learning model for overlapping speaker detection and online speaker diarisation[9]. The system consists of three components. First, the voice activity detector finds speech segments. Second, the speech segments are divided into two seconds wide overlapping windows and each window is classified either as single speaker or overlapping speakers. Finally, the speaker embeddings are computed for each single speaker window and utilised to recognise the speaker identities and speaker changes. The work has been documented in detail in Tuomas Kaseva's MSc thesis [63] available online[10]. The abstract of the thesis has additionally been attached to this report in Appendix B. The system was also evaluated in the VOXSRC speaker recognition challenge[11] where it reached a very respectable 11th position out of the over 50 entries that were submitted by the challenge deadline. AALTO's system was published in the ASRU 2019 conference proceedings [6] and the paper is included as Appendix B.7. The online speaker diarisation is mainly intended to annotate realtime speech recognition output with speaker change information, but it may also become useful in segmenting videos into moments or it can be used as an input to a video description system.

Lingsoft and LLS have developed a diarisation module for Finnish and Swedish. The diarisation module is accessible via the Lingsoft Speech Service API. The module is based on

[9] https://github.com/Livefull/SphereDiar
[10] https://aaltodoc.aalto.fi/bitstream/handle/123456789/39063/master_Kaseva_Tuomas_2019.pdf
[11] http://www.robots.ox.ac.uk/~vgg/data/voxceleb/competition.html

open source software and performs fast diarisation using a deep neural network that maps variable-length utterances to fixed-dimensional embeddings called *x-vectors* [64]. The module is language independent, but as diarisation results are of little use without the accompanying speech recognition result, the module incorporates a speech recognition result in Finnish or Swedish and the corresponding diarisation. In Table 7, we report diarisation error scores (DER) with and without Variational Bayes resegmentation, which improves results, but is also resource intensive, slowing down the diarisation process, and heavy on memory use. All results are obtained using the same YLE broadcast media dataset described earlier. Compared to the ASR evaluation, no splitting at speaker changes was made, but the diarisation evaluation has been carried out for each full program item (seven program items for Finnish, nine for Swedish). At this stage, no state-of-the-art comparison for Finnish or Swedish diarisation results has been made, as to our knowledge, other comparable datasets and results do not exist. The usefulness of the diarisation results will be evaluated along with the speech recognition in general user evaluation of Finnish and Swedish ASR in subtitling processes.

Language	no VB	with VB
Finnish	23.37	20.07
Swedish	27.23	29.34

Table 7: Diarisation Error Rates for Finnish and Swedish using the YLE media dataset with and without Variational Bayes resegmentation (VB).

5 Multimodal approaches

In addition to solely visual-based and solely audio-based approaches, MeMAD's Task T2.2 has introduced a number of multimodal approaches that combine the two modalities together in video content description that takes both information sources into account. That kind of multimodal methods will be needed used in MeMAD's Task T6.2 *Prototype implementation*. Four such techniques, *audio-enhanced video captioning*, *combined visual and auditory gender classification*, *person re-identification and re-referencing* and *multimodal ASR*, were found as promising approaches and they are described in the following sections.

5.1 Audio-enhanced video captioning

MeMAD's Deliverable D2.1 already included the *AudioTagger* software [65] which was used to describe the aural contents of audio or video files with 527 audio tags derived from the annotated AudioSet data by Google Research [66]. In Task T2.2 the audio tagging software has been integrated more tightly in the PicSOM framework and can now be used widely for audio feature extraction from videos and consequently also for content description including caption generation with the *DeepCaption* library.

In TRECVID 2018 Video to Text Description (VTT) task, none of the best teams used audio features. One of the main reasons for this was that the primary datasets for training the captioning models were MS COCO and TGIF, neither of which contain sound. However, in TRECVID 2019 VTT task, the winning team [44] had used also genuine video datasets, which have audio, for their model training. Most importantly, they used the new VATEX video captioning dataset [45] that contains over 41,250 videos and 825,000 captions in both English and Chinese.

In order to study the benefit offered by both the VATEX dataset and the *AudioTagger* features over our previously used "silent" MS COCO and TGIF datasets and only visual features, we

id	datasets	ResNet	audio	METEOR	CIDEr	CIDErD	BLEU
c1	COCO+TGIF	X		0.1996	0.2163	0.1043	0.0313
c2	COCO+TGIF+VATEX	X		0.2108	0.2244	0.1295	0.0346
c3	VATEX	X		0.2019	0.1770	0.1076	0.0301
c4	VATEX	X	X	0.2096	0.2226	0.1276	0.0355
c5	COCO+TGIF+VATEX	X	X	0.1978	0.1746	0.1025	0.0314

Table 8: Performance of captioning models in TRECVID 2019 VTT task when the used datasets and features were varied and cross-entropy loss used as the training objective. ResNet means concatenated ResNet-101 and ResNet-152 features.

performed a study whose results are comparable with those of the TRECVID 2019 VTT task. We trained a set of video captioning models similar to those we used in our TRECVID 2019 VTT submissions, which are described in Section 3.3.2 and in the TRECVID VTT notebook paper in Appendix B. In the models used in this study, we always used only the cross-entropy loss based training objective and, therefore, the results are mainly comparable to the PicSOM team's VTT submission "s4" in Figure 4.

The results of the experiment are shown in Table 8. As can be seen, all models used the concatenated ResNet-101 and Resnet-152 [39] visual features and the use of the *AudioTagger* audio features varied. Comparing the two lines identified as "c3" and "c4" it is evident that the use of the audio features clearly improved the quality of the generated captions with respect to all four automatic performance measures if only VATEX data was used in training. However, as can be seen by inspecting lines "c2" and "c5", there clearly was no benefit from the audio features of the VATEX videos if also the COCO and TGIF datasets were used.

Independent of the slightly contradictory findings above, the overall conclusion is that the VATEX data is a useful addition to the set of available captioning datasets for model training. We will definitely use it in our future works including those for MeMAD's Task T2.3.

5.2 Combined visual and auditory gender classification

Prospective work has been carried out in INA in order to design multimodal systems using speech and face gender classification modules described in Sections 3.2 and 4.1. The proposed approach takes advantage of *active speaker verification* procedures [67], aiming at predicting if the speech sound track of a video matches with the lip movements of one of the detected faces in the visual stream. This active speaker verification step allows us to know which face found in the video stream should be used and combined with the audio stream for multimodal gender classification. First experiments with positive results were carried out based on open source implementations, but were limited to cases where speakers' lip movements are visible, and aimed for maximal precision at low recall [67].

Several use-cases (fully automatic or involving human interaction) can be defined based on this strategy:

Automatic gender classification error estimation: The accuracy of gender classification in our large-scale studies is estimated on commonly used annotated corpora, which may not reflect the diversity of audiovisual broadcasts. Consequently, the error rates shown in our studies are dependent on specific corpora and may not reflect the performance of the systems on new materials. This may be particularly problematic on speech data, when a model trained on a specific language is used with a new language (French to Finnish for instance). An active speaker detection system can be used to estimate the classification error on each audiovisual document with a simple heuristic, such as the amount of divergent visual and audio classifications.

Semi-automatic database annotation: Active speaker detection can be used to process audiovisual streams to detect excerpts associated with different face and speech gender classifications. Such excerpts can be presented at a latter stage to annotators, allowing them to perform gender annotation campaigns based on examples that challenge the face or speech classification models.

Description of gender limit cases: Audiovisual excerpts associated with classification errors can be used in qualitative and quantitative studies aimed at describing gender limit cases based on acoustic features (prosody) or visual features (classification saliency maps).

Audiovisual gender classification: Research efforts may be devoted to combining acoustic and visual features and increasing the robustness of gender classification. Such systems will rely on unsupervised speaker segmentation procedures (diarisation) and active speaker detection. These will be worked on during the last stage of the MeMAD project.

5.3 Person re-identification and re-referencing

As a joint effort of EURECOM, Lingsoft, AALTO, INA and Limecraft, research has been carried out to merge the outputs of EURECOM's face recognition system (described in Section 3.1), Lingsoft's speaker diarisation (described in Section 4.3), INA's facial and aural gender recognition (described in Sections 3.2 and 4.1) and AALTO's video captioning (described in Section 3.3.2). The face and voice information are first combined together for associating people's faces and voices into multimodal person representations. After video content descriptions have been generated with the *DeepCaption* subsystem, the recognised persons' identities can be used to replace generic references to *a man* and *a woman* with their proper names in the captions. This stage has been implemented as a text-based postprocessing step for the captions.

As can be understood, this cannot be a fully automatic procedure as some metadata information about the names and genders of the people and the timecodes of their appearances in the footage are needed to initialise the procedure. However, facial example images can be collected and re-used for persons who repeatedly appear in news broadcasts or in different episodes of the same program series, which will expand the applicability of the approach.

An example of the results of person re-identification and re-referencing is shown in Figure 6. The voice of the female journalist has been identified by Lingsoft's speaker diarisation as "SPEAKER_6", but thanks to her visual appearances in the program, it has been possible to associate her real name, her voice and face together in one multimodal representation. In the following stages of the MeMAD project, we will collect a sample of reoccurring persons in INA's and YLE's TV broadcasts and study the accuracy of person re-identification across programs where their faces and/or voices occur.

Figure 6: An example of a re-identified and re-referred person in YLE's *Kuningaskuluttaja* program. The original caption: *"A man is sitting in a chair and smiling."* The modified caption: *"Marko Rajamäki is sitting in a chair and smiling while Maarit Åström-Kupsanen speaks."*

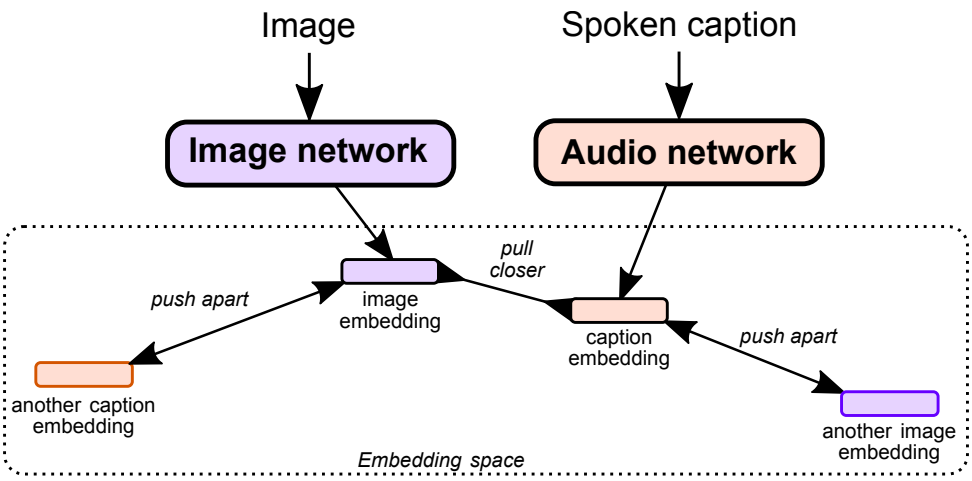


Figure 7: Training of the audiovisual embedding model. The model is encouraged to embed matching image-caption pairs close to each other while pushing dissimilar pairs apart.

5.4 Multimodal ASR

In multimodal automatic speech recognition, other input modalities, like video and images, are used alongside speech audio. AALTO has studied audiovisual embeddings introduced in [68] and further developed in [69] as a potential method for connecting audio and video inputs. The experiments in [69] show that the audiovisual embeddings form semantically meaningful clusters. For example, *sky* in an image is associated with the spoken words *blue* and *clouds*. The embeddings could thus provide a semantically meaningful features for a multimodal ASR system.

The task of the models in [68, 69] is to transform an image and its corresponding spoken caption into a shared embedding space. The models have two convolutional neural network (CNN) branches, one for audio and one for images. The image branch is a standard image classification network (VGG16) without the last classification layer. The audio branch is either a simpler [68] or more complex [69] CNN that takes a spectrogram as input. Both branches output a fixed-size vector that are measured for similarity in a loss function.

Figure 7 shows how the model is trained using triplet loss. With triplet loss the model is rewarded for embedding an image and its corresponding caption close to each other. At the same time, these embeddings are compared to the embeddings of one image and one caption randomly sampled from the training data. The model is rewarded for pushing both the random caption away from the original input image and the random image away from the original input caption.

AALTO has replicated the model and verified the results of [69] using code[12] and data[13] published by the authors and made an enhanced version of the software available[14]. In the original audiovisual retrieval task, the image corresponding to a given caption was among the top 10 most similar images 55.4 % of the time. The caption corresponding to an image was among the top 10 captions 46.4 % of the time. The comparable values in [69] are 60.4% and 52.8%. This is reasonably close to the performance reported in the original publication.

[12] https://github.com/dharwath/DAVEnet-pytorch
[13] https://groups.csail.mit.edu/sls/downloads/placesaudio/
[14] https://github.com/aalto-speech/avsr

6 Discussion

In this deliverable we have presented the work done and the current status of the multimodal media content analysis tools used or developed in the MeMAD project. The software libraries and tools have been made available to the project partners and a majority of them are also publicly available in GitHub. Most of the development tasks have been joint efforts of two or more MeMAD partners towards a common goal and the results could not have been obtained without the co-operation. For the rest of the project, the tight collaboration between the project partners will be even more essential.

In all cases when it has been possible, we have shown that the methods are on par with or close to the current state of the art. In many cases we have also been able to demonstrate the continuous improvement of our results while also the general state of the art has been progressing simultaneously. In particular, the visual domain and audio domain parts of the project have been supported with comparisons to the state of the art and we have shown good performances on a broad set of tasks. For the face recognition task, our innovation is in the application of state-of-the-art methods which combine MTCNN [14] and FaceNet [15] and their productisation via an API. We have also evaluated the performance of the detection and recognition of known faces on our own broadcast video data which are typically long videos as shown in Table 1. The facial gender classification method has been compared to the models with good results as reported in Table 3. The performance of our video captioning library has been continuously improving and is keeping up with the development of the state of the art as depicted in Figure 4. Speech recognition has been evaluated on a new and highly challenging YLE dataset gathered as a part of the MeMAD project. Along the evaluations, we have observed that our proposed models outperform the baseline models as seen in Table 5 and our Lingsoft ASR software outperforms the Google ASR in Finnish and Swedish as shown in Table 4.

More important than rigorous state-of-the-art comparisons, however, has been the application of the methods to the MeMAD project's own broadcast video materials provided by INA and YLE, and to SURREY's study corpus of movie clips. The importance of applying the methods to the "real" datasets instead of benchmarking data stems from the fact that many of the best-performing methods have been develóped and trained to work best for the particular benchmark's testing data and their performance on real-world data can realistically be expected to be worse. This behaviour results from the inevitable differences between the available large-scale training datasets and the MeMAD-specific testing datasets, and has been observed also in the experiments reported in this deliverable. To broadcasting companies such as YLE, getting to know the practical performance of the methods on their own broadcast programs is also valuable to help them in their future research and investments.

The real applicability and usefulness of the multimodal media content analysis tools studied in MeMAD's Work Package WP2 will be evaluated by the end of the project in Work Package WP6. Even if the automatic content analysis methods developed will not fully fulfill the expectations set for them in the beginning of the project, we already have and will still obtain very useful insight into the applicability of each of the analysis components alone and in combination with others. Some of these applicability issues can still be resolved during the MeMAD project, whereas others will remain to be solved in the future by the multimedia research community as a whole.

7 Summary of the MeMAD multimodal analysis software

name	st	provider	license	code	description
PicSOM	U	AALTO	Apache 2	C++	multimedia content analysis framework
DeepCaption	U	AALTO	Apache 2	Python3	image and video captioning
visual-storytelling	N	AALTO	Apache 2	Python3	visual storytelling
AALTO ASR	U	AALTO	MIT		speech recognition scripts using Kaldi
Speaker-aware training	N	AALTO	MIT	Python3	speaker-aware training of end-to-end ASR using Espnet
SphereDiar	N	AALTO	MIT	Python3	tools for overlapping speaker detection, speaker verification and speaker diarisation
avsr	N	AALTO	MIT	Python3	multimodal ASR
AudioTagger	U	AALTO	Apache 2	Python3	audio event classification
OpenNMT-py	O	AALTO	MIT	Python3	multi-modal image caption translation for WP4
statistical-tools	O	AALTO	MIT	Python3	tools for creating dataset statistics for WP5
Face-Celebrity-Recognition	U	EURECOM	Apache 2	Python3	tools for detecting, aligning and recognising faces in video
inaSpeechSegmenter	U	INA	MIT	Python3	speech, music and noise segmentation; speaker gender detection
inaFaceGender	N	INA	proprietary	Python3	face detection, tracking and gender classification
Flow Shot Cut Detector	O	Limecraft	proprietary	C	subprogram of broadcast video production system
Lingsoft Speech Service	U	Lingsoft	proprietary	Python3, C++, JavaScript	automatic speech recognition service via an API

Table 9: Software components of MeMAD related to multimodal content analysis. Column "st" shows the status symbols standing for "O" = old version of MeMAD D2.1, "U" = updated version from D2.1 to D2.2, and "N" = new component in D2.2.

Table 9 contains a summary of the software components used in the MeMAD project for multimodal content analysis and available to the project members. Software components that have proprietary licenses are available for the MeMAD partners as software or as a service. Those that have been identified to have a liberal licensing scheme, such as MIT or Apache 2, are publicly available as source code in MeMAD's GitHub page located at:

`https://github.com/MeMAD-project`

The liberally licensed software components discussed in this report have been specifically collected for ease of installation in a repository named `mmca`:

`https://github.com/MeMAD-project/mmca`

Some of the modules in `mmca` are physically located outside of the MeMAD GitHub project, but the Git submodule mechanism facilitates their seamless availability from their true locations. All of the software packages can be obtained with a single operation:

```
git clone https://github.com/MeMAD-project/mmca.git --recursive
```

Each of the subdirectories created inside the `mmca` directory contains its own further installation and use instructions. Specifically, each package will have up-to-date instructions for installation and usage in a file called `README.md` in the corresponding directory. The licensing information of each submodule is available in a file named `LICENSE`.

8 References

[1] Héctor Laria Mantecón, Jorma Laaksonen, Danny Francis, and Benoit Huet. PicSOM and EURECOM experiments in TRECVID 2019. In Proceedings of the TRECVID 2019 Workshop, Gaithersburg, MD, USA, November 2019.

[2] Danny Francis, Phuong Anh Nguyen, Benoit Huet, and Chong-Wah Ngo. EURECOM at TRECVid AVS 2019. In Proceedings of the TRECVID 2019 Workshop, Gaithersburg, MD, USA, November 2019.

[3] David Doukhan, Géraldine Poels, Zohra Rezgui, and Jean Carrive. Describing gender equality in french audiovisual streams with a deep learning approach. VIEW Journal of European Television History and Culture, 7(14), 2018.

[4] David Doukhan. A la radio et à la télé, les femmes parlent deux fois moins que les hommes. La revue des Médias Femmes dans les médias: rôles de dames-épisode, 2(8), 2019.

[5] Aku Rouhe, Tuomas Kaseva, and Mikko Kurimo. Speaker-aware training of attention-based end-to-end speech recognition using neural speaker embeddings. In Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing (ICASSP), 2020.

[6] Tuomas Kaseva, Aku Rouhe, and Mikko Kurimo. Spherediar - an efficient speaker diarization system for meeting data. In 2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU), 2019.

[7] Danny Francis and Benoit Huet. L-stap: Learned spatio-temporal adaptive pooling for video captioning. In Proceedings of the 1st International Workshop on AI for Smart TV Content Production, Access and Delivery, pages 33–41. ACM, 2019.

[8] Tzu-Jui Julius Wang, Hamed Rezazadegan Tavakoli, Mats Sjöberg, and Jorma Laaksonen. Geometry-aware relational exemplar attention for dense captioning. In Proceedings of the 1st International Workshop on Multimodal Understanding and Learning for Embodied Applications (MULEA '19) in ACM Multimedia Conference, pages 3–11, Nice, France, October 2019.

[9] Rao Muhammad Anwer, Fahad Shahbaz Khan, Jorma Laaksonen, and Nazar Zaki. Multi-stream convolutional networks for indoor scene recognition. In Proceedings of the 18th International Conference on Computer Analysis of Images and Patterns (CAIP2019), pages 196–208, Salerno, Italy, September 2019.

[10] Tiancai Wang, Rao Muhammad Anwer, Muhammad Haris Khan, Fahad Shahbaz Khan, Yanwei Pang, Ling Shao, and Jorma Laaksonen. Deep contextual attention for human-object interaction detection. In Proceedings of the International Conference on Computer Vision (ICCV2019), pages 5694–5702, Seoul, Korea, October 2019.

[11] Paul Viola and Michael J. Jones. Robust real-time face detection. International Journal of Computer Vision, 57(2):137–154, May 2004.

[12] Timo Ahonen, Abdenour Hadid, and Matti Pietikäinen. Face description with local binary patterns: Application to face recognition. IEEE Transactions on Pattern Analysis & Machine Intelligence, 28(12):2037–2041, 2006.

[13] Davis E. King. Dlib-ml: A machine learning toolkit. Journal of Machine Learning Research, 10:1755–1758, 2009.

[14] Kaipeng Zhang, Zhanpeng Zhang, Zhifeng Li, and Yu Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. IEEE Signal Processing Letters, 23(10):1499–1503, 2016.

[15] Florian Schroff, Dmitry Kalenichenko, and James Philbin. Facenet: A unified embedding for face recognition and clustering. In Proceedings of the IEEE conference on computer vision and pattern recognition, pages 815–823, 2015.

[16] Ivan William, De Rosal Ignatius Moses Setiadi, Eko Hari Rachmawanto, Heru Agus Santoso, and Christy Atika Sari. Face Recognition using FaceNet (Survey, Performance Test, and Comparison). In 4^{th} International Conference on Informatics and Computing (ICIC), pages 1–6, 2019.

[17] Guodong Guo and Na Zhang. A survey on deep learning based face recognition. Computer Vision and Image Understanding, 189:102805, 2019.

[18] Mohammad Shafin, Rojina Hansda, Ekta Pallavi, Deo Kumar, Sumanta Bhattacharyya, and Sanjeev Kumar. Partial Face Recognition: A Survey. In 3^{rd} International Conference on Advanced Informatics for Computing Research, ICAICR '19, New York, NY, USA, 2019. Association for Computing Machinery.

[19] Adamu Ali-Gombe, Eyad Elyan, and Johan Zwiegelaar. Towards a Reliable Face Recognition System. In Lazaros Iliadis, Plamen Parvanov Angelov, Chrisina Jayne, and Elias Pimenidis, editors, 21^{st} Engineering Applications of Neural Networks Conference (EANN), pages 304–316, Cham, 2020. Springer International Publishing.

[20] Shan Li and Weihong Deng. Deep Facial Expression Recognition: A Survey. IEEE Transactions on Affective Computing, pages 1–1, 2020.

[21] Qiong Cao, Li Shen, Weidi Xie, Omkar M Parkhi, and Andrew Zisserman. Vggface2: A dataset for recognising faces across pose and age. In 2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018), pages 67–74. IEEE, 2018.

[22] Alex Bewley, Zongyuan Ge, Lionel Ott, Fabio Ramos, and Ben Upcroft. Simple online and realtime tracking. In 2016 IEEE International Conference on Image Processing (ICIP), pages 3464–3468. IEEE, 2016.

[23] Zohra Rezgui. Détection et classification de visages pour la description de l'égalité femme-homme dans les archives télévisuelles. Master's thesis, Université de Carthage – Ecole Supérieure de la Statistique et de l'Analyse de l'Information, Tunisie, 2019.

[24] Gary B. Huang, Manu Ramesh, Tamara Berg, and Erik Learned-Miller. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. Technical Report 07-49, University of Massachusetts, Amherst, October 2007.

[25] Lior Wolf, Tal Hassner, and Itay Maoz. Face recognition in unconstrained videos with matched background similarity. IEEE, 2011.

[26] Eran Eidinger, Roee Enbar, and Tal Hassner. Age and gender estimation of unfiltered faces. IEEE Transactions on Information Forensics and Security, 9(12):2170–2179, 2014.

[27] Sen Jia, Thomas Lansdall-Welfare, and Nello Cristianini. Gender classification by deep learning on millions of weakly labelled images. In 2016 IEEE 16th International Conference on Data Mining Workshops (ICDMW), pages 462–467. IEEE, 2016.

[28] Grigory Antipov, Sid-Ahmed Berrani, and Jean-Luc Dugelay. Minimalistic cnn-based ensemble model for gender prediction from face images. Pattern recognition letters, 70:59–65, 2016.

[29] Jos van de Wolfshaar, Mahir F Karaaba, and Marco A Wiering. Deep convolutional neural networks and support vector machines for gender recognition. In 2015 IEEE Symposium Series on Computational Intelligence, pages 188–195. IEEE, 2015.

[30] Mahmoud Afifi and Abdelrahman Abdelhamed. Afif4: deep gender classification based on adaboost-based fusion of isolated facial features and foggy faces. Journal of Visual Communication and Image Representation, 62:77–86, 2019.

[31] Gokhan Ozbulak, Yusuf Aytar, and Hazim Kemal Ekenel. How transferable are cnn-based features for age and gender classification? In 2016 International Conference of the Biometrics Special Interest Group (BIOSIG), pages 1–6. IEEE, 2016.

[32] Arturs Polis. Paragraph-length image captioning using hierarchical recurrent neural networks. Master's thesis, University of Helsinki, 2019.

[33] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. Show and tell: A neural image caption generator. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), June 2015.

[34] Jonathan Krause, Justin Johnson, Ranjay Krishna, and Li Fei-Fei. A hierarchical approach for generating descriptive image paragraphs. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 3337–3345. IEEE, 2017.

[35] Moitreya Chatterjee and Alexander G Schwing. Diverse and coherent paragraph generation from images. In Proceedings of the European Conference on Computer Vision (ECCV), pages 729–744, 2018.

[36] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common objects in context. In European Conference on Computer Vision (ECCV), 2014.

[37] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A. Shamma, Michael S. Bernstein, and Li Fei-Fei. Visual genome: Connecting language and vision using crowdsourced dense image annotations. International Journal of Computer Vision, 123(1):32–73, May 2017.

[38] Justin Johnson, Andrej Karpathy, and Li Fei-Fei. DenseCap: Fully convolutional localization networks for dense captioning. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 4565–4574, 2016.

[39] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 770–778, 2016.

[40] Satanjeev Banerjee and Alon Lavie. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In Proceedings of the ACL Workshop on Intrinsic and Extrinsic Evaluation Measures for Machine Translation and/or Summarization, pages 65–72, 2005.

[41] Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), pages 4566–4575, 2015.

[42] Yuncheng Li, Yale Song, Liangliang Cao, Joel Tetreault, Larry Goldberg, Alejandro Jaimes, and Jiebo Luo. TGIF: A new dataset and benchmark on animated gif description. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pages 4641–4650, 2016.

[43] Héctor Laria Mantecón. Deep reinforcement sequence learning for visual captioning. Master's thesis, Aalto University, 2019.

[44] Yuqing Song, Yida Zhao, Shizhe Chen, and Qin Jin. RUC_AIM3 at TRECVID 2019: Video to Text. In Proceedings of the TRECVID 2019 Workshop, Gaithersburg, MD, USA, November 2019.

[45] Xin Wang, Jiawei Wu, Junkun Chen, Lei Li, Yuan-Fang Wang, and William Yang Wang. Vatex: A large-scale, high-quality multilingual dataset for video-and-language research, 2019.

[46] Aditya Surikuchi. Visual storytelling: Captioning of image sequences. Master's thesis, Aalto University, 2019.

[47] Ting-Hao, Huang, Francis Ferraro, Nasrin Mostafazadeh, Ishan Misra, Aishwarya Agrawal, Jacob Devlin, Ross Girshick, Xiaodong He, Pushmeet Kohli, Dhruv Batra, C. Lawrence Zitnick, Devi Parikh, Lucy Vanderwende, Michel Galley, and Margaret Mitchell. Visual storytelling, 2016.

[48] Xin Wang, Wenhu Chen, Yuan-Fang Wang, and William Yang Wang. No metrics are perfect: Adversarial reward learning for visual storytelling, 2018.

[49] Ian J. Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial networks, 2014.

[50] Danny Francis. Semantic Representations of Images and Videos. PhD thesis, EURECOM, 2019.

[51] João Carreira and Andrew Zisserman. Quo vadis, action recognition? A new model and the kinetics dataset. CoRR, abs/1705.07750, 2017.

[52] Kyunghyun Cho, Bart van Merrienboer, Çaglar Gülçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL, pages 1724–1734, 2014.

[53] David Snyder, Guoguo Chen, and Daniel Povey. Musan: A music, speech, and noise corpus. arXiv preprint arXiv:1510.08484, 2015.

[54] Tomáš Mikolov, Martin Karafiát, Lukáš Burget, Jan Černocký, and Sanjeev Khudanpur. Recurrent neural network based language model. In Proceedings of the 11th Annual Conference of the International Speech Communication Association (INTERSPEECH 2010), 2010.

[55] D. Bahdanau, J. Chorowski, D. Serdyuk, P. Brakel, and Y. Bengio. End-to-end attention-based large vocabulary speech recognition. In 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 4945–4949, March 2016.

[56] W. Chan, N. Jaitly, Q. Le, and O. Vinyals. Listen, attend and spell: A neural network for large vocabulary conversational speech recognition. In 2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), pages 4960–4964, March 2016.

[57] Christoph Lüscher, Eugen Beck, Kazuki Irie, Markus Kitza, Wilfried Michel, Albert Zeyer, Ralf Schlüter, and Hermann Ney. RWTH ASR systems for LibriSpeech: Hybrid vs attention. In Proc. Interspeech 2019, pages 231–235, 2019.

[58] Marc Delcroix, Shinji Watanabe, Atsunori Ogawa, Shigeki Karita, and Tomohiro Nakatani. Auxiliary feature based adaptation of end-to-end asr systems. In Proc. Interspeech 2018, pages 2444–2448, 2018.

[59] Anthony Rousseau, Paul Deléglise, and Yannick Esteve. Enhancing the ted-lium corpus with selected data for language modeling and more ted talks. In LREC, pages 3935–3939, 2014.

[60] A. Nagrani, J. S. Chung, and A. Zisserman. Voxceleb: a large-scale speaker identification dataset. In INTERSPEECH, 2017.

[61] J. S. Chung, A. Nagrani, and A. Zisserman. Voxceleb2: Deep speaker recognition. In INTERSPEECH, 2018.

[62] Aku Rouhe, Tuomas Kaseva, and Mikko Kurimo. Speaker-aware training of attention-based end-to-end speech recognition using neural speaker embeddings. In Proceeding of International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2020.

[63] Tuomas Kaseva. Spherediar – an efficient speaker diarization system for meeting data. Master's thesis, Aalto University, 2019.

[64] David Snyder, Daniel Garcia-Romero, Gregory Sell, Daniel Povey, and Sanjeev Khudanpur. X-vectors: Robust DNN embeddings for speaker recognition. In Proceeding of International Conference on Acoustics, Speech and Signal Processing (ICASSP), 2018.

[65] Zhicun Xu Peter Smit and Mikko Kurimo. The Aalto system based on fine-tuned audioset features for DCASE2018 task2 - general purpose audio tagging. In Proceedings of the Detection and Classification of Acoustic Scenes and Events 2018 Workshop (DCASE2018), Surrey, UK, November 2018.

[66] Jort F Gemmeke, Daniel PW Ellis, Dylan Freedman, Aren Jansen, Wade Lawrence, R Channing Moore, Manoj Plakal, and Marvin Ritter. Audio set: An ontology and human-labeled dataset for audio events. In Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on, pages 776–780. IEEE, 2017.

[67] Joon Son Chung and Andrew Zisserman. Out of time: automated lip sync in the wild. In Asian conference on computer vision, pages 251–263. Springer, 2016.

[68] David Harwath, Antonio Torralba, and James Glass. Unsupervised learning of spoken language with visual context. In Advances in Neural Information Processing Systems 29, pages 1858–1866, 2016.

[69] David Harwath, Adria Recasens, Didac Suris, Galen Chuang, Antonio Torralba, and James Glass. Jointly discovering visual objects and spoken words from raw sensory input. In The European Conference on Computer Vision (ECCV), 2018.

A Dissemination activities

- **Conference presentation** 4.9.2019: CAIP 2019: *The 18th International Conference on Computer Analysis of Images and Patterns*, Salerno, Italy. Rao Muhammad Anwer presented *Multi-stream Convolutional Networks for Indoor Scene Recognition*.
- **Workshop organisation** 21.10.2019: AI4TV 2019: *1st International Workshop on AI for Smart TV Content Production, Access and Delivery*, a workshop in ACM International Conference on Multimedia, Nice, France. Raphaël Troncy and Jorma Laaksonen chaired the workshop.
- **Workshop presentation** 21.10.2019: AI4TV 2019: *1st International Workshop on AI for Smart TV Content Production, Access and Delivery*, Nice, France. Danny Francis presented *L-STAP: Learned Spatio-Temporal Adaptive Pooling for Video Captioning*.
- **Keynote** 22.10.2019: ACMMM19: *ACM Multimedia 2019 Conference*, Nice, France. Jean Carrive presented the introducing keynote *Using Artificial Intelligence to Preserve Audiovisual Archives: New Horizons, More Questions*.
- **Workshop presentation** 25.10.2019: MULEA '19: *1st International Workshop on Multimodal Understanding and Learning for Embodied Applications*, Nice, France. Zhu-Jui Wang presented *Geometry-aware Relational Exemplar Attention for Dense Captioning*.
- **Conference presentation** 31.10.2019: ICCV 2019: *International Conference on Computer Vision*, Seoul, Korea. Rao Anwer presented *Deep Contextual Attention for Human-Object Interaction Detection*.
- **Workshop presentation** 12–13.11.2019: TRECVID 2019: *TREC Video Retrieval Evaluation*, Gaithersburg, USA. Jorma Laaksonen presented an invited talk titled *Image Data, Video Data and Both in VTT Model Training*.
- **Workshop presentation** 28.11.2019: AIDI: AI in distribution and production, Manchester, UK. David Doukhan presented an invited talk titled *Describing gender representation in French TV and radio with AI*.
- **Conference presentation** 22-25.10.2019: FIAT/IFTA World Conference, Dubrovnik, Croatia. David Doukhan presented an invited talk titled *Artificial intelligence to measure gender imbalances over 700,000 hours of media*.
- **Workshop presentation** 13.12.2019: Corpus Workshop at BnF: Jean Carrive presented *New Analysis Methods for Audiovisual Media: ANTRACT and MeMAD projects*, Collect, Preserve, Explore Massive Audiovisual Corpora Workshop, National Library of France (BnF)

B Appendices

B.1 Abstracts of Master's and PhD Theses

The following pages contain abstracts of the Master's and PhD Theses whose full contents can be accessed through the links below:

- Arturs Polis: *Paragraph-length image captioning using hierarchical recurrent neural networks*. Master's Thesis, University of Helsinki, 2019. [32]
- Héctor Laria Mantecón: *Deep Reinforcement Sequence Learning for Visual Captioning*. Master's Thesis, Aalto University, 2019. [43]
- Aditya Surikuchi: *Visual Storytelling: Captioning of Image Sequences*. Master's Thesis, Aalto University, 2019. [46]
- Tuomas Kaseva: *SphereDiar – an efficient speaker diarization system for meeting data.* Master's Thesis, Aalto University, 2019. [63]
- Danny Francis: *Semantic Representations of Images and Videos.* PhD Thesis, Sorbonne University-EURECOM, 2019. [50]

HELSINGIN YLIOPISTO — HELSINGFORS UNIVERSITET — UNIVERSITY OF HELSINKI

Tiedekunta — Fakultet — Faculty	Koulutusohjelma — Utbildningsprogram — Degree programme
Faculty of Science	Master's Programme in Data Science

Tekijä — Författare — Author

Arturs Polis

Työn nimi — Arbetets titel — Title

Paragraph-length image captioning using hierarchical recurrent neural networks

Työn laji — Arbetets art — Level	Aika — Datum — Month and year	Sivumäärä — Sidantal — Number of pages
Master's thesis	March 29, 2019	83

Tiivistelmä — Referat — Abstract

Recently, a neural network based approach to automatic generation of image descriptions has become popular. Originally introduced as *neural image captioning*, it refers to a family of models where several neural network components are connected end-to-end to infer the most likely caption given an input image. Neural image captioning models usually comprise a Convolutional Neural Network (CNN) based image encoder and a Recurrent Neural Network (RNN) language model for generating image captions based on the output of the CNN.

Generating long image captions – commonly referred to as *paragraph captions* – is more challenging than producing shorter, sentence-length captions. When generating paragraph captions, the model has more degrees of freedom, due to a larger total number of combinations of possible sentences that can be produced. In this thesis, we describe a combination of two approaches to improve paragraph captioning: using a hierarchical RNN model that adds a top-level RNN to keep track of the sentence context, and using richer visual features obtained from dense captioning networks. In addition to the standard MS-COCO Captions dataset used for image captioning, we also utilize the Stanford-Paragraph dataset specifically designed for paragraph captioning.

This thesis describes experiments performed on three variants of RNNs for generating paragraph captions. The *flat* model uses a non-hierarchical RNN, the *hierarchical* model implements a two-level, hierarchical RNN, and the *hierarchical-coherent* model improves the *hierarchical* model by optimizing the coherence between sentences.

In the experiments, the *flat* model outperforms the published non-hierarchical baseline and reaches similar results to our *hierarchical* model. The *hierarchical* model performs similarly to the corresponding published model, thus validating it. The *hierarchical-coherent* model gives us inconclusive results – it outperforms our *hierarchical* model but does not reach the same scores as the corresponding published model.

With our *flat* model implementation, we have shown that with minor improvements to a simple image captioning model, one can obtain much higher scores on standard metrics than previously reported. However, it is yet unclear whether a hierarchical RNN is required to model the paragraph captions, or whether a single RNN layer on its own can be powerful enough. Our initial human evaluation indicates that the captions produced by a hierarchical RNN may in fact be more fluent, however the standard automatic evaluation metrics do not capture this.

Avainsanat — Nyckelord — Keywords

neural networks, image captioning, paragraph captioning, hierarchical RNN

Säilytyspaikka — Förvaringsställe — Where deposited

Muita tietoja — Övriga uppgifter — Additional information

Aalto University
School of Science

Aalto University
School of Science
Master's Programme in Computer, Communication and Information Sciences

ABSTRACT OF MASTER'S THESIS

Author:	Héctor Laria Mantecón		
Title:	Deep Reinforcement Sequence Learning for Visual Captioning		
Date:	August 8, 2019	**Pages:**	77
Major:	Machine Learning, Data Science and Artificial Intelligence	**Code:**	SCI3044
Supervisor:	Docent Jorma Laaksonen		
Advisor:	Docent Jorma Laaksonen		

Methods to describe an image or video with natural language, namely image and video captioning, have recently converged into an encoder-decoder architecture. The encoder here is a deep convolutional neural network (CNN) that learns a fixed-length representation of the input image, and the decoder is a recurrent neural network (RNN), initialised with this representation, that generates a description of the scene in natural language.

Traditional training mechanisms for this architecture usually optimise models using cross-entropy loss, which experiences two major problems. First, it inherently presents exposure bias (the model is only exposed to real descriptions, not to its own words), causing an incremental error in test time. Second, the ultimate objective is not directly optimised because the scoring metrics cannot be used in the procedure, as they are non-differentiable. New applications of reinforcement learning algorithms, such as self-critical training, overcome the exposure bias, while directly optimising non-differentiable sequence-based test metrics.

This thesis reviews and analyses the performance of these different optimisation algorithms. Experiments on self-critic loss denote the importance of robust metrics against gaming to be used as the reward for the model, otherwise the qualitative performance is completely undermined. Sorting that out, the results do not reflect a huge quality improvement, but rather the expressiveness worsens and the vocabulary moves closer to what the reference uses.

Subsequent experiments with a greatly improved encoder result in a marginal enhancing of the overall results, suggesting that the policy obtained is shown to be heavily constrained by the decoder language model. The thesis concludes that further analysis with higher capacity language models needs to be performed.

Keywords:	deep learning, machine learning, neural networks, reinforcement learning, policy gradient, reinforce, self critic, captioning, description generation, computer vision
Language:	English

Aalto University
School of Science
Master's Programme in Computer, Communication and Information Sciences

ABSTRACT OF MASTER'S THESIS

Author:	Aditya Surikuchi		
Title:	Visual Storytelling: Captioning of Image Sequences		
Date:	November 25, 2019	**Pages:**	78
Major:	Machine Learning, Data Science and Artificial Intelligence	**Code:**	SCI3044
Supervisor:	Jorma Laaksonen D.Sc. (Tech.), Aalto University		
Advisor:	Jorma Laaksonen D.Sc. (Tech.), Aalto University		

In the space of automated captioning, the task of visual storytelling is one dimension. Given sequences of images as inputs, visual storytelling (VIST) is about automatically generating textual narratives as outputs. Automatically producing stories for an order of pictures or video frames have several potential applications in diverse domains ranging from multimedia consumption to autonomous systems. The task has evolved over recent years and is moving into adolescence. The availability of a dedicated VIST dataset for the task has mainstreamed research for visual storytelling and related sub-tasks.

This thesis work systematically reports the developments of standard captioning as a parent task with accompanying facets such as dense captioning, and gradually delves into the domain of visual storytelling. Existing models proposed for VIST are described by examining respective characteristics and scope. All the methods for VIST adapt from the typical encoder-decoder style design, owing to its success in addressing the standard image captioning task. Several subtle differences in the underlying intentions of these methods for approaching the VIST are subsequently summarized.

Additionally, alternate perspectives around the existing approaches are explored by re-modeling and modifying their learning mechanisms. Experiments with different objective functions are reported with subjective comparisons and relevant results. Eventually, the sub-field of character relationships within storytelling is studied and a novel idea called character-centric storytelling is proposed to account for prospective characters in the extent of data modalities.

Keywords:	captioning, visual storytelling, sequence modeling, natural language processing, computer vision, semantic relationships, deep reinforcement learning
Language:	English

Aalto University
School of Electrical
Engineering

Aalto University, P.O. BOX 11000, 00076 AALTO
www.aalto.fi
Abstract of the master's thesis

Author Tuomas Kaseva

Title SphereDiar - an efficient speaker diarization system for meeting data

Degree programme Computer, Communication and Information Sciences

Major Signal, Speech and Language Processing **Code of major** ELEC3031

Supervisor Prof. Mikko Kurimo

Advisor M.Sc. Aku Rouhe

Date 27.5.2019 **Number of pages** 2 **Language** English

Abstract

The objective of speaker diarization is to determine who spoke and when in a given audio stream. This information is useful in multiple different speech related tasks such as speech recognition, automatic creation of rich transcriptions and text-to-speech synthesis. Moreover, speaker diarization can also play a central role in the creation and organization of speech-related datasets.

Speaker diarization is made difficult by the immense variability in speakers and recording conditions, and the unpredictable and overlapping speaker turns of spontaneous discussion. Especially diarization of meeting data has been very challenging. Even the most advanced speaker diarization systems still struggle with this type of data.

In this thesis, a novel speaker diarization system, named SphereDiar and designed for the diarization of meeting data, is proposed. This system combines three novel subsystems: the SphereSpeaker neural network for speaker modeling, a segmentation method named Homogeneity Based Segmentation and a clustering algorithm Top Two Silhouettes. The system harnesses up-to-date deep learning approaches for speaker diarization and addresses the problem of overlapping speech in this task.

Experiments are performed on a dataset consisting of over 200 meetings. The experiments have two main outcomes. Firstly, the use of Homogeneity Based Segmentation is not vital for the system. Thus, the configuration of SphereDiar can be simplified by omitting segmentation. Furthermore, SphereDiar is shown to surpass the performance of two different state-of-the-art speaker diarization systems.

Keywords speaker diarization, speaker modeling, segmentation, clustering, meeting data

EURECOM
Sophia Antipolis

Doctoral Thesis Abstract

Author:	Danny Francis
Title:	Semantic Representations of Images and Videos
Date:	December 12, 2019 **Pages:** 151
Department:	Data Science
Supervisors:	Bernard Merialdo and Benoit Huet

Describing images or videos is a task that we all have been able to tackle since our earliest childhood. However, having a machine automatically describe visual objects or match them with texts is a tough endeavor, as it requires to extract complex semantic information from images or videos. Recent research in Deep Learning has sent the quality of results in multimedia tasks rocketing: thanks to the creation of big datasets of annotated images and videos, Deep Neural Networks (DNN) have outperformed other models in most cases. In this thesis, we aim at developing novel DNN models for automatically deriving semantic representations of images and videos. In particular we focus on two main tasks : vision-text matching and image/video automatic captioning.

Addressing the matching task can be done by comparing visual objects and texts in a visual space, a textual space or a multimodal space. In this thesis, we experiment with these three possible methods. Moreover, based on recent works on capsule networks, we define two novel models to address the vision-text matching problem: Recurrent Capsule Networks and Gated Recurrent Capsules. We find that replacing Recurrent Neural Networks usually used for natural language processing such as Long Short-Term Memories or Gated Recurrent Units by our novel models improve results in matching tasks. On top of that, we show that intrinsic characteristics of our models should make them useful for other tasks.

In image and video captioning, we have to tackle a challenging task where a visual object has to be analyzed, and translated into a textual description in natural language. For that purpose, we propose two novel curriculum learning methods. Experiments on captioning datasets show that our methods lead to better results and faster convergence than usual methods. Moreover regarding video captioning, analyzing videos requires not only to parse still images, but also to draw correspondences through time. We propose a novel Learned Spatio-Temporal Adaptive Pooling (L-STAP) method for video captioning that combines spatial and temporal analysis. We show that our L-STAP method outperforms state-of-the-art methods on the video captioning task in terms of several evaluation metrics.

Extensive experiments are also conducted to discuss the interest of the different models and methods we introduce throughout this thesis, and in particular how results can be improved by jointly addressing the matching task and the captioning task.

Keywords:	Deep Learning, Multimedia, Computer Vision, Natural Language Processing, Image, Video
Language:	English

B.2 AALTO and EURECOM's paper in TRECVID 2019 VTT [1]

This paper describes the runs that the AALTO and EURECOM teams submitted to TRECVID 2019 VTT and summarises their results.

PicSOM and EURECOM Experiments in TRECVID 2019

Pre-workshop draft – Revision: 0.9

Héctor Laria Mantecón[+], Jorma Laaksonen[+], Danny Francis[*], Benoit Huet[*]

[+]Department of Computer Science
Aalto University School of Science
P.O.Box 15400, FI-00076 Aalto, Finland
firstname.lastname@aalto.fi

[*]Department of Data Science
EURECOM, Campus SophiaTech
450 route des Chappes
06410 Biot, France
firstname.lastname@eurecom.fr

Abstract

This year, the PicSOM and EURECOM teams participated only in the Video to Text Description (VTT), Description Generation subtask. Both groups submitted one or two runs labeled as a "MeMAD" submission, stemming from a joint EU H2020 research project with that name. In total, the PicSOM team submitted four runs and EURECOM one run. The goal of the PicSOM submissions was to study the effect of using either image or video features or both. The goal of the EURECOM submission was to experiment with the use of Curriculum Learning in video captioning. The submitted five runs are as follows:

- PicSOM.1-MeMAD.primary: uses ResNet and I3D features for initialising the LSTM generator, and is trained on MS COCO + TGIF using self-critical loss,
- PicSOM.2-MeMAD: uses I3D features as initialisation, and is trained on TGIF using self-critical loss,
- PicSOM.3: uses ResNet features as initialisation, and is trained on MS COCO + TGIF using self-critical loss,
- PicSOM.4: is the same as PicSOM.1-MeMAD.primary except that the loss function used is cross-entropy,
- EURECOM.MeMAD.primary: uses I3D features to initialize a GRU generator, and is trained on TGIF + MSR-VTT + MSVD with cross-entropy and curriculum learning.

The runs aim at comparing the use of cross-entropy and self-critical training loss functions and to showing whether one can successfully use both still image and video features even when the COCO dataset does not allow the extractions of I3D video features. Based on the results of the runs, it seems that using both video and still image features, one can obtain better captioning results than with either one of the single modalities alone. The Curriculum Learning process proposed does not seem to be beneficial.

I. Introduction

In this notebook paper, we describe the PicSOM and EURECOM teams' experiments for the TRECVID 2019 evaluation [1]. We participated only in the Video to Text Description (VTT) subtask Description Generation. Our approaches are variations of the "Show and tell" model [2], augmented with a richer set of contextual features [3], self-critical training [4] and Curriculum Learning [5]. Both teams' systems have been used to produce the runs presented in this paper. The captioning models are described in more detail in Section II and their used training loss functions in Section III. Then, we describe the features in Section IV and the datasets used for training in Section V. Our experiments, submitted runs and results are discussed in Section VI and conclusions are drawn in Section VII.

II. Neural captioning models

In our experiments we have used two different Python-based software projects for caption generation. The PicSOM team's *DeepCaption*, uses the PyTorch library, whereas EURECOM's *CLCaption* approach is based on using the TensorFlow library.

A. DeepCaption

The PicSOM team's LSTM [6] model has been implemented in PyTorch and is available as open source.[1] The features are translated to the hidden size of the LSTM by using a fully connected layer. We apply dropout and batch normalization [7] at this layer. As the loss function, we similarly use cross entropy, in addition to Reinforcement Learning with self-critical loss function [4] in order to fine-tune a well-performing model. The fine-tuning is implemented either by switching to the self-critical loss in training time or by specifying a pre-trained model to load and fine-tune.

B. CLCaption

For EURECOM's first participation in the TRECVID VTT captioning task, we submitted a run based on a model trained

[1]https://github.com/aalto-cbir/DeepCaption

by Curriculum Learning [8]. We implemented our model using the TensorFlow framework for Python [9].

The idea behind Curriculum Learning is to present data during training in an ascending order of difficulty: first epochs are based on easy samples, and after each epoch, more difficult samples are added to training data. We computed a difficulty score for a given sample composed of a video and a corresponding caption as follows: the caption is translated into a list of indices (the bigger the indices the less frequent the corresponding word), the score of the sample is then the maximum index of its caption. Once samples have been scored, we trained the model starting with an easy subset of the training set, and adding after each epoch more complex samples.

Video features have been extracted with an I3D neural network [10], input to a fully connected layer and then processed by a GRU [11] to generate captions. Cross-entropy loss has been used for training the model.

III. Training loss functions

In order to train the architecture so that its output distribution approximates the target distribution at each decoding step t, several optimisation objectives are used. Recent progress on sequence training enables new optimisation paradigms, which are applied and compared in this work.

A. Cross-entropy

Traditionally, the teacher forcing algorithm [5] is the most common method to maximise the log-likelihood of a model output X to match the ground truth $y = \{y_1, y_2, \cdots, y_T\}$. It minimises the cross-entropy objective

$$\mathcal{L}_{CE} = -\sum_{t=1}^{T} \log p_\theta \left(y_t \mid y_{t-1}, \mathbf{h}_{t-1}, X\right), \qquad (1)$$

where $\mathbf{h}_{t-1}$ is the hidden state of the RNN from the previous step and p_θ the probability of an output parametrized by θ. In the inference time, the output can be produced simply by greedy sampling of the sequence being generated.

B. Self-critical

Lately, Reinforcement Learning ideas have been used to optimise a captioning system based on recurrent neural network language models. Such a system can be seen as an agent taking actions according to a policy π_θ and outputting a word $\hat{y}_t$ as an action.

One proposed approach is the self-critical algorithm [4], where the output at inference time of the model $\hat{y}^g_{i,t}$ is used, normally applying greedy search. The sequences are scored using a reward function r. Thanks to the properties of this optimisation, NLP metrics can be used as reward to affect the actual loss. In our case, CIDErD [12] is used. The final objective reads

$$\begin{aligned}\mathcal{L}_\theta = \tfrac{1}{N} \textstyle\sum_{i=1}^{N} \sum_t \log \pi_\theta \left(\hat{y}_{i,t} \mid \hat{y}_{i,t-1}, \mathbf{s}_{i,t}, \mathbf{h}_{i,t-1}\right) \\ \cdot \Big(r(\hat{y}_{i,1}, \cdots, \hat{y}_{i,T}) - r(\hat{y}^g_{i,1}, \cdots, \hat{y}^g_{i,T})\Big)\end{aligned} . \qquad (2)$$

IV. Features

Table I summarizes the features used in our experiments and their dimensionalities.

TABLE I
SUMMARY OF THE FEATURES USED IN OUR EXPERIMENTS.

abbr.	feature	dim.	modality
rn	CNN ResNet	4096	image
fr	Faster R-CNN	80	image
i3d	I3D	2048	video

A. CNN

We are using pre-trained CNN features from ResNet 101 and 152. The 2048-dimensional features from the pool5 layer average to five crops from the original and horizontally flipped images. These features have then been concatenated together and are referred to as "rn" in Table I and later in this paper. When applied to a video object, we have used the middlemost frame of the video.

B. FasterRCNN

The existence of certain objects in the visual scene has an effect on sentence formation and influences the adjectives used in human sentences. To extract this information, we use an object detector, specifically the Faster Region-based Convolutional Neural Network (R-CNN) [13]. This network predicts the object locations as bounding boxes and object detection scores of the 80 object categories of Microsoft Common Objects in Common Context (MS-COCO) database.[2] In our current approach we, however, ignore the location information and encode the object detection scores on the image level. We obtain thus an 80-dimensional feature vector using the detection score for each category, and refer to it as "fr". When applied to a video object, we have used the middlemost frame of the video.

C. I3D

To encode video features, the PicSOM team adopted Inflated 3D Convolutional Network (I3D) [10]. It builds upon already competent image recognition models (2D) and inflates the filters and kernels to 3D, thus creating an additional temporal dimension. Concretely, the base network used is ImageNet-pretrained Inception-V1 [14] using two streams [15]. The videos were first resampled to 25 frames per second as in the original I3D paper and 128 frames were taken from the center. For DeepCaption, the extractor is applied convolutionally over the whole video and the output is average-pooled in order to produce a 2048-dimensional feature vector.

Regarding CLCaption, features have been extracted before the softmax layer, thus obtaining a 600-dimensional features vector. These features have then been input to the CLCaption model.

[2] http://cocodataset.org/

V. Training data

Table II gives a summary of the databases and the features we have extracted for them. In Tables II and III, we have shortened the dataset names with one letter abbreviations.

TABLE II
SUMMARY OF THE TRAINING DATASETS USED IN OUR EXPERIMENTS.

	dataset	items	captions	features
C	COCO	82,783 img	414,113	rn fr
M	MSR-VTT	6,513 vid	130,260	rn i3d
T	TGIF	125,713 vid	125,713	rn fr i3d
V	MSVD	1,969 vid	80,800	rn i3d

A. COCO

The *Microsoft Common Objects in COntext (MS COCO)* dataset [16] has 2,500,000 labeled instances in 328,000 images, consisting on 80 object categories. COCO is focused on non-iconic views (or non-canonical perspectives) of objects, contextual reasoning between objects, and precise 2D localization of objects.

B. MSR-VTT

The *MSR-Video to Text (MSR-VTT)* dataset [17] provides 10,000 web video clips with 41.2 hours and 200,000 clip-sentence pairs in total, covering a comprehensive list of 20 categories and a wide variety of video content. Each clip was annotated with about 20 natural sentences. Additionally, the audio channel is provided too.

C. TGIF

The *Tumblr GIF (TGIF)* dataset [18] contains 100,000 animated GIFs and 120,000 natural language sentences. This dataset aims to provide motion information involved between image sequences (or frames).

D. MSVD

The *Microsoft Research Video Description Corpus (MSVD)* [19] consists of 85,000 English video description sentences and more than 1,000 for a dozen more languages. It contains a set of 2,089 videos, showing a single, unambiguous action or event.

VI. Experiments and results

During the development stage, the PicSOM team ran a number of experiments to select the best combinations of features and training datasets. We evaluated our results using the previously released ground truth of TRECVID VTT 2018 test set. The four runs submitted are identified as "p-19-s1" to "p-19-s4" in Table III. The runs "p-18-s2" and "p-18-a3" we created using our best model in the last year's submissions and the best model we experimented with after the last year's workshop, respectively.

Runs identified as "p-19-s1" and "p-19-s4" use I3D video features extracted from the TGIF dataset. We used also the COCO dataset for training the models for those runs, but because we could not extract I3D video features from the still images of that dataset, we used the average value of the I3D feature vectors of the TGIF dataset for each COCO image.

Based on evaluation on the TRECVID VTT 2018 test set, we ended up using a 2-layer LSTM for DeepCaption with an embedding vector size of 512, and 1024 for the hidden state dimensionality in all PicSOM team's runs. Both in the input translation layer and in the LSTM we applied a dropout of 0.5. We used Adam optimiser [20] for the self-critical stage with a learning rate of 5×10^{-5} and no weight decay. Additionally, gradient clipping is performed when a range $[-0.1, 0.1]$ is exceeded. The models were pretrained using centered RMSprop [21] with a learning rate of 0.001 and weight decay (L2 penalty) of 10^{-6}.

EURECOM's CLCaption is based on a GRU with 1024-dimensional hidden states. The size of the input I3D vectors is 600. The fully-connected layer output is of dimension 1024. No dropout nor batch normalization were used. The training algorithm we used was RMSProp with a learning rate of 0.0001 and mini-batches of size 64. The CLCaption run is identified as "e-19-e1" in Table III where all experiments are briefly summarized and their results presented.

The four "setup" columns in Table III specify the submission type (I=image, V=video, B=both), the loss function (ce=cross-entropy, sc=self-critical), the features, and the datasets used in the RNN model training.

The features are concatenations of the following:

rn = CNN ResNet, see IV-A
fr = Faster R-CNN, see IV-B
i3d = I3D, see IV-C

The used datasets are combinations of the following:

C = COCO, see V-A
M = MSR-VTT, see V-B
T = TGIF, see V-C
V = MSVD, see V-D

Our results compared to those of the other submitted runs are visualized with bar charts for each automatic performance measure in Figures 1–5.

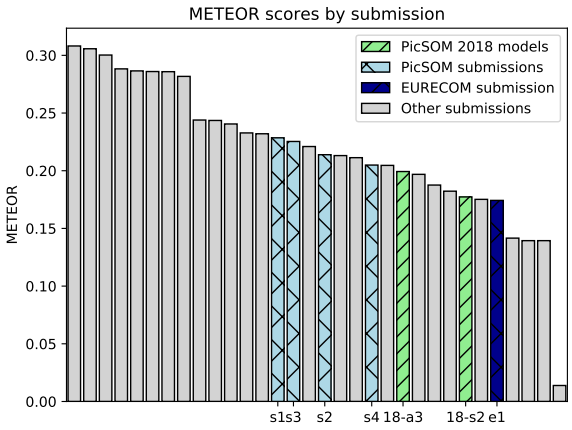


Fig. 1. METEOR results of our teams and others.

TABLE III
RESULTS OF OUR SUBMISSIONS (P-19-S1,...,4, E-19-E1) AND SOME NOTEWORTHY EARLIER MODELS (P-18-S2, P-18-A3). THE P-* RUNS ARE BY THE PICSOM TEAM AND THE E-* RUN BY THE EURECOM TEAM.

	setup				2018				2019				
id	**t**	**loss**	**feat**	**data**	**METEOR**	**CIDEr**	**CIDErD**	**BLEU**	**METEOR**	**CIDEr**	**CIDErD**	**BLEU**	**STS**
p-18-s2	I	ce	rn+fr	C+M	0.1541	0.1657	0.0476	0.0091	0.1773	0.1858	0.0722	0.0207	–
p-18-a3	I	ce	rn	C+T	0.1776	0.1948	0.0700	0.0197	0.1993	0.2174	0.1004	0.0288	–
p-19-s1	B	sc	rn+i3d	C+T	**0.2055**	**0.3025**	**0.1157**	0.0294	**0.2285**	**0.3277**	**0.1615**	**0.0385**	0.4168
p-19-s2	V	sc	i3d	T	0.1958	0.2718	0.0949	**0.0348**	0.2139	0.2773	0.1245	*0.0379*	*0.4169*
p-19-s3	I	sc	rn	C+T	*0.2007*	*0.2777*	*0.1074*	*0.0301*	*0.2254*	*0.3130*	*0.1569*	0.0345	**0.4282**
p-19-s4	B	ce	rn+i3d	C+T	0.1850	0.2190	0.0822	0.0213	0.2049	0.2348	0.1147	0.0319	0.4057
e-19-e1	V	ce	i3d	M+T+V	–	–	–	–	0.1743	0.2340	0.0710	0.0068	0.4214

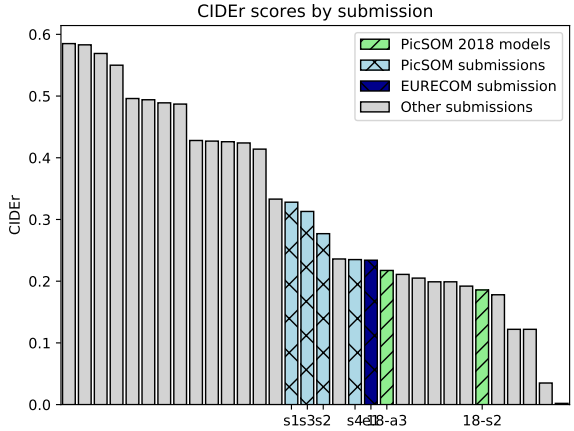


Fig. 2. CIDEr results of our teams and others.

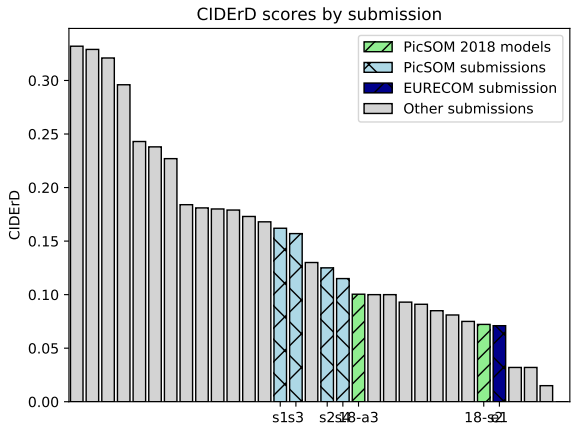


Fig. 3. CIDErD results of our teams and others.

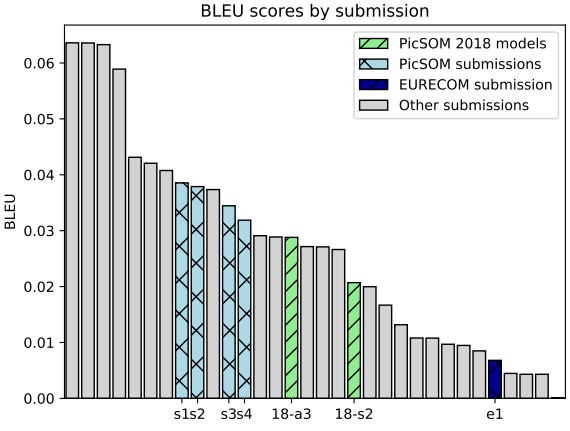


Fig. 4. BLEU results of our teams and others.

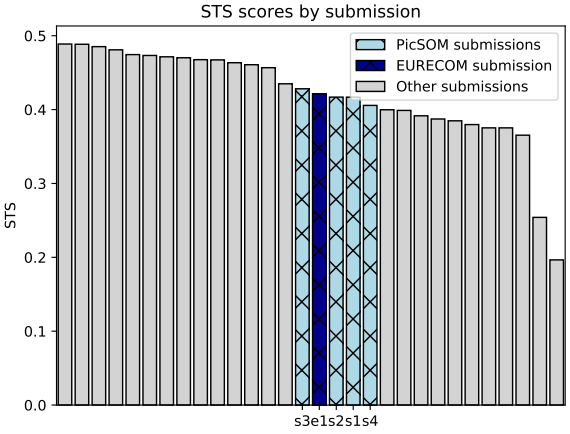


Fig. 5. STS results of our teams and others.

VII. CONCLUSIONS

There were two main research question in the PicSOM team's set of four submissions. First, we wanted to compare the implementations of cross-entropy and self-critical training loss functions in our DeepCaption code. The results with self-critical training were better in all measures, but this could of course be expected based on our and other teams' earlier experiments. Based on our observations, however, the use of this loss alone does not imply a straightforward jump in caption quality as much as the score increment suggests.

Second, we aimed to know whether we could successfully use both still image and video features even when the COCO dataset does not allow the extractions of I3D video features. The trick we applied was to use the average of the I3D video features extracted from the TGIF dataset for all images in the COCO dataset. For the COCO images the video features were thus non-informative, but still allowed us to use two datasets

and two different feature extraction schemes together. The results of this approach were encouraging as they were better than those with either dataset or either feature used alone.

Additionally, we could now compared the current performance of the PicSOM team's DeepCaption model to its performance in the last year's evaluation. We have clearly made substantial progress compared to both the last year's submission and to the post-workshop experiments reported in our previous notebook paper. However, compared to the level of performance reached by some of the other research groups, we are still clearly behind as all the groups seem to have improved from the previous year.

The results obtained by CLCaption are far from standing comparison with the best runs of TRECVID VTT 2019. However, multiple ways to improve them can be explored, such as different scoring methods or finer curriculum learning algorithms. We will explore these directions to boost the results of CLCaption.

Acknowledgments

This work has been funded by the grant 313988 *Deep neural networks in scene graph generation for perception of visual multimedia semantics* (DeepGraph) of the Academy of Finland, the ANR (the French National Research Agency) via the ANTRACT project, and the European Union's Horizon 2020 research and innovation programme under grant agreement No 780069 *Methods for Managing Audiovisual Data: Combining Automatic Efficiency with Human Accuracy* (MeMAD). This work was supported by the Academy of Finland Flagship programme: Finnish Center for Artificial Intelligence, FCAI. The calculations were performed using computer resources provided by the Aalto University's *Aalto Science IT* project and CSC – IT Center for Science Ltd. We also acknowledge the support of NVIDIA Corporation with the donation of TITAN X and Quadro P6000 GPUs used for parts of this research.

References

[1] George Awad, Asad Butt, Keith Curtis, Yooyoung Lee, Jonathan Fiscus, Afzal Godil, Andrew Delgado, Alan F. Smeaton, Yvette Graham, Wessel Kraaij, and Georges Quénot. Trecvid 2019: An evaluation campaign to benchmark video activity detection, video captioning and matching, and video search & retrieval. In *Proceedings of TRECVID 2019*. NIST, USA, 2019.

[2] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. Show and tell: A neural image caption generator. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[3] Rakshith Shetty, Hamed R.-Tavakoli, and Jorma Laaksonen. Image and video captioning with augmented neural architectures. *IEEE MultiMedia*, 25(2):34–46, April 2018.

[4] Steven J. Rennie, Etienne Marcheret, Youssef Mroueh, Jarret Ross, and Vaibhava Goel. Self-critical sequence training for image captioning. *CoRR*, abs/1612.00563, 2016.

[5] Samy Bengio, Oriol Vinyals, Navdeep Jaitly, and Noam Shazeer. Scheduled sampling for sequence prediction with recurrent neural networks. *CoRR*, abs/1506.03099, 2015.

[6] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Comput.*, 9(8):1735–1780, November 1997.

[7] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning*, pages 448–456, 2015.

[8] Yoshua Bengio, Jérôme Louradour, Ronan Collobert, and Jason Weston. Curriculum learning. In *Proceedings of the 26th Annual International Conference on Machine Learning, ICML 2009, Montreal, Quebec, Canada, June 14-18, 2009*, pages 41–48, 2009.

[9] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, Manjunath Kudlur, Josh Levenberg, Rajat Monga, Sherry Moore, Derek Gordon Murray, Benoit Steiner, Paul A. Tucker, Vijay Vasudevan, Pete Warden, Martin Wicke, Yuan Yu, and Xiaoqiang Zheng. Tensorflow: A system for large-scale machine learning. In *12th USENIX Symposium on Operating Systems Design and Implementation, OSDI 2016, Savannah, GA, USA, November 2-4, 2016.*, pages 265–283, 2016.

[10] João Carreira and Andrew Zisserman. Quo vadis, action recognition? A new model and the kinetics dataset. *CoRR*, abs/1705.07750, 2017.

[11] Kyunghyun Cho, Bart van Merrienboer, Çaglar Gülçehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. Learning phrase representations using RNN encoder-decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing, EMNLP 2014, October 25-29, 2014, Doha, Qatar, A meeting of SIGDAT, a Special Interest Group of the ACL*, pages 1724–1734, 2014.

[12] Ramakrishna Vedantam, C. Lawrence Zitnick, and Devi Parikh. Cider: Consensus-based image description evaluation. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[13] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *NIPS*, 2015.

[14] Sergey Ioffe and Christian Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. *CoRR*, abs/1502.03167, 2015.

[15] Christoph Feichtenhofer, Axel Pinz, and Andrew Zisserman. Convolutional two-stream network fusion for video action recognition. *CoRR*, abs/1604.06573, 2016.

[16] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C. Lawrence Zitnick. Microsoft COCO: Common objects in context. In *European Conference on Computer Vision (ECCV)*, 2014.

[17] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. MSR-VTT: A large video description dataset for bridging video and language. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5288–5296, 2016.

[18] Yuncheng Li, Yale Song, Liangliang Cao, Joel Tetreault, Larry Goldberg, Alejandro Jaimes, and Jiebo Luo. TGIF: A new dataset and benchmark on animated gif description. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4641–4650, 2016.

[19] David L. Chen and William B. Dolan. Collecting highly parallel data for paraphrase evaluation. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies - Volume 1*, HLT '11, pages 190–200, Stroudsburg, PA, USA, 2011. Association for Computational Linguistics.

[20] Diederik P. Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In Yoshua Bengio and Yann LeCun, editors, *3rd International Conference on Learning Representations, ICLR 2015, San Diego, CA, USA, May 7-9, 2015, Conference Track Proceedings*, 2015.

[21] Alex Graves. Generating sequences with recurrent neural networks. *arXiv preprint arXiv:1308.0850*, 2013.

B.3 EURECOM's paper in TRECVID 2019 AVS workshop [2]

This paper describes the runs that the EURECOM team submitted to TRECVID 2019 AVS and summarises the results.

EURECOM at TRECVid AVS 2019

Danny Francis[†], Phuong Anh Nguyen[‡], Benoit Huet[†], Chong-Wah Ngo[‡]
[†]EURECOM, Sophia-Antipolis, France [‡]City University of Hong-Kong, Kowloon, Hong Kong
francis@eurecom.fr, panguyen2-c@my.cityu.edu.hk, huet@eurecom.fr, cscwngo@cityu.edu.hk

Abstract

This notebook reports the model and results of the EURECOM runs at TRECVID AVS 2019.

1. Introduction

In our runs of TRECVid AVS 2019, we propose using a fusion of two multimodal modules trained on different datasets. Our runs are based on the work we introduced in [4].

The remaining sections are organized as follows. Section 2 presents related works in AVS. Section 4 introduces the cross-modal learning employed for training two different modules, Section 4 describes the followed fusion method, and Section 5 reports our results at TRECVid AVS 2019 [2].

2. Related Works

From AVS 2018, the general approaches from the participants can be summarized as follows: linguistic analysis for query understanding combining different techniques for concept selection and fusion; or learning joint embedding space of textual queries and images; or the integration of two mentioned approaches. From the results of ten participants, we conclude that the approach of learning the embedding space is the key of success for AVS task. Following up this direction, we propose to learn two embedding spaces including objects counting and semantic concepts separately, and a fusion method to incorporate these models.

3. Cross-Modal Learning

In this section we will describe the multimodal models we employed. More precisely we will first define their architecture and then how we trained them.

3.1. Feature Representation

Let Q be a textual query and V an image or a video. We want to build a model so that Q and V can be compared. More precisely, we want to be able to assign a score to any (Q, V) to describe the relevance of V with respect to Q. For that purpose, we use a similar model to [3].

For processing textual queries, we represent any query Q of length L as a sequence $(w_1, ..., w_L)$ of one-hot vectors of dimension N, where N is the size of our vocabulary. These one-hot vectors are then embedded in a vector space of dimension D. More formally, we obtain a sequence of word embeddings $(x_1, ..., x_L)$ where $x_k = w_k W_e$ for each k in $\{1, ..., L\}$. The weights of the embedding matrix $W_e \in \mathbb{R}^{D \times N}$ are trainable.

The obtained sequence of word embeddings is then processed by a GRU, whose last hidden state $h_L = \text{GRU}(h_{L-1}, x_L)$ is kept and input to a Fully-Connected layer to get a sentence embedding v_s.

Regarding visual objects, the generic process we employ is to extract a vector representation $\varphi(V)$ of a visual object V where φ corresponds to any relevant concepts or features extractor. Then, we input $\varphi(V)$ to a Fully-Connected layer to obtain a visual embedding v_v.

Our goal is to train these models to be able to compare v_s and v_v. We will explain how these models are trained in Section 3.2.

3.2. Model Training

The objective is to learn a mapping such that the relevancy of a pair of a query and a video (Q, V) can be evaluated. As explained in Section 3.1, our model derives a query representation v_s from Q and a video representation v_v from V. Triplet loss is used as the loss function for model training. Mathematically, if we consider a query representation v_s, a positive video representation v_v (corresponding to v_s) and a negative video representation $\bar{v}_v$ (that does not correspond to v_s), the triplet loss $\mathcal{L}$ for $(v_s, v_v, \bar{v}_v)$ to minimize is defined as follows:

$$\mathcal{L}(v_s, v_v, \bar{v}_v) = \max(0, \alpha - \cos(v_s, v_v) + \cos(v_s, \bar{v}_v)) \quad (1)$$

where α is a margin hyperparameter that we set to 0.2. We chose to employ the hard-margin loss presented in [3], where $\bar{v}_v$ is chosen to be the representation of the negative

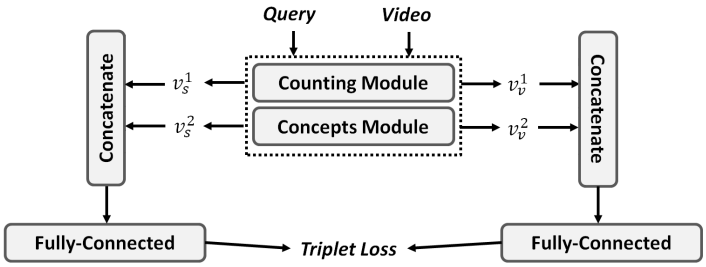


Figure 1. Proposed model derived from [4]. We extract embeddings from two modules: a counting module and a concepts module. These embeddings are then concatenated and input to Fully-Connected layers to obtain new embeddings. That model is also trained using a triplet loss.

video with the highest similarity with the query representation v_s among all videos in the current training mini-batch.

4. Fusion Strategy

In this section we will describe the two multimodal modules we used and how we fused them.

4.1. Multimodal Modules

Our model relies on two multimodal modules: a counting module and a concepts module (see Figure 1). Each of them has the architecture we described in Section 3.1 and has been trained according to the optimization scheme we defined in Section 3.2.

The counting module is based on a Faster-RCNN [10] trained on the OpenImagesv4 dataset [7]. It takes images as inputs. For each input, it detects objects belonging to the 600 classes of OpenImagesv4 and counts them to obtain a vector of dimension 600, where the value at index i corresponds to the number of detected objects of class i. Embeddings are then derived from that vector.

The concepts module takes as input concepts detections coming from four different concept detectors. These concept detectors are ResNet [5] models trained on ImageNet1k, Places-365 [16], TRECVID SIN [15] and HAVIC [12]. Following the same process as for other two modules, we generate embeddings from the concatenation of the concept detections coming from these four detectors.

4.2. Fusion Model

Instead of simply averaging similarity scores to compare videos and queries, we chose to train a model to draw finer similarities between them. For that purpose, we derived embeddings from our modules for videos and queries, and passed them through Fully-Connected layers to obtain new embeddings. More formally, if v_v^1 and v_v^2 are video embeddings respectively generated by the counting module and the concepts module, we derived the new video embedding v_v by inputting the concatenation of v_v^1 and v_v^2 to a fully-connected layer. We obtained the new sentence embedding v_s similarly, based on v_s^1 and v_s^2 (sentence embeddings generated by the counting and the concepts modules, respectively).

We trained our fusion models using the same triplet loss as we did for multimodal modules, as decribed in Section 3.2.

5. Results of runs

In this section, we report the results we obtained at TRECVid 2019.

5.1. Datasets

We trained our models based on the MSCOCO [9] dataset the TGIF [8] dataset and the train and test splits of the MSR-VTT [14] dataset. Validation has been performed on the validation split of MSR-VTT.

5.2. Implementation details

We implemented our models using the Tensorflow [1] framework for Python. Each of them has been trained for 150k iterations with mini-batches of size 64. We used the RMSProp [13] algorithm, with gradients capped to values between -5 and 5 and a learning rate of 10^{-4}. Hidden dimensions of GRUs are always 1024, and embeddings output by multimodal modules and fusion models are of dimension 512. The size of vocabularies has been set to 20k. We applied dropout [11] with rate 0.3 to all outputs of Fully-Connected layers, and batch normalization [6] to the inputs of our models. In triplet losses, the α parameter has been set to 0.2.

MSR-VTT videos have been processed as follows: we extracted uniformly one frame every fifteen frames, applied the extractor on each frame (Faster-RCNN for the counting module or concepts extractors for the concepts module) and averaged obtained vectors.

5.3. Results of Runs

The runs we submitted were the following:

Run	MAP
Run 1	0.014
Run 2	0.014
Run 3	0.020

Table 1. Results of our runs

- Run 1: Fusion of Concepts and Counting modules;
- Run 2: Concepts module alone;
- Run 3: If Q is a query, V a video, $S_1(Q, V)$ the score of the pair (Q, V) computed in run 1 and $S_2(Q, V)$ the score in run 2, the score in run 3 is $S_1(Q, V) + S_2(Q, V)$.

The scores we obtained with these three runs are reported in Table 1.

Results of all automatic runs are reported in Figure 2. Detailed results of Run 1, Run 2 and Run 3 are reported in Figure 3, Figure 4 and Figure 5, respectively.

6. Conclusion

EURECOM runs performed badly with respect to other runs. However, results got better when ensembling run 1 and run 2 into run 3. For future work, we think we should investigate how other methods than multimodal embeddings perform. Moreover, we think that a finer sentence processing method than using a single GRU should be found, for instance putting emphasis on visual concepts.

Acknowledgments

This work was supported by ANR (the French National Research Agency) via the ANTRACT project, the European H2020 research and innovation programme via the project MeMAD (Reference Np.: GA780069), a grant from the Research Grants Council of the Hong Kong SAR, China (Reference No.: CityU 11250716), and a grant from the PROCORE-France/Hong Kong Joint Research Scheme sponsored by the Research Grants Council of Hong Kong and the Consulate General of France in Hong Kong (Reference No.: F-CityU104/17).

References

[1] M. Abadi, P. Barham, J. Chen, Z. Chen, A. Davis, J. Dean, M. Devin, S. Ghemawat, G. Irving, M. Isard, et al. Tensorflow: A system for large-scale machine learning. In *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*, pages 265–283, 2016.

[2] G. Awad, A. Butt, K. Curtis, Y. Lee, J. Fiscus, A. Godil, A. Delgado, A. F. Smeaton, Y. Graham, W. Kraaij, and G. Quénot. Trecvid 2019: An evaluation campaign to benchmark video activity detection, video captioning and matching, and video search & retrieval. In *Proceedings of TRECVID 2019*. NIST, USA, 2019.

[3] F. Faghri, D. J. Fleet, J. R. Kiros, and S. Fidler. Vse++: Improving visual-semantic embeddings with hard negatives.

[4] D. Francis, P. A. Nguyen, B. Huet, and C.-W. Ngo. Fusion of multimodal embeddings for ad-hoc video search. In *ViRaL 2019, 1st International Workshop on Video Retrieval Methods and Their Limits, co-located with ICCV 2019, 28 October 2019, Seoul, Korea*, 10 2019.

[5] K. He, X. Zhang, S. Ren, and J. Sun. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 770–778, 2016.

[6] S. Ioffe and C. Szegedy. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In *International Conference on Machine Learning*, pages 448–456, 2015.

[7] A. Kuznetsova, H. Rom, N. Alldrin, J. Uijlings, I. Krasin, J. Pont-Tuset, S. Kamali, S. Popov, M. Malloci, T. Duerig, et al. The open images dataset v4: Unified image classification, object detection, and visual relationship detection at scale. *arXiv preprint arXiv:1811.00982*, 2018.

[8] Y. Li, Y. Song, L. Cao, J. Tetreault, L. Goldberg, A. Jaimes, and J. Luo. Tgif: A new dataset and benchmark on animated gif description. In *2016 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 4641–4650. IEEE, 2016.

[9] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick. Microsoft coco: Common objects in context. In *European conference on computer vision*, pages 740–755. Springer, 2014.

[10] S. Ren, K. He, R. Girshick, and J. Sun. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*, pages 91–99, 2015.

[11] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov. Dropout: a simple way to prevent neural networks from overfitting. *The journal of machine learning research*, 15(1):1929–1958, 2014.

[12] S. M. Strassel, A. Morris, J. G. Fiscus, C. Caruso, H. Lee, P. Over, J. Fiumara, B. Shaw, B. Antonishek, and M. Michel. Creating havic: Heterogeneous audio visual internet collection. Citeseer.

[13] T. Tieleman and G. Hinton. Lecture 6.5—RmsProp: Divide the gradient by a running average of its recent magnitude. COURSERA: Neural Networks for Machine Learning, 2012.

[14] J. Xu, T. Mei, T. Yao, and Y. Rui. Msr-vtt: A large video description dataset for bridging video and language. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5288–5296, 2016.

[15] W. Zhang, H. Zhang, T. Yao, Y. Lu, J. Chen, and C. Ngo. Vireo@ trecvid 2014: instance search and semantic indexing. In *NIST TRECVID Workshop*, 2014.

[16] B. Zhou, A. Lapedriza, A. Khosla, A. Oliva, and A. Torralba. Places: A 10 million image database for scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017.

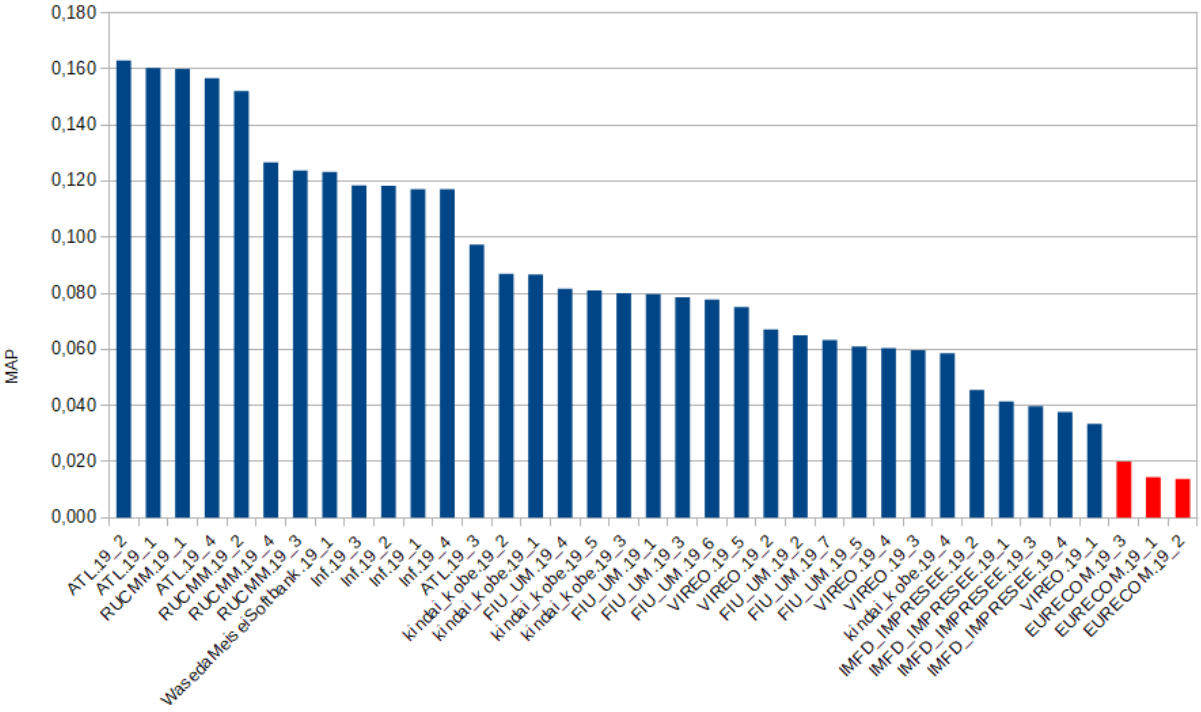


Figure 2. AVS Results (Fully Automated runs only)

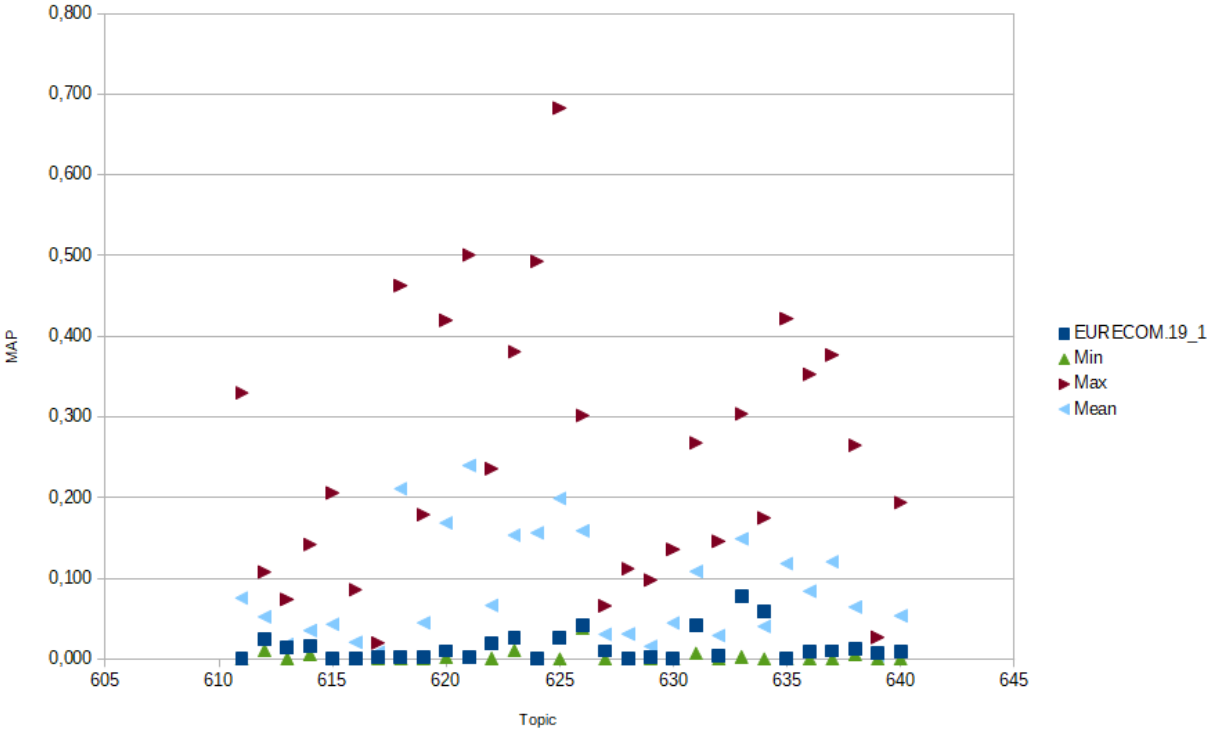


Figure 3. Detailed results of EURECOM run 1

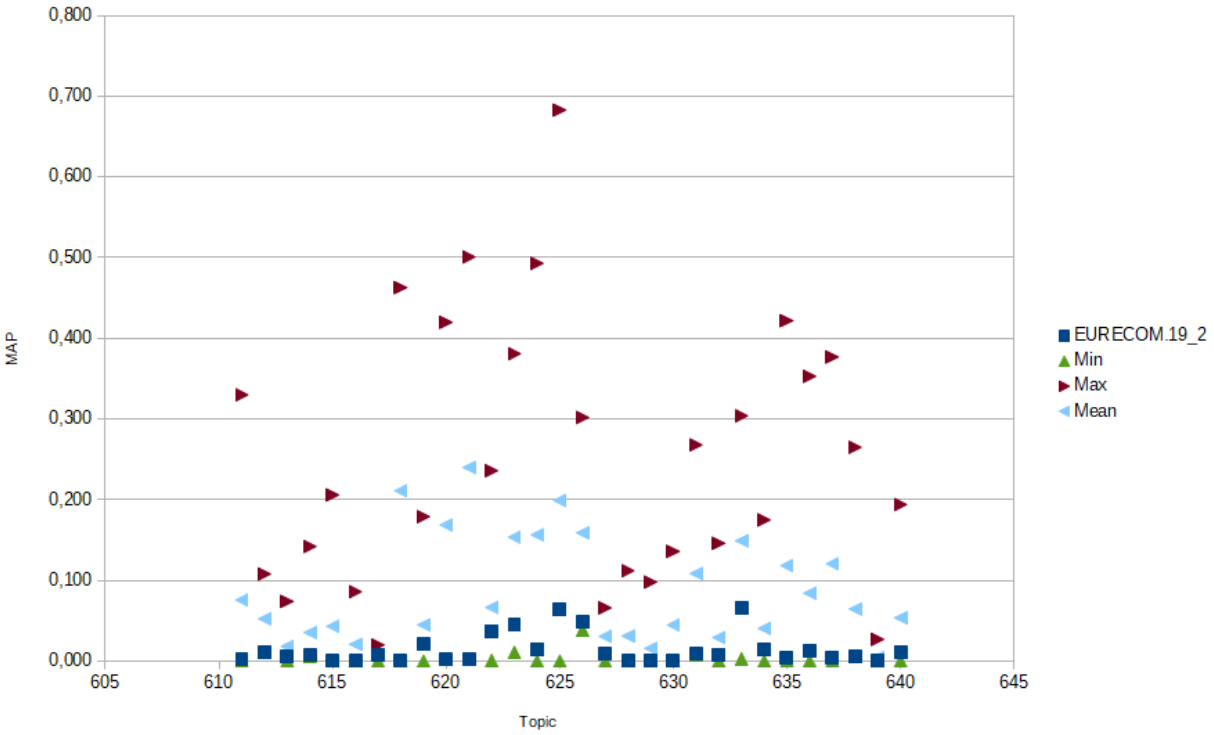


Figure 4. Detailed results of EURECOM run 2

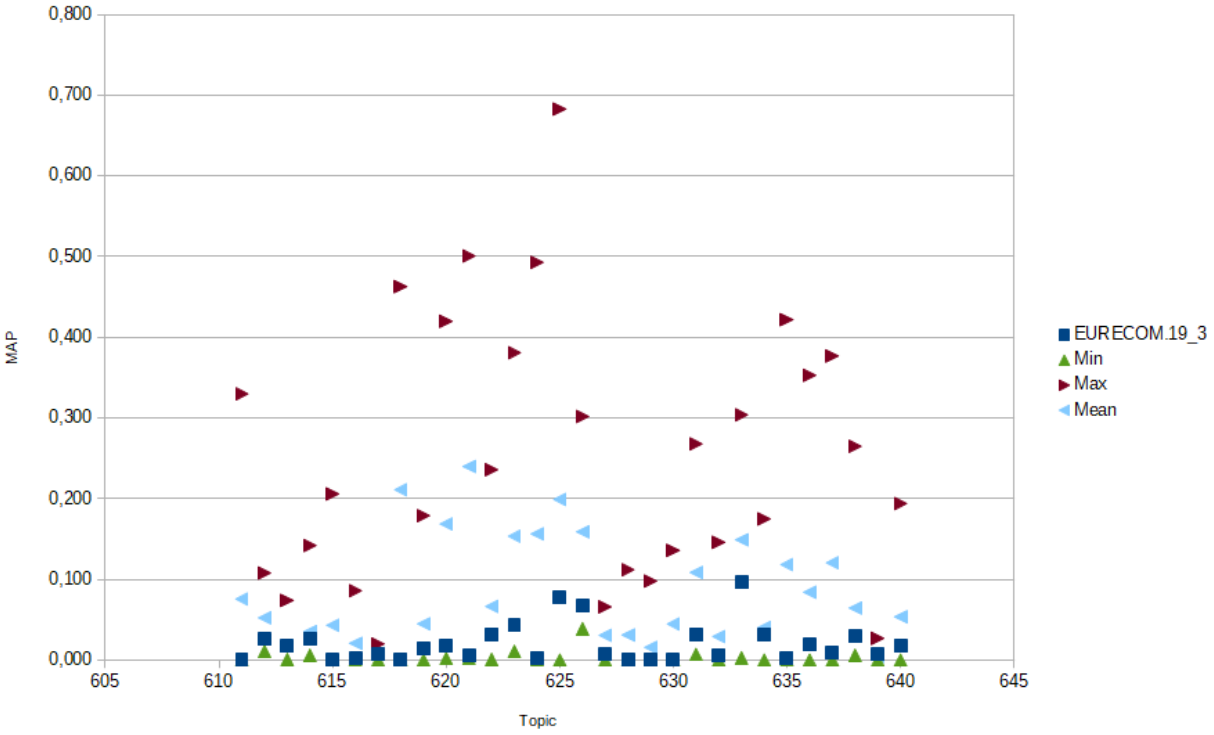


Figure 5. Detailed results of EURECOM run 3

B.4 INA's paper in VIEW [3]

This paper describes the results obtained using inaSpeechSegmenter on massive amount of data. Despite the date shown in the paper, it was published in April 2019.

volume 7 issue 14/2018
VIEU
JOURNAL OF EUROPEAN TELEVISION HISTORY & CULTURE

DESCRIBING GENDER EQUALITY IN FRENCH AUDIOVISUAL STREAMS WITH A DEEP LEARNING APPROACH

David Doukhan
French National Audiovisual Institute (INA)
4 Avenue de l'Europe
94360 Bry-sur-Marne
France
ddoukhan@ina.fr

Géraldine Poels
French National Audiovisual Institute (INA)
4 Avenue de l'Europe
94360 Bry-sur-Marne
France
gpoels@ina.fr

Zohra Rezgui
French National Audiovisual Institute (INA)
4 Avenue de l'Europe
94360 Bry-sur-Marne
France
zrezgui@ina.fr

Jean Carrive
French National Audiovisual Institute (INA)
4 Avenue de l'Europe
94360 Bry-sur-Marne
France
jcarrive@ina.fr

Abstract: A large-scale description of men and women speaking-time in media is presented, based on the analysis of about 700.000 hours of French audiovisual documents, broadcasted from 2001 to 2018 on 22 TV channels and 21 radio stations.

Speaking-time is described using Women Speaking Time Percentage (WSTP), which is estimated using automatic speaker gender detection algorithms, based on acoustic machine learning models.

WSTP variations are presented across channels, years, hours, and regions. Results show that men speak twice as much as women on TV and on radio in 2018, and that they used to speak three times longer than women in 2004. We also show only one radio station out of the 43 channels considered is associated to a WSTP larger than 50%. Lastly, we show that WSTP is lower during high-audience time-slots on private channels.

This work constitutes a massive gender equality study based on the automatic analysis of audiovisual material and offers concrete perspectives for monitoring gender equality in media.The software used for the analysis has been released in open-source, and the detailed results obtained have been released in open-data.

Keywords: Gender Equality, Digital Humanities, Machine Learning, Machine Listening, Speaker Gender Detection, Women speaking time percentage, Audiovisual description, open-data

1 Introduction

Gender equality in media is a concern which has been described using various methodologies. A panel of studies based on quantitative analysis are listed below.

International Women's Media Foundation realized world-wide comparative studies based on a sample of 59 countries.[1] Gender equality was described based on the percentage of women occupying top-decision making posts in medias according to their occupational status (governance, reporters, junior or senior professionals, ...), together with their average salaries and terms of employment (full-time, part-time, freelance, …). The World Association for Christian Communication has carried out quinquennial comparative studies since 1995, known as the *Global Media Monitoring Project* (GMMP).[2] The last edition of this analysis was based on a sample of 114 countries, and is known as the largest international study of gender in the news media. GMMP describes gender equality as the proportion of female subjects covered in the news, as well as the percentage of female presenters and reporters detailed by age and topics (health, economy, …).

In France, studies on gender equality in media have been ordered by the government, and released as public reports based on the analysis of Gomez-Michelis-Mielczareck corpus (GMM).[3] Equality was described based on the *identification rate*, defined as the percentage of oral references to male or female characters, and *presence rate,* defined as the proportion of male and female participants found in programs.

Since 2014, French media has been monitored by the French *Higher Council of Audiovisual* (Conseil supérieur de l'audiovisuel - CSA), which is an independent administrative authority in charge of ensuring a fair representation of men and women in French audiovisual programs.[4] Gender equality issues are described through women and men *presence rates*. Participants are split into five categories based on the declarations of TV channels and radio stations: *presenter*, *journalist*, *political guest*, *expert* and *other.* Rates of presence are presented across time-slots associated to low and high audiences, as well as channels based on their status (public, private) and topics (news, generalist, music).

Among these descriptors of gender equality in media, men and women *speaking time percentage*, also known as *expression rate*, has been used in a relatively low amount of studies. Reiser & Gresy used it in their report based on GMM corpus analysis.[5] Their corpus contains programs broadcasted on May 15, 2008 on 6 TV channels and 6 radio stations. The amount of recordings collected per channel or station ranges from 6 minutes to three hours. They show that in the 6 news programs analyzed, only 32 % of the speech-time was attributed to women speakers (excluding presenters), and that mean speech-turn time was 12 seconds for men and 9.1 secondes for women. Women and men speaking time was also presented in an experimental study conducted by Belgian CSA

1 Carolyn M. Byerly, *Global Report on the Status Women in the News Media*, International Women's Media Foundation [IWMF], 2011.

2 Sarah Macharia, *Who makes the news?*, Global Media Monitoring Project, 2015.

3 Michèle Reiser and Brigitte Gresy, *L'image des femmes dans les médias*, Secrétariat d'Etat à la solidarité, 2008.

4 CSA, *La représentation des femmes à la télévision et à la radio - Rapport sur l'exercice 2016*, Conseil supérieur de l'audiovisuel, 2017.

5 Reiser, *L'image des femmes (shortened)*.

(Higher Council of Audiovisual).[6] The study was based on the analysis of 36 hours of programs broadcasted during a week and speech-time was presented for several age categories of men and women.

Manual estimation of speaking time in TV and radio programs is expensive and time-consuming. Studies describing *expression rate* and women speaking-time percentage are therefore limited to the analysis of relatively small amounts of data. This limitation induces biases related to the particular socio-political context of the day in which these estimates were made: this context may influence the topics covered in medias as well as the selection of program's participants. Consequently, expression rate analyses are systematically presented together with a detailed description of the events corresponding to the particular day being analysed. This event description is necessary to characterize the bias affecting the description of the status of men and women in media.

More recently, larger scale studies based on the analysis of word-count per speaker were conducted, allowing us to obtain descriptors correlated to speech-time. These strategies were based on the use of external meta-data corresponding to document screenplays, describing the speaking characters' names together with the lexical content of their utterances, and were restricted to fictions. This allowed to replace the costly viewing and annotation process of audiovisual documents by automatic procedures of textual film script analysis, and resulted in studies based on the analysis of 12 disney princess movies,[7] and 2000 Hollywood movies.[8] As pointed out by the authors, the limitation of this strategy is that screenplays are not a perfect transcription of film dialogues. Moreover, this approach is limited to materials associated to accessible screenplay, which excludes a large amount of the broadcasted materials (live shows, debates, ...). Authors of the 2000 films analysis made a quite polemical statement in favor of the introduction of data-driven approaches in civic debates: *"But it's all rhetoric and no data, which gets us nowhere in terms of having an informed discussion".*

Based on the recent advances in artificial intelligence and machine learning, this article presents an automatic approach aimed at describing **Women Speaking Time Percentage** (**WSTP)**. This method relies on acoustic analysis systems allowing to distinguish male from female speech. Resulting analyses are performed on massive amounts of audiovisual documents. This analysis scale is aimed at reducing biases associated to manual studies realized on relatively small amounts of data. This approach is aimed at describing the evolution of the French audiovisual landscape, putting in evidence phenomena guiding the definition of qualitative studies, and proving to broadcasters with automated tools allowing them to estimate the impact of their policies for better gender representation.

2 Automatic Speaker Gender Segmentation System

2.1 Auditory Perception of Speaker's Gender

Differences between women and men speech are based on several auditory clues. Women speech is generally associated to higher pitch, to vowel formants located in higher frequencies and is more breathy. Contrast between men and women speech is partly due to physiological differences in vocal organs. Differences existing between men and women speech are also language-dependent, and related to the construction of gender identity in a given socio-cultural context.[9] Gender recognition is therefore harder for speakers having marked accents (regional, foreign), extreme pitch ranges, or speaking using non-standard intonation (very expressive voice, imitation, mental disorder, ...).

6 Sabri Derinoz, Muriel Hanot and Bertrand Levant, 'How gender representations matter with generation in television?', *II International Conference Gender and Communication*, 2014.

7 Carmen Fought et Karen Eisenhauer, 'A quantitative analysis of Gendered compliments in Disney Princess Films', *Linguistic Society of America*, 2016.

8 Hanah Anderson and Matt Daniels, 'Film Dialogue from 2,000 screenplays, Broken Down by Gender and Age', *The Pudding*, 2016, **https://pudding.cool/2017/03/film-dialogue/**.

9 Erwan Pépiot, 'Voice, speech and gender: Male-female acoustic differences and cross-language variation in English and French speakers,' *Corela (Cognition, représentation, langage)*, HS-16, 2015.

2.2 Automatic Speaker Gender Detection

Analyses presented in this study were realized using inaSpeechSegmenter.[10] This software, based on the acoustic analysis of audiovisual document soundtrack, outputs time-coded segments corresponding to music, women speech and men speech (Figure 1). This allows us to obtain hourly estimates of men and women speaking time, required to compute WSTP (Figure 2).

It has been built using deep Convolutional Neural Network models (CNN), a family of machine-learning algorithms that showed superior performances over other state-of-the art methods. This open-source software is freely available,[11] and is associated to an average processing time of about 70 seconds for one hour long documents, using machines equipped with Graphical Processing Units (Geforce 1080 Ti).

Machine-learning algorithms require examples corresponding to the concepts to be learned. Training examples should be representative of the diversity of the material handled by the software: accent, speaking-style, expressive modality, recording conditions... InaSpeechSegmenter's models were trained using INA's speaker dictionary, which is to our knowledge the biggest manually-annotated database of speakers issued from broadcast material.[12] This dictionary was realized using semi-automatic annotation procedures based on Optical Character Recognition. TV news excerpts with personality name appearing on screen were presented to annotators in charge of manual validation. The resulting dictionary is composed of documents collected from 1957 to 2012, allowing a comprehensive representations of speaking styles and recording conditions across decades. It contains 32.000 speech samples corresponding to 1780 distinct mens (94h) and 494 womens (27h).

InaSpeechSegmenter's evaluation was based in its ability to estimate WSTP. Estimator's robustness was shown to be proportional to archive's durations, since instantaneous gender detection errors counter-balance for reasonable long time intervals. Evaluations carried on manually annotated TV news resulted in WSTP estimation errors below 0.6% for archives longer than 30 minutes.

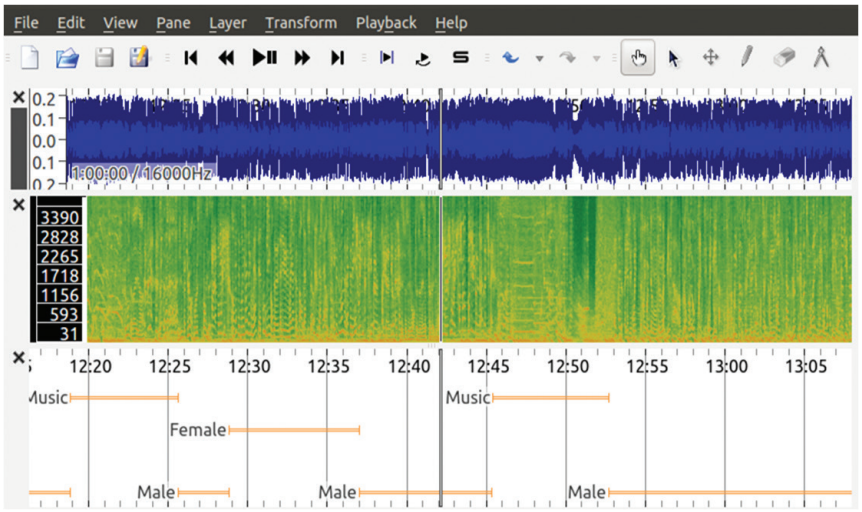


Figure 1. Interactive display of automatic speech segmentation output. First layer is the raw audio signal, second layer is the time-frequency representation of the signal. Last layer is the automatic prediction of the proposed speech segmenter in Music, Male and Female excerpts.

10 David Doukhan, Jean Carrive, Félicien Vallet, Anthony Larcher and Sylvain Meignier, 'An open-source speaker gender detection framework for monitoring gender equality', *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, 2018.

11 **https://github.com/ina-foss/inaSpeechSegmenter**.

12 Félicien Vallet, Jim Uro, Jérémy Andriamakaoly, Hakim Nabi, Mathieu Derval and Jean Carrive, 'Speech Trax: A Bottom to the Top Approach for Speaker Tracking and Indexing in an Archiving Context', *LREC 2016*.

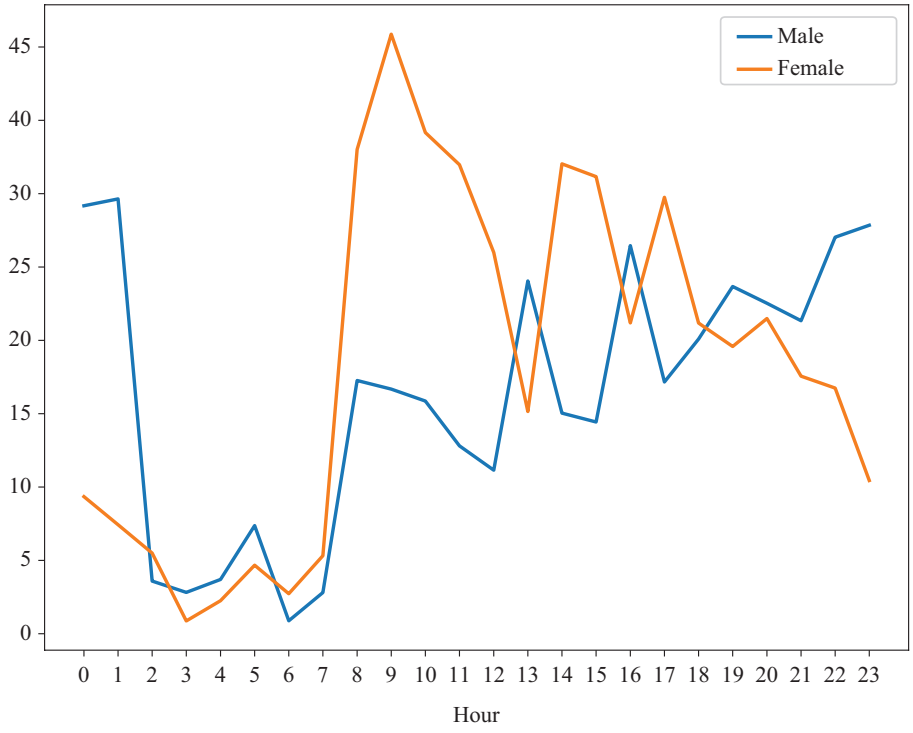


Figure 2. Hourly speaking time obtained with our proposed gender equality monitor

2.3 Analysis Biases

InaSpeechSegmenter's was trained and evaluated using only adult voices. Automatic detection of children voices is known to be challenging, and very few language resources allow us to train and evaluate systems aimed at detecting these voices.[13] Since low acoustic differences exist between male and female children, automatic recognition systems generally focus on the recognition of child category regardless of their gender. Moreover, children voices used in cartoons, dubbed programs, or radio advertisements, are generally performed by adult actors, who do not necessarily have the same sex than the character they're dubbing. Informal observations showed *children voices* encountered in audiovisual documents were either detected as music (cartoon characters with very theatral voices), or as women voices. This analysis bias was minimized by excluding from analysis children's interest channels, as well as TV time-slots associated to child-oriented programs (6-9AM).

13 Björn Schuller, et al., 'Paralinguistics in speech and language - State-of-the-art and the challenge'. *Computer Speech & Language, 27(1), 4-39*, 2013.

Video 1. Brigitte Lecordier: the French voice of several famous male characters: Son Goku, Kevin Arnold (*The Wonder Years*), Oui Oui, Le Petit Nicolas, Booba...

Another bias to our analysis is related to the content of our evaluation material, which is mostly composed of news and debates, and do not contains fictions. Once again, this limitation is due to the scarcity of annotated speech resources related to fictions. Therefore, the error rate of our system was estimated using informal evaluations and looked similar to the rates obtained on the news and debates corpus.

3 Corpora

Since 2001, INA has been collecting all the streams broadcast on a selection of TV and radio stations. Saving 24-hour streams is the result of political choices specific to France, which, to our knowledge, have no equivalent in the world. National audiovisual heritage safeguarding policies implemented in other countries are limited to a limited selection of programs. This French specificity allows the implementation of comprehensive approaches, based on the systematic analysis of all programs broadcast, resulting in a corpus of 700.00 hours of audiovisual documents. At the time of this analysis, TV feeds prior to 2010 were still stored on DVD and were not yet accessible via servers. For this reason, the analyzes performed on TV streams only covered the period 2010-2018.

3.1 French Radio Corpus

Table 1 presents the 21 national radio station selection used for describing WSTP variations in French radio streams. Radio stations are described according to their status and their content. Content is based on Médiamétrie (French audience measurement company) classification for all stations except Radio Sud.[14] Status is described using CSA's taxonomy.[15] This classification distinguishes public radios on the one hand, and five private radio categories on the other hand, each category being indicated by a letter from A to E:

14 Médiamétrie, 'Grilles radio d'été, l'audience de la Radio en France en Juillet-Août 2017,' 2017.

15 CSA, *Hertzian private radio stations,* Conseil supérieur de l'audiovisuel **http://en.www.csa.fre05d.systranlinks.net/Radio/Les-stations-de-radio/Les-radios-FM/Les-stations-de-radio-privees-hertziennes**.

Category A – Associative radio services performing a mission of social communication of proximity
Category B – independent local or regional radio services that do not broadcast nationally-recognized programs
Category C – local or regional radio services broadcasting the program of a national thematic network
Category D – national thematic radio services
Category E – general-purpose radio services with a national vocation

Radio station selection includes 7 public and 14 private radio stations. Analyzes were carried on streams broadcasted between 2001 and 2018. They were restricted to the time slots between 5 AM and midnight, in order to include largest audiences peaks in the analyses: 6-9 AM for the majority of radio stations,[16] 9PM-midnight for stations aimed at a teenage audience.[17]

Radio streams were split in one hour-long excerpts which were randomly selected for analysis with a 18% selection probability in order to lower computation time.

The amount of data kept for the description of expression rate is therefore corresponding to the amount of channels (21), multiplied by the number of hours considered per day (19), the number of days analyzed (18 years) and the random selection rate (18%); accounting for about 486.000 hours of audio content (55 years of continuous stream).

Table 1. *National radio channels selection used in the current study*

Name	Available since	Status	Content
Chérie FM	2002	Category C, D	Music
Europe 1	2001	Category E	Generalist
France Bleu	2001	Public	Generalist
France Culture	2001	Public	Thematic
France Info	2001	Public	Thematic
France Inter	2001	Public	Generalist
France Musique	2001	Public	Thematic
Fun Radio	2001	Category C, D	Music
MOUV	2012	Public	Music
Nostalgie	2001	Category C, D	Music
NRJ	2002	Category C, D	Music
Radio Classique	2009	Category D	Thematic
Radio France Internationale	2001	Public	Thematic
RFM	2002	Category C, D	Music
Rire et Chansons	2009	Category C, D	Music
RMC	2001	Category E	Generalist
RTL 2	2002	Category C, D	Music
RTL	2001	Category E	Generalist
Skyrock	2001	Category C, D	Music
Sud Radio	2012	Category B, E	Generalist (*)
Virgin Radio	2008	Category C, D	Music

16 CSA, *La représentation des femmes (shortened)*.
17 Reiser, *L'image des femmes (shortened)*.

3.2 French TV Corpus

Table 2 presents the 22 TV channels selection used for describing women speaking time percentage variations in French televisual streams. This selection includes 7 public and 15 private channels. It has been realized in order to consider channels associated to the largest audiences, as well as channels associated to targeted specialities (news, sports, history, music, content aimed at a women audience).

Analyzes were carried on streams broadcasted between 2010 and 2018. They were restricted to the time slots between 10 AM and midnight, corresponding to TV audiences above 10%.[18]

TV streams were split in one hour-long excerpts. These were randomly selected for analysis with a 27% selection probability in order to lower computation time.

The amount of data kept for the description of expression rate is therefore corresponding to the amount of channels (22), multiplied by the number of hours considered per day (14), the number of days analyzed (9 years) and the random selection rate (27%); accounting for about 270.000 hours of audio content (30 years of continuous stream).

Table 2. *TV channels selection used in the current study*

Name	Status	Available since	Content
Arte	Public	2010	French-German channel promoting culture and arts
BFM TV	Private	2010	National news
Canal+	Private	2010	Generalist with focus on movies and sports
Chérie 25	Private	2013	Generalist aimed at a female audience
C8	Private	2013	Generalist (Formerly D8 until September 5, 2016).
L'Équipe TV	Private	2013	Sports
Eurosport	Private	2010	Sports
France 24	Public	2011	International news broadcasted in 4 languages and 180 countries
France 2	Public	2010	Generalist. Second most watched channel in France
France 3	Public	2010	Generalist with regional and national programs: 24 regional editions et 44 local editions
France 5	Public	2010	Generalist with focus on educational and documentary
France Ô	Public	2010	Generalist with focus on overseas France
Histoire	Private	2011	History
CNews	Private	2010	National news (Formerly I-Télé until February 27, 2017)
La Chaîne Info	Private	2010	National news
LCP/Public Sénat	Public	2010	Politics (French National Assembly and Senate) and news
M6	Private	2010	Generalist. Third most watched channel in France
NRJ 12	Private	2010	Generalist with focus on entertainments
Téva	Private	2011	Generalist aimed at female and familial audience
TF1	Private	2010	Généralist. Most watched channel in France and Europe
TMC	Private	2010	Generalist
W9	Private	2010	Generalist with focus on music and entertainments

18 CSA, *Les chiffres clés de l'audiovisuel français, édition du premier semestre 2015*, Conseil supérieur de l'audiovisuel, 2015.

4 Global Analysis of Audiovisual Streams

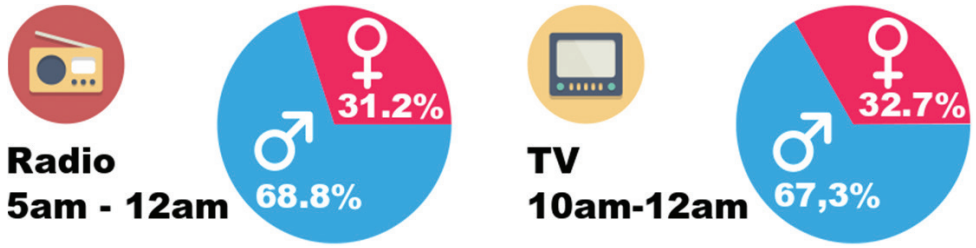


Figure 3. Mean Women and Men speech-time from 2010 to 2018

Massive analysis of TV and radio programs broadcasted between 2010 and 2018 show a strong imbalance in the distribution of speech time between women and men (Figure 3). On both mediums, men's speech-time is at least twice as long than women's speech-time. Women's speech-time percentage is slightly larger on TV (32,7 %) than on radio (31,2 %).

Average results per channel observed between 2010 and 2018 are presented in Figures 4 and 5. Channels are displayed given two dimensions. Abscissa stands for the *speech percentage*, defined as 100-*music percentage*. Ordinate is *women speaking time percentage* (WSTP), defined as 100-*men speaking time percentage*.

In TV corpus, speech percentage varies between 62.5 and 93.8 %. It is minimal for W9 (music channel) and maximal for news channels, and to a lesser extent: sport channels. Larger variations of the speech percentage are observed in the radio corpus, ranging from 15.4% (RFM) to 95.5% (France Info). Two groups of stations can be done based on the value of the speech rate. *Musical stations* refers to the group of 12 stations having more than half of musical content (9 stations having more than two third of music). *Non-Musical stations* to the remaining stations having more speech than music, including 7 stations having more than 77% of speech.

TV and radio channels are all associated to speaking time percentages larger for men than for women, except Cherie FM, which is a musical station with a low amount of speech (19.2 %).

Non-musical radio is associated to a higher women expression rate in public than in private stations. Lowest women expression rates in radio are obtained on Skyrock (16.2%, hip-hop music and teenage audience) and RMC (16.9 %, large amount of sport).

In TV, WSTP varies between 7.4 et 47.9 %. Speaking time percentage is therefore higher for male than for female in all considered TV-channels. It is minimal for sport channels (Eurosport, L'Équipe, and in a lesser extent CANAL+), and slightly lower than average in channels specialized in cultural or educational programs (Histoire, Arte, France 5). Private news channels (I-Télé, LCI, BFM-TV) have similar characteristics (speech percentage between 89.7 and 90.7 %, WSTP between 33.5 and 35.4%). Only four channels were associated to women expression rates above 40%: the two channels aimed at women audience (Téva et Chérie 25), France 24 et M6.

The case of France 24 is of particular interest: this channel presents the highest women speech time percentage (44.8 %) among TV stations that do not focus explicitly on women-oriented programs. This singularity is quite paradoxical since France 24 is the international showcase of French TV, contributing to convey a distorted image of French audiovisual landscape, where global women and men speech-time is similar.

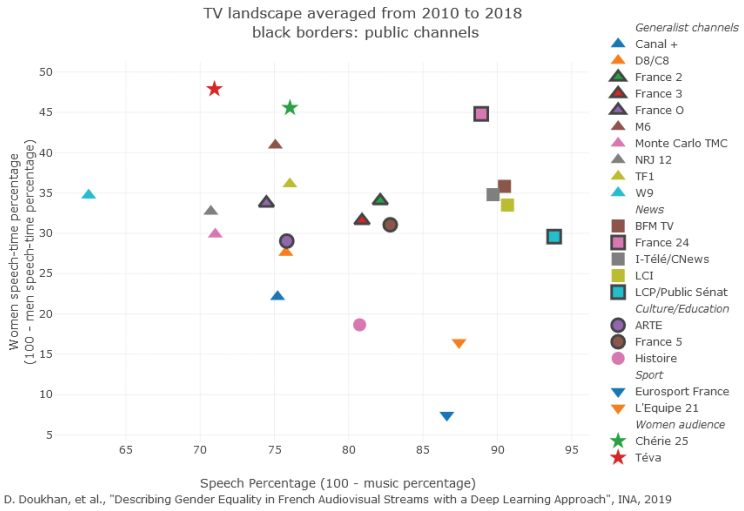


Figure 4. Speech percentage and women speaking time percentage observed in TV-channels from 2010 to 2018. Click here for the dynamic version of this figure.

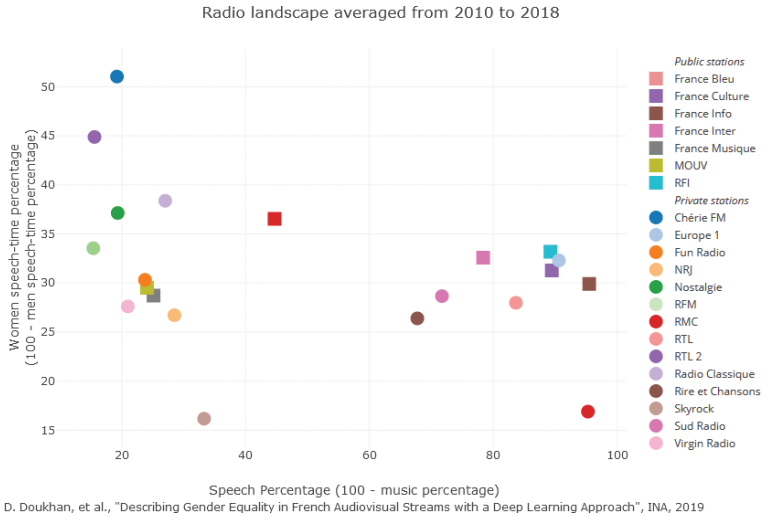


Figure 5. Speech percentage and women speaking time percentage observed in radio stations from 2010 to 2018. Click here for the dynamic version of this figure.

5 Yearly Evolution of Women Expression Rate

Figures 7 and 6 presents the evolution of women speaking-time percentage (WSTP), on TV from 2010 to 2018, and on radio from 2001 to 2018. Results are presented together with the median expression rates observed on public and on private channels. Linear regression procedures were used to associate to each channel an annual slope of WSTP evolution, as well as a p-score allowing us to describe the statistical significativeness of the corresponding slope.[19] Statistically significative evolutions were defined as those associated to a p-score < 0.05.

19 **https://docs.scipy.org/doc/scipy/reference/generated/scipy.stats.linregress.html**.

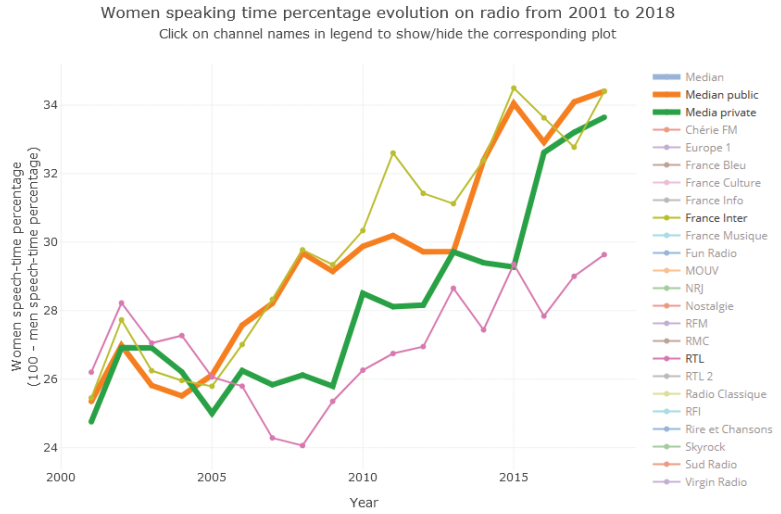


Figure 6. Women speaking time evolution on Radio from 2001 to 2017. Click here for the dynamic version of this figure.

Median WSTP evolution in radio channels has increased regularly since 2004. It increased from 25.1 % in 2001 to 34.4 % in 2018. In other words, the French radio landscape changed from a configuration where men's speaking-time was three time longer than women's to a configuration where men's speaking-time is twice longer than women's. While these proportions are still highly unbalanced, this shows fast changes in French radio landscape. While private radio station have slightly lower WSTP than public, the annual evolution of WSTP of about 0.5 % point is observed in public and private stations. Statistically significant evolutions were found for 17 stations out of 21. Three stations were associated with a negative WSTP evolution: Radio Classique (-1.02 / year), RMC (-0.52 % / year) and Skyrock (-0.24 %/year). Stations associated to the highest WSTP evolutions are Sud Radio (+1.7% / year), France Musique (+1.08 % / year) and RTL2 (+0.95 % / year).

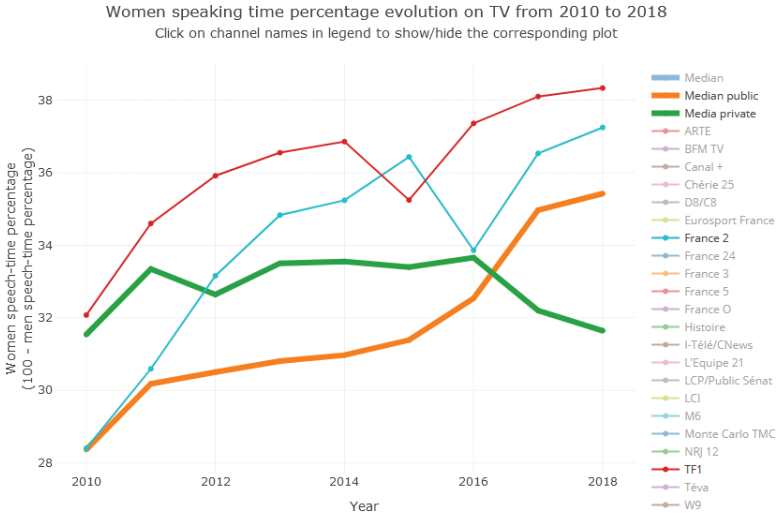


Figure 7. Women speaking time evolution on TV from 2010 to 2018. Click here for the dynamic version of this figure.

While median WSTP evolution is statistically significative in TV channels (+0.53 % / year), several differences can be observed between public and private channels. Median WSTP evolution is statistically significative for public channels (+0.79 %/ year) and increased from 28.4 % in 2010 to 35.4 % in 2018. No significative evolution was found for the median WSTP of private channels. In 2010, WSTP used to be larger in private channels (31.5 %) than in public channels. Since 2017, WSTP is larger in public channels. Statistically significative evolutions of WSTP were found for 11 TV stations out of 22. These evolutions were found to be negative for two stations stations L'Equipe 21 (-2.44 % / year) and I-Télé/CNews (-1.18 %/ year). Largest WSTP evolutions were found for France 5 (+1.28 % / year), Histoire (+1.01 %/ year), LCP/Public Sénat (+0.94 % / year) and France 2 (+0.94 % / year).

6 Hourly Analyzes

Analyzes presented below describe variations of audiovisual content, on a hourly basis. Music and speech time estimates were obtained from archives broadcasted from 2010 to 2018, excluding weekends, civil and school holidays.

Higher and lower audience time-slots were approximated, inspired by CSA studies.[20] High audience time-slots were defined as 3-hours long contiguous time-range: 6-9AM for radio and 7-10 PM for TV.

6.1 Speech and Music Hourly Percentages in Musical Radio Stations

Speech and music percentage hourly variations are necessary to put in context hourly WSTP variations. These descriptions allows us to tell if a given WSTP extreme is related to a time-slot associated to a reasonably large amount of speech, which is of special importance for musical radio stations which may have very scarce amount of speech according to the time-slot considered.

Figure 8 presents speech-percentage hourly variations observed in the twelve identified *musical radio* stations, having more than 50% of music in their programs. Median speech percentage is associated to its largest values during high-audience slots with a peak of 59.5 % between 8 and 9 AM. Three main broadcasting strategies can be observed from the data. A first group of channels is associated to a peak of speech in early morning (Chérie FM, Nostalgie) and a lower amount of speech the rest of the time. The second group is associated to two peaks of speech: a first one in the early morning, and a second one in the early or late evening. This most representative stations of this group are those targeting teenage audiences: Skyrock, NRJ and Fun Radio. This group includes to a lesser extent: Virgin Radio, RFM and RTL 2. Last group has three main speech peaks: a first one in the early morning, a second one at lunch time and a third one in evening: it includes Radio Classique and le MOUV and in a lesser extent France Musique.

It has to be noted that some time-slots of musical stations contain a very low amount of music. This percentage is below 4% from 7 to 9 AM on Radio Classique, and below 15% from 9PM to 12AM on Skyrock.

20 CSA, *La représentation des femmes (shortened)*.

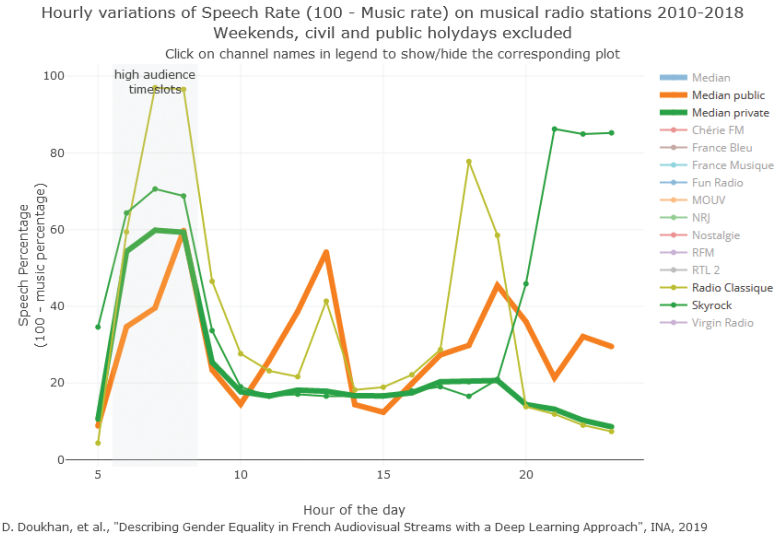


Figure 8. Speech time percentage (100 - Music time percentage) variations in musical radio stations from 5 AM to midnight. Click here for the dynamic version of this figure.

6.2 Women Hourly Expression Rate

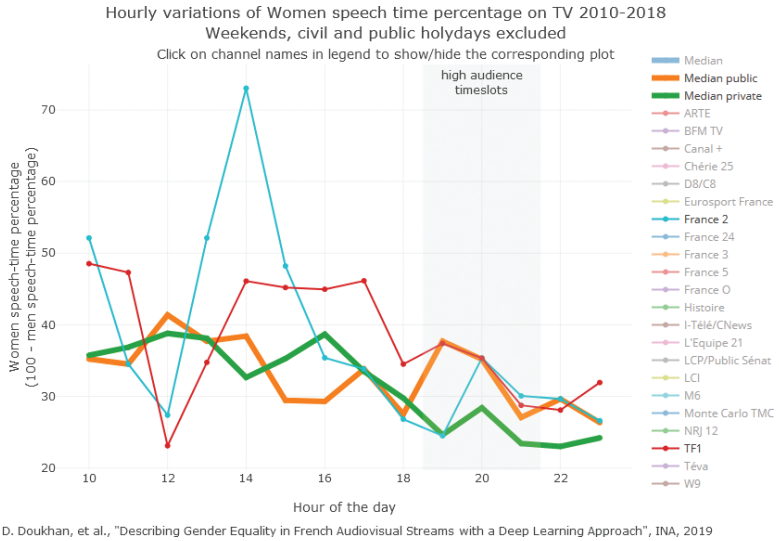


Figure 9. Women speaking time variations on TV from 10 AM to midnight. Click here for the dynamic version of this figure.

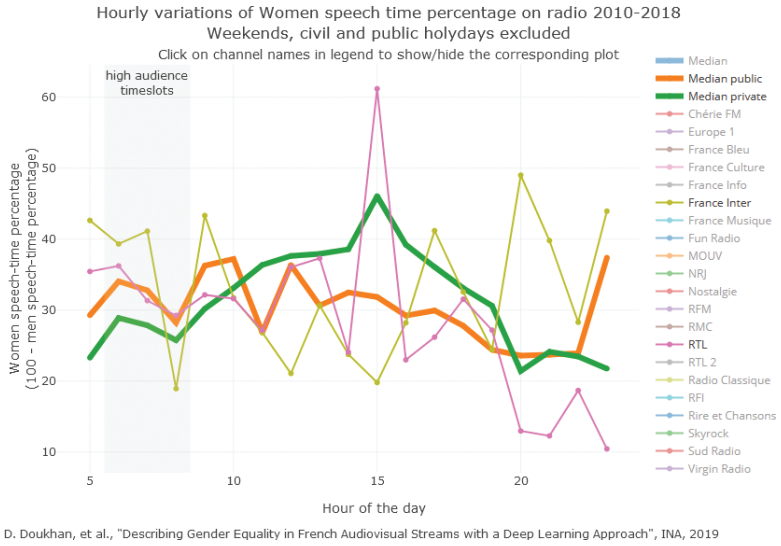


Figure 10. Women speaking time variations on TV from 5 AM to midnight. Click here for the dynamic version of this figure.

Figures 9 and 10 present WSTP variations over hours, for measures obtained from 2010 to 2018; excluding weekends, civil and school holidays.

Median WSTP were lower during high audience time slots for private TV channels (-7.8 %) and private radio stations (-4.5 %). They were similar for public TV channels (+0.29 %) and slightly higher for public radio stations (+1.63 %).

TV stations associated to the largest negative WSTP differences between high and low audience time-slots are France 2 (-10 %), NRJ12 (-8.1 %) and Chérie 25 (-7 %), while those associated to the largest positive differences are France 3 (+8.7 %), ARTE (+6.2 %) and Histoire (+3.2 %).

Observed WSTP variations between high and low audience time-slots are stronger for radio stations. Stations associated to largest negative differences are Radio Classique (-14.6 %), Virgin Radio (-14.2 %), NRJ (-12.9 %) and Fun Radio (-10.6 %). Stations associated to the largest positive differences are France Musique (+12.1 %), RTL 2 (+9.7 %) and RMC (+5.4%).

7 Regional Disparities

7.1 French Regional TV News Corpus

Regional TV corpus contains the entire collection of *19/20* regional editions broadcasted on France 3 in 2016. 19/20 is a regional news program broadcasted in *prime-time* having large audience parts varying between 14 and 21%[21] [22] [23].

21 Florian Guadalupe, *Audiences access : Le "19/20" de france 3 leader, "c à vous" et "28 minutes" en forme,* Pure médias, 2016.

22 *France 3 : Le 19/20 au dessus des 20% de parts d'audience cette semaine,* Pure médias, 2010.

23 Benjamin Meffre, *Audiences access : Nagui leader en baisse, "le 19/20" devant "dna", "c à vous" en forme*, Pure médias, 2017.

24 regional editions of 19/20 are broadcasted simultaneously from 7 to 7:30 PM. They may be interrupted by advertisements or weather reports and are followed by the national edition of 19/20. Regional editions correspond to France's metropolitan division prior to 2016 in 21 administrative regions, with the addition of Corsica. Region Provence-Alpes-Côte d'Azur has two distinct editions (Provence-Alpes, Côte d'Azur) as well as Rhône-Alpes (Rhône, Alpes).

Regional news editions were detected in TV streams using automatic image processing methods, based on the recognition of the specific banner displayed (Figure 11). This strategy allows robust detection of regional news start and end times. It also allows us to discard programs that may interrupt regional news: advertisements, weather forecasts, special national editions, substitute programs used in case of strike or a technical issue. Each regional edition was associated to 132 hours of programs per year, accounting for a total of 3200 hours.

Figure 11. Regional edition of 19/20 newscast screenshot, containing a banner displaying 19/20 followed by the name of the considered region or locality.

7.2 Global Analysis of French Regional TV News

Figure 12 details WSTP observed in the 24 regional editions of 19/20 news program. This percentage varies between 25.89 and 52.9%. Alsace and Nord-Pas-de Calais are the only editions associated to an expression rate larger for women than for men. Seven editions out of 24 have approximately equal speaking time percentages per gender (between 45 and 55%): Alsace, Nord-Pas-de-Calais, Ile-de-France, Picardie, Bretagne, Provence-Alpes, Languedoc-Roussillon. Women expression rate was found to be lower than a third in four regional editions: Lorraine, Midi-Pyrénées, Auvergne and Aquitaine.

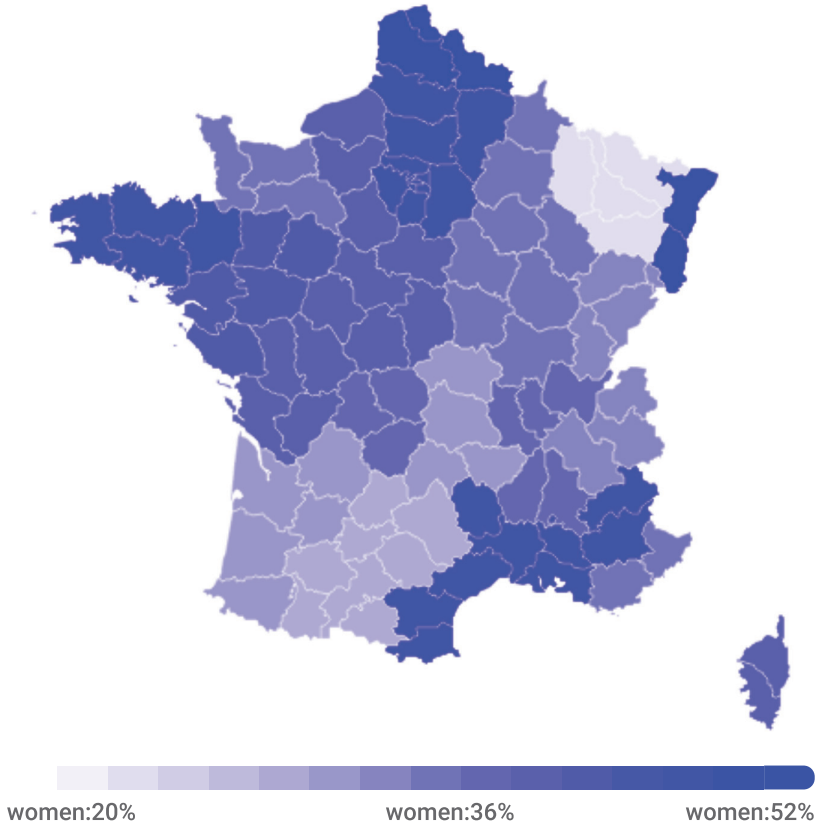


Figure 12. Women speech-time percentage in 19/20 regional news, broadcasted by France 3 in 2016. Click here for the dynamic version of this figure.

A correlation analysis was realized between WSTP and the number of inhabitants per departments. Non-parametric Spearman's test was used for the estimation of this correlation.[24] Moderate positive (rho=0.453) and statistically significant (pvalue < 10^{-5}) correlation suggest that departments with larger amount of inhabitants are generally associated to larger women expression rate in 19/20 programs.

7.3 Women Speech Time and Percentage of Female Presenters in Regional News

For each regional news program, the identity of the presenter was obtained from manual documentation procedures, realized within INA's archiving missions. The exploitation of this data allowed us to obtain the percentage of female presenters occuring in 19/20 regional news broadcasted in 2016, shown in Figure 13.

24 Daniel Zwillinger and Stephen Kokoska, *CRC standard probability and statistics tables and formulae,* Chapman & Hall, 2000.

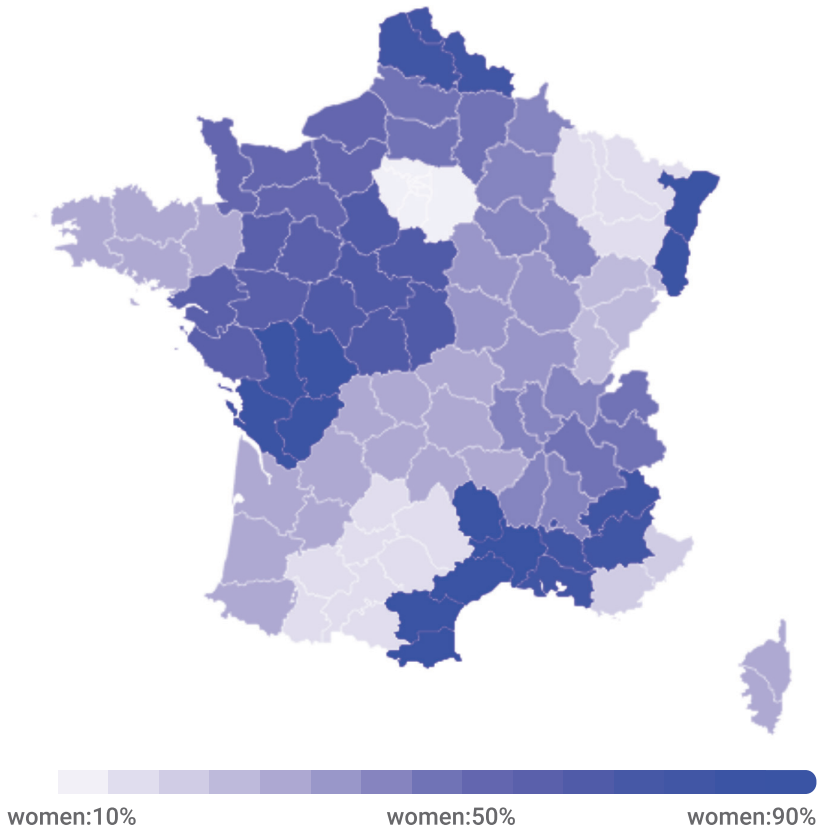


Figure 13. Percentage of female presenters in regional TV news. Click here for the dynamic version of this figure.

The variations of female presenters is much wider than the variations of WSTP observed in regional news. 11 regional editions have a larger proportion of women presenters, with percentages above 80 % found for Languedoc Roussillon, Alsace and Poitou Charentes. A single regional edition was found to have more than 80 % of male presenters: Paris Île de France.

The relations between women speech-time percentage and the percentage of women presenters allows us to describe further the complexity of image equality issues in TV. The relatively high WSTP found in Alsace, Poitou Charentes and Languedoc Roussillon is therefore mainly due to a large presence of women presenters, but may hide a low amount of non-presenter women speaking during regional news. Conversely, despite its low amount of women presenters, Paris Île-de-France managed to have similar speech-times percentages for men and women.

8 Conclusion

This study presented describes gender equality in French media, based on the description of Women Speaking Time Percentage (**WSTP**). This estimate of equality was obtained using automatic machine learning procedures allowing us to detect music, men and women speech in audio streams. This automation allowed the estimation of men and

women speaking time percentage on 700.000 hours of audiovisual documents, which would be unfeasible through manual analysis.

Several tendencies were highlighted: Men speaking time percentage is about twice that of than women's in French TV and Radio, but used to be three time bigger in 2004. We show WSTP is much lower on private channels during high-audience time slots. We also show that WSTP is lower in sport and cultural TV channels, which correlates manual gender equality studies:[25] *Le sérieux d'Arte se fait donc avec les hommes ; l'émotion de M6 se fait avec les femmes* (seriousness of Arte is done with men; emotion of M6 is done with women).

While WSTP is a metric well suited to automatic extraction, *presence rate* (amount of distinct speakers), which is a reference metric in several manual studies, is still challenging to obtain through automatic procedures. We believe these two metrics should be used together in order to improve the description of equality issues in media. An informal comparison of these two metrics is presented, based results obtained in 2016 through our approach and CSA estimates.[26] This comparison should be treated with caution, since the channels list considered in our 2 studies have few differences. In 2016, we found a WSTP of 33.6 % for TV and 32.9 % on radio, while CSA reported women presence rates of 40 % in TV and 36 % on radio. During high-audience time slots on radio, we found a WSTP of 30.1% while CSA reported a women presence rate of 35%. These observations suggest WSTP estimates are lower than women presence rates. Similar conclusions can be obtained for the channels reported in the CSA study: C8, Canal+, France 2, France 3, France O, and W9. This observation may be relevant with respect to Reiser & Gresy's study showing women speech turns are shorter than men's,[27] in other words: having the same amount of men and women in programs does not guarantee equal amount of speech-time.

The opportunities of exploitation of the massive amount of data obtained through our methodology are numerous, and may benefit from the use of additional structured data allowing to put WSTP into context: channel governance and budget, detailed audience metrics, program description, identity of presenters, regional statistics… The work required to constitute such structured data is huge and goes far beyond the scope of our study. Consequently, we released the results of our analyses in open-data, which are now freely accessible through **data.gouv.fr**, which is the open platform for French public data.[28] The proposed dataset contain additional data which we was not described in this study corresponding to 21 radio and 34 TV stations, broadcasted from 1995 to March 2019, accounting for more than 1 million of analyzed time-slots. Data is presented as a raw csv database containing 1 million of entries, each of them corresponding to the duration of music, women speech and men speech for a particular hour, together with meta-data (private or public channel, civil and school holidays, week-day,...). We hope this data will help further research in digital humanities and contribute to a better understanding of gender equality issues in media.

Although this would be tempting to compare the results obtained in France to other countries, some technological locks need to be addressed. The first lock is related to the gender detection system, which is language dependent (see section 2). The management of audiovisual documents in other languages would require us to build and evaluate similar systems of each language. This could be done with STEM efforts, and may require the creation of annotated data to be used for training and evaluation. The second lock is much more difficult to address. As stated in section 3, France is to our knowledge the only country in the world which records and archives the integrality of its audiovisual streams (since 2001): most countries do archive only a specific selection of programs. Unrecorded audiovisual streams are definitely lost, limiting the knowledge that could be obtained through the improvement of automatic audiovisual analysis procedures. Consequently,

25 Reiser, *L'image des femmes (shortened)*.

26 CSA, *La représentation des femmes (shortened)*.

27 Reiser, *L'image des femmes (shortened)*.

28 **https://www.data.gouv.fr/fr/datasets/temps-de-parole-des-hommes-et-des-femmes-a-la-television-et-a-la-radio/**.

comparisons across countries would require a definition of methodologies optimizing the use of available archives for each considered country.

Speaking time percentage per gender is a *surface* equality descriptor, which is not sufficient to fully describe gender representations in media. Further STEM research efforts are required to prove the viability of additional descriptors obtained through automatic analysis of audiovisual documents. Among these descriptors, we're currently working on face detection and gender classification systems, aimed at comparing the differences between speech-time and facial exposition in TV, and helping in the estimation of the *presence rate.* We also built early prototypes based on modern speech-to-text softwares, in order to obtain information related to speech transcriptions. Such information allowed us to obtain *Identification rate* estimators (number of oral references to men and women characters), and may probably be extended to the description of the topics covered by men and women.[29] Unlike speech-time estimation, image and speech transcription processings are costly and require much longer processing times (ten to sixty times as much). Large scale analyses based on these descriptors will require significantly larger computational power, and will hopefully benefit from the future advances of computing hardware.

9 Acknowledgments

This work was partly supported by the European Union's Horizon 2020 Research and Innovation programme via the project MeMAD under Grant Agreement No 780069.

Biographies

David Doukhan is a research engineer working at INA (French National Audiovisual Institute). He received his Ph.D. in Computer Science from Paris Sud University in 2013 before working two years for LIMSI-CNRS and IRCAM as post-doctoral researcher. His research deals with speech analysis, music information retrieval, corpus linguistics, machine learning, multimedia collections management and big data.

Zohra Rezgui is a statistics engineering student from University of Carthage. She is currently carrying out her graduation internship at INA's Research and Innovation Department, which covers studying and comparing time on screen between men and women using computer vision techniques.

Géraldine Poels holds a PhD in history and is a specialist in audiovisual media. Author of a history of television viewers in France (Les Trente Glorieuses du télépectateur, INA, 2015), she is currently in charge of the scientific promotion of INA's collections and, as such, plays an active role in the development of new scientific uses for audiovisual archives.

29 David Doukhan, Zohra Rezgui, Géraldine Poels and Jean Carrive, 'Estimer automatiquement les différences de représentation existant entre les femmes et les hommes dans les médias', *Journée DAHLIA (DigitAL Humanities and cuLtural herItAge): Informatique et Humanités numériques: quelles problématiques pour quels domaines?,* 2019.

In 2000, Jean Carrive received a PhD in Computer Sciences from Pierre & Marie Curie University in collaboration with INA (French National Audiovisual Institute), which he later joined as a research engineer. He has conducted several research projects in the areas of automatic analysis of audiovisual and multimedia content. In the field of digital humanities, he is particularly interested in the application of audiovisual content analysis technologies for historical and heritage uses. He is now deputy head of INA's Research and Innovation Department.

VIEW Journal of European Television History and Culture Vol. 7, 14, 2018
DOI: 10.18146/2213-0969.2018.jethc156
Publisher: Netherlands Institute for Sound and Vision in collaboration with Utrecht University, University of Luxembourg and Royal Holloway University of London.

B.5 INA's paper in La revue des Médias [4]

This paper describes the results obtained using inaSpeechSegmenter on massive amount of data. This paper met with a large media success in the French audiovisual landscape.

À la radio et à la télé, les femmes parlent deux fois moins que les hommes

Article · March 2019

CITATIONS
2

READS
20

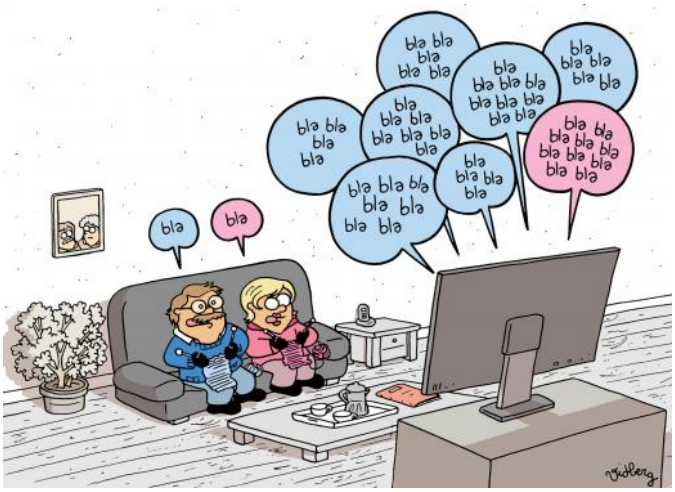
1 author:

David Doukhan
Institut national de l'audiovisuel
31 PUBLICATIONS **271** CITATIONS

SEE PROFILE

À la radio et à la télé, les femmes parlent deux fois moins que les hommes

ARTICLE par David DOUKHAN • Publié le 04.03.2019 • Mis à jour le 07.03.2019



Pour la première fois, une intelligence artificielle a mesuré le temps de parole des femmes et des hommes dans les médias français. Réalisée sur 700 000 heures de programmes, soit le plus gros volume de données jamais analysé au monde, cette étude dresse un état des lieux, chaîne par chaîne, depuis 2001.

Sommaire

Récemment déclarée « grande cause du quinquennat », la question de l'égalité entre les femmes et les hommes est un sujet qui suscite de nombreux débats et passions. La description objective des différences de représentation existant entre les hommes et les femmes dans les médias est un enjeu sociétal majeur, nécessaire pour rationaliser les débats citoyens et orienter les décisions politiques.

Lorsque j'ai été amené à présenter mes travaux dans des congrès scientifiques, j'ai à de nombreuses reprises rencontré des hommes qui m'ont déclaré « Les femmes parlent trop » lorsque je leur ai décrit ma thématique de recherche. Ces mêmes hommes ont été les premiers surpris lorsqu'ils ont découvert les conclusions des analyses présentées dans cette étude.

Plusieurs études fondées sur l'analyse quantitative du contenu des médias ont été menées ces dernières années pour décrire ces différences de représentation. On pourra citer le Global Media Monitoring Project (GMMP), ou encore les analyses menées par le Conseil supérieur de l'audiovisuel (CSA), qui sont fondées entre autres sur la description du taux de présence, défini comme le

pourcentage d'hommes et de femmes présents à l'antenne. Ce taux peut être présenté par catégories correspondant au statut des intervenants (reporter, expert, journaliste, témoin, invité politique...), ou encore aux sujets traités (économie, santé, éducation…).

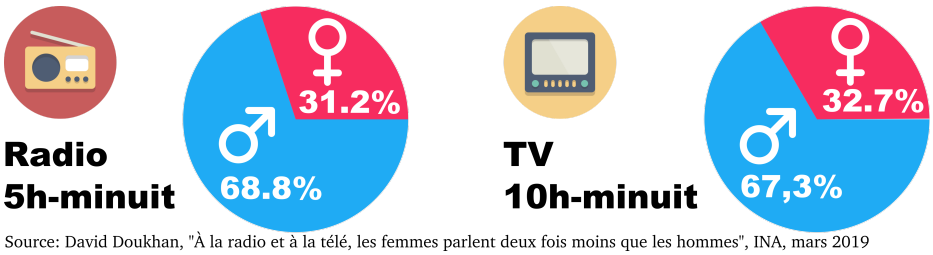
Les études fondées sur l'analyse du taux d'expression, défini comme le pourcentage de temps de parole attribué à des femmes ou à des hommes, sont rares. On pourra néanmoins citer le rapport public coordonné par Michèle Reiser et Brigitte Grésy, *L'image des femmes dans les médias* (commandé par le secrétariat d'État à la Solidarité), fondé sur l'analyse d'émissions diffusées le 15 mai 2008 (temps d'analyse compris entre 6 minutes et 3 heures par chaîne), ainsi qu'une étude menée à titre expérimental par le CSA de la Communauté française de Belgique, *Les représentations femmes-hommes sont-elles influencées par les générations ?*, fondée sur l'analyse de 36 heures de programmes collectées pendant une semaine.

La mesure manuelle du temps de parole est coûteuse, ce qui explique que les analyses produites à ce jour ont été réalisées sur des échantillons de taille limitée. Ces limitations peuvent se traduire par des biais d'analyse, tendant à généraliser des situations observées sur des échantillons non représentatifs, et ne permettant pas d'apprécier l'évolution de ces phénomènes observés dans le temps.

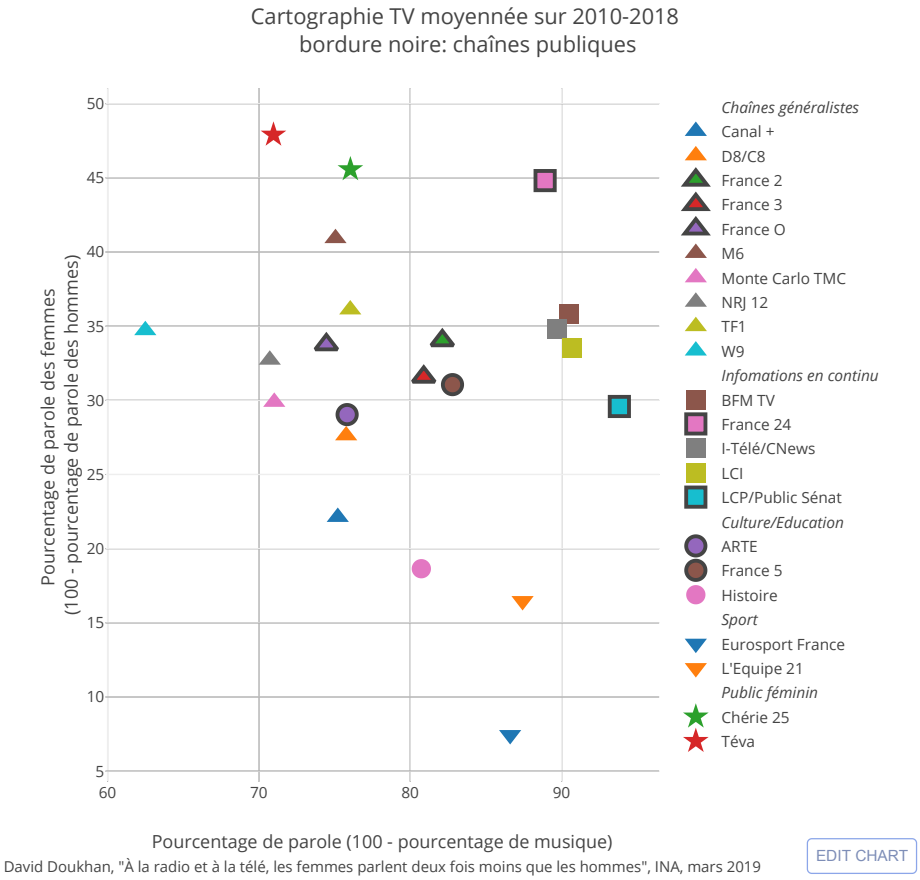
Cette étude, conduite à l'INA (Institut national de l'audiovisuel), propose d'utiliser une intelligence artificielle pour estimer le temps de parole des femmes et des hommes automatiquement (la méthodologie complète est à retrouver en bas de l'article) Ce type d'approche vise à traiter une masse de documents beaucoup plus importante et réduire les effets de biais liés à la taille de l'échantillon d'analyse. Les phénomènes observés sont également susceptibles d'orienter des analyses qualitatives, et de contribuer ainsi à créer de nouvelles connaissances en sciences humaines.

Un déséquilibre à géométrie variable

L'analyse massive des fonds collectés de 2010 à 2018 dresse un état des lieux caractérisé par un fort déséquilibre entre le temps de parole utilisé par les hommes et les femmes. Les prises de parole des femmes à la télévision représentent moins d'un tiers du temps de parole alloué (32,7 %). Ce constat est encore plus frappant à la radio où il n'est que de 31,2 %, comme le montre la figure ci-dessous.



Source: David Doukhan, "À la radio et à la télé, les femmes parlent deux fois moins que les hommes", INA, mars 2019



David Doukhan, "À la radio et à la télé, les femmes parlent deux fois moins que les hommes", INA, mars 2019

Le taux de parole (axe horizontal) correspond au pourcentage de parole observé sur les chaînes proportionnellement aux autres phénomènes (musique, bruits d'environnement, applaudissements...). Il est indépendant du sexe du locuteur. Un taux de parole égal à 0 signifie que la chaîne ne diffuse que de la musique, et un taux égal à 100 correspond à une chaîne ne diffusant que de la parole. Le taux de parole varie entre 62,5 et 93,8 %. Il est minimal sur W9 (chaîne musicale) et maximal pour l'ensemble des chaînes d'information ainsi que pour les chaînes de sport.

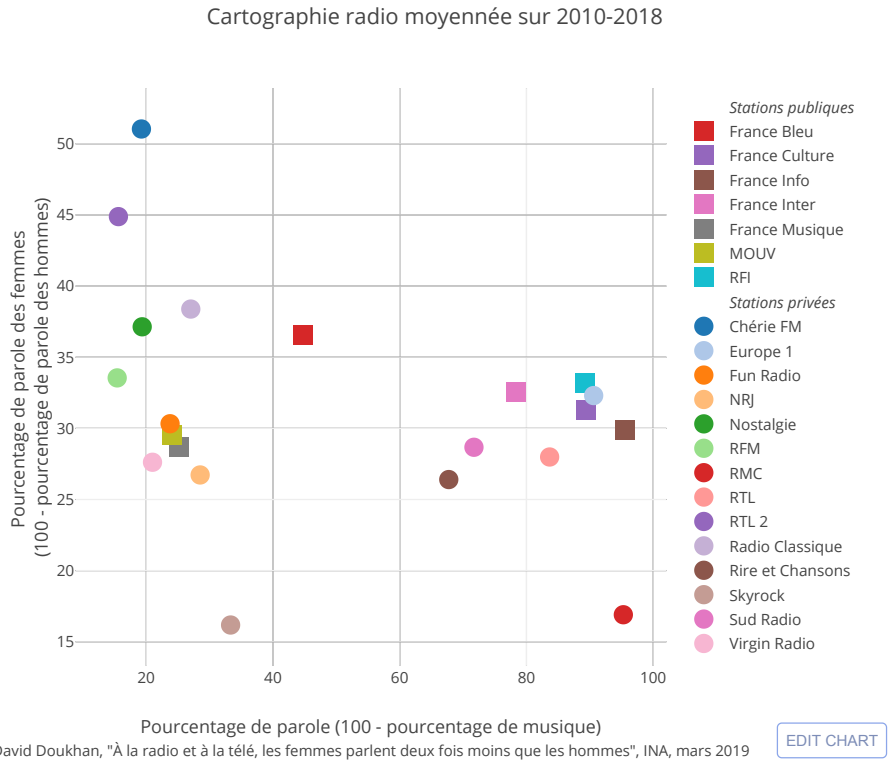
Le taux d'expression des femmes (axe vertical) correspond au pourcentage de temps de parole attribué à des femmes. Un taux égal à 0 signifie que les femmes ne parlent pas à la télévision, un taux égal à 50 signifie que le temps de parole des femmes et des hommes est égal, un taux égal à 100 signifie que seules des femmes parlent à l'antenne.

Quelle que soit la chaîne considérée, le taux d'expression des femmes est inférieur à 50 %, ce qui signifie que sur l'ensemble des chaînes TV, les hommes parlent davantage que les femmes. Les chaînes où les femmes ont un temps de parole maximal sont les chaînes visant un public féminin (Téva, Chérie 25). Les chaînes diffusant du contenu sportif, sont quant à elles associées aux plus faibles taux d'expression des femmes : 7,4 % pour Eurosport et 16,5 % pour La chaîne L'Équipe. Le taux d'expression des femmes est plus faible pour les chaînes à programmation culturelle ou éducative (Histoire, Arte, France 5) que pour les chaînes à contenu généraliste. Au sein des chaînes généralistes, les plus forts taux d'expression des femmes sont observés sur M6 (40,9 %) et sur TF1 (36,1 %), tandis que le plus faible taux est observé sur Canal+, chaîne qui se démarque par une plus grande programmation de contenu sportif. Ces observations renforcent les analyses réalisées à plus petite échelle en 2008 dans le rapport d'État coordonné par Michèle Reiser et Brigitte Grésy, qui observent que « le sérieux d'Arte se fait donc avec les hommes ; l'émotion de M6 se fait avec les femmes »...

Le cas de la chaîne France 24 est particulièrement intéressant. Cette chaîne d'information internationale présente, après Téva, le deuxième plus haut taux d'expression des femmes (44,8 %). Cette singularité est assez paradoxale, sachant que cette chaîne est une vitrine du paysage audiovisuel français à l'étranger. Les disproportions de temps de parole entre les sexes observées sur l'ensemble des chaînes nationales y étant beaucoup plus faibles, France 24 contribue ainsi à véhiculer l'image faussée d'un pays où la répartition du temps de parole entre hommes et femmes est relativement équitable...

En résumé, on observe donc que sur l'ensemble des chaînes de télévision, les femmes parlent assez nettement moins longtemps que les hommes. Leur parole est davantage présente dans les chaînes s'adressant à un public féminin, et elle est extrêmement faible dans les chaînes diffusant du contenu sportif. Le temps de parole des femmes est comparable sur les chaînes d'information en continu et les chaînes généralistes, et légèrement inférieur sur les chaînes diffusant des programmes culturels ou éducatifs.

Chérie FM, seule station de radio où les femmes parlent davantage que les hommes



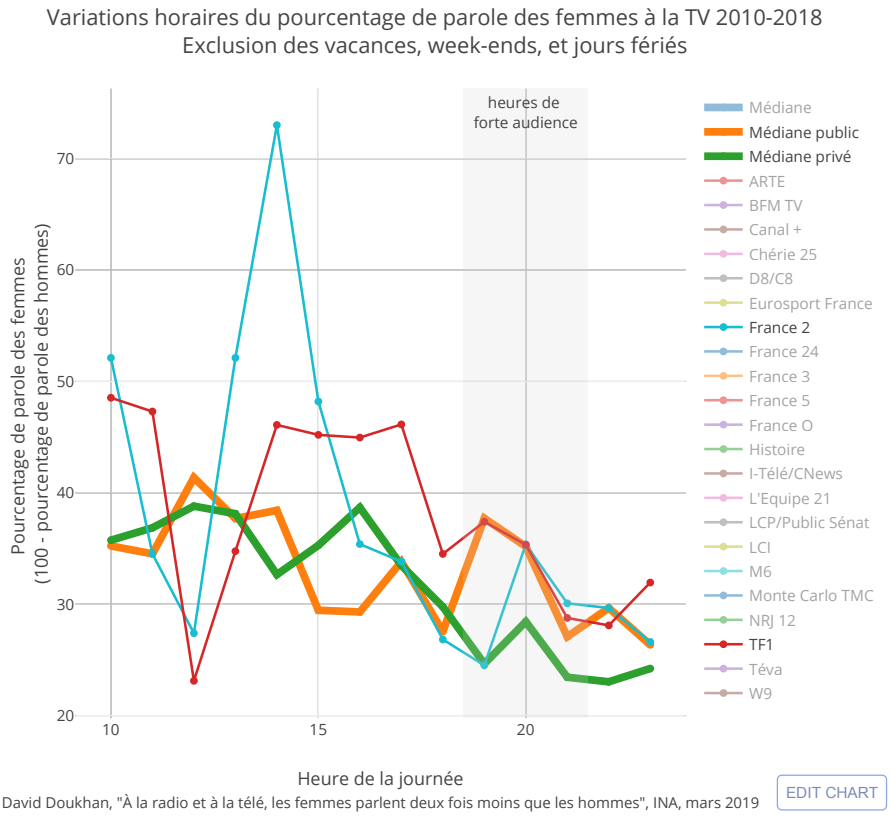
David Doukhan, "À la radio et à la télé, les femmes parlent deux fois moins que les hommes", INA, mars 2019

À la radio, on constate beaucoup plus de disparité dans le pourcentage de parole (par opposition au pourcentage de musique), avec d'une part, les stations à programmation majoritairement musicale (RFM, RTL2, Chérie FM...) qui diffusent plus de deux tiers de contenu musical, et d'autre part, des stations contenant principalement de la parole (France Inter, France Info, RMC). Pour les stations dont le contenu est principalement de la parole, le taux d'expression des femmes maximal est de 33,2 % sur Radio France International, et minimal sur RMC (16,9 %), une station diffusant beaucoup de contenu sportif.
La seule station contenant plus de parole de femmes que d'hommes est Chérie FM, ce qui reste marginal sachant que cette station diffuse plus de 80 % de musique. Skyrock quant à elle, est de loin la station musicale où les femmes ont le moins la parole, avec un taux d'expression de seulement 16,2 %.

Sur les chaînes privées, les femmes parlent moins aux heures de forte audience

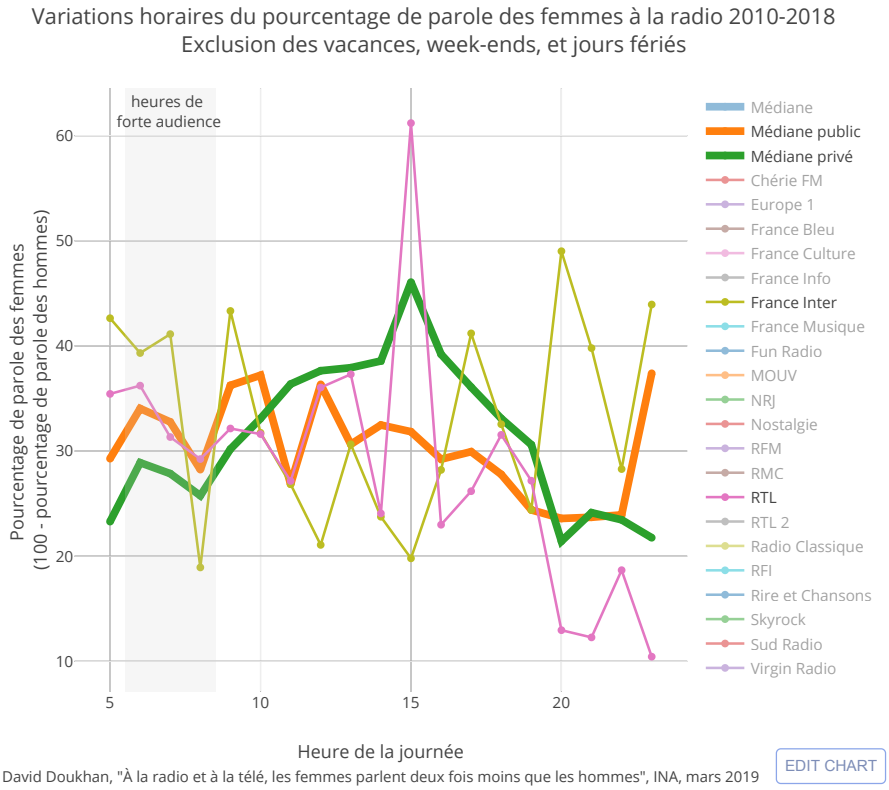
Quand parlent les femmes ? Cette partie décrit les différences de taux d'expression des femmes observées en fonction des heures de la journée, observées sur la période 2010-2018. Pour plus de cohérence, nous avons exclus de ces mesures les programmes diffusés lors des week-ends, jours fériés et vacances scolaires. Nous nous sommes inspirés de la méthodologie du CSA, définissant les heures de forte audience comme le créneau 6 h – 9 h à la radio et 19 h – 22 h à la télévision.

Sur les chaînes de télévision privées, les femmes parlent moins aux heures de forte audience



Aux heures de forte audience, le taux d'expression des femmes baisse en moyenne de 3,2 %[1] sur les chaînes de télévision privées, ainsi que sur les chaînes publiques (-1,1 %). Les plus fortes augmentations du temps de parole des femmes aux heures de forte audience sont observées pour les chaînes France 3 (+8,7 %) et Arte (+6,2 %). Les plus fortes baisses pour France 2 (-10 %), NRJ 12 (-8,1 %), et Chérie 25 (-7 %). Sur le créneau *prime time* (19 h -20 h), le taux d'expression des femmes est de 37,7 % sur les chaînes publiques contre 24,6 % sur les chaînes privées.

Sur les stations de radio privées, les femmes ont un plus faible taux de parole aux heures de forte audience



Aux heures de forte audience, le taux d'expression des femmes baisse de 2,7 % sur les chaînes privées, et augmente de 0,3 % sur les chaînes publiques.
Les plus grosses disparités sont observées pour Radio Classique (-14,6 %), Virgin Radio (-14,2 %), NRJ (-12,9 %), France Musique (+12,1 %)

Des évolutions positives, en particulier sur les chaînes publiques

Dans les parties précédentes, nous avons mis en évidence de fortes disparités existant dans la répartition du temps de parole entre les femmes et les hommes, plus particulièrement aux heures de forte écoute. Ces inégalités de représentation tendent à se réduire au fil des années.

Le temps de parole des femmes à la radio a augmenté de 9,3 % de 2001 à 2018

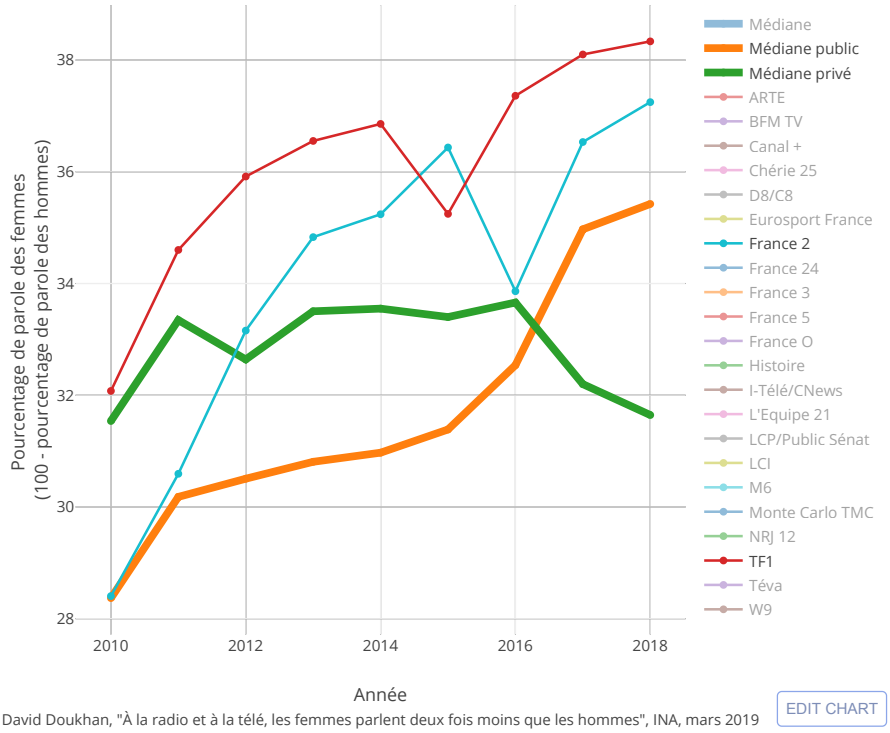
Évolution du pourcentage de parole des femmes à la radio de 2001 à 2018

David Doukhan, "À la radio et à la télé, les femmes parlent deux fois moins que les hommes", INA, mars 2019 EDIT CHART

Le taux médian d'expression des femmes est passé de 25,1 % en 2001 à 34,4 % en 2018, soit une augmentation d'environ 0,5 % par an pendant 18 ans. Les femmes parlent davantage sur les stations publiques : l'écart maximal ayant été observé en 2015 (+ 4,7 %). Les évolutions les plus importantes (statistiquement significatives) sont observées pour France Musique (+17,1 %), Europe 1 (+17 %), RTL2 (+13,1 %), Sud Radio (+10,9 %), Fun Radio (+10,7 %), Nostalgie (+10 %), France Culture (+9,6 %), France Inter (+8,8 %). Cette tendance à la hausse du temps de parole des femmes n'est pas observée sur l'ensemble des chaînes, notamment RMC (-7,7 %), Radio Classique (-5,8 %) ou encore Skyrock (-2,1 %).

Le temps de parole des femmes à la télévision a augmenté 4,7 % de 2010 à 2018.

Évolution du pourcentage de parole des femmes à la TV de 2010 à 2018



David Doukhan, "À la radio et à la télé, les femmes parlent deux fois moins que les hommes", INA, mars 2019 EDIT CHART

Le taux d'expression des femmes, toutes chaînes confondues, a évolué de 30,4 % en 2010 à 35,1 % en 2018. Cette évolution est particulièrement visible sur les chaînes publiques (+7 %). Jusqu'en 2016, c'est sur les chaînes privées que les femmes avaient le plus de temps de parole, mais cette tendance n'est plus vraie depuis 2017.

Les évolutions les plus importantes concernent France 5 (+9,3 %) et France 2 (+8,8 %). On observe également une forte baisse de la présence des femmes sur la chaîne sportive L'Équipe (-10,1 %) et sur CNews (-8 %).

Quelques disparités régionales : le 19/20 de France 3

L'exploration de l'intégralité des éditions régionales du *19/20* de France 3 — émissions d'informations diffusées en *prime time* de 19 h à 19 h 30 — sur l'année 2016, permet d'établir des disparités régionales du taux d'expression par sexe. Cette analyse a nécessité de mettre au point une intelligence artificielle fondée sur l'analyse d'images, permettant de détecter le début et la fin des éditions du *19/20*, en se basant sur la reconnaissance des bandeaux caractéristiques de ces JT. En raison des évolutions de l'aspect de ces bandeaux et de la complexité de cette recherche, celle-ci n'a pu porter que sur une seule année. Elle offre toutefois une intéressante topographie de la situation sur une année pleine.

Temps de parole des femmes et inégalités territoriales

Pourcentage de temps de parole des femmes observé dans les 24 éditions régionales d'actualités du 19/20 de France 3, diffusées en 2016.

David Doukhan, "À la radio et à la télé, les femmes parlent deux fois moins que les hommes", INA, mars 2019

Share

Le pourcentage de parole attribué aux femmes dans les 24 éditions régionales varie entre 25.89 % et 52.9 %. L'Alsace et le Nord-Pas-de Calais sont les seules éditions pour lesquelles le temps de parole des femmes est supérieur à celui des hommes. Sept éditions sur 24 sont associées à un temps de parole par sexe à peu près égal (compris entre 45 et 55 %) : Alsace, Nord-Pas-de-Calais, Île-de-France, Picardie, Bretagne, Provence-Alpes, Languedoc-Roussillon. Quatre éditions régionales sont associées à un temps de parole des femmes inférieur à un tiers : Lorraine, Midi-Pyrénées, Auvergne, Aquitaine.

S’emparer des vastes possibilités d’exploitation de ces données : un enjeu sociétal

L’étude présentée dans cet article a permis de décrire une partie des différences de représentation existant entre les hommes et les femmes dans les médias, en se focalisant sur la répartition du temps de parole.

Les analyses présentées pourraient être approfondies en croisant les temps de parole avec d'autres sources de données : budget de fonctionnement des chaînes, données d'audience détaillée, grilles de programmation, événements sociaux et politiques… L'étendue des possibilités d'exploitation de ce matériau est vaste, et dépasse très largement les limites de cette étude. Dans le cadre de la politique d'ouverture des données entreprise par l'INA, l'ensemble des indicateurs générés lors de cette étude a été mis à disposition sur data.gouv.fr, la plateforme ouverte des données publiques françaises. Elles pourront ainsi être utilisées par l'ensemble des acteurs de l'audiovisuel, les chercheurs en sciences humaines et sociales (SHS), journalistes, instances politiques, ainsi que l'ensemble des citoyens.

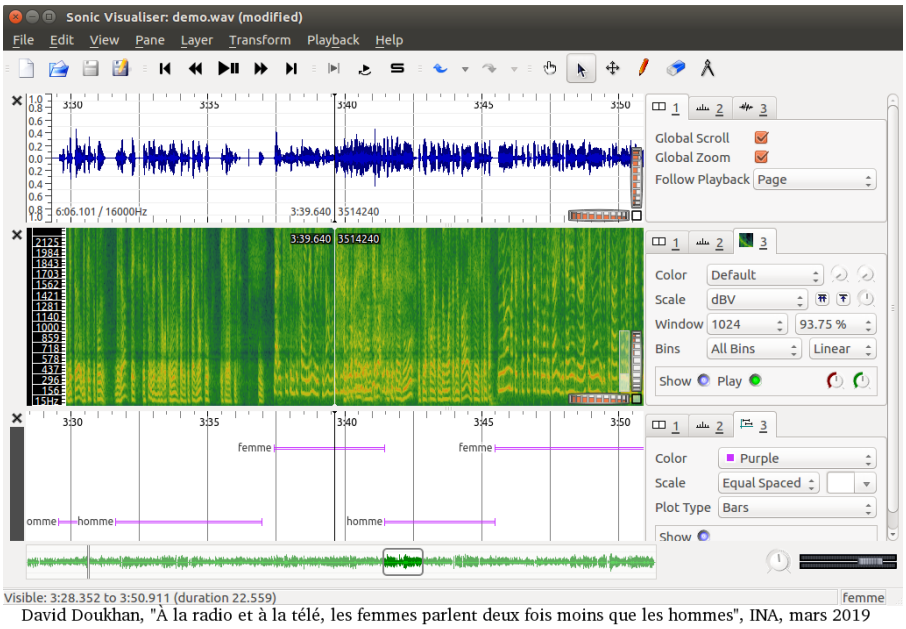
La description des différences de représentation est complexe et on ne peut, bien évidemment, pas se contenter de ne l'aborder qu'à travers le prisme du taux d'expression. Des efforts de recherche supplémentaires sont nécessaires pour mettre au point des systèmes fiables d'analyse automatique permettant de détecter les thèmes évoqués par les différents intervenants, les personnalités dont il est question, ou encore le rôle des locuteurs.

C'est à cette fin que le projet de recherche Gender Equality Monitor a récemment été soumis à l'Agence nationale de la recherche (ANR). Ce projet est porté par un consortium pluridisciplinaire composé de deux professionnels de l'audiovisuel (INA, Deezer), deux laboratoires STIC (Limsi / CNRS, Lium) et trois laboratoires SHS spécialisés dans l'étude du genre et des médias (Carism, Lerass, ENS Lyon-CMW). Les retombées de ce projet devraient contribuer à une meilleure compréhension du fonctionnement des médias et à alimenter les débats citoyens sur la place des hommes et des femmes dans la société.

Méthodologie

Comment l'IA a permis de distinguer les voix d'hommes et de femmes

L'analyse des fonds télévision et radio a été réalisée à l'aide d'InaSpeechSegmenter, un logiciel libre issu d'une collaboration entre le service de recherche de l'INA et le laboratoire d'informatique de l'université du Mans. Il s'agit d'un logiciel d'analyse acoustique, permettant de localiser les zones de parole au sein de documents multimédias, et de déterminer le sexe des *locuteurs* (personnes qui parlent). La voix chantée n'est pas à proprement parler de la parole : elle est catégorisée comme musique et exclue des analyses.



David Doukhan, "À la radio et à la télé, les femmes parlent deux fois moins que les hommes", INA, mars 2019

Exemple d'analyse automatique : la figure du haut représente le signal audio brut. La figure du milieu correspond à la représentation « temps-fréquence » du signal sonore, qui sert de base à son analyse automatique. La figure du bas représente les portions de signal qui ont été attribué à des hommes (H) et à des femmes (F).

La distinction entre les voix d'hommes et de femmes est modélisée à l'aide de réseaux de neurones profonds (*deep learning*) : une famille de méthodes issues de l'intelligence artificielle. Les réseaux de neurones nécessitent d'être entraînés à partir d'exemples pour pouvoir reconnaître des concepts : plus les exemples sont variés et plus le système résultant est performant. Les exemples d'entraînement utilisés sont issus du dictionnaire de locuteurs de l'INA, qui est à notre connaissance la plus grande base de locuteurs annotée manuellement à partir de données audiovisuelles (télévision et radio). Ce dictionnaire est composé de 32 000 extraits sonores diffusés de 1957 à 2012, correspondant à 1 780 locuteurs et 494 locutrices distincts, s'exprimant en français.

Le système résultant permet d'estimer le taux d'expression des hommes et des femmes avec une erreur inférieure à 0,6 %. Plus la durée analysée est importante, et plus l'estimation du taux d'expression est robuste.

700 000 heures de programmes TV et radio analysées

Depuis 2001, l'INA collecte l'intégralité des flux diffusés sur une sélection de stations TV et radio. La sauvegarde de flux captés 24 h/24 est le résultat de choix politiques spécifiques à la France, qui n'ont, à notre connaissance, pas d'équivalent dans le monde. Les politiques de sauvegarde du patrimoine audiovisuel national mises en place dans les autres pays sont limitées à une sélection restreinte de programmes. Cette spécificité française permet la mise en place d'approches exhaustives, fondées sur l'analyse systématique de l'ensemble des programmes diffusés.

L'échantillon d'analyse sélectionné pour cette étude est composé de 22 chaînes de télévision et 21 stations de radio. La sélection des stations a été réalisée de manière à faire apparaître les chaînes associées aux plus fortes audiences, ainsi que certaines chaînes thématiques ciblées (informations, sport, histoire, musique, contenu visant un public féminin). Les créneaux horaires sélectionnés correspondent aux heures de plus forte audience : ils ont été fixés entre cinq heures et minuit pour la radio, et dix heures et minuit pour la télévision.

L'échantillon radio est composé de 7 stations publiques (France Bleu, France Culture, France Info, France Inter, France Musique, Le Mouv' (désormais Mouv'), Radio France internationale/RFI) et 14 stations privées (Chérie FM, Europe 1, Fun Radio, Nostalgie, NRJ, Radio Classique, RFM, Rire et Chansons, RMC, RTL, RTL 2, Skyrock, Sud Radio, Virgin Radio). Pour réduire le temps de calcul, les flux ont été découpés en tranches d'une heure et 486 000 heures ont sélectionnées aléatoirement pour être analysés, soit environ 18 % de la totalité des programmes diffusés sur ces créneaux.

L'échantillon TV est composé de 7 chaînes publiques (Arte, France 24, France 2, France 3, France 5, France Ô, LCP/Public Sénat) et 14 chaînes privées : BFM TV, Canal+, Chérie 25, La chaîne L'Équipe, Eurosport, Histoire, CNews (anciennement I-Télé), C8 (anciennement D8), LCI, M6, NRJ 12, Téva, TF1, TMC et W9. À l'époque où a eu lieu cette analyse, achevée en janvier 2019, les flux TV antérieurs à 2010 étaient conservés sur DVD et n'étaient pas encore accessibles via des serveurs. Pour cette raison, les analyses réalisées sur les flux TV n'ont porté que sur la période 2010-2018. Un total de 270 000 heures de contenu TV a été analysé, soit environ 27 % de la totalité des programmes diffusés sur ces créneaux.

Crédit :
INA. Illustration : Martin Vidberg.

Ce projet a reçu des financements du programme Horizon 2020 de l'Union européenne pour la recherche et l'innovation sous le Grant Agreement N°780069 (projet MeMAD, http://memad.eu).

1. les valeurs sont exprimées en pourcentage absolu ou point de pourcentage

B.6 AALTO's paper in ICASSP 2020 conference [5]

This paper describes AALTO experiments in speaker-aware training of attention based end-to-end ASR. It has been published in the IEEE ICASSP 2020 conference.

SPEAKER-AWARE TRAINING OF ATTENTION-BASED END-TO-END SPEECH RECOGNITION USING NEURAL SPEAKER EMBEDDINGS

Aku Rouhe *Tuomas Kaseva* *Mikko Kurimo*

Aalto University, Department of Signal processing and acoustics

ABSTRACT

In speaker-aware training, a speaker embedding is appended to DNN input features. This allows the DNN to effectively learn representations, which are robust to speaker variability.

We apply speaker-aware training to attention-based end-to-end speech recognition. We show that it can improve over a purely end-to-end baseline. We also propose speaker-aware training as a viable method to leverage untranscribed, speaker annotated data.

We apply state-of-the-art embedding approaches, both i-vectors and neural embeddings, such as x-vectors. We experiment with embeddings trained in two conditions: on the fixed ASR data, and on a large untranscribed dataset. We run our experiments on the TED-LIUM and Wall Street Journal datasets. No embedding consistently outperforms all others, but in many settings neural embeddings outperform i-vectors.

Index Terms— end-to-end speech recognition, speaker-adaptation, speaker-aware training, speaker embedding

1. INTRODUCTION

Speaker independent speech recognition models attempt to find a suitable compromise for all speakers. Speaker adaptation lets the speaker independent models readjust to each speaker, by leveraging some speaker specific information. Fine-tuning the parameters of a DNN for each speaker separately would be computationally expensive and difficult, because the models are black boxes with a very large number of parameters. In hybrid HMM-DNN speech recognition, an effective speaker adaptation method is appending speaker embeddings to the input features, and having the DNN learn to use this information [1]. We call this speaker-aware training [2].

In the attention-based encoder-decoder end-to-end (AED) speech recognition [3, 4], fine-tuning the parameters for each speaker's acoustic characteristics is even more complicated, since the DNN also implements an implicit language model. In AED ASR, only a few speaker adaptation approaches have been proposed so far [5, 6], and to the best of our knowledge, speaker-aware training has not been applied to AED ASR.

In this work, we investigate speaker-aware training of AED ASR. We compare three different speaker embedding types: *i-vectors* [7], and two neural methods: *x-vectors* [8] and a *thin-ResNet* neural network architecture [9]. We present three main contributions.

Firstly, we show competitive word error rate improvements on the TED-LIUM [10] and Wall Street Journal (WSJ) [11] corpora. In our experiments, speaker-aware training outperforms an additional, end-to-end trained sequence summary network component [5].

Secondly, we propose speaker-aware training as a viable strategy to incorporate untranscribed data into the AED paradigm. Similar to the popular method of incorporating an external language model in shallow fusion [12], speaker-aware training is not purely end-to-end. The speaker embeddings are trained separately. This is beneficial, since the speaker embeddings can then be trained on untranscribed speech data, which only needs speaker annotations. We exploit state-of-the-art speaker embeddings trained on the large VoxCeleb datasets [13, 14]. We also compare these VoxCeleb embeddings with speaker embeddings trained only on the smaller fixed ASR datasets.

Thirdly, we show that neural embeddings outperform i-vectors in some settings, although no embedding consistently outperforms all others. In concurrent work, Rownicka et al. [15] explore neural embeddings for speaker-aware training of HMM-DNN ASR, but do not find improvements over i-vectors.

As a part of this work we present our findings in applying typical post-processing methods to the speaker embeddings: mean subtraction, dimensionality reduction and length normalization. Particularly, in our experiments, we find L2-normalization to be crucial. We show that neural embeddings may not need any other post-processing steps.

2. RELATED WORK

Only a few speaker adaptation methods have been proposed in AED ASR. In [5], a sequence summary network is added to the model architecture, and in [6], additional learning objectives are used to regularize the output of a speaker-dependent network.

Speaker-aware training has been applied to connectionist temporal classification models (CTC) [16, 17], which are trained with an end-to-end criterion. However, CTC models

are an implicit HMM [18], and in practice they are typically applied similarly to hybrid HMM-DNN models [19].

Rownicka et al. [15] are the first to present results where neural embeddings are used in speaker-aware training. In their work, i-vectors still outperform neural embeddings in speaker-aware ASR. The authors argue that compared to neural embeddings, i-vectors capture more additional information, other than speaker identity. In speech recognition, this other information, such as channel effects, are beneficial. However, also in concurrent work, Raj et al. [20] use probing tasks to show that x-vectors also encode channel information.

3. SPEAKER EMBEDDINGS

In speaker verification, the task is to distinguish whether two speech segments are spoken by the same speaker or not. Typically a speaker embedding extractor is trained separately, and then the embeddings are used as features for a binary classifier (such as cosine distance scoring or probabilistic linear discriminant analysis). [21] In this work, we use three embedding methods: i-vectors, x-vectors, and *thin-ResNet* embeddings.

I-vectors are based on factor analysis of Gaussian Mixture Model (GMM) supervectors. Thorough overviews of the method can be found in the related literature [22], but we omit it here for brevity.

3.1. Neural speaker embeddings

In the context of all-neural AED ASR, neural speaker embeddings could enable further work in fine-tuning the embeddings in the end-to-end ASR task. Furthermore, recently in speaker verification, neural speaker embeddings have been shown to outperform i-vectors [13, 8, 9].

X-vectors are a popular neural speaker embedding type. They use TDNN-layers, and are trained in speaker classification. After training, the embedding is extracted as the output of the second to last layer before the softmax. For details, see the original paper [8].

Thin-Resnet **embeddings** are also trained in speaker classification. Unlike the x-vectors, the model is a (2D) convolutional neural network, which operates directly on spectrograms, and includes an L2-normalization layer, after which the embedding is extracted. More details can be found in the original paper [9].

4. ATTENTION-BASED END-TO-END NEURAL SPEECH RECOGNITION

Attention-based encoder-decoder end-to-end neural speech recognition models [3, 4] have become a popular alternative to conventional HMM-based systems. These models directly transcribe speech to text. No language model or external lexicon is needed, but they are learned implicitly.

	VoxCeleb 1	VoxCeleb 2
Training hours	352	2442
Training speakers	1251	6112

Table 1. Details of the speaker embedding training datasets

Typically, the encoder is a pyramidal stack of bi-directional LSTM layers (BLTSM). The decoder is typically a unidirectional LSTM, and uses an attention mechanism to extract a relevant weighted sum of the encoder outputs at each output step.

The decoder of an attention-based E2E model learns an implicit language model. However, it is only trained on the transcripts of audio data. Much larger text-only datasets can be leveraged to train an external language model.

The language model probabilities are then interpolated with the ASR probabilites during decoding in shallow fusion.

4.1. ESPnet encoder

The encoder in our model is slightly different from the standard approach above. Our implementation comes from the ESPnet toolkit[23]. The encoder is trained in a multi-task setting, by adding a CTC-decoder in parallel. The CTC-decoder is also used in inference, by interpolating the likelihoods from both decoders. [24]

ESPnet also implements a hybrid convolution and BLSTM-based encoder. However the convolution operation does not make sense for the appended speaker embeddings, because the embedding dimensions do not have any ordered structure. Therefore, in our experiments, we do not use the convolutional front-end.

5. EXPERIMENTS

5.1. Data

The untrascribed data x-vector and i-vector embedding extractors are trained on VoxCeleb [13] and VoxCeleb2 [14]. Furthermore, for the x-vectors, a large amount of data augmentation is applied [8]. The *thin-ResNet* model is only trained on VoxCeleb2. Table 1 lists the salient dataset details.

We run the speech recognition experiments on the TED-LIUM (release 2) [10] and Wall Street Journal (si-284 training set) [11] datasets. Table 2 shows the dataset characteristics. The fixed data speaker embeddings are trained on these ASR datasets' respective training sets.

5.2. Embedding models

For the untranscribed VoxCeleb data embeddings, we use pretrained models available online [25, 26]. Table 3 compares these embeddings in a speaker verification task. Neural embeddings outperform i-vectors.

	TED-LIUM	WSJ
Training hours	207	82
Dev hours	1.6	1.1
Test hours	2.6	0.7
Training speakers	1242	283
Dev speakers	8	10
Test speakers	10	8

Table 2. Details of the speech recognition corpora used in this paper

For the embeddings trained on the ASR data (we call this the fixed data scenario), we adjust the embedding model hyperparameters to better suit these datasets, which are smaller than the VoxCeleb datasets. For the i-vector model, we choose 512 full-covariance Gaussians in the universal background model, and 100-dimensional i-vectors, without LDA. For x-vectors, the configuration is otherwise the same as the original VoxCeleb x-vector model [8], but the embedding size is halved to 256, and the number of training epochs doubled to 6. We arrived at these values using a heuristic: we pick the values which yield the highest adjusted rand index (ARI) [27], when clustered using spherical K-means. Spherical K-means is L2-normalized, which was shown to be important in earlier experiments. High ARI should reflect consistent embeddings, which we believe should help in ASR, and the procedure is computationally inexpensive. We first optimized the values on the TED-LIUM data. On WSJ, the TED-LIUM i-vector configuration resulted in a perfect 1.0 ARI, so we decided to simply reuse the TED-LIUM-tuned configurations without further optimization.

Furthermore, in the fixed data setting, we do not test the *thin-ResNet* embeddings, because the implementation was not readily available in the Kaldi toolkit.

	EER
i-vector [25]	5.3
x-vector [25]	3.1
thin-ResNet [9]	3.22

Table 3. The pretrained VoxCeleb speaker embeddings compared in speaker verification, on the VoxCeleb 1 test set. In speaker verification, the common performance metric is equal error rate (EER). It is the error rate at which there are equally many false acceptances and false rejections.

5.3. Post-processing the embeddings

For the i-vector and x-vector embeddings, we test standard post-processing procedures: subtracting the training set mean, dimensionality reduction by LDA, and using L2-normalization. The LDA transform is trained on the speech recognition training set; we reduce the dimensionality to 200, which is the x-vector default. The *thin-ResNet* output is already L2-normalized, which would be undone by any post-processing, so therefore we use the *thin-ResNet* outputs as they are.

Table 4 shows x-vector and i-vector results without LDA, and either subtracting the global mean or not. These experiments indicate that with x-vectors the mean subtraction hurts performance and with i-vectors it helps. We keep these choices for all x-vector and i-vector experiments.

In the Kaldi toolkit [28] (which we use for feature dumping), the default is to normalize to $\sqrt{d}$, where d is the dimensionality of the embedding. In preliminary experiments, we found that it is crucial to normalize to length 1. Otherwise, the embeddings only hurt performance. Thus in all of reported results, we have applied L2-normalization to unit length.

TED-LIUM	Test		Dev	
	No LM	+LM	No LM	+LM
x-vector	**20.1**	**17.2**	**20.9**	**18.1**
x-vector subtract mean	20.5	**17.2**	21.0	18.2
i-vector	20.7	17.8	21.5	18.7
i-vector subtract mean	**20.4**	**17.2**	**21.0**	**18.3**

Table 4. WER results with and without mean subtraction, for the VoxCeleb i-vector and x-vector embeddings without LDA.

5.4. ASR model configurations

With all of our models, we follow the same ESPnet recipes as Delcroix et al. for their sequence summary network (SeqSum) approach [5], except we do not use convolutional layers in the encoder for speaker-aware models as explained in section 4.1. We also train standard character level RNNLMs similar to Delcroix et al. [5], on the datasets' respective text resources, although note that Delcroix et al. do not present LM results on TED-LIUM. Details are omitted here for brevity. We achieve very similar baseline results, and therefore we present some of their results in comparison with ours.

In all models, the encoder consists of six 320-unit BLSTM layers, which subsample the input in time by a factor of four. The decoder has one 300-unit LSTM layer, and uses location-based attention, followed by a softmax layer, which outputs a distribution over characters (32 in TED-LIUM, 50 in WSJ). The models are trained for 15 epochs with the adadelta optimizer, with a batchsize of 30. In decoding use beam search with a beamsize of 20 for TED-LIUM and 30 for WSJ.

The encoders are trained with the multitask CTC-loss of ESPNet, and this is incorporated in decoding [23]. We also train some models on the WSJ task without the CTC-loss. Without the CTC-loss we retune the language model weight for the baseline model on the Dev93 set and use that same weight in all other non-CTC-loss experiments.

	TED-LIUM	Test		Dev	
		No LM	+LM	No LM	+LM
Fixed	Baseline	21.7	18.6	22.6	20.0
	SeqSum [5]	21.1	-	21.7	-
	i-vector$_{100}$	**20.9**	**17.9**	**21.4**	**18.9**
	x-vector$_{256}$	21.5	18.4	23.0	20.0
+VoxCeleb	i-vector$_{200\text{-LDA}}$	20.2	17.4	**20.7**	18.2
	i-vector$_{400}$	20.4	**17.2**	21.0	18.3
	x-vector$_{200\text{-LDA}}$	20.9	17.4	21.6	18.6
	x-vector$_{512}$	**20.1**	**17.2**	20.9	**18.1**
	thin-ResNet$_{512}$	20.7	**17.2**	21.0	18.3

	WSJ	Eval92		Dev93	
		No LM	+LM	No LM	+LM
Fixed	Baseline	17.5	9.3	22.1	13.2
	SeqSum [5]	16.3	8.7	21.3	13.2
	i-vector$_{100}$	17.6	**8.5**	22.3	**11.3**
	x-vector$_{256}$	**16.2**	8.6	**20.3**	11.6
+VoxCeleb	i-vector$_{200\text{-LDA}}$	17.2	9.1	21.2	11.9
	i-vector$_{400}$	**15.3**	**8.0**	20.5	11.7
	x-vector$_{200\text{-LDA}}$	18.8	9.5	25.0	13.5
	x-vector$_{512}$	16.2	8.7	20.5	**11.2**
	thin-ResNet$_{512}$	16.7	8.7	**20.4**	11.6

Table 5. WER results of the main speech recognition experiments. SeqSum refers to the sequence summary network approach of Delcroix et al. The embedding methods denote dimension, and whether LDA was used, in subscript. The +VoxCeleb sections present the results with the pretrained VoxCeleb embeddings. The Fixed sections present results with embeddings trained on the fixed ASR data.

Our input features are mean and variance normalized 80-dimensional Mel-filterbank energies, and pitch information, which might not contribute in English, but we retain it for conformity. We extract one speaker embedding for the whole utterance, and append it to each feature vector.

5.5. ASR Results

Table 5 shows the main results of our experiments. The models without the CTC multitask loss are not directly comparable, so their results are presented separately, in Table 6.

On the WSJ dataset, when not using an LM we get a better baseline without the CTC-loss than with it. This is likely due to the original recipe being tuned for the performance with a language model.

6. DISCUSSION

We achieve around 7% relative WER improvements with the VoxCeleb speaker embeddings. The VoxCeleb embeddings consistently perform better than the fixed ASR data embeddings, which obtain around around 4% relative improvements. The fixed data embeddings still consistently outperform the end-to-end sequence summary method. We see speaker-aware training as a useful competitive evaluation baseline when developing true end-to-end methods, such as the sequence summary network.

	WSJ	Eval92		Dev93	
		No LM	+LM	No LM	+LM
	Baseline	14.9	10.7	18.7	**13.7**
+VoxCeleb	i-vector$_{200\text{-LDA}}$	16.0	12.9	19.8	15.4
	i-vector$_{400}$	13.2	10.9	17.5	14.5
	x-vector$_{200\text{-LDA}}$	16.0	12.4	20.1	15.5
	x-vector$_{512}$	13.5	**10.4**	**16.9**	15.0
	thin-ResNet$_{512}$	**12.9**	10.6	17.2	14.1

Table 6. Results without the CTC task, i.e. a purely attentional model. Again, the embedding methods denote dimension, and whether LDA was used, in subscript. All of the speaker-aware methods used the Voxceleb embeddings.

No single embedding type consistently outperforms others. However, when embeddings are trained on the larger VoxCeleb dataset, the neural embeddings often outperform i-vectors. We suspect the neural embeddings are better able to leverage very large training sets. The *thin-ResNet* model is, without any modification, a good candidate for end-to-end finetuning in future work. For the x-vector approach, it seems an L2-normalization layer is needed.

Our hyperparameter tuning procedure for the fixed ASR data embeddings was quite ad-hoc. The ARI metric is probably closer to the speaker verification metric than ASR. However, the VoxCeleb embeddings are also originally tuned for speaker verification. Good, sound criteria, which could be used to separately optimize speaker embeddings for speaker-aware ASR training, are an open research question.

Of the embedding post-processing steps, we see that L2-normalization is crucial. We suspect the additional sensitivity of the normalization to unit length might not be universal, but rather particular to our implementation. Mean subtraction seems to work for i-vectors, but not for x-vectors. In most experiments, LDA did not help, with the exception of the VoxCeleb i-vector embeddings on TED-LIUM. However, we do not investigate different LDA dimension sizes.

7. CONCLUSIONS

We have shown that speaker-aware training is a competitive speaker adaptation approach in attention-based end-to-end ASR. We propose speaker-aware training as a viable strategy to incorporate untranscribed, speaker annotated data. When trained on large speaker annotated data, we find that neural embeddings can outperform i-vectors in speaker-aware ASR.

8. ACKNOWLEDGEMENTS

This work was supported by the European Unions Horizon 2020 research andinnovation programme via the project MeMAD (GA780069). We acknowledge the computational resources provided by the Aalto Science-IT project.

9. REFERENCES

[1] G. Saon, H. Soltau, D. Nahamoo, and M. Picheny, "Speaker adaptation of neural network acoustic models using i-vectors," in *2013 IEEE Workshop on Automatic Speech Recognition and Understanding*, Dec 2013, pp. 55–59.

[2] Xiaodong Cui, Vaibhava Goel, and George Saon, "Embedding-based speaker adaptive training of deep neural networks," in *Proc. Interspeech 2017*, 2017, pp. 122–126.

[3] W. Chan, N. Jaitly, Q. Le, and O. Vinyals, "Listen, attend and spell: A neural network for large vocabulary conversational speech recognition," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, March 2016, pp. 4960–4964.

[4] D. Bahdanau, J. Chorowski, D. Serdyuk, P. Brakel, and Y. Bengio, "End-to-end attention-based large vocabulary speech recognition," in *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, March 2016, pp. 4945–4949.

[5] Marc Delcroix, Shinji Watanabe, Atsunori Ogawa, Shigeki Karita, and Tomohiro Nakatani, "Auxiliary feature based adaptation of end-to-end asr systems," in *Proc. Interspeech 2018*, 2018, pp. 2444–2448.

[6] Zhong Meng, Yashesh Gaur, Jinyu Li, and Yifan Gong, "Speaker Adaptation for Attention-Based End-to-End Speech Recognition," in *Proc. Interspeech 2019*, 2019, pp. 241–245.

[7] Najim Dehak, Patrick J Kenny, Réda Dehak, Pierre Dumouchel, and Pierre Ouellet, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, 2011.

[8] David Snyder, Daniel Garcia-Romero, Gregory Sell, Daniel Povey, and Sanjeev Khudanpur, "X-vectors: Robust dnn embeddings for speaker recognition," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5329–5333.

[9] Weidi Xie, Arsha Nagrani, Joon Son Chung, and Andrew Zisserman, "Utterance-level aggregation for speaker recognition in the wild," in *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2019, pp. 5791–5795.

[10] Anthony Rousseau, Paul Deléglise, and Yannick Esteve, "Enhancing the ted-lium corpus with selected data for language modeling and more ted talks.," in *LREC*, 2014, pp. 3935–3939.

[11] Douglas B Paul and Janet M Baker, "The design for the wall street journal-based csr corpus," in *Proceedings of the workshop on Speech and Natural Language*. Association for Computational Linguistics, 1992, pp. 357–362.

[12] Anjuli Kannan, Yonghui Wu, Patrick Nguyen, Tara N Sainath, ZhiJeng Chen, and Rohit Prabhavalkar, "An analysis of incorporating an external language model into a sequence-to-sequence model," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 1–5828.

[13] A. Nagrani, J. S. Chung, and A. Zisserman, "Voxceleb: a large-scale speaker identification dataset," in *INTERSPEECH*, 2017.

[14] J. S. Chung, A. Nagrani, and A. Zisserman, "Voxceleb2: Deep speaker recognition," in *INTERSPEECH*, 2018.

[15] Joanna Rownicka, Peter Bell, and Steve Renals, "Embeddings for dnn speaker adaptive training," in *2019 IEEE Workshop on Automatic Speech Recognition and Understanding*, 2019, Accepted for publication, preprint accessed online 15.10.2019: https://arxiv.org/pdf/1909.13537.pdf.

[16] Kartik Audhkhasi, Bhuvana Ramabhadran, George Saon, Michael Picheny, and David Nahamoo, "Direct acoustics-to-word models for english conversational speech recognition," in *Proc. Interspeech 2017*, 2017, pp. 959–963.

[17] Natalia Tomashenko and Yannick Estève, "Evaluation of feature-space speaker adaptation for end-to-end acoustic models," in *Proceedings of the Eleventh International Conference on Language Resources and Evaluation (LREC-2018)*, 2018.

[18] Hossein Hadian, Hossein Sameti, Daniel Povey, and Sanjeev Khudanpur, "End-to-end speech recognition using lattice-free mmi," in *Proc. Interspeech 2018*, 2018, pp. 12–16.

[19] Yajie Miao, Mohammad Gowayyed, and Florian Metze, "Eesen: End-to-end speech recognition using deep rnn models and wfst-based decoding," in *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*. IEEE, 2015, pp. 167–174.

[20] Desh Raj, David Snyder, Daniel Povey, and Sanjeev Khudanpur, "Probing the information encoded in x-vectors," in *2019 IEEE Workshop on Automatic Speech Recognition and Understanding*, 2019, Accepted for publication, preprint accessed online 15.10.2019: https://arxiv.org/pdf/1909.06351.pdf.

[21] John HL Hansen and Taufiq Hasan, "Speaker recognition by machines and humans: A tutorial review," *IEEE Signal processing magazine*, vol. 32, no. 6, pp. 74–99, 2015.

[22] O. Glembek, L. Burget, P. Matějka, M. Karafiát, and P. Kenny, "Simplification and optimization of i-vector extraction," in *2011 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, May 2011, pp. 4516–4519.

[23] Shinji Watanabe, Takaaki Hori, Shigeki Karita, Tomoki Hayashi, Jiro Nishitoba, Yuya Unno, Nelson Enrique Yalta Soplin, Jahn Heymann, Matthew Wiesner, Nanxin Chen, Adithya Renduchintala, and Tsubasa Ochiai, "Espnet: End-to-end speech processing toolkit," in *Proc. Interspeech 2018*, 2018, pp. 2207–2211.

[24] S. Kim, T. Hori, and S. Watanabe, "Joint ctc-attention based end-to-end speech recognition using multi-task learning," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, March 2017, pp. 4835–4839.

[25] David Snyder, "x-vector and i-vector pretrained models download page," Online, http://kaldi-asr.org/models/m7, accessed 8.7.2019.

[26] Weidi Xie, "thin-resnet vlad pretrained models," Online, https://github.com/WeidiXie/VGG-Speaker-Recognition, accessed 8.7.2019.

[27] Lawrence Hubert and Phipps Arabie, "Comparing partitions," *Journal of classification*, vol. 2, no. 1, pp. 193–218, 1985.

[28] Daniel Povey, Arnab Ghoshal, Gilles Boulianne, Lukas Burget, Ondrej Glembek, Nagendra Goel, Mirko Hannemann, Petr Motlicek, Yanmin Qian, Petr Schwarz, Jan Silovsky, Georg Stemmer, and Karel Vesely, "The kaldi speech recognition toolkit," in *IEEE 2011 Workshop on Automatic Speech Recognition and Understanding*. Dec. 2011, IEEE Signal Processing Society, IEEE Catalog No.: CFP11SRW-USB.

B.7 AALTO's paper in ASRU 2019 workshop [6]

This paper describes AALTO experiments in speaker diarisation. It has been published in the IEEE ASRU 2019 workshop.

SPHEREDIAR: AN EFFECTIVE SPEAKER DIARIZATION SYSTEM FOR MEETING DATA

Tuomas Kaseva[1]*, Aku Rouhe*[1]*, Mikko Kurimo*[1]

Aalto University, Department of Signal Processing and Acoustics [1]

ABSTRACT

In this paper, we present *SphereDiar*, a speaker diarization system composed of three novel subsystems: the SphereSpeaker (SS) neural network, designed for speaker embedding extraction, a segmentation method called Homogeneity Based Segmentation (HBS) and a clustering algorithm called Top Two Silhouettes (Top2S). The system is evaluated on a set of over 200 manually transcribed multiparty meetings. The evaluation reveals that the system can be further simplified by omitting the use of HBS. Furthermore, we illustrate that SphereDiar achieves state-of-the-art results with two different meeting data sets.

Index Terms: speaker diarization, speaker embeddings, segmentation, spherical K-means, silhouette coefficients

1. INTRODUCTION

Speaker diarization answers the question "who spoke and when" [1]. In this process, a given audio stream is segmented into speaker turns: time intervals in which one speaker is speaking. It is determined, which of the speaker turns have the same speaker, but the actual identity (e.g. name) of the speakers is not required. Speaker diarization is a necessary subtask in many different speech applications such as creation of speech corpora, speech translation and speech recognition [1, 2, 3].

Speaker diarization is made difficult by the immense variability in speakers and recording conditions, and the unpredictable and overlapping speaker turns of spontaneous discussion [1, 4]. For these reasons, speaker diarization is still far from solved. In this paper, our main contribution is to propose a novel speaker diarization system which we have made available online[1]. The system consists of three main components which operate on three main tasks in speaker diarization: speaker modeling, segmentation and clustering [1].

The objective of speaker modeling is to embed a given speech utterance in a space which is more suitable for speaker discrimination [5]. Traditionally, this transformation has been performed with either Gaussian Mixture model (GMM) or i-vectors [6, 7]. Recently, also deep learning methods, both metric learning based [2, 8, 9, 10] and classification based [11, 12, 13], have been investigated. These methods have focused on creating neural speaker embeddings which have been shown to outperform i-vectors on many occasions [2, 13, 14]. Furthermore, especially classification based methods have shown great promise also in face verification [15, 16]. Motivated by these works, we choose to apply deep learning in our speaker modeling approach. We develop a novel neural network which learns the speaker embeddings through speaker classification. In this process, the network forces the embeddings to be L^2 normalized, or in other words, spherical. In our experiments, we show that this relatively simple operation has a profound positive impact on the speaker diarization task. Consequently, we name the network SphereSpeaker (SS), and our system SphereDiar.

In speaker diarization, segmentation refers to the task in which audio stream is divided into partitions which can be assigned to a single dominant speaker [1]. This procedure consists of speaker change detection (SCD) and overlapping speech detection (OSD) [1]. Whereas hypothesis testing has been the standard approach in the former [1, 9], Hidden Markov models accompanied with GMMs have been used in the latter [1, 17]. However, just as in speaker modeling, deep learning has recently been very successful in both OSD and SCD [9, 18, 19, 20]. Nevertheless, a segmentation approach which combines both OSD and SCD into a single process has not been proposed, although the connection of OSD and SCD has been well documented in literature [17]. In this paper, we develop such an approach, which we call Homogeneity Based Segmentation (HBS), and investigate its importance for our speaker diarization system. HBS uses deep learning and transforms the segmentation into a binary classification task.

The most popular clustering approach in speaker diarization has been agglomerative hierarchical clustering (AHC) [1, 4, 14, 21]. In addition, approaches exploiting Integer Linear Programming (ILP) [22], Information Bottleneck (IB) [23] and supervised learning [24] have been proposed. In our approach, we choose a slightly different clustering method which is based on using spherical K-means algorithm. This algorithm is essentially the same as K-means but uses cosine similarity as a distance distance metric and has L^2 normalized cluster centers [25]. The choice of the algorithm is based on our preliminary experiments for clustering the speaker embeddings created with SS. However, the algorithm requires the number of cluster centers as an input, which is typically

[1] https://github.com/Livefull/SphereDiar

unknown. Hence, in our method, we create multiple spherical K-means clusterings with a different number of clusters and choose the best clustering based on an empirically found and unsupervised criteria. These criteria are based on using silhouette coefficients [26] which, along with spherical K-means were also found to be beneficial for the clustering process. We call this method Top Two Silhouettes.

We show that our system achieves state-of-the-art results with a challenging dataset consisting of meeting recordings. Furthermore, we illustrate that these results are obtained even without using HBS and that HBS has overall a little significance for our system. As a consequence, our system can then be simplified considerably by excluding segmentation entirely. This is not only convenient but also an interesting discovery since especially OSD has been a prominent research direction in speaker diarization [1, 4, 17, 20].

2. DATA

The meeting corpus is composed of AMI (Augmented Multi-party Interaction) and ICSI (International Computer Science Institute) corpus, both of which consist of audio recordings of different meetings in various sites [27, 28]. Both corpora provide the recordings in multiple different audio formats from which the 16 kHz *Headset Mix* is used in all of our experiments. In order to create speaker diarization labels for a given meeting, we combine both manually generated and automatic speech recognition (ASR) based transcriptions. Unfortunately, complete ASR based transcriptions were not available for all meetings in AMI and ICSI corpus. The meetings which did not include ASR transcriptions were then excluded. These meetings can be found from Table 1. As a result, the number of remaining meetings is 237 consisting of 163 AMI and 74 ICSI meetings.

Each meeting in the meeting corpus is transformed into a sequence of overlapping frames $S = \{\mathbf{s_1}, ..., \mathbf{s_N}\}$, where frames $\mathbf{s_i}$ have a duration of 2s and are extracted every 0.5s. Before this framing operation, all non-speech segments are removed according to the reference transcriptions.

The choices of frame and overlap duration are based on several factors. Firstly, it is necessary that a frame is long enough so that proper modeling of the speaker corresponding to the frame is possible. Secondly, the frame has to be a short enough so that spontaneous speaker changes would not go unnoticed. As a result, a duration of 2 seconds was chosen, which has also been used in [9, 14].

Relatively large overlapping in turn is beneficial for the clustering procedure as it enables more samples for forming the clusters. However, an increase in overlap duration also results in a increase in computing time as the number of frames in S increases. Preliminary experiments illustrated that an overlap duration of 1.5 seconds would then be a suitable compromise.

Since speaker turns change unpredictably in spontaneous discussion, each two-second frame can include speech from multiple different speakers. That is, in general, each speaker speaks for only some percentage of the frame's duration. For each frame $\mathbf{s}$, we compute a quantity we call *homogeneity percentage* $H_{\%}$. It is the highest percentage of frame time covered by a single speaker. The frame's *speaker label* l is this most prominent speaker. Equivalently,

$$l = \arg\max_i |T_i|, \quad H_{\%} = \frac{\max_i |T_{i\neq -1}|}{|T|} * 100\%, \qquad (1)$$

where $T = \{T_{-1}, T_1, ..., T_{n_s}\}$ is a set of transcription labels of $\mathbf{s}$ with T_{-1} corresponding to samples which include overlapping speech and $T_{i\neq -1}$ depicting samples which are assigned to a speaker i.

Table 1. Removed meetings.

AMI	EN2001a, EN2001e, EN2002c, EN2003a, EN2006a EN2006b, IB4005, IS1003b
ICSI	Bmr012

The speaker corpora comprise of four different partitions, LS_{1000} and LS_{2000} which are collected from Librispeech corpus and VC_{1000} and VC_{2000} which are extracted from the Voxceleb2 dataset [3, 29]. The number of speakers in a partition is given as the subscript. To the best of our knowledge, the speakers in each partition are disjoint from the speakers in the meeting corpus. The speech material of each partition consists of frames of 2s duration which are extracted without overlap. The sampling frequency is the same as with the meeting corpus. In the extraction procedure, we use the webRTC speech activity detection (SAD) system [30], since reference transcriptions are not available. The gender distributions and the frame compositions are depicted in Table 2 and 3.

In order to balance the speaker label distributions with the partitions with the same number of speakers, the maximum number of frames per speaker is limited. The limit for the partitions LS_{2000} and VC_{2000} is assigned as 670 whereas the limit for LS_{1000} and VC_{1000} is 1000. The LS_{1000} partition, however, did not include quite as much speech material as VC_{1000}, so the maximum number of frames per speaker is only 764.

3. SPHEREDIAR

The block diagram of SphereDiar speaker diarization system is presented in Figure 1. In this Figure, input S depicts a sequence of $2s$ frames sampled with 16 kHz frequency and the output L the corresponding speaker label sequence. Note that we do not provide SAD in this system. This is by no

Table 2. Gender distribution in speaker corpora.

	Number of females	Number of males	Total
LS_{1000}	500	500	1000
LS_{2000}	987	1013	2000
VC_{1000}	500	500	1000
VC_{2000}	731	1269	2000

Table 3. Frame compositions in speaker corpora.

	Minimum number of frames per speaker	Maximum number of frames per speaker	Total number of frames
LS_{1000}	382	764	654 297
LS_{2000}	341	670	1 204 967
VC_{1000}	838	1000	995 443
VC_{2000}	577	670	1 337 601

means a trivial exclusion since SAD is an essential component in any speaker diarization system [1]. However, when diarization systems are developed, reference SAD labels are often used in order to focus on the actual speaker diarization [17, 20, 21, 31]. This is also the case with the speaker diarization systems against which we compare our system in section 4.

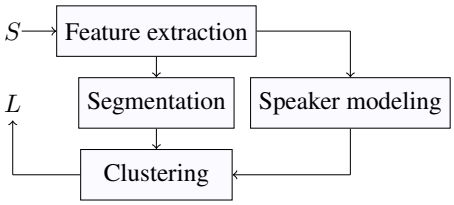


Fig. 1. Block diagram of SphereDiar.

Feature extraction. In the beginning of the diarization procedure, each frame $\mathbf{s}$ in S is converted to $\mathbf{x} \in \mathbb{R}^{201\times59}$, which consists of a sequence of 19 Mel-Frequency Cepstral Coefficients (MFCC), their first and second derivatives, and the first and second derivatives of energy just as in [32]. MFCCs are extracted every 10ms with a 25ms window duration using Librosa [33] and normalized with zero mean and unit variance.

SphereSpeaker. In speaker modeling, each feature sequence $\mathbf{x}$ is projected into a speaker embedding $\mathbf{f}(\mathbf{x})$. The projection is attained by using the neural network depicted in Figure 2 and Table 4. This network is initially designed to predict a class, or in our setting, a speaker identity for $\mathbf{x}$. Consequently, the final layer has a softmax activation function which assures that the output is an N_s dimensional probability distribution, where N_s is the number of classes. The speaker embedding $\mathbf{f}$ is produced in this classification process as the output of the last hidden layer. As a result, the final layer is only used during the training.

The network consists of two main components: a cascade of three bidirectional Long Short-Term Memory (LSTM) neural networks with skip connections which adheres to the architecture of [32] and an embedding layer. In this layer, we assign two conditions on the embedding: $\mathbf{f} \in \mathbb{R}^{1000}$ and $\|\mathbf{f}\|_2 = 1$. The use of L^2 normalization is influenced by the work in [12, 16] whereas the embedding dimension and the overall configuration of the embedding layer are based on our preliminary experiments. The importance of the normalization operation inside the network will be emphasized further in the experiments section where we compare SS with SS*. The latter is otherwise the same network as SS, but does not include L^2 normalization layer. Instead, the speaker embeddings extracted with this network are L^2 normalized externally.

Table 4. Output dimensions of each layer in SphereSpeaker and HBS neural networks.

SphereSpeaker neural network	Output dimensions
Bidirectional LSTM_1	201×500
Bidirectional LSTM_2	201×500
Bidirectional LSTM_3	201×500
Concatenation	201×1500
Embedding layer	1000
Fully connected layer (softmax)	N_s
HBS neural network	**Output dimensions**
Bidirectional LSTM	201×600
Attention layer	201×600
Average pooling layer	600
Fully connected layer (sigmoid)	1

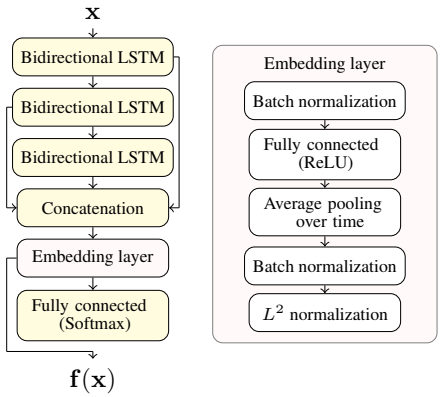


Fig. 2. SphereSpeaker neural network.

Homogeneity Based Segmentation. Segmentation is performed as a binary classification where the formulation of classes is based on the concept of homogeneity percentage. In this approach, our aim is to label each $\mathbf{x}$ as 0, if $H_{\%}$ of the

corresponding frame $\mathbf{s}$ of $\mathbf{x}$ exceeds a given threshold $H_{\theta\%}$ and otherwise as 1. As a result, we call this method Homogeneity Based Segmentation. Ideally, due to the definition of the homogeneity percentage, class 1 consists of frames which include speaker change boundaries and overlapping speech whereas class 0 comprises of frames which can be assigned to a single dominant speaker. Nevertheless, a gray area between classes does exist when homogeneity percentages are close to the threshold $H_{\theta\%}$. Moreover, there is no optimal threshold: we set $H_{\theta\%} = 65\%$ in our experiments as we consider it to be a suitable compromise. The ultimate goal of HBS is to exclude the frames assigned to class 1 from the clustering procedure. The main feature of the HBS is that in theory, it allows performing both OSD and SCD in a single process. Such simplification is yet to be proposed or experimented in speaker diarization.

The class labels are predicted with the neural network illustrated in Figure 3 and in Table 4. The two main components of this network are the bidirectional LSTM neural network which is motivated by the works in [9, 18, 19] and the attention layer which is based on the implementation in [34]. With the former, we use both regular and recurrent dropout and assign both dropouts as 0.2. All other layers are chosen based on our preliminary experiments. The class $h(\mathbf{x}) \in \{0, 1\}$ of each $\mathbf{x}$ is determined based on rounding the output of the network $\hat{h}(\mathbf{x}) \in [0, 1]$ to the nearest integer.

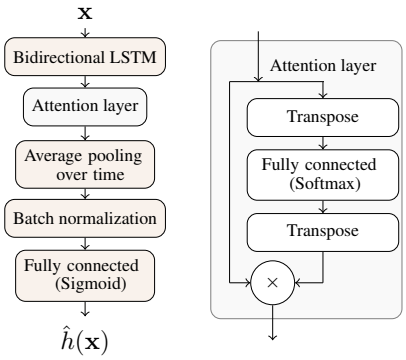


Fig. 3. HBS neural network.

Top Two Silhouettes. After the speaker modeling and segmentation we have obtained a sequence of speaker embeddings F and a sequence of HBS labels H. As a final step, we assign each $\mathbf{f}$ in F with a speaker label. In our approach, this assignment is determined by clustering E, a subset of F consisting of embeddings $\mathbf{f_i}$ with the HBS label $h_i = 0$. The clustering is performed with a novel algorithm which can be divided into two steps: the proposal generation and the optimal proposal determination.

In the first step, E is fitted with multiple different spherical K-means configurations with K ranging from 2 to N_{max}. Here, N_{max} refers to an initial guess of a maximum number of speakers in E. Each configuration is run with R different initializations from which the final configuration is determined based on the run which yielded the highest silhouette score. This score is the average of silhouette coefficients which are computed for each speaker embedding. In this computation, cosine similarity is used as a distance metric. More details of the calculation of the coefficients can be found in [26]. The proposals P_i are then created based on these final configurations.

In the second step, the optimal proposal P_{opt} is chosen. First, the proposals corresponding to the two largest silhouette scores, P_{top-1} and P_{top-2} are recovered. If (i) P_{top-1} has more clusters, or (ii) the silhouette score of P_{top-2} is below a threshold δ, then $P_{opt} = P_{top-1}$. This is a heuristic rule which we have found experimentally and can be interpreted as a further confidence that P_{top-1} is the optimal proposal.

Otherwise, if both (i) and (ii) are unsatisfied, the algorithm deduces that P_{top-2} could also be chosen. As P_{top-2} has then more clusters than P_{top-1}, the algorithm investigates if any of the clusters in P_{top-1} might contain inner clusters. This investigation is performed in a similar fashion as in the first step but for each each cluster in P_{top-1}. The assignment $P_{opt} = P_{top-2}$ is then chosen if for any initialization or cluster, both maximum silhouette value is above δ and a corresponding $K \in \{2, 3\}$. In this condition, the maximum number of inner clusters is restricted to 3 since a higher number would be highly improbable. However, if this condition is not satisfied the algorithm again chooses $P_{opt} = P_{top-1}$.

Algorithm 1: Top Two Silhouettes
Input: Set of speaker embeddings E, a number of initializations R, a maximum number of speakers N_{max} and a threshold δ
Output: Proposal $P = \{L, C\}$.
Steps:

1. Initialize $K = \{2, ..., N_{max}\}$ and $s = \{0, ..., 0\}, |s| = |K|$
2. **for** $r = 1$ to R **do**
   for $i = 1$ to $|K|$ **do**
   $\phi(K_i, E) \rightarrow L_i \rightarrow \upsilon(L_i, E) \rightarrow \hat{s}_i$
   if $(\hat{s}_i > s_i) \rightarrow s_i = \hat{s}_i$.
3. Find largest and second largest silhouette scores s_{top-1} and s_{top-2}, respectively.
   If not top-2 > top-1 $\wedge s_{top-2} > \delta$)
   $\rightarrow$ **return** L_{top-1}, C_{top-1}
4. Repeat step 2 for each $E_j \in E = \{\mathbf{f_i} \mid l_i = k \in L_{top-1}\}$ In the process, for any j, r:
   If $(\max_i \upsilon(L_{ij}, E_j) > \delta \wedge K_i \in \{2, 3\})$
   $\rightarrow$ **return** L_{top-2}, C_{top-2}
5. **return** L_{top-1}, C_{top-1}.

The labels L for F are then generated using associated cluster centers C_{opt} of P_{opt}. As the proposals corresponding to the two largest silhouette scores are central to the algorithm, we have named it Top Two Silhouettes. In the experiments section, we demonstrate the validity of this algorithm by comparing it with Top Silhouette (TopS), which is essentially the same as Top2S but always assigns $P_{opt} = P_{top-1}$.

Top Two Silhouettes is described more formally in Algorithm 1. In this description, spherical K-means is denoted with ϕ and the calculation of the silhouette score with a variable v. Moreover, instead of two steps, the description consists of three main steps consisting of the calculation of the silhouette scores, the evaluation of the conditions (i) and (ii) and the possible inner cluster search.

4. EXPERIMENTS

4.1. Experimental setup

Evaluation metric. All experiments are conducted using the same evaluation metric called diarization error rate (DER) [1]. In general, DER consists of SAD related errors (false alarm and false rejection) and speaker errors which, in our case, can be interpreted as a clustering errors between reference and predicted speaker labels [1]. However, since we have performed SAD on all meetings in the meeting corpus as a preprocessing step, the computation of DER simplifies to a calculation of the speaker error which we compute with Hungarian algorithm [35]. Furthermore, when calculating DER, we consider only labels corresponding to the frames which have $H_{\%}$ above the threshold $H_{\theta\%} = 65\%$ unless explicitly mentioned otherwise.

Neural network training and evaluation. We train 10 models in total: 8 for speaker modeling and 2 to be used for segmentation. The first eight models are trained using SS and SS* and four different training and evaluation set splits. The splits are generated from each partition in the speaker corpora by choosing randomly 45 frames from each speaker for testing and leaving the rest for training.

The last two models both use HBS neural network, but are trained solely using the meeting corpus with two different evaluation sets: AMI_{eval} which is a same as in [21] or $ICSI_{eval}$ consisting of 9 ICSI meetings [1]. In both cases, all other meetings in the meeting corpus are reserved for training. Moreover, only frames which have $H_{\%} = 100\%$ (labeled as 0) and frames with $H_{\%} \leq 65\%$ (labeled as 1) are used in training and evaluation. This choice is based on ensuring proper discrimination between classes that we found beneficial in our preliminary experiments.

All 10 models are trained using Keras deep learning library [36]with batch size 256, for 45 epochs. We use the cross entropy as a loss function and using Adam [37] optimizer. When training the last two models, we also weight class 1 twice as much as class 0 in order to balance the class distributions.

Clustering parameters. We assign $R = 50$ and $N_{max} = 11$ in all experiments. We choose to set R this high since spherical K-means has a tendency to converge to a local maximum [25]. The value of N_{max} is selected to exceed the highest possible participant number, 9, of the meetings in the meeting corpus. In addition, we set $\delta = 0.1$, which we attained by conducting a grid search on a clustering development set $Clust_{\text{dev}}$ of 12 meetings extracted from the meeting corpus[1]. This set is disjoint with both AMI_{eval} and $ICSI_{eval}$. In the grid search, we evaluated each threshold using DER, did not use HBS and performed speaker modeling with SphereSpeaker trained with VC_{1000}.

4.2. Results

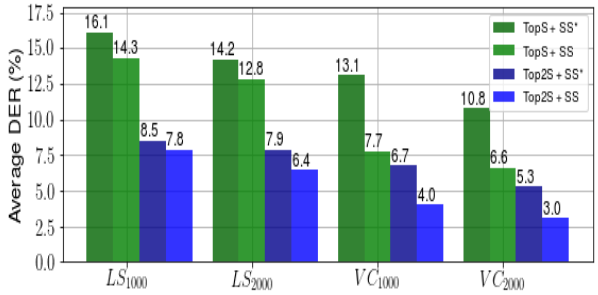


Fig. 4. Average DER over 225 meetings from the meeting corpus with different SphereDiar configurations which omit HBS.

In Figure 4, we visualize speaker diarization results with 225 meetings from the meeting corpus that are disjoint with $Clust_{\text{dev}}$. These results are obtained using all possible SphereDiar configurations introduced in this paper but without using HBS ($h_i = 0, \forall i$) as most of the meetings have been used in HBS training. The results illustrate that SS outperforms SS*, especially when these neural networks are trained with Voxceleb2 partitions, and that Top2S performs markedly better than TopS. Moreover, the results show that both the increase in the number of training speakers and the use of Voxceleb2 partitions over Librispeech partitions are preferable in speaker modeling training. The best configuration is attained by combining SS trained with VC_{2000} and Top2S and it achieves 3% average DER over the 225 meetings.

The results in Table 5 show that our HBS system fails to benefit the speaker diarization task. In the experiments which were briefly discussed with neural network training and evaluation, the HBS system achieved mean average precision of 0.953 with AMI_{eval} and 0.935 with $ICSI_{eval}$. Clearly, these scores were not high enough to make HBS beneficial for speaker diarization.

Table 5. Average DER (%) over different evaluation sets and HBS setups with the best SphereDiar configuration.

Segmentation	AMI_{eval}	$ICSI_{eval}$	225 meetings
-	2.4	2.9	3.0
HBS	3.5	4.8	-
Optimal HBS	2.0	2.5	2.8

However, the results also depict that even when using optimal HBS, which assigns h_i based on reference HBS labels, no significant improvement for the task is attained. This remark is especially distinct when the evaluation set consists of all of the 225 meetings. Interestingly, these results imply that our system is neither too dependent on OSD or SCD which have been previously shown to be important factors in speaker diarization [1, 4]. We hypothesize that this outcome is due to two reasons: a good generalization ability of the speaker embeddings and relatively low significance of HBS for the Top2S algorithm. Especially the latter can be emphasized, since HBS labels are only utilized to exclude the embeddings from the clustering procedure but not in any other manner. For example, the labels could have also been used in the initialization of spherical K-means. Nevertheless, based on the results in Table 5 we can deduce that SphereDiar achieves good results even without OSD or SCD.

Table 6. Average DER (%) comparison.

Test set	Previous best	Ours ($H_{\theta\%} = 55\%$)
AMI_{eval}	4.8 [21]	**3.6**
ICSI subset	13.1 [38]	**4.5**

In Table 6, a comparison between the best SphereDiar configuration and two other speaker diarization systems which have obtained top scores on AMI and ICSI subsets in the literature is provided. These systems include a state-of-the-art i-vector based speaker diarization system [21] and the ICSI RT07s speaker diarization system, which uses both MFCCs and deep learning based features [38, 39]. The average DER for both systems has been calculated from the segments which do not include overlapping speech and by using a forgiveness collar around speaker change boundaries [21, 38]. With [38], this collar is ± 0.25 seconds, whereas [21] uses the collar of ± 0.5 seconds. As was mentioned, the DER scores for both systems have been attained using reference SAD labels.

The computation of DER for SphereDiar is based on using the frames which have homogeneity percentages above the threshold $H_{\theta\%} = 55\%$. Due to the formulation of the percentage, this means that virtually all overlapping speech is removed from the DER calculation. Furthermore, decreasing the $H_{\theta\%}$ from 65%, which was used previously, to 55%, can be interpreted as shrinking the collar around speaker change boundaries. This decrease allows the average DER comparison to be as fair as possible since any further decrease in the value of $H_{\theta\%}$ results in severe difficulties of labeling the frames accurately. Consider, for instance, the example given in subsection 2.1.5. If a frame would have $H_{\%} = 50\%$, and would contain two speakers without any overlapping speech. Then, the speaker label of this frame could not be determined.

The results illustrate that our system is able to outperform the systems in [21, 38]. Our result is particularly better when comparing to [38] but we admit that our system has been trained with Voxceleb2 which was not available at the time for [38]. However, the system in [21] has been trained with a very similar data as ours, using Voxceleb [2] and other relevant datasets, but our result is still better. Moreover, as we do not use HBS in the comparison, our domain adaptation is only based on 12 meetings in $Clust_{dev}$. This is significantly less than used in either [21] or [38] and further emphasizes the generality of our system.

5. CONCLUSIONS

This paper proposed a novel speaker diarization system SphereDiar. The system includes two neural networks and one clustering algoritm: SphereSpeaker neural network for speaker embedding extraction, HBS neural network for segmentation and Top Two Silhouettes for clustering. In our experiments, we focused on evaluating the system with 225 meetings and illustrated that the system could be simplified by excluding HBS. Using the best system configuration, we achieved average DER over the meetings as 3%. We compared our system with two state-of-the-art speaker diarization systems and showed that the results obtained with our system were better.

Nevertheless, the system still suffers from deficiencies. Firstly, the dimension of the speaker embeddings is relatively large which slows clustering. Secondly, Top2S does not yet have any proper theoretical foundation. Furthermore, this algorithm is also not very suitable for situations were only few frames for each speaker can be attained. Finally, we have not presented any methods for SAD. In future work, we would like to address each of these shortcomings.

6. ACKNOWLEDGEMENTS

We would like to thank Anja Virkkunen and Stig-Arne Grönroos for their helpful comments. This work was supported by the European Unions Horizon 2020 research and innovation programme via the project MeMAD (GA780069). Computational resources were provided by the Aalto Science-IT project.

7. REFERENCES

[1] X. Anguera, S. Bozonnet, N. Evans, C. Fredouille, G. Friedland, and O. Vinyals, "Speaker diarization: A review of recent research," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 20, no. 2, pp. 356–370, 2012.

[2] A. Nagrani, J. S. Chung, and A. Zisserman, "Voxceleb: a large-scale speaker identification dataset," in *INTERSPEECH*, 2017.

[3] J. S. Chung, A. Nagrani, and A. Zisserman, "Voxceleb2: Deep speaker recognition," in *INTERSPEECH*, 2018.

[4] G. Sell, D. Snyder, A. McCree, D. Garcia-Romero, J. Villalba, M. Maciejewski, V. Manohar, N. Dehak, D. Povey, S. Watanabe *et al.*, "Diarization is hard: Some experiences and lessons learned for the JHU team in the inaugural DIHARD challenge," in *Proc. INTERSPEECH*, 2018, pp. 2808–2812.

[5] J. H. Hansen and T. Hasan, "Speaker recognition by machines and humans: A tutorial review," *IEEE Signal processing magazine*, vol. 32, no. 6, pp. 74–99, 2015.

[6] W. M. Campbell, D. E. Sturim, and D. A. Reynolds, "Support vector machines using GMM supervectors for speaker verification," *IEEE Signal Process. Lett.*, vol. 13, no. 5, pp. 308–311, 2006.

[7] N. Dehak, P. J. Kenny, R. Dehak, P. Dumouchel, and P. Ouellet, "Front-end factor analysis for speaker verification," *IEEE Transactions on Audio, Speech, and Language Processing*, vol. 19, no. 4, pp. 788–798, 2011.

[8] C. Li, X. Ma, B. Jiang, X. Li, X. Zhang, X. Liu, Y. Cao, A. Kannan, and Z. Zhu, "Deep speaker: an end-to-end neural speaker embedding system," *arXiv preprint arXiv:1705.02304*, 2017.

[9] H. Bredin, "Tristounet: triplet loss for speaker turn embedding," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 5430–5434.

[10] D. Snyder, D. Garcia-Romero, D. Povey, and S. Khudanpur, "Deep neural network embeddings for text-independent speaker verification," in *Proc. Interspeech*, 2017, pp. 999–1003.

[11] E. Variani, X. Lei, E. McDermott, I. Lopez-Moreno, and J. Gonzalez-Dominguez, "Deep neural networks for small footprint text-dependent speaker verification." in *ICASSP*, vol. 14. Citeseer, 2014, pp. 4052–4056.

[12] M. Hajibabaei and D. Dai, "Unified hypersphere embedding for speaker recognition," *arXiv preprint arXiv:1807.08312*, 2018.

[13] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust DNN embeddings for speaker recognition," *Submitted to ICASSP*, 2018.

[14] D. Garcia-Romero, D. Snyder, G. Sell, D. Povey, and A. McCree, "Speaker diarization using deep neural network embeddings," in *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2017, pp. 4930–4934.

[15] W. Liu, Y. Wen, Z. Yu, M. Li, B. Raj, and L. Song, "Sphereface: Deep hypersphere embedding for face recognition," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 1, 2017, p. 1.

[16] F. Wang, J. Cheng, W. Liu, and H. Liu, "Additive margin softmax for face verification," *IEEE Signal Processing Letters*, vol. 25, no. 7, pp. 926–930, 2018.

[17] S. H. Yella and H. Bourlard, "Overlapping speech detection using long-term conversational features for speaker diarization in meeting room conversations," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 22, no. 12, pp. 1688–1700, 2014.

[18] R. Yin, H. Bredin, and C. Barras, "Speaker change detection in broadcast tv using bidirectional long short-term memory networks," in *Interspeech 2017*. ISCA, 2017.

[19] G. Hagerer, V. Pandit, F. Eyben, and B. Schuller, "Enhancing LSTM RNN-based speech overlap detection by artificially mixed data," in *Audio Engineering Society Conference: 2017 AES International Conference on Semantic Audio*. Audio Engineering Society, 2017.

[20] J. T. Geiger, F. Eyben, B. Schuller, and G. Rigoll, "Detecting overlapping speech with long short-term memory recurrent neural networks," in *Proceedings INTERSPEECH 2013, 14th Annual Conference of the International Speech Communication Association, Lyon, France*, 2013.

[21] M. Maciejewski, D. Snyder, V. Manohar, N. Dehak, and S. Khudanpur, "Characterizing performance of speaker diarization systems on far-field speech using standard methods," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 2018, pp. 5244–5248.

[22] M. Rouvier and S. Meignier, "A global optimization framework for speaker diarization," in *Odyssey 2012*, 2012.

[23] D. Vijayasenan, F. Valente, and H. Bourlard, "Agglomerative information bottleneck for speaker diarization of meetings data," in *2007 IEEE Workshop on Automatic Speech Recognition & Understanding (ASRU)*. IEEE, 2007, pp. 250–255.

[24] A. Zhang, Q. Wang, Z. Zhu, J. Paisley, and C. Wang, "Fully supervised speaker diarization," *CoRR*, vol. abs/1810.04719, 2018.

[25] S. Zhong, "Efficient online spherical K-means clustering," in *Neural Networks, 2005. IJCNN'05. Proceedings. 2005 IEEE International Joint Conference on*, vol. 5. IEEE, 2005, pp. 3180–3185.

[26] P. J. Rousseeuw, "Silhouettes: a graphical aid to the interpretation and validation of cluster analysis," *Journal of computational and applied mathematics*, vol. 20, pp. 53–65, 1987.

[27] I. McCowan, J. Carletta, W. Kraaij, S. Ashby, S. Bourban, M. Flynn, M. Guillemot, T. Hain, J. Kadlec, V. Karaiskos *et al.*, "The AMI meeting corpus," in *Proceedings of the 5th International Conference on Methods and Techniques in Behavioral Research*, vol. 88, 2005, p. 100.

[28] A. Janin, D. Baron, J. Edwards, D. Ellis, D. Gelbart, N. Morgan, B. Peskin, T. Pfau, E. Shriberg, A. Stolcke *et al.*, "The ICSI meeting corpus," in *Acoustics, Speech, and Signal Processing, 2003. Proceedings.(ICASSP'03). 2003 IEEE International Conference on*, vol. 1. IEEE, 2003, pp. I–I.

[29] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: an ASR corpus based on public domain audio books," in *Acoustics, Speech and Signal Processing (ICASSP), 2015 IEEE International Conference on*. IEEE, 2015, pp. 5206–5210.

[30] A. Johnston, J. Yoakum, and K. Singh, "Taking on WebRTC in an enterprise," *IEEE Communications Magazine*, vol. 51, no. 4, pp. 48–54, 2013.

[31] S. H. Yella, A. Stolcke, and M. Slaney, "Artificial neural network features for speaker diarization," in *Spoken Language Technology Workshop (SLT), 2014 IEEE*. IEEE, 2014, pp. 402–406.

[32] G. Wisniewksi, H. Bredin, G. Gelly, and C. Barras, "Combining speaker turn embedding and incremental structure prediction for low-latency speaker diarization," in *Proc. Interspeech*, 2017.

[33] B. McFee, C. Raffel, D. Liang, D. P. Ellis, M. McVicar, E. Battenberg, and O. Nieto, "librosa: Audio and music signal analysis in python," in *Proceedings of the 14th python in science conference*, 2015, pp. 18–25.

[34] P. Rémy, "Keras Attention Mechanism," https://github.com/philipperemy/keras-attention-mechanism, 2017.

[35] O. Galibert, "Methodologies for the evaluation of speaker diarization and automatic speech recognition in the presence of overlapping speech." in *INTERSPEECH*, 2013, pp. 1131–1134.

[36] F. Chollet *et al.*, "Keras," https://keras.io, 2015.

[37] D. Kingma and J. Ba, "Adam: A method for stochastic optimization," *International Conference on Learning Representations*, 12, 2014.

[38] S. H. Yella and A. Stolcke, "A comparison of neural network feature transforms for speaker diarization," in *Sixteenth Annual Conference of the International Speech Communication Association*, 2015.

[39] C. Wooters and M. Huijbregts, "The ICSI RT07s speaker diarization system," in *Multimodal Technologies for Perception of Humans*. Springer, 2007, pp. 509–519.

B.8 EURECOM's paper in AI4TV 2019 workshop [7]

This paper describes EURECOM experiments in video captioning using spatio-temporal attention. It has been published in the AI4TV 2019 workshop.

L-STAP: Learned Spatio-Temporal Adaptive Pooling for Video Captioning

Danny Francis
danny.francis@eurecom.fr
EURECOM
Biot, France

Benoit Huet
benoit.huet@eurecom.fr
EURECOM
Biot, France

ABSTRACT

Automatic video captioning can be used to enrich TV programs with textual informations on scenes. These informations can be useful for visually impaired people, but can also be used to enhance indexing and research of TV records. Video captioning can be seen as being more challenging than image captioning. In both cases, we have to tackle a challenging task where a visual object has to be analyzed, and translated into a textual description in natural language. However, analyzing videos requires not only to parse still images, but also to draw correspondences through time. Recent works in video captioning have intended to deal with these issues by separating spatial and temporal analysis of videos. In this paper, we propose a Learned Spatio-Temporal Adaptive Pooling (L-STAP) method that combines spatial and temporal analysis. More specifically, we first process a video frame-by-frame through a Convolutional Neural Network. Then, instead of applying an average pooling operation to reduce dimensionality, we apply our L-STAP, which attends to specific regions in a given frame based on what appeared in previous frames. Experiments on MSVD and MSR-VTT datasets show that our method outperforms state-of-the-art methods on the video captioning task in terms of several evaluation metrics.

CCS CONCEPTS

• **Information systems → Content analysis and feature selection**; • **Computing methodologies → Natural language generation**; • **Applied computing → Annotation**.

KEYWORDS

deep learning, neural networks, video captioning

ACM Reference Format:
Danny Francis and Benoit Huet. 2019. L-STAP: Learned Spatio-Temporal Adaptive Pooling for Video Captioning. In *1st International Workshop on AI for Smart TV Content Production, Access and Delivery (AI4TV '19), October 21, 2019, Nice, France.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3347449.3357484

AI4TV '19, October 21, 2019, Nice, France

ACM ISBN 978-1-4503-6917-6/19/10…$15.00
https://doi.org/10.1145/3347449.3357484

1 INTRODUCTION

Automatic video captioning can be used to enrich TV programs with textual informations on scenes. These informations can be useful for visually impaired people, but can also be used to enhance indexing and research of TV records. The video captioning task consists in automatically generating short textual descriptions for videos. It is a challenging multimedia task as it requires to grasp all information contained in a video, such as objects, persons, context, actions, location, and to translate this information into text. This task can be compared to a translation task: except instead of translating a sequence of words in a source language into a sequence of words in a target language, the aim is to translate a sequence of frames into a sequence of words. Therefore, most of recent works in video captioning rely on the encoder-decoder framework proposed in [25], initially for text translation. In video captioning, the encoder aims at deriving a video representation. Recent advances in deep learning have shown to fit very well to that task. In particular, Convolutional Neural Networks (CNNs) have proved to give excellent results in producing highly descriptive image representations or video representations. The decoder part aims at generating a sentence based on the representation produced by the encoder. Long Short-Term Units (LSTMs) [12] and Gated Recurrent Units (GRUs) [5] are usually chosen for that task. Image captioning [30] and video captioning can seem to be similar tasks, as both of them require to "translate" a visual object into a textual one. However, video captioning poses a problem that makes it more challenging than image captioning: it requires to take into account temporality.

In [18], authors showed that for text translation tasks based on the encoder-decoder framework, results could be improved if the decoder attended to hidden states of the encoder based on its hidden states. Some other works showed that the same attention mechanisms could be applied to video captioning [10, 31, 38, 39]. The improvement induced by that attention mechanism can be interpreted as follows: when the decoder is predicting the next word of a sentence, it attends to relevant frames to perform that task accurately. Some other works have also shown that attending to relevant regions in a video during the encoding phase could lead to better representations of videos, and thus better results [32, 40]. However these works attend to local regions based on frame-level considerations, without taking into account previous frames. In our work, we aim at attending to relevant regions of a video based on previous frames, because the relevance of objects, persons or actions relies on the context in which they appear; and that context should be inferred from previous frames. More precisely, after deriving frame-level local features using the last convolutional layer of a ResNet-152 [11], we do not apply an average pooling to pool these local features. We process them by Learned

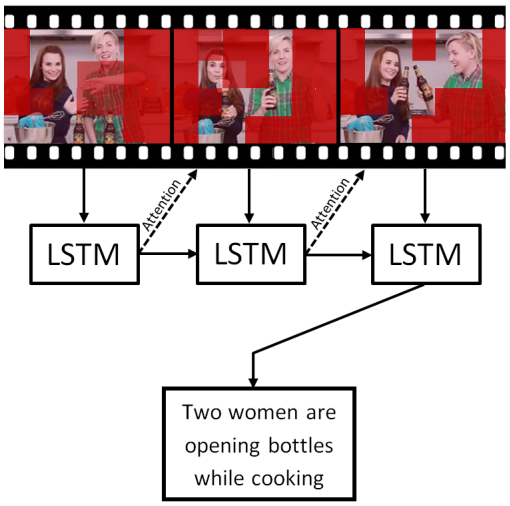


Figure 1: Overview of our L-STAP method. Frame-level local features are derived using a ResNet-152. Then, an LSTM processes these local features, and updates its hidden state by attending to them based on previous frames. The result is that space and time are jointly taken into account to build video representations.

Spatio-Temporal Adaptive Pooling (L-STAP). L-STAP attends to specific regions of a frame based on what occurred previously in the video. Our pooling method is learned because it is based on an LSTM whose parameters are learned. It is spatio-temporal because it takes into account space and time in a joint fashion. In addition, it is adaptive because the attention paid to local regions is based on previous hidden states of the LSTM; pooling depends not only on the processed frame but also on previous ones. A high-level schematic view of our proposed model is depicted in Figure 1.

We evaluated our results on two common datasets used for benchmarking video captioning tasks: MSVD [4] and MSR-VTT [36]. Results show that our model based on L-STAP outperforms state-of-the-art models in terms of several metrics. An ablation study also shows that our method leads to significant improvements with respect to state-of-the-art methods.

Our contributions can be summarized as follows: we propose a novel pooling method for video processing, which we evaluate on the video captioning task, even though it could be applied to any other task involving video processing, such as video classification. Moreover, we demonstrate the interest of our pooling method over usual approaches. The paper is organized as follows. In Section 2, we introduce previous works on video captioning. In Section 3, we present our model based on L-STAP. Section 4 is dedicated to experiments. We conclude the paper in Section 5.

2 RELATED WORK

Video captioning can be seen as a translation task: a sequence of frames, which can be compared to a sequence of words in a source language, have to be translated in a target language. Some pioneering works such as [23] make use of Statistical Machine Translation techniques to generate captions from videos. However nowadays, most of recent works on video captioning rely on Deep Learning techniques, and more particularly on the encoder-decoder framework that has been developed in [25] for text translation [8]. Moreover, attending to the hidden states of the encoder during the decoding phase has shown to give significant improvements in Neural Machine Translation [18], which have been confirmed in by [38] in the context of video captioning.

In some works, videos are split into frames, global features are derived for each frame using a CNN [11, 13, 24, 26], and the obtained features vectors are sequentially processed by the encoder [10, 14, 17, 19, 31, 34]. The drawback in such approaches is that spatial information is lost. In our approach, we aim at taking into account this spatial information.

Other approaches take into account locality. However, these approaches have some significant differences with our approach. In [38], the authors separate their model into two parts: a usual encoder-decoder based on global features of frames, and a 3D-CNN that derives a single representation for a whole video. The 3D-CNN they employ does take into account locality, but it has two major conceptual differences with respect to our method. First, it is based on handcrafted features, which do not provide as much semantic information as CNN features. Moreover, the pooling operations that are used to get their video representations are neither learned nor adaptive. In our approach, pooling takes into account the relevance of local features in a frame with respect to previous frames. In [39], authors use local features to trace semantic concepts along videos, which is conceptually different from our approach, as we aim to derive a video representation based on these local features. In [35], authors propose another method to compute trajectories through videos. In both papers, these trajectories are combined with global features to build video representations. In [32], local features are used to generate video representations. However, local features from different spatial locations are not related together, contrary to our work, which proposes to attend to local features based on all local features from previous frames. Eventually, some other works used 3D-CNN architectures [33] or convolutional RNNs [32] to relate local features through time. However, due to the nature of convolution operations, relations drawn through these methods remain local: they are not able to spatially relate objects from the video which are far from each other in a video for instance. Our method, as we will show in the following, is build to grasp jointly spatial and temporal information, by attending to relevant locations of a frame with respect to previous ones.

3 PROPOSED MODEL

Let us first formulate the problem we are to deal with. Given a video V, which is a sequence of T frames $(v^{(1)}, ..., v^{(T)})$, our goal is to derive a descriptive sentence $Y = (y_1, ..., y_L)$. The approach that we have followed is based on the encoder-decoder framework. The encoder first derives frame-level representations $(x^{(1)}, ..., x^{(T)}) = X$, and then pool these representations together to form frame-level video representations $(\overline{h}^{(1)}, ..., \overline{h}^{(T)})$. Based on these representations, the decoder reconstructs a descriptive sentence in a recurrent fashion. Figure 2 summarizes the important steps our model. In the

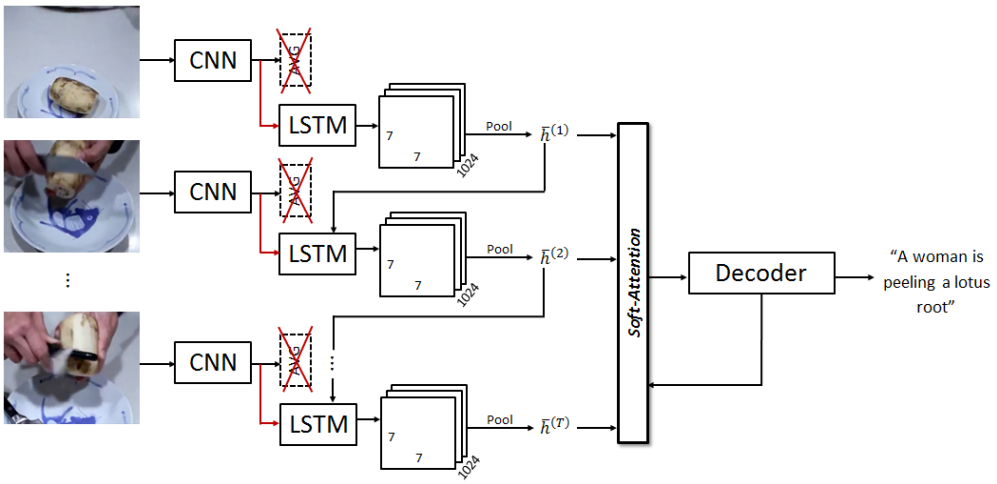


Figure 2: Illustration of our model, based on the proposed L-STAP method. Frames are processed sequentially by a CNN (a ResNet-152 in this case). However, instead of applying an average pooling on local features as some recent works do, we make use of an LSTM to capture time dependencies. Local hidden states are computed to obtain a 7x7x1024-dimensional tensor. These local hidden states are then pooled together (using average pooling or soft attention), and processed by an LSTM decoder to output a sentence.

following section, we will describe it in detail and report how we train it.

3.1 Grasping Spatio-Temporal Dependencies with L-STAP

As we stated above, the first step is to produce a representation of the input video. In the following subsections, we will explain how we derive frame-level features, and how we pool them together.

3.1.1 Image-Level Features. Given a video $V = (v^{(1)}, ..., v^{(T)})$, we need to derive features for each frame $v^{(t)}$. A common way to do so is to process each frame using a CNN, which has been previously pretrained on a large-scale dataset. In works such as [17], the outputs of the penultimate layer of a ResNet-152 have been chosen as frames representations, which consist of 2048-dimensional vectors. However, such representations discard locality, which results in loss of information. Therefore, in this work, we choose to take the output of the last convolutional layer of a ResNet-152. Thus, we obtain frame-level representations $(x^{(1)}, ..., x^{(T)}) = X$, where $x^{(t)} \in \mathbb{R}^{7\times7\times2048}$ for all t. The next step is to process these dense frame-level representations to derive compact frame-level representations, using the proposed L-STAP method instead of conventional pooling.

3.1.2 How L-STAP Works. L-STAP aims at replacing the average pooling operation after the last convolutional layer in a CNN, and to pool local features according to previous frames. The goal is to capture where important actions are occurring, and to discard locations that are not relevant to summarize what is happening in a video. For that purpose, we use an LSTM, taking local features as inputs, resulting in local hidden states, which are then combined in a way we will describe later in this subsection. More formally, given local features $x_{ij}^{(t)} \in \mathbb{R}^{2048}$, the aggregated local features $h_{ij}^{(t)}$ are computed recursively as follows:

$$i_{ij}^{(t)} = \sigma(W_{ix}x_{ij}^{(t)} + W_{ih}\overline{h}^{(t-1)} + b_i) \quad (1)$$

$$f_{ij}^{(t)} = \sigma(W_{fx}x_{ij}^{(t)} + W_{fh}\overline{h}^{(t-1)} + b_f) \quad (2)$$

$$o_{ij}^{(t)} = \sigma(W_{ox}x_{ij}^{(t)} + W_{oh}\overline{h}^{(t-1)} + b_o) \quad (3)$$

$$c_{ij}^{(t)} = f_{ij}^{(t)} \circ \overline{c}^{(t-1)} + i_{ij}^{(t)} \tanh(W_{cx}x_{ij}^{(t)} + W_{ch}\overline{h}^{(t-1)} + b_c) \quad (4)$$

$$h_{ij}^{(t)} = o_{ij}^{(t)} \circ \tanh(c_{ij}^{(t)}) \quad (5)$$

where W_{ix}, W_{ih}, b_i, W_{fx}, W_{fh}, b_f, W_{ox}, W_{oh}, b_o, W_{cx}, W_{ch} and b_c are trainable parameters, and $\overline{c}^{(t-1)}$ and $\overline{h}^{(t-1)}$ are respectively the memory cell and the hidden state of the LSTM. Please note that memory cells and hidden states are shared for computing all aggregated local features. The memory cell and the hidden state at time t are computed as follows:

$$\overline{c}^{(t)} = \sum_{i=1}^{7}\sum_{j=1}^{7} \alpha_{ij}^{(t)} c_{ij}^{(t)} \quad (6)$$

$$\overline{h}^{(t)} = \sum_{i=1}^{7}\sum_{j=1}^{7} \alpha_{ij}^{(t)} h_{ij}^{(t)} \quad (7)$$

where $\alpha_{ij}^{(t)}$ are local weights. In our work, we experimented with two types of local weights. We first tried to use uniform weights:

$$\alpha_{ij}^{(t)} = \frac{1}{7\times7} \quad (8)$$

which actually correspond to an average pooling of aggregated local features. The second solution that we tried was to derive local weights using an attention mechanism, as follows:

$$\tilde{\alpha}_{ij}^{(t)} = w^T \tanh(W_{\alpha x} x_{ij}^{(t)} + W_{\alpha h} \overline{h}^{(t-1)} + b_\alpha). \quad (9)$$

$$\alpha_{ij}^{(t)} = \frac{\exp(\tilde{\alpha}_{ij}^{(t)})}{\sum_{k=1}^{7}\sum_{l=1}^{7}\exp(\tilde{\alpha}_{kl}^{(t)})}, \quad (10)$$

where $W_{\alpha x}$, $W_{\alpha h}$, b_α are trainable parameters.

3.2 Encoding Videos

In our model, we encode videos using the L-STAP method we presented previously. We initialized the memory cell and the hidden state of the LSTM using the output of an I3D [3] (before the final softmax) which had been trained on Kinetics-600 [3]. More formally, if V is an input video:

$$c_{ij}^{(0)} = \tanh(W_c^{\mathrm{e}} \mathrm{e}(V) + b_c^{\mathrm{e}}) \quad (11)$$

$$h_{ij}^{(0)} = \tanh(W_h^{\mathrm{e}} \mathrm{e}(V) + b_h^{\mathrm{e}}) \quad (12)$$

where W_c^{e}, b_c^{e}, W_h^{e} and b_h^{e} are trainable parameters. The decoder produces $\overline{c}^{(T)}$ and $\overline{h}^{(T)}$ as outputs, where T is the length of the input video. These outputs will be used to initialize the sentence decoder that we will introduce in the next section.

3.3 Decoding Sentences

For decoding sentences, we chose to use an LSTM. In the following, we assume that sentences Y are represented by sequences of one-hot vectors $y_1, ..., y_L \in \mathbb{R}^N$ where N is the vocabulary size. The aim of the LSTM is to compute the probabilities $\mathrm{P}(y_l | y_{l-1}, ..., y_1, V; \theta)$ for $l \in \{1, ..., L\}$, where θ is the set of all parameters in the encoder and the decoder, and V an input video. In the following, we will describe formally how we compute these probabilities.

We initialize the memory cell and the hidden state of the decoder LSTM using the last memory cell and the last hidden state of the decoder:

$$c_0^d = \overline{c}^{(T)}, \quad (13)$$

$$h_0^d = \overline{h}^{(T)}. \quad (14)$$

It has been shown in [18] for text translation tasks that attending to hidden states of the encoder during the decoding phase improved results. Some works in video captioning have followed that approach successfully [37, 38]. We followed a similar approach for our decoding phase. More precisely, at each step l, we compute a weighted sum of hidden states of the encoder:

$$\varphi(\overline{h}, h_{l-1}^d) = \sum_{t=1}^{T} \beta_l^{(t)} \overline{h}^{(t)} \quad (15)$$

where $\beta_l^{(1)}, ..., \beta_l^{(T)}$ are computed as follows:

$$\tilde{\beta}_l^{(t)} = w_\beta^T \tanh(W_{\beta e} \overline{h}^{(t)} + W_{\beta h} h_{l-1}^d + b_\beta), \quad (16)$$

$$\beta_l^{(t)} = \frac{\exp(\tilde{\beta}_l^{(t)})}{\sum_{k=1}^{L}\exp(\tilde{\beta}_k^{(t)})}, \quad (17)$$

where $W_{\beta e}$, $W_{\beta h}$, b_β are trainable parameters. Assuming that the word y_{l-1} has been decoded at step $l-1$, we aim to decode y_l based on y_{l-1} and $\varphi(\overline{h}, h_{l-1}^d)$. For that purpose, we first compute a word embedding x_l^d:

$$w_l^d = W_{\mathrm{emb}} y_{l-1}, \quad (18)$$

where W_{emb} is a learned embedding matrix. Then, we concatenate w_l^d and $\varphi(\overline{h}, h_{l-1}^d)$ to obtain and x_l^d:

$$x_l^d = [w_l^d; \varphi(\overline{h}, h_{l-1}^d)] \quad (19)$$

Eventually, we input x_l^d to the decoder LSTM:

$$i_l^d = \sigma(W_{ix}^d x_l^d + W_{ih}^d h_{l-1}^d + b_i^d) \quad (20)$$

$$f_l^d = \sigma(W_{fx}^d x_l^d + W_{fh}^d h_{l-1}^d + b_f^d) \quad (21)$$

$$o_l^d = \sigma(W_{ox}^d x_l^d + W_{oh}^d h_{l-1}^d + b_o^d) \quad (22)$$

$$c_l^d = f_l^d \circ c_{l-1}^d + i_l^d \tanh(W_{cx}^d x_l^d + W_{ch}^d h_{l-1}^d + b_c^d) \quad (23)$$

$$h_l^d = o_l^d \circ \tanh(c_l^d) \quad (24)$$

where W_{ix}^d, W_{ih}^d, b_i^d, W_{fx}^d, W_{fh}^d, b_f^d, W_{ox}^d, W_{oh}^d, b_o^d, W_{cx}^d, W_{ch}^d and b_c^d are trainable parameters.

The last step is to infer a word y_l. For that purpose, we derive $\tilde{y}_l$ as follows:

$$\tilde{y}_l = \mathrm{softmax}(W_d h_l^d) \quad (25)$$

where W_d is a trainable parameter. We state that y_l is the one-hot vector corresponding to the maximum coordinate of $\tilde{y}_l$.

3.4 Training

Assuming that $y_1, ..., y_L$ correspond to ground-truth words, we aim to minimize the following cross-entropy loss:

$$\mathcal{L}_d(\theta) = -\sum_{l=1}^{L} \log \mathrm{P}(\tilde{y}_l | y_{l-1}, ..., y_1, V; \theta) \quad (26)$$

where V is a video corresponding to the caption $(y_1, ..., y_L)$.

In addition to that, some works have shown that regularizing the cross-entropy loss with a matching loss between video encodings and ground-truth sentences could improve results by bridging the semantic gap between them [10, 17]. As reported in Section 4.4, such improvement has been noticed in our experiments. The matching model we employed is described in the following. Let us assume that $Y = (y_1, ..., y_L)$ is a sentence corresponding to a video V. First, we translate this sequence of one-hot vectors into a sequence of word embeddings $(x_1^s, ..., x_L^s)$ using the matrix W_{emb} from Section 3.3. Then, we compute a sentence embedding $\psi(Y)$ by processing this sequence of word embeddings into another LSTM: each word embedding is entered sequentially as an input to that LSTM, and $\psi(Y)$ is defined to be its last hidden state. We want the initialization of the decoder to be as close as possible to an accurate representation of its corresponding sentence. Therefore, if

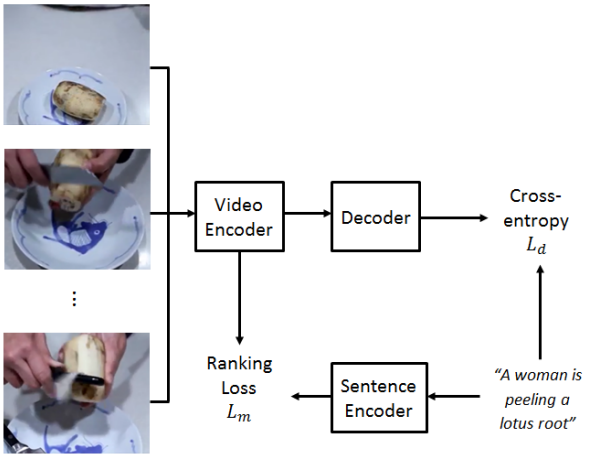


Figure 3: Overview of our training losses. The first training loss is the Cross-Entropy loss, which aims to make the probability distribution of sentences in the training set and the probability distribution of the inferred sentences match. The second one is a ranking loss, aiming to bridge the semantic gap between video representations and sentences.

$\varphi(V) = \overline{h}^{(T)}$ is the initial hidden state of the decoder, we will aim to minimize the following ranking loss from [9]:

$$\mathcal{L}_m(\theta) = \max_{\overline{V} \neq V} \Big(\max(0, \alpha - S(\varphi(V), \psi(Y)) + S(\varphi(\overline{V}), \psi(Y))) \Big) + \max_{\overline{Y} \neq Y} \Big(\max(0, \alpha - S(\varphi(V), \psi(Y)) + S(\varphi(V), \psi(\overline{Y}))) \Big) \quad (27)$$

where $\overline{V}$ is a negative video sample, and $\overline{Y}$ is a negative sentence sample coming from another video than V. The final loss is the following:

$$\mathcal{L}(\theta) = \mathcal{L}_d(\theta) + \lambda \mathcal{L}_m(\theta) \quad (28)$$

where λ is a hyperparameter that we set to 0.4 according to results on validation.

4 EXPERIMENTS

4.1 Datasets

We evaluated our models on two video captioning datasets: MSVD [4] and MSR-VTT [36]. MSVD is a dataset composed of 1,970 videos from YouTube, which have been annotated by Amazon Mechanical Turks (AMT). Each video has approximately 40 captions in English. We split that dataset following [29]: 1,200 videos for training, 100 videos for validation and 670 videos for testing. MSR-VTT is a similar dataset, but with much more videos, and less captions per video. It is composed of 10,000 videos, and 20 captions per video. Following [36], we split that dataset into 6,513 videos for training, 497 videos for validation and 2,990 videos for testing.

For both datasets, we uniformly sampled 30 frames per video as done in [40], and extracted features for each frame based on the last convolutional layer of a ResNet-152 [11], which had been trained on the image-text matching task on MSCOCO [16], after pre-training on ImageNet-1000 [6] following [9]. In addition, we extracted activity features for each video using an I3D pretrained on Kinetics-600 [3]. For MSVD, we converted sentences to lowercase and removed special characters, which lead to a vocabulary of about 14k words. We converted each word into an integer, and cut sentences after the thirtieth word if their lengths were higher than thirty. The same approach for MSR-VTT lead to a much bigger vocabulary size of about 29k words. Therefore, we kept only the 15k most common words, and replaced all the others by an <UNK> token. We applied the same process otherwise.

4.2 Implementation Details

Our models have been implemented with the TensorFlow framework [1]. We use 1024-dimensional LSTMs in both encoder and decoder. Soft attention spaces are 256-dimensional. Word embeddings are 300-dimensional.

We trained our model using the RMSProp algorithm [27], with decay = 0.9, momentum = 0.0 and epsilon = 1e-10. Batch size is set to 64. Learning rate is 1e-4, and we apply gradient clipping to a threshold of 5. Eventually, we apply dropout on the output of the decoder (before the prediction layer) with a rate of 0.5 to avoid overfitting.

4.3 Results on MSVD and MSR-VTT

We evaluated our models in terms of BLEU [20], ROUGE [15], METEOR [7] and CIDEr [28] scores, which are metrics commonly used to evaluate automated captioning tasks. We compared them to the following recent models for video captioning. Our results on MSVD are presented in Table 1. Results on MSR-VTT are presented in Table 2.

On MSVD, it can be noticed that L-STAP achieves the best results on six out of seven metrics. It is also relevant to mention that E2E [14], which achieves better CIDEr results than our model, has been trained using reinforcement learning techniques to be optimized regarding that CIDEr metric. Works on image captioning and video captioning have shown that significant improvements could be done using such techniques [2, 22, 34], at the price of much longer training times. We did not use reinforcement learning to train our models, instead we use cross-entropy minimization which has the advantage of being fast and simpler to implement.

Results on MSR-VTT show that our model outperforms models trained using a cross-entropy loss on two metrics out of four (METEOR and ROUGE). HRL [34] obtains better results overall, however it makes use of reinforcement learning techniques, which leads to better results as stated in the previous paragraph.

We report some qualitative results of our model on MSR-VTT in Figure 4. On the second video, the man who is singing appears during a very limited amount of time. This shows that our model has been able to attend to important frames to identify what the main action of the video was. In the first video, a woman starts talking about makeup, and then puts some lipstick on her lips. The caption generated by our model shows that it has been able to draw a relation between the first and the second parts of the video. Moreover, the lipstick is applied on a very localized part of the video frames: we can infer that our model could efficiently attend to the right part of the frame to generate a caption. The fourth video

GT: A woman is applying lipstick and explaining about it
Inferred: A woman is talking about makeup

GT: A man is singing and walking through the street
Inferred: A man is singing in the street

GT: A group of teens dancing together
Inferred: A group of people are dancing

GT: Cartoon characters are saying colors
Inferred: A cartoon character is shown

GT: Someone is making food
Inferred: A man in a blue shirt is making a dish

Figure 4: Some qualitative results of L-STAP on MSR-VTT.

Model	Bleu-1	Bleu-2	Bleu-3	Bleu-4	ROUGE	METEOR	CIDEr
TSL [35]	-	-	-	51.7	-	34.0	74.9
RecNet [31]	-	-	-	52.3	69.8	34.1	80.3
mGRU [21]	82.5	72.2	63.3	53.8	-	34.5	81.2
AGHA [40]	83.1	73.0	64.3	**55.1**	-	35.3	83.3
SAM [32]	-	-	-	54.0	-	35.3	87.4
E2E* [14]	-	-	-	50.3	70.8	34.1	87.5
SibNet [17]	-	-	-	54.2	71.7	34.8	**88.2**
L-STAP (Ours)	**84.0**	**74.1**	**64.5**	**55.1**	**72.7**	**35.4**	86.7

Table 1: Results on the MSVD dataset. The * sign means that the model is using reinforcement learning techniques to optimize over the CIDEr metric. Best results are in bold characters.

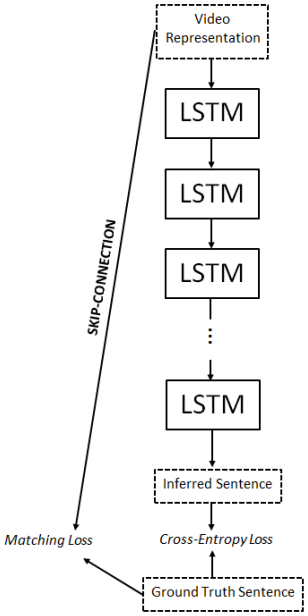


Figure 5: Our second interpretation about the efficiency of the second term of our loss function. Skip connections between video representations and ground-truth sentences improve results.

shows that results could be improved by adding sound processing to our model: it was not possible from the video only to know that colors were said.

4.4 Ablation Study

Results of an ablation study on the MSVD dataset are reported in Table 3. The encoder we used in our baseline model is an Long-term Recurrent Convolutional Network (LRCN) [8]. As shown in previous works such as [10, 17], adding a component to the training loss to make video representations match sentence representations improves results. Two interpretations can be given to these results. A first one is that adding a ranking loss to match video representations and sentence representations helps bridging the semantic gap between these two modalities. A second one could be that propagating the gradient across all the layers of the decoder could make it vanish through depth. Thus, adding a matching loss to the cross-entropy loss could be seen as a skip-connection between the sentence to be generated and the video representation used by the decoder. We illustrate that second interpretation in Figure 5.

Replacing the average pooling at the end of a CNN by our L-STAP induces a major improvement with respect to all metrics as reported in Table 3. On top of that, results shown in Table 1 demonstrate that L-STAP leads to better results than other models based on local features such as AGHA and SAM, and results shown in both Table 1 and Table 2 show the interest of L-STAP over average pooling.

We can notice in Table 3 that using a soft-attention mechanism to pool local hidden states in the encoder does not provide significant improvements over average pooling for all metrics except from CIDEr. Our interpretation is that the LSTM of the encoder can learn to attend to relevant local features by itself: before applying the average pooling, attention has already been drawn quite efficiently.

5 CONCLUSION

Video captioning is a way for TV broadcasters to enhance user experience, in particular regarding accessibility. In this paper, we presented a novel Learned Spatio-Temporal Adaptive Pooling (L-STAP) method for video captioning. It consists in taking into account spatial and temporal information jointly in a video to produce good video representations. As we have shown, these video representations can be successfully used to perform automated video captioning. We demonstrated the quality of our models based on L-STAP by comparing them with state-of-the-art models on MSVD and MSR-VTT, which are two video captioning datasets. On top of that, we assessed the interest of L-STAP through an ablation study. Although this paper concentrates on video captioning we believe that the proposed L-STAP method could be also applied to other video-related tasks such as video classification.

ACKNOWLEDGMENTS

This work was partially funded by ANR (the French National Research Agency) via the ANTRACT project and the European H2020 research and innovation programme via the project MeMAD (GA780069).

REFERENCES

[1] Martín Abadi, Paul Barham, Jianmin Chen, Zhifeng Chen, Andy Davis, Jeffrey Dean, Matthieu Devin, Sanjay Ghemawat, Geoffrey Irving, Michael Isard, et al.

Model	Bleu-4	ROUGE	METEOR	CIDEr
RecNet [31]	39.1	59.3	26.6	42.7
E2E* [14]	40.1	61.0	27.0	48.3
SibNet [17]	40.9	60.2	27.5	47.5
HRL* [34]	**41.3**	**61.7**	**28.7**	**48.0**
L-STAP (Ours)	40.7	61.2	27.6	44.8

Table 2: Results on the MSR-VTT dataset. The * sign means that the model is using reinforcement learning techniques to optimize over the CIDEr metric. Best results are in bold characters.

Model	Bleu-1	Bleu-2	Bleu-3	Bleu-4	ROUGE	METEOR	CIDEr
Baseline	83.2	72.5	63.1	52.7	71.4	34.1	79.5
Baseline + matching	83.4	72.8	63.3	53.3	71.2	34.5	82.2
L-STAP (avg) + matching	84.1	74.0	65.0	55.1	72.3	35.4	84.3
L-STAP (attention) + matching	84.0	74.1	64.5	55.1	72.7	35.4	86.8

Table 3: Results of ablation study on MSVD. Results show that a significant improvement can be reached using our Learned Spatio-Temporal Adaptive Pooling instead of the usual average pooling. Pooling hidden states of the encoder using soft-attention (line 4) instead of average pooling (line 3) does not always improve results. Our interpretation of that outcome is that the LSTM actually performs a kind of attention on local features before local hidden states are pooled together.

2016. Tensorflow: A system for large-scale machine learning. In *12th {USENIX} Symposium on Operating Systems Design and Implementation ({OSDI} 16)*. 265–283.

[2] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. 2018. Bottom-up and top-down attention for image captioning and visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 6077–6086.

[3] Joao Carreira and Andrew Zisserman. 2017. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 6299–6308.

[4] David L Chen and William B Dolan. 2011. Collecting highly parallel data for paraphrase evaluation. In *Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies-Volume 1*. Association for Computational Linguistics, 190–200.

[5] Kyunghyun Cho, Bart Van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder-decoder for statistical machine translation. *arXiv preprint arXiv:1406.1078* (2014).

[6] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. 2009. Imagenet: A large-scale hierarchical image database. In *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 248–255.

[7] Michael Denkowski and Alon Lavie. 2014. Meteor universal: Language specific translation evaluation for any target language. In *Proceedings of the ninth workshop on statistical machine translation*. 376–380.

[8] Jeffrey Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Kate Saenko, and Trevor Darrell. 2015. Long-term recurrent convolutional networks for visual recognition and description. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2625–2634.

[9] Fartash Faghri, David J Fleet, Jamie Ryan Kiros, and Sanja Fidler. 2017. Vse++: Improving visual-semantic embeddings with hard negatives. *arXiv preprint arXiv:1707.05612* (2017).

[10] Zhao Guo, Lianli Gao, Jingkuan Song, Xing Xu, Jie Shao, and Heng Tao Shen. 2016. Attention-based LSTM with semantic consistency for videos captioning. In *Proceedings of the 24th ACM international conference on Multimedia*. ACM, 357–361.

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.

[12] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neural computation* 9, 8 (1997), 1735–1780.

[13] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. 2012. Imagenet classification with deep convolutional neural networks. In *Advances in neural information processing systems*. 1097–1105.

[14] Lijun Li and Boqing Gong. 2019. End-to-end video captioning with multitask reinforcement learning. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*. IEEE, 339–348.

[15] Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. *Text Summarization Branches Out* (2004).

[16] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. 2014. Microsoft coco: Common objects in context. In *European conference on computer vision*. Springer, 740–755.

[17] Sheng Liu, Zhou Ren, and Junsong Yuan. 2018. SibNet: Sibling Convolutional Encoder for Video Captioning. In *2018 ACM Multimedia Conference on Multimedia Conference*. ACM, 1425–1434.

[18] Minh-Thang Luong, Hieu Pham, and Christopher D Manning. 2015. Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025* (2015).

[19] Yingwei Pan, Ting Yao, Houqiang Li, and Tao Mei. 2017. Video captioning with transferred semantic attributes. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 6504–6512.

[20] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting on association for computational linguistics*. Association for Computational Linguistics, 311–318.

[21] Ramakanth Pasunuru and Mohit Bansal. 2017. Multi-task video captioning with video and entailment generation. *arXiv preprint arXiv:1704.07489* (2017).

[22] Steven J Rennie, Etienne Marcheret, Youssef Mroueh, Jerret Ross, and Vaibhava Goel. 2017. Self-critical sequence training for image captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 7008–7024.

[23] Marcus Rohrbach, Wei Qiu, Ivan Titov, Stefan Thater, Manfred Pinkal, and Bernt Schiele. 2013. Translating video content to natural language descriptions. In *Proceedings of the IEEE International Conference on Computer Vision*. 433–440.

[24] Karen Simonyan and Andrew Zisserman. 2014. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556* (2014).

[25] Ilya Sutskever, Oriol Vinyals, and Quoc V Le. 2014. Sequence to sequence learning with neural networks. In *Advances in neural information processing systems*. 3104–3112.

[26] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. 2015. Going deeper with convolutions. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 1–9.

[27] T. Tieleman and G. Hinton. 2012. Lecture 6.5—RmsProp: Divide the gradient by a running average of its recent magnitude. COURSERA: Neural Networks for Machine Learning.

[28] Ramakrishna Vedantam, C Lawrence Zitnick, and Devi Parikh. 2015. Cider: Consensus-based image description evaluation. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 4566–4575.

[29] Subhashini Venugopalan, Marcus Rohrbach, Jeffrey Donahue, Raymond Mooney, Trevor Darrell, and Kate Saenko. 2015. Sequence to sequence-video to text. In *Proceedings of the IEEE international conference on computer vision*. 4534–4542.

[30] Oriol Vinyals, Alexander Toshev, Samy Bengio, and Dumitru Erhan. 2015. Show and tell: A neural image caption generator. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 3156–3164.

[31] Bairui Wang, Lin Ma, Wei Zhang, and Wei Liu. 2018. Reconstruction network for video captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 7622–7631.
[32] Huiyun Wang, Youjiang Xu, and Yahong Han. 2018. Spotting and aggregating salient regions for video captioning. In *2018 ACM Multimedia Conference on Multimedia Conference*. ACM, 1519–1526.
[33] Junbo Wang, Wei Wang, Yan Huang, Liang Wang, and Tieniu Tan. 2018. M3: multimodal memory modelling for video captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 7512–7520.
[34] Xin Wang, Wenhu Chen, Jiawei Wu, Yuan-Fang Wang, and William Yang Wang. 2018. Video captioning via hierarchical reinforcement learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 4213–4222.
[35] Xian Wu, Guanbin Li, Qingxing Cao, Qingge Ji, and Liang Lin. 2018. Interpretable Video Captioning via Trajectory Structured Localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 6829–6837.
[36] Jun Xu, Tao Mei, Ting Yao, and Yong Rui. 2016. Msr-vtt: A large video description dataset for bridging video and language. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 5288–5296.
[37] Ziwei Yang, Yahong Han, and Zheng Wang. 2017. Catching the temporal regions-of-interest for video captioning. In *Proceedings of the 25th ACM international conference on Multimedia*. ACM, 146–153.
[38] Li Yao, Atousa Torabi, Kyunghyun Cho, Nicolas Ballas, Christopher Pal, Hugo Larochelle, and Aaron Courville. 2015. Describing videos by exploiting temporal structure. In *Proceedings of the IEEE international conference on computer vision*. 4507–4515.
[39] Youngjae Yu, Hyungjin Ko, Jongwook Choi, and Gunhee Kim. 2017. End-to-end concept word detection for video captioning, retrieval, and question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 3165–3173.
[40] Junchao Zhang and Yuxin Peng. 2019. Hierarchical Vision-Language Alignment for Video Captioning. In *International Conference on Multimedia Modeling*. Springer, 42–54.

B.9 AALTO's paper in MULEA '19 workshop [8]

This paper describes the development of dense image captioning techniques at AALTO that may later have influence on the work in MeMAD. It has been published in the MULEA '19 workshop.

Geometry-aware Relational Exemplar Attention for Dense Captioning

Tzu-Jui Julius Wang
Department of Computer Science, Aalto University
Espoo, Finland
tzu-jui.wang@aalto.fi

Hamed R. Tavakoli
Nokia Technologies
Espoo, Finland
hamed.rezazadegan_tavakoli@nokia.com

Mats Sjöberg
CSC – IT Center for Science
Espoo, Finland
mats.sjoberg@csc.fi

Jorma Laaksonen
Department of Computer Science, Aalto University
Espoo, Finland
jorma.laaksonen@aalto.fi

ABSTRACT

Dense captioning (DC), which provides a comprehensive context understanding of images by describing all salient visual groundings in an image, facilitates multimodal understanding and learning. As an extension of image captioning, DC is developed to discover richer sets of visual contents and to generate captions of wider diversity and increased details. The state-of-the-art models of DC consist of three stages: (1) region proposals, (2) region classification, and (3) caption generation for each proposal. They are typically built upon the following ideas: (a) guiding the caption generation with image-level features as the context cues along with regional features and (b) refining locations of region proposals with caption information. In this work, we propose (a) a joint visual-textual criterion exploited by the region classifier that further improves both region detection and caption accuracy, and (b) a Geometry-aware Relational Exemplar attention (GREatt) mechanism to relate region proposals. The former helps the model learn a region classifier by effectively exploiting both visual groundings and caption descriptions. Rather than treating each region proposal in isolation, the latter relates regions in complementary relations, i.e. *contextually dependent*, *visually supported* and *geometry* relations, to enrich context information in regional representations. We conduct an extensive set of experiments and demonstrate that our proposed model improves the state-of-the-art by at least +5.3% in terms of the mean average precision on the Visual Genome dataset.

CCS CONCEPTS

• **Computing methodologies** → **Scene understanding**.

KEYWORDS

dense captioning, attention, relationship modeling

MULEA '19, October 25, 2019, Nice, France

ACM ISBN 978-1-4503-6918-3/19/10…$15.00
https://doi.org/10.1145/3347450.3357656

ACM Reference Format:
Tzu-Jui Julius Wang, Hamed R. Tavakoli, Mats Sjöberg, and Jorma Laaksonen. 2019. Geometry-aware Relational Exemplar Attention for Dense Captioning. In *1st International Workshop on Multimodal Understanding and Learning for Embodied Applications (MULEA '19), October 25, 2019, Nice, France.* ACM, New York, NY, USA, 9 pages. https://doi.org/10.1145/3347450.3357656

1 INTRODUCTION

Advancements in computer vision applications, such as object detection and segmentation, have laid a strong foundation of comprehensive context understanding in images. Besides learning on visual domain, tasks such as image captioning (IC) [3, 7, 24] and visual question answering (VQA) [1] are the iconic examples that connect vision and language modalities to not only provide better visual reasoning, but also enable multimodal context understanding. The IC task is to generate a human understandable sentence from a given image. Such a sentence should be grammatically correct, adequately expressive, and capture holistic view of the image content. The VQA task is to generate a sentence to answer a given question targeting at an image. While such a multimodal model (e.g. an IC model) is able to describe an image, it continues to express varying image contents with a sentence that can hardly capture multiple perspectives of the image content.

To extend the capability of a captioning model, Johnson et al. introduced the *Dense Captioning* (DC) task where the aim is to describe as many as possible regions of interest (RoIs) in an image [9]. More specifically, DC comprises two joint tasks: (a) localizing the RoIs (e.g. by bounding boxes) and (b) generating a sentence describing each grounded region. These tasks introduce two more challenges to image captioning: (1) detecting and proposing meaningful ROIs for captions and (2) understanding the relations between the region proposals. For example, in Figure 1, two visual groundings surrounding the man are closely related in visual contents and captions. Besides, the larger RoI surrounding the whole body of the man provides the most informative context for the smaller RoI captioned with "blue jeans of *man*". This indicates that the captioning process can benefit from a DC model that is capable of capturing relationships between regions.

We address the aforementioned challenges by (1) introducing a joint visual-textual criterion for detecting RoIs and (2) proposing a Geometry-aware Relational Exemplar attention (GREatt) module

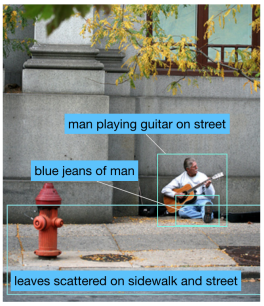


Figure 1: An instance in the Visual Genome dataset [11] that reveals how the visual and caption information from other regions can be particularly informative to some regions. Here, the informative context for region captioned with *blue jeans of man* is the region captioned with *man playing guitar on street*, which is cued by the street area captioned with *leaves scattered on sidewalk and street*.

for capturing relations between the RoIs. Utilizing the caption embedding along with the visual representations enforces the model to learn a better alignment between the visual content and the corresponding captions by projecting them into a shared subspace. Simultaneously, the optimality of the proposed RoIs is improved. GREatt accounts for three intrinsically distinct types of region-level relationships, including (a) *spatially correlated*, (b) *contextually dependent*, and (c) *visually similar and supported* relations. The spatially correlated relation considers regions that are correlated by their locations and sizes. The contextually dependent relation considers if a region provides contextual information for another region. The visually similar and supported relation focuses on visually similar contexts to enhance the evidence of the existence of a specific context.

To summarize, our contributions are (1) a new geometry-aware relational exemplar attention module and (2) a joint visual-textual region classification criteria. which together lead to a new state-of-the-art in the dense captioning task.

2 RELATED WORK

2.1 Image Captioning

Understanding image captioning is essential because it is the fundamental building block of any captioning pipeline. We, thus, briefly overview some of the most relevant works and refer the readers to [7] for further reading.

The classical image captioning methods such as [3] relied on linking a sentence to an image via feature mapping and were limited to retrieving a pre-existing sentence from a corpus of sentences. The techniques which utilize a language model, however, show more flexibility in generating a sentence from a feature vector representing the image. The most successful of such methods are neural-based techniques [10, 16, 21]. Many of the recent image captioning pipelines follow a similar path.

The most relevant works to us are, in particular, the attention-based image captioning methods. For example [24] defined a soft-attention mechanism (also known as top-down attention) that learns to align the visual features with textual features dynamically over time while generating a sentence. Pedersoli et al. [13] extended the same idea by employing geometrical transformations to the regions used for captioning. The top-down attention mechanism often loses its effectiveness after the visual features are fine-tuned for the captioning task [13]. In contrast to the top-down mechanism, R. Tavakoli et al. [18] investigated the bottom-up attention mechanism. While they demonstrated that bottom-up attention cannot help much improving the caption qualities, they showed such a mechanism enhances the robustness of captioning models. Recently, He et al. [6] proposed an effective approach for combining both bottom-up and top-down attention.

Our proposed approach follows a similar path to attention-based image captioning, specifically using top-down attention. Nevertheless, we focus on dense captioning and try to encode the relations between regions for building powerful context features.

2.2 Dense Captioning

Dense captioning was introduced along with the Visual Genome dataset [11], which aims to promote vision and language research in conjunctions with a range of perceptual reasoning and question answering tasks. The dataset provides 5.4 million region annotations with bounding boxes and captions for 108,077 images, averaging ~50 annotations per image.

The first dense captioning model was introduced by the pioneering work of Johnson et al. [9]. Their framework consists of three components: (1) an image feature extractor (e.g. implemented by a VGG net [17]), (2) a region detector, and (3) a caption generator. Given an image, it first projects the image into the feature space. Then, it detects a series of RoIs using the region proposal mechanism. Finally, each RoI is described with a sentence using the caption generator language model based on recurrent neural networks (RNN) [12] and image features corresponding to that RoI. They tested their model on Visual Genome version 1 [11] and established the first baseline for this task.

Yang et al. [26] extended the idea by replacing the localization layer with Faster-RCNN [14], using captions for improving the localization of region proposals generated by Faster-RCNN, and exploiting both regional and image-level features for the language model. They demonstrated that each of these modifications and their combinations significantly improve the dense captioning.

Nevertheless, image-level features as context can mislead the caption generator towards describing the global context rather than the region of interest [26]. In contrast, our proposed GREatt mechanism learns the context features from the proposed regions by considering distinct types of pairwise relationships between the RoIs. Hence, our pipeline uses features which are more contextually dependent yet region-specific and improve caption quality. In addition, to further capitalize on the idea of engaging captions in the proposal process, we propose a region classifier (which determines the likelihood of a proposal being a genuine RoI) learned on a subspace shared by textual features and their visual counterparts. Developing these two novel designs on top of the pipeline proposed in [26] further enhances the performance in both region classification and caption generation.

2.3 Attention and Relation Reasoning

Reasoning about the relation of two feature vectors which represent objects, entities, and elements with neural networks has gained a recent interest and has been a core module in wide range of applications, such as image captioning [27], object detection [8], and visual question answering (VQA) [15], and scene graph generation (SGG) [23, 25].

Many existing works have proposed different means to associate two feature vectors (e.g. $\mathbf{v}_i$ and $\mathbf{v}_j$) and capture their mutual importance as $\alpha_{i,j}$. Introducing the notion of importance, one can link relation reasoning to attention and interpret $\alpha_{i,j}$ as a quantity of how much one should also pay attention to $\mathbf{v}_j$ during inference about $\mathbf{v}_i$ given a task. The most notable work for our purpose in this annals is transformer networks [19] (originally for natural language processing (NLP) tasks) in which the attention weights are defined by the function of scaled dot-product (SDP) between $\mathbf{v}_i$ and $\mathbf{v}_j$, emphasizing similarity of representations.

In the context of object detection, Hu et al. [8] proposed a revised SDP attention, which additionally considers the geometry relationship between object proposals, allowing them to be refined and classified jointly rather than in isolation. Yao et al. [27] constructed a directed graph over the object proposals, in which each node of the graph is represented by the visual features of the proposals, in order to do image captioning. The refined object-level representation which embeds with the graph structure is then calculated through graph convolutional networks (GCN). Yang et al. [25] capitalized on a similar idea to relate the region proposals for scene graph generation.

Two other relevant ideas are graph attention networks [20] and Neural Turing Machine [4]. The first one was originally proposed for the graph classification task, and in it two features interact through concatenation followed by a multi-layer perceptron (MLP). The second one extends the same line of research with external memory modules and employs the cosine similarity function to capture the interaction between entities.

Even though many works have proposed different attention mechanisms for the downstream tasks, most of them learn the attention embodied by *single* relation (e.g. by SDP attention [8, 19]). What remains less studied is can *multiple* attentions formulated in different computational forms benefit each other for a given computer vision task. This work addresses 1) *do different attention mechanisms work better in isolation?* and 2) *are they complementary to each other?* By examining and exploiting the complementary relations captured by visual and geometry features, we propose a novel attention mechanism built upon distinct types of relations which improve the dense captioning task.

3 METHOD

In this section, we describe the problem formulation, our proposed architecture and each component in the pipeline. The code is publicly available at https://github.com/aalto-cbir/greatt_densecap.

3.1 Problem Formulation

We devise the dense captioning problem to consist of four sub-tasks: 1) region proposal (RP), 2) region classification (RC), 3) proposal refinement (PR), and 4) region caption generation (CG). Region proposal firstly generates a set of region proposals which are then classified by a region classifier. The locations of region proposals are refined gradually as the caption generation process proceeds. The objectives of each task are formulated as follows:

Region proposal (RP). Region proposal is to learn to generate a set of proposals $\hat{\mathcal{B}} = \{\hat{B}_i\}_{i=1}^{N_r}$ that well match to the ground-truth proposals $\mathcal{B} = \{B_i\}_{i=1}^{N}$, where N_r is the number of the generated proposals and N is the number of proposals in an image. Each proposal is characterized by a rigid box, defined by its center coordinate, width and height. Note that, here we use N and N_r for notational simplicity, though they may be different for each image.

Region classification (RC). Region classification decides whether a region proposal is good enough to be captioned or should be ignored. We classify the regions by additionally conditioning them on the captions $\hat{\mathcal{S}} = \{\hat{S}_i\}_{i=1}^{N_r}$ (which are generated by the model learned on the ground-truth captions $\mathcal{S} = \{S_i\}_{i=1}^{N}$) and the relationships between proposals. For an image $\mathcal{I}$ we build a directed graph $\mathcal{G} = (\mathcal{V}, \mathcal{E})$ over the representations of N_r proposed regions, denoted by $\mathcal{V} = \{\mathbf{v}_i, \mathbf{b}_i\}_{i=1}^{N_r}$, where $\mathbf{v}_i$ refers to the visual representation and $\mathbf{b}_i$ to the geometry representation, which are defined later in Sec. 3.3. The edges $\mathcal{E}$ correspond to the relationships. We, thus, minimize

$$E_{cls} = -\sum_i \log P(c_i|\hat{B}_i, \hat{S}_i, \mathcal{G}), \tag{1}$$

where E_{cls} is the energy function for region classification and c_i indicates the class label, i.e. captioned ($c_i = 1$) or non-captioned ($c_i = 0$) region.

Proposal refinement (PR). We further refine the proposed regions by leveraging the caption information, akin to [26]. That is, we minimize the following energy function:

$$E_{box} = \sum_{i \in \mathbf{pos}} E_i^{box}(\Delta\hat{B}_i|\hat{B}_i, \hat{S}_i), \tag{2}$$

where $\Delta\hat{B}_i$ is the offsets to the proposal $\hat{B}_i$ estimated in the region proposal task and **pos** denotes the set of positive proposals.

Region caption generation (CG). To generate a caption for each region, we consider the relation graph $\mathcal{G}$ to minimize

$$E_{cap} = \sum_{i \in \mathbf{pos}} E_i^{cap}(S_i|\mathcal{G}). \tag{3}$$

3.2 Overview of the Framework

Figure 2 depicts a high-level sketch of the proposed framework for dense captioning. The input image is first processed by a region proposal network (RPN) [14] to attain proposals from which the regional visual representations $\{\mathbf{v}_i\}_{i=1}^{N_r}$ are extracted. A graph $\mathcal{G}$, whose edge weights are calculated by GREatt, is constructed over $\mathbf{v}_i$ and employed to obtain a relational representation $\mathbf{g}_i$. Both $\mathbf{v}_i$ and $\mathbf{g}_i$ are then fed into the captioning module to generate a caption embedding. Finally, the caption embedding along with $\mathbf{v}_i$ and $\mathbf{g}_i$ are used to classify the region as captioned or non-captioned class. In the following subsections, we introduce the formation of $\mathbf{g}_i$ and describe the proposal refinement and caption nets in detail.

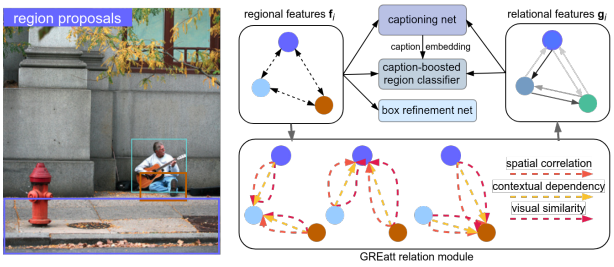


Figure 2: The proposed framework divides the dense captioning problem into four sub-tasks tackled by four sub-modules, i.e. a) region proposal network, b) region classifier, c) proposal refinement net, and d) captioning net. It features Geometry-aware Relational Exemplar attention mechanism (GREatt), a relation module which is constructed on different types of relationships among region proposals and learns region-specific features which account for the most relevant context in the image. In addition, the proposed region classifier is learned on relational features delivered by GREatt and additionally on the caption information.

3.3 Geometry-aware Relational Exemplar Attention

In this section, we discuss the construction of the graph structure between the proposed regions and demonstrate how one can learn a powerful representation by considering the latent relationships between the proposed regions. To this end, we propose the Geometry-aware Relational Exemplar Attention (GREatt) module.

Having the region proposals $\hat{B}$ and their representations $\mathcal{V}$ generated by the RPN, we aim to learn contextual representations which are constructed on different types of relationships, namely, visual relationships and geometry relationships. The visual relationships account for the contextual dependency and visual similarity. The geometry relationships explain the spatial correlation and arrangement between any two region proposals (i.e. bounding boxes).

Given the individual regional features $\mathbf{v}_i \in \mathbb{R}^{D_v}, i = 1, ..., N_r$, GREatt calculates the relational features $\mathbf{g}_i$ by

$$\mathbf{g}_i = \mathbf{v}_i + \sum_{j=1}^{N_r} \alpha_{i,j} \mathbf{v}_j, \ \forall i, \tag{4}$$

$$\alpha_{i,j} = f_\alpha(\alpha^g_{i,j}, \alpha^v_{i,j}, \alpha^\omega_{i,j}), \tag{5}$$

where $\alpha_{i,j}$ reflects how much $\mathbf{v}_j$ should be associated with $\mathbf{v}_i$ in region classification and caption generation. $f_\alpha(\cdot)$ is GREatt *contextual function* (details provided in Sec. 3.3.4) that learns to embed three different relationships into $\alpha_{i,j}$. These relationships are 1) contextually dependent relation $\alpha^g_{i,j}$, 2) visually similar relation $\alpha^v_{i,j}$, and 3) geometry relation $\alpha^\omega_{i,j}$. The first two relations are based on the visual representation and the third relation is based on the geometry representation. In the following paragraphs, we describe how $\alpha^g_{i,j}$, $\alpha^v_{i,j}$, and $\alpha^\omega_{i,j}$ can be addressed computationally and discuss the possible options to implement f_α.

3.3.1 Contextually Dependent Relations $\alpha^g_{i,j}$. Used in [8, 22] for the object detection task, and in [20] for aggregating representations in graphical structures for graph classification, concatenating one representation (e.g. $\mathbf{v}_j$) to another (e.g. $\mathbf{v}_i$) augments the information that might be missing in $\mathbf{v}_i$, but can be provided by $\mathbf{v}_j$. Specifically, we define $\alpha^g_{i,j}$ as

$$\alpha'^g_{i,j} = W^g_\alpha(\mathbf{v}'_i \,||\, \mathbf{v}'_j), \ \mathbf{v}'_i = \tanh(W^g_v \mathbf{v}_i), \tag{6}$$

$$\alpha^g_{i,j} = \frac{\exp(\alpha'^g_{i,j})}{\sum_{j=1}^{N_r} \exp(\alpha'^g_{i,j})}, \ i = 1, ..., N_r, \tag{7}$$

where $||$ denotes concatenation, $\tanh(\cdot)$ is the hyperbolic tangent activation function, $W^g_v \in \mathbb{R}^{D_w \times D_v}$, and $W^g_\alpha \in \mathbb{R}^{1 \times 2D_w}$. Concatenation is used to associate any two feature vectors, i.e. $\mathbf{v}'_i$ and $\mathbf{v}'_j$ to learn how much importance $\mathbf{v}_j$ has to $\mathbf{v}_i$ through W^g_α and W^g_v. It is worth noting that applying concatenation imposes a *directedness* assumption on the link between any two regional features $\mathbf{v}_i$ and $\mathbf{v}_j$ since, in general, $\alpha_{i,j} \neq \alpha_{j,i}$, when $i \neq j$.

3.3.2 Visually Similar Relations $\alpha^v_{i,j}$. We introduce two visual relations based on dot-product and cosine distance. We categorize the relation modules based on these two operations together because they naturally capture the similarity between two representations and can help enhance the visual signals by identifying other similar ones.

Scaled Dot-Product: Firstly introduced in [19], scaled dot-product (SDP) attention mechanism calculates $\alpha^s_{i,j}$ as

$$\alpha^s_{i,j} = \frac{(W^s_{v_1} \mathbf{v}_i) \cdot (W^s_{v_2} \mathbf{v}_j)}{\sqrt{D_w}}, \tag{8}$$

where $W^s_{v_1}, W^s_{v_2} \in \mathbb{R}^{D_w \times D_v}$. What is worth noting is that $\alpha^s_{i,j}$ in our framework is used to weight $\mathbf{v}_i$ directly, whereas it is used to weight another embedding projected from $\mathbf{f}_i$ in [19].

Cosine Similarity: Eq. (8) learns the attention weights according to the correlation of $W^s_{v_1} \mathbf{v}_i$ and $W^s_{v_2} \mathbf{v}_i$ measured by the dot-product. Used for learning the attention weighting in Neural Turing Machine [4], cosine similarity measures the angle between vectors:

$$\alpha^c_{i,j} = \frac{(W^s_{v_1} \mathbf{v}_i) \cdot (W^s_{v_2} \mathbf{v}_j)}{||W^s_{v_1} \mathbf{v}_i|| \cdot ||W^s_{v_2} \mathbf{v}_j||}. \tag{9}$$

We model the relational weight $\alpha^v_{i,j}$, which is determined by visual similarity between two vectors in Eq. (5), with either $\alpha^s_{i,j}$ or $\alpha^c_{i,j}$, i.e.

$$\alpha^v_{i,j} = \gamma^s \alpha^s_{i,j} + \gamma^c \alpha^c_{i,j}, \tag{10}$$

where $\gamma^s, \gamma^c \in \{0, 1\}$ are hyperparameters deciding either $\alpha^s_{i,j}$ or $\alpha^c_{i,j}$ to be adopted. This marks the difference between $\alpha^v_{i,j}$ and $\alpha^g_{i,j}$ where the latter learns to identify dependent context with respect to the representation $\mathbf{v}_i$.

3.3.3 Geometry Relations $\alpha^\omega_{i,j}$. Relative geometry relation that encodes the spatial relationship between two proposals has shown to be important when modeling contextual information [8, 27]. We model it with $\alpha^\omega_{i,j}$ [8], where

$$\alpha^\omega_{i,j} = f^\omega(W^\omega_2 \sigma^\omega(W^\omega_1 \mathbf{b}_{i,j})), \tag{11}$$

$$\mathbf{b}_{i,j} = [\log(\frac{|x_i - x_j|}{w_i}), \log(\frac{|y_i - y_j|}{h_i}), \log(\frac{w_i}{w_j}), \log(\frac{h_i}{h_j})]^T. \tag{12}$$

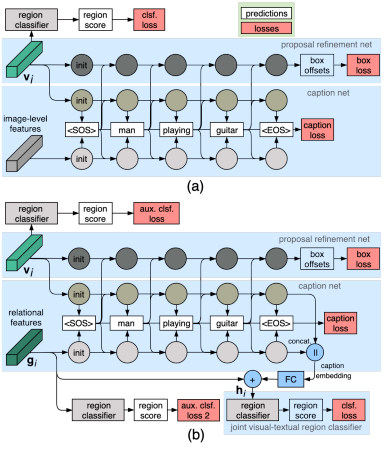


Figure 3: Architectures of (a) Yang et al. [26] and (b) our proposed model. Both architectures consist of three RNN branches which comprise the proposal refinement and caption nets. The proposed model is empowered by the features learned with GREatt and a joint visual-textual region classifier.

$\mathbf{b}_{i,j}$ is the geometry features encoded by center coordinates (x_*, y_*), width and height of the bounding box w_*, and h_*. Since $x_i - x_j$ or $y_i - y_j$ can be zero, we set a lower bound (i.e. 10^{-3}) on them. $f^\omega : \mathbb{R} \rightarrow \mathbb{R}_{\geq 0}$ can be: 1) $\max(x, 10^{-3})$ similar to ReLU or 2) a softmax operation. We empirically find that $\omega_{i,j}$ tends to be rather uniformly distributed when learning it with ReLU for any fixed i and $j = 1, ..., N_r$. Hence, we adopt softmax in f^ω throughout the experiments.

3.3.4 Contextual Function f_α. Before introducing how one can define the contextual function f_α in Eq. (5), we would like to emphasize the differences in three visual relationships, defined in Eqs. (7)-(9). We hypothesize that the first scheme (i.e. concatenation-based) learns how to identify the essential contextual cues with respect to each proposal, while the latter two (similarity-based) learn how to enhance the evidence on the existence of the similar content to be recognized. This further leads to the assumption that these two types of interactions in visual domain can potentially provide distinct contextual information. With this hypothesis, we write the contextual function f_α as

$$f_\alpha(\alpha^g_{i,j}, \alpha^v_{i,j}, \alpha^\omega_{i,j}) = \frac{\alpha^\omega_{i,j} \exp(\gamma^g \alpha^g_{i,j} + \alpha^v_{i,j})}{\sum_{j=1}^{N_r} \alpha^\omega_{i,j} \exp(\gamma^g \alpha^g_{i,j} + \alpha^v_{i,j})}, \tag{13}$$

where γ^g is predefined hyperparameters. $\alpha^\omega_{i,j}$, $\alpha^g_{i,j}$ $\alpha^v_{i,j}$ are defined in Eqs. (11), (7), and (10), respectively. We empirically validate the hypothesis by studying the quantities of the attentions (provided in Figure 5) estimated from different schemes.

3.4 Proposal Refinement and Caption Nets

Yang et al. [26] proposed a triple-stream RNN architecture (shown in Fig. 3(a)) for refining the proposals generated by the region proposal network (RPN) [14] and generating the captions. We mainly follow a similar architecture, i.e. the proposal refinement net RNN^r_t, and the caption nets composed by RNN^v_t and RNN^g_t, $t = 0, ..., T+1$, where t indexes the RNN steps with $T+1$ being the maximal length of a caption including the start (<SOS>) and end (<END>) symbols. At step t, each RNN^*_t receives a word predicted in step $(t-1)$ and updates its hidden states $h^*_t \in \mathbb{R}^{D_r}$ and cell states $c^*_t \in \mathbb{R}^{D_r}$.

The main difference between the proposed architecture and that in [26] can be seen in Figure 3. While RNN^r_0 and RNN^v_0 take $\mathbf{v}_i$ as input, RNN^g_0 is fed with the context features $\mathbf{g}_i$ learned with GREatt instead of image-level features. The hidden state h^r_τ is used to predict the offsets to the x and y coordinates, width and height of the region proposals with a MLP, where τ is the step that predicts (<END>). As for caption branches, h^v_t and h^g_t are concatenated to make a prediction on the distribution of the next word through another MLP. The proposed context features $\mathbf{g}_i$, adapted with respect to each region, are endowed with contextual relationships captured in the scene. By contrast, the image-level features devised by [26] in Figure 3(a) can only provide a fixed and generic guidance to all the regions to be captioned.

3.5 Joint Visual-Textual Region Classifier

Conventionally, the region classifier estimates $P(c_i|\mathcal{V})$ which indicates that the prediction is purely conditioned on corresponding regional features. In this work, we aim to improve the classifier by replacing the target of estimation with $P(c_i|\hat{B}_i, S_i, \mathcal{G}, \mathcal{V})$, as shown in Eq. (1), which additionally considers the learned relationships among the proposals and the caption information. Specifically, we estimate $P(c_i|\cdot)$ with

$$P(c_i|\mathcal{I}, \hat{\mathcal{B}}_i, \mathcal{S}_i, \mathcal{G}, \mathcal{V}) = \text{MLP}_{rc}(\mathbf{h}_i), \tag{14}$$

$$\mathbf{h}_i = \mathbf{g}_i + W^r(\mathbf{h}^v_\tau || \mathbf{h}^g_\tau). \tag{15}$$

In the above equation, the relational representation $\mathbf{g}_i$ is defined in Eq. (4), c_i is the class label defined in Eq. (1), $\text{MLP}_{rc}(\cdot)$ represents a MLP with a sigmoid activation function placed at the output, and $W^r \in \mathbb{R}^{D^v \times 2D^r}$ is learned to project the caption embedding to the same domain in which the visual features reside. The rationale behind this approach is two-fold:

1) **Better vision-caption consistency:** Projecting (or "translating") caption embedding back to the visual domain in which the classification is performed can potentially improve the model's consistency between the generated caption embedding and the embedding of the visual counterpart.

2) **Mimicking human annotator's behavior:** We hypothesize that two actions in the annotation process, i.e. 1) sizing up the bounding boxes around the interesting contents and 2) captioning, are bonded in both directions. A human annotator's attention may be drawn to a relatively salient object, caption it, and then refine the bounding area and the caption. This indicates that the caption information can as well provide evidence to infer the region saliency.

3.6 The Losses

The proposed model is trained by minimizing the total loss L addressing all sub-tasks, i.e. the region proposal (**RP**), region classification (**RC**), proposal refinement (**PR**), and caption generation

(**CG**) sub-tasks as presented in Sec. 3.1. Specifically,

$$L = L^{RP} + L^{RC} + L^{PR} + L^{CG}, \tag{16}$$

$$L^{RP} = \alpha_1 L^{RP}_{det} + \alpha_2 L^{RP}_{box}, \tag{17}$$

$$L^{RC} = \beta(L^{RC}_{v} + L^{RC}_{g} + L^{RC}_{h}), \tag{18}$$

$$L^{PR} = \gamma L^{PR}_{box}, \tag{19}$$

$$L^{CG} = L^{cap}, \tag{20}$$

where

$$L^{RP}_{det} = \alpha_r \sum_{i=1}^{N_r} L^{RP}_{det,i}, L^{RP}_{box} = \alpha_r \sum_{i=1}^{N_r} L^{RP}_{box,i}, \tag{21}$$

$$L^{RC}_{v} = \alpha_r \sum_{i=1}^{N_r} L^{RC}_{v,i}, L^{RC}_{g} = \alpha_r \sum_{i=1}^{N_r} L^{RC}_{g,i}, L^{RC}_{h} = \alpha_r \sum_{i=1}^{N_r} L^{RC}_{h,i}, \tag{22}$$

$$L^{PR}_{box} = \frac{1}{|\mathbf{pos}|} \sum_{i \in \mathbf{pos}} L^{PR}_{box,i}, L^{cap} = \frac{1}{|\mathbf{pos}|} \sum_{i \in \mathbf{pos}} L^{cap}_{i}, \tag{23}$$

$\alpha_r = \frac{1}{N_r}$ is a normalization factor, **pos** represents the set of positive regions in the batch of N_r regions, and $|\mathbf{pos}|$ denotes the size of the set. α_1, α_2, β, and γ are hyperparameters.

RP Losses. Per-sample losses for training RPN are the detection loss $L^{RP}_{det,i}$ and regression loss $L^{RP}_{box,i}$. The former is defined as the cross-entropy function over the predicted and the ground-truth classes, in which the classes refer to either $c_i = 0$, negative non-captioned regions, or $c_i = 1$, positive captioned regions. The latter loss is defined by the smooth L1 function used in [14].

RC Losses. Region classification involves three losses with respect to $\mathbf{v}_i$, $\mathbf{g}_i$, and $\mathbf{h}_i$, respectively. These three losses are defined as the cross-entropy function over the predicted and the ground-truth classes. $L^{RC}_{v,i}$, $L^{RC}_{g,i}$, and $L^{RC}_{h,i}$ are evaluated based on the ground-truth classes and the predicted classes given by $\mathrm{MLP}_{rc}(\mathbf{v}_i)$, $\mathrm{MLP}_{rc}(\mathbf{g}_i)$, and $\mathrm{MLP}_{rc}(\mathbf{h}_i)$, respectively. As we take the predictions from $\mathrm{MLP}_{rc}(\mathbf{h}_i)$ during evaluation, $\mathrm{MLP}_{rc}(\mathbf{v}_i)$ and $\mathrm{MLP}_{rc}(\mathbf{g}_i)$ are treated as auxiliary predictions which are meant for enhancing the discriminative power of individual $\mathbf{v}_i$ and $\mathbf{g}_i$. Note that these three predictions share the same set of parameters from $\mathrm{MLP}_{rc}(\cdot)$. Minimizing L^{RC} corresponds to minimizing E_{cls} in Eq. (1).

PR Loss. Proposal refinement loss $L^{PR}_{box,i}$ in Eq. (23), same as $L^{RP}_{box,i}$, is defined by the smooth L1 function over coordinates of the predicted box and the ground-truth box. Note that minimizing L^{PR}_{box} corresponds to minimizing E_{box} in Eq. (2).

CG Loss. Caption generation loss L^{cap}_{i}, defined over word distributions in i^{th} ground-truth caption and predicted word distribution, is measured by the cross-entropy function. Minimizing L^{cap} corresponds to minimizing E_{cap} in Eq. (3).

4 EXPERIMENTS

4.1 Dataset

All the experiments are conducted on the Visual Genome dataset [11], created for various vision-language tasks such as dense captioning, VQA, and SGG. For the DC task, the annotations with region bounding boxes and corresponding captions are provided. Even though three versions, V1.0, V1.2, and V1.4 are available, we compare different DC models on V1.2 since the changes in V1.4 do not affect the data used in the DC task, and the state-of-the-art models are extensively evaluated mainly on V1.2 [26].

4.2 Experimental Setting

Following the split protocol provided in [9, 26], the images are divided into training, validation, and test sets, comprising 77398, 5000, and 5000 images, respectively. The provided bounding box annotations are often highly overlapping, hence all the annotations with IoU > 0.7 of their bounding boxes are merged into one [26]. Accordingly, each merged region across all sets can contain multiple reference captions, in which a caption for a merged region is randomly drawn during training. The parameter settings in the RPN strictly follow those in [26].

4.3 Hyperparameter Setting and Model Training

The hyperparameters defined in Eqs. (21)–(23) are given by $\alpha_1 = 0.1$, $\alpha_2 = 0.05$, $\beta = 0.1$, and $\gamma = 0.01$. The input image is resized so that the longer side is of 720 pixels. The most frequent 10,000 words are used and those excluded are replaced with an `<UNK>` (unknown word) symbol. Hence, this amounts to 10,003 words (10,000 most frequent words plus `<SOS>`, `<END>`, and `<UNK>`) available for the caption model. Regions with captions longer than 10 words are discarded, and each caption of the remaining ones is padded with `<SOS>` in the beginning and `<EOS>` at the tail. The proposal refinement and caption nets adopt three seperate LSTMs with 512 hidden units. The experiments with three visual features: VGG16 [17], which has two fully-connected layers both consisting of 4096 units at the output, extracts 4096-dimensional features for each region proposal. ResNet50 and ResNet101 [5] extract 1024-dimensional features. The training batch size is set to be 1 (i.e. a single image) with $N_r = 256$ (referred in Eqs. (21)–(23)) region proposals evenly sampled from positive and negative proposals in the RPN.

All the models throughout the experiments are trained with stochastic gradient descent with momentum set at 0.98. The initial learning rate is 0.001, reduced by half every 100,000 steps ($\approx$ 1.3 epochs). Models with VGG16 are trained only with `Conv4_*` and `Conv5_*` being fine-tuned in the periods of 1.5–4 epochs and 5.5–10 epochs. Models with ResNet50 and ResNet101 are trained with 4^{th} residual block being fine-tuned in 0–1.5 epochs and 4–5.5 epochs, and 3^{rd} residual block as well being fine-tuned in the periods of 1.5–4 epochs and 5.5–10 epochs. We follow the stage-wise training scheme suggested in [26] to train the proposed models. Firstly, we train the RPN, the proposal refinement net, and the caption net end-to-end. Here at this stage, only one caption LSTM (i.e. RNN^{v}_{t}, but not RNN^{g}_{t}) which receives the regional features $\mathbf{v}_i$ is trained. Secondly, we add the second LSTM stream RNN^{g}_{t} with the context features $\mathbf{g}_i$ into the models and fine-tune the other parts. Finally, we fine-tune the models and feed the region classifier MLP_{rc} with $\mathbf{h}_i$ of Eq. (15), the features containing both visual and caption embedding. This training scheme helps the models in which the performance of each component is based upon each other, e.g., the proposal refinement net can only start to refine the proposals generated by the RPN once the RPN has learned to produce reasonable proposals.

4.4 Evaluation Metric

The main metric adopted to evaluate the DC models is the mean average precision (mAP) that jointly considers the goodness of the region proposals and the generated captions in terms of IoU and METEOR [2] scores with the ground-truth annotations [9]. mAP is calculated by averaging the average precision scores evaluated at different IoU thresholds, {0.3, 0.4, 0.5, 0.6, 0.7}, and METEOR thresholds, {0, 0.05, 0.1, 0.15, 0.2, 0.25}. Besides, we also adopt mAP@{IoU=0.3,0.4,0.5,0.6,0.7} and mAP@{*small*,*medium*,*large*} (evaluated at proposals smaller than 48^2, between $48^2 - 108^2$, and larger than 108^2 pixels) to facilitate a deeper comparison between models.

Table 1: The representation of different attention modules defined by γ^g and γ^v in Eqs. (13) and (10). The geometry relationship captured by $\alpha^{\omega}_{i,j}$ is considered by all different modules listed.

models	ctx	sim(sdp)	sim(cos)	ctx+sim(sdp)	ctx+sim(cos)
(γ^g, γ^s, γ^c)	(1,0,0)	(0,1,0)	(0,0,1)	(1,1,0)	(1,0,1)

4.5 Quantitative Comparison

We compare the proposed framework with the state-of-the-art DC models [26]. The pioneer DC framework from Johnson et al. [9] reported the performance of their models on Visual Genome V1.0, and thus a direct comparison with their results is not possible. It is difficult to compare results also from many other different DC models since, to the best of our knowledge, the only notable and reliable results one can compare against are from [26]. In the following subsections, we compare different models of our own with configurations listed in Table 1 and those described in [26].

4.5.1 Comparing with State of the Art. We have tried our best to replicate the best performing architecture reported in [26], and the highest mAP we can obtain is 9.72, which is reasonably close to 9.96 reported in their work. First, we study whether the models with added geometry relation and a single visual attention mechanism can improve over those without. The results in the second to the fourth rows (against those in the first row) in Table 2 highlight the effect of a model that considers a single visual relationship (implemented by either $\alpha^g_{i,j}$, $\alpha^s_{i,j}$, or $\alpha^c_{i,j}$, referred in Sec. 3.3.1 and 3.3.2) and the geometry relationship captured by $\alpha^{\omega}_{i,j}$ (referred in Sec. 3.3.3). We observe the consistent improvement made by the proposed models in the mAP across VGG16, ResNet50, and ResNet101 visual features.

Moving to the fifth row onwards in Table 2, one can observe the best mAP is obtained from the proposed architecture when GREatt (with geometry, concatenation-based, cosine distance based attention modules simultaneously employed) and caption-boosted classifier (described in Sec. 3.5) are used. The best result with VGG16 achieves 10.23, which, to date, surpasses the state-of-the-art number that has been reported. A greater margin of improvement in mAP can be observed (+5.3%, +5.4%, +6.23% with VGG16, ResNet50, and ResNet101, respectively) when comparing the best performing models of ours and those in [26].

We also report the mAP at different proposal sizes in Table 3. One can easily observe a similar trend where our architectures bring steady improvement for all proposal size groups. This shows that our models do not favor proposals of certain sizes, but provide all-around improvement over arbitrary sizes of proposals. Moreover, the largest improvement often comes from the mAP@small, indicating that our context modeling scheme has the largest positive impact on making inference on the small region proposals.

4.5.2 Comparing Models with Different Attention Modules. Here, we study the effect on varying computational attention modules proposed. The aim of the study is to answer whether (1) models with GREatt employing one geometry and two visual attention mechanisms (out of three presented in Sec. 3.3.1 and 3.3.2), improves the results over those with one geometry and a single visual attention mechanisms, and (2) models equipped with the region classifier exposed with caption information improves the results over those without.

Fusing attentions. From Table 2, one can also compare two types of models: (1) those with combined visual attentions (presented in the fifth to sixth rows) and (2) those with single visual attention (presented in the second to fourth rows). We compare them by picking the best result (e.g. mAP) that a model in each type can achieve. One can observe the improvement in mAP made by the models with combined visual attentions on VGG16 and ResNet50, but not on ResNet101.

Classifying regions with captions. From Table 2, one can observe a significant improvement made by the models with the caption-boosted region classifier based on all visual feature extractors. From Table 3, we see that the largest improvements are made on mAP@small, demonstrating that the caption information is crucial to make smaller RoIs detectable.

4.6 Qualitative Results

We compare qualitative results from our model (i.e. the best performing one, "ctx+sim(cos)" model listed in Table 1) and the one from [26] with ResNet101 features in Figure 4. Clearly shown, Yang's model tends to ignore the relationship (Figure 4(a): missing "on a cutting board"), or fail to encode the context (e.g. Figure 4(e): missing "laptop" in the caption). By contrast, our proposed model not only captures the correct relationships, but also correctly recognizes and names the objects in the context.

Next, we study attention weights (i.e. $\alpha^{\omega}_{i,j}$, $\alpha^g_{i,j}$, and $\alpha^c_{i,j}$) learned to capture different relationships in Figure 5. One can observe that three types of weights attend to quite distinct and sometimes complementary sets of areas with respect to each proposal. While the $\alpha^{\omega}_{i,j}$ and $\alpha^g_{i,j}$ tend to capture the necessary context (i.e. the tennis field in this example), cosine distance based visual attention $\alpha^c_{i,j}$ tends to capture visually similar context. For example, while the subject in the proposal is the tennis player in the distance, it tries to retrieve similar person-like objects. The combined attention is able to capture the most relevant context, e.g. in Figure 5(c), it identifies who is holding the racket, and in Figure 5(d), it captures almost the whole tennis court to be able to recognize that the clock is in the court.

5 CONCLUSIONS

In this paper, we visited the dense captioning task, which serves as a powerful means to facilitate multimodal context understanding

Table 2: Quantitative results of models with VGG16, ResNet50, and ResNet101, respectively, on Visual Genome V1.2. `models` column shows models with varying visual attention modules named in Table 1. `cap` indicates if the caption embedding is added when classifying the region proposals. The best model with respect to each metric is highlighted in bold, and the second best is underlined. (*) indicates the figure reported in [26] while the other figures are obtained from our implementation. @n indicates the mAP score evaluated at IoU=n, $n = \{0.3, 0.4, 0.5, 0.6, 0.7\}$.

		VGG16						ResNet50						ResNet101					
models	cap	mAP	@0.3	@0.4	@0.5	@0.6	@0.7	mAP	@0.3	@0.4	@0.5	@0.6	@0.7	mAP	@0.3	@0.4	@0.5	@0.6	@0.7
Yang et al. [26]	-	9.72 (9.96*)	15.13	13.16	10.25	6.77	3.28	10.89	16.85	14.62	11.55	7.73	3.70	11.92	18.16	15.83	12.58	8.68	4.37
ctx	-	9.85	15.22	13.25	10.41	6.96	3.39	11.00	16.48	14.52	11.70	8.12	4.14	12.51	17.73	15.76	12.84	9.14	4.82
sim(sdp)	-	9.88	15.29	13.32	10.44	6.96	3.39	11.00	16.51	14.58	11.68	8.11	4.12	11.79	17.95	15.67	12.43	8.59	4.30
sim(cos)	-	9.73	15.10	13.15	10.29	6.78	3.33	11.07	17.09	14.85	11.77	7.90	3.75	11.73	17.91	15.64	12.40	8.49	4.20
ctx+sim(sdp)	-	9.97	15.33	13.40	10.55	**7.06**	**3.48**	11.03	16.54	14.63	11.70	8.14	4.14	12.14	18.37	16.09	12.87	8.88	4.96
ctx+sim(cos)	-	9.93	15.90	13.59	10.36	6.68	3.11	11.10	16.62	14.73	11.77	8.20	4.19	12.15	18.37	16.09	12.88	8.90	4.48
ctx+sim(sdp)	✓	10.22	16.30	14.00	10.71	6.91	3.14	11.39	17.03	14.98	12.09	8.43	4.40	12.52	**18.72**	**16.37**	13.23	9.34	4.93
ctx+sim(cos)	✓	**10.23**	**16.39**	**14.04**	**10.76**	6.85	3.13	**11.48**	**17.14**	**15.08**	**12.15**	**8.56**	**4.45**	**12.67**	18.39	16.32	**13.44**	**9.79**	**5.40**

Table 3: Results on comparing models on mAP@{small, medium, large}, denoted by @S, @M, @L.

		VGG16			ResNet50			ResNet101		
models	cap	@S	@M	@L	@S	@M	@L	@S	@M	@L
Yang et al. [26]		3.99	8.15	14.22	4.09	9.08	16.03	4.78	9.94	17.35
ctx	-	4.03	8.39	14.46	4.19	8.82	16.26	4.61	9.98	17.82
sim(sdp)	-	4.00	8.25	14.36	4.33	8.97	16.24	4.28	9.57	17.44
sim(cos)	-	4.14	8.23	14.19	4.46	9.21	16.16	4.36	9.68	17.42
ctx+sim(sdp)	-	3.83	8.34	**14.63**	4.35	9.13	16.22	4.46	9.86	17.79
ctx+sim(cos)	-	3.95	8.53	14.24	4.48	9.11	16.27	4.52	10.15	17.71
ctx+sim(sdp)	✓	4.19	**8.63**	14.46	4.25	**9.52**	16.44	4.83	**10.55**	18.14
ctx+sim(cos)	✓	**4.39**	8.61	14.42	**4.68**	9.34	**16.80**	**4.94**	10.48	**18.39**

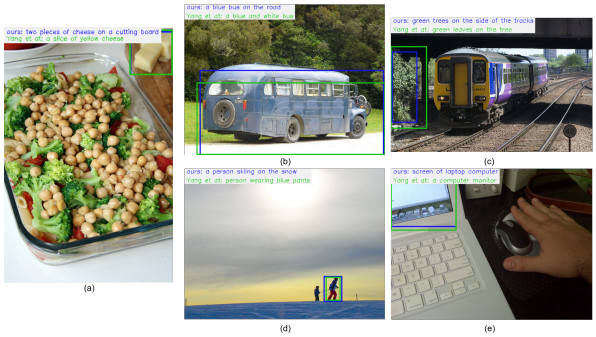


Figure 4: Qualitative comparison between the proposed method and that proposed by Yang et al. [26]. More relationships and context information are revealed in the captions generated by our method. Captions (ours / [26]): (a) two pieces of cheese on a cutting board / a slice of yellow cheese, (b) a blue bus on the road / a blue and white bus, (c) green trees on the side of the tracks / green leaves on the tree, (d) a person skiing on the snow / person wearing blue pants, (e) screen of laptop computer / a computer monitor.

and learning. We proposed an improved architecture which features (1) a Geometry-aware Relational Exemplar attention (GREatt) mechanism and (2) a joint visual-textual relational region classifier, for the dense captioning problem. Our proposed methods bring significant improvements over the state-of-the-art results. In addition,

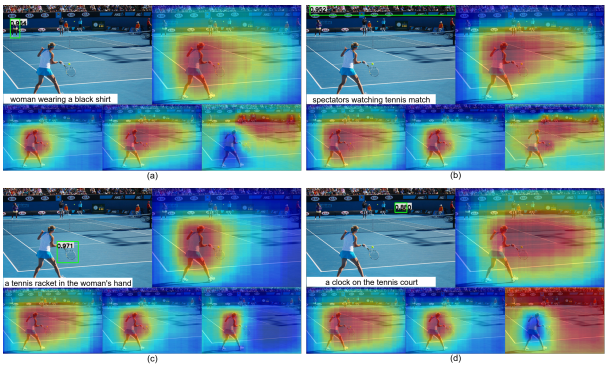


Figure 5: Different attention mechanisms jointly learned with model "ctx+sim(cos)" (referred in Table 1). Each set of image, from top to bottom, left to right, shows 1) detection and caption results, 2) combined attention, $\alpha_{i,j}$, 3) geometry attention, $\alpha^{\omega}_{i,j}$, 4) contextual dependent visual attention, $\alpha^{g}_{i,j}$, and 5) visually similar and supported attention, $\alpha^{c}_{i,j}$.

we demonstrated that GREatt captures varying and meaningful contexts for different regions to construct contextually dependent and region-specific features. The proposed region classifier which learns on the subspace shared with visual and textual embeddings has also demonstrated its effectiveness and led to improvements in almost all metrics. Qualitatively, our proposed models, comparing to the prior arts, are more capable of generating captions that capture relationships between objects and are able to accurately recognize and name the objects in the context. However, how to optimally combine the heterogeneous types of attention still remains an open question, and we leave it as a future avenue of research.

ACKNOWLEDGMENTS

This work has been funded by the Academy of Finland project number 313988 (DeepGraph), and the European Union's Horizon 2020 research and innovation programme under grant agreement No. 780069 (MeMAD). We also acknowledge the Aalto University's Aalto Science IT project and CSC âĂŞ IT Center for Science Ltd. for providing computer resources and NVIDIA Corporation for donation of GPU for this research.

REFERENCES

[1] Peter Anderson, Xiaodong He, Chris Buehler, Damien Teney, Mark Johnson, Stephen Gould, and Lei Zhang. 2018. Bottom-up and top-down attention for image captioning and visual question answering. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 6077–6086.

[2] Satanjeev Banerjee and Alon Lavie. 2005. METEOR: An automatic metric for MT evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*. 65–72.

[3] Ali Farhadi, Mohsen Hejrati, Mohammad Amin Sadeghi, Peter Young, Cyrus Rashtchian, Julia Hockenmaier, and David Forsyth. 2010. Every picture tells a story: Generating sentences from images. In *European conference on computer vision*. Springer, 15–29.

[4] Alex Graves, Greg Wayne, and Ivo Danihelka. 2014. Neural turing machines. *arXiv preprint arXiv:1410.5401* (2014).

[5] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. 2016. Deep residual learning for image recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*. 770–778.

[6] Sen He, Hamed R. Tavakoli, Ali Borji, and Nicolas Pugeault. 2019. A Synchronized Multi-Modal Attention-Caption Dataset and Analysis. *CoRR* abs/1903.02499 (2019). arXiv:1903.02499 http://arxiv.org/abs/1903.02499

[7] MD. Zakir Hossain, Ferdous Sohel, Mohd Fairuz Shiratuddin, and Hamid Laga. 2019. A Comprehensive Survey of Deep Learning for Image Captioning. *ACM Comput. Surv.* 51, 6, Article 118 (Feb. 2019), 36 pages. https://doi.org/10.1145/3295748

[8] Han Hu, Jiayuan Gu, Zheng Zhang, Jifeng Dai, and Yichen Wei. 2018. Relation networks for object detection. In *Computer Vision and Pattern Recognition (CVPR)*, Vol. 2.

[9] Justin Johnson, Andrej Karpathy, and Li Fei-Fei. 2016. Densecap: Fully convolutional localization networks for dense captioning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 4565–4574.

[10] A. Karpathy and L. Fei-Fei. 2017. Deep Visual-Semantic Alignments for Generating Image Descriptions. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 39, 4 (April 2017), 664–676. https://doi.org/10.1109/TPAMI.2016.2598339

[11] Ranjay Krishna, Yuke Zhu, Oliver Groth, Justin Johnson, Kenji Hata, Joshua Kravitz, Stephanie Chen, Yannis Kalantidis, Li-Jia Li, David A Shamma, et al. 2017. Visual genome: Connecting language and vision using crowdsourced dense image annotations. *International Journal of Computer Vision* 123, 1 (2017), 32–73.

[12] Tomáš Mikolov, Martin Karafiát, Lukáš Burget, Jan Černockỳ, and Sanjeev Khudanpur. 2010. Recurrent neural network based language model. In *Eleventh annual conference of the international speech communication association*.

[13] Marco Pedersoli, Thomas Lucas, Cordelia Schmid, and Jakob J. Verbeek. 2017. Areas of Attention for Image Captioning. *2017 IEEE International Conference on Computer Vision (ICCV)* (2017), 1251–1259.

[14] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. 2015. Faster r-cnn: Towards real-time object detection with region proposal networks. In *Advances in neural information processing systems*. 91–99.

[15] Adam Santoro, David Raposo, David G Barrett, Mateusz Malinowski, Razvan Pascanu, Peter Battaglia, and Timothy Lillicrap. 2017. A simple neural network module for relational reasoning. In *Advances in neural information processing systems*. 4967–4976.

[16] Rakshith Shetty, Hamed Rezazadegan Tavakoli, and Jorma Laaksonen. 2018. Image and Video Captioning with Augmented Neural Architectures. *IEEE MultiMedia* 25, 2 (2018), 34–46. https://doi.org/10.1109/MMUL.2018.112135923

[17] K. Simonyan and A. Zisserman. 2015. Very Deep Convolutional Networks for Large-Scale Image Recognition. In *International Conference on Learning Representations*.

[18] Hamed R. Tavakoli, Rakshith Shetty, Ali Borji, and Jorma Laaksonen. 2017. Paying Attention to Descriptions Generated by Image Captioning Models. *2017 IEEE International Conference on Computer Vision (ICCV)* (2017), 2506–2515.

[19] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems*. 5998–6008.

[20] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph Attention Networks. *International Conference on Learning Representations* (2018). https://openreview.net/forum?id=rJXMpikCZ

[21] O. Vinyals, A. Toshev, S. Bengio, and D. Erhan. 2015. Show and tell: A neural image caption generator. In *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*. 3156–3164. https://doi.org/10.1109/CVPR.2015.7298935

[22] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. 2018. Non-local neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 7794–7803.

[23] Sanghyun Woo, Dahun Kim, Donghyeon Cho, and In So Kweon. 2018. LinkNet: Relational Embedding for Scene Graph. In *Advances in Neural Information Processing Systems*. 558–568.

[24] Kelvin Xu, Jimmy Ba, Ryan Kiros, Kyunghyun Cho, Aaron Courville, Ruslan Salakhudinov, Rich Zemel, and Yoshua Bengio. 2015. Show, attend and tell: Neural image caption generation with visual attention. In *International conference on machine learning*. 2048–2057.

[25] Jianwei Yang, Jiasen Lu, Stefan Lee, Dhruv Batra, and Devi Parikh. 2018. Graph r-cnn for scene graph generation. In *Proceedings of the European Conference on Computer Vision (ECCV)*. 670–685.

[26] Linjie Yang, Kevin D Tang, Jianchao Yang, and Li-Jia Li. 2017. Dense Captioning with Joint Inference and Visual Context.. In *CVPR*. 1978–1987.

[27] Ting Yao, Yingwei Pan, Yehao Li, and Tao Mei. 2018. Exploring visual relationship for image captioning. In *European Conference on Computer Vision (ECCV)*. 684–699.

B.10 AALTO's paper in CAIP 2019 conference [9]

This paper describes the development of deep neural network based techniques for indoor scene recognition in which researchers at AALTO have participated. The results may later have influence on the work in MeMAD. It has been published in the CAIP 2019 conference.

Multi-stream Convolutional Networks for Indoor Scene Recognition

Rao Muhammad Anwer[1,2( )], Fahad Shahbaz Khan[2,3( )], Jorma Laaksonen[1( )], and Nazar Zaki[4( )]

[1] Department of Computer Science, Aalto University School of Science, Espoo, Finland
{rao.anwer,jorma.laaksonen}@aalto.fi
[2] Inception Institute of Artificial Intelligence, Abu Dhabi, UAE
[3] Computer Vision Laboratory, Linköping University, Linköping, Sweden
fahad.khan@liu.se
[4] Computer Science and Software Engineering Department, College of Information Technology, United Arab Emirates University, Al Ain, UAE
Nzaki@uaeu.ac.ae

Abstract. Convolutional neural networks (CNNs) have recently achieved outstanding results for various vision tasks, including indoor scene understanding. The de facto practice employed by state-of-the-art indoor scene recognition approaches is to use RGB pixel values as input to CNN models that are trained on large amounts of labeled data (ImageNet or Places). Here, we investigate CNN architectures by augmenting RGB images with estimated depth and texture information, as multiple streams, for monocular indoor scene recognition. First, we exploit the recent advancements in the field of depth estimation from monocular images and use the estimated depth information to train a CNN model for learning deep depth features. Second, we train a CNN model to exploit the successful Local Binary Patterns (LBP) by using mapped coded images with explicit LBP encoding to capture texture information available in indoor scenes. We further investigate different fusion strategies to combine the learned deep depth and texture streams with the traditional RGB stream. Comprehensive experiments are performed on three indoor scene classification benchmarks: MIT-67, OCIS and SUN-397. The proposed multi-stream network significantly outperforms the standard RGB network by achieving an absolute gain of 9.3%, 4.7%, 7.3% on the MIT-67, OCIS and SUN-397 datasets respectively.

Keywords: Scene recognition · Depth features · Texture features

1 Introduction

Scene recognition is a fundamental problem in computer vision with numerous real-world applications. The problem can be divided into recognizing indoor

M. Vento and G. Percannella (Eds.): CAIP 2019, LNCS 11678, pp. 196–208, 2019.
https://doi.org/10.1007/978-3-030-29888-3_16

versus outdoor scene types. Initially, most approaches target the problem of outdoor scene classification with methods demonstrating impressive performance on standard benchmarks, such as fifteen scene categories [17]. Later, the problem of recognizing indoor scene categories have received much attention with the introduction of specialized indoor scene datasets, including MIT-67 [23]. Different to outdoor scene categorization, where global spatial layout is distinctive and one of the most discriminative cues, indoor scenes are better characterized either based on global spatial properties or local appearance information depending on the objects they contain. In this work, we investigate the challenging problem of automatically recognizing indoor scene categories.

In recent years, deep convolutional neural networks (CNNs) have revolutionized the field of computer vision setting new state-of-the-art results in many applications, including scene recognition [32]. In the typical scenario, deep networks or CNNs take raw pixel values as an input. They are trained using a large amount of labeled data and perform a series of convolution, local normalization and pooling operations (called layers). Generally, the final layers of a deep network are fully connected (FC) and employed for the classification purpose. Initially, deep learning based scene recognition approaches employed CNNs pre-trained on the ImageNet [26] for object recognition task. These pre-trained deep networks were then transferred for the scene recognition problem. However, recent approaches have shown superior results when training deep networks on a specialized large-scale scene recognition dataset [32]. In all cases, the de facto practice is to use RGB patches as input when training these networks.

As mentioned above, the standard procedure is to employ RGB pixel values as input for training deep networks. Besides color, texture features also provide a strong cue for scene identification at both the superordinate and basic category levels [24]. Significant research efforts have been dedicated in the past in designing discriminative texture features. One of the most successful handcrafted texture descriptors is that of Local Binary Patterns (LBP) and its variants [12,21,22]. LBP is based on the signs of differences of neighboring pixels in an image and is invariant to monotonic gray scale variations. Recent studies [1,4] have investigated employing deep learning to design deep texture representations.

Other than color and texture, previous works [7,11,27,28] have shown the effectiveness of depth information and that depth images can be used simultaneously with RGB images to obtain improved recognition performance. However, most of these approaches require depth data acquired from depth sensors together with camera parameters to associate point clouds to image pixels. Despite increased availability of RGB-D sensors, standard large-scale object and scene recognition benchmarks (ImageNet and Places) still contain RGB images captured using different image sensors with no camera parameters to generate accurate point clouds. In a separate research line, recent works [5,20] have investigated estimating depth information from single monocular images. These methods employ RGB-D acquired through depth sensors during the training stage to infer the depth of each pixel in a single RGB image. Here, we aim

to exploit these advancements in depth estimation from monocular images *and* hand-crafted discriminative texture features to integrate explicit depth and texture information for indoor scene recognition in the deep learning architecture.

In this work, we propose a multi-stream deep architecture where the estimated depth and texture streams are fused with the standard RGB image stream for monocular indoor scene recognition. The three streams can be integrated at different stages in the deep learning architecture to make use of the complementary information available in these different modalities. In the first strategy, the three streams are integrated at an early stage by aggregating the RGB, texture and estimated depth image channels as the input to train a joint multi-stream deep CNN model. In the second strategy, the three streams are trained separately and combined at a later stage of the deep network. To the best of our knowledge, we are the first to propose a multi-stream deep architecture and investigate fusion strategies to combine RGB, estimated depth and texture information for monocular indoor scene recognition. Figure 1 shows example indoor scene categories from the MIT-67 dataset and their respective classification accuracies (in %) when using different streams and their combination in the proposed multi-stream architecture.

	Auditorium	Bedroom	Closet	Grocerystore	kitchen	Dining Room	Classroom	Fastfood Restaurant
RGB	33	57	72	62	52	44	67	53
Depth	56	48	83	48	56	33	61	35
Texture	56	61	72	52	57	50	72	35
Three-Stream	**72**	**67**	**84**	**86**	**71**	**78**	**83**	**94**

Fig. 1. Example categories from MIT-67 indoor scene dataset and their respective classification accuracies (in %) when using different streams: baseline standard RGB, estimated depth and texture. We also show the classification accuracies when combining these streams in our late fusion based three-stream architecture. The classification results are consistently improved with our three-stream architecture, highlighting the complementary information possessed by the three streams.

2 Related Work

Indoor Scene Recognition: Recently, indoor scene recognition has gained a lot of attention [6,8,14–16]. Koskela [16] propose an approach where CNNs, trained on object recognition data, using different architectures are employed as

feature extractors in a standard linear-SVM-based multi-feature scene recognition framework. A discriminative image representation based on discriminative mid-level convolutional activations is proposed by [14] to counter variability in indoor scenes. Guo et al. [6] propose an approach by integrating local convolutional supervision layer that is constructed upon the convolutional layers of deep network. The work of [15] proposes an approach based on spectral transformation of CNN activations integrated as a unitary transformation within a deep network. All these aforementioned deep learning based approaches are trained using RGB pixel values of an image.

Depth Estimation: Recent approaches [5,19,20] employ deep learning to learn depth estimation in monocular images. The work of [5] proposes a multi-scale convolutional architecture for depth prediction, surface normals and semantic labeling. Li et al. [19] introduce an approach by regressing CNN features together with a post-processing refinement step employing conditional random fields (CRF) for depth estimation. The work of [20] proposes a deep convolutional neural field model that jointly learns the unary term and pairwise term of continuous CRF in a unified CNN framework. Different to [5], where the depth map is directly regressed via convolutions from an input image, the approach of [20] explicitly models the relations of neighbouring superpixels by employing CRF. Both unary and binary potentials are learned in a unified deep network framework. Here, we employ deep convolutional neural field model of [20] as a depth estimation strategy for our monocular deep depth network stream. In our multi-stream architecture, the monocular depth stream is trained from scratch, on the large-scale ImageNet and Places datasets, for indoor scene recognition.

Texture Representation: Robust texture description is one of the fundamental problems in computer vision and is extensively studied in literature. Among existing methods, the Local Binary Patterns (LBP) descriptor [22] is one of the most popular hand-crafted texture description methods and several of its variants have been proposed in literature [21]. Recent approaches [1,4] have investigated deep learning for the problem of texture description. Cimpoi et al. [4] propose to encode convolutional layers of the deep network using the Fisher Vector scheme. Rao et al. [1] investigate the problem of learning texture representation and integrate LBP within deep learning architecture. In that approach, LBP codes are mapped to points in a 3D metric space using the approach of [18]. Here, we employ the strategy proposed in [1] to learn the texture stream and combine it with RGB and estimated depth streams in a multi-stream deep architecture for indoor scene recognition.

3 Our Multi-stream Deep Architecture

Here, we present our multi-stream deep architecture for indoor scene recognition. We also investigate fusion schemes to integrate different modalities in the deep learning architecture. We base our approach on the VGG architecture [3] that takes as input an image of 224×224 pixels and consists of five convolutional (conv) and three fully-connected (FC) layers.

3.1 Deep Depth Stream

The first step in designing of the depth stream is to compute the estimated depth image given its RGB counterpart. We employ the method of [20] for depth estimation of each pixel in a monocular image. The depth estimation approach employs continuous CRF to explicitly model the relations of neighbouring superpixels. Both unary and binary terms of continuous CRF are learned in an unified deep network framework. In the depth estimation model, each image is comprised of small regions, termed as superpixels, with nodes of a graphical model defined on them. Each superpixel in an image is described by the depth value of its centroid. Let I be an image and $y = [\mathrm{sp}_1, \ldots, sp_m]^\top \in \mathbb{R}^m$ be a vector of all m superpixels in image I. The conditional probability distribution of the data is then modelled by employing the following density function:

$$P(y \mid I) = \frac{1}{Z(I)} \exp(-EN(y, I)), \tag{1}$$

where EN is the energy function and the partition function represented by Z is defined as:

$$Z(I) = \int_y \exp\{-EN(y, I)\}\, dy. \tag{2}$$

Due to the continuous nature of the depth values y, no approximation method is required to be applied. The subsequent MAP inference problem is then solved in order to obtain the depth value of a new image. The energy function is written as a combination of unary potentials UN and pairwise potentials PV over the superpixels $\mathcal{M}$ and edges $\mathcal{S}$ of the image I:

$$EN(y, I) = \sum_{p \in \mathcal{M}} UN(y_p, I) + \sum_{(p,q) \in \mathcal{S}} PV(y_p, y_q, I), \tag{3}$$

Here, the unary potential UN regresses the depth value for a single superpixel whereas the pairwise potential PV invigorates the superpixel neighborhoods with similar appearances to hold similar depth values. In the work of [20], both the unary potentials UN and the pairwise potentials PV are learned jointly in a unified deep network framework. The deep network comprises the following components: a continuous CRF loss layer consisting of a unary part and a pairwise part. Given an input image, image patches centred around each superpixel centroid are considered. Each image patch is used as an input to the unary part which is fed into the deep network. The network returns a single value representing the regressed depth value of the superpixel. The unary part of the deep network consists of five convolutional and four fully-connected layers. The unary potential is formulated by the output of the deep network by considering the following least square loss:

$$UN(y_p, I; \theta) = (y_p - z_p(\theta))^2, \forall p = 1, \ldots, m, \tag{4}$$

Here, z_p is the regressed depth of the homogeneous region (superpixel) p, parameterized by the deep network parameters θ. In case of the pairwise part of the network, the input is the similarity vectors of all neighboring superpixel pairs, fed to the FC layer with shared parameters among different superpixel pairs. The pairwise term enables neighboring superpixels with similar appearances to have similar depth values. Three types of pairwise similarities are considered: color histogram difference, color difference and texture disparity based on LBP. The output is then a 1-dimensional similarity vector for each of the neighboring superpixel pairs. Consequently, outputs from the unary and the pairwise terms are taken by the continuous CRF loss layer in order to minimize the negative log-likelihood. Standard RGB-D datasets, including NYUD2 have the same viewing angles for both the camera and the depth sensor. This implies that objects in a depth image possess the same 2D shapes as in RGB image with the only difference is that the RGB values are replaced by depth values. The estimated depth images alleviate the problem of intra-object variations, which is desired for scene understanding. During the construction of the depth stream, we first estimate depth values of the input RGB image using the approach described above resulting in a single-channel depth map. The estimated depth values are log-normalized to the range of $[0, 255]$ and duplicated into three channels which are then input to the deep learning framework. Figure 2 shows example RGB images and their corresponding estimated depth maps.

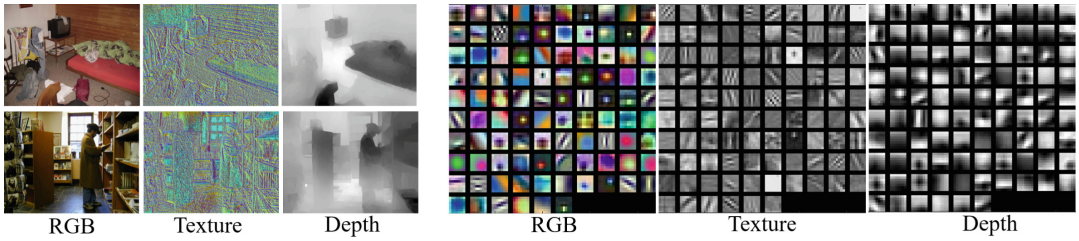


Fig. 2. On the left: example RGB images and the corresponding texture coded mapped images (visualized here in color) together with estimated depth images. On the right: visualization of filter weights from the RGB, texture and estimated depth CNN models.

3.2 Deep Texture Stream

In addition to the standard RGB and estimated depth streams, we propose to integrate an explicit texture stream for indoor scene recognition since texture features have shown to be crucial for scene understanding. Here, we base our texture stream on the popular LBP descriptor [22] where the neighborhood of a pixel is described by its binary derivatives used to form a short code for the neighborhood description of the pixel. These short codes are binary numbers (lower than threshold (0) or higher than the threshold (1)), where each LBP code can be regarded as a micro-texton. Each pixel in the image is allocated a code of the texture primitive with its best local neighborhood match.

When integrating the LBP operator in the deep learning architecture, a straightforward way is to directly employ LBP codes as an input to the deep network. However, the direct incorporation of LBP codes as input is infeasible since the convolution operations, equivalent to a weighted average of the input values, employed within CNNs are unsuitable for the unordered nature of the values of the LBP code. To counter this issue, the work of [18] proposes to map the LBP code values to points in a 3D metric space. In this metric space, the Euclidean distance approximates the distance between the LBP code values. Such a transformation enables averaging of LBP code values during convolution operations within CNN models. First, a distance $\delta_{j,k}$ is defined between the LBP codes $LBPT_j$ and $LBPT_k$. In the work of [18], Earth Mover's Distance (EMD) [25] is employed since it takes into account both the different bit values and their locations. Afterwards, a mapping is derived of the LBP codes into a DM-dimensional space which approximately preserves the distance between them. The mapping is derived by applying Multi Dimensional Scaling (MDS) [2]. The mapping enables the transfer of LBP code values into a representation that is suitable to be used as input to the deep network. As in [1,18], the dimensionality DM is set to three and the resulting texture representation is used to train a texture stream for indoor scene recognition. Figure 2 shows example RGB images and their corresponding texture coded mapped images.

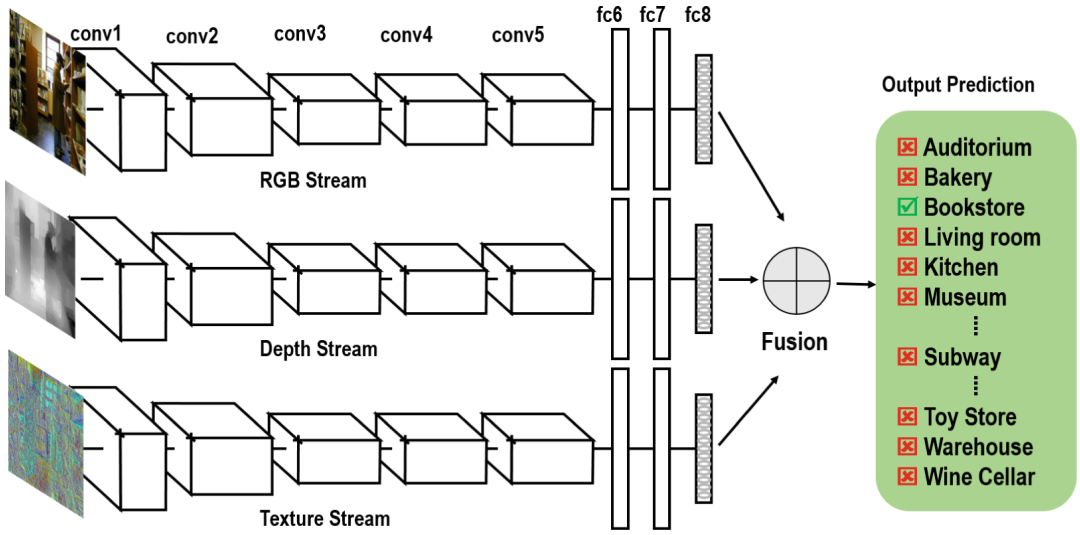


Fig. 3. Our late fusion based multi-stream deep architecture. In this architecture, RGB, estimated depth and texture streams are kept separate and the point of fusion, which combines the three network towers, is at the end of the network.

3.3 Multi-stream Fusion Strategies

We consider two fusion strategies to integrate the RGB, estimated depth and texture streams in a multi-stream architecture. In the first strategy, termed as

early fusion, the three network streams are combined at an early stage as inputs to the deep network. As a result, the input to CNN is of $224 \times 224 \times N$ dimensions, where N is the number of image channels. When combining the three streams in an early fusion strategy, the number of image channels is $N = 7$ (3 RGB, 1 depth and 3 texture). A joint deep model is trained due to the aggregation of the image channels. In the second fusion strategy, termed as late fusion, the three networks are trained separately. The standard RGB stream network takes raw RGB values as input. The texture stream network takes texture coded mapped image as an input to the CNN model. This texture coded mapped image is obtained by first computing the LBP encoding that transforms intensity values in an image to one of the 256 LBP codes. The code values are then mapped into a 3D metric space, as described above, resulting in a 3-channel texture coded mapped image. The depth image is obtained by converting an RGB image to an estimated depth map, based on the procedure described earlier, to be used as an input to the depth stream. Consequently, the three streams are fused at the final stages of the deep network either by using FC layer activations with linear SVMs or combining the score predictions from individual streams. Figure 3 shows our late fusion based three stream architecture. The three streams are separately trained, from scratch, on both ImageNet [26] and Places [32] datasets. Figure 2 shows the VGG architecture based visualization of filter weights from the RGB, texture and estimated depth models trained on the ImageNet.

4 Experimental Results

Experimental Setup: We train our multi-stream network, described in Sect. 3, from scratch on the ImageNet 2012 [26] and Places 365 [32] training sets, respectively. In all cases, the learning rate is set to 0.001. The weight decay which contributes reducing the training error of the deep network is set to 0.0005. The momentum rate which is associated with the gradient descent method employed to minimize the objective function is set to 0.9. In case of fine-tuning the pre-trained deep models, we employ training samples with a batch size of 80, a momentum value of 0.9 and an initial learning rate of 0.005. Furthermore, in all experiments the recognition results are reported as the mean classification accuracy over all scene categories in a scene recognition dataset. From the network prediction, the scene category label providing the highest confidence is assigned to the test image. The overall results are obtained by calculating the mean recognition score over all scene classes in each scene recognition dataset.

Datasets: MIT-67 [23] consists of 15,620 images of 67 indoor scene categories. We follow the standard protocol provided by the authors [23] by using 80 images per scene category for training and another 20 images for testing. **OCIS** [14] is the recently introduced large-scale object categories in indoor scenes dataset. It comprises of 15,324 images spanning more than 1300 commonly encountered indoor object categories. We follow the standard protocol provided by the authors [14] by defining a train-test split of (67% vs 33%) for each category. **SUN-397** [30] dataset consists of 108,754 images of 397 scene categories.

Here, the scene categories are both from indoor and outdoor environments. Each category in this dataset has at least 100 images. We follow the standard protocol provided by the authors [30] by dividing the dataset into 50 training and 50 test images per scene category. Since our aim is to investigate indoor scene recognition, we focus on the 177 indoor scene categories for the baseline comparison. Later, we show the results on the full SUN-397 dataset for state-of-the-art comparison.

Baseline Comparison: We compare our three-stream approach with the baseline standard RGB stream. Further, both early and late fusion strategies are evaluated for fusing the RGB, estimated depth and texture streams. For a fair comparison, we employ the same network architecture together with the same set of parameters for both the standard RGB and our multi-stream networks. Table 1 shows the baseline comparison with deep models trained on both ImageNet and Places datasets. We first discuss the results based on deep models pre-trained on the ImageNet. The baseline standard RGB deep network achieves average classification scores of 63.0%, 39.1%, and 46.0% on the MIT-67, OCIS, and SUN-397 datasets, respectively. The estimated depth based deep stream obtains mean recognition rates of 41.0%, 25.2%, and 26.0% on the MIT-67, OCIS and SUN-397 datasets, respectively. The texture coded deep image stream yields average classification accuracies of 59.1%, 33.6%, and 38.9% on the three scene datasets. In the case of the two fusion strategies, superior results are obtained with late fusion. The late fusion based two-stream network with RGB and depth streams obtains average classification scores of 67.1%, 40.9%, and 48.4% on the MIT-67, OCIS and SUN-397 datasets, respectively. Further, the late fusion based two-stream network with RGB and texture streams achieves average recognition rates of 69.3%, 42.5%, and 51.1% on the MIT-67, OCIS and SUN-397 datasets, respectively. The proposed late fusion based three-stream deep network significantly outperforms the baseline standard RGB deep stream on all datasets. Significant absolute gains of 9.3%, 4.7%, and 7.3% is achieved on the MIT-67, OCIS and SUN-397 datasets, respectively.

Other than the OCIS dataset, results are improved overall when employing deep models pre-trained on the Places scene dataset. The inferior recognition results in the case of the OCIS dataset are likely due to the fact that this dataset is based on indoor objects as categories instead of scenes. When comparing models trained on the Places dataset, our late fusion based three-stream deep architecture provides a substantial gains of 7.6%, 5.7%, and 4.9% on the MIT-67, OCIS and SUN-397 datasets respectively, compared to the baseline RGB stream.

We further analyze the impact of integrating depth and texture information within the deep learning architecture by looking into different indoor scene hierarchies available in the SUN-397 dataset. The indoor categories in the SUN-397 dataset are further annotated with the following scene hierarchies: shopping/dining with 40 indoor scene classes, workplace (office building, factory, lab, etc.) with 40 indoor scene classes, home/hotel with 35 indoor scene classes, transportation (vehicle interiors, stations, etc.) with 21 indoor scene classes, sports/leisure with 22 indoor scene classes, and cultural (art, education, religion,

Table 1. Comparison (overall accuracy in %) of our proposed three-stream deep architecture with the baseline standard RGB stream on the three scene datasets. We show multi-stream results with both early and late fusion schemes using deep networks either pre-trained on ImageNet or Places. Our proposed late-fusion based three-stream architecture significantly outperforms the baseline standard RGB stream on *all* datasets.

Architecture	Pre-training: imagenet			Pre-training: places		
	MIT-67	OCIS	SUN-397	MIT-67	OCIS	SUN-397
RGB deep stream (baseline)	63.0	39.1	46.0	73.6	32.5	58.6
Depth deep stream	41.0	25.2	26.0	51.5	21.4	34.6
Texture deep stream	59.1	33.6	38.9	68.7	27.2	49.3
Two-stream {RGB, depth} (early fusion)	65.2	39.5	46.7	74.3	32.8	59.3
Two-stream {RGB, depth} (late fusion)	67.1	40.9	48.4	76.5	34.1	60.5
Two-stream {RGB, texture} (early fusion)	65.7	39.9	47.9	75.3	33.3	59.7
Two-stream {RGB, texture} (late fusion)	69.3	42.5	51.1	78.8	36.5	61.8
Three-stream {RGB, depth, texture} (early fusion)	67.8	40.7	48.8	76.5	34.9	60.6
Three-stream {RGB, depth, texture} (late fusion)	**72.3**	**43.8**	**53.3**	**81.2**	**38.2**	**63.5**

Table 2. Comparison (overall accuracy in %) of our three-stream deep architecture with the baseline standard RGB stream on different indoor scene hierarchies available in SUN-397 dataset. The proposed three-stream deep architecture (late fusion) consistently improves the baseline standard RGB stream on all indoor scene hierarchies.

Architecture	Shopping/dining	Workplace	Home/hotel	Transportation	Sports/leisure	Cultural
RGB deep stream (baseline)	38.4	46.5	44.3	56.1	63.8	43.6
Ours {RGB, depth, texture} (late fusion)	**45.5**	**52.5**	**54.3**	**64.7**	**67.6**	**51.3**

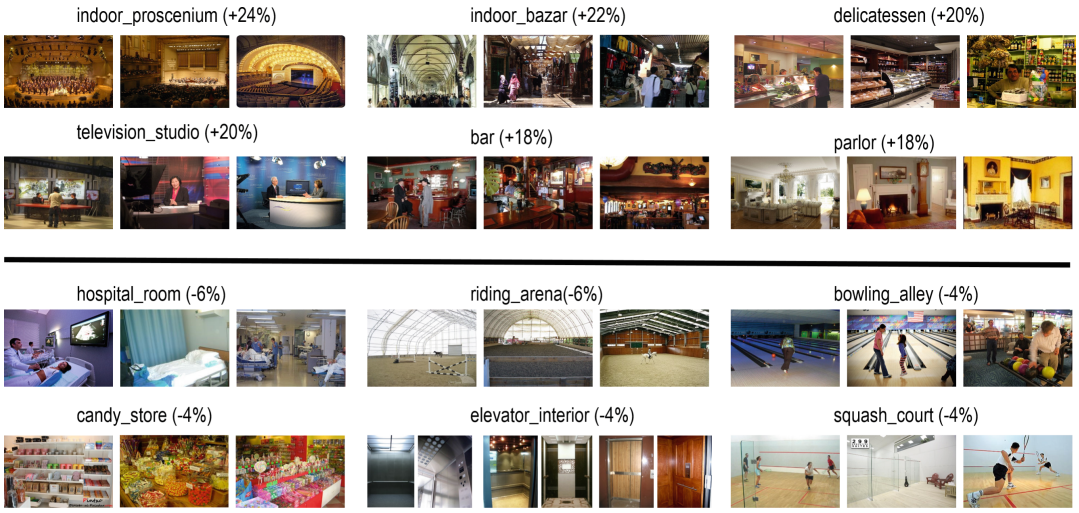


Fig. 4. Example images from SUN-397 indoor categories where our approach provides the biggest increase (top) and the biggest decrease (bottom), compared to the baseline.

Table 3. Comparison (overall accuracy in %) with the state-of-the-art approaches.

Method	Publication	MIT-67	OCIS	SUN-397
Multi-scale hybrid CNNs [10]	CVPR 2016	86.0	-	70.2
DRCF-CNN [14]	TIP 2016	71.8	32.0	-
SLSIF-CNN [8]	TIP 2016	74.4	-	-
PatchNets [29]	TIP 2017	84.9	-	**71.7**
LSHybrid-CNNs [6]	TIP 2017	83.8	-	67.6
Hybrid CNN models [31]	TCSVT 2017	86.0	-	70.7
Spectral-CNNs [15]	ICCV 2017	84.3	-	67.6
SCF-CNNs [13]	MVA 2018	83.1	-	-
This paper	-	**86.4**	**45.3**	69.2

military, law, politics, etc.) with 36 indoor scene classes. Note that some indoor scene categories are shared across different scene hierarchies. Table 2 shows the results obtained using the standard RGB and our three-stream network on the six scene hierarchies. Our approach provides significant gains of 7.1%, 6.0%, 10.0%, 3.8%, 7.5% and 7.3% on the six scene hierarchies (shopping/dining, Workplace, home/hotel, transportation, sports/leisure, and cultural), respectively. Figure 4 shows example images from different indoor scene categories from the SUN-397 dataset on which our three-stream architecture provides the biggest improvement (top) and the biggest drop (bottom), compared to the standard RGB network.

State-of-the-Art Comparison: State-of-the-art approaches employ very deep hybrid models pre-trained on both the ImageNet and Places datasets. Therefore, we also combine our late fusion based three-stream network, at the score/prediction level, with the very deep networks: ResNet-50 architecture [9]. Table 3 shows the comparison. Among existing methods, the works of [10,31] provide superior performance with a mean classification accuracy of 86.0% on the MIT-67 dataset. Our approach achieves improved results compared to both these methods with a mean recognition rate of 86.4%. On the OCIS dataset, our approach significantly outperforms the existing DRCF-CNN [14] by achieving a mean accuracy of 45.3%. On the SUN-397 dataset, the best results are obtained by PatchNets [29] approach. Our approach obtains an average classification accuracy of 69.2%.

5 Conclusions

We introduced a three-stream deep architecture for monocular indoor scene recognition. In addition to the standard RGB, we proposed to integrate explicit estimated depth and texture streams in the deep learning architecture. We further investigated different fusion strategies to integrate the three sources of information. To the best of our knowledge, we are the first to investigate fusion strate-

gies to integrate RGB, estimated depth and texture information for monocular indoor scene recognition.

Acknowledgement. This work has been supported by the Academy of Finland project number 313988 *Deep neural networks in scene graph generation for perception of visual multimedia semantics* and the European Union's Horizon 2020 research and innovation programme under grant agreement No 780069 *Methods for Managing Audiovisual Data: Combining Automatic Efficiency with Human Accuracy*. Computational resources have been provided by the Aalto Science-IT project and NVIDIA Corporation.

References

1. Anwer, R.M., Khan, F.S., van de Weijer, J., Molinier, M., Laaksonen, J.: Binary patterns encoded convolutional neural networks for texture recognition and remote sensing scene classification. ISPRS J. Photogramm. Remote Sens. **138**, 74–85 (2018)
2. Borg, I., Groenen, F.: Modern Multidimensional Scaling: Theory and Applications. Springer, New York (2005). https://doi.org/10.1007/0-387-28981-X
3. Chatfield, K., Simonyan, K., Vedaldi, A., Zisserman, A.: Return of the devil in the details: delving deep into convolutional nets. In: BMVC (2014)
4. Cimpoi, M., Maji, S., Vedaldi, A.: Deep filter banks for texture recognition and segmentation. In: CVPR, pp. 3828–3836 (2015)
5. Eigen, D., Fergus, R.: Predicting depth, surface normals and semantic labels with a common multi-scale convolutional architecture. In: ICCV (2015)
6. Guo, S., Huang, W., Wang, L., Qiao, Y.: Locally supervised deep hybrid model for scene recognition. TIP **26**(2), 808–820 (2017)
7. Gupta, S., Arbelaez, P., Girshick, R., Malik, J.: Local binary features for texture classification: taxonomy and experimental study. IJCV **112**(2), 133–149 (2015)
8. Hayat, M., Khan, S., Bennamoun, M., An, S.: A spatial layout and scale invariant feature representation for indoor scene classification. TIP **25**(10), 4829–4841 (2016)
9. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: CVPR (2016)
10. Herranz, L., Jiang, S., Li, X.: Scene recognition with CNNs: objects, scales and dataset bias. In: CVPR (2016)
11. Hoffman, J., Gupta, S., Darrell, T.: Learning with side information through modality hallucination. In: CVPR (2016)
12. Khan, F.S., Anwer, R.M., van de Weijer, J., Felsberg, M., Laaksonen, J.: Compact color-texture description for texture classification. PRL **51**, 16–22 (2015)
13. Khan, F.S., van de Weijer, J., Anwer, R.M., Bagdanov, A., Felsberg, M., Laaksonen, J.: Scale coding bag of deep features for human attribute and action recognition. MVA **29**(1), 25–71 (2018)
14. Khan, S., Hayat, M., Bennamoun, M., Togneri, R., Sohel, F.: A discriminative representation of convolutional features for indoor scene recognition. TIP **25**(7), 3372–3383 (2016)
15. Khan, S., Hayat, M., Porikli, F.: Scene categorization with spectral features. In: ICCV (2017)
16. Koskela, M., Laaksonen, J.: Convolutional network features for scene recognition. In: ACM MM (2014)

17. Lazebnik, S., Schmid, C., Ponce, J.: Beyond bags of features: spatial pyramid matching for recognizing natural scene categories. In: CVPR, pp. 2169–2178 (2006)
18. Levi, G., Hassner, T.: Emotion recognition in the wild via convolutional neural networks and mapped binary patterns. In: ICMI (2015)
19. Li, B., Shen, C., Dai, Y., van den Hengel, A., He, M.: Depth and surface normal estimation from monocular images using regression on deep features and hierarchical CRFs. In: CVPR (2015)
20. Liu, F., Shen, C., Lin, G., Reid, I.: Learning depth from single monocular images using deep convolutional neural fields. PAMI **38**(10), 2024–2039 (2016)
21. Liu, L., Fieguth, P., Guo, Y., Wang, X., Pietikainen, M.: Local binary features for texture classification: taxonomy and experimental study. PR **62**, 135–160 (2017)
22. Ojala, T., Pietikainen, M., Maenpaa, T.: Multiresolution gray-scale and rotation invariant texture classification with local binary patterns. PAMI **24**(7), 971–987 (2002)
23. Quattoni, A., Torralba, A.: Recognizing indoor scenes. In: CVPR (2009)
24. Renninger, L.W., Malik, J.: When is scene identification just texture recognition? Vis. Res. **44**(19), 2301–2311 (2004)
25. Rubner, Y., Tomasi, C., Guibas, L.: The earth mover's distance as a metric for image retrieval. IJCV **40**(2), 99–121 (2000)
26. Russakovsky, O., et al.: ImageNet large scale visual recognition challenge. arXiv preprint arXiv:1409.0575 (2014)
27. Song, X., Herranz, L., Jiang, S.: Depth CNNs for RGB-D scene recognition: learning from scratch better than transferring from RGB-CNNs. In: AAAI (2017)
28. Wang, A., Cai, J., Lu, J., Cham, T.J.: Modality and component aware feature fusion for RGB-D scene classification. In: CVPR (2016)
29. Wang, Z., Wang, L., Wang, Y., Zhang, B., Qiao, Y.: Weakly supervised patchnets: describing and aggregating local patches for scene recognition. TIP **26**(4), 2028–2041 (2017)
30. Xiao, J., Hays, J., Ehinger, K., Oliva, A., Torralba, A.: Sun database: large-scale scene recognition from abbey to zoo. In: CVPR (2010)
31. Xie, G.S., Zhang, X.Y., Yan, S., Liu, C.L.: Hybrid CNN and dictionary-based models for scene recognition and domain adaptation. TCSVT **27**(6), 1263–1274 (2016)
32. Zhou, B., Lapedriza, A., Xiao, J., Torralba, A., Oliva, A.: Learning deep features for scene recognition using places database. In: NIPS, pp. 487–495 (2014)

B.11 AALTO's paper in ICCV 2019 conference [10]

This paper describes the development of deep neural network based techniques for human-object interaction detection in which researchers at AALTO have participated. The results may later have influence on the work in MeMAD. It has been published in the ICCV 2019 conference.

Deep Contextual Attention for Human-Object Interaction Detection

Tiancai Wang[1]*†, Rao Muhammad Anwer[2]*, Muhammad Haris Khan[2], Fahad Shahbaz Khan[2],
Yanwei Pang[1], Ling Shao[2], Jorma Laaksonen[3]
[1]School of Electrical and Information Engineering, Tianjin University
[2]Inception Institute of Artificial Intelligence (IIAI), UAE
[3]Department of Computer Science, Aalto University School of Science, Finland
[1]{wangtc, pyw}@tju.edu.cn, [2]{rao.anwer, muhammad.haris, fahad.khan, ling.shao}@inceptioniai.org
[3]{jorma.laaksonen}@aalto.fi

Abstract

Human-object interaction detection is an important and relatively new class of visual relationship detection tasks, essential for deeper scene understanding. Most existing approaches decompose the problem into object localization and interaction recognition. Despite showing progress, these approaches only rely on the appearances of humans and objects and overlook the available context information, crucial for capturing subtle interactions between them. We propose a contextual attention framework for human-object interaction detection. Our approach leverages context by learning contextually-aware appearance features for human and object instances. The proposed attention module then adaptively selects relevant instance-centric context information to highlight image regions likely to contain human-object interactions. Experiments are performed on three benchmarks: V-COCO, HICO-DET and HCVRD. Our approach outperforms the state-of-the-art on all datasets. On the V-COCO dataset, our method achieves a relative gain of 4.4% in terms of role mean average precision (mAP$_{role}$), compared to the existing best approach.

1. Introduction

Recent years have witnessed tremendous progress in various instance-level recognition tasks, including object detection and segmentation. These instance-level problems have numerous applications in robotics, autonomous driving and surveillance. However, such applications demand a deeper knowledge of scene semantics beyond instance-level recognition, such as the inference of visual relationships between object pairs. Detecting human-object interactions (HOI) is a class of visual relationship detection. Given an image, the task is to not only localize a human and an object, but also recognize the interaction between them. Specifically, it boils down to detecting ⟨*human, action, object*⟩ triplets. The problem is challenging as it focuses on both human-centric interactions with fine-grained actions (*i.e.*, riding a horse vs. feeding a horse) and involves multiple co-occurring actions (*i.e.*, eating a donut and interacting with a computer while sitting on a chair).

Most existing HOI detection approaches typically tackle the problem by decomposing it into two parts: object localization and interaction recognition [1, 10, 11, 13, 20, 26]. In the first part, off-the-shelf two-stage object detectors [7, 22, 8] localize both human and object instances in an image. In the second part, detected human and object instances and the pairwise interaction between them are treated separately in a multi-stream network architecture. Recent works have attempted to improve HOI detection by integrating, *e.g.*, structural information [20], gaze and pose cues [26]. Despite these recent advances, the HOI detection performance is still far from satisfactory compared to other vision tasks, such as object detection and instance segmentation.

Current HOI detection approaches tend to focus on appearance features of human and object instances (bounding-boxes) that are central to scoring human-object interactions, and thereby identifying triplets. However, the readily available auxiliary information, such as context, at various levels of image granularity is overlooked. Context information is known to play a crucial role in improving the performance of several computer vision tasks [4, 27, 18, 2]. However, it is relatively underexplored for the high-level task of HOI detection, where context around each candidate detection is likely to provide complementary information to standard bounding-box appearance features. Global context provides valuable image-level information by determining the presence or absence of a specific object category. For instance, when detecting *driving a boat* interaction category, person, boat and water are likely to co-occur in an image. However for *drive a car* category, interaction (drive) remains the

*Equal contribution
†Work done at IIAI during Tiancai's internship.

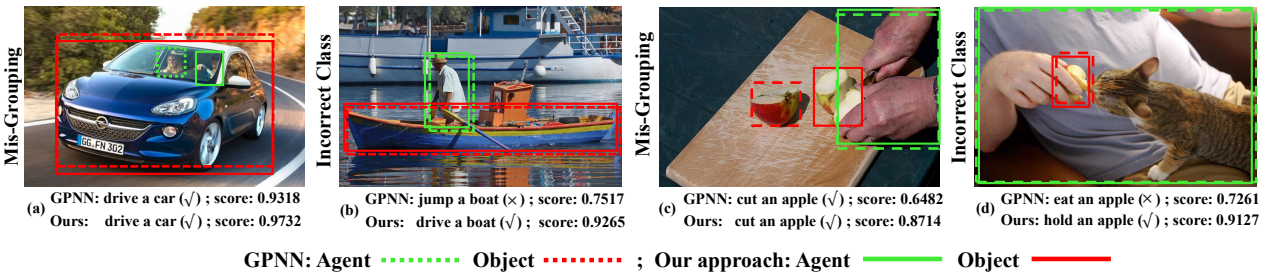


Figure 1. Example of HOI detections using the proposed approach and the recently introduced GPNN method [20]. The four examples depict two HOI detection cases. First in (a) and (b), different object categories (*car* and *boat*) involve the same human-object interaction (*drive*). Second in (c) and (d), different human-object interactions (*cut an apple* and *hold an apple*) involve the same object (*apple*). In case of (a) and (c), GPNN method fails to correctly pair the agent (person) and object, while it miss-classifies the action categories (b) and (d). Our approach accurately groups the agent and the respective object, while correctly classifying the action labels (scores) in all four cases.

same and only context (water) is changed. Besides global context, information in the immediate vicinity of each human/object instance provides additional cues to distinguish different interactions, *e.g.*, various interactions involving the same object. For instance, the surrounding neighborhood in *eating an apple* category is the face of the person whereas for *cutting an apple* category, it is knife and part of the hand (see Fig. 1). In this work, we leverage the context information to the relatively new problem of HOI detection.

Contributions: We first introduce a contextually enriched appearance representation for human and object instances. While providing auxiliary information, global context also introduces background noise which hampers interaction recognition performance. We therefore propose an attention module to suppress the background noise, while preserving the relevant contextual information. Our attention module is conditioned to specific instances of humans and objects to highlight the interaction regions, *i.e.*, *kick a sports ball* versus *throw a sports ball* categories. The resulting human/object attention maps are then used to modulate the global features to highlight image regions that are likely to contain a human-object interaction.

We validate our approach on three HOI detection benchmarks: V-COCO [11], HICO-DET [1] and HCVRD [32]. We perform a thorough ablation study to show the impact of context information for HOI detection. The results clearly demonstrate that the proposed approach provides a significant improvement over its non-contextual baseline counterpart. Further, our contextual attention-based HOI detection framework sets a new state-of-the-art on all datasets. On HICO-DET dataset, our approach yields a relative gain of 9.4% in terms of mean average precision (mAP), compared to the best published method [5]. Fig. 1 shows a comparison of our approach with GPNN [20] on HICO-DET images.

2. Related Work

Object Detection: Significant progress has been made in the field of object detection [7, 23, 22, 8, 29, 15, 21, 17], predominantly due to deep convolutional neural networks (CNNs). Generally, CNN-based object detectors can be divided into two-stage and single-stage approaches. In the two-stage approach, object detection methods [7, 22, 8] first employ an object proposal generator to generate regions of interests, which are then passed through an object classification and bounding-box regression pipeline. In contrast, single-stage detection methods [21, 17] directly learn object category predictions (classification) and bounding-box locations (regression) using anchors to predict the offsets of boxes instead of coordinates. Two-stage object detectors are generally more accurate compared to their single-stage counterparts. As in previous HOI detection works [10, 1], we employ an off-the-shelf two-stage FPN detector [15] to detect both human and object instances.

Human-Object Interaction Detection: Gupta and Malik [11] were the first to introduce the problem of visual semantic role labeling. In this problem, the aim is to detect a human, an object, and label the interaction between them. Gkioxrari *et al.*, [10] proposed a human-centric approach by extending the Faster R-CNN pipeline [22] with an additional branch to classify both actions and action-specific probability density estimation over the target object location. The work of [20] proposed a Graph Parsing Neural Network (GPNN) in which HOI structures are represented with graphs and then optimal graph structures are parsed in an end-to-end fashion. The work of [26] introduced a human intention-driven approach, where both pose and gaze information are exploited in a three-branch framework: object detection, human-object pairwise interaction and gaze-driven stream. Kolesnikov *et al.*, [13] proposed a joint probabilistic model for detecting visual relationships. Chao *et al.*, [1] introduced a human-object region-based CNN approach that extends the region-based object detector (Fast R-CNN) and has three streams: human, object and pairwise. Further, they introduced a new large-scale human-object interaction detection benchmark (HICO-DET).

Contextual Cues in Vision: Context provides an auxiliary cue for several vision problems, such as object detection

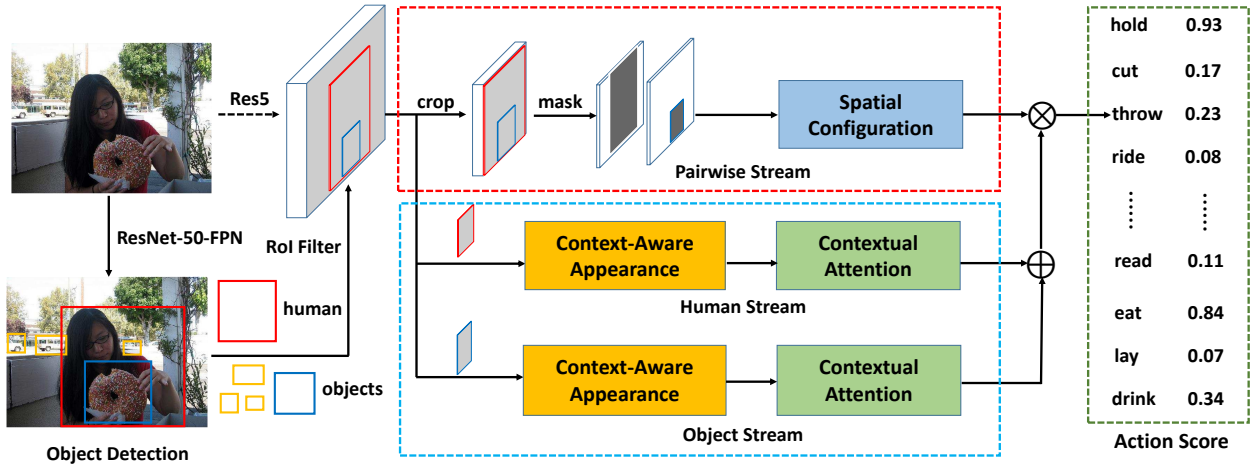


Figure 2. Overall multi-stream architecture of our proposed HOI detection framework comprising a localization and an interaction stage. For localization, we follow the standard object detector [15] to obtain human and object bounding-box predictions. For interaction prediction, we fuse scores from a human, an object, and a pairwise stream. We introduce context-aware appearance and contextual attention modules in the human and object streams. Final predictions are obtained by fusing the scores from human, object and pairwise streams.

[18, 2], action recognition [27], and semantic segmentation [4]. Recently, learnable context has gained popularity with the advent of deep neural networks [6, 18]. Despite its success in several tasks [19, 6, 18, 31, 14, 28], the impact of contextual information to the relatively new task of HOI detection is yet to be fully explored.

3. Overall Framework

The overall framework comprises two stages: localization and interaction prediction (see Fig. 2). For localization, we follow the popular paradigm of FPN [15] as a standard object detector to generate bounding-boxes for all possible human and object instances in the input image. For interaction prediction, following [1], we fuse scores from the three individual streams: a human, an object, and a pairwise. Scores from human and object streams are added. The resulting scores are then multiplied with pairwise stream.

Multi-Stream Pipeline: The inputs to the multi-stream architecture are the bounding-box predictions from FPN [15] and the original image. The output of the multi-stream architecture is a detected $\langle$*human, action, object*$\rangle$ triplet. The overall framework comprises three separate streams: human, object and pairwise interaction. Both the human and object streams are appearance oriented; they employ CNN feature extraction to generate confidence scores on the detected human and object bounding-boxes. The pairwise interaction stream encodes the spatial relationship between the person and object as in [1].

3.1. Proposed Human/Object Stream

The standard multi-stream architecture encodes instance-centric (bounding-box) appearance features in the human and object streams and ignores the associated contextual information. In this work, we argue that the bounding-box appearance alone is insufficient and that the contextual information in the vicinity of a human and object instances provides complementary information useful to distinguish complex human-object interactions. We therefore enrich the human and object streams (see Fig. 3) with contextual information by introducing contextually-aware appearance features f_{app} (sec. 3.1.1). These contextual appearance features f_{app} are then fed into the contextual attention module (sec. 3.1.2), where they are used to modulate the global feature map A to obtain a modulated feature representation F_m. The modulated feature representation F_m is further refined in the attention refinement block to obtain the refined modulated features F_r, which further passes through global average pooling to obtain refined modulated vector f_r. Subsequently, both representations f_{app} and f_r are concatenated to obtain action predictions from the human/object streams. Note that the same architecture is employed for both the human and object streams. Thus, the only difference between the two streams is their inputs, which are human and object bounding-box predictions, respectively. Next, we describe different components of our proposed human/object stream.

3.1.1 Contextually-Aware Appearance Features

Given the CNN features (Res5 block of the ResNet-50 backbone) of the whole image, as well as human/object bounding-box predictions from the detector, standard instance-centric appearance features are extracted by employing region-of-interest (ROI) pooling followed by a residual block and global average pooling. Though theoretically the image-level CNN features used in the construction

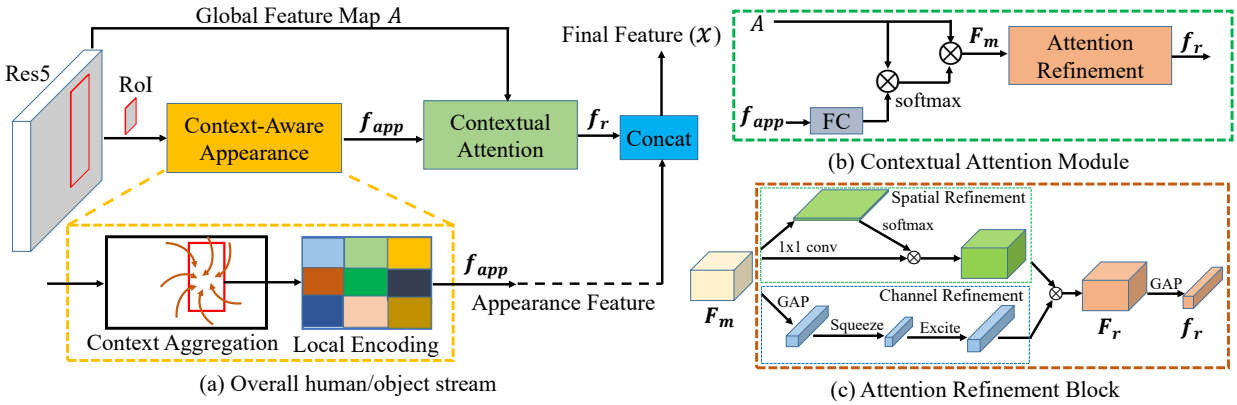


Figure 3. On the left (a), the proposed overall human/object stream. Both the contextual attention module (b) and the attention refinement block (c) are shown on the right. The context-aware appearance module produces contextual appearance features that encode both appearance and context information. The contextual appearance features are then fed into the contextual attention module to suppress the background noise resulting in a modulated feature representation. The modulated feature representation is further enriched in the attention refinement block to obtain refined modulated features. Consequently, both contextual appearance and refined modulated features are concatenated to obtain action predictions from the human/object stream.

of the standard appearance representation are supposed to cover entire spatial image extent, their valid receptive field is much smaller in practice [30]. This implies that the larger global scene context prior is ignored in such a standard appearance feature construction. Our context-aware appearance module is designed to capture additional context information and consists of context aggregation and local encoding blocks (see Fig. 3(a)).

The context aggregation block aims to capture a larger field-of-view (FOV) to integrate context information in instance-centric appearance features, while preserving spatial information. A straightforward way to capture a larger FOV is through a fully connected (FC) layer or cascaded dilated convolutions. However, the former collapses spatial dimensions, while the latter produces sparser features. Therefore, our context aggregation block employs a large convolutional kernel (LK) previously used for semantic segmentation [19]. To the best of our knowledge, we are the first to introduce a large kernel-based context aggregation block to construct contextual appearance features for the problem of HOI detection. The input to the context aggregation block is the CNN features (Res5 block) of the image with size $h \times w \times c_{in}$, where c_{in} denotes the number of channels and h and w denote the input feature dimensions. The output of the context aggregation block is then context-enriched features of size $h \times w \times c_{out}$, obtained after applying a large kernel of size $k \times k$ to the original CNN features. In this work, we utilize the factorized large kernel, which is efficient as its computational complexity and number of parameters are only $O(2/k)$, compared to the trivial $k \times k$ convolution.

Beside context aggregation, our context-aware appearance module contains a local encoding block. Existing HOI detection approaches employ standard ROI warping, which involves a max-pooling operation performed on the cropped ROI region. Our local encoding block aims to preserve locality-sensitive information in each bounding-box ROI region by encoding the position information with respect to a relative spatial position. Such a strategy has been previously investigated to encode spatial information within ROI regions in the context of generic object detection [3]. However, [3] directly employs a 1×1 convolution on the standard CNN feature map (Res5). Instead, we encode locality-sensitive information in each ROI region based on the contextualized CNN feature map obtained from our context aggregation block. Further, [3] utilizes PSRoIpooling with average pooling. Instead, we employ the PSRoIAlign together with max-pooling. PSRoIAlign is employed to reduce the impact of coarse quantization caused by PSRoIpooling through bilinear interpolation. Fig. 4 shows the impact of PSRoIAlign-based local encoding on the input feature maps of an image. Consequently, the output of the local encoding block is flattened and passed through a fully-connected layer to obtain conextual appearance features f_{app}.

3.1.2 Contextual Attention

The contextual appearance features, described above, encode both appearance and global context information. However, not all background information is equally useful for the HOI problem. Further, integrating meaningless background noise can even deteriorate the HOI detection performance. Therefore, a careful identification of useful contextual information is desired to distinguish subtle human-object interactions that are difficult to handle otherwise. Generally, attention mechanisms are used to highlight the discriminative features particularly important for a given

task [25]. The contextual attention module in our human/object stream consists of bottom-up attention and attention refinement components. The bottom-up attention component is based on the recently introduced approach of [6] for action recognition and exploits a scene-level prior to focus on relevant features. Note, [6] computes image-level attention, whereas we aim to generate bounding-box based attention. Further, contrary to standard appearance features, the bottom-up attention maps in our attention module are generated using *contextually-aware appearance* features f_{app} (sec. 3.1.1) that encode both appearance and context. We generate modulated features by first constructing a contextual attention map, which is then deployed to modulate the input CNN feature map (see Fig. 3(b)).

Specifically, we project the input (Res5) feature maps f using a 1×1 convolution onto a 512-dimensional space, denoted as A. Then, we compute the dot product between these projected global features A and contextual-appearance features f_{app} to obtain an attention map, which is then used to modulate A, such that,

$$F_m = \text{softmax}(f_{app} \otimes A) \otimes A \tag{1}$$

Here, F_m are the resulting modulated features. The discriminative ability of F_m is further enhanced in the attention refinement block, which consists of spatial and channel-wise attention refinement. The attention refinement block is simple and light-weight (see Fig. 3(c)). During spatial refinement, we first apply a 1×1 conv on modulated features F_m to generate a single-channel heatmap H, followed by a softmax-operation-based normalization. Then, we perform an element-wise multiplication between the normalized heatmap and the modulated features F_m. The resulting spatial refinement S_{att} learns the most relevant features as:

$$S_{att}(F_m) = H \otimes F_m \tag{2}$$

Beside spatial refinement, we also perform a channel-wise refinement. Inspired by the squeeze-and-excitation network (SENet) of [12], we first apply global average pooling on the modulated features F_m to squeeze global spatial information into a channel descriptor z. Then, the excitation stage is a stack of two FC layers, followed by a sigmoid activation with input z and is described as:

$$C_{att}(F_m) = \sigma(W_1\delta(W_2 z)) \tag{3}$$

Here, z is the output of the squeeze operation, and W_1 and W_2 refer to fully-connected operations. δ and σ are ReLU and sigmoid activations, respectively. Finally, C_{att} modulates the spatially-attended features S_{att} to further highlight regions relevant to human-object interaction to obtain a refined modulated feature representation F_r as:

$$F_r = S_{att}(F_m) \otimes C_{att}(F_m) \tag{4}$$

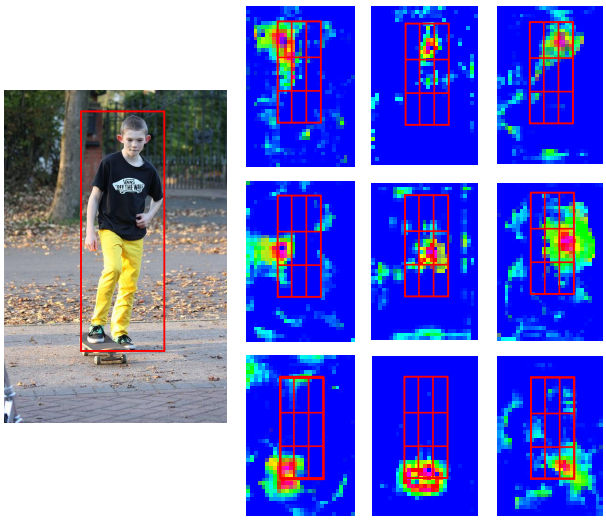


Figure 4. Visual depiction of the local encoding block that preserves locality-sensitive information. For illustration purposes, the detected human bounding-box is divided into 3×3 sub-regions and there are 9 score maps. Each sub-region votes for the presence of a specific object part, relative to the position of the object, based on how good the bounding-box overlaps with the score maps.

Finally, the refined modulated features F_r are passed through global average pooling to obtain the refined modulated vector f_r. We combine contextual appearance features f_{app} and the refined modulated vector f_r to produce the final representation x. This representation x is then passed through two FC layers to estimate action predictions from the human/object stream, respectively. Given an HOI predicted bounding-box, the final prediction is obtained by fusing the scores from the human, object and pairwise streams.

4. Experiments

4.1. Dataset and Evaluation Protocol

V-COCO [11]: is the first HOI detection benchmark and a subset of popular MS-COCO dataset [16]. The V-COCO dataset contains 10,346 images in total, with 16,199 human instances. Each human instance is annotated with 26 binary action labels. Note that three action classes (*i.e.*, cut, hit, eat) are annotated with two types of targets (*i.e.*, instrument and direct object). It includes 2533, 2867, and 4946 images for training, validation and testing, respectively.

HICO-DET [1]: is a challenging dataset and has 47,776 images in total, with 38,118 images for training and 9658 images for testing. There are more than 150k human instances annotated with 600 types of different human-object interactions. The HICO-DET dataset contains same 80 object categories as MS-COCO and 117 action verbs.

HCVRD [32]: is a large-scale dataset and is labeled with both human-centric visual relationships and corresponding human and object bounding boxes. It has 52,855 images

Add-on	Baseline		
Res5-share	✓	✓	✓
Context-aware appearance (sec. 3.1.1)		✓	✓
Contextual attention (sec. 3.1.2)			✓
mAP_{role}	44.5	46.0	**47.3**

Table 1. A baseline comparison when integrating our proposed context-aware appearance and contextual attention modules into the multi-stream architecture. Results are reported in terms of role mean average precision (mAP_{role}) on the V-COCO dataset. For fair comparison, we use the same feature backbone (Res 5 block of ResNet-50) for both our approach and the baseline. Both context-aware appearance and contextual attention modules contribute in the overall improvement in HOI detection performance. Our overall architecture achieves a relative gain of 6.3% over the baseline.

with 1,824 object categories and 927 predicates. It contains 256,550 relationships instances and there are on average 10.63 predicates per object category. We evaluate our method on the predicate detection task, where the goal is to perform predicate recognition given the labels and bounding boxes for both object and human.

Evaluation Protocol: We use the original evaluation protocols for all three datasets, as provided by their respective authors. For the V-COCO dataset, we use role mean Average Precision (mAP_{role}) as an evaluation metric. Here, the aim is to detect the $\langle$*human, action, object*$\rangle$ triplet. The HOI detection is considered correct if the intersection-over-union (IoU) between the human and object bounding-box predictions and the respective ground-truth boxes is greater than the threshold 0.5 together with the correct action label prediction. For HICO-DET, results are reported in terms of mean average precision (mAP). For HCVRD, we report top-1 and top-3 results at 50 and 100 recall.

4.2. Implementation Details

We deploy Detectron [9] with a ResNet-50-FPN [15] backbone to obtain human and object bounding-box predictions. To select a predicted bounding-box as a training sample, we set the confidence threshold to be higher than 0.8 for humans and 0.4 for objects. For interaction prediction, we employ ResNet-50 as the feature extraction backbone pre-trained on ImageNet. The initial learning rate is set to 0.001, weight decay of 0.0001 and a momentum of 0.9 is used for all datasets. The network is trained for 300k on V-COCO and 1800k iterations on HICO-DET and HCVRD, respectively. For input image of size 480×640, our interaction recognition part of the approach takes 130 milliseconds (ms) to process, compared to its baseline counterpart (111ms) on a Titan X GPU.

4.3. Results on V-COCO Dataset

Baseline Comparison: We first evaluate the impact of integrating our proposed context-aware appearance (sec. 3.1.1) and contextual attention (sec. 3.1.2) modules into the human/object stream of the multi-stream architecture. Tab. 1 shows the results on the V-COCO dataset. The baseline multi-stream architecture contains standard appearance features from the *Res5* block of the ResNet-50 backbone, which have a size of $h \times w \times 2048$. These standard appearance features are directly passed through the classifier to obtain the final action scores in the human/object stream, achieving a mAP_{role} of 44.5. The introduction of the proposed contextual appearance features improves the HOI detection performance from 44.5 to 46.0 in terms of mAP_{role}. The performance is further improved by 1.3%, in terms of mAP_{role} when integrating our proposed contextual attention module. Our final framework achieves an absolute gain of 2.8% in terms of mAP_{role}, compared to the baseline.

We further evaluate the impact of contextual information on improving the classification capabilities of the multi-stream architecture. This is done by selecting different IoU thresholds in the range [0.1-0.9] used in the test evaluation of interaction recognition performance. Tab. 2 shows the results on the V-COCO dataset. At a high threshold value (0.9), few ground-truth bounding-boxes are matched, whereas at a low threshold (0.1) most them are matched. Therefore, comparison at lower thresholds mainly focuses

Overlap thresh	0.1	0.3	0.5	0.7	0.9
Baseline	50.1	47.8	44.5	35.9	2.5
Our Approach	**53.5**	**50.8**	**47.3**	**37.0**	**2.8**

Table 2. Performance (in terms of mAP_{role}) with different IoU thresholds, used in the testing, to compare the classification capabilities of our approach with the baseline on the V-COCO dataset. The performance gap between our approach and the baseline increases at lower threshold values.

Backbone Architecture	Baseline	Our Approach
VGG-16	42.0	**44.5**
ResNet-50	44.5	**47.3**
ResNet-101	45.0	**47.8**

Table 3. A comparison (in terms of mAP_{role}) of our approach with the baseline when using different backbone network architectures on the V-COCO dataset. Our approach always provides consistent improvements over the baseline using different backbones.

Methods	Feature Backbone	**mAP**$_{role}$
Gupta et .al[11]*	ResNet-50-FPN	31.8
InteractNet [10]	ResNet-50-FPN	40.0
BAR [13]	Inception-ResNet	41.1
GPNN [20]	ResNet-50	44.0
iCAN [5]	ResNet-50	45.3
Our Approach	ResNet-50	**47.3**

Table 4. State-of-the-art comparison on the V-COCO dataset. * refers to implementation of the approach of [11] by [10]. The scores are reported in mAP_{role} and the best result is in bold. Our approach sets a new state-of-the-art on this dataset, achieving an absolute gain of 2.0% over the best existing method.

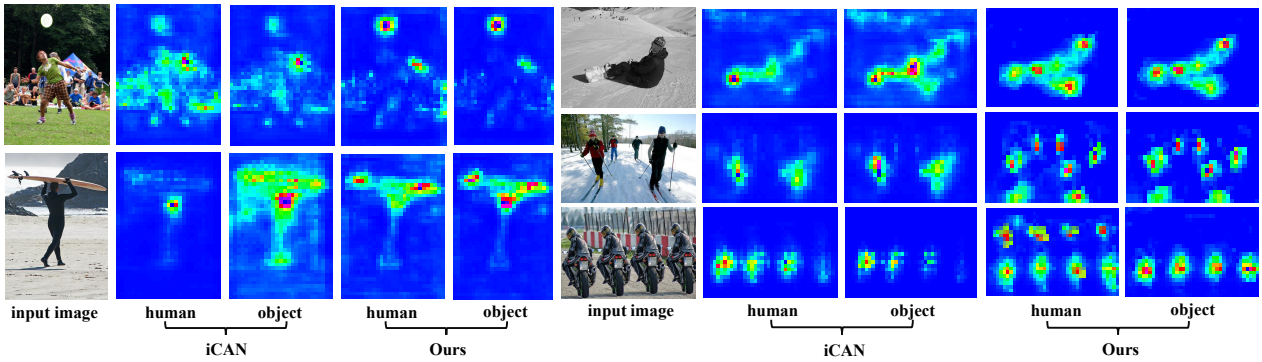


Figure 5. Comparison of attention maps obtained using our approach and iCAN [5] on example images from the V-COCO dataset. Human and object attention maps in iCAN are constructed using standard appearance features. In contrast, human and object attention maps in our approach are constructed using contextual appearance features extracted using the context aggregation and local encoding blocks in our context-aware appearance module. We show examples for both single and multiple human-object interactions.

Figure 6. Example detection results on V-COCO dataset. Each example can involve a single human- object interaction such as *skateboarding* and *eat donut* or multiple humans sharing the same interaction and object - *hold* and *eat pizza*, *throw* and *catch ball*.

on the classification capabilities of our approach. Tab. 2 shows that our approach is superior in terms of classification capabilities, compared to the baseline.

Tab. 3 shows the generalization capabilities of our approach with respect to different network architectures. We perform experiments using VGG-16, ResNet-50 and ResNet-101, each pre-trained on the ImageNet dataset, as the underlying network architectures. In all cases, our approach provides consistent improvements over the baseline.

Comparison with State-of-the-art: In Tab. 4, we compare our approach with state-of-the-art methods in the literature on the V-COCO dataset. Among existing works, InteractNet [10] jointly learns to detect humans, objects and their interactions achieving a mAP$_{role}$ of 40.0. The GPNN approach [20] integrates structural information in a graph neural network architecture and provides a mAP$_{role}$ of 44.0. The iCAN approach [5] combines human, object and their pairwise interaction streams in an early fusion manner using the standard appearance features and bottom-up attention strategy. Our approach sets a new state-of-the-art on this dataset by achieving a mAP$_{role}$ of 47.3.

Qualitative Comparison: Fig. 5 shows comparison between the attention maps obtained using our approach and iCAN [5] on example images from the V-COCO dataset. Note that the attention maps in iCAN [5] are constructed using standard appearance features. In contrast, the attention maps in our approach are constructed using contextual appearance features generated using the context aggrega-

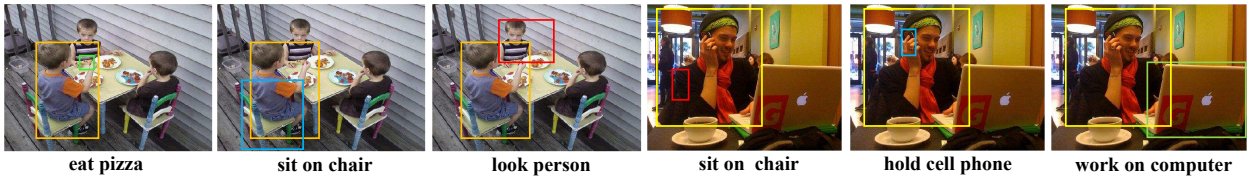


Figure 7. Multiple interaction detection on V-COCO. Our approach detects human instance doing multiple (different) actions and interacting with various objects (represented with different colors). In all cases, the detected agent is represented with the same color.

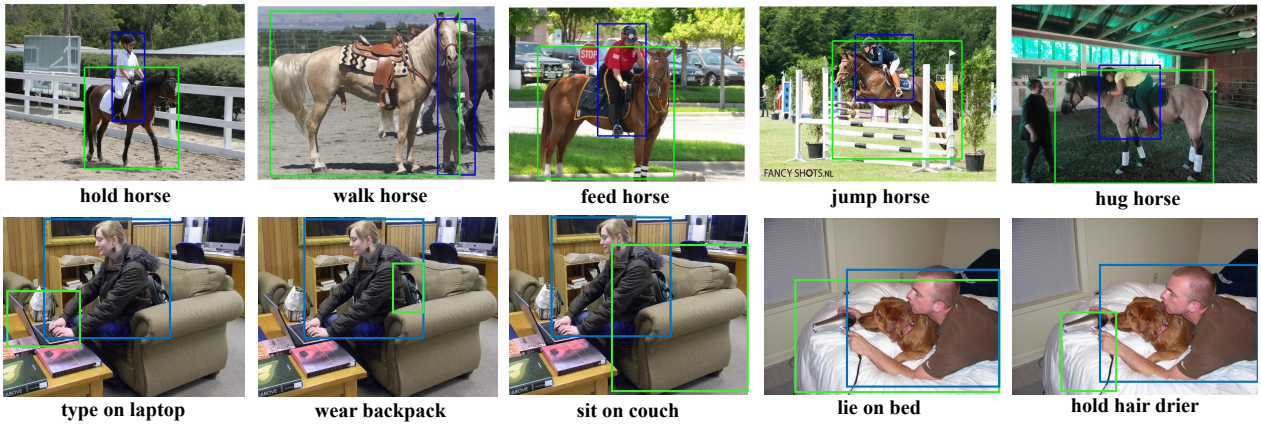


Figure 8. Results on HICO-DET showing one detected triplet. Blue boxes represent a detected human instance, while the green boxes show the detected object of interaction. Our approach detects various fine-grained interactions (top row) and multiple interactions (second row).

tion and local encoding blocks in our context-aware appearance module. Our attention maps focus on relevant regions in the human and object branches that are likely to contain human-object interactions (*e.g.*, in case of *throwing frisbee* and *riding bike*). In addition, for both single and multiple human-object interactions, our approach produces more anchored attention maps compared to the iCAN method.

Fig. 6 shows examples showing both single human-object interactions such as *skateboarding* and *eat a donut*, and multiple humans sharing same interaction and object – *holding* and *eating pizza*, *throw* and *catch ball*. Fig. 7 shows examples of a human performing multiple interactions.

4.4. Results on HICO-DET and HCVRD datasets

On HICO-DET we report results on three different HOI category sets: full, rare, and non-rare with two different settings of Default and Known Objects [1]. Our approach outperforms the state-of-the-art in all three category sets under both Default and Known Object settings (see Tab. 5. The relative gain of 9.4%, 6.7%, and 9.8% is obtained over the best existing method on all three sets in Default settings. Fig. 8 shows results on HICO-DET. On HCVRD dataset, iCAN achieves top-1 and top-3 accuracies at R@50 of 33.8 and 48.9, respectively. Our approach outperforms iCAN with top-1 and top-3 accuracies at R@50 of 37.1 and 51.3, respectively. Similarly, our approach provides superior results at R@100 (top-3 accuracy of iCAN: 49.4 vs. top-3 accuracy of ours: 51.9).

Methods	Default			Known Object		
	full	rare	non-rare	full	rare	non-rare
Shen *et al.*, [24]	6.46	4.24	7.12	-	-	-
Chao *et al.*, [1]	7.81	5.37	8.54	10.41	8.94	10.85
InteractNet [10]	9.94	7.16	10.77	-	-	-
GPNN [20]	13.11	9.34	14.23	-	-	-
iCAN [5]	14.84	10.45	16.15	16.43	12.01	17.75
Ours	**16.24**	**11.16**	**17.75**	**17.73**	**12.78**	**19.21**

Table 5. State-of-the-art comparison on the HICO-DET using two different settings: Default and Known Object on all three sets (full, rare, non-rare). Note that Shen *et al.* [24], InteractNet [10] and GPNN [20] only report results on the Default settings. Our approach achieves a relative gain of 9.4%, 6.7%, and 9.9% over the best existing method on all three HOI sets in Default settings.

5. Conclusion

We propose a deep contextual attention framework for HOI detection. Our approach learns contextually-aware appearance features for human and object instances. To suppress the background noise, our attention module adaptively selects relevant instance-centric context information crucial for capturing human-object interactions. Experiments are performed on three HOI detection benchmarks: V-COCO, HICO-DET and HCVRD. Our approach has been shown to outperform state-of-the-art methods on all datasets.

Acknowledgments: This work was supported by the National Natural Science Foundation of China (Grant # 61632018), Academy of Finland project number 313988 and the European Unions' Horizon 2020 (Grant # 780069).

References

[1] Yuwei Chao, Yunfan Liu, Xieyang Liu, Huayi Zeng, and Jia Deng. Learning to detect human-object interactions. In *WACV*, 2018. 1, 2, 3, 5, 8

[2] Xinlei Chen and Abhinav Gupta. Spatial memory for context reasoning in object detection. In *CVPR*, 2017. 1, 3

[3] Jifeng Dai, Yi Li, Kaiming He, and Jian Sun. R-FCN: Object detection via region-based fully convolutional networks. In *NIPS*, 2016. 4

[4] Henghui Ding, Xudong Jiang, Bing Shuai, Ai Qun Liu, and Gang Wang. Context contrasted feature and gated multi-scale aggregation for scene segmentation. In *CVPR*, 2018. 1, 3

[5] Chen Gao, Yuliang Zou, and Jia-Bin Huang. iCAN: Instance-centric attention network for human-object interaction detection. In *BMVC*, 2018. 2, 6, 7, 8

[6] Rohit Girdhar and Deva Ramanan. Attentional pooling for action recognition. In *NIPS*, 2017. 3, 5

[7] Ross Girshick. Fast R-CNN. In *ICCV*, 2015. 1, 2

[8] Ross Girshick, Jeff Donahue, Trevor Darrell, and Jitendra Malik. Rich feature hierarchies for accurate object detection and semantic segmentation. In *CVPR*, 2014. 1, 2

[9] Ross Girshick, Ilija Radosavovic, Georgia Gkioxari, Piotr Dollár, and Kaiming He. Detectron. https://github.com/facebookresearch/detectron, 2018. 6

[10] Georgia Gkioxari, Ross Girshick, Piotr Dollár, and Kaiming He. Detecting and recognizing human-object intaractions. In *CVPR*, 2018. 1, 2, 6, 7, 8

[11] Saurabh Gupta and Jitendra Malik. Visual semantic role labeling. *arXiv preprint arXiv:1505.04474*, 2015. 1, 2, 5, 6

[12] Jie Hu, Li Shen, and Gang Sun. Squeeze-and-Excitation networks. *arXiv preprint arXiv:1709.01507*, 2017. 5

[13] Alexander Kolesnikov, Christoph H. Lampert, and Vittorio Ferrari. Detecting visual relationships using box attention. In *arXiv preprint arXiv:1807.02136*, 2018. 1, 2, 6

[14] Jianan Li, Yunchao Wei, Xiaodan Liang, Jian Dong, Tingfa Xu, Jiashi Feng, and Shuicheng Yan. Attentive contexts for object detection. *TMM*, 2017. 3

[15] Tsung-Yi Lin, Piotr Dollar, Ross Girshick, Kaiming He, Bharath Hariharan, and Serge Belongie. Feature pyramid networks for object detection. In *CVPR*, 2017. 2, 3, 6

[16] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollr, and C. Lawrence Zitnick. Microsoft coco: Common objects in context. In *ECCV*, 2014. 5

[17] Wei Liu, Dragomir Anguelov, Dumitru Erhan, Christian Szegedy, Scott Reed, Cheng-Yang Fu, and Alexander C. Berg. SSD: Single shot multibox detector. In *ECCV*, 2016. 2

[18] Yong Liu, Ruiping Wang, Shiguang Shan, and Xilin Chen. Structure inference net: Object detection using scene-level context and instance-level relationships. In *CVPR*, 2018. 1, 3

[19] Chao Peng, Xiangyu Zhang, Gang Yu, Guiming Luo, and Jian Sun. Large kernel mattersimprove semantic segmentation by global convolutional network. In *CVPR*, 2017. 3, 4

[20] Siyuan Qi, Wenguan Wang, Baoxiong Jia, Jianbing Shen, and Song-Chun Zhu. Learning human-object interactions by graph parsing neural networks. In *ECCV*, 2018. 1, 2, 6, 7, 8

[21] Joseph Redmon, Santosh Divvala, Ross Girshick, and Ali Farhadi. Look once: Unified, real-time object detection. In *CVPR*, 2016. 2

[22] Shaoqing Ren, Kaiming He, Ross Girshick, and Jian Sun. Faster R-CNN: Towards real-time object detection with region proposal networks. In *NIPS*, 2015. 1, 2

[23] Fahad Shahbaz Khan, Jiaolong Xu, Joost van de Weijer, Andrew Bagdanov, Rao Muhammad Anwer, and Antonio Lopez. Recognizing actions through action-specific person detection. *IEEE Transactions on Image Processing*, 24(11):4422–4432, 2015. 2

[24] Liyue Shen, Serena Yeung, Judy Hoffman, Greg Mori, and Li Fei-Fei. Scaling human-object interaction recognition through zero-shot learning. In *WACV*, 2018. 8

[25] John Tsotsos, Sean Culhane, Winky Yan, Kei Wai, Yuzhong Lai, Neal Davis, and Fernando Nuflo. Modeling visual attention via selective tuning. *Artificial Intelligence*, 1995. 5

[26] Bingjie Xu, Junnan Li, Yongkang Wong, Mohan S. Kankanhalli, and Qi Zhao. Interact as you intend: Intention-driven human-object interaction detection. *arXiv preprint arXiv:1808.09796*, 2018. 1, 2

[27] Bangpeng Yao and Li Fei-Fei. Modeling mutual context of object and human pose in human-object interaction activities. In *CVPR*, 2010. 1, 3

[28] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas Huang. Generative image inpainting with contextual attention. In *CVPR*, 2018. 3

[29] Shifeng Zhang, Longyin Wen, Xiao Bian, Zhen Lei, and Stan Z. Li. Single-shot refinement neural network for object detection. In *CVPR*, 2018. 2

[30] Bolei Zhou, Aditya Khosla, Agata Lapedriza, Aude Oliva, and Antonio Torralba. Object detectors emerge in deep scene cnns. *arXiv preprint arXiv:1412.6856*, 2014. 4

[31] Bohan Zhuang, Lingqiao Liu, Chunhua Shen, and Ian Reid. Towards context-aware interaction recognition for visual relationship detection. In *ICCV*, 2017. 3

[32] Bohan Zhuang, Qi Wu, Chunhua Shen, Ian Reid, and Anton van den Hengel. Hcvrd: a benchmark for large-scale human-centered visual relationship detection. In *AAAI*, 2018. 2, 5