Multimodal person recognition in audio-visual streams

Thèse N° 9442

Présentée le 6 septembre 2019

à la Faculté des sciences et techniques de l'ingénieur
Laboratoire de l'IDIAP
Programme doctoral en génie électrique

pour l'obtention du grade de Docteur ès Sciences

par

Do Hoang Nam LE

Acceptée sur proposition du jury

Prof. P. Frossard, président du jury
Dr J.-M. Odobez, directeur de thèse
Prof. S. Meignier, rapporteur
Dr D. Dai, rapporteur
Dr M. Salzmann, rapporteur

2019

ÉCOLE POLYTECHNIQUE
FÉDÉRALE DE LAUSANNE

Acknowledgements

Working towards this dissertation has been the most challenging yet fascinating period of my life so far. Looking back at the end of the journey, this thesis would not have been possible without the support of many people, to whom I am extremely grateful.

First and foremost, to my advisor Jean-Marc for offering me the opportunity to research interesting topics in Idiap and EPFL. Through his supervision and feedback, I have grown not only as a researcher but also as a person.

To my thesis committee, Prof. Pascal Frossard, Prof. Sylvain Meignier, Dr. Mathieu Salzmann, and Dr. Dengxin Dai, for their constructive feedback and interest in my research. This dissertation has been improved significantly thanks to them.

To the secretariats and the system team for making research at Idiap stress free. Sylvie and Nadine have always been available to answer my questions and to help me fill in all the forms in French that I still have no idea what they mean.

To the Perception group for our fruitful collaborations. Especially, to Vasil and Alex for their support in building the foundation of my research. To Gulcan, Kenneth, and Rui for the feedback after every group meeting.

To all Idiapers for making Idiap an awesome place. To Cijo, James, Subhadeep, and Angelos for the insightful and stimulating discussions. To my great friend Andreas for turning me from a nerd into a mountain enthusiast.

To the Shopping Ads team in Google, Zurich. While my internship was not directly related to this thesis, thank you for improving my programming and project managing skills.

And lastly, to my dear parents.

Martigny, March 2019 Nam

Abstract

Multimedia databases are growing rapidly in size in the digital age. To increase the value of these data and to enhance the user experience, there is a need to make these videos searchable through automatic indexing. Because people appearing and talking in the videos are often of high interest for end users, indices that represent the location and identity of people in the archive are indispensable for video search and browsing tools. On the other hand, multimedia videos contain resourceful data of people in both visual and auditory domains. This offers a potential for multimodal learning in the task of human identification. Hence, the main theme of this thesis is on algorithms to create indexes and exploit the audio-visual correspondence in large multimedia corpuses based on person identities.

First, this thesis deals with algorithms to create indexes through person discovery in videos. It involves several components: face and speaker diarization, face-voice association, and person naming. To obtain face clusters, we propose a novel face tracking approach that leverages face detectors with a tracking-by-detection framework relying on long term time-interval sensitive association costs. We use also shot context to further accelerate and improve face clustering. Face clusters are then associated to speaker clusters using dubbing and talking detection, in which a multimodal framework is introduced to represent the temporal relationship between the auditory and visual streams. We also improve speaker embeddings for recognition and clustering by using a regularizer called intra-class loss.

In the second half, the thesis focuses on multimodal learning with face-voice data. Here, we aim to answer two research questions. First, can one improve a voice embedding using knowledge transferred from a face representation? We investigate several transfer learning approaches to constrain the target voice embedding space to share latent attributes with the source face embedding space. The crossmodal constrains act as regularizers helping voice models, especially in the low-data setting. The second question is can face clusters be used as training labels to learn a speaker embedding? To answer this, we explore the tolerance of embedding losses under label uncertainty. From the risk minimization perspective, we obtain the analytical results that provide the heuristics in strategies to improve the tolerance against label noise. We apply the findings into our task of learning speaker embeddings using face clusters as labels. While the experimental results agree with the analytical heuristics, there is still a large gap in performance between the supervised and the weakly supervised models, which requires further investigation in the future.

Keywords: tracking, face clustering, speaker diarization, embedding learning, transfer learning

Résumé

A l'ère du digital, les bases de données vidéos multimédia prolifèrent. Afin d'augmenter la valeur de ces données et d'améliorer l'expérience des consommateurs de ces vidéos, il est nécessaire de faciliter leur accès en rendant possible la recherche et la navigation dans ces vidéos grâce à une meilleure indexation automatique. Comme les personnes sont souvent d'un grand intérêt pour l'utilisateur, l'indexation des moments où elles figurent ou s'expriment ainsi que leur identité se révèle très pertinente dans ce but. Comme ces vidéos contiennent en elle même les informations utiles tant visuelles qu'auditives pour effectuer cette tâche, elles offrent un potentiel d'exploration de méthodes d'apprentissage multimodal pour la reconnaissance et l'identification automatiques de personnes dans de grand corpus multimédia. Cette thèse s'inscrit dans cette perspective et aborde deux problématiques principales.

Tout d'abord, le design d'algorithmes d'indexation basés sur la reconnaissance des personnes. Cette tâche repose sur plusieurs composantes : extraction des segments temporels comprenant le même visage ou le même locuteur ; association entre les visages et les voix ; nommage des personnes. Afin d'obtenir des groupements de visage pertinents, nous proposons une nouvelle approche de suivi de visage qui tire parti de détecteurs de visage performants (mais coûteux) couplés à une méthode de suivi par détection reposant sur des coûts d'association long-terme sensibles aux intervalles de temps. Nous exploitons également le contexte temporel de la segmentation vidéo en shots pour accélérer et améliorer le groupement des visages. Ces derniers sont ensuite associés aux groupes de locuteurs en tenant compte d'un module de détection de conversation visuel et de détection du doublage audio reposant sur une modélisation multimodale de la relation temporelle entre les flux auditifs et visuels. Nous proposons également une méthode qui améliore un module d'extraction de représentation de la voix à base de réseau de neurones utile pour la reconnaissance et le regroupement, et qui repose sur un terme de régularisation minimisant les variabilité intra-class.

La deuxième moitié de la thèse porte sur l'apprentissage multimodal exploitant des données voix-visage. Nous voulons répondre à deux questions de recherche. Premièrement, peut-on améliorer un module d'extraction d'un descripteur de voix à l'aide de connaissances transférées des représentations de visage ? Nous étudions plusieurs approches d'apprentissage par transfert pour contraindre l'espace de représentation de la voix à partager des attributs latents avec l'espace de représentation du visage. Les contraintes multimodales agissent comme des termes de régularisation aidant la création d'un espace de représentation des voix, en particulier dans le cas où une faible quantité de données audio est disponible. La seconde question posée est : le

regroupements de visages peut-il être utilisé pour produire des étiquettes d'entraînement utiles à l'apprentissage d'une représentation du locuteur ? Pour répondre à cette question, nous explorons la robustesse de la représentation vis à vis des erreurs d'étiquetage des données. Nous étudions la minimisation des risques d'erreurs et présentons des résultats analytiques qui fournissent des méthodes heuristiques dans les stratégies visant à améliorer la robustesse au bruit des étiquettes d'entraînement. Nous appliquons ces résultats à notre tâche d'apprentissage de représentation d'un locuteur en utilisant des groupements de visages comme étiquettes d'entraînement. Bien que les résultats expérimentaux soient en accord avec les heuristiques analytiques, il existe encore un écart de performance important entre les modèles supervisés et les modèles faiblement supervisés, ce qui appelle à des investigations supplémentaires dans le futur.

Mots clés : suivi, groupement de visages, regroupement en locuteur, apprentissage de représentation, apprentissage par transfert de connaissance.

Contents

List of Figures

List of Tables

1 Introduction

1.1 Event Understanding through Multimodal Social Stream Interpretation - EUMSSI project

Nowadays, a large amount of multimedia data like news, debates, talkshows, documentaries or series is being produced and broadcast through multiple TV or internet channels. From the perspective of a producer of content, a multimedia journalist has to monitor, gather, curate and contextualize the relevant information for the target audience. To research a topic, he needs to go through an enormous amount of records with information of very diverse degrees of granularity. On the other side, many TV viewers are also receiving a large amount of information through the media. It is harder and harder to put information into context to understand stories and to reduce the noise of irrelevant content. Both the journalist and the TV viewer would greatly benefit from a system capable of automatically analyzing unstructured multimedia data stream and its social background, and contextualizing the data or contributing related information.

The EUMSSI consortium has been created with the main objective of developing methodologies and technologies for identifying and aggregating data presented as unstructured information in sources of very different nature (video, image, audio, speech, text and social context).The core idea is that the process of integrating content from different media sources is carried out in an interactive manner, so that the data resulting from one media helps reinforce the aggregation of information from other media. Once all the available descriptive information has been collected, an interpretation component will reason over the semantic representation in order to derive knowledge following an event-centered structure. This will be accomplished thanks to the integration in a multimodal platform of information extraction and analysis techniques from the different fields involved (image, audio and text analysis).

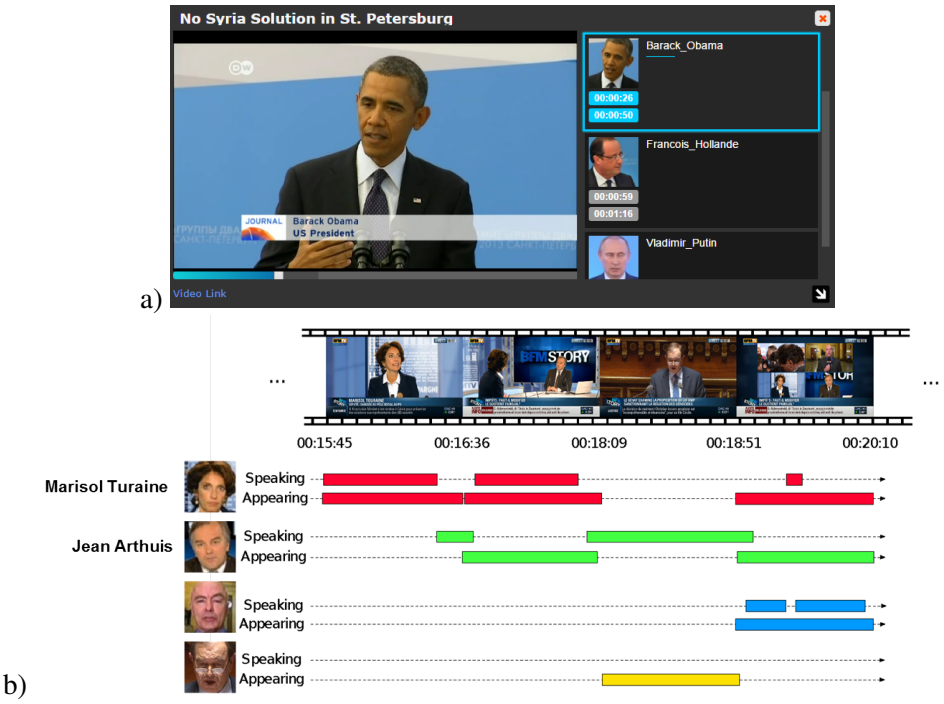


Figure 1.1 – Two different applications of human identification in video documents. a) Indices of when people appear in video. b) Visualization of how people interact in a TV report [1]

1.2 Thesis goals and motivations

In the context of the EUMSSI project, we are interested in identifying people in TV news to be exploited as semantic indices for efficient retrieval of multimedia content. For example, timestamps of when people appear can be used to search for related contents in TV news as shown in figure 1.1-a or viewers can browse interface such as figure 1.1-b to discover interesting contents of TV news based on how people interact with each other. This practical need leads to our research problem of how to identify people presence in videos or to answer "who appears when?" and "who speaks when?", or how to index videos through person discovery. Therefore, research efforts have been devoted to unsupervised segmentation of videos into homogeneous segments according to person identity, like speaker diarization, face diarization, and audio-visual (AV) person diarization. Combined with names extracted from overlaid text, AV person diarization makes it possible to identify people in videos.

Given the diversity and amount of multimedia documents, there are 2 main challenges: robustness and computation cost. Firstly, because of the wide range of media content, people can appear in widely different situations. Secondly, video corpuses, such as the corpus provided by EUMSSI partner Deutsche Welle, contain thousands of hours of videos. We need to optimize the computation cost for person diarization, which is the bottleneck for the whole system. Furthermore, it is an open question of how to utilize multimodal features to perform AV person diarization. Hence, the first part of the thesis focuses on improving each individual component as well as addressing

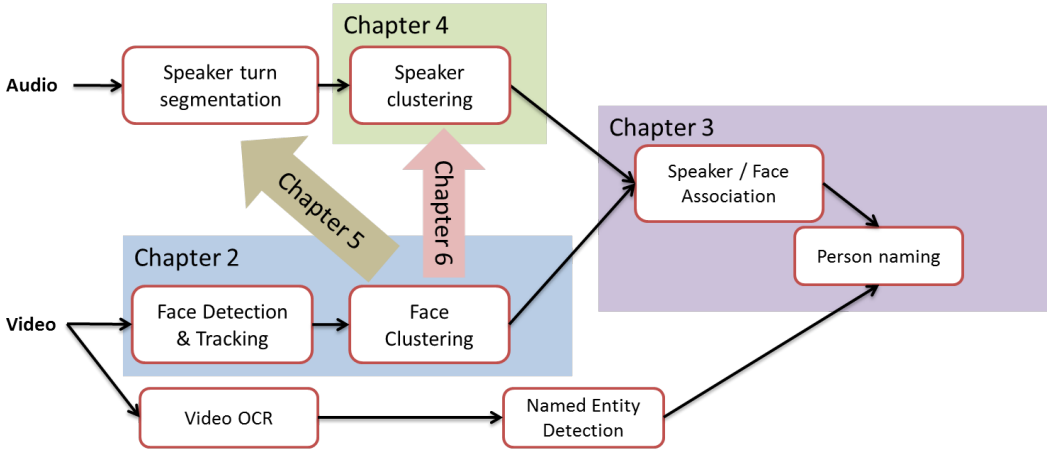


Figure 1.2 – Outlines of the thesis. Each highlighted box corresponds to one main contribution and its chapter. Chapter 2, 3, and 4 deal with improving and integrating components of a audio-visual person discovery system. Chapter 5 and 6 focuses on using models on the face domain to improve that of the auditory domain.

the audio-visual association problem in the integrated person discovery system.

Besides the immediate goal in indexing videos, person diarization results in a large scaled database of associated face-voice segments. Hence, it leads to research questions in how to exploit such correspondence between the facial domain and the auditory domain. Can we use one domain to improve the recognition models in another domain? Or can we use the result of one domain, such as face clustering, to automatically collect training data to use in another domain, i.e. speaker recognition? These research questions lead to the second part of the thesis in crossmodal transfer learning and weakly supervised learning.

1.3 Contributions

The main focus of this thesis is on algorithms to create and exploit indexes in large multimedia corpuses based on person identities. As indicated above, the first part the thesis was devoted to the design of a person discovery system. Hence, the first set of contributions are focused on the components of a person discovery system: (1) face tracking and clustering, (2) audio-visual streams association and person naming, and (3) learning speaker embeddings for clustering and recognition.

The second part involves multimodal learning with face-voice data. In this context, the contributions revolved around 2 main themes:(1) crossmodal transfer learning from facial domain to improve speaker turn embeddings, and (2) analysis of learning with unreliable labels and its application in speaker embedding learning using face clusters as labels. Figure 1.2 visually places how the contributions and their corresponding chapters fit into the whole context of the thesis. More details of each contribution are presented below.

1.3.1 Face diarization

Face diarization, i.e. face tracking and clustering within video documents, is useful and important for video indexing and fast browsing but it is also a difficult and time consuming task. To address this task, this thesis presents two main contributions. First, we propose a novel tracking approach that leverages deformable part-based model (DPM) face detector with a multi-cue discriminant tracking-by-detection framework that can automatically learn long-term time-interval sensitive association costs for each document type. The method is able to skip frames, i.e. process only 3 to 4 frames per second - thus cutting down computational cost - while performing better than state-of-the-art methods as evaluated on public benchmarks. Second, a shot constrained face clustering method is proposed, which significantly reduces the processing time while keeping or improving performance. We further show that complementing biometric i-vector representation with matching similarity measure improves performance.

1.3.2 Dubbing and talking face detection and person naming

Person discovery in the absence of prior identity knowledge requires accurate association of visual and auditory cues. In broadcast data, multimodal analysis faces additional challenges due to narrated voices over muted scenes or dubbing in different languages. To address these challenges, we define and analyze the problem of dubbing detection in broadcast data, which has not been explored before. We propose a method to represent the temporal relationship between the auditory and visual streams. This method consists of canonical correlation analysis to learn a joint multimodal space, and long short term memory (LSTM) networks to model cross-modality temporal dependencies. Our contributions also include the introduction of a newly acquired dataset of face-speech segments from TV data, which we have made publicly available. The proposed method achieves promising performance on this real world dataset as compared to several baselines.

Another contribution of this thesis is the integrated system for person naming in videos. Besides the individual improvement in face diarization and multimodal association, we have developed the toolchain and system to process large scaled audio-visual databases, in particular for the EUMSSI demonstration. To benchmark our contributions, we participated in the MediaEval Person Discovery challenge in 2015 and 2016. Our improved systems achieved the first place in both evaluation campaigns.

1.3.3 Improving speaker embeddings with intra-class loss

Learning a good speaker embedding is critical for many speech processing tasks, including recognition, verification, and diarization. To this end, we propose a complementary optimizing goal called intra-class loss to improve deep speaker embeddings learned with triplet loss. This loss function is formulated as a soft constraint on the averaged pair-wise distance between samples from the same class. Its goal is to prevent the scattering of these samples within the embedding

space to increase the intra-class compactness.When intra-class loss is jointly optimized with triplet loss, we can observe 2 major improvements: the deep embedding network can achieve a more robust and discriminative representation and the training process is more stable with a faster convergence rate. We conduct experiments on 2 large public benchmarking datasets for speaker verification, VoxCeleb and VoxForge. The results show that intra-class loss helps accelerating the convergence of deep network training and significantly improves the overall performance of the resulting embeddings.

1.3.4 Transfer learning from facial domain to improve speaker turn embeddings

Learning a discriminative voice embedding allows speaker turns to be compared directly and efficiently, which is crucial for tasks such as diarization and verification. This thesis investigates several transfer learning approaches to improve a voice embedding using knowledge transferred from a face representation. The main idea of our crossmodal approaches is to constrain the target voice embedding space to share latent attributes with the source face embedding space. The shared latent attributes can be formalized as geometric properties or distribution characteristics between these embedding spaces. We propose four transfer learning approaches belonging to two categories: the first category relies on the structure of the source face embedding space to regularize at different granularities the speaker turn embedding space. The second category -a domain adaptation approach- improves the embedding space of speaker turns by applying a maximum mean discrepancy loss to minimize the disparity between the distributions of the embedded features. Experiments are conducted on TV news datasets, REPERE and ETAPE, to demonstrate our methods. Quantitative results in verification and clustering tasks show promising improvement, especially in cases where speaker turns are short or the training data size is limited. The analysis also gives insights on the embedding spaces and shows their potential applications.

1.3.5 Weakly supervised learning with triplet loss

Collecting labeled data to train deep neural networks is costly and even impractical for many tasks. Thus, research effort has been focused in automatically curated datasets or unsupervised and weakly supervised learning. Given the results we obtained through person diarization, there is a possibility to use face clustering as weak labels to create large scaled database for training speaker recognition models, especially with embedding losses such as triplet loss.

This is a very practical problem as face clustering can automatically provide a massive amount of speech training data instead of the costly annotation process. The main problem in this direction is learning with unreliable label information. In this thesis, we first address the tolerance of deep embedding learning losses against label noise, i.e. when the observed labels are different from the true labels. Specifically, we conduct an analysis on the triplet loss, which shows the bound on the expected risk when optimizing triplet loss under noise. From the analytical results, we provide practical heuristics on sampling strategies and noise rate can affect the level of resistance against

label noise. These heuristics help providing more effective insights in unsupervised and weakly supervised deep embedding learning. We apply the guidelines to our task of learning speaker embeddings using face clusters as labels and the result validates our conjectures. However, the models learned with these weak labels still exhibits an accuracy gap in comparison to supervised models learned with clean data. This suggests potential to further improve weakly supervised speaker embedding learning.

1.4 Dissertation Outline

The body of this thesis is organized into 5 main chapters. Each chapter is self contained and can be read independently. The detailed outline is as follows:

- In Chapter 2, the face tracking and clustering framework, which significantly reduces the computation time while achieving state-of-the-art results, is presented. This chapter is based on 2 papers:
  - N. Le, A. Heili, D. Wu, and J.-M. Odobez, "Temporally subsampled detection for accurate and efficient face tracking and diarization," in *International Conference on Pattern Recognition*, IEEE, Dec. 2016
  - N. Le, A. Heili, and J.-M. Odobez, "Long-term time-sensitive costs for CRF-based tracking by detection," in *European Conference on Computer Vision Workshops*, pp. 43–51, Springer, 2016
- In Chapter 3, we introduce the dubbing and talking face detection problem and discuss an AV asynchrony detection algorithm. Then we describe the integrated person naming system used in the EUMSSI projects and related evaluation campaigns. This chapter is constructed with elements from the following papers:
  - N. Le and J.-M. Odobez, "Learning multimodal temporal representation for dubbing detection in broadcast media," in *ACM Multimedia*, ACM, Oct. 2016
  - N. Le, D. Wu, S. Meignier, and J.-M. Odobez, "EUMSSI team at the mediaeval person discovery challenge," in *MediaEval 2015 Workshop*, 2015
  - N. Le, S. Meignier, and J.-M. Odobez, "EUMSSI team at the mediaeval person discovery challenge 2016," in *MediaEval Benchmarking Initiative for Multimedia Evaluation*, 2016
  - N. Le, H. Bredin, G. Sargent, P. Lopez-Otero, C. Barras, C. Guinaudeau, G. Gravier, G. B. da Fonseca, I. L. Freire, Z. Patrocínio Jr, *et al.*, "Towards large scale multimedia indexing: A case study on person discovery in broadcast news," in *Proceedings of the 15th International Workshop on Content-Based Multimedia Indexing*, p. 18, ACM, 2017
- In Chapter 4, we propose a regularizer called intra-class loss that aims to improve models that use triplet loss for speaker recognition. This work was previously published in:

  - N. Le and J.-M. Odobez, "Robust and discriminative speaker embedding via intra-class distance variance regularization," *Proc. Interspeech 2018*, pp. 2257–2261, 2018

- In Chapter 5, we investigate 4 methods for transfer learning from a facial domain to a speech domain, concretely in the task of speaker turn embedding. Each method exploits the structure of the embedding space at different granularities. This chapter is based on the following papers:
  - N. Le and J.-M. Odobez, "A domain adaptation approach to improve speaker turn embedding using face representation," in *Proceedings of the 19th ACM International Conference on Multimodal Interaction*, pp. 411–415, ACM, 2017
  - N. Le and J.-M. Odobez, "Improving speaker turn embedding by crossmodal transfer learning from face embedding," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pp. 428–437, 2017
  - N. Le and J.-M. Odobez, "Improving speech embedding using crossmodal transfer learning with audio-visual data," *Multimedia Tools and Applications*, pp. 1–24, 2018

- In Chapter 6, the focus is on how to use face clustering results as pseudo-labels to collect training data to learn speaker recognition models with triplet loss. To this end, we first study the theoretical guarantees of learning with triplet loss under label uncertainty. Then, a prototype framework to learn speaker models with supervision data collected from the facial domain is introduced. This chapter is partially based on our paper:
  - N. Le and J.-M. Odobez, "Theoretical guarantees of deep embedding losses under label noise," *arXiv preprint arXiv:1812.02676*, 2018

2 Face Diarization

2.1 Introduction

In this chapter, we address face diarization and develop a method applicable to TV broadcast media in general, like news, debates, documentaries. Given a video segmented into shots, a typical diarization system proceeds in two main steps: extracting face tracks within a shot, which is important as it has been shown that using tracks leads to better face representation than individual images [18], and clustering all face tracks with the same identity. These steps are illustrated in Figure 2.1. Due to the wide range of media content and amount of videos, this has two main challenges that we investigate in this paper for the two above tasks: robustness and computational cost.

To obtain face tracks, typical diarization systems [19, 20] rely on frontal face detectors like the Viola-Jones (VJ) detector [21] due to availability and speed. Then, for tracking, KLT interest point trackers are often used to link detections and associate them over time [20, 18, 22, 23, 24]. Nevertheless, given the diversity of image backgrounds and faces that can appear in challenging illumination and poses, the detector may miss detections and produce a large amount of false ones. To counter this lack of robustness, systems usually only use the frontal face detector (thus missing a large amount of near profile faces), apply it at every frame to obtain better detection statistics, and complement it with forward/backward tracking or complex per track skin filtering procedure [19] to remove false alarms. Although there are much better detectors nowadays, they are usually not used due to their expensive running time in normal hardware.

We propose a novel tracking approach that takes advantage of the powerful multi-view DPM detector[1] within a fast tracking method to benefit from the detector accuracy without the expense of running time. More precisely, we use a fast version [25] of the DPM detector. For tracking, we rely on and extend the human tracking-by-detection framework of [2] to the multi-face tracking domain, resulting in a tracker that exploits time-interval sensitive discriminative multi-cue appearance and motion association costs learned in an unsupervised way, allowing an easy

[1] State-of-the-art in face detection as of 2016

Figure 2.1 – Given a video segmented into shots, we address two main steps in a diarization system: (1) face detection and tracking within a shot to produce face tracks, and (2) face clustering across shots to create face clusters corresponding to identities.

adaptation to each media document type. In particular, since long term connectivity between detections is exploited, to the contrary of most frame-to-frame methods, our approach delivers competitive results while only having to process 3 to 4 frames per second.

Face track clustering is also a time-consuming step. Indeed, traditional unsupervised bottom-up hierarchical approaches [19, 26] are at least quadratic in the number of face tracks to cluster, and thus quickly become time consuming when the number of track is large, as in news or talk shows.

We thus propose a divide-and-conquer strategy that follows the observation that people's faces often appear in similar shots. By prioritizing the merging of faces more likely to have the same identity, we both significantly cut down the computational cost, while allowing to build more robust face representation from multiple face tracks for further clustering. To this end, as representation of group of face tracks, we investigate the use of i-vectors, a biometric approach based on total variability modeling which has been shown to handle well appearance variations [27, 28]. However, in spite of the state-of-the-art performance reported by this model on several comprehensive benchmarks [28], we show that for face diarization on TV datasets, combining it with a keypoints-based representation, which is robust for measuring the similarity of faces acquired in similar conditions, is still important to improve the performance.

To sum-up, our main contributions are as follows:

- a novel multi-face tracking approach relying on multi-cue and long-term time-sensitive association costs;
- a shot-constrained face track clustering approach reducing computation while improving performance;
- an empirical evaluation showing that for face diarization, state-of-the-art biometric models can unexpectedly be advantageously combined with a key-point based similarity measure.

Extensive experiments on public datasets demonstrate the benefit of each individual module as well as the performance of the whole system.

The next section reviews existing works complementary to ours. Sections 2.3 and 2.4 describe

the tracking and clustering methodologies. Section 2.5 presents the conducted experiments to support our propositions. Finally, Section 2.6 concludes the chapter with further discussion and future works.

2.2 Related Work

Face diarization is a collective process with 3 different components: detection, tracking, and clustering. Below we comment on the related works in the context of each task.

Face detection. This is the bottleneck that decides the performance of the whole system. Missing detections can be caused by profile faces, lighting conditions, or intrinsic variability of faces. Meanwhile, background with detailed textures can be easily mistaken for real faces, which creates noise for tracking as well as clustering. Such problems not only diminish the utility of the system but can also annoy practical users. Most diarization systems rely on the standard VJ detector [21] which has shown to have low accuracy on competitive datasets such as PASCAL faces or FDDB [29]. To improve this, 2 strategies are proposed: aggregating multiple detectors to increase the recall rate [22] and filtering with upper body detectors to increase the precision rate [23]. Both strategies slow down the system significantly. In another direction, deep neural networks has achieved high accuracy on face detection problem. However, these networks also require sophisticated infrastructures and are not fast enough for practical usage [30]. Therefore in our work, we rely on the DPM detector [31], which is highly competitive and easy to integrate without side effects [29].

Face tracking. From the face detection result, face tracking aims to create a set of continuous faces. Kanade-Lucas-Tomasi feature tracker (KLT) [32] is commonly applied due to its speed and simplicity [22, 23]. KLT tracker is also used to repopulate detections missed by the detector. However, this tracker is sensitive to long occlusion and drifting over time. On the other hand, tracks can also be obtained by associating detections. In [33], tracklets are formed based on time, motion, and color information and then linked by optimizing a graphical model. Meanwhile in [24], faces are first associated by location, size, and pose; then tracklets are connected by discriminative face appearance models. All the aforementioned systems require the detector to be applied every frame to create face tracks reliably. On the contrary, our model has a major advantage that the parameters are estimated unsupervised for long term connectivity. This allows speeding up by applying the detector sparsely.

Face clustering. This final task aims to associate face tracks with the same identity to the same cluster. This problem is closely related to building face representation for face recognition. In a small dataset with limited people, one can use pre-trained recognition models [34, 23]. However, when dealing with a large dataset with unknown identities, we are more interested in unsupervised approaches. In [23], the Fisher face descriptor [18] is extended to represent face clusters with pose specific comparison. Meanwhile in [19], a person is represented by a Gaussian Mixture Model (GMM) and a set of matching keypoints. To calculate distances, they

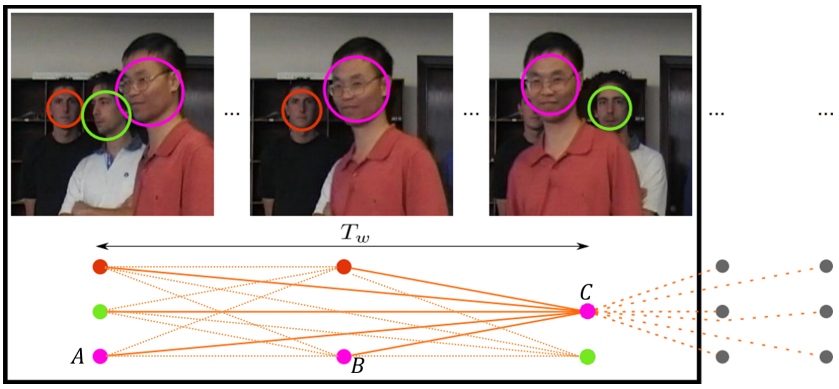


Figure 2.2 – Tracking as graph clustering task. The detections form the nodes, and a long-term connectivity is used, i.e. all links between pairs of nodes within a temporal window T_w are used to define the cost function. Long-term connectivity combined with time-interval sensitive discriminative pairwise models and visual motion enables dealing with missed detections, e.g. due to occlusion, as well as skipped frames.

used cross-likelihood ratio which is time consuming and the adapted GMMs can be biased to specific recording conditions. Therefore, although we follow the idea of combining the matching similarity and biometric similarity in [19], our face descriptor is built based on total variability modeling (TVM). TVM has been shown to be successful for face verification in uncontrolled conditions such as illumination, pose, or expression [28, 27]. Furthermore, applying TVM allows us to directly compare the face representations, or i-vectors, which is more efficiently than other biometric models using cross-likelihood ratio [28].

Besides intrinsic properties of faces, video structure is also a valuable indicator for clustering. In TV news with fixed camera angles, camera classification can help identifying people in regular viewpoints [35]. However, this approach is applicable to specific shows and requires heavy annotations. In [23], threading similar shots helps reducing clustering cost and collecting negative pairs for their exemplar SVM. As we favor our system to be unsupervised and independent of data sources, shot constrained face clustering is utilized to group similar shots without prior knowledge to speed up the face clustering process.

2.3 Face detection and tracking

The first stage of face diarization comprises face detection and tracking, which includes a false alarm track removal step.

2.3.1 Face detection

We employ the multi-view DPM model, which achieved state-of-the-art results [31, 29]. However, due to the numerous convolutions required, a main disadvantage of DPM is its computational cost, which can take up to 3s/frame for HD videos. Thus, we use a sped-up variant leveraging Fourier transforms to accelerate the processing [25]. Furthermore, as shown in the experiments, thanks to the increased accuracy w.r.t. the VJ detector, we only need to apply the face detector 3 to 4 times per second, which considerably decreases the computational cost for detection.

2.3.2 Face tracking overview

We propose to leverage and extend the multi-human tracking method proposed in [36, 2], by adding new features, handling sparse detections over time (Section 2.3.4), and adding a false track removal step (Section 2.3.5).

Our approach is illustrated in Figure 2.2. Face tracking is formulated as a labeling problem within a Conditional Random Field (CRF) framework. Given the set of face detections $Y = \{y_i\}_{i=1:N_y}$, where N_y is the total number of detections, we search for the set of corresponding labels $L = \{l_i\}_{i=1:N_y}$ such that faces belonging to the same identity are assigned the same label. This is done by optimizing the posterior probability $p(L|Y, \lambda)$, where λ denotes the set of model parameters. Under some assumption, this is equivalent to minimizing the following energy potential:

$$U(L) = \left(\sum_{(i,j)\in\mathcal{V}} \sum_{r=1}^{N_s} w_{ij}^r \beta_{ij}^r \delta(l_i - l_j) \right), \tag{2.1}$$

with the coefficients defined as:

$$\beta_{ij}^r = \log \left[\frac{p(S_r(y_i, y_j)|H_0, \lambda_{\Delta_{ij}}^r)}{p(S_r(y_i, y_j)|H_1, \lambda_{\Delta_{ij}}^r)} \right]. \tag{2.2}$$

with the different terms defined as follows. First, the energy involves N_s feature functions $S_r(y_i, y_j)$ measuring a similarity between detection pairs as well as confidence weights w_{ij}^r for each detection pair. Importantly, note that a *long-term connectivity* is exploited, in which the set of valid pairs $\mathcal{V}$ contains *all pairs* whose temporal distance $\Delta_{ij} = |t_j - t_i|$ is lower than T_w, where T_w is usually between 1 and 2 seconds. This contrasts with most frame-to-frame tracking or path optimization approaches. For instance, in Fig. 2.2, even if there is a path from A to B and B to C for the same track, the link A to C is also exploited in the cost function, resulting in better conditioned objective function.

Secondly, the Potts coefficients themselves are defined as the likelihood ratio of the probability of feature distances under two hypotheses: H_0 if $l_i \neq l_j$ (i.e. detections do not belong to the same face), or H_1 when labels are the same. In practice, this allows us to *incorporate discrimination*,

Figure 2.3 – Position. The different iso-contours of value 0 of the Potts costs for different values of Δ (i.e. location of detections occurring after Δ frames around each shown detection and for which $\beta = 0$), learned in an unsupervised fashion from TV REPERE (left) and Hannah (right). In the region delimited by a curve, association will be favored, whereas outside it will be disfavored. Curves show that more motion is expected on the Hannah movie, than on the TV data.

by quantifying how much features are similar and dissimilar under the two hypotheses, and not only on how much they are similar for the same identity as done in traditional path optimization of many graph-based tracking methods. Furthermore, note that as these costs depend on the set of parameters $\lambda^r_{\Delta_{ij}}$, they are *time-interval sensitive*, in that they depend on the time difference Δ_{ij} between the detections. This allows a fine modeling of the problem and will be illustrated below.

Finally, in Eq. 2.1, $\delta(.)$ denotes the Kronecker function ($\delta(a) = 1$ if $a = 0$, $\delta(a) = 0$ otherwise). Therefore, coefficients β^r_{ij} are only counted when the labels are the same. They can thus be considered as "costs" for associating or not a detection pair within the same track. When $\beta^r_{ij} < 0$, the pair of detections should be associated so as to minimize the energy 2.1, whereas when $\beta^r_{ij} > 0$, it should not.

2.3.3 Features and association cost definition

Our approach relies on the unsupervised learning of time sensitive association costs for $N_s = 8$ different features. Below, we briefly motivate and introduce the chosen features and their corresponding distributions. We illustrate them by showing the Potts curves (for their learning see next section), emphasizing the effect of time-interval sensitivity and their easy adaptation to different datasets.

Position. The similarity is the Euclidean distance $S_1(y_i, y_j) = \mathbf{x}_i - \mathbf{x}_j$, with $\mathbf{x}_i$ the image location of the i^{th} detection y_i. The distributions of this feature are modeled as zero mean Gaussians whose covariance Σ^H_Δ depends on the hypothesis (H_0 or H_1) and the time gap Δ between two detections. Fig. 2.3 illustrates the learned models by plotting the zero iso-curves of the resulting β functions. We can notice the non-linearity with respect to increasing time gaps Δ (curves are closer and closer as Δ increases), and the difference between document types: more static heads are expected in the REPERE TV programs than in the Hannah movie.

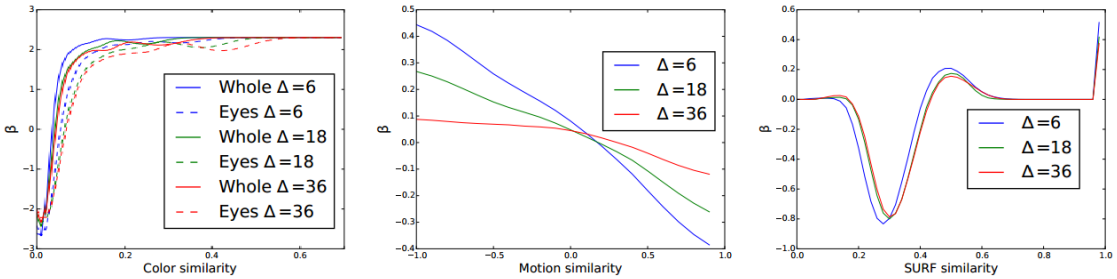


Figure 2.4 – Automatically learned Potts functions β for different similarity functions and some Δ values. Left: color cue. Middle: motion. Right: SURF.

Figure 2.5 – An example of motion vectors in 2 frames t (left) and $t+\Delta$ (right). The cosine similarity between the motion vector in frame t is closer to 1 for the right face in frame $t+\Delta$ while being smaller for the left face in the same frame.

Motion cues. Motion similarity between detection pairs is assessed by comparing their relative displacement and their visual motion. This motion is estimated by [37]. The similarity is computed as the cosine of the angle between these two vectors. Intuitively, if a face moves in a constant direction, the displacement between its detections and their visual motion will be aligned, leading to a motion similarity close to 1, whereas for unrelated faces, this would be more random. An example of these motion vectors is shown in Figure 2.5. Note that the use of such an *instantaneous motion information* differs from frame-to-frame KLT tracking and *is not affected by occlusion or drift*. The resulting β curves in the middle plot of Figure 2.4 confirm the above intuition, but surpringly indicate that this motion information is more discriminative for short time intervals. Indeed, in the TV data, when considereing 1 to 2 seconds time intervals, head motion might be less reliable as people are more likely to shake their heads back and forth, leading to flatter β curves.

Appearance (color). Faces are represented by multi-level color histograms in 4 different regions: the whole face, and the mouth, eye, and nose regions. The similarity between histograms of the same region of the detections is measured using the Bhattacharyya distance D_h, and the distributions of this distance is modeled using a non-parametric method. Example of Potts curve β are shown in Fig. 2.4, Left. We can notice here that the statistics associated to each region are relatively different, and although we would not expect so, also varies with the time gap Δ between detections.

Appearance (SURF). Color is insufficient to discriminate between faces. We thus propose to

exploit SURF [38] descriptors computed at interest points detected within the face bounding box as more structured appearance measures. As similarity measure, we use the average Euclidean distances between pairs of nearest keypoint descriptors from the two detections. We model the distributions of the similarity measures with a non-parametric approach. As can be seen in the right plot of Fig. 2.4, the Potts coefficient β is negative for a SURF similarity around 0.3, thus encouraging association for such values. On the other hand, positive coefficients for larger distances - around 0.5 - discourage the association.

2.3.4 Parameter learning, optimization

Given our non-parametric and time interval sensitive cost model, the number of parameters in λ is quite large. We adopt an unsupervised learning strategy to estimate λ directly from data, removing the need for tedious track annotations. Learning is done in two steps. First, we rely on a simple assumption that up to a short term interval, pairs of closest and second closest detections come from the same person or not, respectively. This allows us to learn model parameters under each of the two hypotheses, and perform a first round of tracking. Second, we use the resulting tracks to refine and obtain the model parameters up to larger time intervals. Note that although on test data we are only interested in the parameters at multiples of Δ_{sk} (the frequency at which face detections is applied), during training tracking is done using all intervals to obtain reliable tracks for the parameter refinement.

Optimization. For computational efficiency, we used a sliding window algorithm that labels the detections in the current frame as the continuation of a previous track or the creation of a new one, using an optimal hungarian association algorithm relying on all the pairwise links to the already labeled detections in the past T_w instants. A second step of block Iterated Conditional Modes is then conducted, which allows reasoning at global level by swapping track fragments at each time instant [2].

2.3.5 False alarm track removal

The CRF provides face tracks, some of which may correspond to false alarms. In other trackers [2, 22], these are often simply removed based on track length. On broadcast data, this is not sufficient given the content diversity and track length limited by shot duration. Thus, in contrast to [2], we further learnt a classifier to filter the false alarm tracks based on more cues [39]. For each face track, motion, position, size, and detection confidence scores were collected and accumulated to form a feature vector. Then, a linear SVM classifier was trained to distinguish true tracks from false ones.

To train our model, we created a training database by annotating 9364 face tracks from 9 videos from the development set of the REPERE corpus, and used the obtained model to conducted our evaluation on other datasets as reported in the experiment section. The linear SVM model achieves 93.3% cross validation accuracy. Based on the weights, the most important cues are

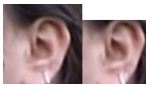
a)  b)  c) 

Figure 2.6 – False alarm removal examples. a) Short but positive track falsely removed by [2] but kept by our model. b) Negative track correctly removed mainly thanks to image position and detection size. c) Negative track falsely kept by [3] due to skin color but correctly removed by our model.

detection score, width, and position. Fig. 2.6 illustrate qualitatively why this multi-cue model is superior to false alarm models based on single feature such as duration [2] or skin filtering [3]. Furthermore, our linear model with simple features is fast to work with large video corpus.

2.4 Face clustering

The goal of face clustering is to merge tracks having the same identity. To achieve this, ideally tracks need first to be represented so that intra-personal variations are reduced while inter-personal differences are enhanced. Then, face tracks can be grouped, potentially using complementary information. We describe below how we addressed these aspects.

2.4.1 Representations and similarity measures

As discussed in the related work, our aim is to exploit face and face track representations good at handling appearance variabilities. To this end, we decided to rely on the state-of-the-art i-vectors which have been shown to perform well on standard biometric verification benchmarks [28, 27]. However, at the same time, face diarization also involves clustering face instances usually shot in similar conditions. Thus, exploiting similarities based on matching methods can be useful in this contex, as shown in the experiments. We thus also propose to use SURF to represent faces. We describe both models below.

SURF similarity. Each track is represented by 9 keyfaces that are equally distributed a long the track. Given the extracted SURF descriptors, the similarity between two faces is measured as the average of the 6 smallest distances between the descriptors of the matched keypoints. The same approach is extended to calculate the similarity D_S between clusters by averaging the $N = 9$ smallest distances between all pairs of keyfaces.

i-vector representation with total variability modeling (TVM). When the face variability increases, matching similarity loses its discrimination as intra-person variance becomes closer to inter-person variance. To improve generalization, features are often sampled densely and the semantic correspondence between images is bridged using statistical models. Here we borrow models from the biometric domain, and used DCT variants as features and i-vectors for this purpose [27].

Regarding features, eyes are detected to register images and the Tan & Triggs [40] algorithm is used to normalize the cropped images. $K = 3657$ 12×12 blocks are densely sampled, preprocessed, and the 44 lowest frequency DCT coefficients are extracted and normalised (across blocks) to zero mean and unit variance, resulting in at set $\mathbf{O}$ of K normalized DCT vectors $\mathbf{o}^k$ to characterize an image.

A face or face cluster i can then typically be represented by the supervector of concatenated means of a Gaussian Mixture Model (GMM) modeling the distribution of the observations $\mathbf{o}^k$ of that face/cluster. This vector, denoted $\mathbf{s}_i$, is obtained through mean-only MAP adaptation of the supervector means $\mathbf{m}$ of a Universal GMM Background Model (UBM) trained from a large set of images, i.e $\mathbf{s}_i = \mathbf{m} + \mathbf{d}_i$, in which $\mathbf{d}_i$ denotes the mean offset specific to the given face. However, due to the inherent sensitivity to the specific conditions in which images are captured [28], offsets $\mathbf{d}$ can be unreliable for cluster comparison. This is critical since in TV broadcast, a person can appear in quite different contexts, especially with respect to viewpoint, pose and illumination.

Thus, factor analysis in general and total variability modeling in particular have been proposed to handle such situations. The main idea to obtain a better and more discriminative representation is to constrain the mean offsets to lie within a linear, low-dimensional subspace representing the principal directions of inter-face variations. Concretely, the TVM supervector representation is modeled as $\mathbf{s}_i = \mathbf{m} + T\mathbf{u}_i + \xi_i$, in which T is the total variability subspace, $\mathbf{u}_i$ the low-dimensional i-vector face representation, and ξ_i is a random gaussian noise with diagonal covariance Σ_{Tv} used to model the residual variability not covered by T. In practice, T and Σ_{Tv} are learned through likelihood optimization, and we used the BANCA and MOBIO dataset as training data. For details about TVM, see [41, 42].

Representation. According to the TVM model, the i-vector representation $\mathbf{u}_i$ of an image i can be computed using the centralised 0th and 1st-order Baum-Welch statistics of the feature vectors $\mathbf{o}^k$ of that image w.r.t. the UBM mean mixture components (N_i and F_i, respectively) as [41, 42]):

$$\mathbf{u}_i = (I + T'\Sigma_{Tv}^{-1} N_i T)^{-1} T'\Sigma_{Tv}^{-1} F_i$$

When dealing with one face track or one face cluster, the same formula can be used but pooling all feature vectors together to calculate the required statistics. Thus a single i-vector is extracted to represent the whole cluster.

To compare clusters, i-vectors are further whitened and L_2-normalized and the distance between clusters C_i and C_j is simply computed using the cosine distance, i.e:

$$D_T(C_i, C_j) = 1 - cos(\mathbf{u}_i, \mathbf{u}_j)$$

Figure 2.7 – Example of shot clusters. Each line shows shot thumbnails of one cluster.

2.4.2 Shot-constrained clustering

To perform the unsupervised clustering of face tracks, we rely on a bottom-up hierarchical clustering, as commonly done [26, 19]. However, since no temporal constraints are usually exploited (except that tracks appear at the same time should not be merged in the same cluster), one must compute the similarity between all pairs of tracks during the hierarchical clustering. Thus the complexity for this task raises at least quadratically with the number of tracks/clusters. Furthermore, as shots are rather short in TV programs, the face variability (esp. in terms of pose) is usually quite low, and thus can limit the representativeness of the extracted i-vectors.

To address these problems, we propose a divide-and-conquer strategy, observing that in TV programs the same person tend to appear in groups of similar shots. By first limiting the clustering within such groups of shots, we can quickly associate tracks which are obviously from the same person, and from there build better face representations for a smaller number of cluster, thus reducing the computational cost.

More concrete elements are as follows.

- First, shots are grouped based on Bhattacharyya distance between color histograms of keyframes. We rely on this representation and a simple hierarchical approach because of its speed and simplicity over other alternatives [23, 35]. To reduce complexity, we constrain association between pairs of shots (or shot clusters) that are within 20 shots away from each other. A very low threshold is used to stop the process. Typically, 50% of the shots (esp. in commercials) remain alone due to their unique color tones, while clusters of more than 2 shots contain an average of 6 shots often showing a person under different poses as illustrated in Fig. 2.7.
- Then, within each shot cluster, face tracks are clustered locally. As faces in such shots tend to be similar, we only use the matching similarity D_S. This process terminates when D_S reaches a threshold Th_1.
- Third, all face clusters, which are now much less in quantity, are hierarchically merged using a combination of matching and biometric i-vector distances D_S and D_T. More precisely, as D_S is distinctive and adequate for matching between face tracks captured

in comparable conditions, the initial clusters are merged based on it until a conservative threshold Th_2 is reached. Then, reliable i-vectors are extracted from these bigger clusters which are gradually merged according to a distance combination (i.e. using $D_S + \alpha D_T$) until a last threshold Th_3 is attained.

2.5 Experiments

2.5.1 Evaluation Datasets

We conducted evaluation separately at each stage of the face diarization. For this purpose, three public datasets were used to benchmark the results:

- "Frontal and Turning". It consists of 2 videos recorded with a fixed camera [43]. In each video, there are 4 subjects moving around. In Frontal video, there are frequent occlusions and fast movements while the Turning video contain many profile faces.
- "Hannah". It is manually annotated based on the movie "Hannah and her sisters" by W. Allen [22]. This dataset is challenging due to dynamic cameras with faces of many characters at multiple poses and angles.
- REPERE corpus. It features 9 programs including news, debates, and talk shows from two French TV channels (LCP and BFMTV) [44] with sparse annotations. One face track is annotated with its corresponding identity using one reference frame only. From the Test 2 subset of the REPERE challenge, we randomly selected 27 videos equally from each program. These videos cover approximately 18 hours of data. 9 of the videos are used for parameter tuning and the 18 other ones are used for testing.

2.5.2 Tracking evaluation

Evaluation of tracking on Frontal/Turning dataset. We use the face tracking metrics used by [45] on this dataset to evaluate the results: Mostly Tracked (MT, number of groundtruth trajectories correctly tracked for more than 80% of their duration, the bigger the better), Fragmentation (Frag, number of times groundtruth trajectories are interrupted, the smaller the better), ID Switches (IDS, number of times tracked trajectories change matched groundtruth identity).

Our results are reported in Table 2.1 for different parameter configurations. The first number is the tracking window size T_w and the second number X is the frequency at which detection is performed (every X frames).

When varying these parameters, one can observe that when the tracking window T_w is wider, tracks are more likely to recover from temporary occlusions or missed detections, which usually results in less Frag and higher MT (compare sets of results for $T_w = 48$ vs $T_w = 36$). On the other hand, when detection is applied very scarcely (e.g. every 12 frames), we can notice an important performance decrease (e.g. 2 vs 6 Mostly tracked people for $T_w = 36$ on the Frontal sequence).

Figure 2.8 – Snapshots of tracking results on two benchmarking datasets - left, sample frame from Frontal/Turning dataset used for evaluating face tracking. Right, one frame from a broadcast program in the REPERE dataset.

	Frontal				Turning			
T_w-X	PH	MT	Frag	IDS	PH	MT	Frag	IDS
36-1	28	**6**	**16**	**0**	15	**4**	**9**	**0**
36-6	26	5	18	**0**	18	3	16	**0**
36-12	38	2	30	1	30	0	27	**0**
48-1	27	**6**	**15**	**0**	14	**4**	**8**	**0**
48-6	25	5	17	**0**	16	**4**	14	**0**
48-12	37	3	30	2	30	1	26	1
[24]	11	4	24	13	11	2	**8**	4
[33]	15	5	25	10	15	**4**	**8**	5

Table 2.1 – Tracking results on "Frontal and Turning". The parameters denote: T_w, up-to how many frames apart are pairwise links built. X: detections are only extracted every X frames.

However, applying the detection every 6 frames produces only a small loss of performance, and since detection is one of the bottlenecks for the face diarization stage, provides a good trade-off between performance and speed.

When comparing with other methods in the literature [24, 33], our system outperforms them in both scenarios, with much less ID switches and Frag overall, and higher or the same MT. Indeed, as tracking is done within a long temporal window and the detection enough recall, we can track most of the groundtruth tracks. Most importantly, the number of IDS is minimized, which is crucial for further person clustering and naming. Though tracks are still frequently fragmented due to long occlusion, they could be further joined by face clustering in the next step. It is important to note that we did not apply false track removal on this dataset, explaining why we have more predicted hypotheses (PH). A qualitative visualization is presented in Figure. 2.8.

Evaluation on Hannah dataset. Frame by frame annotation allows us to evaluate the detector

	Frame-based			Track-based		
	FP (%)	FN (%)	MultT (%)	OPu (%)	TPu (%)	Purity (%)
[22]	17.4	39.2	0.39	22.5	50.6	31.2
[3]	13.2	66.6	**0.07**	12.3	66.7	20.7
Ours, $X = 1$, no FAR	36.3	**33.7**	1.25	**35.9**	54.2	**43.2**
Ours, $X = 6$, no FAR	29.6	40.6	6.08	28.1	56.3	37.5
Ours, $X = 6$, FAR	**5.3**	42.6	0.72	27.5	**91.1**	42.3

Table 2.2 – Evaluation of our tracking framework against other baseline systems on Hannah dataset.

and tracker with both detection-based as well as track-based metrics, as used by [22] for this dataset:

- Frame-based evaluation compares faces returned by the system (track boxes) and groundtruth faces (GT boxes) to calculate 3 measures: False Positive (FP), False Negative (FN), and Multiple Track (MultT, the ratio of GT boxes with multiple matches). It is important to note that these boxes are considered **after** the tracking phase.
- Track-based evaluation reflects the purity of matching through 3 metrics: Tracker Purity (TPu), Object Purity (OPu), and Purity. For each output track, its tracker purity is calculated as the ratio of frames for which it correctly identifies the GT track it is associated with, over the total length of the output track. Similarly, for each GT track, its object purity is calculated as the ratio of frames for which it is correctly identified by the output track it is associated with, over the total length of the GT track. Averaging over all output tracks and GT tracks yields TPu, and OPu, respectively. Purity measures the overall quality of face tracks based on TPu and OPu.

We compare the proposed system with 2 strong baselines, each of them illustrating a different approach to the problem. The first baseline is proposed by [22]. Their detector is a combination of frontal and profile VJ detectors with Zhu and Ramanan multi-pose detector [46], which produces high frame-based score. Tracking is done with an improved version of the KLT tracker. The second baseline utilizes only the frontal Viola & Jones detector and GMM-based skin filtering [3]. Tracking is done by associating pairs of detections based on matching similarity together with forward and backward search. Because false alarms are minimized and frontal faces are easier to connect with exhaustive search, this baseline produces fewer false positives and a high tracker purity score.

For our systems, there are 3 different configurations, with $T_w = 36$ in all cases: $X = 1$ with no false alarm removal (FAR), $X = 6$ without FAR, and $X = 6$ with FAR. Table 2.2 shows the detailed comparison of all the systems. First, we observe the frame-based evaluation. At step $X = 1$, there are more detections, thus lower FN, but high FP. When step X gets larger, FN increases and FP decreases as expected. When FAR is applied, it greatly reduces the FP with only a minor loss in FN. Our tracker yields the highest results at frame-based level. At the track-based level, our

	Dev set					Test set				
	Recall	Precision	F1	Missed T.	FA T.	Recall	Precision	F1	Missed T.	FA T.
Baseline [3]	39.2	**94.3**	55.4	68.1	5	43.9	**96.8**	60.3	62.7	8.9
Ours, no FAR	**60.6**	79.1	68.6	52.0	8.7	**58.2**	82.2	68.2	53.5	11.8
Ours, with FAR	59.1	93.1	**72.3**	52.5	4	57.0	94.8	**71.2**	55.6	7.3

Table 2.3 – Detection and tracking performance on the 27 videos coming from the REPERE Test2 dataset, split according to your training and test sets (see Section 4.1).

system outperform all other baselines. Because false alarm tracks are taken into account when computing the tracker purity TPu, FAR significantly contributes to improving TPu (moving from 56.3% to 91.1%) with a very minor drop of OPu. When comparing the Purity indicator, $X = 6$ with FAR performs equally to $X = 1$ with an advantage of acceleration by 6 folds. This further confirms our expectation of a powerful and high-speed detector/tracker system.

Evaluation on REPERE corpus. The REPERE corpus does not provide dense annotation but only temporal bounds and one head position at a single reference frame are given for each track. Therefore, we evaluate the performance only indirectly by measuring the detection performance on these reference frames, where the detections in one reference frame come either from the raw detector, or are generated through interpolation of one track at that frame.

In the 27 selected videos, there are 4130 annotated heads. Based on the intersection of the groundtruth polygons and tracked hypotheses at these reference frames, one can calculate the recall, precision, and F1-measure. Then to report performance at the track-level, we simply weight the detection errors by the duration of each track active at the reference frame. This results in the false alarm time rate (denoted FA. T), i.e. the sum of the false alarms weighted by the track duration divided by the total duration of all reported tracks. We can similarly compute the Missed time (Missed T.).

The tracker configuration is the following: faces are detected every 6 frames, and links between detections could be made up to 36 frames apart. The baseline consists of frontal detector, skin filtering, and SURF-based tracker as in previous experiments [3].

Table 2.3 shows the results of each system. At frame-based evaluation, thanks to the DPM face detector, the recall is increased by quite a large margin (on the test set, from 43.8% for the baseline relying on VJ detector to 58.2%). However, this is at the cost of an increase of false alarms (reduction of precision). These number can be improved by applying the false alarm track removal step. In that case, the precision increases by almost 13% (from 82.2% to 94.8%) while only losing 1% in recall. Compared to the baseline, we gain more than 12% in F1-measure. Looking at the weighted measure, we can note that though our system without FAR has around 20% false detections at frame level, the total FA time is only around 10%. This means that our false alarms tend to form short tracks. We visualize a tracking result from a REPERE broadcast example in Figure. 2.8.

Our system can also be applied for more general multi-object tracking tasks such as pedestrian tracking. We present these tracking results in the Appendix A.

2.5.3 Face clustering evaluation

Metrics and Baseline. The performance of the clustering task is measured according to two different metrics: Cluster Purity (CP) and Clustering Confusion Error Rate (CER).

- The CP of one cluster is the proportion of the largest number of frames of that cluster belonging to one identity to the total number of frames of this cluster. The CP aims to control if we overcluster tracks.
- CER is measured by computing the overall person time that is attributed to the wrong person and normalized by total duration of the videos [3].

Face tracks in all cases are provided by our system. To compare with our clustering framework, tracks are also grouped using the baseline proposed by [19].

Evaluation of shot constrained face clustering. We assess the impact of 2 different clustering schemes: (i) using only global face clustering (global) with matching similarity and (ii) applying a shot constrained face clustering (SCFC) step before our global face clustering (local + global). Because the main goal of SCFC is to quickly merge clusters without making mistakes, CP is the main metric of concern.

Fig. 2.9-left shows how CP changes as the clusters are merged locally or globally. SCFC can still preserve the purity comparable to exhaustive clustering while decreasing significantly computation time. In figure 2.9-right, CER stops to decrease in SCFC after a certain point. This shows that SCFC has reached its limit. Therefore, we choose the threshold Th_1 to stop SCFC at 800 clusters on the training set and apply it on the test set.

Table 2.4 compares the global exhaustive clustering without and with local SCFC. After the first local clustering, most easy face tracks are quickly associated. Therefore, the number of face clusters as input for global clustering is drastically reduced to 40% of the original numbers and the total running time decreases accordingly. The total running time is reduced to 25%. The effect is even more visible for long videos (more than 45 mins) where running time is cut down by 8 to 10 times. Besides the speed, we can also observe the improvement in CER in figure 2.9-right when applying global clustering after local SCFC due to less noise in the SCFC.

Evaluation of fusing similarity. Figure 2.10 shows how the CER evolves during the process with 3 different similarities: keypoint-based similarity D_S, TVM-based similarity D_T, and the fusing similarity D_{S+T}. In the first stage of the graph, face clusters are matched using only keypoint-based similarity. During this stage, closely similar clusters are easily merged first, thus reducing the CER sharply. However, as the intra-person variance increases, matching similarity gradually loses its discriminative advantage. Based on the curve of D_S, we can choose the

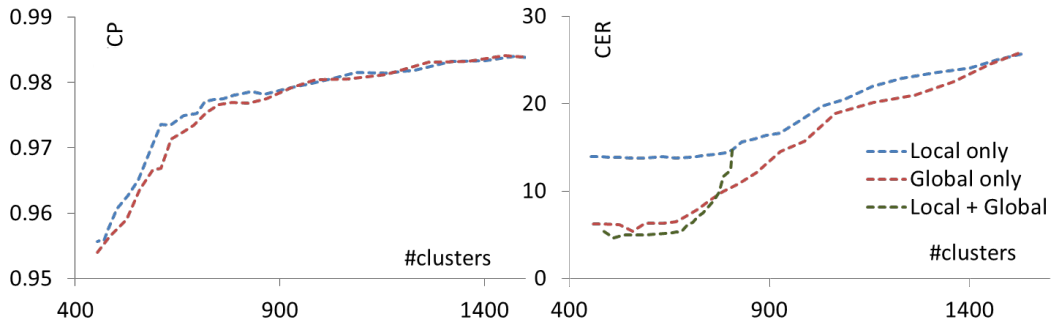


Figure 2.9 – SCFC evaluation based on 2 metrics versus the number clusters. Left: cluster purity (CP) Right: confusion error rate (CER).

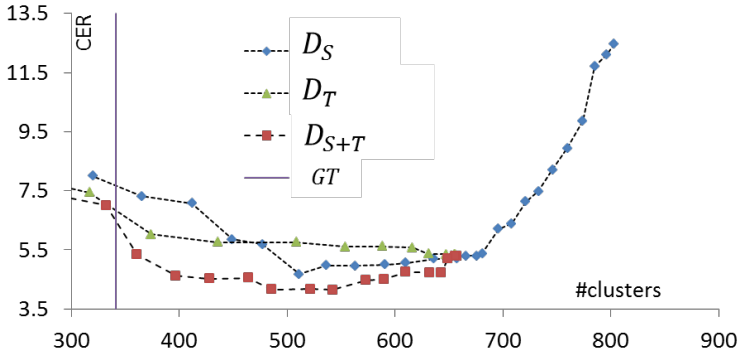


Figure 2.10 – Evolution of CER during hierarchical clustering with different similarities. During the first part of the clustering (from more than 800 clusters to less than 700, only the matching similarity D_S is used.

threshold Th_2 used to start representing clusters with TVM on the test set. TVM performs consistently as the i-vectors are capable of capturing both the person-dependent variabilities and condition-dependent variabilities. The combination of these two similarities achieves the best result in the development dataset at the minimum CER as well as when the number of clusters reaches the number of groundtruth identities $GT = 341$.

In Table 2.5, we can observe more clearly the improvement of total variability modelling over the baseline [19] using standard GMM adaptation. It is also interesting that matching similarity D_S is still important to improve the performance. This shows that although i-vector can capture well intra-person variabilities, there is still room for improvement with further discriminative

	No. clusters	Time (mins)
Global	21263	22497
Local + Global	8883	5465

Table 2.4 – Comparison on 27 videos of REPERE test 2 when performing SCFC using matching similarity. "No. clusters" denotes the number of face clusters available as input for global clustering step (which contains false alarms and unannotated clusters).

	[19]	D_S	D_T	D_{S+T}
CER	11.2	9.3	9.7	8.2
No. clusters	1182	798	728	917

Table 2.5 – Comparison of clustering using different similarities on the test set of REPERE

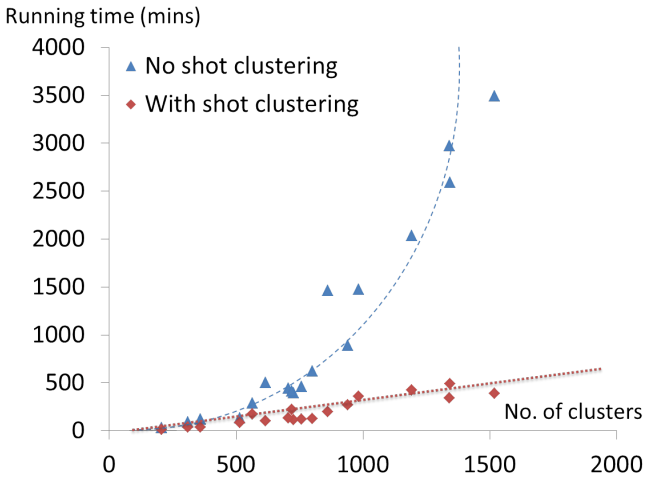


Figure 2.11 – Face clustering time given the number of initial face tracks. For videos with more face tracks, the clustering time increases quadratically. By dividing a video into clusters of shots, the total time for our 2-stage face clustering only increases linearly with the number of face tracks.

modeling.

Computational performance. Besides accuracy, the running time is a major concern for face diarizaton systems. On an Intel(R) Core(TM) i7-4930K CPU @ 3.40GHz machine, for HD images (1024x756), the detector can process 3-4 frames/s, yielding real time speed when applying it only on 4 frames per second. For comparison, frontal and profile VJ detectors run at 6 - 7 frames/s on the same machine. For 1 hour of HD video, the tracker costs around 1.5 hour in total including motion estimation on detections. Comparing to standard hierarchical clustering, our divide-and-conquer strategy and the faster biometric measure comparison using i-vectors lead to an important computation decrease by a factor ranging from 2 to 3 on short videos and 8 to 10 on longer videos (more than 45 minutes). In Figure 2.11, we plot how the face clustering time evolves with different number of face tracks per video. Without shot clustering, when the number of face tracks increases, the computational cost increases quadratically. By dividing-and-conquering each shot clusters, we reduce the growth of face clustering time to a linear growth with respect to the number of face tracks.

2.6 Conclusion

We have presented our diarization system with a novel tracking and shot-constrained clustering methods. Unlike other methods, our tracker is able to exploit long term connectivity to perform tracking across a long gap of frames, thus allowing us to take advantage of the robust DPM detector. Meanwhile, the efficient shot-constrained clustering scheme speeds up the face clustering process significantly. We also apply and combine the total variability modeling with the matching similarity, which further improve the accuracy of our system. Our contributions are evaluated on standard datasets and yield state-of-the-art results. Nevertheless, the experiments show room for more exploration of face representations. As our methods still rely on primitive elements (color, motion, position) and local features (SURF, DCT), further improvement can be expected using recent deep face recognition. It is also beneficial to combine the tracker with other state-of-the-art deep face detectors.

Building from the result of face processing in this chapter, we will introduce the full system , which takes advantage of both audio and visual streams for person naming, in the next chapter.

3 Multimodal Person Discovery

3.1 Introduction

In this chapter, we address the main problem of person discovery in broadcast TV. As person indices can be used to retrieve identity of people presented in the archive and to obtain their respective quotations, a person naming system is indispensable for searching archives. To this end, a video must be segmented into segments during which a person appears or speaks. Then, person names must be extracted from the videos and assigned to the corresponding segments to when a person appears or speaks. As the identity information comes from both the face and the voice of a person, there is a need to correctly associate a face track and a voice track.

Hence, in the first part of the chapter, we introduce the task of talking face and dubbing detection, which enables face and speech association for audio-visual person diarization. In the second half, the full person discovery system with all components are detailed. Finally, quantitative results in the person retrieval task and qualitative demonstration of the EUMSSI project are presented. Overall, the main highlights of this chapter are:

Face-speech association problem. To correctly associate a speech segment to a face track and an overlaid name, one needs to verify the synchronicity between the face and the voice. This is especially important in TV broadcast when there are commentary scenes or when a different language is dubbed over the true voice. To solve this task, we propose a method to represent the temporal relationship between the auditory and visual streams. It consists of a canonical component analysis (CCA) transform to learn a joint multimodal space and a long short term memory (LSTM) network to model cross-modality temporal dependencies.

Audio-visual naming system. The system consists of four main stages: face and speech diarization, candidate identity retrieval, and audio-visual association, and person naming. Figure 3.1 shows the diagram of these four stages. In the first stage, we use the face diarization system from the previous chapter and the speech diarization provided by LIUM [47], our project partner, which will be briefly described. For the second stage, person identities can be retrieved either

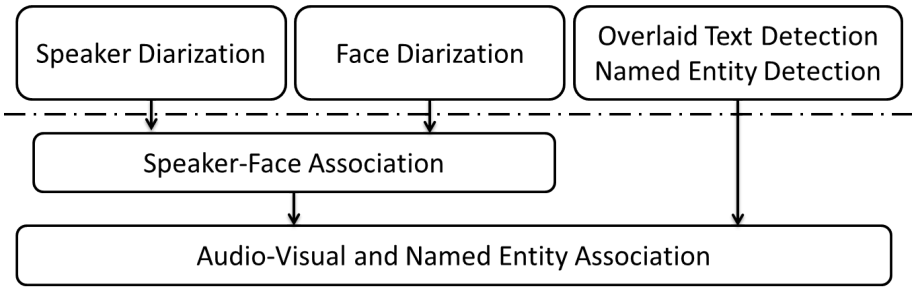


Figure 3.1 – Architecture of our system

from speech transcripts or from the onscreen names commonly used to introduce the current speaker. Here, we use the onscreen names as the main cue because identities can be reliably extracted using Optical Character Recognition (OCR) techniques, and their association with people in the videos is easier than analysing pronounced names in ASR transcripts. And finally, faces and speakers need to be associated using the talking face and dubbing detection model. Then the names are propagated to the speakers or faces of the identities of the persons in the show.

The chapter is structured as follows: Section 3.2 introduces our work in audio-visual association, Section 3.3 describes the person naming system in details, experimental and evaluation results are presented in Section 3.4, Section 3.5 shows the the EUMSSI demonstration and project outcome, and Section 3.6 concludes the chapter.

3.2 Talking face detection and dubbing detection

Solving the AV person diarization and naming tasks requires associating visual person tracks or overlaid names with auditory voices, which has several difficulties. Firstly, the visible person may not be the current speaker. This issue occurs when anchors or invited speakers are commenting on video footage displaying famous people who might be talking or when several persons appear to be talking in the background. This cannot be solved with existing systems which visually detect talking faces [48, 49, 50, 51] to reinforce the AV association. Secondly, another recurrent issue in international TV is dubbing. The problem is common when an interviewed person, shown in the video, is speaking in a different language than that of the target audience, and is dubbed by a narrator.

We focus on dubbing detection in broadcast data, which involves modelling the synchrony of audio and lip motion. This task can be used to handle the two issues mentioned above, by detecting *which of the talking persons (if any) actually produces* the audio discourse. Although it is related to several research problems (AV speech recognition, voice over detection, spoofing in AV biometry), to the best or our knowledge, this dubbing problem has not been addressed previously. To initiate further research, we acquired the DW-dubbing corpus comprising 4722 segments of 2 seconds with the corresponding face track and audio. In addition, from a methodological

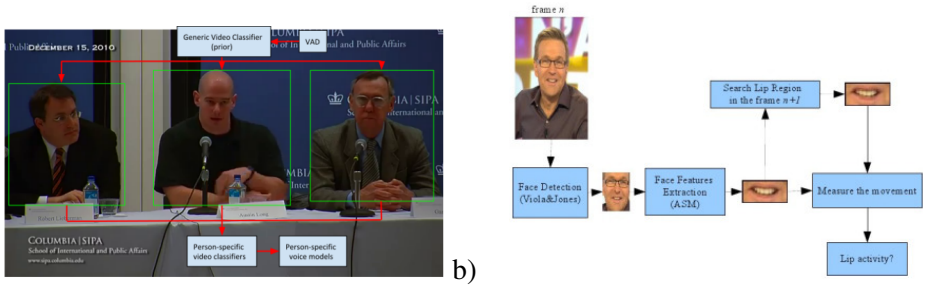


Figure 3.2 – Examples of vision only systems. (a) used head and upperbody [50] while (b) used only mouth motion [49]

perspective, we propose to exploit the recently revived LSTM networks to model the joint dynamics of synchronized AV segments in a multimodal space obtained via canonical correlation analysis (CCA). Experiments demonstrate the benefit of our method over several baselines. In summary, our contributions are:

- We address for the first time the problem of dubbing detection in broadcast data;
- We propose a method relying on LSTM and multimodal feature extraction, which achieves promising result on this problem;
- We make publicly available a dubbing dataset collected from TV news for future research.

3.2.1 Related work

Dubbing detection shares some similarities to several problems discussed below along with the related works.

Talking faces. Person diarization and naming require matching audio segments with face tracks of talking people. To detect talking people, mean squared intensity differences [48] or motion entropy [49] within mouth regions were typically used, potentially combined with head motion [50]. These vision only methods are illustrated in Figure 3.2 Such visual-based approaches could be further enhanced using multimodal contextual information, like audio segment-face track overlap duration, or face size and position [51]. It is interesting to see that none of the existing systems relied on temporal models for this task. Also, when several persons are seen talking, or in dubbing situations, visual information is insufficient to address the task, and audio and video need to be jointly considered.

AV speech recognition. Modeling the relationship between audio and visual streams can be traced back to early researches in multimodal speech recognition [52]. Typical examples include coupled hidden Markov Model (CHMM) [53] or asynchronous HMM to model anticipation and retention phenomena. More recently, multimodal deep networks [54] showed good performance. However, the approach did not include temporal models: it relied on neural networks applied to groups of frames, whose outputs were averaged over time. Furthermore, the task is limited to the

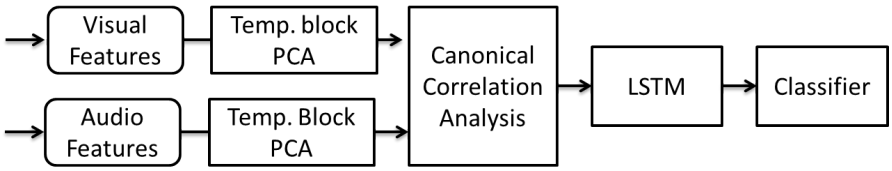


Figure 3.3 – System overview: feature extraction, dimensionality reduction over concatenated feature from block of frames, mutual information extraction via Canonical-Correlation Analysis (CCA), and temporal modelling with LSTM.

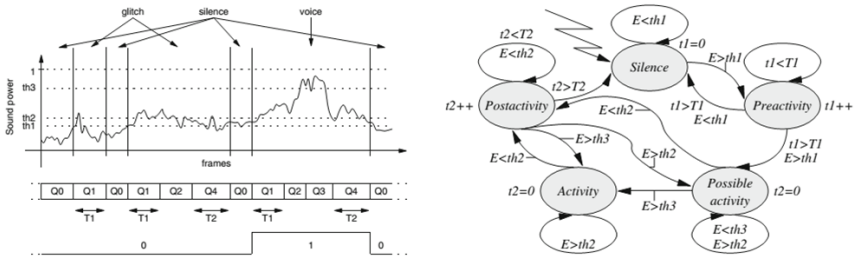


Figure 3.4 – A related work by [57] that was based on either co-inertia to detected uncorrelated AV signals (spoofing cases), or on coupled HMM to detect unsynchronization. This work focus more on the temporal classification models but not learning the representation of the sequences. Also, the method was applied to a constrained biometric environment, with specific test sentences used as input to the system.

recognition of simple sounds [55] with little noise (head movements, illumination).

AV biometry and synchronization. The dubbing problem somehow resembles AV spoofing detection, where the task is to detect when visual attackers pretend to match with a playback audio. As an early work, cross-modal fusion with latent semantic analysis or canonical correlation analysis were applied, but only tested attacks composed of a single photograph, potentially animated with simple synthetic movements [56]. To deal with real face tracks with different voices, [57] investigated co-inertia or coupled HMM approaches to detected uncorrelated AV signals or unsynchronization. However, the method was applied to a constrained biometric environment, with specific test sentences used as input to the system. This work, as shown in Figure 3.4, only focus on the classification model but not the representation of the sequences.

In addition, synchrony detection has also been addressed for speaker location & association [58, 59] in scenes with two people, and has focused more on mutual information modeling than temporal aspects. Mutual information was also shown to be important in monologue detection where a system needs to identify real speakers among sets of confusers [60, 61]. However, temporal modeling using HMM to evaluated likelihood of word utterance given joint AV distribution only yielded limited results [60]. Another related problem is to distinguish narrated vs genuine voices in TV news addressed in [62], where only primitive lip features were used without joint modality space or temporal modeling, and the dataset was very small (40 video

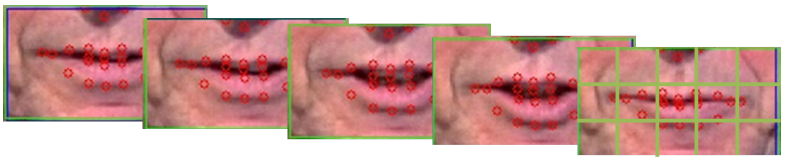


Figure 3.5 – Example of mouth boxes. Mouth region is detected based on landmarks. Features are computed in 3×5 grid and grouped in a block of 5 frames.

clips). In contrast to the above works, our approach utilizes both cross-modal correlation analysis and temporal modeling with state-of-the-art LSTM. Furthermore, our dataset is collected from TV with unconstrained settings and unrestricted speech content.

AV modeling with Neural networks. In addition to the AV speech recognition [54], there has been more attention towards using deep neural networks (DNNs) for AV speaker naming with audio and visual streams. [63] used DNNs to jointly learn recognition models from 2 input streams. This work is further extended in the temporal domain with multimodal LSTM by [64]. Nevertheless, these works require identity information and are thus closer to biometric joint recognition than unsynchronization speech detection.

3.2.2 Multimodal framework

The overview of our system is illustrated in Fig. 3.3. First, features are extracted per frame for each modality. Subsequently, blocks of frames are concatenated and dimensionality reduction is applied. This is followed by cross-modality correlation modelling, whose outputs are modelled in the temporal domain using an LSTM to get the high level representation used for classification.

3.2.3 Feature extraction

Our goal is to build a full neural network to represent audio-visual speech. However, in this paper, we rely on standard features which should be sufficient for the task.

Visual stream. First, to obtain face tracks, we rely on the tracking-by-detection method described in [10]. Then, the mouth region is localized within each frame. This is done by detecting landmarks using the DLIB implementation of [65].

To characterise the mouth dynamics, dense optical flow is computed using the OpenCV implementation of [66]. The average flow is subtracted to remove head motion, and the residual flows are quantized into 8 bins based on their angular values, with 1 additional bin for close to static points. The mouth region is divided into 3×5 spatial regions in which flow histograms are computed, resulting in a vector of $3 \times 5 \times 9 = 135$ dimensions.

Audio stream. Every 10ms, we extract from 20ms windows Mel-frequency cepstral coefficient

(MFCC) features with 13 coefficients and energy level together with first and second derivatives, resulting in a vector of 42 dimensions.

3.2.4 Multimodal processing

As often done in gesture recognition [67] and in NN-based AV speech recognition [54], we consider observations over a short interval (0.2s as in [67, 54]) to capture short-term temporal dynamics. Here, a block of 5 visual frames are grouped together (675-dim vector), which corresponds to 20 audio frames (840-dim vector). Principal component analysis (PCA) is applied separately to each modality to keep 95% of the variance, resulting in vectors of $N_{\text{V}} = 100$ (visual) and $N_{\text{A}} = 90$ (audio) dimensions.

Canonical-correlation analysis (CCA). The two modality streams contain different types of information. For example, audio may contain features about identity, semantics, or emotions which are irrelevant for our task and may have little correlation with the visual stream. To capture the synchrony between the two modalities, we use CCA, a powerful multivariate statistical technique. Its principle consists of learning matrices, one for each modality, which project the paired modality samples into a common space where the cross-correlation between the projected samples is maximized. For instance, let $X_A \in \mathbf{R}^{N_{\text{A}} \times N}$ and $X_V \in \mathbf{R}^{N_{\text{V}} \times N}$ be N audio and visual samples, respectively. Looking at a one dimensional subspace, CCA looks for the projection weights $w_{\text{A}} \in \mathbf{R}^{N_{\text{A}}}$ and $w_{\text{V}} \in \mathbf{R}^{N_{\text{V}}}$ such that:

$$\max_{w_{\text{A}}, w_{\text{V}}} corr\left[w_{\text{A}}^T X_A, w_{\text{V}}^T X_V\right] \qquad s.t. \, ||w_{\text{A}}|| = 1, ||w_{\text{V}}|| = 1.$$

Such optimization is conducted by finding w_{A} and w_{V} using the eigenvalue decomposition method on the correlation matrix, and then generalized to find the common subspace in which the audio stream and visual stream are most correlated [68]. Thus, features from this subspace can represent how two modalities harmonize with each other, which will be important to detect dubbing events.

3.2.5 Temporal modeling and classification

In this part, we introduce the LSTM architecture and then describe how it is used in our dubbing detection task.

Long Short Term Memory. In sequence modelling, the typical challenge is to learn a model mapping an input sequence $\{x_0, x_1, ..., x_n\}$ to an output sequence $\{y_0, y_1, ..., y_n\}$ where predictions at step t should use the knowledge from x_0 to x_t. To tackle this challenge, RNNs were introduced and shown to learn both high level representation of input signals and temporal dependencies. However, due to gradients multiplications during back propagation through time, they suffered from *exploding* or *vanishing* gradients, making it hard to learn long range dependencies [69].

LSTMs were introduced to overcome these issues [70]. The key ideas were to add a memory

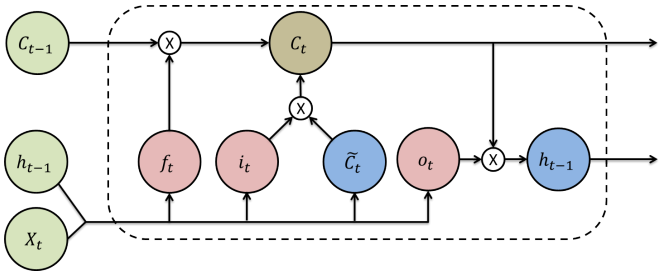


Figure 3.6 – Temporal models. LSTM illustration. Red circles denote sigmoid activation of the gates while blue circles denote tanh activation of the states. × circles denote point-wise multiplication.

cells C_t to store useful information to model long term dependencies, as well as explicit gating mechanisms to regulate the memory updates, as illustrated in Fig. 3.6 and indicated by the formulae below:

$$
\begin{aligned}
Gates: \quad f_t &= sigm(W_{xf}x_t + W_{hf}h_{t-1} + b_f), && (3.1)\\
i_t &= sigm(W_{xi}x_t + W_{hi}h_{t-1} + b_i), && (3.2)\\
o_t &= sigm(W_{xo}x_t + W_{ho}h_{t-1} + b_o), && (3.3)\\
States: \quad \tilde{C}_t &= tanh(W_{xc}x_t + W_{hc}h_{t-1} + b_c), && (3.4)\\
C_t &= f_t \times C_{t-1} + i_t \times \tilde{C}_t, h_t = o_t \times c_t && (3.5)\\
Output: \quad y_t &= W_y h_t + b_y, && (3.6)
\end{aligned}
$$

where $W_.$ and $b_.$ denote weight matrices and biases. The mechanism works as follows. First, new information are processed from current states x_t and h_{t-1} to yield $\tilde{C}_t$. Then, to update C_t, the LSTM can selectively decide how much information from the past needs to be "remembered" or forgotten by passing C_{t-1} through the forget gate f_t, and replaced (reset) by new information $\tilde{C}_t$ through the input gate i_t. Finally, through the output gate o_t, the LSTM selects which C_t components to use to generate the hidden states h_t, from which the LSTM output y_t is produced. Importantly, the strategy to open or close gates is data driven and automatically learned from the data through the trainable $W_.$ and $b_.$. Also, the weighted addition of $\tilde{C}_t$ and C_{t-1} is crucial for LSTMs to avoid the vanishing gradient issue and to propagate gradient through long intervals.

Multimodal LSTM. Let $X = \{x_0, x_1, ..., x_n\}$ be a sequence of CCA projections for one segment. Because our task is binary, we have only one supervised signal denoting the class (Authentic or Dubbing). Straightforwardly, one could thus define the output sequence as a series of only 0s or of only 1s when appropriate and learn the LSTM classifier. However, such an approach does not constrain enough the network parameters, thus quickly leads to overfitting. Furthermore, in one dubbing segment, not all frames look asynchronous, thus forcing the label to 0 at every step can be misleading for the network to learn.

To overcome this challenge, similarly to [71, 72], we propose to train the LSTM in an unsuper-

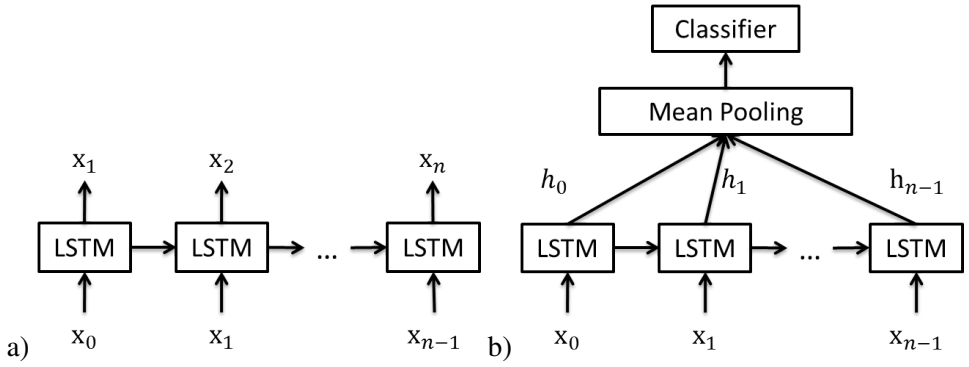


Figure 3.7 – LSTM model. a) At each step i the LSTM learns to predict the feature vector x_{i+1} from the next time step. b) The LSTM is applied to the input sequence, and the sequence of hidden states h_i are averaged and used as input for classification.

vised fashion with a bottleneck hidden layer h of size N_h: at each step t, the LSTM needs to predict the feature x_{t+1} of the next step, as shown in Fig. 3.7a. This architecture can learn good features for two related reasons. First, the hidden layer must be able to extract and compress the essential information from the input vector to make predictions. Since an input vector x_i is composed of two feature vectors of equal size coming from each modality, several hidden units will be able to capture the existing correlation between modalities, whereas others will perform intra-modality predictions (see Fig. 3.10). Second, to make better predictions and learn retention and anticipation temporal phenomena across modalities, the LSTM must also rely on features observed several steps in the past.

Finally, on a test sequence, the extracted hidden representations are mean pooled over the whole segment to form a single vector used for classification, as shown in Fig. 3.7b.

3.3 Integrated person discovery system

To automatically index all people in raw TV broadcasts, each shot must be automatically tagged with the name(s) of people who can be both seen as well as heard in the shot along with the confidence score. The list of people is not known apriori and their names must be discovered from video text overlay or speech transcripts [73]. To this end, a video must be segmented in an unsupervised way into homogeneous segments according to person identity, like speaker diarization and face diarization, to be combined with the extracted names.

Our goal is to benchmark our system in all components and address the fusion of multimodal results. The system we proposed is illustrated in Fig. 3.1. It consists of 4 main parts: video optical character recognition (OCR) and named entity recognition (NER), face diariation, speaker diarization, and fusion naming.

Person discovery challenge at MediaEval 2016

To evaluate our multimodal system, the EUMSSI consortium participated in the MediaEval challenge 2016 [73]. The goal of this challenge is to *identify all people who simultaneously appear and speak in a video corpus,* which in principle requires information from both the audio and visual streams. The submission of competitors must contain not only the final automatic annotations but also the evidence of such claims. The evidence enables manual verification of the annotations and in general human annotators can use them as inputs to greatly decrease manual effort in a collaborative process. In the end, the EUMSSI fusion system won the challenge and scored highest in the test data.

3.3.1 Video OCR and NER

To detect OCR segments in videos and exploit them for retrieval, we first relied on the approaches described in [74, 75] for text recognition in videos, and on [76, 77] for text recognition and indexing. In brief, given an input video, two main steps are applied: first the video is preprocessed with a motion filtering to reduce noise, and individual frames are processed to localize and binarize the text regions for text recognition. As compared to printed documents, OCR in TV news videos encounters several challenges: low resolution of text regions, sequence of different texts continuously displayed, or small amount of text to be recognized etc. To tackle these, multiple image segmentations of the same text region are decoded, and then all results are compared and aggregated over time to produce several hypotheses. The best hypothesis is used to extract people names for identification. To recognize names from texts, we use the MITIE open library [1], which provides state-of-the-art NER tool. To improve the raw MITIE results, a heuristics preprocessing step identifies names of editorial staff based on their roles (cameraman, editor, or writer) because they do not appear within the video, thus are not useful for identification.

3.3.2 Face diarization

Given the video shots, face diarization consists of (i) face detection, (ii) face tracking, and (iii) face clustering. Recall from the previous chapter, this system includes (i) a fast version of deformable part-based model (DPM) [31, 29, 78](ii) the CRF-based multi-target tracker [2], which relies on the unsupervised learning of time sensitive association costs for different features, and finally (iii), a face clusteringframework using matching and biometric similarity measures similarly to [19] with two improvements: shot-constrained face clustering (SCFC) and the use of total variability modeling (TVM).

[1] https://github.com/mit-nlp/MITIE

3.3.3 Speaker diarization

The speaker diarization system ("who speak when?") is based on the LIUM Speaker Diarization system [47], freely distributed[2]. This system has achieved the best or second best results in the speaker diarization task on REPERE French broadcast evaluation campaigns 2012 and 2013 [79].

The diarization system is first composed of an acoustic Bayesian Information Criterion (BIC)-based segmentation followed by a BIC-based hierarchical clustering. Each cluster represents a speaker and is modeled with a full covariance Gaussian. A Viterbi decoding re-segments the signal using GMMs with 8 diagonal components learned by EM-ML, for each cluster. Segmentation, clustering and decoding are performed with 12 MFCC+E, computed with a 10ms frame rate. Music and jingle regions are removed using a Viterbi decoding with 8 GMMs (trained on french broadcast news data) for music, jingle, silence, and speech (with wide/narrow band variants for the last two, and clean or noised or musical background variants for wideband speech).

In the above steps, features were used unnormalized in order to preserve information on the background environment, which may help differentiating between speakers. At this point however, each cluster contains the voice of only one speaker, but several clusters can be related to a same speaker. The background environment contribution must be removed from each GMM cluster, through feature gaussianization. Finally, the system is completed with clustering method based on the i-vectors paradigm and Integer Linear Programming (ILP). This new clustering method is fully described in [80] and [81]. The ILP clustering along with i-vectors speaker models gives better results than the usual hierarchical agglomerative clustering based on GMMs and cross-likelihood distances [82].

3.3.4 Identification and result ranking

After obtaining homogeneous clusters during which distinct identities speak or appear, one needs to assign each name output from NER module to the correct clusters. However, associating auditory voices with visual person clusters or names has two major difficulties. The visible person may not be the current speaker and the speaking person can be dubbed by a narrator in a different language. Although we have introduced a temporal learning method to solve the dubbing problem, incorporating it into an AV diarization system is still an open question. Because of these problems of AV association, we use a direct naming method [83] which finds the mapping between clusters and names to maximize the co-occurrences between them. This direct naming scheme is illustrated in Figure 3.8.

Naming. Names are propagated based on the outputs of face diarization and speaker diarization independently. The direct naming method is applied to speaker clusters to produce a mapping between names and clusters. All shots which overlap with the clusters are tagged with the corresponding names with equal confident scores. The same direct method is applied to face

[2] www-lium.univ-lemans.fr/en/content/liumspkdiarization

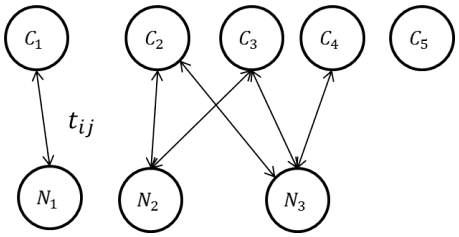


Figure 3.8 – Direct naming example. Given the face (or speech) clusters C_i and the names extracted from OCR N_j, we create an edge between a cluster and a name if their durations overlap. In this example, N_1 is assigned to C_1, C_5 will not have any name, and $C_{2..4}$ will be assigned names to maximize the total overlap scores t_{ij}.

clusters to produce a set of named clusters. Unlike speaker naming, for one shot, a name coming from face naming is ranked based on the talking score of the cluster's segment within that shot. The talking score is predicted using lip motion and temporal modeling with LSTM [9]. Based on the two results, we propose a strategy to appropriately combine them.

Ranking in MediaEval. In MediaEval challenge, beside retrieving the people appearing and talking during each shot, we also need to rank the names to compute mean average precision. Let $S = \{s_k\}$ be the list of testing shots. Within each shot, $\{N_i^F, t(N_i^F)\}$ is the set of names returned by face naming and the corresponding talking scores and $\{N_i^A, 1.0\}$ is the set of names returned by speaker naming, each is ranked equally with score 1.0. The names which the two methods agree on are ranked highest. Then, names from face naming are ranked higher than speaker naming because we found that face naming is more reliable in empirical experiments. Alternative strategies that rank speaker naming equal or higher than face naming gave inferior results. Our ranking strategy is described in Algo. 1.

3.4 Evaluation

3.4.1 Talking faces and dubbing detection

We describe below our experimental protocol and analysis of the talking face and dubbing detections results.

Experimental protocol

DW-Dubbing dataset[3]**.** We collect face tracks with their corresponding audio from Deutsche-Welle broadcast programs including debates and documentaries. Each track was divided into $2s$ segments. Segments with multiple arguing voices, inaudible speeches, or profile faces were

[3] http://www.idiap.ch/scientific-research/resources

Algorithm 1 Ranking names within shots. For each shot s_k the list of shots S, for each shot, we apply the face naming and speaker naming methods to acquire the names and scores $(N_i^F, t(N_i^F))$, $(N_j^A, 1.0)$ respectively. Then ranking is apply to return the final sorted list of names Q_{s_k}.

for $s_k \in S$ **do**
  $Q_{s_k} = \emptyset$
  Face_naming$(s_k) \Rightarrow (N_i^F, t(N_i^F))$
  Speaker_naming$(s_k) \Rightarrow (N_j^A, 1.0)$
  for each N_i^F **do**
    if $\exists N_j^A / N_j^A = N_i^F$ **then**
      $Q_{s_k} = Q_{s_k} \cup \{(N_i^F, t(N_i^F) + 2.0)\}$
    else
      $Q_{s_k} = Q_{s_k} \cup \{(N_i^F, t(N_i^F) + 1.0)\}$
  for each N_j^A **do**
    if not $\exists N_i^F / N_i^F = N_j^A$ **then**
      $Q_{s_k} = Q_{s_k} \cup \{(N_j^A, 1.0)\}$

	Training set	Testing set	Unsupervised set
Authentic	617	444	1598
Dubbing	440	209	0
Silence	406	237	771

Table 3.1 – Number of segments belonging to different splits and classes in the DW-dubbing dataset

discarded. The statistics of the dataset is shown in Tab. 3.1. Data from different videos were split into subsets used for unsupervised training, training and test data. The language of authentic speech/speaker segments was English, and dubbing segments taken from DW international documentaries had English voice dubbing a wide range of languages including Spanish, German, or other minority languages.

Protocol. For all models below, the authentic segments of the unsupervised set and of the training set were used to learn the PCA, CCA, and LSTM representations. Linear SVM classifiers were trained from the authentic and dubbing segments from the training set, using cross validation (CV) to determine hyperparameters. Evaluation was done on the test set, using accuracy as

	CV Acc.	Testing Acc.
MSD [48]	80.67	77.79
Mv [49]	78.92	82.16
HOF + SVM	78.39	79.06
HOF + LSTM	81.59	83.08

Table 3.2 – Talking face detection results.

		Testing		
	CV Acc.	Acc.	Prec.	Recall
Audio	67.50	76.92	96.31	72.08
PCA	91.01	79.91	97.03	73.65
CCA	74.58	81.80	89.64	83.78
PCA + LSTM	85.44	83.76	94.69	81.53
CCA + LSTM	86.36	88.03	95.78	86.79

Table 3.3 – Dubbing classification results on DW data.

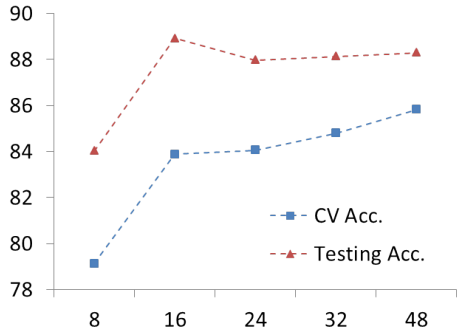


Figure 3.9 – Training and testing accuracies for different values of N_h for the CCA+LSTM model.

performance measures, along with recall and precision of authentic segments.

Models. To evaluate the contributions of the different elements, we tested several models:

1. **Audio.** This uses only MFCC features as the input for a SVM classifier.
2. **PCA.** It consists of applying another PCA on the concatenation of the PCA representation of each modality. Keeping 95% of the representation, we obtained a 75 dimension vector for each block of frames, which were averaged and used as input to a SVM classifier.
3. **CCA.** For each block, as shown in Fig. 3.3, the CCA projections (32 dimensions per modality) were computed, averaged and fed to a SVM.
4. **PCA+LSTM.** It consists of a LSTM with $N_h = 16$ applied to the multimodal PCA representation of the PCA baseline.
5. **CCA+LSTM.** A LSTM with $N_h = 16$ is trained with the CCA projection vectors of the CCA baseline.

Experimental Results

Talking faces. In a preliminary experiment, we trained a LSTM model to detect talking faces from optical flow histograms computed at every frame. As in dubbing, the LSTM was trained to predict the next frame observations, and the average hidden state was used as input to a silent-vs-speaking classifier. Results in Tab. 3.2 demonstrate the benefit of the temporal information over

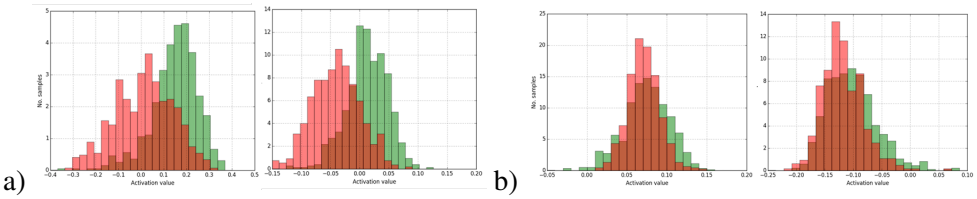


Figure 3.10 – Hidden neurons activation distributions. Green distributions are from the authentic samples, red ones from dubbing samples. a) discriminative neurons. b) non-discriminative neurons.

other baselines (see Sec. 3.2.1 for details of [48] and [49]).

Dubbing. Tab. 3.3 displays the obtained results. Because one can possibly distinguish dubbing cases based on languages or quality of voices in the audio, Audio only can give some positive results. However, using both streams in PCA slightly improves the accuracy, this signifies the importance of multimodal analysis in this task. Nevertheless, the joint PCA subspace computed by maximizing variance is not expressive enough and results in confusing class observations, the classifier cannot be well generalized for the test set. CCA learns a better space where high or low correlation are expected depending on the class, leading to more stable results. By modeling the temporal dynamics within segments rather than averaging, the hidden state representation extracted from LSTM better discriminates the two classes and boosts the performance of both types of input. In this view, CCA offers a more suitable space for LSTM predictions of normal speech, and LSTM trained on CCA inputs outperforms LSTM trained from PCA.

This is confirmed by visualizing the activation distribution of the hidden neurons (i.e. each dimension of the hidden state). Typical examples are illustrated in Fig. 3.10 (CCA+ LSTM with $N_h = 16$). The two left neurons fire stronger when the two streams are correlated (in green), and are inhibited otherwise (in red). Neurons on the right fire similarly regardless of the classes, suggesting that they are probably specialized to process single modality inputs, whereas left ones incorporate cross-modality information, thus contributing significantly to detecting asynchrony.

Finally, to explore the LSTM parameter space, we vary the hidden size N_h from 8 to 48. Results are shown in Fig. 3.9. As N_h increases, the cross validation training accuracy increases, but not the testing results. This shows that large hidden size can lead to overfitting on the training set.

3.4.2 Person discovery

Challenge task

In the MediaEval Person Discovery task, the goal is the following. Given the raw TV broadcasts, each shot must be automatically tagged with the name(s) of people who can be both seen as well as heard in the shot. The list of people is not known in prior and their names must be discovered in an unsupervised way from video text overlay or speech transcripts. This situation corresponds

	MAP@1	MAP@10	MAP@100
(1) Face naming + baseline OCR	30.3	22.0	21.0
(2) Face naming + our OCR	58.6	42.9	42.0
(3) Talking face naming + our OCR	64.2	53.1	52.1
(4) Talking face naming + speaker naming	68.3	56.2	54.7
(5) Fusion (4) with baselines	79.2	65.2	63.4

Table 3.4 – Benchmarking results of our submissions. Details of each submission in the text.

to cases where at the moment a content is created or broadcast, some of the appearing people are relatively unknown but may later on become a trending topic on social networks or search engines. In addition, to ensure high quality indexes, algorithms should also help human annotators double-check these indexes by providing an evidence of the claimed identity (especially for people who are not yet famous).

Datasets

The test set is divided into three datasets: INA, DW and 3-24. The INA dataset contains 2 TV video channels for a total duration of 90 hours. The DW dataset is composed of video downloaded from Deutsche Welle website, in English and German for a total duration of 50 hours. This dataset was provided by EUMSSI so that we can have benchmarking on actual data of our project. The last dataset contains 13 hours of broadcast from 3/24 Catalan TV news channel.

As the test set comes completely free of any annotation, it was annotated a posteriori based on participants' submissions and by participants themselves. Using the evidence provided by participants, an annotator can double-check the automatically-generated index. Two types of evidence are allowed coming from video OCR and automatic audio transcripts.

Evaluation

Participants are scored based on a set of queries. Each query is a person name in the corpus, each participant has to return all shots when that person appears and talks. The metric is Mean Average Precision (MAP) over all queries. In Tab. 3.4, we report our result on the test set as of 24/09/2016 [4]. Each of our 5 submissions (Sub.) is as following:

- Sub. (1) and Sub. (2) used our face naming without talking score with baseline OCR-NER (1) or with our OCR-NER (2).
- Sub. (3) used our face naming with talking score.
- Sub. (4) used the combination of talking face naming in sub. (3) with speaker naming.
- And sub. (5) used the combination of sub. (4) with other systems using baseline OCR-NER or baseline face diarization. This is also our primary submission.

[4]The groundtruth was been updated by a collaborative annotation until 20/10/2016.

	MAP@1	MAP@10	MAP@100
TokyoTech [87]	31.5	20.0	NA
UVigo [86]	31.5	23.6	21.1
HCMUS [85]	36.3	29.3	27.3
Baseline [73]	37.0	30.3	29.2
UPC [88]	63.0	50.5	48.4
PUC Minas and IRISA [84]	64.4	49.3	47.8
EUMSSI	79.2	65.2	63.4

Table 3.5 – Test result ranking of all participants.

System comparisons. When comparing sub. (1) and sub. (2), one can observe that our OCR-NER outperforms the baseline OCR-NER by a large margin. This may be contributed by the high recall of our system. Because the metric is averaged over all queries, any missing name can significantly decrease the overall MAP. On the other hand, false names are less problematic because of two reasons: they may not be associated with any clusters and they are not queried at all. In sub. (3), using talking face detection with LSTM, we can further improve by 5.6%. By combining face naming and speaker naming, we manage to increase the precision. This shows the potential for further research of better audio-visual naming. In our primary submission (5), the result are greatly boosted when other methods are added. From this we can note that these methods are complementary to each other and how to exploit their advantages is an open question in the future.

Comparison with the other participants. Table 3.5 showed the ranking of all teams participating in MediaEval 2016. Interestingly, EUMSSI was the only team with a dedicated talking face detection module. Meanwhile UPC systems achieved high result based on 2 factors: combining the results of two OCR systems and fusing face naming and speaker naming. As these factors are similar to our system, it further highlights our improvement in year 3 of talking face detection and dubbing detection within EUMSSI project. On the other hand, PUC Minas and IRISA [84] proposed a system with 2 improvements: a multimodal combination of face-speaker clustering face and name propagation methods based on minimum spanning tree and random walk. The improvement of their result shows the potential for us to further investigate in multimodal diarization and name propagation. Other teams (HCMUS [85], UVigo [86], TokyoTech [87]) all used only baseline OCR-NER provided by the organizers, which turned out to be detrimental because of the low recall of the results. This showcases both the importance of video OCR and NER for person identification in videos and the strong performance of the OCR-NER system developed in the EUMSSI project. We also present the result of the challenge in MediaEval 2015 in the Appendix B.

Figure 3.11 – Screenshot of one video entry in the EUMSSI interface.

3.5 EUMSSI Outcome

3.5.1 Online demonstration

We present the demonstration of the online platform of the EUMSSI project[5] which showcases an application of our system. To provide a concrete idea of where the AV processing is involved, Figure 3.11 depicts a screenshot of the EUMSSI web interface for a particular video. The interface displays the title of the video, offers to view it (clickable play logo), and then provides diverse meta-data either gathered from the original website (e.g. summary, publication data) or automatically extracted from diverse resources (text, audio, video), like named entities. The results of the AV processing tools are displayed, namely, face thumbnails associated to extracted face clusters (clickable to reach their corresponding appearance in the video), the main captions extracted from the video OCR, and more OCR text used for video indexing and temporal segment indexing (not visible to the user). In addition, person clusters whose name has been automatically detected from the OCR and associated to them are used to created browsing entries in the Amalia video browsing tool, see Fig. 3.12. In this browsing tool, a user can click on the interesting people to jump to the relevant segments.

3.5.2 Data processing and outcome

It is also important to highlight my contribution in extending and maintaining the processing toolchain as illustrated in Figure 3.13. This toolchain integrated all the components (OCR, ASR,

[5] http://demo.eumssi.eu/demo/

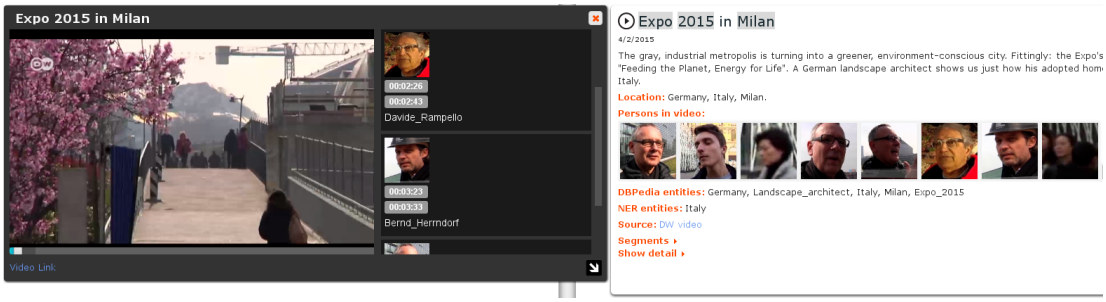


Figure 3.12 – Video browser indexed with person appearance temporal information.

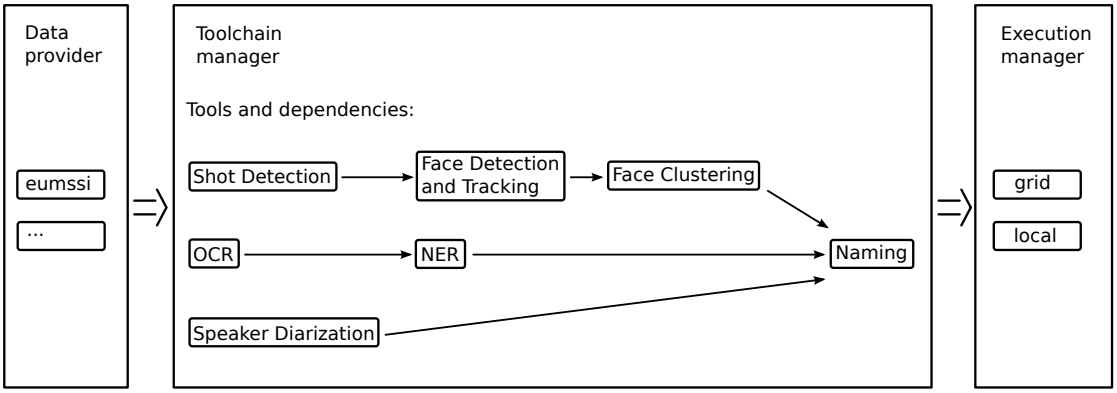


Figure 3.13 – Video processing toolchain. Data provider feeds video files and metadata to the toolchain manager. Tools are prepared based on desired configuration properties. They are and then given to the execution manager. Based on their dependencies, tools are executed on computational grid or on local workstation.

speaker diarization, face diarization, and naming) and is highly modular and easy to modify. Thanks to the toolchain and the signigicant improvement in processing speed, we had processed more than 14000 videos as of 31/10/2016 for the demonstration. The number of results is 2 times more than the number of OCR results and 20 times more than that of person identification results processed before I joined the project. Overall, the EUMSSI project was rated "Excellent" by the European Commission.

3.6 Conclusion

In this chapter, we have presented the integrate system for person discovery in TV broadcast. To this end, we have addressed talking face and dubbing detection in broadcast data, which involves detecting asynchrony between a visible speaker and the actual audio. We proposed an algorithm comprising a CCA step, to capture the correlation between the 2 modalities, and a LSTM to capture the joint evolution of audio and mouth features. Our DW-dubbing dataset available has been made publicly available.

Currently, our work is limited to TV broadcast. To detect more challenging dubbing situations,

semantic understanding of the asynchrony origin will be needed. The framework can be further improved with deep CCA or stacked LSTM for more discriminative feature extraction. We used only the visual features from the mouth region, thus leaving the visual attributes of the speakers such as: age, gender, or ethnicity. Incorporating these information can further enhance the correlation between the audio-visual streams.

To demonstrate the performance of our person naming system, submissions containing our recent advances in video processing and temporal modeling were benchmarked at the MediaEval challenge 2016. Each modality showed positive performance and we achieved the first place at the challenge. The EUMSSI project, which utilized our video processing and naming system, was also rated as "Excellent." One prospect of the person diarization system is that it creates a rich dataset of person tracks with face-speech correspondence. In Chapters 5 and 6, we explore the rich nature of this audio-visual person data in the context of cross domain transfer learning and weakly supervised learning.

4 Intra-Class Variance Regularization to Improve Speaker Embedding

4.1 Introdution

Learning speaker representations that can enable comparing speech utterances directly is crucial for multiple speaker related tasks in speech processing, including diarization, recognition, and verification. Recently, deep learning systems have achieved better benchmarking results than i-vectors in these speaker related tasks [89, 90, 91, 92, 6]. In these systems, a speaker embedding can be learned in two main ways. First, it can be extracted as the derivatives of the speaker recognition task by using the activation of the last layer before classification [92, 93, 94]. Second, it can be learned directly by optimizing the loss functions constraining the distances between same-speaker and different-speaker utterance pairs [91, 95, 5]. Among the distance-based losses, triplet loss has become more and more widely used in deep embedding networks [90, 91, 95].

The main idea of triplet loss is that the distance between a given pair of same-speaker utterances should be smaller than the distance from each of these utterances to any different-speaker utterance by a constant margin [96]. While this idea is attractive, learning with triplet loss can result in suboptimal performance in practice, especially in text-independent verification, where the content of speech is not predefined. The label information is not explicitly used in this loss function. Therefore, the model has to figure out the identity related factors that differentiate an utterance pairs besides the variation in content, accents, etc. This wide range of variation can lead to a dispersion of intra-class samples, thus rendering the embedding sensitive to noise. Furthermore, the number of triplets increases exponentially with the number of samples, which makes it hard to extract meaningful triplets to learn. Therefore learning with triplet loss can be slow to converge and can result in suboptimal performance. To overcome these challenges, one can employ effective sampling strategies [96, 97] or training embedding networks on top of pretrained classification models [6, 5].

In this chapter, we address the problem of training embedding networks with triplet loss by proposing a complementary loss function called intra-class loss. This loss acts as a regularizer that reduces the averaged intra-class distance variance of the final embedded features. The

effects of this loss is twofold. First, by reducing intra-class distance variance, the embedded features for each class are more compact and less sensitive to noise. Second, by minimizing the variation in utterances due to content or recording condition, the model can subsequently focus on differentiating utterances based on identities. Hence, using intra-class loss can help stabilize training and result in performance improvement. In practice, we optimize an equivalence of intra-class distance variance, which is the averaged pair-wise distance of same-speaker utterances. This upperbound can be efficiently estimated without parametrized means as in [98] and can be combined with triplet loss without expensive overhead cost.

To validate our contribution, experiments are conducted on two benchmark datasets for speaker verification: VoxCeleb and VoxForge. In both datasets, our method improves the overall accuracy and accelerates the training of embedding learning with triplet loss. Our results are also competitive with state-of-the-art systems.

The rest of the chapter is organized as follows: Section 4.2 reviews the literature in speaker recognition, Section 4.3 details our proposed intra-class loss, Section 4.4 presents the experiments, and Section 4.5 concludes the chapter.

4.2 Related Work

Below we discuss prior works on speaker embedding for recognition and verification as well as related work in computer vision which share similarities with our proposed method.

Conventionally, speaker representations are based on i-vectors [42]. To extract i-vectors, Baum-Welch statistics are computed from a Gaussian Mixture Model-Universal Background Model (GMM-UBM), which is learned using a sequence of feature vectors. I-vectors then can be used to compare utterances directly using cosine similarity or probabilistic linear discriminant analysis (PLDA) [99, 100, 101]. To improve upon i-vectors, deep neural networks (DNNs) have been first applied to gradually replace each step in computing i-vectors traditional speaker recognition systems [102, 103].

With the recent advances in deep learning, research effort has been devoted to learn end-to-end DNNs for speaker classification and verification. One common task is to learn a good speaker embedding to compare utterances, which can be addressed by two main types of approaches: learning a representation as a byproduct of classification or directly learning an embedding using distance based losses.

In the first approach, a DNN is trained to classify speakers and the activations of the final hidden layer are averaged over the utterance to create a ”d-vector” [93]. D-vectors can be enhanced by concatenating multiple levels of representation [92], PLDA scoring [89], and data augmentation [94]. The speaker embeddings extracted in this manner are not discriminatively trained and therefore often require an classifier such as PLDA or another DNN.

In the second approach, the scoring scheme is fixed as the distance between embedded features, thus the DNNs are optimized with distance-based loss to directly extract the embeddings. The distance-based loss can be contrastive loss [5] or triplet loss [96]. Especially, triplet loss has shown improvements in speaker turn detection [95], speaker diarization [104], and text-independent verification [90, 91]. The main idea is that the distance between same-speaker utterances should be smaller than the distance between different-speaker utterances. The challenge of this approach is the wide range of variation of text-independent utterances. It is hard for a network to distinguish the speaker related factors from other factors, which can lead to suboptimal results. Therefore, the network is often pretrained for classification task in advance to achieve good performance [91, 5]. Pretraining with classification uses the explicit identity labels to the network into speaker discriminative features, thus filtering other sources of variation.

In our work, we are interested in the problem of large variation in text-independent utterances. In deep face recognition, increasing intra-class compactness has been shown to improve the discrimination power of the activation features of the last hidden layer [98]. We follow the same idea but in the embedding space. Regularzing same-class neighbors has also been applied in [105]. In our work, instead of minimizing the distances to means [98] or the empirical positive pair-wise distances [105], we regularize the soft upperbound derived from the intra-class variance, which is the averaged intra-class distance.

4.3 Proposed Method

In this section, we first present the general framework to learn an embedding space with triplet loss and discuss its pros and cons to motivate our new loss function, which is described subsequently.

4.3.1 Triplet loss

Given a labeled training set of $\{(x_i, y_i)\}$, in which $x_i \in \mathbb{R}^D$, $y_i \in \{1, 2, .., K\}$, we define an embedding as $f(x) \in \mathbb{R}^h$, which maps an instance x into a h-dimensional Euclidean space. Additionally, this embedding is constrained to live on the h-dimensional hypersphere, *i.e.* $||f(x)||_2 = 1$. Within the hypersphere, we will simply use the Eulidean distance as the distance between 2 projected instances: $d(f(x_i), f(x_j)) = ||f(x_i) - f(x_j)||_2$

In this embedding space, we want the intra-class distances $d(f(x_i), f(x_j)), \forall x_i, x_j / y_i = y_j$ to be minimized and the inter-class distances $d(f(x_i), f(x_j)), \forall x_i, x_j / y_i \neq y_j$ to be maximized. To achieve such embedding, one method is to learn the projection that optimizes the triplet loss in the embedding space. Unlike other losses such as verification loss [106], triplet loss encourages a relative distance constraint. A triplet consists of 3 data points: (x_a, x_p, x_n) such that $y_a = y_p$ and $y_a \neq y_n$ and thus, we would like the 2 points (x_a, x_p) to be close together and the 2 points (x_a, x_n)

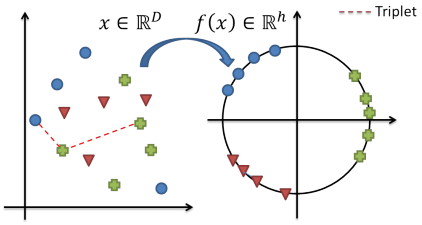


Figure 4.1 – Illustration of an embedding space. In this example, an embedding function f is learned to project the input samples in R^D into the embedding space R^h. In this embedding space, samples from the same class (with the same color) will have smaller distances than their distances with samples from a different class.

to be further away by a margin α in the embedding space [1]. Formally, a triplet must satisfy:

$$d(f(x_a), f(x_p)) + \alpha < d(f(x_a), f(x_n)), \forall (x_a, x_p, x_n) \in T \tag{4.1}$$

where T is the set of all possible triplets of the training set, and α is the margin enforced between the positive and negative pairs. Thus, the triplet loss to train a projection f is defined as:

$$\mathscr{L}(f) = \frac{1}{|T|} \sum_{(x_a, x_p, x_n) \in T} l(x_a, x_p, x_n, f) \tag{4.2}$$

in which

$$l(x_a, x_p, x_n; f) = [d(f(x_a), f(x_p)) - d(f(x_a), f(x_n)) + \alpha]_+ \tag{4.3}$$

with $[x]_+ = \max\{x, 0\}$.

Figure 4.1 shows an example of an embedding space, in which samples from different classes are separated. By choosing $h << D$, one can learn a projection to a space that is both distinctive and compact.

A major advantage of embedding learning is that the projection f is class independent. At test time, we can expect examples from a different class, or identity, to still satisfy the embedding goals. This makes embedding learning suitable for verification and clustering tasks.

We can observe in Eq. 4.2 that the parameters of f are updated based on the relative distance difference between the positive and negative pairs. Embedded features can be spread out to achieve the margin, thus making the representation sensitive to noise. On the other hand, two speech segments can be differentiated by not only the speaker identities but also by the content of speech, accents, etc. This large intra-class variation can make triplet loss result in low accuracy, especially when trained from scratch. Our intra-class loss is proposed in the next section to

[1] The value of α varies depending on the particular loss function to optimize We use one value of $\alpha = 0.2$ in all cases.

address these problems.

4.3.2 Reducing intra-class variance in the embedding space

Let $S_c = \{(x_i, y_i)\}$ be the set of samples from the class c. We want to minimize the intra-class distance variance of c:

$$\min_f \sum_{x_i / y_i = c} \frac{d(f(x_i), \mu_c)^2}{n_c} \tag{4.4}$$

in which $n_c = |S_c|$ and the mean of class c features is $\mu_c = \sum_{x_i / y_i = c} \frac{f(x_i)}{n_c}$. Eq. 4.4 requires estimating the mean μ_c, which changes with each update. To address this problem, a possibility is to compute a moving average of μ_c, but this can be unreliable during early training stage and requires a hyperparameter to tune. To circumvent this issue, we instead minimize an upperbound of the variance, which uses the pair-wise squared distances within the class. This upperbound can be derived as follows:

$$\begin{aligned}\sum_{x_i / y_i = c} \frac{d(f(x_i), \mu_c)^2}{n_c} &= \sum_{x_i / y_i = c} \frac{||f(x_i) - \sum_{x_j} \frac{f(x_j)}{n_c}||_2^2}{n_c} \\ &= \sum_{x_i} \frac{||\sum_{x_j} (f(x_i) - f(x_j))||_2^2}{n_c^3} \leq \sum_{x_i, x_j} \frac{||f(x_i) - f(x_j)||_2^2}{n_c^3}\end{aligned} \tag{4.5}$$

One can observe that minimizing Eq. 4.5 can lead to a trivial solution when all samples are projected to a single point. This can encourage model collapse when training with triplet loss [96]. Hence, we optimize the squared root of Eq. 4.5 and devise a second upperbound:

$$\sqrt{\sum_{x_i, x_j} \frac{||f(x_i) - f(x_j)||_2^2}{n_c^3}} \leq \sum_{x_i, x_j} \frac{\sqrt{||f(x_i) - f(x_j)||_2^2}}{n_c \sqrt{n_c}} = \sum_{x_i, x_j / y_i = y_j = c} \frac{d(f(x_i), f(x_j))}{n_c \sqrt{n_c}} \tag{4.6}$$

In Eq. 4.6, the objective is based on the true distance instead of the squared distance, which makes the loss more stable to noise and model collapse [97]. Also, we propose a soft constraint that only requires each pair-wise distance to be smaller than a threshold β. In practice, because n_c is constant across minibatches, we choose the denominator to be n_c^2, thus formulating the loss

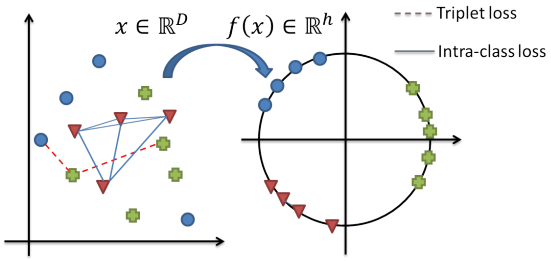


Figure 4.2 – Illustration of triplet loss and intra-class loss.

as the soft averaged pair-wise distance. Concretely, our intra-class loss function becomes:

$$\mathscr{L}_c(c) = \sum_{x_i, x_j / y_i = y_j = c} \frac{[d(f(x_i), f(x_j)) - \beta]_+}{n_c^2} \tag{4.7}$$

This new intra-class loss can be weighted by λ to be combined with the triplet loss in Eq. 4.2 (as illutrasted in Fig. 4.2) to form the final loss function:

$$\mathscr{L} = \mathscr{L}_t + \frac{\lambda}{K} \sum_c \mathscr{L}_c(c) \tag{4.8}$$

Using this intra-class loss as a regularizer has 2 main effects. Firstly, it prevents features to disperse in the embedding space, thus making the representation more robust to noise. Secondly, minimizing variance can reduce the influence of other factors such as speech content or recording conditions. Therefore, the learned model is more discriminative with respect to speaker identities. We also note that the distances calculated in intra-class loss can be effectively reused from triplet loss, thus reducing the overhead of adding a new loss function.

4.4 Experiments

We first describe the datasets and implementation details before discussing the experiments and the results.

Data and metrics

VoxCeleb. This dataset contains videos of celebrities collected from Youtube [5]. There are more than 140K utterances of 1251 speakers in a free context. 40 speakers are reserved as test data for the verification protocol. We report Equal Error Rate (EER) computed using the provided trial

Table 4.1 – ResNet architecture used in the experiments. Residual block follows the same definition in [4]. Each convolution layer is followed by ReLU and batch normalization.

Layer	# filt.	Stride
Conv 5×5	64	2×2
Max Pool 3×1	-	2×1
Res. block	64	2×2
Res. block	128	2×2
Res. block	256	2×2
Conv 1×9	256	1×1
Conv 1×9	512	1×1
Stats Pool $n \times 1$	-	1×1
L_2 norm	-	-

pairs.

VoxForge. This is an open source speech database, where speakers voluntarily contribute speech data for development of open resource speech recognition systems [2]. The utterances have lower variability as the text is read and the data is collected in a clean environment. We follow the same protocol as in [6]. From 300 chosen speakers, three subsets of 100 speakers are constructed for training, development, and evaluation. The training set is used to train / finetune embedding networks. The development set is used to choose a threshold based on EER, and the threshold is applied on the evaluation set to report Half Total Error Rate (HTER).

Implementation Details

CNN architecture. Our model is built using the ResNet architecture[4]. There are 31 layers configured as in Tab. 4.1. The key modification is the statistical pooling layer, which concatenates both mean and standard deviation of the previous layer across the whole sequence in time. We also change the configuration of the first max pooling layer to work only on the time domain.

Feature extraction. For each utterance, a spectrogram is computed using 512-point FFT, a temporal window of 25ms, and a window shift of 10ms. Mean and variance normalization on each frequency bin is performed as in [5].

Training details. All networks are trained using RMSProp optimizer [107] with a 10^{-3} learning rate. Each minibatch contains 120 samples, and negative triplets are sampled using distance-based sampling method [97]. We train with truncated utterances of 2 seconds or 3 seconds as input. For hyperparameters, we choose $\alpha = 0.2$, $\beta = 0.2$, and $\lambda = 0.001$.

[2] http://www.voxforge.org/

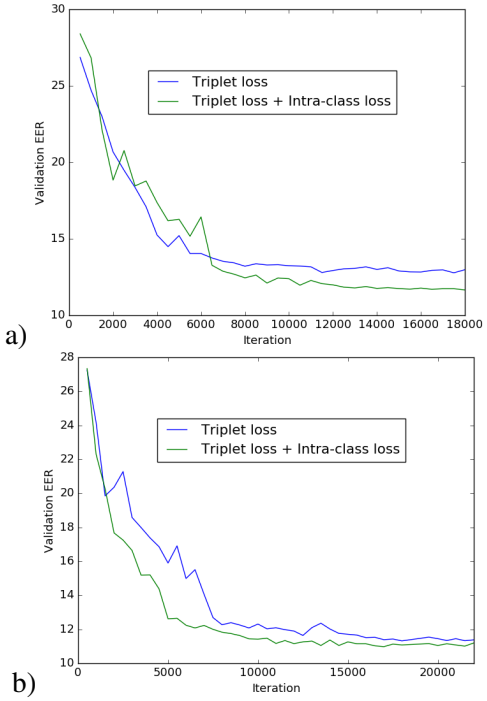


Figure 4.3 – EER on the validation set of VoxCeleb during training with training samples of different lengths: (a) 2s or (b) 3s.

Experimental Results

Training from scratch. In this setting, a ResNet is initialized randomly and then learned on the VoxCeleb training set using either triplet loss alone or in combination with intra-class loss.

In Fig. 4.3, we visualize the EER on the VoxCeleb validation set as the model training progress. One can observe that intra-loss accelerates the training speed. The model not only converges faster but also to a lower EER. In Tab. 4.2, EERs on the validation and test sets with different utterance input lengths are shown. Intra-class loss substantially improves the overall performance of the deep model. The EER is reduced relatively by 14% for 2s-segment input and 7% for 3s-segment input. Overall, 2s-segment input yields worse EER in comparison to using 3s. However, it is important to note that when intra-loss is added, the model learned with 2s-segment input can still reach the same performance as in using 3s-segment. This shows intra-class loss can enhance the embedding space even when the input signals contain less information.

Embedding learning from a pretrained model. In this experiment, a ResNet for speaker recognition is first trained with softmax loss using the speakers in the VoxCeleb training set. Then the convolutional weights are frozen and the last embedding layer is trained with the embedding losses.

When using the activation of the last hidden layer of the pretrained models, one can achieve

Table 4.2 – Ablation study of how using intra-class loss effect the EER on the validation and test set of VoxCeleb. We also compare how results differ when the training utterances are truncated to 2s or 3s.

Setting	In. len.	Loss	Val. EER	Test EER
Scratch	2s	Trip.	12.73	12.44
		Trip. + Intra.	11.71	10.74
	3s	Trip.	11.17	10.68
		Trip. + Intra.	10.31	9.93
Pretrained	2s	Softmax	-	14.43
		Trip.	7.21	8.31
		Trip. + Intra.	6.30	7.97
	3s	Softmax	-	11.96
		Trip.	6.84	8.20
		Trip. + Intra.	6.03	8.12

Table 4.3 – Comparison of our embedding method to other state-of-the-arts on VoxCeleb dataset. (*are reported in [5])

GMM-UBM*	15.0
i-vector + PLDA*	8.8
Bi-LSTM Embedding [95]	14.1
CNN Embedding [5]	7.8
Ours (Pretrained + Intra.)	7.97

14.43% and 11.96% EER on the test set using input of 2s or 3s respectively. As the models were pretrained to predict explicitly the identities, they can focus more on the discriminative features for classification. Therefore, training an embedding layer on top of these models can significantly enhance the results. As the initial model is already well-trained, both cases of with and without intra-class loss yield statistically similar EERs.

In Tab. 4.3, we compare our method with state-of-the-art systems. Our embedding network with intra-class loss outperforms traditional methods using factor analysis with GMM-UBM. When comparing with other embedding methods, one can see that bidirectional LSTM trained with triplet loss [95] cannot capture the discriminative variation of the data well. Meanwhile, our systems perform on par with [5], which uses pretrained classification model and contrastive loss for embedding learning. This agrees with the conclusion from [97] that shows similar performance between contrastive loss and triplet loss.

Verification task on VoxForge. In this experiment, we use the pretrained classification network from VoxCeleb and the embedding layer is learned using either the VoxCeleb or the VoxForge training sets and report test results on the VoxForge evaluation set. In evaluation stage, all distances from a probe utterance to every enrollment utterance is computed and the identity is simply decided based on a threshold. The development set is used to set the threshold with lowest

Table 4.4 – Verification result of our embedding method comparing to other state-of-the-arts on VoxForge dataset. (*are reported in [6]). In the left column, 'VoxCeleb' means a model was trained entirely on VoxCeleb and 'VoxForge' means a model was pretrained on VoxCeleb and finetuned on VoxForge.

VoxCeleb	Triplet loss	2.09
	Triplet + Intra-class loss	1.50
VoxForge	Triplet loss	1.69
	Triplet + Intra-class loss	1.16
	GMM-UBM*	3.05
	i-vector + PLDA*	5.87
	ISV*	2.40
	CNN Clas. [6]	1.20

EER. HTER is reported on the evaluation set using this threshold.

Tab. 4.4 shows our ablation results together with other methods. Comparing our models when using intra-class loss against using triplet loss only, we can observe a significant relative reduction of 30% in EER in both cases of training sets. Interestingly, the model trained with intra-class loss on only out-domain data (VoxCeleb) can still perform better than the model finetuned with only triplet loss on in-domain data (VoxForge). The improvement shows that intra-class loss can help adapting models to new datasets. This can be explained as the variance of each class is regularized, the learned embedded features are less sensitive to noises which are not present in the original dataset.

When comparing to other methods on this dataset, our deep embedding models are better than traditional factor analysis systems. Using intra-class loss, our model can work slightly better than the deep method that uses classification CNNs to model specific speakers [6]. It is important to note that in our system, we do not build a specific model for each speaker using their enrollment data. Only the distances from a probe utterance to all enrollment data are used to verify directly. This advantage allows our system to be used when there is no enrollment phase, for example in the setting of speaker verification in the wild.

4.5 Conclusion

We have presented a novel loss function as an supportive learning goal to improve the speaker embedding spaces learned by deep neural networks. By reducing the averaged intra-class pair-wise distances, our loss aims to increase the robustness of learned features. The results of speaker verification task on two public datasets, VoxCeleb and VoxForge, validate the improvement of our approach. Models learned with intra-class loss not only converge faster but also achieve better accuracy. However, these effects are only limited to text-independent speaker verification and learning embedding from scratch without pretraining. In the future, more experiments with different strategies for reducing intra-class variance such as using moving averaged class

means [98] or using embedding margin based loss [97] can be conducted.

5 Improving speech embedding by transfer learning with visual data

5.1 Introduction

Learning speaker turn representation is the fundamental problem to enable comparing or clustering speech segments for multimedia indexing or interactive dialogue analysis. State-of-the-art Gaussian-based speaker diarization methods have been shown to be successful in various types of content such as radio, TV broadcast news, telephone conversation and meetings [108, 109, 110]. In these contents, the speech signal is mostly prepared speech and clean audio, the number of speakers is limited, and the duration of speaker turn (i.e. a speech segment of one speaker) is more than 2 seconds on average. When these conditions are not valid, in particular the assumption of speaker turn duration, the quality of speaker diarization deteriorates [111]. As shown in TV series or movies, state-of-the-art approaches do not perform well [112, 113] when there are many speakers (from 28 to 48 speakers), or speaker turns are spontaneous and short (1.6 seconds on average in the Game of Thrones TV series).

To alleviate these shortcomings of speaker diarization, research has been conducted along two fronts: better methods to learn speaker turn embeddings or utilizing the multimodal nature of video content. For instance, the recent work on speaker turn embedding using triplet loss shows certain improvements [95, 114, 13], where as other multimodal related works focus on late fusion of two streams by propagating labels [115, 116] or high level information such as distances or overlapping duration [51, 84].

5.1.1 Motivation

In this chapter, we combine the two fronts of embedding learning and multimodal processing by investigating crossmodal transfer learning approaches to improve directly a speaker turn embedding using a face embedding . An overview of our framework is illustrated in Figure 5.1. First, on the visual side, we rely on the state-of-the-art advances in deep face embedding [117, 96]. Indeed recently, learning face embeddings has made significant achievements in all tasks, including recognition, verification, and clustering [118, 119, 96, 117, 120]. On the acoustic side,

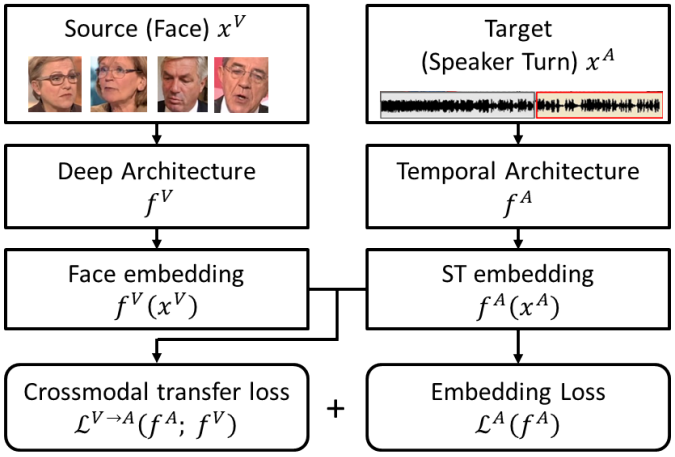


Figure 5.1 – Overview of our proposed method. Face embedding model is pretrained and used to guide the training of speaker turn embedding model through crossmodal transfer loss $\mathscr{L}^{V\to A}(f^A; f^V)$. Speaker turn embedding is trained with the combination of the embedding loss $\mathscr{L}^A(f^A)$ and the crossmodal transfer loss.

we exploit the deep architecture to learn a speaker turn embedding with triplet loss (*TristouNet*) of [95], which achieved improvement on short utterances. By projecting both acoustic signals and face images into a common hypersphere, one can unify the two embedding spaces, thus enabling the knowledge to be shared across modalities. The discrepancy between the two domains is formulated as an added regularizing term which measures differences between the two embedding spaces.

Our motivation for crossmodal transfer learning and adaptation is twofold. First, we can point to the difference in training data of two modalities. There are hundreds of thousands images from thousands identities in any standard face dataset. However, collecting labeled speech data is more challenging[1] because we cannot use Internet search engines similarly to face images in [117, 106]. Also, manual labeling speech segments is much more costly or the labels have to be obtained indirectly from visual modules [5]. Thus, we aim at mitigating the need for massive datasets and take advantage of pretrained face embeddings through transfer learning and domain adaptation.

Second, we can observe that although one cannot find the exact voice of a person given only a face, when given a small set of voice candidates, it is possible to pick a voice which is more likely to come from the given face than others. This means that there are shared commonalities between the two embedding spaces such as age, gender, or ethnicity; or in other words, if a group of people share common facial traits, we expect their voices to also share common acoustic features. Thus, there are latent attributes which are shared between the two modalities.

[1] In 2018, a large dataset VoxCeleb2 for speaker recognition with 7000+ speakers was released [121]

5.1.2 Our approach and main contributions

Rather than relying on multimodal data with explicit shared labels such as genders, ages, or accent and ethnicity, we want to discover the latent commonalities from the source domain, a face embedding, and transfer them to the target domain, a speaker turn embedding. We hypothesize that these latent attributes can be related to the geometry of the space or to the underlying distributions of features. Therefore, by transferring properties of the source embedding feature space (*i.e.* face embedding) onto the target embedding feature space (*i.e.* speaker turn embedding), we can improve the performance.

Because different properties can be used as constraints to be transferred, we investigate 4 different strategies. Out of these, 3 strategies aim at transferring spatial constraints of the embedding at different levels of granularity. Meanwhile, the fourth strategy focuses on the distributions of multimodal features. More precisely, they are:

- Target embedding transfer: We are given the identity correspondences between the 2 modalities. Hence, given the 2 inputs from the same identity, one can force the desired embedded features of the speaker turn to be close to embedded features of the face. Minimizing the disparity between the 2 embedding spaces with respect to identity will act as a regularizing term for optimizing the speaker turn embedding.
- Relative distance transfer: One can argue that exact similar location in the embedding spaces is hard to achieve given the fuzzy relationship between the 2 modalities. It may be sufficient to only enforce relative order between identities. Therefore, this approach constrains that 2 people who look more similar will have more similar voices.
- Clustering structure transfer: This approach focuses on discovering shared commonalities between the 2 embedding spaces such as age, gender, or ethnicity. If a group of people share common facial traits, we expect their voices to also share common acoustic features. In particular, the shared common traits in our case is expressed as belonging to the same cluster of identities in the face embedding space.
- Maximum mean discrepancy: This approach is different in nature from the previous ones, following the hypothesis that the crossmodal commonalities can be expressed as the discrepancy between the distributions of the two embedded features. In other words, by minimizing the difference between the distribution of speech features and the distribution of the visual features, one can achieve a better speech embedding. We use maximum mean discrepancy, which is the statistical measure of difference between 2 distributions [122], as the regularizing term.

Experiments conducted on 2 public datasets REPERE and ETAPE show significant improvement over the competitive baselines, especially when dealing with short utterances. Our results also show that by transferring knowledge from the visual domain, one can still learn competitive speaker turn embeddings when there are limited data. Our contributions are also supported by crossmodal retrieval experiments and the visualization of our intuition.

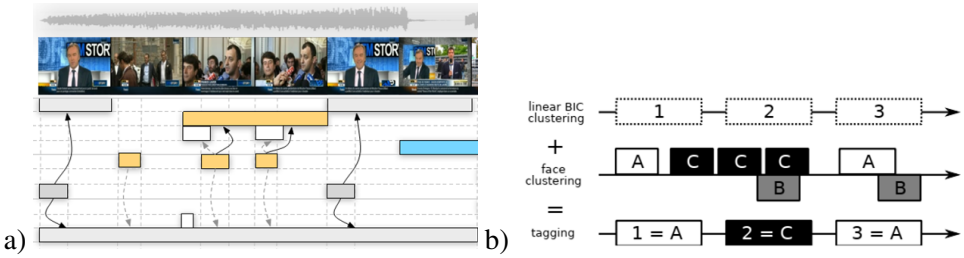


Figure 5.2 – Examples of late fusion systems. (a) formulated the joint clustering problem in a CRF framework with the acoustic distance and the face representation distance as pair-wise potential functions [51] while (b) used face clustering labels to classify the results from speaker turn segmentation [116]

The organization of the chapter is as follows: Section 5.2 reviews other works related to audio-visual learning, Section 5.3 introduces embedding learning with triplet loss and the architecture that we use in our work, Section 5.4 describes our transfer learning methods in details, Section 5.5 presents the experimental results, and Section 5.6 closes the chapter with further discussion.

5.2 Related Work

Below we discuss prior works on audio-visual person recognition and transfer learning which share similarities with our proposed methods.

As person analysis tasks in multimedia content such as diarization or recognition are multimodal by nature, significant effort has been devoted to using one modality to improve another. Several works exploit labels from the modality that has superior performance to correct the other modality. In TV news, as detecting speaker changes produces a smaller false alarm rate and less noise than detecting and clustering faces, speaker diarization hypothesis is used to constrain face clustering, *i.e.* talking faces with different voice labels should not have the same name [115]. Meanwhile in [116], because face clustering outperforms speaker diarization in TV series, labels of face clusters are propagated to the corresponding speaker turns. Another approach is to perform clustering jointly in the audio-visual domain. [84] linearly combines the acoustic distance and the face representation distance of speaking tracks to perform graph-based optimization; while [51] formulates the joint clustering problem in a Conditional Random Field framework with the acoustic distance and the face representation distance as pair-wise potential functions. Examples of these works are illustrated in Figure 5.2.

Beside late fusion of labels, early fusion of features has been proposed, including using deep neural networks [63, 64] (shown in Figure 5.3). However, it is only suitable for supervised tasks and has only been tested on limited datasets limited with 6 identities. Note that the aforementioned works focus on aggregating two streams of information whereas we emphasize on the transfer of knowledge from one embedding space to another.

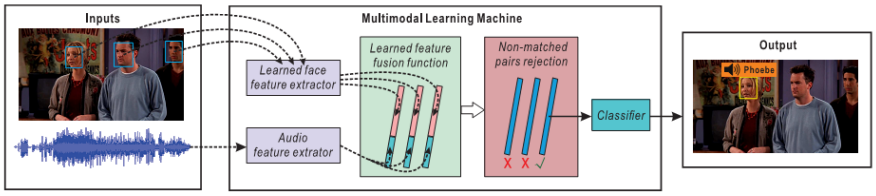


Figure 5.3 – A related work by [63] that focused on aggregating two streams of information whereas we emphasize on the transfer of knowledge from one embedding space to another.

By applying recent advances in embedding learning, with deep networks for face [117, 96] and speaker turn [95] our goal is not only to improve the target task (as speaker turn embedding in our case) but also provide a unified way for multimodal combination.

Each of our three geometric-based approaches draws inspiration from a different line of research in transfer learning. First, we can point to coupled matching of image-text and heterogeneous recognition [123, 124, 125] or harmonic embedding [96] as related background for our target embedding transfer. These works focus on learning the direct mapping between an item in one domain to one item in the other. Since it is arguable that audio-visual identities contain less correlated information, our method uses the one-one correspondences as a regularization term rather than as a main loss to optimize. Meanwhile, as the learning target is a Euclidean embedding space in both modalities, relative distance transfer is inspired by metric imitation [126] or multi-tasks metric learning [127]. In our work, the triangular relationship is transferred across modalities instead of neighbourhood structure [126] or across tasks of the same modality [127]. Finally, as one identity is enforced to have the same neighbors in both face embedding and speech embedding spaces, our clustering structure transfer is therefore closely related to transfer learning through projection ensemble [128]. Although co-clustering information and cluster correspondence inference have also been used in transfer learning on traditional tasks of text mining [129, 130], we are first to expand that concept into exploiting clustering structure of person identities for crossmodal learning.

Unlike the previous 3 transferring methods which emphasize on the geometric properties of the embedding spaces, the domain adaptation approach relies on the underlying distributions of the features within the embedding spaces. A popular method in visual domain adaptation [131, 132, 133] is to minimize the maximum mean discrepancy (MMD) loss between the 2 feature sets. Maximum mean discrepancy (MMD) loss, proposed by [122], is a non-parametric approach to compare distributions. Intuitively, the 2 feature sets are projected into the kernel space and the distance between the means of the 2 distributions in this kernel space is used as the measurement of discrepancy. This is an interesting approach since non-parametric representations are well-suited for representing complex multimodal data in high-dimensional spaces. Our work is the first attempt in unifying the audio and visual domains into a single feature space which shares the commonalities by minimizing MMD loss between feature distributions in 2 embedding spaces.

5.3 Preliminaries

This section first briefly recall the concept of learning embedding and the triplet loss from previous chapter. Then we review the TristouNet architecture, which is used as the main architecture for learning speech embedding within this chapter.

5.3.1 Embedding Learning with Triplet Loss

Recall from the previous chapter, a triplet consists of 3 data points: an anchor point x_a, a positive point x_p, and a negative point x_n such that $y_a = y_p$ and $y_a \neq y_n$. Following the embedding goal, we would like the 2 points (x_a, x_p) to be close together and the 2 points (x_a, x_n) to be further away by a margin α in the embedding space. Formally, we define the triplet loss to be minimized as:

$$L_t = \frac{1}{|T|} \sum [d(f(x_a), f(x_p)) - d(f(x_a), f(x_n)) + \alpha]_+ \tag{5.1}$$

where T is the set of all possible triplets of the training set and d is the Euclidean distance in the embedding space.

In spite of its advantages, the triplet loss training is empirical and depends on the training data, the initialization, and triplet sampling methods. For a certain set of training samples, there can be an exponential number of possible solutions that yield the same training loss. One approach to guarantee good performance is to make sure that the training data come from the same distribution of the test data (as in [117]). Another solution for the projection to work in more general unseen cases may be to gather a massive training dataset with more data (as FaceNet was trained with 100-200 millions images of 8 millions of identities [96]). Although it is possible to gather such a large scale dataset for visual information, it is less the case for acoustic data. This explains why speaker turn embedding *TristouNet* only gains slight improvement over Gaussian-based methods [95]. To alleviate the data concern, we tackle the problem of embedding learning from the multimodal point of view. By using a superior face embedding network that was trained on a face dataset with the same identities as in the acoustic dataset, we can regularize the speaker embedding space and thus guide the training process to a better minima.

5.3.2 Learning speaker turn embedding with triplet loss

The fundamental task we are interested in is to learn a good speaker turn embedding so that given 2 speaker segments, one can compare them directly for verification or clustering. To this end, one can employ different architectures with different input representations such as time delay neural networks [92] or convolutional neural networks on spectrograms [5]. In this work, we use the architecture proposed by [95], which learns speech emebedding using triplet loss, for compatibility with our visual models.

The TristouNet architecture is illustrated in Figure 5.4. Firstly, a bidirectional Long Short-Term Memory (LSTM) recurrent networks [134] receives the input sequence x^A to produce the hidden forward and backward outputs. Secondly, average pooling is then applied along the temporal axis of each output yielding 2 fixed length vectors. These 2 forward and backward vectors are then concatenated as input for the fully connected layers for projection into higher dimensional space. Finally, the output are L^2-normalized into the Euclidean hypersphere.

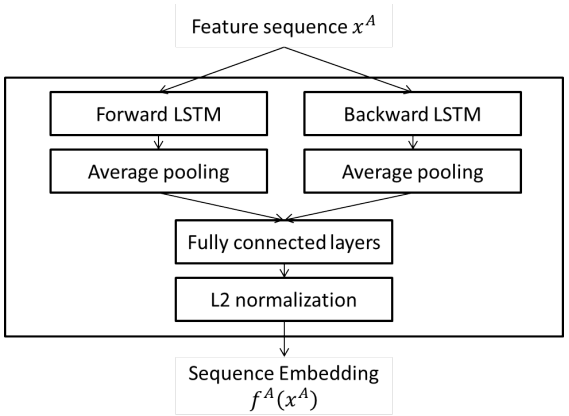


Figure 5.4 – TristouNet architecture

In spite of its advantages, the triplet loss training is empirical and depends on the training data, the initialization, and triplet sampling methods. For a certain set of training samples, there can be an exponential number of possible solutions that yield the same training loss. One approach to guarantee good performance is to make sure that the training data come from the same distribution of the test data (as in [117]). Another solution for the projection to work in more general unseen cases may be to gather a massive training dataset with more data (as FaceNet was trained with 100-200 millions images of 8 millions of identities [96]). Although it is possible to gather such a large scale dataset for visual information, it is less the case for acoustic data. This explains why speaker turn embedding *TristouNet* only gains slight improvement over Gaussian-based methods [95]. To alleviate the data concern, we tackle the problem of embedding learning from the multimodal point of view. By using a superior face embedding network that was trained on a face dataset with the same identities as in the acoustic dataset, we can regularize the speaker embedding space and thus guide the training process to a better minima.

5.4 Crossmodal transfer learning

In this section, we will expand the embedding learning concept into multimodal data to learn different feature embedding spaces. Then we will in turn describe our transfer learning methods to use one embedding space to improve the other.

In audio-visual (or multimodal data in general) settings, data contain 2 corresponding streams $\{(x_i^A, x_i^V, y_i)\}$. If the learning process is applied independently to each modality, we can learn

2 projections f_A and f_V into 2 embedding spaces $\mathbb{R}^{d_A}$ and $\mathbb{R}^{d_V}$ following their own respective losses:

$$\mathscr{L}^A(f^A) = \frac{1}{|T^A|} \sum_{(x_a^A, x_p^A, x_n^A) \in T^A} l(x_a^A, x_p^A, x_n^A; f^A) \tag{5.2}$$

and

$$\mathscr{L}^V(f^V) = \frac{1}{|T^V|} \sum_{(x_a^V, x_p^V, x_n^V) \in T^V} l(x_a^V, x_p^V, x_n^V; f^V) \tag{5.3}$$

in which $\mathscr{L}^A$ and $\mathscr{L}^V$ are defined from the general embedding loss Eq. 4.2 to speaker turn embedding and face embedding.

As shown in the experiments, f^V can already achieve a significantly better accuracy than the counterpart in acoustic domain, therefore our goal is to transfer the knowledge from face embedding to the speaker turn embedding. Hence, we assume that f^V is already trained with Eq. 5.3 using the corresponding face dataset (as well as optional external data). Using f_V, an auxiliary term $\mathscr{L}^{V \to A}(f^A)$ is defined to regularize the relationship between voices and faces from the same identity in addition to the loss function used to train speaker turn embedding in Eq. 4.2. Formally, the final loss function can be written as:

$$\mathscr{L}(f^A) = \mathscr{L}^A(f^A) + \lambda \mathscr{L}^{V \to A}(f^A) \tag{5.4}$$

The transfer loss $\mathscr{L}^{V \to A}(f^A)$ depends on what type of knowledge is transferred across modalities. λ is a constant hyper-parameter chosen through experiments specifically for each transfer type. In the following sections, different types of $\mathscr{L}^{V \to A}(f^A)$ will be described in details.

5.4.1 Target embedding transfer

Assuming that f^A projects x_i^A into the same hypersphere as $f^V(x_j^V)$, one can observe that by enforcing $f^A(x_i^A)$ to be in close proximity of $f^V(x_j^V)$ when $y_i = y_j$, f^A could achieve a similar training loss as f^V. In that case, the regularizing term in Eq. 5.4 can be defined as the disparity between crossmodal instances of the same identity:

$$\mathscr{L}^{V \to A}(f^A) = \sum_{(x_i^A, x_j^V) / y_i = y_j} d(f^A(x_i^A), f^V(x_j^V)) \tag{5.5}$$

The goal of Eq. 5.5 is to minimize intra-class distances by binding embedded speaker turns and embedded faces within the same class similarly to coupled multimodal projection methods [123, 125]. In this work, we extend this goal further by adopting the multimodal triplet paradigm to jointly minimize intra-class distances and maximize inter-class distances.

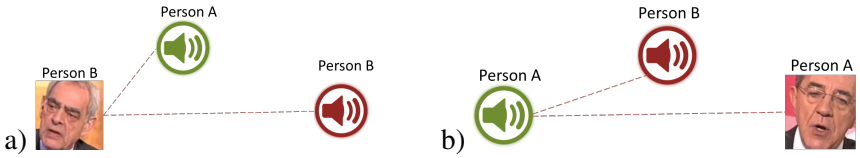


Figure 5.5 – Examples of multimodal triplets in target embedding transfer. Each triplet consists of mixing samples from both modalities. (a) (V, A, A) triplet where the anchor comes from visual domain. (b) (A, V, A) triplet where the positive comes from visual domain.

Algorithm 2 Target embedding transfer triplet set. Given the audio-visual input $\{(x_i^A, x_i^V, y_i)\}_{i=1..N}$ and the corresponding embeddings f^A, f^V, we mixed the inputs from 2 domain to collect the set of multimodal triplets T_{tar}.

Input f^A, f^V, $Q_{A,V}$, $\{(x_i^A, x_i^V, y_i)\}_{i=1..N}$
$T_{tar} = \emptyset$
for $\forall (a, p, n) / y_a = y_p \wedge y_a \neq y_n$ **do**
  for $m_a, m_p, m_n \in \{Q_{A,V}\}$ **do**
    $d_{a,p} = d(f^{m_a}(x_a^{m_a}), f^{m_p}(x_p^{m_p}))$
    $d_{a,n} = d(f^{m_a}(x_a^{m_a}), f^{m_n}(x_n^{m_n}))$
    if $d_{a,p} + \alpha > d_{a,n}$ **then**
      $T_{tar} = T_{tar} \cup (a, p, n)$
Output T_{tar}

Multimodal triplet loss. In addition to minimizing the audio triplet loss of Eq. 5.2, we also want two embedded instances to be close if they come from the same identity, regardless of the modality they comes from, and to be far from embedded instances of all other identities in both modalities as well. Concretely, the regularizing term is thus defined as the triplet loss over multimodal triplets:

$$\mathscr{L}^{V \to A}(f^A) = \frac{1}{|T_{tar}|} \sum_{(x_a^{m_a}, x_p^{m_p}, x_n^{m_n}) \in T_{tar}} l(x_a^{m_a}, x_p^{m_p}, x_n^{m_n}; f^A, f^V) \tag{5.6}$$

where $m_\bullet$ is the modality associated with the sample $x_\bullet^{m_\bullet}$, and the loss l is adapted from Eq. 4.3 by using the embedding appropriate to each sample modality. The set T_{tar} denotes all useful and valid cross-modal triplets, i.e. with the positive sample to be of the same identity of the anchor ($y_a = y_p$), and the negative sample to be from another identity ($y_a \neq y_n$); and with $(m_a, m_p, m_n) \in Q_{A,V}$, the set of valid modalities (all combinations except (V, V, V), (V, V, A), and (A, A, A) already considered in the primary loss of Eq. 5.2). In Figure 5.5, 2 examples of (V, A, A) and (A, V, A) are shown. For instance, if $(m_a, m_p, m_n) = (A, V, V)$, the loss will foster the decrease of the intra-class distance between $f^A(x_a^A)$ and $f^V(x_p^V)$ while increasing the inter-class distance between x_a^A and x_n^V. The strategy to collect the set T_{tar} at each epoch of the training is described in Alg. 2.

Using Eq. 5.6 as regularizing term in $\mathscr{L}(f^A)$, one can effectively use the embedded faces as targets to learn a speaker turn embedding. Note that this is similar in spirit to the neural network

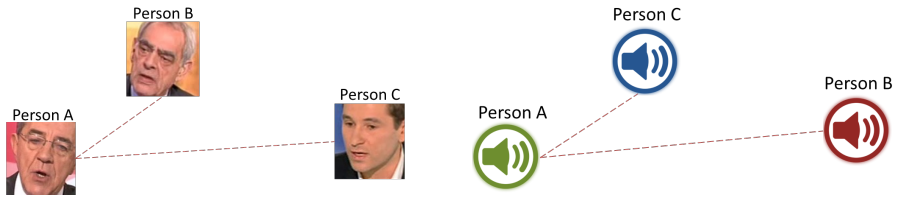


Figure 5.6 – An example of a transfer triplet in relative distance transfer. In the visual domain, $d(personA, personB) < d(personA, personC)$. However, this inequality is not satisfied in the audio domain. Therefore they form a negative triplet for training.

distillation [135], using one embedding as a teacher for the other. Moreover, the two modalities can be combined straightforwardly as their embedding spaces can be viewed as one harmonic space [96], and the embedding features across 2 domains can be compared directly with one another.

5.4.2 Relative distance transfer

The correspondence between faces and voices is not a definitive one-to-one, *i.e.* it is not trivial to precisely select the face corresponding to a voice one has heard. Therefore target embedding transfer might not generalize well even when achieving low training error. Instead of the exact locations, the relative distance transfer approach works at a lower granularity and aims to mimic the discriminative power (i.e. the notion of being close or far) of the face embedding space. Thus, it does not directly transfer the embeddings individual instances but the relative distances between their identities.

Before computing relative distances, let us define the mean face representation M_y of a person and the distance between identities within the face embedding space. Concretely, let X_{y_i} be the set of faces of identity y_i, the mean face representation M_{y_i} of person y_i is computed as:

$$M_{y_i} = \frac{1}{|X_{y_i}|} \sum_{x_i \in X_{y_i}} f^V(x_i) \tag{5.7}$$

where X_{y_i} is the set of visual samples with identity y_i. From $\{M_{y_i}\}$, we can define the distance between identities as:

$$d(y_i, y_j) = d(M_{y_i}, M_{y_j}), \tag{5.8}$$

The goal is then to collect in the set T_{rel} all audio triplets (a, p, n) *with arbitrary identities* where the sample p has an identity which is closer to the identity of the anchor sample a than the identity of the sample n, as defined in the face embedding. In other words, if within the face embedding space the relative distances among the 3 identities of the triplet (a, p, n) follows:

$$d(M^V_{y_a}, M^V_{y_p}) < d(M^V_{y_a}, M^V_{y_n}), \tag{5.9}$$

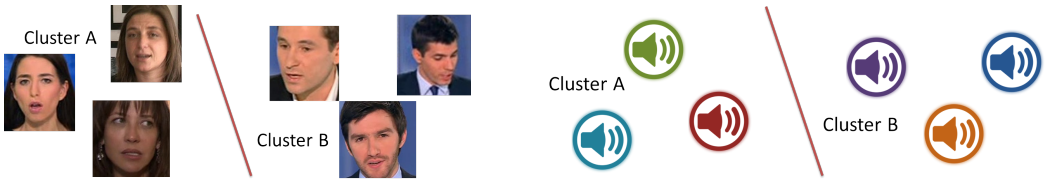


Figure 5.7 – In the visual domain, the identities form 2 clusters (*i.e.* male vs female). We expect the samples in the audio domain to also from the same clustering structure. The audio embedding model is trained to not only discriminate between identities but also to form the same structure.

then this relative condition must hold in the speaker turn embedding space as well:

$$d(f^A(x_a^A), f^A(x_p^A)) + \alpha < d(f^A(x_a^A), f^A(x_n^A)) \tag{5.10}$$

Then, at each epoch, Eq. 5.9 and 5.10 can be used to collect the set T_{rel}, as shown in Alg. 3, and the regularizing transfer loss $\mathscr{L}^{V\rightarrow A}(f^A)$ can then be defined as the average sum of the standard triplet loss over this set. Figure 5.6 illustrates how such negative triplets are formed using Eq. 5.9 and 5.10. In theory, relative distance transfer can achieve the same training error as with target embedding transfer, but leave more freedom to the relaxation of the exact location of the embedded features.

Algorithm 3 Relative distance transfer triplet set. Based on the relative order between identities in the visual domain, $d(M_{y_a}^V, M_{y_p}^V) < d(M_{y_a}^V, M_{y_n}^V)$, we collect the audio triplets that violated this relative order into T_{rel}.

Input f^A, f^V, $\{M^y\}_{y=1..K}$, $\{(x_i^A, x_i^V, y_i)\}_{i=1..N}$
$T_{rel} = \emptyset$
for $\forall (a,p,n) / y_a \neq y_p \wedge y_a \neq y_n$ **do**
  if $d(M_{y_a}^V, M_{y_p}^V) < d(M_{y_a}^V, M_{y_n}^V)$ **then**
    $d_{a,p} = d(f^A(x_a^A), f^A(x_p^A))$
    $d_{a,n} = d(f^A(x_a^A), f^A(x_n^A))$
    if $d_{a,p} + \alpha > d_{a,n}$ **then**
      $T_{rel} = T_{rel} \cup (a,p,n)$
Output T_{rel}

5.4.3 Clustering structure transfer

The common idea of the target transfer and relative distance transfer methods is that people with similar faces should have similar voices. Thus it aims at putting constrains based on the distances among individual instances in the face embedding space. In clustering structure transfer, the central idea does not focus on pair of identities but rather, we hypothesize that commonalities between 2 modalities can be discovered amongst groups of identities. For example, people within a similar age group are more likely to be close together in the face embedding space, and we also expect them to have more similar voices in comparison to other groups.

Based on this hypothesis, we propose to regularize the target speaker turn embedding space to have the same clustering structure with the source face embedding space, *i.e.* an identities should have the same neighbors in the speaker embedding space as in the face embedding space. To achieve that, we first discover groups in the face embedding space by performing a K-Means clustering on the set of mean identity representations $\{M^V_{y_i}\}$ by following 2 steps:

- The set of mean faces of each idendity $\{M^V_{y_i}\}$ is calculated following Eq. 5.7 in relative distance transfer.
- K-Means is performed on the set of $\{M^V_{y_i}\}$. We denote by C the number of clusters, the resulting cluster mapping function is defined as:

$$g_m : \{1..K\} \rightarrow \{1..C\}$$
$$y \rightarrow c_y$$

To define the regularizing term $\mathscr{L}^{V\rightarrow A}(f^A)$, we simply consider the set of cluster labels c_{y_i} attached to each audio sample (x^A_i, y_i) as the second label, and define accordingly a triplet loss relying on this second label (i.e by considering the instances (x^A_i, c_{y_i})). This step is illustrated in Figure 5.7, where the audio samples are assigned the cluster labels from the face domain. In this way, one can guide the acoustic instances of identities from the same cluster to be close together, thus preserving the source clustering structure. How to collect the set of triplet T_{str} to be used for the regularizing term at each epoch is detailed in Alg.4.

Algorithm 4 Clustering structure transfer triplet set. We use the face cluster labeling c_y as additional labels to collect triplets into T_{str}.

Input f^A, f^V, g_m, $\{(x^A_i, x^V_i, y_i)\}_{i=1..N}$
Cluster mapping g_m: $y \rightarrow c_y, \forall y \in 1 \dots K$
$T_{str} = \emptyset$
for $\forall (a, p, n) / c_{y_a} \neq c_{y_p} \wedge c_{y_a} \neq c_{y_n}$ **do**
    $d_{a,p} = d(f^A(x^A_a), f^A(x^A_p))$
    $d_{a,n} = d(f^A(x^A_a), f^A(x^A_n))$
    if $d_{a,p} + \alpha > d_{a,n}$ **then**
        $T_{str} = T_{str} \cup (a, p, n)$
Output T_{str}

This group structure can be expected to generalize for new identities because even though a person is unknown, he/she belongs to a certain group which share similarities in the face and voice domains. In our work, we only apply K-Means once on the mean facial representations. However, as people usually belong to multiple non-exclusive common groups, each with a different attribute, it would be interesting in further works to aggregate multiple clustering partitions with different initial seeds or with different number of clusters. As the space can be hierarchically structured, one other possibility could be to apply hierarchical clustering to obtain these multiple partitions.

5.4.4 Domain adaptation with maximal mean discrepancy

In the previous 3 methods, the emphasis was put on the geometric properties of the embedding spaces with respect to the labels. Therefore, the constraints between spaces are established only if we are given the multimodal correspondence between identities. Hence, these methods may not make full use of the face embedding, which was trained with more identities, and most of these are not present in the audio dataset. To overcome the dependence on labels, we focus on minimizing the difference between the two embedding feature distributions directly.

MMD is a statistical test to quantify the similarity between two distributions p and q on a domain $\mathscr{X}$ by mapping the data to a high dimensional feature space. The observations $X = x_1, ..., x_m$ and $Y = y_1, ..., y_n$ are drawn independently and identically distributed (i.i.d.) from p and q respectively.

To test whether $p = q$, we first introduce a class of function $\mathscr{F}$, which contains $f : \mathscr{X} \to \mathbb{R}$, each f can be simply viewed as a linear mapping function. Given $\mathscr{F}$, the measure of discrepancy between p and q can be estimated as:

$$\mathrm{MMD}[X, Y] := \sup_{f \in \mathscr{F}} \Big(\frac{1}{m} \sum_{i=1}^{m} f(x_i) - \frac{1}{n} \sum_{j=1}^{n} f(y_j) \Big) \tag{5.11}$$

By defining $\mathscr{F}$ as the set of functions in the unit ball in a universal Reproducing Kernel Hilbert Space (RKHS), it was shown that $\mathrm{MMD}[\mathscr{F}, X, Y] = 0$ if and only $if\ p = q$ [122].

Let ϕ be the the mapping to the RKHS and $k(\cdot, \cdot) = < \phi(\cdot), \phi(\cdot) >$ be the universal kernel associated with this mapping. MMD can be computed as the distance between the mean of the two sets after mapping each sample to the RKHS:

$$\begin{aligned} \mathrm{MMD}^2(X, Y) &= \Big|\Big| \frac{1}{m} \sum_{i=1}^{m} \phi(x_i) - \frac{1}{n} \sum_{j=1}^{n} \phi(y_j) \Big|\Big|^2 \\ &= \sum_{i,j=1}^{m} \frac{k(x_i, x_j)}{m^2} - 2 \sum_{i,j=1}^{m,n} \frac{k(x_i, y_j)}{mn} + \sum_{i,j=1}^{n} \frac{k(y_i, y_j)}{n^2} \end{aligned} \tag{5.12}$$

The MMD between the distributions of two sets of observations is equivalent to the distance between the sample means in a high-dimensional feature space. In practice, Gaussian or Laplace kernels are often chosen as they are shown to be universal [136]. We choose kernel k associated to ϕ to be Radial Basis Function kernel, *i.e.* $k(u, v) = \exp(-d(u, v)^2)/\sigma$ in Eq. 5.12.

Originally proposed as a statistic measure between 2 distributions, MMD is widely used as the loss for domain adaptation[131, 132, 133]. Let x_s be the samples from the source domain, x_t be the samples from the target domain, and f^s, f^t be their respective feature mapping functions. By minimizing $\mathrm{MMD}(\{f^s(x^s)\}, \{f^t(x^t)\})$, one can minimize the discrepancy between the feature spaces learned from the two domains, thus enhancing the performance on the target domain using

the knowledge from the source domain. In our work, we adopted the same strategy after unifying the two embedding spaces of faces and speaker turns respectively.

Given that the two embedding spaces can be constrained to lie within the same hypersphere, one can measure the discrepancy between the distributions of face embedded features $f^V(x_i^V)$ and auditory embedded features $f^A(x_j^A)$ using Eq. 5.12 as:

$$\mathscr{L}^{V\rightarrow A}(f^A) = \text{MMD}(\{f^V(x_i^V)\}, \{f^A(x_j^A)\}) \tag{5.13}$$

Based on Eq. 5.13, our objective is to find an embedding which is capable of inferring cross-domain statistical relationships when one exists. Instead of trying to bind faces and voices of the same individual identity geometrically, minimizing Eq. 5.13 only regulates the statistical properties of the whole population in an unsupervised fashion. Intuitively, minimizing the MMD forces the auditory features to have the same distribution with facial features, which includes having similar density around common attributes or identites. This can be interpreted as a regularizing term in $\mathscr{L}(f^A)$ to effectively use the embedded faces to guide the speaker turn embedding.

5.5 Experiments

We first describe the datasets and evaluation protocols before discussing the implementation details and the experimental results.

5.5.1 Datasets

REPERE [44]. We use this standard dataset to collect people tracks with corresponding voice-face information. It features programs including news, debates, and talk shows from two French TV channels, LCP and BFMTV, along with annotations available through the REPERE challenge. The annotations consist of the timestamps when a person appears and talks. By intersecting the talking and appearing information, we can obtain all segments with face and voice from the same identity. As REPERE only contains sparse reference bounding box annotation, automatic face tracks [7] are aligned with reference bounding boxes to get the full face tracks. This collection process is followed by manual examination for correctness and consistency and to remove short tracks (less than 18 frames $\approx$ 0.72s). The resulting data is split into training and test sets. Statistics are shown in Table 5.1.

ETAPE [137]. This standard dataset contains 29 hours of TV broadcast. In this work, we only consider the development set to compare with state-of-the-art methods. Specifically, we use similar settings for the "same/different" audio experiments than in [95]. From this development set, 5130 1-second segments of 58 identities are extracted. Because 15 identities appear in the REPERE training set, we remove them and retain 3746 segments of 43 identities.

Table 5.1 – Statistics of tracks extracted from REPERE. The training and test sets have disjoint identities.

	# shows	# people	# tracks
training	98	208	3360
test	35	98	629

5.5.2 Experimental protocols and metrics

The experiments are designed to benchmarking the quality of the embedding space improved by transfer learning. The same/different experiments are designed following a verification protocol, which is based on assessing distances between pairs of samples. Meanwhile the clustering experiments are designed to quantify if the embedding space is discriminative enough to group segments of each identity among other candidates.

Same/different experiments. Given a set of segments, distances between all pairs are computed. One can then decide if a pair of instances has the same identity if their (embedded) distance is below a threshold. We can then report the equal error rate (EER), *i.e.* the value when the false negative rate and the false positive rate become equal as we vary the threshold.

Clustering experiments. From a set of all audio (or video) segments, a standard hierarchical clustering is applied using the distance between cluster means in the embedded space as merging criteria. At each step, 2 clusters with the minimum distance are merged and a new mean is computed. For every step, we compute 3 metrics on the clustering set:

- Weighted cluster purity (WCP) [23]: For a given set of clusters $C = \{c\}$, each cluster c has a weight of n_c, which is the number of segments within that cluster. At initialization, we start from N segments with a weight of 1 for each segment. The purity $purity_c$ of a cluster c is the fraction of the largest number of segments from the same identities to the total number of segments in the cluster n_c. We can define WCP as:

$$WCP = \frac{1}{N} \sum_{c \in C} n_c \cdot purity_c$$

- Weighted cluster entropy (WCE): A drawback from WCP is that it does not distinguish the errors. For instance, a cluster with 80% purity, 20% error due to 5 different identities is more severe than if it is only due to 2 identities. To characterize this point, we thus compute the entropy of a cluster, from which WCE is calculated as:

$$WCE = \frac{1}{N} \sum_{c \in C} n_c \cdot entropy_c$$

- Operator clicks index (OCI-k) [138]: This is the total number of clicks required to label all clusters. If a cluster is 100% pure, only 1 click is required. Otherwise, besides the 1 click needed to annotate segments of the dominant class, to correct each erroneous track of

a different class, 1 more click will be added. For a cluster c of n_c speaker segments, the cluster cost is formally defined as:

$$\text{OCI-k}(c) = 1 + (n_c - max(\{n_i^c\})),$$

where n_i^c denotes the number of segments from identity i in the cluster. The cluster clicks are then added to produce the overall OCI-k measure. This metric simultaneously combines the number of clusters and cluster quality in one number to represent the manual effort for correctly annotating all speaker segments given an initial clustering.

5.5.3 Implementation details

Face embedding. Our face model is based on the ResNet-34 architecture [4] trained on the CASIA-WebFaces dataset [106]. This is a collection of 494,414 images from 10,575 identities. We follow the procedure of [117] as follows:

- A DPM face detector [78] is run to extract a tight bounding box around each face. No further preprocessing is performed except for randomly flipping training images.
- ResNet-34 is first trained to predict 10,575 identities by minimizing cross entropy criteria. Then the last layer is removed and the weights are frozen.
- The last embedding layer with a dimension of $h = 128$ is learned using Eq. 5.3 and the face tracks of the REPERE training set.

Speaker turn embedding. Our implementation of *TristouNet* consists of a bidirectional LSTM with the hidden size of 32. It is followed by an average pooling of the hidden state over the different time steps of the audio sequence, followed by 2 fully connected layers of size 64 and 128 respectively. As input acoustic features to the LSTM, 13 Mel-Frequency Cepstral Coefficients (MFCC) are extracted with energy and their first and second derivatives.

Optimization. All embedding networks are trained using a fixed $\alpha = 0.2$ and the RMSProp optimizer [107] with a 10^{-3} learning rate. From each mini-batch, both hard and soft negative triplets are used for learning.

Baselines. We compare our speaker turn embedding with 3 approaches: Bayesian Information Criterion (BIC) [139], Gaussian divergence (Div.) [140], and the original *TristouNet* [95].

Additional details. Our codes use PyTorch library and are publicly available at:
gitlab.idiap.ch/software/CTL-AV-Identification/

5.5.4 Experimental results

REPERE - Clustering experiment

We applied the audio (or video) hierarchical clustering to the 629 audio-visual test tracks of REPERE. In Figure 5.8-a, we can compare other methods with the two reference systems: Tristounet for speech embedding and ResNet-34 (Rn34-Emb)for face embedding. Face clustering with Rn34-Emb clearly outperforms all speaker turn based methods. This visual system is used as a reference to show the significant difference between the two domains, thus motivating transfer learning to improve the speech embedding.

Our transferring methods surpass *TristouNet* in both metrics, especially in the middle stages, when the distances between clusters becomes more confusing. This shows that the knowledge from the face embedding helps distinguishing confusing pairs of clusters. The gap in WCE also means that our embedding is also more robust with respect to the inter-cluster distances. Overall, all transfer learning methods are consistent and show steady improvements. On average, MMD has a slightly better result than the others. For the geometry-based methods, the lower the granularity level, the higher the gain in performance is. This shows that due to low the correlation between the 2 domains, it is better to use the latent attributes in the data with some relaxation rather than directly enforcing the embedding targets to be the same for both audio and video domain.

Figure 5.8-b, we compare our best method, MMD, with other state of the art methods. At the beginning, Div. first merges longer audio segments with enough data so it achieves higher purity. However, as small segments get progressively merged, the performance of BIC and Div. quickly deteriorate due to the lack of good voice statistics. We should note that in WCP and WCE, segments count as one unit and are not weighted according to their duration as done in traditional diarization metrics. This is one reason why traditional approaches BIC and Div methods appear much worse with the clustering metrics. More experiments on full diarization are needed in future works.

Table 5.2 reports the number of clicks to label and correct the clustering results. Our MMD approach reduces the OCI-k by 17 from the closest competitor in both the best cases at the minimum OCI-k and at the ideal number of clusters. This in practice can decrease the effort of human annotation by around 7.5%. While target embedding transfer does not yield any improvement, clustering structure and relative distance transferring methods also show decreases of $4.5 - 6.5\%$.

ETAPE - biometrics experiment

From the ETAPE development set, 3746 segments of 43 identities are extracted. From these segments, all possible pairs are used for testing and the EER is reported in Table 5.3. Most of our networks with transferred knowledge outperform the baselines. With short segments of 1 second,

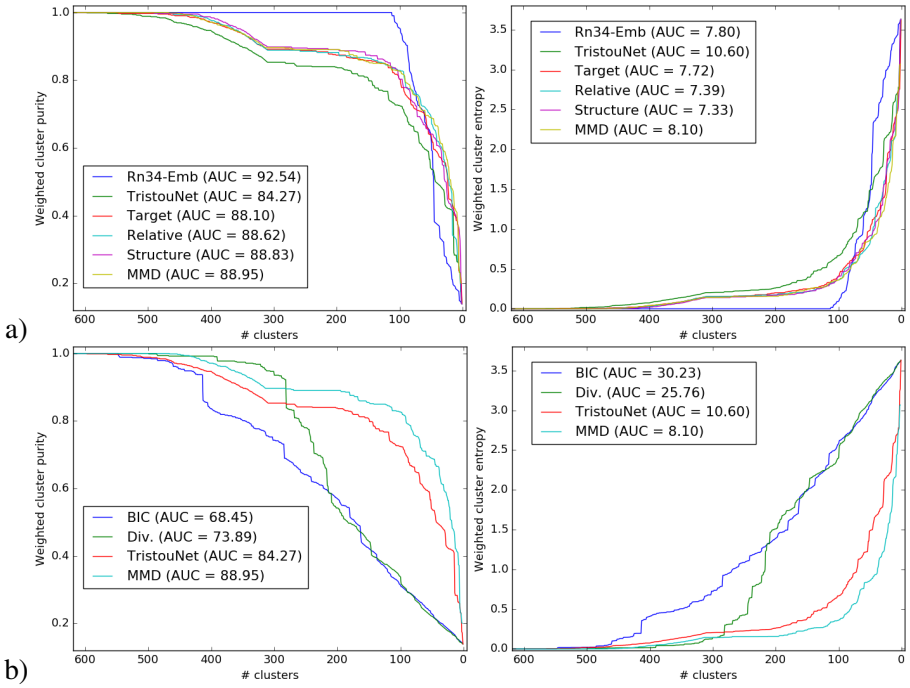


Figure 5.8 – Weighted cluster purity (WCP) and weighted cluster entropy (WCE) evaluation of hierarchical clustering on REPERE. (a) Comparison of our transferring approaches against the baseline TristouNet and the face embedding using ResNet-34 (b) Comparison of our MMD approach against state-of-art audio systems.

Table 5.2 – Result of OCI-k metric on the REPERE test set. 'Min' reports minimum value of OCI-k and in parenthesis is the number of clusters where this is achieved. 'At ideal clusters' reports OCI-k obtained when clustering reaches clusters the ideal number of clusters corresponding to 98 identities.

Methods		Min (# clusters)	At 98 clusters
Vision	Rn34-Emb (V)	113 (113)	136
Audio	BIC [139]	451 (390)	525
	Div. [140]	330 (289)	521
	TristouNet [95]	216 (119)	226
Audio-Visual (Ours)	Target	214 (112)	228
	Relative	204 (99)	211
	Structure	207 (107)	216
	MMD	202 (94)	209

Table 5.3 – EER reported on all pairs of 3746 sequences in ETAPE dev set.

	BIC[139]	Div.[140]	*TristouNet*[95]	Target	Relative	Structure	MMD
EER	32.4	28.9	16.1	16.30	15.59	15.62	15.5

BIC and Div. do not have enough data to fit the Gaussian models well, therefore they perform poorly. By transferring from visual embedding using MMD, we can improve *TristouNet* with a relative improvement of 3.7% of EER. We should remark that in [95], the original *TristouNet* achieved 17.3% and 14.4% when being trained and tested on 1s sequences and 2s sequences respectively. It is important to note that our models are trained on a smaller dataset (8h vs. 13.8h of ETAPE data in [95]) and from an independent training set (REPERE vs. ETAPE). Using our transfer learning methods, the speaker turn embedding model could be easily trained by combining different datasets, *i.e.* combining REPERE and ETAPE training sets.

Results with limited data.

To benchmark the generalization of our approaches, the same verification and clustering protocols from previous subsections are applied when the amount of training audio data is reduced and the results are reported in Table 5.4. Transferring methods perform particularly better in this scenario. In most cases, networks trained with MMD loss achieves better figures. As the amount of training data decreases, the performance of the audio-only system quickly deteriorates, especially in the clustering protocol. On the other hand, our visual guided system is less affected. When using only 60% of data, MMD outperforms audio-only *TristouNet* in OCI-k by 45 points , *i.e.* reducing the manually effort by 16%. Interestingly, both systems perform better with 30% of data than with 60%. One explanation is that although there are fewer samples, they are more balanced among identities. Considering both metrics, this balance in identities helps MMD the most because it is a density-based method. Therefore, the imbalance in the dataset can leads to a skew in the distribution and reduce the effectiveness of MMD. Meanwhile, as target transfer is a

Table 5.4 – Performance when training data are limited. EER is reported on ETAPE dev set. OCI-k is reported on REPERE.

	60%					30%				
	[95]	Tar.	Rel.	Str.	MMD	[95]	Tar.	Rel.	Str.	MMD
Min OCI-k	274	241	256	255	229	249	232	218	225	213
OCI-k@98	285	255	268	271	231	263	250	242	229	221
EER	19.1	18.0	18.2	18.3	18.4	16.9	16.7	16.4	15.9	16.5

Table 5.5 – Performance when combining crossmodal regularizers and intra-class loss. EER is reported on ETAPE dev set. OCI-k is reported on REPERE.

	[13]	[13] + Tar.	[13] + Rel.	[13] + Str.	[13] + MMD
Min OCI-k	214	231	226	216	209
OCI-k@98	221	244	240	228	223
EER	16.3	16.7	15.8	16.0	16.3

sample-based method, the balance in the 30% set does not improve as significantly.

Combining with intra-class regularizer

In this experiment, we explore how our crossmodal losses can be combined with the intra-class loss as described in Chapter 4. Intra-class loss is a soft constraint on the averaged pair-wise distance between samples from the same class. It is also a regularizer preventing the scattering of these samples within the embedding space to increase the intra-class compactness.

In Table 5.5, we first present the results of [13] using the same benchmarks with 30% of the training dataset. Comparing to our methods in Table 5.4, it achieves comparable results with MMD. Subsequently, intra-class loss is linearly combined with all of our crossmodal losses to yield the rest of Table 5.5.

One can first see that combining target transfer and intra-class loss does not give any improvements. This can be explained as the face embedding has a different intra-class structure. Hence, forcing the audio embedding to match both this structure and another explicit intra-class constraint can be conflicting as both losses work on the sample-level granularity. Meanwhile, combining intra-class loss with relative distance transfer and structure transfer can slightly improve EER because these geometric properties are at inter-class level, ie. relationship across identities, and are complementary with intra-class compactness. Finally, MMD concerns with sample density in an unsupervised manner, it is only slightly beneficial in clustering metrics to use intra-class loss with MMD.

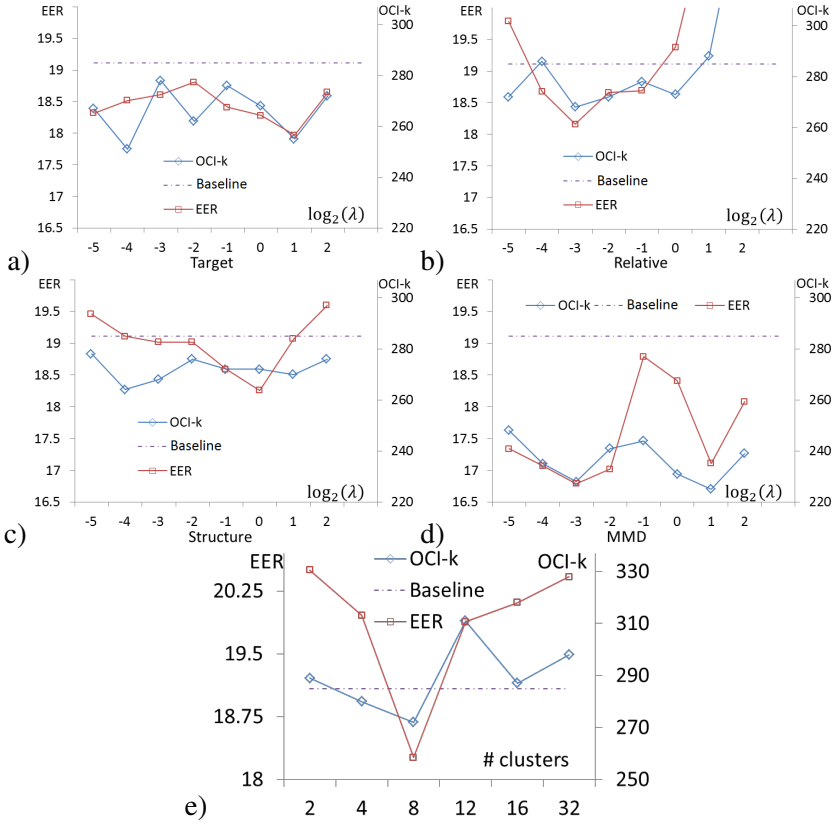


Figure 5.9 – Result of different values of hyperparameters. The baseline is EER and OCI-k of the standard Tristounet. (a-d) EER on ETAPE and OCI-k on REPERE of target, relative, structure, and MMD respectively as λ changes. (e) EER on ETAPE and OCI-k on REPERE as the number of clusters for structure transfer changes.

Parameter sensitivity analysis

In all our transfer learning settings, we need to choose one hyper parameter λ, and the number of clusters for structure transfer setting. Hence, we perform benchmarking with different values of λ varying as power of 2. This experiment is performed on the training set with 60% of data for computational reason and results are reported in Figure 5.9. All of our methods are insensitive to λ except for relative distance transfer. Target embedding transfer is the most stable one, as its constraint is more specific than that of the rest. Each of them has a different optimal value, which is due to the difference in the nature of each method. One possible explanation for the case of relative distance transfer (Figure 5.9-b) when $\lambda \geq 2$ is that there is no proximity constraints on the location of the embedded features, thus instability is not bounded and can increase at test time. Figure5.9-(e) shows how structure transfer performs under different granularity. Further analysis in the characteristics of clusters is presented in next subsection.

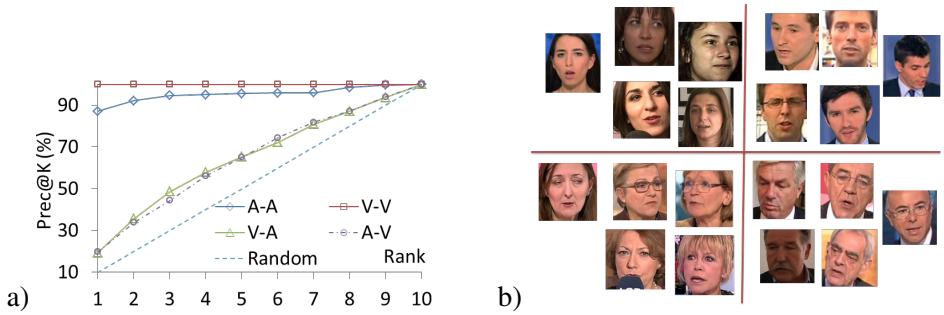


Figure 5.10 – Analysis of different transferring type. (a) Prec@K of cross modal id retrieval using target transfer, (b) visualization of shared identities in 4 clusters across both modalities.

Further multimodal analysis

Cross modal retrieval. One interesting potential of target embedding transfer is the ability to connect a voice to a face of the same identity. To explore this aspect, we formulate a retrieval experiment: given 1 instance of the source embedding domain (voice or face), its distances to the embedding of 1 instance with the same identity and to 9 distractors in the enrolled domain are computed and ranked accordingly. This experiment is similar to that of crossmodal biometrics [141, 142]. There are 4 different settings depending on the within or cross domain retrieval: audio-audio, visual-visual, audio-visual, and visual-audio. Figure 5.10-(a) shows the average precision of 980 different runs when choosing from the top 1 to 10 ranked results (Prec@K). Although the cross modal retrieval settings cannot compete with their single modality counterparts, they perform better than random chance and show consistency between the face embedding and speaker turn embedding. This shows that though the two modalities cannot be used as in coupled matching learning, they can be used as a regularizer of one another.

Shared clusters across modalities. Figure 5.10-(b) visualizes 4 clusters which share the most common identities across the 2 modalities, when using the face embedding and the speaker embedding with structure transfer. One can observe 2 distinct characteristics among the clusters which are automatically captured: gender and age. It is noteworthy that these characteristics are discovered without any supervision.

5.6 Conclusion

In this chapter, we have investigated 4 different methods to transfer knowledge from a source face embedding to a target speaker turn embedding. Inspired by state-of-the-art machine learning literature, each of our approaches explore different properties of the embedding spaces. Depending on the properties exploited, our methods can be categorized into two groups. The first group uses the geometric features at different levels of granularity; *i.e.* through direct target, relative distances, or neighborhood regularization. Meanwhile the second group uses the regularization of the underlying common feature distribution; *i.e.* through regularizing maximum mean discrepancy.

The results show that our methods improved speaker turn embedding in the tasks of verification and clustering. This is particularly significant in cases of short utterances, an important situation that can be found in many dialog cases, *e.g.* TV series, debates, or in multi-party human-robot interactions where backchannels and short answers/utterances are very frequent. The embedding spaces can also provide potential discovery of latent characteristics and a unified crossmodal combination.

Another advantage of the transfer learning approaches is that each modality can be trained independently with their respective data, thus allowing future extension using advance learning techniques or more available data. However, we have just considered the task of speaker turn embedding, which is only the first task in speaker diarization.

In the future, experiments with more complicated tasks such as full person diarization or large scale indexing can be performed to explore the possibilities of each proposal. The main focus of our work is on regularization of the output spaces, thus regularizing the learning models or the intermediate features can be complementary. It is also interesting to use the face embedding guidance to expand the speech identification data.

6 An Analysis of Triplet Loss Under Label Noise

6.1 Introduction

Embedding learning methods aim at learning a parametrized mapping function from a labeled set of samples to a metric space, in which samples with the same labels are close together and samples with different labels are far apart. To learn such an embedding with deep neural networks, the most common losses are contrastive loss [143] and triplet loss [96]. These losses are based on the distances between pairs or triplets of samples respectively. Because these losses only require the information whether 2 samples have the same or different labels, they have potential for learning with uncertainty in sample label information. In fact, there have been an increasing number of works that use embedding losses for unsupervised tasks by inferring the pair-wise label relationship using other sources of information [144, 145, 146, 147, 148]. As the inferred label information is not reliable, they can be misleading and will subvert the model during training. This leads to our research question:

Given a dataset with unreliable labels, what are the guarantees when one learns an embedding using triplet loss or contrastive loss?

This question is becoming more important as there are more datasets where labels are no longer curated by human but by internet queries [117, 149, 106], crossmodal supervision and transfer learning [15, 5], associated social information [150, 151], or data mining [144, 152].

To answer the question, we conducted an analysis on triplet loss under label noise from the risk minimization perspective. Under this perspective, a loss function is said to be tolerant to label noise of rate p if the minimizer of the risk in the noise free condition is also the minimizer of the risk under noise. We first focus on the triplet loss without the hinged function. In this case, we have proved the conditions so that unhinged triplet loss is robust to label noise and shown the bound on the expected risk when there is label noise. From these analytical results, we conjectured two main heuristics in optimizing triplet loss with label uncertainty:

- When the labels are known to be unreliable, it is better to keep the high precision when

choosing positive pairs.
- Random semi-hard mining is more robust to label noise than fixed semi-hard mining.

We have conducted experiments on standard vision datasets to demonstrate how triplet loss perform under label noise in practice and how different sampling strategies and noise rate can affect the tolerance.

The rest of the chapter is organized as follows: Section 6.2 reviews the related work, Section 6.3 introduces preliminary settings for analysis, Section 6.4 presents the main analytical results, Section 6.5 exhibits the exprimental results, and Section 6.6 presents our system to collect speaker labels using face clusters.

6.2 Related work

While embedding losses under label noise have not been studied before, there has been a vast literature in analyzing label noise for classification. For an in-depth introduction to label noise and a comprehensive analysis of traditional algorithms, we refer the readers to the survey of [153].

In the context of deep learning, most effort has been dedicated to improve training networks under label noise. One major direction is to approximate a model of noise to improve training. There are a few examples of this direction. In [154], the authors use active learning to select clean data from noisy training set, in [155], there are multiple iterations of training a model, formulating the noise, and retraining, and in [156] the estimation of noisy labels is used to reweigh the training samples. Another direction is to improve the networks directly to make them robust to label noise. For example, one can add a noise adapting layer to correct the network for the latent noise in training datasets [157] or augment a standard deep network with a softmax layer that models the label noise statistics [158].

While all the above methods focus on changing the learning model or strategy, there is another interesting body of works in analyzing the loss functions used to train the models [159, 160, 161]. Here we want to highlight one such work presented in [160]. In this work, the authors introduced the notion of symmetric loss functions and proved that such symmetric losses are tolerant to label noise. From the theoretical analysis, they have shown mean absolute error as a more robust alternative for cross entropy loss in training classification deep neural networks.

Within this literature, our work can be viewed as a counterpart of [160] for embedding losses. In our work, we not only explore how the per sample label noise affects the pair-wise and triplet-wise labels but also provide further analysis on the impact of sampling and initialization, which are integral parts of learning embeddings.

6.3 Preliminaries

We will recall the losses used for deep embedding learning and then define the scope of label noise to be used in subsequent sections.

6.3.1 Deep embedding learning and triplet loss

Given a labeled training set of $\{(x_i, y_i)\}$, in which $x_i \in \mathbb{R}^D$, $y_i \in \{1, 2, .., K\}$, we define an embedding function as a parameterized $f(x;\theta) \in \mathbb{R}^d$, which maps an instance x into a unit hypersphere in a h-dimensional Euclidean space. In this new embedding space, we want the intra-class distances $d(f(x_i;\theta), f(x_j;\theta))$, $\forall x_i, x_j / y_i = y_j$ to be minimized and the inter-class distances $d(f(x_i;\theta), f(x_j;\theta))$, $\forall x_i, x_j / y_i \neq y_j$ to be maximized. [1]

To achieve such embedding, one method is to learn the projection that optimizes the triplet loss in the embedding space. A triplet consists of 3 data points: (x_a, x_p, x_n) such that $y_a = y_p$ and $y_a \neq y_n$ and thus, we would like the 2 points (x_a, x_p) to be close together and the 2 points (x_a, x_n) to be further away by a margin α in the embedding space. Hence, the loss for one triplet is defined as:

$$l^T(x_a, x_p, x_n;\theta) = [d_{ap} - d_{an} + \alpha]_+. \tag{6.1}$$

6.3.2 Label noise

Given a sample x_i with its true label y_i, we assume this true label can be wrongly observed with a probability p. Let $\hat{y}_i$ be the observed label with the following rule:

$$\hat{y}_i = \begin{cases} y_i & \text{with prob. } 1 - p_{x_i} \\ u & \text{with prob. } p_{x_i u} \qquad \forall u \neq y_i \end{cases} \tag{6.2}$$

in which $\sum_u p_{x_i u} = p_{x_i}$. If the individual noise probability is uniform and independent with the input x_i, we can simply write:

$$\hat{y}_i = \begin{cases} y_i & \text{with prob. } 1 - p \\ u & \text{with prob. } \frac{p}{K-1} \qquad \forall u \neq y_i \end{cases} \tag{6.3}$$

While the analysis can be applied on complicated distributions of noise, we assume that the label noise on the individual sample is uniform and independent of x_i. Thus we only take into account the sample label noise rate p in Eq. 6.3.

[1] For shorthand, we will simply use d_{ij} to replace $d(f(x_i;\theta), f(x_j;\theta))$.

6.3.3 Relationship between sample label noise p and pair label noise q

We want to compute given the sample label noise rate of p, what is the pair label noise rate q. In another word, for a pair of samples with original pair label of $t_{ij} \in \{-1, 1\}$, we want to find the chance that t_{ij} is corrupted into $-t_{ij}$

Negative case $t_{ij} = -1$. The probability a negative pair is corrupted into a positive pair is decomposed into 2 cases:

- one of the two samples changes its label, and the new label is the same with the other one: $2p\frac{(1-p)}{K-1}$.
- both samples' labels change into 2 different labels, and both labels are the same: $\frac{p^2(K-2)}{(K-1)^2}$.

Hence, in this negative case:

$$q_{-1} = 2p\frac{(1-p)}{K-1} + \frac{p^2(K-2)}{(K-1)^2}. \tag{6.4}$$

Positive case $t_{ij} = 1$. The probability a positive pair is corrupted into a negative pair is decomposed into when:

- one of the two samples changes its label any different label: $2p(1-p)$.
- both samples change into different labels: $p^2\big(1-\frac{2}{K-1}\big)$.

In this positive case:

$$q_1 = 2p(1-p) + p^2\Big(1 - \frac{2}{K-1}\Big). \tag{6.5}$$

6.4 Triplet loss under label noise

A triplet is chosen based on the observed labels, $\hat{y}_a$, $\hat{y}_p$, and $\hat{y}_n$. However, as these labels can be noisy, the true labels can be one of 3 following cases:

- $y_a = y_p = y_n$,
- $y_a \neq y_p \neq y_n$,
- $y_a \neq y_p$ and $y_a = y_n$.

The determine the condition for triplet loss to be robust to label noise, we first decompose it into a combination of auxiliary pair-wise losses and consider the unhinged triplet loss.

6.4.1 Auxiliary pair-wise and unhinged triplet loss

Definition 1. *We define an auxiliary pair-wise loss l^A as:*

$$l^A(x_i, x_j, t_{ij}; \theta) = \begin{cases} d_{ij} t_{ij} & \text{if } t_{ij} = 1 \\ d_{max} + d_{ij} t_{ij} & \text{if } t_{ij} = -1 \end{cases} \tag{6.6}$$

in which $d_{max} = 2$ is the maximum distance between 2 points on the hypersphere. Note the property that:

$$l^A_{ij}(-t_{ij}; \theta) = d_{max} - l^A_{ij}(t_{ij}; \theta), \forall i, j. \tag{6.7}$$

Definition 2. *We define the label-dependent weighted version of the auxiliary loss as when each pair (x_i, x_j, t_{ij}) is weighted differently by $w_{t_{ij}}$, in which the weight only depends on the pair label. Under noise, when a pair changes its pair label from t_{ij} into $-t_{ij}$, its weight only changes from $w_{t_{ij}}$ into $w_{-t_{ij}}$.*

Definition 3. *For a given triplet of $x_a, x_p, x_n / y_a = y_p \wedge y_a \neq y_n$, we define the unhinged triplet loss l_U, which can be decomposed into auxiliary pair-wise loss, as:*

$$\begin{aligned} l^U(x_a, x_p, x_n; \theta) &= d_{max} + d_{ap} - d_{an} + \alpha \\ &= l^A_{ap}(t_{ap}, \theta) + l^A_{an}(t_{an}, \theta) + \alpha. \end{aligned} \tag{6.8}$$

Definition 4. *We define a 1-1 sampling scheme for triplet loss as when for a given positive pair, out of all possible negative pairs of the anchor, only 1 negative pair is chosen.*

6.4.2 Risk minimization

From the risk minimization perspective, one might aim at optimizing the expected loss over all triplets:

$$R_L(\theta) = \mathbb{E}_{x_a, x_p, x_n}[l(x_a, x_p, x_n; \theta)]. \tag{6.9}$$

As the unhinged triplet loss l^U can be decomposed as a linear combination of auxiliary pair loss l^A, the unhinged expected risk can be rewritten as:

$$\begin{aligned} R_{l^U}(\theta) &= \mathbb{E}_{x_a, x_p, x_n}[l^U(x_a, x_p, x_n; \theta)] \\ &= \mathbb{E}_{x_i, x_j, t_{ij}}[N_{ij} l^A(x_i, x_j, t_{ij}, \theta)]. \end{aligned} \tag{6.10}$$

where N_{ij} is the weight as each pair can be chosen multiple times in triplet loss. Assuming that there are uniformly s samples per every class and there are K classes, then $N^{ij} = (K-1)s$ if $t_{ij} = 1$ (each positive pair can be combined with $(K-1)s$ negative pairs) and $N^{ij} = s - 1$ if $t_{ij} = -1$ (each negative pair can be combined with $s - 1$ positive pairs).

When a 1-1 sampling scheme is applied, each positive pair is chosen only once, while the probability that one negative pair is chosen is approximately $\frac{1}{K}$ if each class has a uniform number of samples. After the sampling scheme, we can rewrite the risk to be:

$$R_{l^U}(\theta) = \mathbb{E}_{x_i,x_j,t_{ij}}[w_{t_{ij}} l^A(x_i, x_j, t_{ij}, \theta)]. \tag{6.11}$$

where $w_{t_{ij}}$ is the weight associated to the probability each pair is chosen:

$$w_{t_{ij}} = \begin{cases} 1 & \text{if } t_{ij} = 1 \\ \frac{1}{K} & \text{if } t_{ij} = -1 \end{cases} \tag{6.12}$$

Under label noise, the pair label t_{ij} can be corrupted into $\hat{t}_{ij}$. Then, the expected risk unhinged triplet loss under noise $\hat{R}_{l^U}(\theta)$ is:

$$\begin{aligned} \hat{R}_{l^U}(\theta) &= \mathbb{E}_{x_i,x_j,\hat{t}_{ij}}[w_{t_{ij}} l^A(x_i, x_j, t_{ij}, \theta)] \\ &= \mathbb{E}_{x_i,x_j} \mathbb{E}_{t_{ij}|x_i,x_j} \mathbb{E}_{\hat{t}_{ij}|x_i,x_j,t_{ij}}[w_{t_{ij}} l^A(x_i, x_j, \hat{t}_{ij}, \theta)]. \end{aligned} \tag{6.13}$$

Using the probability $q_{t_{ij}}$ of a pair label changing from t_{ij} into $-t_{ij}$, we replace the term $\mathbb{E}_{\hat{t}_{ij}|x_i,x_j,t_{ij}}[w_{t_{ij}} l^A(x_i, x_j, \hat{t}_{ij}, \theta)]$ in the expected risk to have:

$$\hat{R}_{l^U}(\theta) = \mathbb{E}_{x_i,x_j,t_{ij}}\Big[w_{t_{ij}}(1 - q_{t_{ij}}) l^A(x_i, x_j, t_{ij}, \theta) + w_{-t_{ij}} q_{t_{ij}} l^A(x_i, x_j, -t_{ij}, \theta)\Big]. \tag{6.14}$$

Here we will use the fact that $l^A_{ij}(-t_{ij}; \theta) = d_{max} - l^A_{ij}(t_{ij}; \theta)$ in Def. 1 to expand the risk under noise as:

$$\begin{aligned} \hat{R}_{l^U}(\theta) &= \mathbb{E}_{x_i,x_j,t_{ij}}\Big[w_{t_{ij}}(1 - q_{t_{ij}}) l^A(x_i, x_j, t_{ij}, \theta) + w_{-t_{ij}} q_{t_{ij}}\big(d_{max} - l^A(x_i, x_j, t_{ij}, \theta)\big)\Big] \\ &= \mathbb{E}_{x_i,x_j,t_{ij}}\Big[\Big(1 - q_{t_{ij}} - q_{t_{ij}} \frac{w_{-t_{ij}}}{w_{t_{ij}}}\Big) w_{t_{ij}} l^A(x_i, x_j, t_{ij}, \theta) + w_{-t_{ij}} q_{t_{ij}} d_{max}\Big] \\ &= \mathbb{E}_{x_i,x_j,t_{ij}}\Big[Q_{t_{ij}} w_{t_{ij}} l^A(x_i, x_j, t_{ij}, \theta) + w_{-t_{ij}} q_{t_{ij}} d_{max}\Big]. \end{aligned} \tag{6.15}$$

For short notation, we have set:

$$Q_{t_{ij}} = 1 - q_{t_{ij}} - q_{t_{ij}} \frac{w_{-t_{ij}}}{w_{t_{ij}}}. \tag{6.16}$$

Remark. In Equation 6.15, one can see that in the expected risk under noise $\hat{R}_{l^U}(\theta)$, we actually only need the clean loss $l^A(x_i, x_j, t_{ij}, \theta)$. We will use this fact to show that the minimizer of the expected risk under noise will also be the minimizer of the clean expected risk.

6.4.3 Unhinged triplet loss under label noise

Proposition 1. *A minimizer θ^* of the expected risk with unhinged triplet loss in the noise free condition $R_{l^U}(\theta)$ is also the minimizer of the expected risk with unhinged triplet loss under noise $\hat{R}_{l^U}(\theta)$ if:*

1. *A 1-1 sampling scheme is used.*
2. *θ^* is also the minimizer of 2 expected risk $\mathscr{S}^+$ and $\mathscr{S}^-$ over the distributions of positive pairs and negative pairs respectively:*
   - $\mathscr{S}^+ = \mathbb{E}_{x_i,x_j,t_{ij}=1}\, l^A(x_i, x_j, t_{ij}, \theta)$.
   - $\mathscr{S}^- = \mathbb{E}_{x_i,x_j,t_{ij}=-1}\, l^A(x_i, x_j, t_{ij}, \theta)$.
3. $\min_{t_{ij}}\{Q_{t_{ij}}\} = \min_{t_{ij}}\left(1 - q_{t_{ij}} - q_{t_{ij}}\frac{w_{-t_{ij}}}{w_{t_{ij}}}\right) \geq 0.$

Proof. The proof is provided in the Appendix C. The proof sketch is as follows. We first consider θ^* be the optimizer of the clean risk $R_{l^U}(\theta)$, which gives us $R_{l^U}(\theta^*) - R_{l^U}(\theta) \leq 0 \qquad \forall\theta$. We apply the same θ^* into the noisy risk case $\hat{R}_{l^U}(\theta^*) - \hat{R}_{l^U}(\theta)$, then use Eq. 6.15 and condition 2 to expand it to have:

$$\hat{R}_{l^U}(S,\theta^*) - \hat{R}_{l^U}(S,\theta) \leq \min_{t_{ij}}\{Q_{t_{ij}}\}\Big(R_{l^U}(S,\theta^*) - R_{l^U}(S,\theta)\Big). \tag{6.17}$$

Hence, θ^* will also be the minimizer of the noisy risk $\hat{R}_{l^U}(S,\theta)$ if this condition is satisfied

$$\min_{t_{ij}}\{Q_{t_{ij}}\} = \min_{t_{ij}}\left(1 - q_{t_{ij}} - q_{t_{ij}}\frac{w_{-t_{ij}}}{w_{t_{ij}}}\right) \geq 0. \tag{6.18}$$

□

Proposition 2. *Let $\hat{\theta}^*$ be a minimizer of the noisy expected risk with unhinged triplet loss $\hat{R}_{l^U}(\theta)$. Let θ^* be a minimizer of the clean expected risk with unhinged triplet loss $R_{l^U}(\theta)$. Then, when an 1-1 sampling is used, $R_{l^U}(\hat{\theta}^*)$ is upper bounded by:*

$$R_{l^U}(\hat{\theta}^*) \leq \frac{\max_{t_{ij}}\{Q_{t_{ij}}\}}{\min_{t_{ij}}\{Q_{t_{ij}}\}} R_{l^U}(\theta^*)$$

Proof. Because $\hat{\theta}^*$ be the optimizer of the noisy risk $\hat{R}_{l^U}(\theta)$, which gives us:

$$\begin{aligned}
&\hat{R}_{l^U}(\hat{\theta}^*) - \hat{R}_{l^U}(\theta^*) \leq 0 \\
\Rightarrow &\mathbb{E}_{x_i,x_j,t_{ij}}\left[Q_{t_{ij}} w_{t_{ij}} l^A(x_i, x_j, t_{ij}, \hat{\theta}^*)\right] - \mathbb{E}_{x_i,x_j,t_{ij}}\left[Q_{t_{ij}} w_{t_{ij}} l^A(x_i, x_j, t_{ij}, \theta^*)\right] \leq 0 \\
\Rightarrow &\min_{t_{ij}}\{Q_{t_{ij}}\} R_{l^U}(\hat{\theta}^*) - \max_{t_{ij}}\{Q_{t_{ij}}\} R_{l^U}(\theta^*) \leq 0 \\
\Rightarrow &R_{l^U}(\hat{\theta}^*) \leq \frac{\max_{t_{ij}}\{Q_{t_{ij}}\}}{\min_{t_{ij}}\{Q_{t_{ij}}\}} R_{l^U}(\theta^*).
\end{aligned} \tag{6.19}$$

□

A model can achieve the condition 2 in Proposition 1 that θ^* is the minimizer of $\mathscr{S}^+$ and $\mathscr{S}^-$ when on average, all the positive pairs are as close as they can be and the negative pairs are as far as they can be. In other words, an ideal noise free model learned with triplet loss should be sufficiently good at separating inputs into their respective clusters to guarantee that the model learned under noise will be robust to noise. However, this condition is only satisfied if the data is trivial to be separated and is not practical.

In the pratical case when condition 2 is not satisfied, one will reach a different minimizer $\hat{\theta}^*$. The Proposition 2 provided the bound on the difference of expected risks in this case. Essentially, we want the ratio $\frac{\max_{t_{ij}}\{Q_{t_{ij}}\}}{\min_{t_{ij}}\{Q_{t_{ij}}\}}$ to be as close to 1 as possible. This ratio is affected by 2 factors: the noise rate $p_{t_{ij}}$ and the weight ratio $\frac{w_{-t_{ij}}}{w_{t_{ij}}}$.

We first consider the noise rate $p_{t_{ij}}$. One can use the values in Eq. 6.4 and 6.5 to show that it is the positive noise rate p_{+1} that impacts $\min_{t_{ij}}\{Q_{t_{ij}}\}$. Intuitively, a wrong positive pair is always sampled while a wrong negative pair may not be sampled at all. Hence, we can have the heuristic about learning with label noise:

When the labels are known to be unreliable, it is better to keep the high precision when choosing positive pairs.

We will consider the weight ratio $\frac{w_{-t_{ij}}}{w_{t_{ij}}}$ together with the hinged triplet loss in the next part.

6.4.4 Triplet loss and semi-hard mining

When applying triplet loss in practice, there are 2 main differences from the theoretical unhinged version: the hinge function and semi-hard triplet mining. We first consider the hinge function. By setting a threshold in choosing the triplets, it gives higher weights to harder negative pairs and lower weights to easier negative pairs with respect to the positive distance. Concretely, for $t_{ij} = -1$, w_{ij} can be $\frac{\eta_{ij}}{K}$ for the harder pairs and $\frac{1}{\eta_{ij}K}$ for easier pairs, with η_{ij} being some value greater than 1.

Let $\eta = max\{\eta_{ij}\}$, the bigger η is, the bigger the ratio $\frac{\max_{t_{ij}}\{Q_{t_{ij}}\}}{\min_{t_{ij}}\{Q_{t_{ij}}\}}$ in the bound of Proposition 2, which also means the loss is less robust to noise by a factor of η. Intuitively, because noisy negative pairs have smaller distances, they are more likely to be chosen by a factor of η. This makes triplet loss less resistant to label noise also by a factor of η. Though η cannot be computed in practice, ideally the more uniformly negative pairs are sampled, the smaller the value of η is.

η and sampling strategies. Due to the fact that the way negative pairs are sampled depends on the mining strategy used, we now investigate 2 different variants of semi-hard triplet mining, namely random semi-hard and fixed semi-hard, as follows:

- Random semi-hard: for every positive pair, we randomly sample one negative pair so that

the corresponding triplet loss is non negative [96]. Concretely, given the positive pair x_a, x_p, the negative index n^* is chosen as:

$$n^* = \underset{n}{\text{rand}}\{n / d_{ap} - d_{an} + \alpha > 0\}. \tag{6.20}$$

- Fixed semi-hard: for every positive pair, we sample the hardest negative pair so that the corresponding triplet loss is still less than α (ie. the hardest semi-hard negative pair) [97]. Thus, given the positive pair x_a, x_p, the negative index n is chosen as:

$$n^* = \underset{n / d_{ap} < d_{an}}{\text{argmin}}\ d_{an}. \tag{6.21}$$

One can observe that both semi-hard mining strategies are 1-1 sampling schemes. The negative pairs sampled by fixed semi-hard mining will be more concentrated in the harder range than the negative pairs sampled by random semi-hard mining. Therefore, we can conjecture that fixed semi-hard mining will have a larger skew value η than that of random semi-hard, or $\eta_{rand} < \eta_{fixed}$, or as another heuristic:

Random semi-hard mining is more robust to label noise than fixed semi-hard mining.

In the experiments, we will show further how the value of η varies based on the sampling scheme in the investigated datasets.

6.5 Experiments

6.5.1 Preliminary settings

Datasets. We illustrate the guarantees through experiments on 3 datasets: Stanford online product (SOP) dataset [149], CUB-200-2011 bird dataset [162], and Oxford-102 Flowers dataset [163].

Metrics. For the image retrieval task, we use the Recall@K as in [149]. For the clustering task, we use the Normalized Mutual Information (NMI) score to evaluate the quality of clustering alignments given a labled groundtruth clustering [97]. We use K-means algorithm for clustering.

Architecture and training. We use the ResNet architecture with 34 layers [4]. The optimizer is RMSProp [107] and the minibatch size is 60 (12 classes x 5 images). For the CUB and Flowers datasets, we use the pretrained classification model on ImageNet.

Reference topline. As having noisy labels means there are fewer correct data points for training. Hence, to disentangle the effect of lacking data, we compare the result of learning with noise rate p with the topline result of learning with only $1 - p$ remaining random data samples. We also perform analysis on the marginal loss (a generalized contrastive loss) [97] for comparison.

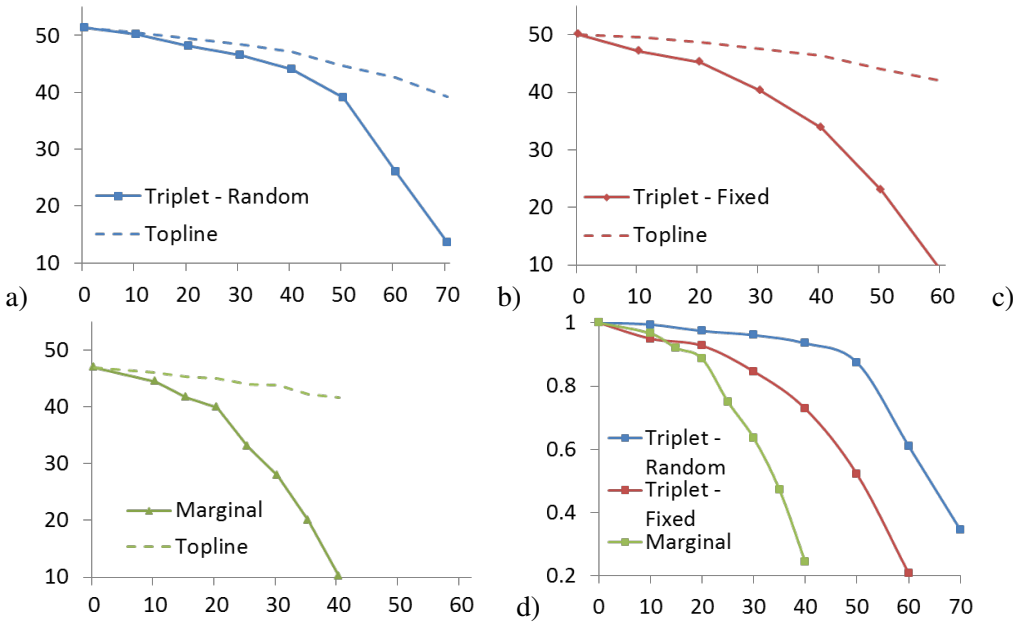


Figure 6.1 – Retrieval results reported on Standford Online Products dataset. x-axis: noise rate p. (a-c) y-axis: Rec@1 of triplet loss with random semi-hard mining, fixed semi-hard mining, and marginal loss with random semi-hard mining, respectively. (d) y-axis: the ratio of Rec@1 for noise rate p over Rec@1 when there are $1-p$ data samples (topline) for all three cases.

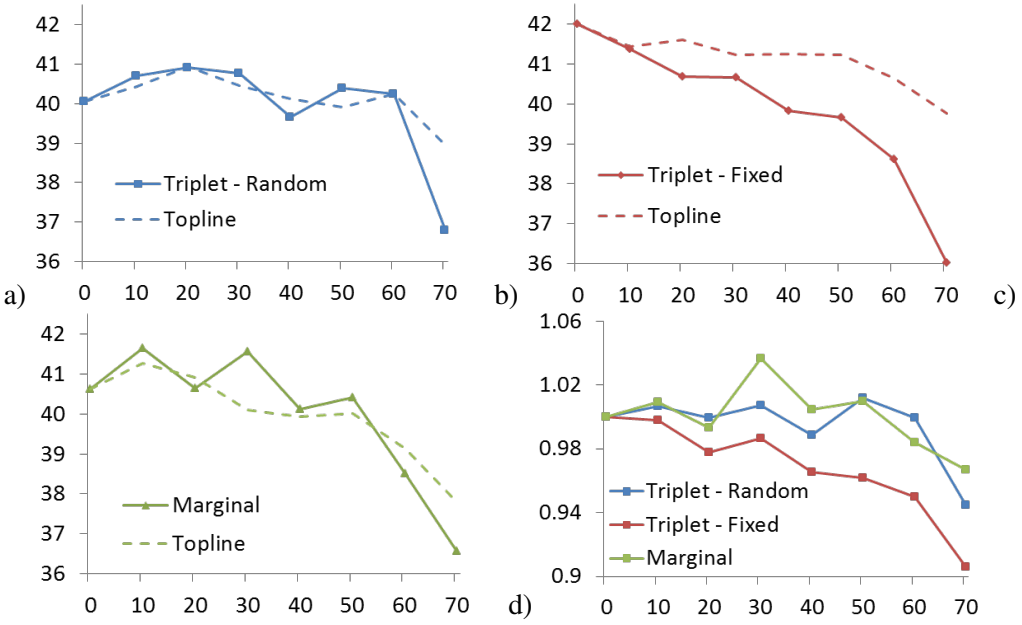


Figure 6.2 – Retrieval results reported on CUB-200-2011 birds dataset. x-axis: noise rate p.(a-c) y-axis: Rec@1 of triplet loss with random semi-hard mining, fixed semi-hard mining, and marginal loss with random semi-hard mining, respectively. (d) y-axis: the ratio of Rec@1 for noise rate p over Rec@1 when there are $1-p$ data samples (topline) for all three cases.

6.5.2 Analysis

Triplet loss. In image retrieval task, triplet loss is robust to label noise less than 20% and varies differently based on each dataset. When there is no label noise, triplet loss with fixed semi-hard mining performs slightly better than with random semi-hard mining. However, when there is label noise, fixed semi-hard deteriorates faster. In SOP dataset (Fig. 6.1-a, b), the gap between learning with noise and learning with fewer clean labels widen significantly after 30% for fixed semi-hard mining while random semi-hard mining still retains good relative performance after 50%. The difference can be examined directly by comparing the ratio between accuracy with noise over accuracy with fewer samples in Fig. 6.1-d, where fixed semi-hard mining is clearly below random semi-hard mining. The same behaviour is observed in CUB dataset (Fig. 6.2-a, b, d) and Flowers dataset (Fig. 6.3-a, b, d) This result shows how different sampling strategies affect the robustness to label noise differently. It also corroborates our conjecture that $\eta_{fixed} > \eta_{rand}$.

Marginal loss. Compared to triplet loss, marginal loss exhibits a higher variance of robustness across datasets in image retrieval task. In SOP, marginal loss degrades much faster than both versions of triplet loss, as shown in Fig. 6.1-c,d. Meanwhile in CUB dataset, marginal loss is relatively as robust as triplet loss with random semi-hard mining, with the breakpoint of 50% comparing to 60% in triplet loss (Fig. 6.2-c,d). In Flowers dataset, even though the performance of marginal loss decreases slightly faster than that of triplet loss, it may due to the fact that marginal loss performs worse with fewer data rather than due to noise (Fig. 6.3-c). When comparing the relative noise, it still shows the same relative robustness with triplet loss. Because in CUB and Flower datasets, we start with the pretrained model, which means the initial θ is already good and the final θ^* is probably reachable through local optimizing steps.

Additional results on clustering tasks. In Fig. 6.4, we show the ratios of the NMIs under noise rate p over the NMIs of missing rate p of data (topline) for all investigated methods in all 3 datasets. Overall, the results in the clustering task agree with our conclusions from the image retrieval task. Using random semi-hard mining with triplet loss yields more diverse negative pairs, making it more robust to label noise than fixed semi-hard mining. Marginal loss with good initialization shows a statistically similar level of robustness with triplet loss. More detailed figures on the clustering task are provided in the supplementary.

6.6 Speaker embedding learning using face cluster labels

As we saw in Chapter 4, data is very important to learn a good speaker embedding for clustering or recognition. However, data annotation is very costly because we cannot use search engine for voice or do mass annotation like images. Therefore, there has been work in using face recognition as the proxy task to collect voice data [5] but this method is limited to only well known people with pretrained face models. We propose to alleviate this problem by using face clusters as indicators to collect positive pairs and negative pairs to train a speaker embedding.

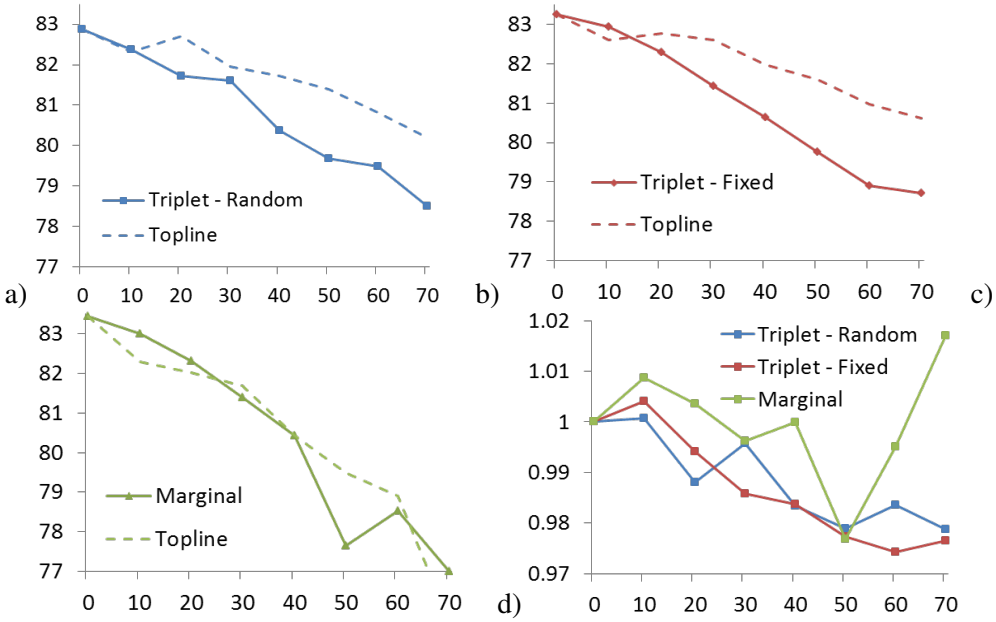


Figure 6.3 – Retrieval results reported on Oxford-102 flowers dataset. x-axis: noise rate p.(a-c) y-axis: Rec@1 of triplet loss with random semi-hard mining, fixed semi-hard mining, and marginal loss with random semi-hard mining, respectively. (d) y-axis: the ratio of Rec@1 for noise rate p over Rec@1 when there are $1-p$ data samples (topline) for all three cases.

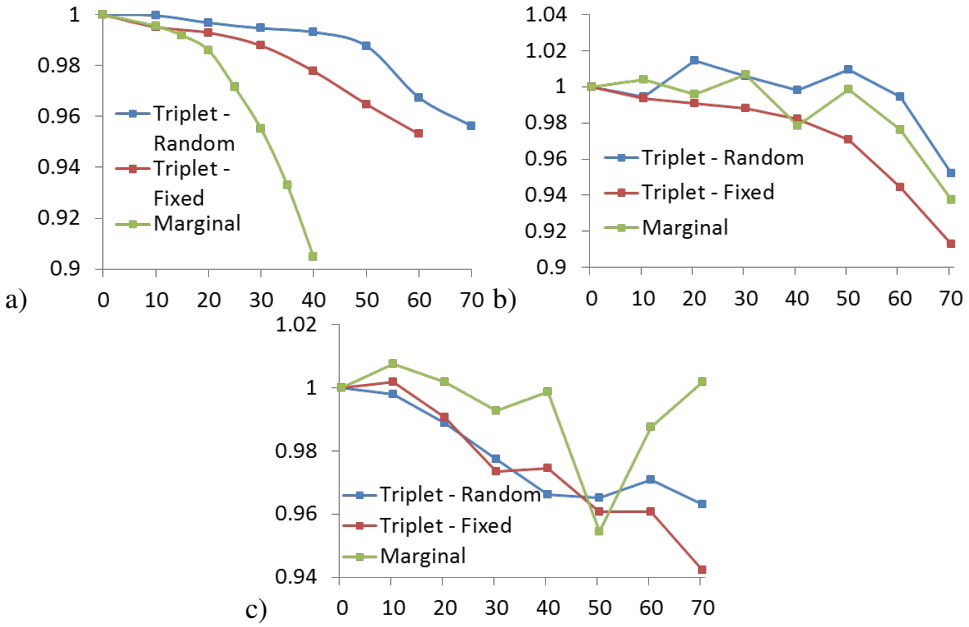


Figure 6.4 – Clustering results x-axis: noise rate p, y-axis: the ratio of NMI for noise rate p over NMI when there are $1-p$ data samples (topline) for triplet loss with random semi-hard sampling, fixed semi-hard mining, and marginal loss with random semi-hard sampling. (a-c) results on the SOP, CUB, and Flowers datasets, respectively

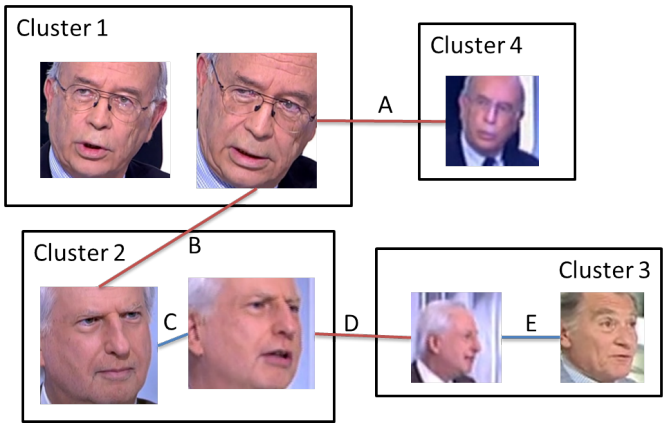


Figure 6.5 – An example of our label collection method. Cluster 1 and cluster 4 are from the same identity but are not merged due to low resolution and different lighting. Meanwhile, cluster 3 contains a track which should have been merged into cluster 2 and a track from a different identity. The positive pair C is true positive while pair E is a positive label noise. The negative pair B is true negative while pairs A and D are negative label noise.

6.6.1 Label collection settings

Here, we assume that a collection of videos without any labels is given. Face tracking and face clustering are first performed using the methodology from Chapter 2. Each face track is then divided based on speaker turn segmentation and talking face detection is applied. After this step, we yield a set of talking face tracks and their respective clusters. Hence, the talking face tracks from the same clusters can be considered to have the same label. Hence, the corresponding voice segments can be sampled to create positive pairs. Similarly, voice segments from different clusters can be sampled to create negative pairs. However, face clustering can have supoptimal results and create wrong pair labels, or label noise. For example, 2 face tracks with different lighting conditions or poses can end up in different clusters. Figure. 6.5 illustrates the method and its potential label noise. We use this practical dataset with label noise to demonstrate our analysis from the previous section.

6.6.2 Experimental settings

In this demonstrating experiment, we use the VoxCeleb training set without using the labels. We perform face clustering on a minibatch of 10 videos. Negative pairs are sampled from clusters of different videos only. For the positive pairs, we vary the maximum distance of hierarchical face clustering for analysis. These pairs are used to form triplets to train a speaker embedding with ResNet-34, similarly to the experiments in Chapter 4. We report metrics for 3 tasks: similar instance retrieval (Rec@K), clustering (NMI), and verification (EER) on the VoxCeleb test set.

Table 6.1 – Results of instance retrieval (Rec@K), clustering (NMI), and verification (EER) on the VoxCeleb test set. We report models learned with weak face labels at different clustering thresholds. 'Supervised' denotes the performance of the model learned with the clean labels for the VoxCeleb training set.

Max dist.	Rec@1	Rec@10	NMI	EER
0.1	87.9	97.6	50.3	29.3
0.4	82.2	96.1	47.0	33.2
0.6	67.3	91.9	40.4	33.1
Supervised	94.7	99.2	82.0	11.7

6.6.3 Analysis

Table 6.1 reports result at different maximum distance for merging face clusters. For each different maximum distance of hierarchical clustering, one receives a different face clustering. The smaller the maximum distance is, the purer each cluster is. Hence, when sampling for the positive pairs, smaller maximum distance will give pairs with higher probability of being correct. Theoretically, this means better resistance against label noise. Comparing different models learned with different face clustering in Table 6.1 confirms the theoretical insights. The more precise we sample the positive pairs, the better the performance across all tasks.

In comparison to the full supervised model, for the similar instance retrieval task, models learned with weak supervision can achieve very high recall (87.9% vs. 94.7% in Rec@1). This shows that face clustering can provide reliable label information with respect to the closest instances. On the other hand, in the verification task, there is a signification gap from the top line to the weakly supervised models. This means that face clustering cannot yield instances with large intra-class variation for training. Also, each minibatch contains only 10 videos. Therefore, the models fail to learn an embedding that can generalize across recording conditions. The result creates a conundrum: we want pure clusters to be more resistant against label noise but we also want each cluster to cover sufficient within class variation. This is an interesting problem for future research.

6.7 Conclusion

We have provided the analysis of the common triplet loss for embedding learning. Our analytical results show a dependence between the sampling strategies and the resistance against label noise in embedding learning. We also analyze and provide practical guidelines for practical tasks when we want to learn a good embedding (without any change in the algorithm or network architecture) even if the training set labels are noisy. We demonstrate our results on standard image retrieval datasets and apply these insights in the task of learning a speaker embedding with pseudo-labels collecting from face clustering. The performance of the speaker models agrees with the findings while presenting interesting future research problem. There are several other potential research

directions to extend our work. The first one is to investigate other sampling strategies such as in [97, 164]. Other embedding losses, for example quadruple loss [165], N-pair loss [166], or marginal loss with learnable β [97].

7 Conclusion

Contributions

The massive amount of multimedia contents produced everyday presents many challenges as well as many opportunities for research in video indexing and understanding. In this thesis, we addressed the challenge of indexing people in videos and explored the opportunity of multimodal learning across audio and visual domains. The contributions can be summarized as follows:

- In Chapter 2, we proposed a face tracking framework which leverages the costly but accurate face detector with a tracking-by-detection CRF-based method. This framework enabled skipping frames to reduce complexity while achieving state-of-the-art results. Furthermore, we accelerated face clustering using shot context and improved the accuracy by combining 2 complementary face similarity measures.

- In Chapter 3, we solved the problem of face and voice association by proposing a multimodal framework that uses a recurrent neural network to learn the temporal representation from the correlation component projection. It was then integrated in a person naming system which won two evaluation campaigns.

- In Chapter 4, a regularizer called intra-class loss was introduced to increase the compactness of speaker embedding features, leading to improvement in speaker recognition.

- In Chapter 5, several methods for transfer learning from a facial domain to a speech domain were explored. These methods constrain different properties of an embedding space such as its geometric arrangement or the its feature distribution. The results were positive and the analysis showed potential for future work.

- In Chapter 6, the problem is how to use face clustering results as pseudo-labels to collect training data to learn speaker recognition models. This problem requires the understanding of learning an embedding with triplet loss under label noise. We made an analysis of embedding learning under label uncertainty and studied how to make models more resistant

to such noise. We used the results to build a framework to learn speaker models with supervision from the facial domain.

Limitations and perspectives

Below, we discuss the limitations of our methods and possible directions for future research within 4 main topics.

Person diarization. The main focus of our framework was to significantly reduce the computational cost for processing large databases. Hence, we have utilized mostly primitive features (color, position) as well as local features (SURF, DCT) that are fast to compute. Provided that there have been rapid advances in high performance hardware as well as in deep learning architectures, the focus can be shifted to investigating more advance learning algorithms and features. For instance, the main idea of our tracking framework is to use the long term sensitive multi-cue CRF, this can be easily expanded with more advanced deep features and similarity measures. The divide-and-conquer clustering strategy using shot context can also be used in other video analysis tasks in complementary with other representations.

Multimodal sequence representation. In our dubbing detection work, there are 2 main limitations. First, we only consider visual features within the mouth region. This leads to the loss of person attributes information. Therefore, we consider further experiments in using full facial or upper body motions as input. Second, the dataset is relatively small and is restricted to only TV broadcast. To alleviate this issue, a few possible directions are to create artificial dubbing data through cross video mixing or to use associated meta data to quickly collect more positive data. Another potential aspect is to use deep canonical analysis or stacked LSTMs for more powerful correlation learning. Nevertheless, our multmodal framework to detect asynchrony has been adopted in the biometrics problem of detecting video tampering [167, 168, 169].

Audio-visual person embedding learning. Recently, we have observed more works in learning joint recognition models using audio-visual data [170, 171]. These works also consider the regularization of the output spaces or the correlation of features between the 2 domains. Going beyond this, we hypothesize that using similar structures and imposing constrains on the model weights can bring further improvement.

Weakly supervised learning with audio-visual data. Our recent findings on the resistance of triplet loss under noise offer several interesting directions. In practice, we have observed the trade-off between having more precise labels and obtaining more training variation within a class. Hence, formalizing this trade-off and designing an effective sampling method are crucial to achieve good performance. As our analysis only has guarantees on the minimizers of the loss functions, optimizing these losses under noise is another vital factor, which includes research on the stability of training or a loss correction method.

A Appendix for Chapter 2

Experiments on Multi-Object Tracking (MOT) Challenge 2016

In [2], the original model was evaluated on the CAVIAR, TUD sequences, PETS-S2L1, TownCenter, and ParkingLot sequences and was providing top results. The new MOT16 benchmark contains 14 sequences with more crowded scenarios, more scene obstacles, different viewpoints and camera motions and weather conditions, making it quite challenging for the method which did not incorporate specific treatments to handle some of these elements (camera motion, scene occluders). The MOT16 challenge thus allows to better evaluate the model under these circumstances.

Parameter setting

For each test sequence, there is a training sequence in similar conditions. As explained earlier, we have used the training sequences to learn Potts models, and used them on the test data. Other parameters (e.g. for reliability factors) were set according to [2] and early results on the training data. Unless stated otherwise, the default parameters (used as well on test data) are: $T_w = 24$, $\Delta_{sk} = 3$ (i.e. only frame 1, 4, 7, ... are processed), $d_{\min} = 12$ (short tracks with length below $d_{\min}$ were removed), $T_{dpm} = -0.4$, and linear interpolation between detections were produced to report results.

Tracking evaluation

We use the metrics (and evaluation tool) of the MOT challenge. Please refer to [172] for details. In general, except the detection filtering, results (MOTA) were not affected much by parameters changes.

Detection filtering. Tab. A.1 reports the metrics at detection level and tracking level when applying the linear height filtering and with different detection threshold T_{dpm}. The filter gives a

	Raw detection	Filtered detection		
	$T_{dpm} = -0.5$	$T_{dpm} = -0.5$	$T_{dpm} = -0.4$	$T_{dpm} = -0.3$
Detection Recall	35.4	35.1	34	32.4
Detection Precision	78.3	86.1	89.9	92.4
MOTA	25.2	29.1	29.8	29.3
MOTP	74.1	74.2	74.3	74.6

Table A.1 – Detection filtering. Detection precision-recall and tracking performance (nota: tracks are not interpolated in results).

Parameters	Rec.	Pre.	FAR	MT	PT	ML	IDS	FM	MOTA	MOTP
Default	38.7	85.9	1.32	49	180	288	211	634	32.1	74.7
T_w =12	35.9	90.5	0.78	39	181	297	275	636	31.9	75.1
Δ_{sk} =1	40.5	82.8	1.74	52	188	277	273	1199	31.8	73.7
Δ_{sk} =3	38.7	85.9	1.32	49	180	288	211	634	32.1	74.7
Δ_{sk} =6	35.5	88.9	0.92	33	177	307	217	459	30.8	75.1
Unsup. models	38.0	86.6	1.22	43	183	291	237	692	31.9	74.7
W/o match. sim.	36.6	89.5	0.89	48	157	312	210	555	32.2	75
With match. sim.	37.2	88.8	0.98	49	161	307	203	638	32.3	74.8

Table A.2 – Evaluation of our tracking framework with various configurations. Results with the default parameters are shown first, and then performance obtained when varying one of the parameters (provided in first column) are provided.

boost in precision with a small decrease in recall and all tracking metrics are improved thanks to fewer false alarms. We can also observe that threshold $T_{dpm} = -0.4$ provides an appropriate trade-off between precision and recall and good tracking performance.

Tracking window T_w and step size Δ_{sk}. Different configurations are reported in Tab. A.2. One can observe that with longer tracking context T_w (default $T_w = 24$ vs shorter $T_w = 12$), tracks are more likely to recover from temporary occlusions or missed detections, resulting in higher MT, ML. When detector is applied scarcely (e.g. $\Delta_{sk} = 3$ or 6), we observe a performance decrease (e.g. decrease of MT, increase of ML). Nevertheless, applying the detection every $\Delta_{sk} = 3$ frames reduces the false alarms and improves IDS and FM metrics. Since detection is one of the computation bottlenecks, this provides a good trade-off between performance and speed. When $\Delta_{sk} = 3$, the overall tracking speed also is increased by up to 6 times.

Supervised vs unsupervised models. The "Unsup. model's' line in Tab. A.2 provides the results when using association models trained from the raw detection *in an unsupervised fashion as in [2]*, which can be compared against of the default ones obtained using tracking models trained from the labeled GT boxes provided in *MOTChallenge* 2016. Interestingly, although the *unsupervised* approach suffer from missing detections and unstable bounding boxes, it performs

	FAR	MT	ML	IDS	FM	MOTA	MOTP
LTTSC_CRF	2.0	9.6 %	55.2 %	481	1012	37.6	75.9

Table A.3 – Results on the MOT 2016 test data.

very close to the supervised models in most tracking metrics.

Matching similarity. Because of the complexity, we used $T_w = 15$ for sequence MOT16-04, the rest use the default parameters. Although SURF matching can be discriminative for objects, it is less effective in human tracking because of clothing similarity, and data resolution where most features are found on human boundaries rather than within. This is reflected in Tab. A.2, where only minor improvement in IDS, ML, MT, and PT are observed. In future work, better tracking oriented cues could be used, such as those developed for re-identification.

Evaluation on test sequences

The results of the method configured with detection filtering and the default parameters for the tracker are reported in Tab. A.3. Overall, the performance are better, showing that the method generalizes well (with its limitation) and qualitative results are aligned with those of the training sequences. The comparison with other trackers can be found in the MOT website[1]. Overall, our tracker achieved fair ranking in comparison to other methods. Considering methods based on the public detections, our tracker exhibit a good precision (rank $5^{th}/20$ on the IDS metric and $8^{th}/20$ on Frag metric) but is penalized by a low recall, resulting on a ranking of $11^{th}/20$ for MOTA. It is important to note that our modeling framework was taken as is from previous paper, and not adapted or over-tuned to the MOT challenge (e.g. for camera motion or viewpoints). In addition, as our framework can leverage any cue in a time-sensitive fashion, other state-of-the-art features like those based on supervised re-identification learning can be exploited and would positively impact performance.

[1] https://motchallenge.net/results/MOT16/

B Appendix for Chapter 3

Results in the MediaEval Challenge 2015

The task is described in details in [173]. We evaluated 3 methods: SpkDia, FaceDia, and SpkFace. In *SpkDia* (primary submission), we apply naming based on audio information only (this is equivalent to assumption that all speakers which are associated with a name are visible and speaking). This is our primary submission for the challenge. Second, in *FaceDia*, we apply naming based on visual information only, and assume that all visible faces (which are associated with a name) are talking. Third, in *SpkFace*, we apply naming based on audio information only, but validate if there exists visible faces during the speech segments (if not, the segment is discarded). Because our approaches are monomodal and fully unsupervised, we did not use the information provided by leaderboard to improve performance.

The results using the challenge performance measures are reported in Tab. B.1 for the REPERE test 2 data [44] as the initial development data and in Tab. B.2 for the challenge testing part of the INA dataset. SpkDia is the most robust and performs the best even without any face information, which might be explained by two points. First, there is usually only one speaker at a time, and not much noise in the challenge data. Meanwhile, face diarization can be difficult due to multiple faces, facial variation, missed detections, etc. Hence, speech clusters tend to be more reliable than face clusters. Second, when a speaker is not visible, it is often the anchor of the show, who is counted as one query equally to those appearing for short duration. Therefore, SpkDia is not penalized much by the visibility of speakers. We can observe this effect more in the last column of Tab. B.2 which shows the number of person presence with names predicted by each scheme. Using faces to filter 1/3 of speech segments does not help to increase precision because these

Method	EwMAP	MAP	C	#(2485)
Baseline	49.98	50.32	58.75	617
SpkDia	**65.31**	66.70	72.50	2817
FaceDia	**66.38**	67.98	71.67	1691

Table B.1 – Results on REPERE test 2 (dev set)

Method	EwMAP	MAP	C	#(21963)
Baseline	78.35	78.64	92.71	12066
FaceDia	83.04	83.33	90.77	7237
SpkDia*	**89.75**	**90.14**	**97.05**	30583
SpkFace	89.53	89.90	96.52	20601
* Primary submission				

Table B.2 – Results on INA (test set)

segments correspond to a small number of repetitive speakers. Also, though face diarization gives only 1/3 of possible names, these names are precise person-wise. This interesting fact may provide outlook on combining 2 modalities.

C Appendix for Chapter 6

Proof of Proposition 2

Recall that the expected risk unhinged triplet loss under noise $\hat{R}_{l^U}(\theta)$ is:

$$\hat{R}_{l^U}(\theta) = \mathbb{E}_{x_i,x_j,t_{ij}}\left[Q_{t_{ij}} w_{t_{ij}} l^A(x_i,x_j,t_{ij},\theta) + w_{-t_{ij}} q_{t_{ij}} d_{max}\right]. \tag{C.1}$$

Let θ^* be the optimizer of the clean risk $R_{l^U}(\theta)$, which gives us:

$$R_{l^U}(\theta^*) - R_{l^U}(\theta) \leq 0 \qquad \forall\theta. \tag{C.2}$$

We consider the same θ^* in the noisy risk $\hat{R}_{l^U}(\theta^*) - \hat{R}_{l^U}(\theta)$, and then apply 6.15:

$$\hat{R}_{l^U}(\theta^*) - \hat{R}_{l^U}(\theta) = \mathbb{E}_{x_i,x_j,t_{ij}}\left[Q_{t_{ij}}\left(w_{t_{ij}} l^A(x_i,x_j,t_{ij},\theta^*) - w_{t_{ij}} l^A(x_i,x_j,t_{ij},\theta)\right)\right]. \tag{C.3}$$

The set of pairs (i,j) can be divided into the positive pairs, $t_{ij}=1$, and negative pairs, $t_{ij}=-1$. Hence the risk difference can be also split into:

$$\begin{aligned}
\hat{R}_{l^U}(\theta^*) - \hat{R}_{l^U}(\theta) &= p(t_{ij}=1)\,\mathbb{E}_{x_i,x_j,t_{ij}=1}\left[Q_{+1} w_{t_{ij}}\left(l^A(x_i,x_j,t_{ij},\theta^*) - l^A(x_i,x_j,t_{ij},\theta)\right)\right] \\
&\quad + p(t_{ij}=-1)\,\mathbb{E}_{x_i,x_j,t_{ij}=-1}\left[Q_{-1} w_{t_{ij}}\left(l^A(x_i,x_j,t_{ij},\theta^*) - l^A(x_i,x_j,t_{ij},\theta)\right)\right] \\
&= Q_{+1}\,p(t_{ij}=1)\,\mathbb{E}_{x_i,x_j,t_{ij}=1}\left[w_{t_{ij}}\left(l^A(x_i,x_j,t_{ij},\theta^*) - l^A(x_i,x_j,t_{ij},\theta)\right)\right] \\
&\quad + Q_{-1}\,p(t_{ij}=-1)\,\mathbb{E}_{x_i,x_j,t_{ij}=-1}\left[w_{t_{ij}}\left(l^A(x_i,x_j,t_{ij},\theta^*) - l^A(x_i,x_j,t_{ij},\theta)\right)\right].
\end{aligned} \tag{C.4}$$

From the condition, we have θ^* is also the minimizer of $\mathscr{S}^+$ and $\mathscr{S}^-$, or:

$$\begin{aligned}
\mathbb{E}_{x_i,x_j,t_{ij}=1}\left[l^A(x_i,x_j,t_{ij},\theta^*) - l^A(x_i,x_j,t_{ij},\theta)\right] \leq 0, \\
\mathbb{E}_{x_i,x_j,t_{ij}=-1}\left[l^A(x_i,x_j,t_{ij},\theta^*) - l^A(x_i,x_j,t_{ij},\theta)\right] \leq 0.
\end{aligned} \tag{C.5}$$

Using this fact to upper bound Eq. C.3 we can come to:

$$\begin{aligned}
&\hat{R}_{l^U}(S,\theta^*) - \hat{R}_{l^U}(S,\theta) \\
&\quad \leq \min_{t_{ij}}\{Q_{t_{ij}}\}\Big[p(t_{ij}=1)\,\mathbb{E}_{x_i,x_j,t_{ij}=1}\Big[w_{t_{ij}}\Big(l^A(x_i,x_j,t_{ij},\theta^*) - l^A(x_i,x_j,t_{ij},\theta)\Big)\Big] \\
&\quad + p(t_{ij}=-1)\,\mathbb{E}_{x_i,x_j,t_{ij}=-1}\Big[w_{t_{ij}}\Big(l^A(x_i,x_j,t_{ij},\theta^*) - l^A(x_i,x_j,t_{ij},\theta)\Big)\Big]\Big],
\end{aligned} \tag{C.6}$$

or:

$$\hat{R}_{l^U}(S,\theta^*) - \hat{R}_{l^U}(S,\theta) \leq \min_{t_{ij}}\{Q_{t_{ij}}\}\Big(R_{l^U}(S,\theta^*) - R_{l^U}(S,\theta)\Big). \tag{C.7}$$

This upper bound in Eq. C.7 is reached when the following condition is satisfied:

$$\min_{t_{ij}}\{Q_{t_{ij}}\} = \min_{t_{ij}}\Big(1 - q_{t_{ij}} - q_{t_{ij}}\frac{w_{-t_{ij}}}{w_{t_{ij}}}\Big) \geq 0. \tag{C.8}$$

From C.2 and C.7, we have:

$$\hat{R}_{l^U}(\theta^*) - \hat{R}_{l^U}(\theta) \leq Q(R_{l^U}(\theta^*) - R_{l^U}(\theta)) \leq 0. \tag{C.9}$$

Hence, θ^* will also be the minimizer of the noisy risk $\hat{R}_{l^U}(S,\theta)$ if the condition Eq. 6.18 is met. This concludes the proof.

Clustering experiment

In the clustering tasks, we use the Normalized Mutual Information (NMI) metrics to quantify the clustering quality. $NMI = I(\Omega,C)/\sqrt{H(\Omega)H(C)}$, with $C = c_1,...,c_n$ being the clustering alignments, and $\Omega = \omega_1,...,\omega_n$ being the given groundtruth clusters (ie. class labels). Here $I(\text{ů},\text{ů})$ and $H(\text{ů})$ denotes mutual information and entropy respectively. We use K-means algorithm for clustering.

Because measuring clustering quality takes into account all nearby neighbors instead of just the nearest one, the difference in NMI between methods are narrower than in Rec@1. Still, the results in the clustering task agree with our conclusions from the image retrieval task. Using random semi-hard mining with triplet loss is more robust to label noise than fixed semi-hard mining and good minimziation helps to make marginal loss more robust to label noise.

In SOP dataset (Fig. C.1), the deterioration of triplet loss with fixed semi-hard mining increases after 40% while random semi-hard mining still retains good relative performance after 50%. Meanwhile marginal loss degrades much faster than both versions of triplet loss. The difference is easier to view when we compare the ratios between the NMI under noise and the NMI with few data.

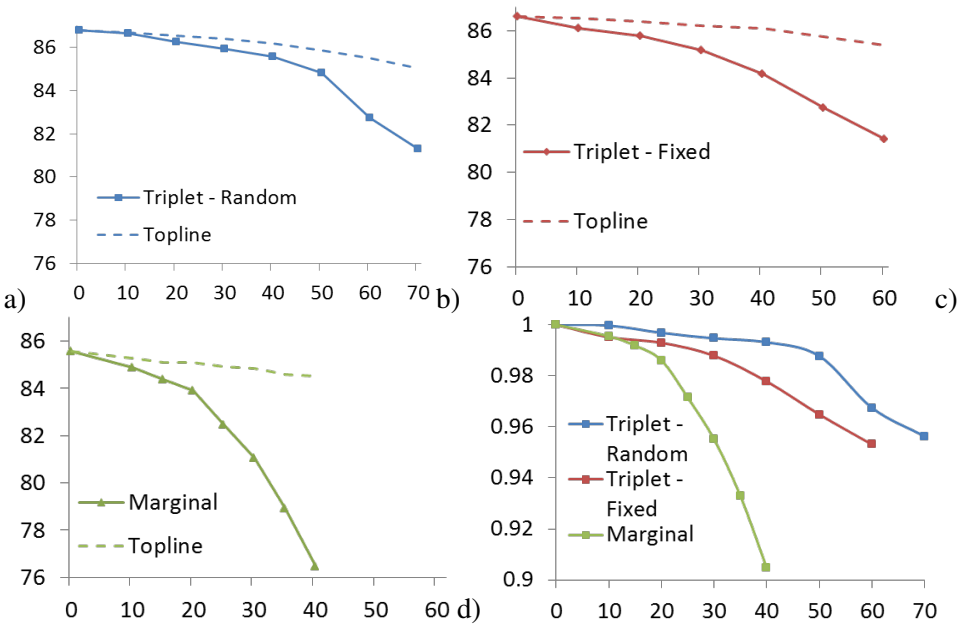


Figure C.1 – Clustering results reported on Standford Online Products dataset. x-axis: noise rate p, y-axis: NMI.(a-c) NMI of triplet loss with random semi-hard mining, fixed semi-hard mining, and marginal loss with random semi-hard mining, respectively. (d) the ratio of NMI for noise rate p over NMI when there are $1-p$ data samples (topline) for all three cases.

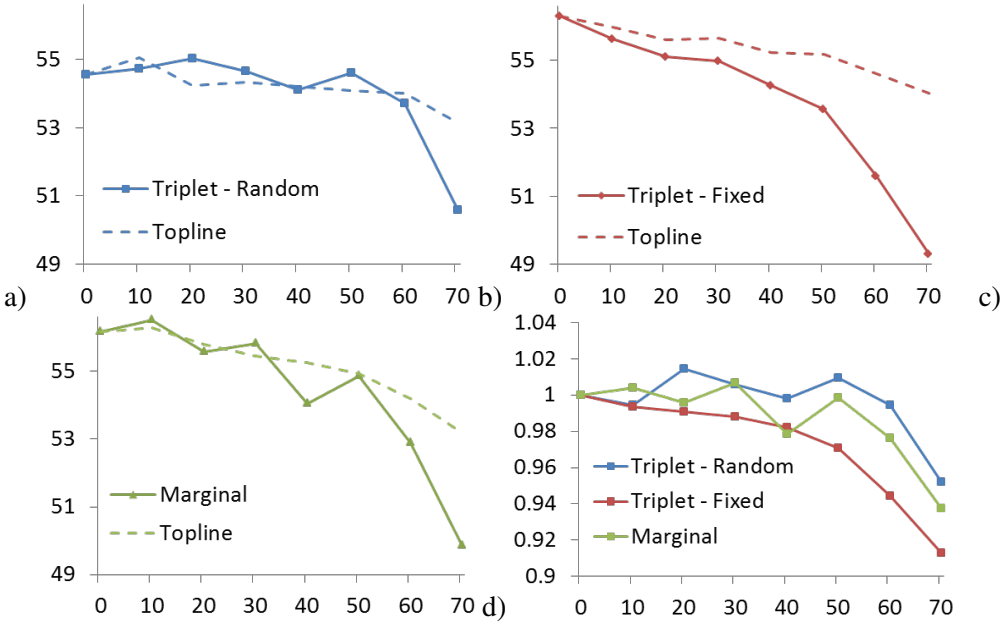


Figure C.2 – Clustering results reported on CUB-200-2011 birds dataset. x-axis: noise rate p, y-axis: NMI.(a-c) NMI of triplet loss with random semi-hard mining, fixed semi-hard mining, and marginal loss with random semi-hard mining, respectively. (d) the ratio of NMI for noise rate p over NMI when there are $1-p$ data samples (topline) for all three cases.

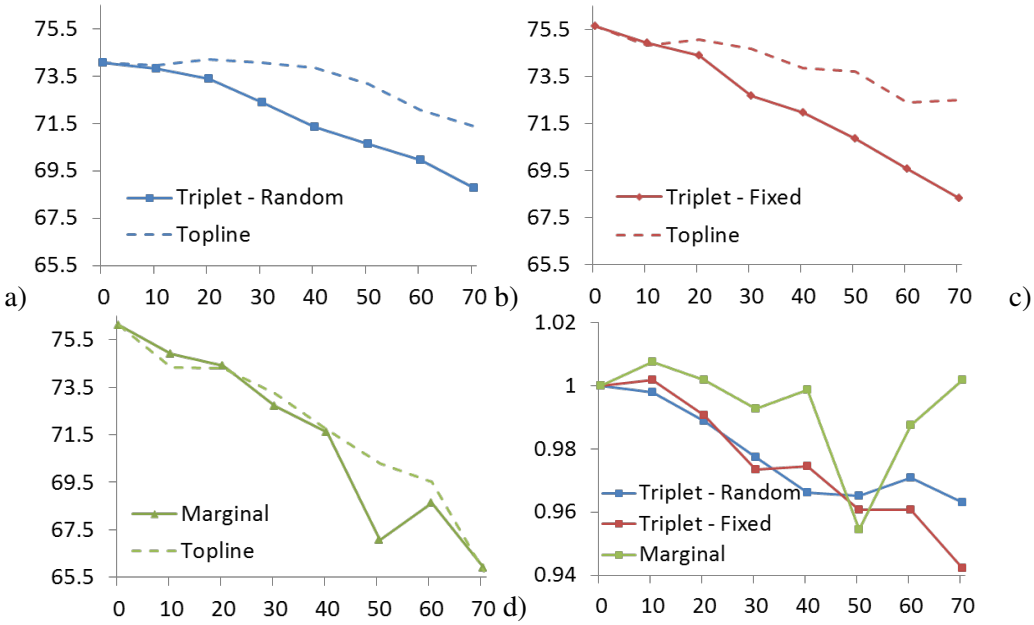


Figure C.3 – Clustering results reported on Oxford-102 flowers dataset. x-axis: noise rate p, y-axis: NMI.(a-c) NMI of triplet loss with random semi-hard mining, fixed semi-hard mining, and marginal loss with random semi-hard mining, respectively. (d) the ratio of NMI for noise rate p over NMI when there are $1-p$ data samples (topline) for all three cases.

In CUB and Flower datasets (Fig. C.2 and Fig. C.3), we observe the same difference between triplet loss with fixed or random semi-hard mining. On the other hand,marginal loss is relatively as robust as triplet loss with random semi-hard mining. This fact, as shown in the paper, can be contributed by initialization with pretrained models.

Bibliography

[1] P. Gay, G. Dupuy, C. Lailler, J.-M. Odobez, S. Meignier, and P. Deléglise, "Comparison of two methods for unsupervised person identification in tv shows," in *Content-Based Multimedia Indexing (CBMI), 2014 12th International Workshop on*, pp. 1–6, IEEE, 2014.

[2] A. Heili, A. Lopez-Mendez, and J. M. Odobez, "Exploiting Long-Term Connectivity and Visual Motion in CRF-based Multi-Person Tracking," *Tran. on Image Processing*, 2014.

[3] E. El Khoury, *Unsupervised video indexing based on audiovisual characterization of persons*. PhD thesis, Université Paul Sabatier-Toulouse III, 2010.

[4] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *CVPR*, IEEE, 2016.

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

[6] H. Muckenhirn, S. Marcel, *et al.*, "Towards directly modeling raw speech signal for speaker verification using cnns," in *ICASSP*, IEEE, 2017.

[7] N. Le, A. Heili, D. Wu, and J.-M. Odobez, "Temporally subsampled detection for accurate and efficient face tracking and diarization," in *International Conference on Pattern Recognition*, IEEE, Dec. 2016.

[8] N. Le, A. Heili, and J.-M. Odobez, "Long-term time-sensitive costs for CRF-based tracking by detection," in *European Conference on Computer Vision Workshops*, pp. 43–51, Springer, 2016.

[9] N. Le and J.-M. Odobez, "Learning multimodal temporal representation for dubbing detection in broadcast media," in *ACM Multimedia*, ACM, Oct. 2016.

[10] N. Le, D. Wu, S. Meignier, and J.-M. Odobez, "EUMSSI team at the mediaeval person discovery challenge," in *MediaEval 2015 Workshop*, 2015.

[11] N. Le, S. Meignier, and J.-M. Odobez, "EUMSSI team at the mediaeval person discovery challenge 2016," in *MediaEval Benchmarking Initiative for Multimedia Evaluation*, 2016.

[12] N. Le, H. Bredin, G. Sargent, P. Lopez-Otero, C. Barras, C. Guinaudeau, G. Gravier, G. B. da Fonseca, I. L. Freire, Z. Patrocínio Jr, *et al.*, "Towards large scale multimedia indexing: A case study on person discovery in broadcast news," in *Proceedings of the 15th International Workshop on Content-Based Multimedia Indexing*, p. 18, ACM, 2017.

[13] N. Le and J.-M. Odobez, "Robust and discriminative speaker embedding via intra-class distance variance regularization," *Proc. Interspeech 2018*, pp. 2257–2261, 2018.

[14] N. Le and J.-M. Odobez, "A domain adaptation approach to improve speaker turn embedding using face representation," in *Proceedings of the 19th ACM International Conference on Multimodal Interaction*, pp. 411–415, ACM, 2017.

[15] N. Le and J.-M. Odobez, "Improving speaker turn embedding by crossmodal transfer learning from face embedding," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, pp. 428–437, 2017.

[16] N. Le and J.-M. Odobez, "Improving speech embedding using crossmodal transfer learning with audio-visual data," *Multimedia Tools and Applications*, pp. 1–24, 2018.

[17] N. Le and J.-M. Odobez, "Theoretical guarantees of deep embedding losses under label noise," *arXiv preprint arXiv:1812.02676*, 2018.

[18] O. M. Parkhi, K. Simonyan, A. Vedaldi, and A. Zisserman, "A compact and discriminative face track descriptor," in *Computer Vision and Pattern Recognition (CVPR), 2014 IEEE Conference on*, pp. 1693–1700, IEEE, 2014.

[19] E. Khoury, P. Gay, and J.-M. Odobez, "Fusing Matching and Biometric Similarity Measures for Face Diarization in Video," in *ACM ICMR*, 2013.

[20] M. Everingham, J. Sivic, and A. Zisserman, "Taking the bite out of automated naming of characters in TV video," *Image and Vision Computing*, pp. 545–559, 2009.

[21] P. Viola and M. Jones, "Rapid object detection using a boosted cascade of simple features," in *Computer Vision and Pattern Recognition, 2001. CVPR 2001. Proceedings of the 2001 IEEE Computer Society Conference on*, vol. 1, pp. I–511, IEEE, 2001.

[22] A. Ozerov, J.-R. Vigouroux, L. Chevallier, and P. Perez, "On evaluating face tracks in movies," in *Int. Conf. on Image Processing*, 2013.

[23] M. Tapaswi, O. M. Parkhi, E. Rahtu, E. Sommerlade, R. Stiefelhagen, and A. Zisserman, "Total Cluster: A person agnostic clustering method for broadcast videos," in *ICVGIP*, 2014.

[24] M. Roth, M. Bauml, R. Nevatia, and R. Stiefelhagen, "Robust multi-pose face tracking by multi-stage tracklet association," in *21st International Conference on Pattern Recognition (ICPR)*, pp. 1012–1016, IEEE, 2012.

[25] C. Dubout and F. Fleuret, "Exact acceleration of linear object detectors," in *Computer Vision–ECCV 2012*, pp. 301–311, Springer, 2012.

[26] R. G. Cinbis, J. Verbeek, and C. Schmid, "Unsupervised metric learning for face identification in {TV} video," in *Proceedings of the IEEE International Conference on Computer Vision*, 2011.

[27] E. Khoury, L. El Shafey, C. McCool, M. Günther, and S. Marcel, "Bi-modal biometric authentication on mobile phones in challenging conditions," *Image and Vision Computing*, vol. 32, no. 12, pp. 1147–1160, 2014.

[28] R. Wallace and M. McLaren, "Total variability modelling for face verification," *Biometrics, IET*, vol. 1, no. 4, pp. 188–199, 2012.

[29] M. Mathias, R. Benenson, M. Pedersoli, and L. Van Gool, "Face detection without bells and whistles," *ECCV*, pp. 720–735, 2014.

[30] S. Yang, P. Luo, C.-C. Loy, and X. Tang, "From facial parts responses to face detection: A deep learning approach," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 3676–3684, 2015.

[31] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1627–1645, 2010.

[32] B. D. Lucas, T. Kanade, *et al.*, "An iterative image registration technique with an application to stereo vision.," in *IJCAI*, vol. 81, pp. 674–679, 1981.

[33] B. Wu, S. Lyu, B.-G. Hu, and Q. Ji, "Simultaneous clustering and tracklet linking for multi-face tracking in videos," in *IEEE International Conference on Computer Vision (ICCV)*, pp. 2856–2863, IEEE, 2013.

[34] M. Bauml, M. Tapaswi, and R. Stiefelhagen, "Semi-supervised Learning with Constraints for Person Identification in Multimedia Data," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2013.

[35] M. Rouvier, B. Favre, M. Bendris, D. Charlet, and G. Damnati, "Scene understanding for identifying persons in TV shows: beyond face authentication," in *Content-Based Multimedia Indexing (CBMI), 2014 12th International Workshop on*, 2014.

[36] A. Heili and J.-M. Odobez, "Parameter Estimation and Contextual Adaptation for a Multi-Object Tracking CRF Model," in *IEEE Workshop on Performance Evaluation of Tracking and Surveillance (PETS), Clearwater,*, 2013.

[37] J.-M. Odobez and P. Bouthemy, "Robust multiresolution estimation of parametric motion models," *Journal of visual communication and image representation*, vol. 6, no. 4, pp. 348–365, 1995.

[38] H. Bay, T. Tuytelaars, and L. Van Gool, "SURF: Speeded up robust features," in *Computer Vision–ECCV 2006*, Springer, 2006.

[39] M. Tapaswi, C. Corez, M. Bauml, H. Ekenel, and R. Stiefelhagen, "Cleaning up after a face tracker: False positive removal," in *IEEE ICIP*, 2014.

[40] X. Tan and W. Triggs, "Enhanced local texture feature sets for face recognition under difficult lighting conditions," *IEEE transactions on image processing*, vol. 19, no. 6, pp. 1635–1650, 2010.

[41] P. Kenny, P. Ouellet, N. Dehak, V. Gupta, and P. Dumouchel, "A study of interspeaker variability in speaker verification," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 16, no. 5, pp. 980–988, 2008.

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

[43] E. Maggio, E. Piccardo, C. Regazzoni, and A. Cavallaro, "Particle phd filtering for multi-target visual tracking," in *IEEE International Conference on Acoustics, Speech and Signal Processing*, vol. 1, pp. I–1101, IEEE, 2007.

[44] A. Giraudel, M. Carré, V. Mapelli, J. Kahn, O. Galibert, and L. Quintard, "The REPERE corpus: a multimodal corpus for person recognition.," in *LREC*, 2012.

[45] Y. Li, C. Huang, and R. Nevatia, "Learning to associate: Hybridboosted multi-target tracker for crowded scene," in *IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2953–2960, IEEE, 2009.

[46] X. Zhu and D. Ramanan, "Face detection, pose estimation, and landmark localization in the wild," in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, pp. 2879–2886, IEEE, 2012.

[47] M. Rouvier, G. Dupuy, P. Gay, E. Khoury, T. Merlin, and S. Meignier, "An open-source state-of-the-art toolbox for broadcast news diarization," in *Interspeech*, (Lyon (France)), 25-29 Aug. 2013.

[48] M. Everingham, J. Sivic, and A. Zisserman, "Hello! my name is... Buffy–automatic naming of characters in TV video," in *BMVC*, 2006.

[49] M. Bendris, D. Charlet, and G. Chollet, "Lip activity detection for talking faces classification in TV-Content," in *ICMV*, 2010.

[50] F. Patrona, A. Iosifidis, A. Tefas, N. Nikolaidis, and I. Pitas, "Visual voice activity detection in the wild," *Transactions on Multimedia*, 2016.

[51] P. Gay, E. Khoury, S. Meignier, J.-M. Odobez, and P. Deleglise, "A Conditional Random Field approach for Audio-Visual people diarization," in *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP 2014)*, 2014.

[52] G. Potamianos, C. Neti, G. Gravier, A. Garg, and A. W. Senior, "Recent advances in the automatic recognition of audiovisual speech," in *Proceedings of the IEEE*, pp. 1306–1325, 2003.

[53] A. Morris, A. Hagen, H. Glotin, and H. Bourlard, "Multi-stream adaptive evidence combination for noise robust asr," *Speech Communication*, 2001.

[54] J. Ngiam, A. Khosla, M. Kim, J. Nam, H. Lee, and A. Y. Ng, "Multimodal deep learning," in *ICML*, 2011.

[55] E. K. Patterson, S. Gurbuz, Z. Tufekci, and J. N. Gowdy, "Cuave: A new audio-visual database for multimodal human-computer interface research," in *ICASSP*, IEEE, 2002.

[56] G. Chetty and M. Wagner, "Audio-visual multimodal fusion for biometric person authentication and liveness verification," in *NICTA-HCSNet Multimodal User Interaction Workshop*, 2006.

[57] E. A. Rúa, H. Bredin, C. G. Mateo, G. Chollet, and D. G. Jiménez, "Audio-visual speech asynchrony detection using co-inertia analysis and coupled hidden markov models," *Pattern Analysis and Applications*, 2009.

[58] J. Hershey and J. Movellan, "Audio-vision: Using audio-visual synchrony to locate sounds," in *NIPS*, 2000.

[59] J. W. Fisher and T. Darrell, "Speaker association with signal-level audiovisual fusion," *IEEE Trans. on Multimedia*, vol. 6, no. 3, pp. 406–413, 2004.

[60] H. J. Nock, G. Iyengar, and C. Neti, "Assessing face and speech consistency for monologue detection in video," in *ACM Multimedia*, 2002.

[61] G. Iyengar, H. J. Nock, and C. Neti, "Audio-visual synchrony for detection of monologues in video archives," in *ICME*, IEEE, 2003.

[62] S. Kumagai, K. Doman, T. Takahashi, D. Deguchi, I. Ide, and H. Murase, "Detection of inconsistency between subject and speaker based on the co-occurrence of lip motion and voice towards speech scene extraction from news videos," in *Multimedia (ISM), 2011 IEEE International Symposium on*, 2011.

[63] Y. Hu, J. S. Ren, J. Dai, C. Yuan, L. Xu, and W. Wang, "Deep multimodal speaker naming," in *ACM Multimedia*, 2015.

[64] J. S. Ren, Y. Hu, Y.-W. Tai, C. Wang, L. Xu, W. Sun, and Q. Yan, "Look, Listen and Learn - A Multimodal LSTM for Speaker Identification," in *AAAI Conference on Artificial Intelligence*, 2016.

[65] V. Kazemi and J. Sullivan, "One millisecond face alignment with an ensemble of regression trees," in *CVPR*, 2014.

[66] G. Farnebäck, "Two-frame motion estimation based on polynomial expansion," in *Image analysis*, Springer, 2003.

[67] L. Pigou, A. Van Den Oord, S. Dieleman, M. V. Herreweghe, and J. Dambre, "Beyond Temporal Pooling: Recurrence and Temporal Convolutions for Gesture Recognition in Video," *Arxiv*, pp. 1–9, 2015.

[68] H. Hotelling, "Relations between two sets of variates," *Biometrika*, vol. 28, no. 3/4, pp. 321–377, 1936.

[69] Y. Bengio, P. Simard, and P. Frasconi, "Learning long-term dependencies with gradient descent is difficult," *IEEE Trans. on Neural Networks*, 1994.

[70] S. Hochreiter, S. Hochreiter, J. Schmidhuber, and J. Schmidhuber, "Long short-term memory.," *Neural computation*, vol. 9, no. 8, pp. 1735–80, 1997.

[71] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to sequence learning with neural networks," in *NIPS*, 2014.

[72] N. Srivastava, E. Mansimov, and R. Salakhutdinov, "Unsupervised Learning of Video Representations using LSTMs," *Int. Conf. Machine Learning (ICML)*, 2015.

[73] C. B. H. Bredin, C. Guinaudeau, "Multimodal person discovery in broadcast tv at mediaeval 2016," in *Proc. of the MediaEval 2016 Workshop*, (Hilversum, Netherlands), Oct. 2016.

[74] D. Chen, J.-M. Odobez, and H. Bourlard, "Text detection and recognition in images and video frames," *Pattern Recognition*, vol. 37, no. 3, pp. 595–608, 2004.

[75] D. Chen and J.-M. Odobez, "Video text recognition using sequential monte carlo and error voting methods," *Pattern Recognition Letters*, vol. 26, no. 9, pp. 1386–1403, 2005.

[76] N. Daddaoua, J.-M. Odobez, and A. Vinciarelli, "Ocr based slide retrieval," in *Eighth International Conference on Document Analysis and Recognition (ICDAR'05)*, pp. 945–949, IEEE, 2005.

[77] A. Vinciarelli and J.-M. Odobez, "Application of information retrieval technologies to presentation slides," *IEEE Transactions on Multimedia*, vol. 8, no. 5, pp. 981–995, 2006.

[78] C. Dubout and F. Fleuret, "Deformable part models with individual part scaling," in *BMVC*, 2013.

[79] O. Galibert and J. Kahn, "The first official REPERE evaluation," in *Interspeech satellite workshop on Speech, Language and Audio in Multimedia (SLAM)*, (Marseille, France), 2013.

[80] M. Rouvier and S. Meignier, "A global optimization framework for speaker diarization," in *Odyssey Workshop*, (Singapore), 2012.

[81] G. Dupuy, S. Meignier, P. Deléglise, and Y. Estève, "Recent improvements towards ILP-based clustering for broadcast news speaker diarization," in *Odyssey-14*, 2014.

[82] C. Barras, X. Zhu, S. Meignier, and J. Gauvain, "Multi-stage speaker diarization of broadcast news," *IEEE-TSAP*, vol. 14, pp. 1505–1512, Feb. 2006.

[83] J. Poignant, H. Bredin, V.-B. Le, L. Besacier, C. Barras, and G. Quénot, "Unsupervised speaker identification using overlaid texts in tv broadcast," in *Interspeech*, p. 4p, 2012.

[84] G. Sargent, G. B. de Fonseca, I. L. Freire, R. Sicre, Z. Do Patrocínio Jr, S. Guimarães, and G. Gravier, "Puc minas and irisa at multimodal person discovery," in *Working Notes Proceedings of the MediaEval Workshop*, vol. 1739, 2016.

[85] V.-T. Nguyen, M.-T. H. Nguyen, Q.-H. Che, V.-T. Ninh, T.-K. Le, T.-A. Nguyen, and M.-T. Tran, "Hcmus team at the multimodal person discovery in broadcast tv task of mediaeval 2016.," in *MediaEval*, 2016.

[86] P. Lopez-Otero, L. D. Fernández, and C. Garcia-Mateo, "Gtm-uvigo system for multimodal person discovery in broadcast tv task at mediaeval 2016.," in *MediaEval*, 2016.

[87] F. Nishi, N. Inoue, K. Iwano, and K. Shinoda, "Tokyo tech at mediaeval 2016 multimodal person discovery in broadcast tv task.," in *MediaEval*, 2016.

[88] M. India, M. Juan Gerard, C. Cortillas, G. Bouritsas, E. C. Sayrol, J. R. R. Morros, and J. Hernando, "Upc system for the 2016 mediaeval multimodal person discovery in broadcast tv task," in *MediaEval*, 2016.

[89] D. Snyder, P. Ghahremani, D. Povey, D. Garcia-Romero, Y. Carmiel, and S. Khudanpur, "Deep neural network-based speaker embeddings for end-to-end speaker verification," in *Spoken Language Technology Workshop (SLT), 2016 IEEE*, pp. 165–170, IEEE, 2016.

[90] C. Zhang and K. Koishida, "End-to-end text-independent speaker verification with triplet loss on short utterances," in *Proc. of Interspeech*, 2017.

[91] 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.

[92] D. Snyder, D. Garcia-Romero, D. Povey, and S. Khudanpur, "Deep neural network embeddings for text-independent speaker verification," *INTERSPEECH*, 2017.

[93] E. Variani, X. Lei, E. McDermott, I. L. Moreno, and J. Gonzalez-Dominguez, "Deep neural networks for small footprint text-dependent speaker verification," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, pp. 4052–4056, IEEE, 2014.

[94] D. Snyder, D. Garcia-Romero, G. Sell, D. Povey, and S. Khudanpur, "X-vectors: Robust dnn embeddings for speaker recognition," *ICASSP, Calgary*, 2018.

[95] H. Bredin, "TristouNet: Triplet Loss for Speaker Turn Embedding," in *ICASSP*, (New Orleans, USA), IEEE, 2017.

[96] F. Schroff, D. Kalenichenko, and J. Philbin, "Facenet: A unified embedding for face recognition and clustering," *arXiv preprint arXiv:1503.03832*, 2015.

[97] R. Manmatha, C.-Y. Wu, A. J. Smola, and P. Krähenbühl, "Sampling matters in deep embedding learning," in *Computer Vision (ICCV), 2017 IEEE International Conference on*, pp. 2859–2867, IEEE, 2017.

[98] Y. Wen, K. Zhang, Z. Li, and Y. Qiao, "A discriminative feature learning approach for deep face recognition," in *European Conference on Computer Vision*, pp. 499–515, Springer, 2016.

[99] S. J. D. Prince and J. H. Elder, "Probabilistic linear discriminant analysis for inferences about identity," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 1–8, IEEE, 2007.

[100] S. Cumani, O. Plchot, and P. Laface, "Probabilistic linear discriminant analysis of i-vector posterior distributions," in *Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on*, pp. 7644–7648, IEEE, 2013.

[101] S. Madikeri, M. Ferras, P. Motlicek, and S. Dey, "Intra-class covariance adaptation in plda back-ends for speaker verification," in *Acoustics, Speech and Signal Processing (ICASSP), 2017 IEEE International Conference on*, pp. 5365–5369, IEEE, 2017.

[102] S. H. Ghalehjegh and R. C. Rose, "Deep bottleneck features for i-vector based text-independent speaker verification," in *Automatic Speech Recognition and Understanding (ASRU), 2015 IEEE Workshop on*, pp. 555–560, IEEE, 2015.

[103] Y. Lei, N. Scheffer, L. Ferrer, and M. McLaren, "A novel scheme for speaker recognition using a phonetically-aware deep neural network," in *Acoustics, Speech and Signal Processing (ICASSP), 2014 IEEE International Conference on*, pp. 1695–1699, IEEE, 2014.

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

[105] K. Q. Weinberger, J. Blitzer, and L. K. Saul, "Distance metric learning for large margin nearest neighbor classification," in *Advances in neural information processing systems*, pp. 1473–1480, 2006.

[106] D. Yi, Z. Lei, S. Liao, and S. Z. Li, "Learning face representation from scratch," *arXiv preprint arXiv:1411.7923*, 2014.

[107] 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*, vol. 4, no. 2, 2012.

[108] C. Ma, P. Nguyen, and M. Mahajan, "Finding speaker identities with a conditional maximum entropy model," in *ICASSP*, 2007.

[109] V. Jousse, S. Petit-Renaud, S. Meignier, Y. Esteve, and C. Jacquin, "Automatic named identification of speakers using diarization and {ASR} systems," in *ICASSP*, 2009.

[110] J. Poignant, L. Besacier, and G. Quénot, "Unsupervised Speaker Identification in {TV} Broadcast Based on Written Names," *IEEE/ACM Trans. on Audio, Speech, and Language Processing*, 2014.

[111] A. K. Sarkar, D. Matrouf, P.-M. Bousquet, and J.-F. Bonastre, "Study of the effect of i-vector modeling on short and mismatch utterance duration for speaker verification.," in *Interspeech*, 2012.

[112] P. Clément, T. Bazillon, and C. Fredouille, "Speaker diarization of heterogeneous web video files: A preliminary study," in *ICASSP*, IEEE, 2011.

[113] X. Bost and G. Linares, "Constrained speaker diarization of TV series based on visual patterns," in *Spoken Language Technology Workshop (SLT), 2014 IEEE*, IEEE, 2014.

[114] S. Dey, S. Madikeri, and P. Motlicek, "End-to-end text-dependent speaker verification using novel distance measures," *Proc. Interspeech 2018*, pp. 3598–3602, 2018.

[115] M. Bendris, B. Favre, D. Charlet, G. Damnati, and R. Auguste, "Multiple-view constrained clustering for unsupervised face identification in TV-broadcast," in *ICASSP)*, pp. 494–498, IEEE, 2014.

[116] H. Bredin and G. Gelly, "Improving speaker diarization of TV series using talking-face detection and clustering," in *ACM Multimedia*, pp. 157–161, ACM, 2016.

[117] O. M. Parkhi, A. Vedaldi, and A. Zisserman, "Deep face recognition," in *BMVC*, 2015.

[118] L. Leng, J. Zhang, G. Chen, M. K. Khan, and K. Alghathbar, "Two-directional two-dimensional random projection and its variations for face and palmprint recognition," in *International Conference on Computational Science and Its Applications*, pp. 458–470, Springer, 2011.

[119] L. Leng, J. Zhang, J. Xu, M. K. Khan, and K. Alghathbar, "Dynamic weighted discrimination power analysis in dct domain for face and palmprint recognition," in *Information and Communication Technology Convergence (ICTC), 2010 International Conference on*, pp. 467–471, IEEE, 2010.

[120] Y. Zheng, D. K. Pal, and M. Savvides, "Ring loss: Convex feature normalization for face recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5089–5097, 2018.

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

[122] A. Gretton, K. M. Borgwardt, M. Rasch, B. Schölkopf, and A. J. Smola, "A kernel method for the two-sample-problem," *NIPS*, 2007.

[123] A. Li, S. Shan, X. Chen, and W. Gao, "Cross-pose face recognition based on partial least squares," *Pattern Recognition Letters*, 2011.

[124] D. Hu, X. Lu, and X. Li, "Multimodal learning via exploring deep semantic similarity," in *ACM Multimedia*, 2016.

[125] V. E. Liong, J. Lu, Y.-P. Tan, and J. Zhou, "Deep coupled metric learning for cross-modal matching," *IEEE Transactions on Multimedia*, 2016.

[126] D. Dai, T. Kroeger, R. Timofte, and L. Van Gool, "Metric imitation by manifold transfer for efficient vision applications," in *CVPR*, IEEE, 2015.

[127] B. Bhattarai, G. Sharma, and F. Jurie, "CP-mtML: Coupled projection multi-task metric learning for large scale face retrieval," in *CVPR*, IEEE, 2016.

[128] D. Dai and L. Van Gool, "Unsupervised high-level feature learning by ensemble projection for semi-supervised image classification and image clustering," *arXiv preprint arXiv:1602.00955*, 2016.

[129] F. Zhuang, P. Luo, H. Xiong, Q. He, Y. Xiong, and Z. Shi, "Exploiting associations between word clusters and document classes for cross-domain text categorization," *Statistical Analysis and Data Mining*, 2011.

[130] M. Long, W. Cheng, X. Jin, J. Wang, and D. Shen, "Transfer learning via cluster correspondence inference," in *ICDM*, IEEE, 2010.

[131] M. Long, Y. Cao, J. Wang, and M. I. Jordan, "Learning transferable features with deep adaptation networks.," in *ICML*, pp. 97–105, 2015.

[132] A. Rozantsev, M. Salzmann, and P. Fua, "Beyond sharing weights for deep domain adaptation," *arXiv preprint arXiv:1603.06432*, 2016.

[133] M. Baktashmotlagh, M. Harandi, and M. Salzmann, "Distribution-matching embedding for visual domain adaptation," *Journal of Machine Learning Research*, 2016.

[134] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, 1997.

[135] G. Hinton, O. Vinyals, and J. Dean, "Distilling the knowledge in a neural network," *arXiv preprint arXiv:1503.02531*, 2015.

[136] I. Steinwart, "On the influence of the kernel on the consistency of support vector machines," *JMLR*, 2001.

[137] G. Gravier, G. Adda, N. Paulson, M. Carré, A. Giraudel, and O. Galibert, "The etape corpus for the evaluation of speech-based tv content processing in the french language," in *LREC*, 2012.

[138] M. Guillaumin, J. Verbeek, and C. Schmid, "Is that you? Metric learning approaches for face identification," in *Proceedings of the IEEE International Conference on Computer Vision*, 2009.

[139] S. Chen and P. Gopalakrishnan, "Speaker, environment and channel change detection and clustering via the Bayesian Information Criterion," in *Proc. DARPA Broadcast News Transcription and Understanding Workshop*, 1998.

[140] C. Barras, X. Zhu, S. Meignier, and J. Gauvain, "Multistage speaker diarization of broadcast news," *Audio, Speech, and Language Processing, IEEE Transactions on*, vol. 14, no. 5, pp. 1505–1512, 2006.

[141] A. Roy and S. Marcel, "Introducing crossmodal biometrics: Person identification from distinct audio & visual streams," in *BTAS*, IEEE, 2010.

[142] A. Nagrani, S. Albanie, and A. Zisserman, ""seeing voices and hearing faces: Cross-modal biometric matching"," in *"IEEE Conference on Computer Vision and Pattern Recognition"*, 2018.

[143] R. Hadsell, S. Chopra, and Y. LeCun, "Dimensionality reduction by learning an invariant mapping," in *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, CVPR 2006*, 2006.

[144] A. Iscen, G. Tolias, Y. Avrithis, and O. Chum, "Mining on manifolds: Metric learning without labels," *arXiv preprint arXiv:1803.11095*, 2018.

[145] A. Jansen, M. Plakal, R. Pandya, D. P. Ellis, S. Hershey, J. Liu, R. C. Moore, and R. A. Saurous, "Unsupervised learning of semantic audio representations," in *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 126–130, IEEE, 2018.

[146] F. Radenović, G. Tolias, and O. Chum, "Cnn image retrieval learns from bow: Unsupervised fine-tuning with hard examples," in *European conference on computer vision*, pp. 3–20, Springer, 2016.

[147] X. Wang and A. Gupta, "Unsupervised learning of visual representations using videos," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2794–2802, 2015.

[148] J. Yang, D. Parikh, and D. Batra, "Joint unsupervised learning of deep representations and image clusters," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 5147–5156, 2016.

[149] H. O. Song, Y. Xiang, S. Jegelka, and S. Savarese, "Deep metric learning via lifted structured feature embedding," in *Computer Vision and Pattern Recognition (CVPR), 2016 IEEE Conference on*, pp. 4004–4012, IEEE, 2016.

[150] J. Lee and S. Abu-El-Haija, "Large-scale content-only video recommendation," in *Computer Vision Workshop (ICCVW), 2017 IEEE International Conference on*, pp. 987–995, IEEE, 2017.

[151] D. Mahajan, R. Girshick, V. Ramanathan, K. He, M. Paluri, Y. Li, A. Bharambe, and L. van der Maaten, "Exploring the limits of weakly supervised pretraining," *arXiv preprint arXiv:1805.00932*, 2018.

[152] X. Wang, K. He, and A. Gupta, "Transitive invariance for selfsupervised visual representation learning," in *Proc. of Int'l Conf. on Computer Vision (ICCV)*, 2017.

[153] B. Frénay, A. Kabán, *et al.*, "A comprehensive introduction to label noise," in *ESANN*, 2014.

[154] J. Krause, B. Sapp, A. Howard, H. Zhou, A. Toshev, T. Duerig, J. Philbin, and L. Fei-Fei, "The unreasonable effectiveness of noisy data for fine-grained recognition," in *European Conference on Computer Vision*, pp. 301–320, Springer, 2016.

[155] T. Xiao, T. Xia, Y. Yang, C. Huang, and X. Wang, "Learning from massive noisy labeled data for image classification," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pp. 2691–2699, 2015.

[156] T. Liu and D. Tao, "Classification with noisy labels by importance reweighting," *IEEE Transactions on pattern analysis and machine intelligence*, vol. 38, no. 3, pp. 447–461, 2016.

[157] S. Sukhbaatar, J. Bruna, M. Paluri, L. Bourdev, and R. Fergus, "Training convolutional networks with noisy labels," *arXiv preprint arXiv:1406.2080*, 2014.

[158] I. Jindal, M. Nokleby, and X. Chen, "Learning deep networks from noisy labels with dropout regularization," in *Data Mining (ICDM), 2016 IEEE 16th International Conference on*, pp. 967–972, IEEE, 2016.

[159] A. Drory, S. Avidan, and R. Giryes, "On the resistance of neural nets to label noise," *arXiv preprint arXiv:1803.11410*, 2018.

[160] A. Ghosh, H. Kumar, and P. Sastry, "Robust loss functions under label noise for deep neural networks.," in *AAAI*, pp. 1919–1925, 2017.

[161] N. Natarajan, I. S. Dhillon, P. K. Ravikumar, and A. Tewari, "Learning with noisy labels," in *Advances in neural information processing systems*, pp. 1196–1204, 2013.

[162] C. Wah, S. Branson, P. Welinder, P. Perona, and S. Belongie, "The Caltech-UCSD Birds-200-2011 Dataset," Tech. Rep. CNS-TR-2011-001, California Institute of Technology, 2011.

[163] M.-E. Nilsback and A. Zisserman, "Automated flower classification over a large number of classes," in *Computer Vision, Graphics & Image Processing, 2008. ICVGIP'08. Sixth Indian Conference on*, pp. 722–729, IEEE, 2008.

[164] B. Harwood, B. Kumar, G. Carneiro, I. Reid, T. Drummond, *et al.*, "Smart mining for deep metric learning," in *Proceedings of the IEEE International Conference on Computer Vision*, pp. 2821–2829, 2017.

[165] W. Chen, X. Chen, J. Zhang, and K. Huang, "Beyond triplet loss: a deep quadruplet network for person re-identification," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, vol. 2, 2017.

[166] K. Sohn, "Improved deep metric learning with multi-class n-pair loss objective," in *Advances in Neural Information Processing Systems*, pp. 1857–1865, 2016.

[167] R. C. Bolles, J. B. Burns, M. Graciarena, A. Kathol, A. Lawson, M. McLaren, and T. Mensink, "Spotting audio-visual inconsistencies (SAVI) in manipulated video.," in *CVPR Workshops*, pp. 1907–1914, 2017.

[168] P. Korshunov and S. Marcel, "Speaker inconsistency detection in tampered video," in *2018 26th European Signal Processing Conference (EUSIPCO)*, pp. 2375–2379, IEEE, 2018.

[169] P. Korshunov and S. Marcel, "Deepfakes: a new threat to face recognition? assessment and detection," *arXiv preprint arXiv:1812.08685*, 2018.

[170] A. Nagrani, S. Albanie, and A. Zisserman, "Learnable PINs: Cross-modal embeddings for person identity," *arXiv preprint arXiv:1805.00833*, 2018.

[171] A. Nagrani, S. Albanie, and A. Zisserman, "Seeing voices and hearing faces: Cross-modal biometric matching," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2018.

[172] A. Milan, L. Leal-Taixe, I. Reid, S. Roth, and K. Schindler, "MOT16: A benchmark for multi-object tracking," *arXiv preprint arXiv:1603.00831*, 2016.

[173] J. Poignant, H. Bredin, and C. Barras, "Multimodal person discovery in broadcast tv at mediaeval 2015," *MediaEval*, 2015.

Nam Le (Lê Đỗ Hoàng Nam)

E-mail: **nle@idiap.ch**
Web: `sites.google.com/site/namdohoangle/`
Citizenship: **Vietnam**
Affiliation: **Idiap Research Institute, Martigny, Switzerland**

Research Interests

Computer vision, machine learning, face tracking and recognition, deep neural networks, unsupervised learning

Education

PhD Candidate in Electrical Engineering 01/2015 – 03/2019
EPFL, Switzerland
Thesis 'Multimodal person recognition in audio-visual streams'

MSc in Information & Communication Technology 09/2012 – 10/2014
University of Science, HCMC, Vietnam
Thesis 'Efficient Representation Learning Using Unsupervised Networks'

BSc in Information Technology 09/2008 – 09/2012
University of Science, HCMC, Vietnam
High distinction, Top 3/600, GPA 9.0/10

Experience

Research Assistant 01/2015 – present
Idiap Research Institute, Switzerland

Software Engineering Intern 04/2018 – 08/2018
Google, Zurich, Switzerland

Research Assistant 09/2012 – 10/2014
John von Neumann Institute, Vietnam

Teaching Assistant 09/2012 – 10/2014
University of Science, HCMC, Vietnam
Courses: Machine Learning, Scientific Method, Algorithms & Complexity

Software Engineering Intern 09/2011 – 11/2011
Software Engineering Lab, University of Science, HCMC, Vietnam

Publications

1. N. Le, J.M. Odobez. "Theoretical Guarantees of Deep Embedding Losses Under Label Noise", *arXiv preprint arXiv:1812.02676*, 2018.
2. N. Le, J.M. Odobez. "Improving speech embedding using crossmodal transfer learning with audio-visual data", *Multimedia Tools and Applications*, 2018.
3. N. Le, J.M. Odobez. "Robust and Discriminative Speaker Embedding via Intra-Class Distance Variance Regularization", In *Interspeech*, pp. 2257-2261. 2018.
4. N. Le, J.M. Odobez. "Improving speaker turn embedding by crossmodal transfer learning from face embedding", In *Computer Vision Workshop (ICCVW), 2017 IEEE International Conference on*, pp. 428-437. IEEE, 2017.
5. N. Le, J.M. Odobez. "A Domain Adaptation Approach to Improve Speaker Turn Embedding Using Face Representation", In *Proceedings of the 19th ACM International Conference on Multimodal Interaction* (pp. 411-415). ACM, 2017.
6. N. Le et al. "Towards Large Scale Multimedia Indexing: A Case Study on Person Discovery in Broadcast News", In *Proceedings of the 15th International Workshop on Content-Based Multimedia Indexing* (p. 18). ACM, 2017.
7. N. Le, J.M. Odobez. "Learning multimodal temporal representation for dubbing detection in broadcast media", In *Proceedings of the 2016 ACM on Multimedia Conference*. ACM, 2016.

8. N. Le, S. Meignier, J.M. Odobez. "EUMSSI team at the MediaEval Person Discovery Challenge 2016", *MediaEval, 2016* (Rank 1^{st}/7 teams)
9. N. Le, A. Heili, J.M. Odobez, *Long-Term Time-Sensitive Costs for CRF-Based Tracking by Detection*, ECCV Workshop 2016
10. N. Le, A. Heili, D. Wu, J.M. Odobez. "Temporally subsampled detection for accurate and efficient face tracking and diarization", *Pattern Recognition (ICPR), 2016 23rd International Conference on. IEEE, 2016.*
11. D. Wu, L. Pigou, P.J. Kindermans, N. Le, L. Shao, J. Dambre, J.M. Odobez. "Deep dynamic neural networks for multimodal gesture segmentation and recognition", *IEEE Transactions on Pattern Analysis and Machine Intelligence, IEEE, 2016*
12. N. Le, D. Wu, S. Meignier, J.M. Odobez. "EUMSSI team at the MediaEval Person Discovery Challenge", *MediaEval 2015* (Rank 1^{st}/9 teams).
13. A.T. Duong, H.T. Phan, N. Le, T.S. Tran. "A Hierarchical Approach for Handwritten Digit Recognition Using Sparse Autoencoder", *Issues and Challenges of Intelligent Systems and Computational Intelligence, Springer, 2014*
14. H.T. Phan, A.T. Duong, N. Le, T.S. Tran. "Hierarchical Sparse Autoencoder Using Linear Regression-based Features in Clustering for Handwritten Digit Recognition", *Image and Signal Processing and Analysis (ISPA), 2013 8th International Symposium on. IEEE, 2013.*
15. N. Le, M.T. Tran. "An Analysis of Inhibitory Pseudo-Interconnections in Unsupervised Neural Networks", *Sixth International Conference on Machine Vision (ICMV). International Society for Optics and Photonics, 2013.*
16. N. Le, M.T. Tran. "A Robust Unsupervised Feature Learning Framework Using Spatial Boosting Networks", *Machine Learning and Applications (ICMLA), 2013 12th International Conference on. Vol. 2. IEEE, 2013.*
17. N. Le, T.S. Tran, M.T. Tran. "Exploring Neighborhood Influence in Text Classification", *Knowledge and Systems Engineering (KSE), 2012 Fourth International Conference on. IEEE, 2012.*
18. N. Le, T.S. Tran, M.T. Tran. Individual Link Model for Text Classification, *International Conference on Information Processing and Management of Uncertainty in Knowledge-Based Systems. Springer, 2012.*

Skills

Languages: C++, Python, Matlab, C#. *Deep Learning libraries:* PyTorch, Theano.

Honors & Awards

Toshiba Corp. Scholarship for Excellent Graduate Students	2013
Vietnam Ministry of Education Award for Excellent Projects	2013
American Chamber of Commerce (Amcham) in Vietnam Scholarship	2011
ACM World Final Programming Contest in China: Honorable Mention	2010
ACM Asia Regional Programming Contest in Thailand: 3^{rd} Place	2009
ACM Asia Regional Programming Contest in the Philippines: 2^{nd} Place	2009
Vietnam National Informatics Collegiate Olympic: 3^{rd} Prize	2009
ACM Asia Regional Programming Contest in Vietnam: Honorable Mention	2008

Related Materials

AV transfer learning code: `gitlab.idiap.ch/software/CTL-AV-Identification`
EUMSSI demo with person identification in videos: `demo.eumssi.eu/demo/`
Presentation at MediaEval 2016: `youtube.com/watch?v=axt_PxhIrJ4`

Languages

English (full professional), French (elementary), and Vietnamese (native)

Last updated on April 3, 2019

**Ce document a été imprimé au Centre d'impression EPFL,
imprimerie climatiquement neutre, certifiée myClimate.**