Music Information Retrieval beyond Audio

A Vision-based Approach for Real-world Data

Bazzica, Alessio

DOI
10.4233/uuid:7ad1aef5-7ce1-4972-9443-9f66a5c727f6

Publication date
2017

Document Version
Final published version

Citation (APA)
Bazzica, A. (2017). *Music Information Retrieval beyond Audio: A Vision-based Approach for Real-world Data*. [Dissertation (TU Delft), Delft University of Technology]. https://doi.org/10.4233/uuid:7ad1aef5-7ce1-4972-9443-9f66a5c727f6

Music Information Retrieval beyond Audio

A Vision-based Approach for Real-world Data

Alessio Bazzica

Music Information Retrieval beyond Audio

A Vision-based Approach for Real-world Data

MUSIC INFORMATION RETRIEVAL BEYOND AUDIO

A VISION-BASED APPROACH FOR REAL-WORLD DATA

Proefschrift

ter verkrijging van de graad van doctor
aan de Technische Universiteit Delft,
op gezag van de Rector Magnificus prof. ir. K.C.A.M. Luyben,
voorzitter van het College voor Promoties,
in het openbaar te verdedigen op
vrijdag 15 december 2017 om 12.30 uur

door

Alessio BAZZICA

Master of Engineering in Computer Science, Università degli Studi di Firenze, Italië
geboren te Florence, Italië.

Dit proefschrift is goedgekeurd door de

promotor: Prof. dr. A. Hanjalic
copromotor: Prof. dr. C.C.S. Liem

Samenstelling promotiecommissie:

Rector Magnificus,	voorzitter
Prof. dr. A. Hanjalic,	Technische Universiteit Delft
Dr. C.C.S. Liem MMus	Technische Universiteit Delft
Onafhankelijke leden:	
Prof. dr. C. Witteveen	Technische Universiteit Delft
Prof. dr. C. Snoek	Qualcomm and Universiteit van Amsterdam
Prof. dr. G. Richard	Télécom ParisTech
Prof. dr. A. Del Bimbo	Università degli Studi di Firenze
Prof. dr. M. Müller	International Audio Laboratories Erlangen
Prof. dr. C.M. Jonker	Technische Universiteit Delft, reserve member

TU Delft Delft University of Technology

PHENICX | Performances as Highly Enriched aNd Interactive Concerts eXperiences

Keywords: music information retrieval, computer vision, cross-modal analysis

Printed by: Ridderprint BV

ISBN 978-94-6299-807-0

An electronic version of this dissertation is available at
http://repository.tudelft.nl/.

Propositions

accompanying the dissertation

MUSIC INFORMATION RETRIEVAL BEYOND AUDIO
A VISION-BASED APPROACH FOR REAL-WORLD DATA

by

Alessio BAZZICA

1. A video recording of a musical performance can be synchronized to a symbolic score without using the audio channel of the recording. [Chapters 3 and 4]
2. Investing in efficient semi-automatic face annotation in video is more productive than developing yet another more complex fully automatic solution. [Chapters 4 and 5]
3. Existing convolutional neural network models are not suitable to learn temporal patterns on videos at full temporal resolution. [Chapter 6]
4. Relying on vision-based Music IR approaches is critical in the case of non-scripted music. [Chapter 2]
5. Convolutional neural networks are just giant hash functions.
6. PhD candidates should not become independent researchers.
7. Giving enough room for failures in scientific publications is critical to the advancement of science.
8. Involving supervisors from industry should be compulsory for all PhD candidates in Computer Science.
9. Keeping personal and professional goals apart is the recipe for failure.
10. The most effective way to let North-EU colleagues loosen up is bringing tiramisù at work.

These propositions are regarded as opposable and defendable, and have been approved as such by the promotor prof. dr. A. Hanjalic.

Most people never ask, and that's what separates, sometimes, the people who do things from the people who just dream about them.

Steve Jobs

CONTENTS

SUMMARY

Digital music platforms have recently become the primary revenue stream for recorded music, making record labels and content owners increasingly interested in developing new digital features for their users. Besides listening to expert-curated playlists and automatically recommended music, users can also benefit from a more informative, non-linearly accessible experience accommodating multiple perspectives on the content. To give some examples of such enriched experiences, an alternative version of a piece can automatically be suggested. Users can skip throughout a long classical music piece guided by a visualization of its structure (e.g., movements, recurring themes). They can also switch viewpoints while watching a music video instead of sticking to the editor's choice.

Developing such features requires innovation of automated content-based methods that extract musical knowledge. Traditionally, Music Information Retrieval (Music IR) researchers have tackled this problem mostly from an audio-only perspective. Several works have however shown that other types of data, such as social tags, listening behaviors, and symbolic music scores, can largely improve the performance of audio-only algorithms, or even enable tasks that cannot be solved at all using audio alone.

In this thesis, we focus on the relatively unexplored field of *vision-based Music IR*, which studies how to analyze the visual channel accompanying a music recording in order to learn more about the music piece being performed. Several existing methods require obtrusive settings, such as 3D motion capture systems, which are not applicable in professional environments (e.g., during a live classical music concert). Other methods rely instead on favorable viewpoints, static cameras, and uniform backgrounds to simplify the musicians' movements analysis process. In both cases, the devised algorithms may not be suitable for commercial music platforms, especially those dealing with *real-world data* — i.e., *unstructured* and *unconstrained* music videos. We therefore consider tasks, algorithms and datasets with the real-world data challenges in mind, advancing the state-of-the-art in two ways: (i) we investigate how to process videos of a single musician aiming to extract musically relevant cues that can be exploited to solve existing, as well as new, Music IR problems, and (ii) we address the challenging case of large ensembles, proposing a way to possibly parse complex scenes and link musician-wise cues to identity and instrumental part annotations.

More in detail, this thesis first presents a global motion feature which aims to represent musicians' movements over time. While lightweight and instrument-generic, it shows limitations with camera motion. For this reason, we switch to detecting "playing/non-playing" (P/NP) labels, which can be guessed from different viewpoints and at different scales and they can be used to encode the instrumentation of a performance over time. We first show the value of such semantic feature by proving that it allows to roughly synchronize a symbolic music score to a performance recording. We then focus on the visual analysis of large classical music ensembles videos, presenting a semi-

automatic framework for P/NP annotation. The experiments show that video face clustering is a critical problem to solve; we therefore illustrate a novel method that exploits the *quasi-static scene* properties of classical music videos to generate better face clusters by relying on an automatically built map of the scene. Finally, we address the challenging problem of detecting note onsets for clarinetist videos as a case study for woodwind and brass instruments. We propose a novel convolutional network architecture based on multiple streams and absence of temporal pooling, aiming to capture the fine spatio-temporal information conveyed by finger movements.

Our proposed methods, outcomes, and envisioned applications show that real-world music videos are an unexploited asset rather than a problem to avoid. Furthermore, the light this thesis sheds on vision-based Music IR gives various indications on where future Computer Vision and Music IR research agendas can meet, bringing further innovation to the digital music platforms market.

SAMENVATTING

Digitale muziekplatformen zijn vandaag de dag een belangrijke inkomensbron voor opgenomen muziek. Hierdoor groeit de interesse van platenlabels en rechthebbenden om nieuwe digitale functionaliteiten aan hun klanten aan te bieden. Naast het luisteren naar door experts opgestelde afspeellijsten en automatisch aanbevolen muziek, kunnen muziekgebruikers ook profiteren van meer informatieve, niet-lineaire toegangsvormen, waarbij meerdere perspectieven op de inhoud worden ondersteund. Als voorbeeld van een mogelijke verrijkte ervaring kan bijvoorbeeld een alternatieve opname van een werk automatisch worden gesuggereerd. Met behulp van visualisaties van lange klassieke muziekwerken (bijvoorbeeld van delen of terugkerende thema's) kunnen gebruikers makkelijker door het werk heen stappen. Verder kunnen ze tijdens het bekijken van een muzikale video-opname van cameraperspectief veranderen, in plaats van dat ze zich moeten houden aan de keuze van de regisseur.

De ontwikkeling van dit soort functionaliteit vereist innovatie van automatische inhoudsgebaseerde analysemethodes om muzikale kennis uit signalen te verkrijgen. Onderzoekers die dit probleem in het onderzoeksgebied van *Music Information Retrieval (Music IR)* hebben bestudeerd, hebben traditioneel een sterke focus gehad op audio als exclusieve informatiebron. Verscheidene werken hebben echter aangetoond dat het meenemen van andere vormen van data, zoals sociale tags, luistergedrag, en symbolische informatie uit muziekpartituren, het succes van audiogebaseerde algoritmes sterk kan verbeteren, en zelfs oplossingen mogelijk kan maken die op grond van alleen audio niet denkbaar zouden zijn.

In deze dissertatie leggen we de focus op het relatief weinig verkende gebied van *beeldgebaseerde Music IR*. Hierbij wordt het visuele informatiekanaal bij een muziekopname geanalyseerd, om meer te leren over het muziekwerk dat wordt uitgevoerd. Verschillende bestaande methodes vereisen een opzet die hinderlijk kan zijn voor de uitvoerder, zoals systemen voor driedimensionale bewegingsvastlegging (*motion capture*), die niet in professionele omgevingen, zoals een live klassiek muziekconcert, kunnen worden toegepast. Andere methodes zijn afhankelijk van gunstige gezichtspunten, statische camera's, en uniforme achtergronden, om het proces van bewegingsanalyse van de musicus te vergemakkelijken. In beide gevallen zullen resulterende algoritmes niet geschikt zijn om toegepast te worden in commerciële muziekplatformen, met name platformen die te maken hebben met realistische data — d.w.z. *ongestructureerde* muziekvideo's met *onbegrensde* mogelijke variatie. Om deze reden houdt deze dissertatie zich bezig met taken, algoritmes en datasets gericht op realistische data-uitdagingen. We verbeteren de stand van zaken op twee manieren: (i) door te onderzoeken hoe video's van een enkele musicus kunnen worden geanalyseerd, om muzikaal relevante aanwijzingen te verkrijgen die kunnen worden gebruikt om bestaande en nieuwe Music IR-problemen op te lossen, en (ii) door de uitdaging aan te gaan om video's van grote ensembles te analyseren, waarbij we een methode voorstellen om complexe situaties te ontleden, en

muzikantgebaseerde aanwijzingen te linken aan annotaties van persoonsidentiteit en instrumentpartijen.

Allereerst zal de dissertatie een methode presenteren om de beweging van musici over tijd automatisch uit een video te halen en te representeren als een globaal bewegingskenmerk. De methode is computationeel licht en niet afhankelijk van specifieke instrumenten, maar toont beperkingen in geval van camerabeweging. Om deze reden stappen we over op het detecteren of iemand zijn/haar instrument bespeelt of niet (*"playing/non-playing"*, P/NP), wat vanuit meerdere gezichtspunten en op verschillende schalen kan worden bepaald, en gebruikt kan worden om instrumentatie van een uitvoering over tijd weer te geven. Allereerst tonen we de waarde van dit semantische kenmerk aan door te demonstreren dat dit een globale synchronisatie van een symbolische partituur met een opname van een uitvoering mogelijk maakt. Vervolgens focussen we op de beeldanalyse van video's van grote klassieke muziekensembles, waarbij we een semi-automatische aanpak voor P/NP-annotatie presenteren. De experimenten tonen aan dat het clusteren van gezichten in video's een kritiek probleem is om op te lossen. Daarom presenteren we een nieuwe methode, waarbij eigenschappen van *quasi-statische* situaties worden benut om betere clusters te genereren, gebruikmakend van een automatisch opgebouwde omgevingsplattegrond. Tot slot richten we ons op het uitdagende probleem om de aanvang van noten in video's van klarinettisten te detecteren, als een casestudy voor blaasinstrumenten. We stellen een nieuwe *convolutional neural network*-architectuur voor, gebaseerd op meerdere informatiestromen en afwezigheid van temporele pooling, waarbij het doel is om fijne spatiotemporele informatie uit vingerbewegingen te detecteren.

Onze voorgestelde methodes, resultaten, en voorziene applicaties geven aan dat realistische muziekvideo's een onontgonnen bron van waardevolle informatie zijn, in plaats van een probleem om te vermijden. Het licht dat deze dissertatie op beeldgebaseerde Music IR werpt, geeft bovendien verschillende aanwijzingen over waar toekomstige onderzoeksagenda's in de onderzoeksgebieden van *Computer Vision* en Music IR elkaar kunnen ontmoeten, om de markt van digitale muziekplatformen verder te kunnen innoveren.

1

INTRODUCTION

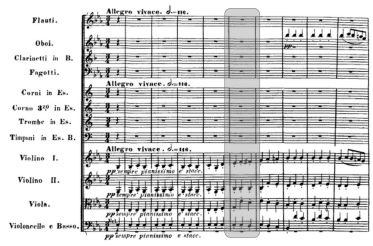
1.1. ONLINE MUSIC PLATFORMS: CONTINUOUS INNOVATION

Due to the large availability of digital platforms like Google Play, Spotify and YouTube, the way we experience music has been shifting. According to the IFPI Global Music Report of 2016[1], "*the global music market achieved a key milestone in 2015 when digital became the primary revenue stream for recorded music, overtaking sales of physical formats for the first time*". This fact indicates that, instead of promoting live events to encourage audiences to buy a recording afterwards, artists and record labels may benefit from promoting online music as first step in order to attract more people to live performances.

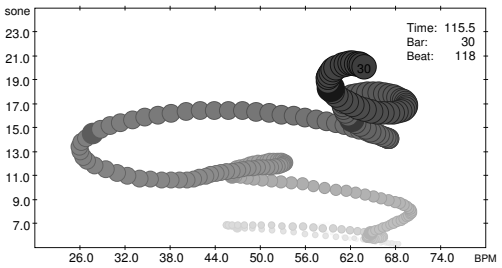
In order to let online music platforms become the key for reaching large audiences, service providers have become increasingly innovative in developing new features for such platforms. As an example, in services like VEVO[2] that are used to watch music videos online, playlists have become a popular feature. They can be automatically rec-

[1] http://ifpi.org/news/IFPI-GLOBAL-MUSIC-REPORT-2016
[2] http://www.vevo.com

Figure 1.1: RCO Editions tablet interface enabling the user to get informed about a music piece from different perspectives, e.g., through an explanation of a conductor, biographies of the composer and musicians, and by listening to the piece in various ways, e.g., by also following the music score.



(a) Score synchronization: the current bar is automatically highlighted.



(b) The Air Worm visualization: loudness and tempo are represented jointly over time [28].

Figure 1.2: Examples of new functionalities to enrich the user experience when interacting with music.

ommended, curated by experts or created by and shared among users of a social platform. More advanced features have been investigated as well. For instance, new emerging platforms like *RCO Editions*[3] and the *Berliner Philharmoniker's Digital Concert Hall*[4] aim to enrich audiovisual recordings of symphonic music performances to make them more informative, accessible in a non-linear fashion and from multiple perspectives [43] [66] (see, for example, Figure 1.1). This allows users to access material related to the specific performance which is being listened to, such as conductor commentaries, musicians' biographies, alternative performance recordings and synchronized sheet music [5] (see Figure 1.2a). Also, visualizations have been deployed to guide users through a performance, highlighting cues and structure like done for instance by the "Air Worm" [28] (see Figure 1.2b) and scape plots [69] respectively. Furthermore, non-linear access offers ways to skip to relevant moments based on temporal annotations. This is particularly useful for long pieces with a complex structure, which is often the case for classical music (see e.g., Figure 1.3). Finally, by allowing multiple perspectives, users are not obliged to stick to the editors' viewpoint anymore. Instead, they can, for instance, isolate the sound of an orchestral section [34] or zoom in on a specific musician to see how a difficult instrumental part is performed.

1.2. MUSIC INFORMATION RETRIEVAL BEYOND AUDIO

When analyzing the trends in the development of online music platforms, it becomes clear that for this development, we need to look broader than the possibilities to process and analyze music audio only. While audio remains the main way to experience music, it is by far not sufficient if we wish to learn as much as possible about a music piece and obtain the desired richness of its representations and perspectives. The *Music Information Retrieval* (Music IR)[5] research community has therefore increasingly focused on a *multimodal approach*, where different modalities, like audio, visual and text, are used

[3] `http://www.concertgebouworkest.nl/en/rco-editions`

[4] `http://www.digitalconcerthall.com`

[5] Music IR is an interdisciplinary science involving musicologists, psychologists, and signal processing and machine learning experts (see `https://en.wikipedia.org/wiki/Music_information_retrieval`)

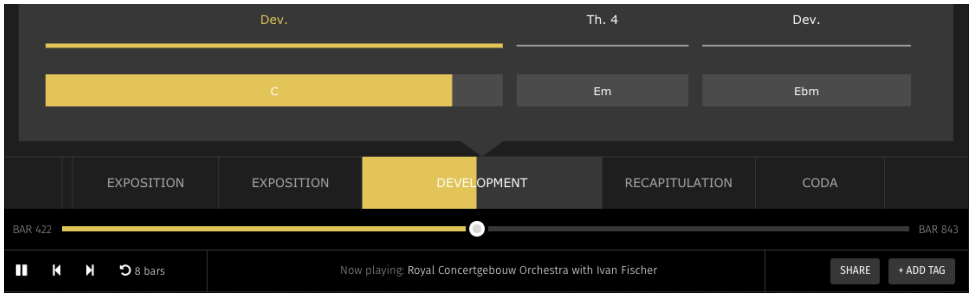


Figure 1.3: Non-linear access for classical music videos: the enriched video timeline allows users to skip to specific parts of a piece. Credit: PHENICX prototype `http://phenicx.com/`

together to maximally benefit from the available information resources.

Adding modalities, like social tags and visual data (e.g., video recordings), to the research in the Music IR field is important for two reasons. The first one is that critical information for developing the desired feature of an online music platform is simply not available in the music audio data stream. For example, the most effective way to show to a user how to play a particular music sequence on an instrument is to do this via a video. Similarly, the tasks of inferring the interest of the user in what to listen to and how to interact with a platform can be done much better by relying on social tags, online profiles, social relationships and listening behavior. In other words, an enriched experience can be obtained by extracting and exploiting information from multiple, preferably complementary modalities. The second reason also has to do with complementary nature of different modalities. Even if one modality contains information that can help us learn more about a music piece, it is often so that adding information from another modality can improve the learning performance. For example, combining social tags and audio features outperforms audio-only methods for music genre classifications [6]. Similarly, audiovisual drum transcription performs better than an unimodal (audio-only) approach [65].

1.3. THESIS FOCUS: VISION-BASED MUSIC IR

In this thesis, we pursue a multimodal approach to Music IR and focus on a combination between the audio and visual modality. Specifically, we explore how an analysis of the visual channel accompanying a music recording—i.e., a music video—can help in learning more about the music piece being performed and about the specific performance context.

Compared to the audio-only Music IR approach, the audiovisual approach is still relatively unexplored, even though the need for it has already been identified [59, 30, 95]. The existing work in this direction has mostly focused on learning the relationships between body movements and musical cues, and on using these relationships for solving traditional Music IR tasks, such as sound source separation, performance-to-score synchronization, and automatic transcription. As an example of the former, Nymoen et al. [74] carried out a user study to analyze the differences between pitched and non-

pitched sounds, and the gestures that these sounds elicit in the participants. Caramiaux et al. [20] presented a segmentation model for clarinetists' movements. Regarding the traditional Music IR tasks, a few works addressed the music transcription task in a multimodal fashion focusing on percussion (e.g., drum [65] and vibraphone [89]) and guitar [75]. Essid & Richard [30] reviewed multimodal and cross-modal signal processing and machine learning methods to propose a general modality fusion and analysis framework for five representative Music IR tasks. In particular for sound source separation, the authors recommended that partially-informed approaches should be pursued relying on additional modalities. That this is still in its infancy can be observed from the fact that a visually informed multi-pitch estimation algorithm for string ensembles has been proposed for the first time only recently [27].

The aforementioned works present experiments mostly conducted on 3D motion capture (mocap) data or "ideal" video recordings. Mocap data is highly *structured*, since it is based on tracking 3D-world coordinates of relevant points, like body joints. This type of data allows, for instance, to accurately identify sound-gesture relationships, which are important to guide us in what to look for when analyzing videos of performing musicians. When video recordings are used, they are typically acquired in a *constrained* way, in order to again make the extraction of structured annotations easier. This is done by using static cameras, favorable viewpoints and homogeneous backgrounds to avoid clutter.

Using structured and constrained visual data is unarguably a critical first step in exploring the possibilities for audiovisual Music IR, since it allows one to focus on fundamental questions on how to link and combine audio and visual signals for effectively and efficiently inferring higher-level information from data. However, a second step needs to be taken as well, expanding and adjusting the obtained insights from the first step towards methods and algorithms that are capable of handling real-world, that is, *unstructured* and *unconstrained* audiovisual data. The need for such solutions becomes visible if we realize that, for instance, a mocap system is too obtrusive for live concerts, and therefore traditional (or depth) cameras should be used instead. These cameras make, however, the extraction of the links between body movements and musical cues much more difficult than in the mocap case as the information about body parts is not available directly but needs to be extracted from the raw visual stream. Recorded videos are also typically not constrained due to limited options to position the cameras in a live concert setting, resulting in suboptimal visibility of the musicians and their instruments, e.g., due to pose variations and occlusions.

1.4. THESIS GOAL AND SCOPE

Developing methods and algorithms that can handle real-world data is critical for being able to optimally benefit from the available information resources in the broadest possible application scope. The goal of this thesis is therefore to investigate how vision-based Music IR techniques can be developed to be able to run on realistic, thus unstructured and unconstrained audiovisual music recordings and extract information that is of relevance for the emerging online music platforms. By pursuing this goal, we expand the state-of-the-art in the field in two ways:

- **Approach 1**: We consider videos of individual musicians like in most of the previous works discussed above, but look beyond the current possibilities to link specific (e.g., instrument- or performer-related) visual and music cues. We aim for a more generic audiovisual analysis of a music video, possibly independent of the played instrument, and make as few assumptions as possible on the exact recording conditions. We specifically look at the type and depth of the performance-relevant information that is extractable in this way. The rationale behind this is that the movements of a musician while playing relate to what they play. If extracted in a generic fashion, this link can help in automatically structuring and indexing a general (and thus also unconstrained) music video.

- **Approach 2**: We expand the audiovisual analysis from video of single performers to videos of large ensembles. Specifically, we focus here on the recordings of classical music concerts. We find this choice representative of the typical real-world data challenges in vision-based Music IR. For instance, a video recording of a symphonic orchestra concert features a large number of instruments, played by a large number of artists in a crowded (thus cluttered) setting and with non-ideal image resolution, e.g., due to the recordings made by a remote camera positioned in front of the concert stage. Furthermore, classical music pieces are typically lengthy, with sequences played by different instrument combinations and showing large variation in musical characteristics, such as tempo and intensity. This poses numerous challenges of how to link these variations in the audio channel to the information in the visual channel and extract performance-level information relevant to share with users of an online music platform.

1.5. THESIS OVERVIEW

For the two approaches described above, we conduct different investigations, which lead to various methods, algorithms and results reported in chapters 2-6 of this thesis, specifically in chapters 2 and 6 for Approach 1 and chapters 3-5 for Approach 2. In the following, we briefly describe the contributions of each of these chapters and their mutual relations.

With the work reported in **Chapter 2** we start our search for generic visual cues that can help us represent musicians' movements over time in a low-dimensional and interpretable space and link them to the characteristics of the musical performance. Specifically, we propose an unsupervised method based on the extraction of a lightweight feature named "Motion Orientation Histogram" and on the computation of novelty peaks in the value development of this feature over time. By following this approach, we find explainable correlations between novelty peaks and the note onset density computed on the isolated audio track corresponding to the recorded musician. Also, we observe that expert-annotated structural boundaries are often aligned with visual novelty peaks and these findings generalize across different musicians playing three different instruments.

In **Chapter 3** and **Chapter 4**, we expand the investigation from **Chapter 2** from the recordings of individual musicians to the ones of larger ensembles, and we also focus on concrete objectives that are potentially of relevance of an online music platform, as discussed earlier in this chapter. Specifically, and in view of the example illustrated in

Figure 1.2a, we consider the problem of synchronizing the performance of a music piece with the score of that piece. Searching for an alternative to the current practice of obtaining such a synchronization, namely using audio rendered from MIDI files that represent the scores, we want to investigate the use of the visual channel for this purpose. We choose to analyze the visual channel for generic cues indicating whether a member of an ensemble is playing or not playing her instrument. The rationale behind this choice is that by extracting the playing/non-playing (P/NP) label sequences from the visual channel and aggregating such labels over time and across musicians, a matrix that represents the instrumentation over time is easily obtained. The same information can be derived via symbolic score reduction, therefore making temporal alignment of the performance and the score straightforward. Looking for generic cues like P/NP labels is important, since this type of labels applies to any played instrument, independent of the position of a musician in the ensemble and even in presence of partial visibility of person and/or instrument, and independent of the properties of a particular video recording (e.g., position of the cameras).

Before we proceed with the visual analysis with the purpose of extracting the P/NP label sequences for all musicians, we first perform a preliminary experimental analysis reported in **Chapter 3** to investigate whether P/NP information aggregated across musicians, if available, is sufficient at all as a cue for synchronization with the score. Our experiments show that P/NP information is sufficient to perform a coarse temporal synchronization between a video and a symbolic score (temporal tolerance of about 2 seconds), even in presence of noise or missing data. The results in **Chapter 3** provide motivation to investigate how to design a P/NP labels extraction system from the visual channel for large-ensemble videos. We pursue this challenge through the work presented in **Chapter 4**. This chapter first describes the characteristics of symphonic orchestral recordings, and then presents a semi-automatic system combining face and human-object interaction clustering techniques for P/NP sequence extraction. We pursue a semi-automatic annotation strategy, since data-driven approaches require datasets that are not available yet for the considered problem. For this experiment, we use real-world recordings of two symphonic music concerts, spanning multiple moving camera and single static camera recordings. Our goal is to identify sub-modules that are critical to solve the P/NP labels extraction problem with satisfying performance. The results show that existing face clustering methods can be improved to reduce musician identity labeling mistakes and the amount of manual work required to label the produced clusters. Regarding the P/NP labeling step, we observe that even if severe occlusion shadows the sound producing movements, other visual cues like facial expression can be exploited for P/NP detection. This indicates that, if one were to make a dataset to train a classifier, visual features beyond body pose and upper body movements should be included.

The next investigation, reported in **Chapter 5**, is inspired by the deficiencies of visual channel analysis we experienced in **Chapter 4**. There, we aim to improve the performance of face clustering for a specific category of scenes that we encounter in our video recordings of large ensembles and that we generally refer to as *quasi-static scenes*. In these scenes, we can assume that the people's positions are (quasi-)stationary. Based on this assumption, we present a method that automatically builds a map of the scene from an unconstrained video in the form of a graph. Thereby, spatial relationships be-

tween the available camera views are used to perform sub-graph matching across face graphs derived from each overlapping view pair. The results show that the spatial information is well exploited in this way and that our method becomes even more effective with crowded shots—which are typically challenging for traditional face clustering algorithms due to the lack of visual detail. We also find it important to note that our method generalizes to other quasi-static scene settings beyond classical music concerts, like for example those found in talk shows or sitcoms, even if special editing effects, like split screens, are present.

In **Chapter 6**, we return back to the type of discussion we initiated in **Chapter 2**, but now we approach the investigation of generic links between visual and audio cues for musical performance characterization from the supervised learning perspective. Specifically, the work presented in this chapter addresses the challenge of vision-based detection and analysis of finger movements on, for instance, woodwind and brass instruments and linking these movements to acoustic events. For this purpose, we present a fully automatic method based on deep learning for note onsets detection from clarinetist videos. In addition, we publish a new dataset of 4.5 hours of video with about 36,000 annotated onsets. The results reported in **Chapter 6** are based on preliminary investigations and show that our proposed classifier currently performs only on par with a ground-truth informed random baseline. We also observe that a gap between vision-based and audio-only note onsets detection (performed on isolated audio tracks) is rather large. However, the problem and the proposed algorithm are both novel and we considered it worth reporting to the scientific community. We are confident that further research on this topic will lead to significantly better results.

1.6. THESIS IMPACT

The main impact of this thesis lies in the experimentally validated insight that the vast potential of the information contained in the visual channel of a music video can be exploited much further than in the existing work in order to build tools allowing us to efficiently, effectively and intuitively interact with music. As shown by the example in Figure 1.4, our findings on P/NP detection already enable the development of new functionalities for online music platforms. Furthermore, our solution to the video face clustering problem (**Chapter 5**) can be exploited together with the sound source separation output to zoom in, both via the visual and auditory channel, onto a specific musician—just by tapping on musicians' faces while watching a video. Face labels can also be exploited to let users switch viewpoints according to the specific instruments/musicians they may want to observe. A proof of concept of such functionalities is available at `http://youtu.be/60j70tvqo9c`. The findings reported in the thesis pave the way for more research in the still emerging research area of vision-based Music IR, and shed light on methodological and algorithmic design choices suitable for handling real-world data. Last but not least, we hope that the dataset we created will further facilitate these efforts.

1.7. HOW TO READ THE THESIS

For the technical part of this thesis, original publications have been adopted as chapters 2, 3, 4, 5 and 6. The references to the corresponding publications are given in the

1

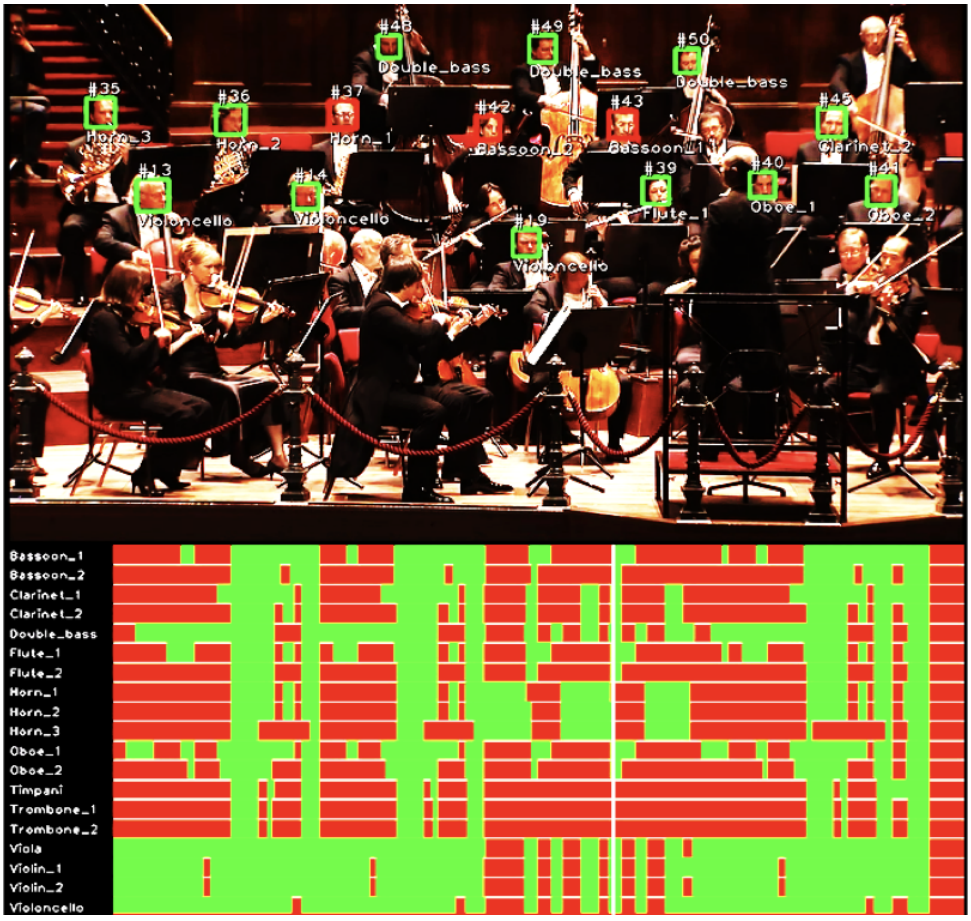


Figure 1.4: Illustration of our proof of concept which combines the output of visual and audio analysis. The full video is available at `http://youtu.be/6Oj70tvqo9c`.

footnote at the beginning of each chapter. Since some of the papers have appeared in conferences, some in workshops and some in scientific journals, the length and depth of the chapters also varies accordingly. Since we retained the original form of the publications, there may be variation in the notation and terminology across the chapters. Also, if chapters address the same general topic, there may be similarity in the motivation, argumentation and some of the material (e.g., sections on related work) they cover.

1.8. LIST OF PUBLICATIONS RELATED TO THE THESIS

The following papers have been published by the author of the thesis while pursuing the PhD degree in the Multimedia Computing Group at the Delft University of Technology. Those publications directly serving as chapters of the thesis are indicated accordingly.

Journals:

- Alessio Bazzica, Cynthia C. S. Liem, Alan Hanjalic. On detecting the playing/non-playing activity of musicians in symphonic music videos. *Computer Vision and Image Understanding* (2016) - **Chapter 4** [13].
- Alessio Bazzica, Cynthia C. S. Liem, Alan Hanjalic. Exploiting scene maps and spatial relationships in quasi-static scenes for video face clustering. *Image and Vision Computing* (2017) - **Chapter 5** [14].

Conferences:

- Alessio Bazzica, Cynthia C. S. Liem, Alan Hanjalic. Exploiting instrument-wise playing/non-playing labels for score synchronization of symphonic music. *International Society for Music Information Retrieval Conference* (ISMIR) (2014) - **Chapter 3** [12].

Workshops:

- Cynthia C. S. Liem, Alessio Bazzica, Alan Hanjalic. Looking beyond sound: unsupervised analysis of musician videos. *International Workshop on Image and Audio Analysis for Multimedia Interactive Services* (2013) - **Chapter 2** [58].
- Alessio Bazzica, Jan C. van Gemert, Cynthia C. S. Liem, Alan Hanjalic. Vision-based Detection of Acoustic Timed Events: a Case Study on Clarinet Note Onsets *International Workshop on Deep Learning for Music, in conjunction with the International Joint Conference on Neural Networks* (2017) - **Chapter 6** [15].

2

LOOKING BEYOND SOUND: UNSUPERVISED ANALYSIS OF MUSICIAN VIDEOS

As a first step, we investigate how to represent musicians' movements, aiming to match motion features to the characteristics of the analyzed musical performance. To this end, we introduce a feature, named Motion Orientation Histograms, that encodes global movements as a sequence of vectors in a low-dimensional and interpretable space. We then perform unsupervised visual analysis on jam session audiovisual recordings, which leads to explainable correlations, both when motion novelty is compared to structural annotations ans also when compared to isolated audio track note onset density.

This chapter was published as: Cynthia C. S. Liem, Alessio Bazzica, Alan Hanjalic. Looking beyond sound: unsupervised analysis of musician videos. *International Workshop on Image and Audio Analysis for Multimedia Interactive Services*, 2013. Alessio Bazzica gave a major contribution to the scientific and technical part of the paper.

When making music, a musician will move. In general, muscular action is needed in order to have an instrument producing the desired musical sounds. On top of this, the musician's experience of the played piece may trigger additional movement, or influence the movement necessary for sound production. Hence, when looking at a performing musician, there will be visual cues regarding developments over the course of the performed musical piece, which will influence an audience member's perception of the musical interpretation [96].

Traditionally, audio analysis is employed to characterize the timeline of a recorded musical piece. A common way to detect new or novel events over time, such as the occurrence of structural boundaries, is to compute frame-level features such as Mel-Frequency Cepstral Coefficients (MFCCs) or chromagrams, followed by a self-similarity analysis [37]. However, in order for these features to give convincing results, they should show sufficient variation throughout the recording. This is not the case for music in which timbre and harmonic content do not develop much throughout a piece. For instance, this frequently happens in jam sessions, when a fixed chord scheme is followed and differentiation between musical sections is based on alternating improvised solo parts.

In this chapter, we aim to take a step forward regarding this problem by considering movement of musicians over the course of their performance. We study this information based on video data, since, in practical situations, the visual channel is a straightforward modality to record in a non-obtrusive way.

Our approach is driven by the interest to find generally applicable movement descriptors, allowing for overall unsupervised timeline indexing of a performance, characterizing highlights and sectional changes over the course of the performance, and supporting or complementing information on this as obtained from audio analysis. As such, our techniques are meant to ultimately support non-linear access scenarios.

This chapter is outlined as follows: in Section 2.1, we discuss related work. We then present our visual analysis approach and its rationale in Section 2.2. After this, we present the data used for our current case study in Section 2.3, after which results are discussed in Section 2.4. Finally, general conclusions and an outlook to future work are presented in Section 2.5.

2.1. RELATED WORK

Many existing studies on characteristics of music-related movement have involved gestural analysis. For example, Wanderley [101] and Caramiaux et al. [20] investigate the consistency and parsing of ancillary gestures (i.e., gestures related to the instrument which are not caused because of sound production) by instrumental musicians. Studies on music-induced motion in listeners rather than performers have been conducted by Nymoen et al. [74]. Typically, for gesture-oriented work 3D motion capture data is used, and due to the generated large amount of sensor data, detailed analysis can often just be feasibly performed on short excerpts. As pointed out by Godøy and Jensenius [42], in this direction of work, video processing methods for extracting features of music-related body movement still are generally lacking.

An exception is the work started by Gillet and Richard in [40], and expanded in McGuinness et al. [65], where the aim was to transcribe drum sequences from video recordings of

performers. Given this goal, the focus was on classification of highly instrument-specific events. As mentioned before, in our current work, we aim at taking a more general perspective on visual information conveyed by performing musicians. Therefore, we will not focus on classification of specific events, nor will we explicitly strive to only analyze ancillary or expressive movements.

2.2. VISUAL ANALYSIS

Given a set of video recordings of performing musicians, we aim to extract a series of visual novelty points over time, relating to the temporal development of the musical performance. We do not wish to depend on a specific set of instruments, and wish to be as flexible as possible regarding characteristic individual motion patterns of musicians. Therefore, we focus on analyzing motion patterns rather than the shape of objects, without restricting to a pre-defined vocabulary of motion patterns, nor attempting to establish such an explicit vocabulary. Regarding the video setup, we require that the video is recorded with a stationary camera, but do not put any specific restriction on positioning of players, instruments and cameras otherwise. The only further requirements are that moving objects are not completely occluded, and that the motion is not uniquely occurring along a spectator's line of sight.

We aim to efficiently detect any moving object with relation to the music (thus, objects associated to the player or his instrument). Many such objects may move at the same time, and each object can move in a specific direction. For instance, a drummer can hit a snare drum while triggering the hi-hat with the pedal. In order to encode a variable number of moving objects moving towards any direction, we choose to detect region of interests (ROIs) adopting the approach of Bradski et al. [17]. First, recent motion is encoded as a *motion-history image* (MHI) accumulating thresholded frame differences (e.g., see Figure 2.1). Then, each MHI is segmented according to an iterative algorithm called *downward stepping floodfill*: the most recent motion is progressively connected to the older through a sequence of gradient descent steps. When two different objects are moving, they usually lead to two different floodfill regions. Each of these is used as ROI and encoded as a silhouette mask moving towards a specific direction.

Inspired by Davis [26], we iterate over the extracted ROIs to build a histogram of motion orientations per frame (see Figure 2.2). The bins are assigned quantizing the orientation in 12 sectors, and the area is measured as the number of silhouette pixels within the ROI. To encode temporal development, we then apply a 2-seconds sliding window, summing together the histograms of frames within the window into a single 12-bin vector. The resulting vector is expanded adding a "no motion flag" which is set to 1 if all the bins are zero — i.e., no ROI has been extracted in the past 2 seconds (see Figure 2.3).

Based on the summed histogram features, we wish to detect significant visual motion variations over time. For this, two properties are particularly useful, the first being "classical" novelty: the measure of temporal change as proposed by Foote [37], obtained from self-similarity matrix analysis employing a checkerboard kernel. For this, we employ the cosine distance, and use a Gaussian checkerboard kernel of 20 seconds to compare motion patterns in the near past and future[1]. The second useful property is

[1] This wide kernel is chosen to favor coarse development over short-time details.

Figure 2.1: Motion History Image of a drummer in performance.

the overall amount of motion, which simply can be computed by summing together the contributions of the 12 histogram bins at each point in time.

Finally, we compute our visual novelty curve by combining the degree of "classical" novelty with the degree of motion. For this, we normalize both measures, and simply multiply them frame-by-frame. This results in a function which peaks when there is a lot of novelty and a lot of motion.

2.3. DATA

For our current study, we use a dataset with multi-track studio recordings of live performances in swing, blues and funk styles, released by Abeßer et al. [1].[2] The dataset consists of multi-track recordings of 3 combos of 3 musicians, playing guitar, bass guitar and drums. Each of the combos is recorded during a session in which swing, blues and funk styles are performed. For each of the styles, improvised solo parts occur,[3] which are annotated in the dataset. Together with the multi-track audio recordings, video material is released with the dataset, showing each of the musicians during their performance in a single, static shot.

Regarding the recorded data, we cut out the excerpts from the sessions which actually corresponded to the featured styles, removing breaks and intermediate talking, thus retaining 70 minutes of recorded material. We then manually synchronized the video and audio streams in the dataset, ensuring that any possible temporal deviation remained under 0.5 seconds.

The multi-track audio recordings consisted of many separate audio tracks: one for the bass, two for the guitar, and six for the drums. As we do not assume that so many tracks per instrument will be available in future work beyond this case study, we mixed[4]

[2] `https://goo.gl/H7rGEY`

[3] As such, there will not be a notated score, and every performance will create a new piece that was not played before.

[4] In all cases, we mixed together the tracks by simply adding them up, and correcting the peak level to be at 0.0

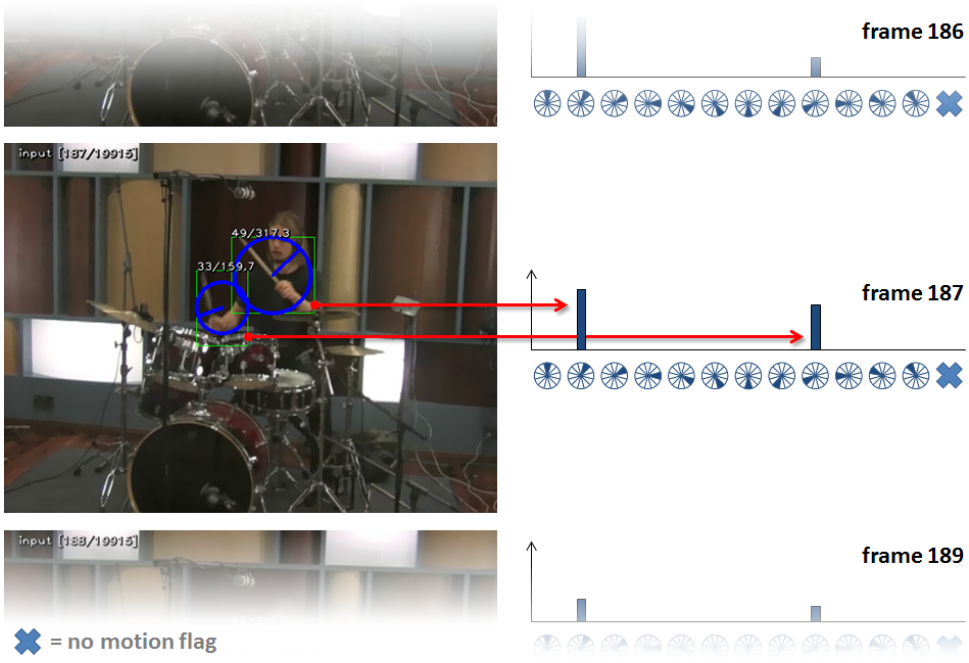


Figure 2.2: Motion Orientation Histograms.

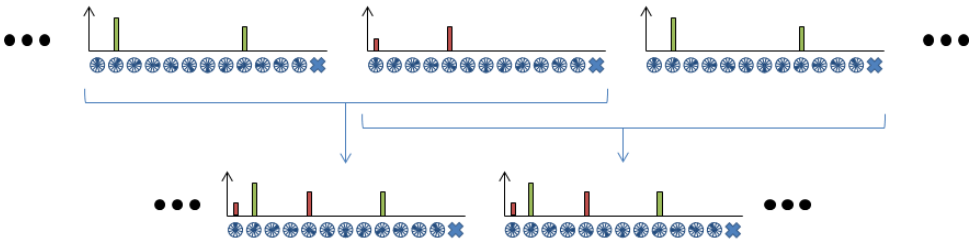


Figure 2.3: Encoding sequences of motion pattern descriptors

these together for each instrument, and as a full mix involving all instruments.

The original dataset provided structural session annotations at the full-second resolution. Respecting this resolution, we corrected annotated boundaries to have them start and end with an acoustic event. To allow deeper analysis, we made additional manual annotations, marking every 4 beats ("a bar") and the starts of repeated chord schemes or cells ("a cycle").[5].

2.4. RESULTS AND DISCUSSION

While the dataset in our current study is small (but rich), the observed behavior of our proposed analysis method with regard to this dataset is promising. Over time, the motion orientation histograms and their derived visual novelty show explainable patterns with respect to the structural annotations. Furthermore, they give indications of internal development throughout a performance, even if timbre and instrumentation will not vary much over the course of the piece.

A good illustration of this can be seen in Figure 2.4, which shows motion histograms and overlayed visual novelty graphs for the instrumentalist videos of the second combo in the dataset, playing in Funk style.[6] This particular Funk session was striking, since it was entirely based on a one-bar, continuously repeated cell in the bass guitar. Despite this constant foundation, the movement behavior of the instrumentalists is not uniform. Peaking behavior in the novelty curves intensifies when an instrumentalist has the solo role.

In order to verify to what extent our visual novelty curve reflects information already present in the audio channel, we wished to compare our visual novelty curves to a representative audio-based descriptor. As, due to our data genre, timbre- or harmony-based descriptors would not be as suitable as usual, we chose a more low-level feature. This feature was computed by running an onset detector on each mixed audio track, and then summing the energy contributions of every detected onset peak per second in the recording. We considered the resulting onset intensity feature to be a reasonable approximation of the auditory event density in the tracks.

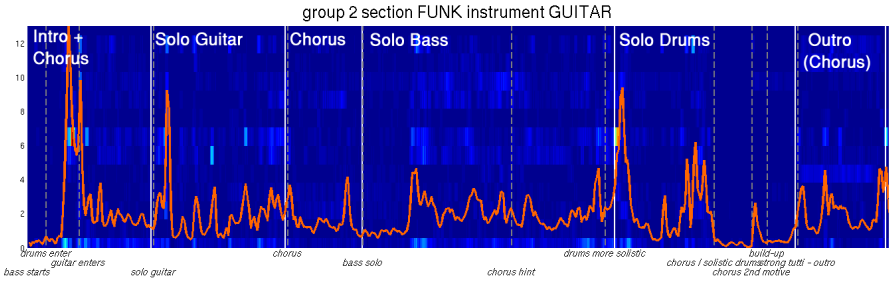
For every video recording, we computed Pearson's correlation coefficient between the visual novelty curve and two onset intensity vectors: the vector computed from the instrument-specific mixed audio track, and the vector computed from the full ensemble-mixed audio track. Results for instrument-specific tracks are shown in Table 2.1, while those for full ensemble mixes are shown in Table 2.2.

From the correlation values, we can conclude that the visual novelty information is largely complementary to onset intensity information, with the exception of the drums player. This is explainable, since the drums player cannot move a lot beyond direct interaction with the instrument. From similar reasoning, the generally poor correlation of the bass guitar player with the onset intensity information can be explained: the truly instrument-related movement on a bass guitar is more subtle than other movement made by the player, such as foot-tapping along with the music. While the latter action

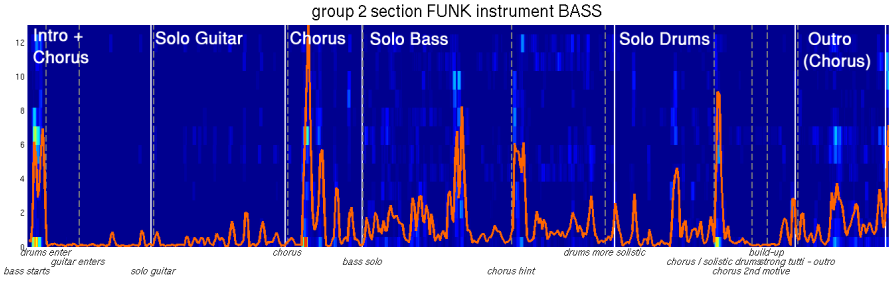
dB.

[5] We release these annotations at `http://homepage.tudelft.nl/04d13/wiamis2013.html`

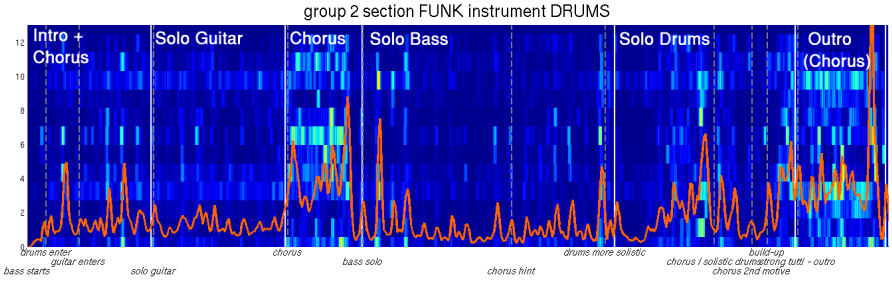
[6] Additional illustrations and examples for other recordings in the dataset are given at the website mentioned in Footnote 5.



(a) Motion histograms and visual novelty: guitar



(b) Motion histograms and visual novelty: bass



(c) Motion histograms and visual novelty: drums

Figure 2.4: Motion features (histograms with visual novelty overlayed) for group 2, Funk style session. Vertical white lines indicate structural boundaries as annotated in the original dataset. Vertical dashed grey lines indicate additional notes as described in Footnote 5.

		Blues	Funk	Swing
Group 1	Bass	0.073	-0.088	-0.245
	Guitar	0.350	0.251	-0.081
	Drums	0.280	0.382	0.475
Group 2	Bass	0.151	-0.088	-0.179
	Guitar	0.248	0.184	0.047
	Drums	0.325	0.518	0.538
Group 3	Bass	-0.0507	-0.012	-0.201
	Guitar	0.095	0.210	-0.052
	Drums	0.204	0.196	0.612

Table 2.1: Pearson's correlation coefficient for visual novelty with onset intensity of instrument audio mix.

		Blues	Funk	Swing
Group 1	Bass	0.113	-0.103	-0.138
	Guitar	0.367	0.130	-0.022
	Drums	0.263	0.299	0.232
Group 2	Bass	0.170	0.034	0.098
	Guitar	0.315	0.140	0.015
	Drums	0.376	0.508	0.492
Group 3	Bass	-0.068	0.111	0.117
	Guitar	0.195	0.170	0.048
	Drums	0.213	0.250	0.562

Table 2.2: Pearson's correlation coefficient for visual novelty with onset intensity of full audio mix.

is not causing sound production, it is synchronized to the music and a sign of entrainment, and as such still related to the music jointly made by the ensemble. While it needs further investigation, we conjecture that this can be an explanation that correlation coefficients with the onset intensities of full ensemble mixes are generally higher than those computed for the individual instrumental mixes.

We noted over multiple recordings that novelty peak maxima indicate major body movement, such as a posture change. Once again, these changes are often in sync with the music, and are related to events in the performance (e.g., picking up a plectrum for an intensified solo part). However, they cannot be fully discerned from truly incidental movement yet, so this needs further consideration in follow-up work.

2.5. CONCLUSIONS AND FUTURE WORK

We presented unsupervised visual analysis techniques for videos of performing musicians. Our initial observations show that explainable results are yielded, which can characterize events and entrainment throughout a performance for different players, and in certain cases complement audio channel information.

One of our priorities in future work will be to establish more quantitative strategies to evaluate feature performance. A clear-cut ground truth does not exist for this type of performance characterization: we do not aim to detect exact structural boundaries, but to clarify development and variation, and find novel events within these boundaries. It is challenging but interesting to devise appropriate measures for this.

Having seen promising initial results on jam session data, we now plan to expand our analysis to performances in other genres, with larger numbers of musicians. An attractive property of using video data is that it takes physical space into account: even in a video featuring multiple musicians at the same time, it is straightforward to study sub-groups or individuals, by just defining an appropriate subset of pixels. This is less trivial for audio data: if contributions of sub-groups within an ensemble are to be studied there, these need to be have been recorded in separate tracks, or source separation techniques will have to be applied. We therefore hope to find more opportunities to use visually-based analysis to support and enhance musical performance analysis.

3

EXPLOITING INSTRUMENT-WISE PLAYING/NON-PLAYING LABELS FOR SCORE SYNCHRONIZATION OF SYMPHONIC MUSIC

The results in the previous chapter encourage us to move from pieces performed by a few players to larger ensembles. We therefore consider classical music videos, in which there are dozens of different instruments and even more players. Differently from the jam session recordings, typical symphonic orchestra videos are recorded from multiple view points and with moving cameras. Preliminary experiments using Motion Orientation Histograms on these videos taught us that the feature is not suitable in this case, since musicians appear at different scales, from different viewpoints and camera movements make the feature too noisy. Hence, in this chapter, we look for a different, yet musically relevant, information to extract. Namely, we investigate whether sequences of "playing" (P) and "non-playing" (NP) labels are of any value in Music IR. We do this by conducting a performance-to-score synchronization experiment on synthetic data using a corpus of MIDI files and simulating different types of error that a vision-based P/NP detector can generate. The results show that a coarse synchronization is possible using the visual channel of the video only, indicating that addressing the vision-based labeling problem is worth pursuing.

This chapter was published as: Alessio Bazzica, Cynthia C. S. Liem, Alan Hanjalic. Exploiting instrument-wise playing/non-playing labels for score synchronization of symphonic music. *International Society for Music Information Retrieval Conference*, 2014

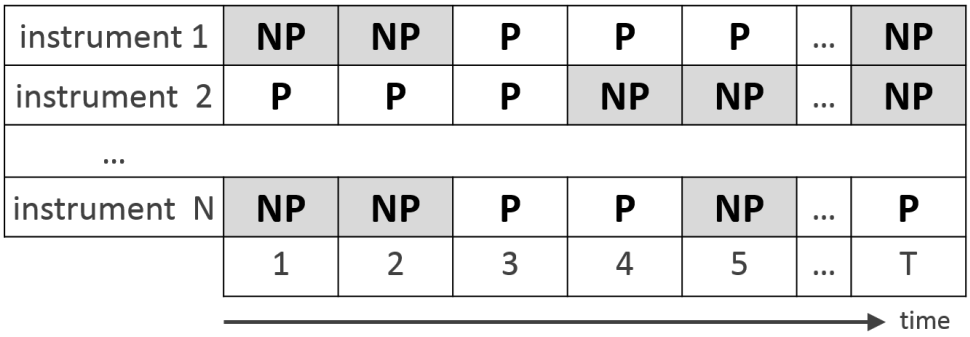


Figure 3.1: An illustration of the representation of a symphonic music piece using the matrix of playing/non-playing labels.

Synchronizing an audio recording to a symbolic representation of the performed musical score is beneficial to many tasks and applications in the domains of music analysis, indexing and retrieval, like audio source separation [32, 45], automatic accompaniment [25], sheet music-audio identification [39] and music transcription [93]. As stated in [38], "sheet music and audio recordings represent and describe music on different semantic levels" thus making them complementary for the functionalities they serve.

The need for effective and efficient solutions for audio-score synchronization is especially present for genres like symphonic classical music, for which the task remains challenging due to the typically long duration of the pieces and a high number of instruments involved [24]. The existing solutions usually turn this synchronization problem into an audio-to-audio alignment one [70], where the score is rendered in audio form using its MIDI representation.

In this chapter, we investigate whether sequences of playing (P) and non-playing (NP) labels, extracted per instrument continuously over time, can alternatively be used to synchronize a recording of a music performance to a MIDI file. At a given time stamp, the P (NP) label is assigned to an instrument if it is (not) being played. If such labels are available, a representation of the music piece as illustrated in Figure 3.1 can be obtained: a matrix encoding the P/NP "state" for different instruments occurring in the piece at subsequent time stamps. Investigating the potential of this representation for synchronization purposes, we will address the following research questions:

- **RQ1**: How robust is P/NP-based synchronization in case of erroneous or missing labels?
- **RQ2**: How does synchronizing P/NP labels behave at different time resolutions?

We are particularly interested in this representation, as P/NP information for orchestra musicians will also be present in the signal information of a recording. While such information will be hard to obtain from the audio channel, it can be obtained from the visual channel. Thus, in case an audiovisual performance is available, using P/NP information opens up possibilities for video-to-score synchronization as a means to solve a score-to-performance synchronization problem.

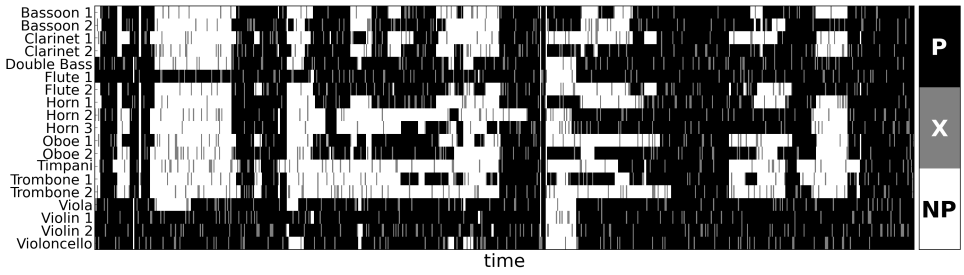


Figure 3.2: Example of a M_{PNP} matrix with missing labels.

The rest of the chapter is structured as follows. In Section 3.1, we formulate the performance-to-score synchronization problem in terms of features based on P/NP labels. Then, we explain how the P/NP matrix is constructed to represent the score (Section 3.2) and we elaborate on the possibilities for extracting the P/NP matrix to represent the analyzed performance (Section 3.3). In Section 3.4 we propose an efficient method for solving the synchronization problem. The experimental setup is described in Section 3.5 and in Section 3.6 we report the results of our experimental assessment of the proposed synchronization methodology and provide answers to our research questions. The discussion in Section 3.7 concludes the chapter.

3.1. PROBLEM DEFINITION

Given an audiovisual recording of a performance and a symbolic representation of the performed scores, we address the problem of synchronizing these two resources by exploiting information about the instruments which are active over time.

Let $L = \{-1, 0, 1\}$ be a set encoding the three labels non-playing (NP), missing (X) and playing (P). Let $M_{\mathrm{PNP}} = \{m_{ij}\}$ be a matrix of $N_{\mathrm{I}} \times N_{\mathrm{T}}$ elements where N_{I} is the number of instruments and N_{T} is the number of time points at which the P/NP state is observed. The value of $m_{ij} \in L$ represents the state of the i-th instrument observed at the j-th time point ($1 \leq i \leq N_{\mathrm{I}}$ and $1 \leq j \leq N_{\mathrm{T}}$). An example of M_{PNP} is given in Figure 3.2.

We now assume that the matrices $M^{\mathrm{AV}}_{\mathrm{PNP}}$ and $M^{\mathrm{S}}_{\mathrm{PNP}}$ are given and represent the P/NP information respectively extracted by the audiovisual recording and the sheet music. The two matrices have the same number of rows and each row is associated to each instrumental part. The number of columns, i.e., observations over time, is in general different. The synchronization problem can be then formulated as the problem of finding a time map $f_{sync} : \{1 \dots N^{\mathrm{AV}}_{\mathrm{T}}\} \rightarrow \{1 \dots N^{\mathrm{S}}_{\mathrm{T}}\}$ linking the observation time points of the two resources.

3.2. SCORE P/NP REPRESENTATION

For a given piece, we generate one P/NP matrix $M^{\mathrm{S}}_{\mathrm{PNP}}$ for the score relying on the corresponding MIDI file as the information source.

We start generating the representation of the score by parsing the data of each available track in the given MIDI file. Typically, one track per instrument is added and is used

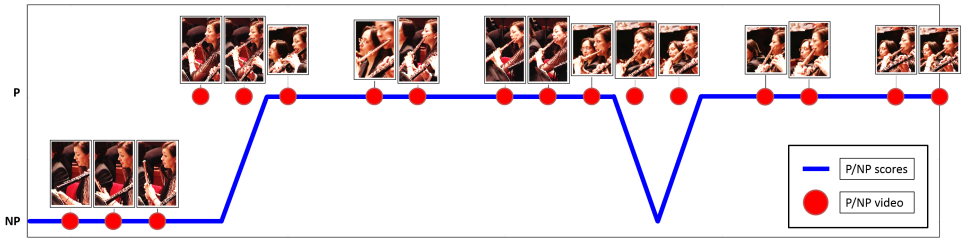


Figure 3.3: Example of P/NP labels extracted from the visual channel (red dots) and compared to labels extracted by the score (blue line).

as a symbolic representation of the instrumental part's score. More precisely, when there is more than one track for the same instrument (e.g., Violin 1, Violin 2 - which are two different instrumental parts), we keep both tracks as separate. In the second step, we use a sliding window that moves along the MIDI file and derive a P/NP label per track and window position. A track receives a P label if there is at least one note played within the window. We work with the window in order to comply with the fact that a played note has a beginning and end and therefore lasts for an interval of time. In this sense, a played note is registered when there is an overlap between the sliding window and the play interval of that note.

The length of the window is selected such that short rests within a musical phrase do not lead to misleading P-NP-P switches. We namely consider a musician in the "play" mode if she is within the "active" sequence of the piece with respect to her instrumental part's score, independently whether at some time stamps no notes are played. In our experiments, we use a window length of 4 seconds which has been determined by empirical evaluation, and a step-size of 1 second. This process generates one label per track every second.

In order to generalize the parameter setting for window length and offset, we also related them to the internal MIDI file time unit. For this purpose, we set a reference value for the tempo. Once the value is assigned, the sliding window parameters are converted from seconds to beats. The easiest choice is adopting a fixed value of tempo for every performance. Alternatively, when an audiovisual recording is available, the reference tempo can be estimated as the number of beats in the MIDI file divided by the length of the recording expressed in minutes. A detailed investigation of different choices of the tempo is reported in [39].

3.3. PERFORMANCE P/NP REPRESENTATION

While an automated method could be thought of to extract the P/NP matrix $M_{\text{PNP}}^{\text{AV}}$ from a given audiovisual recording, developing such a method is beyond the scope of this chapter. Instead, our core focus is assessing the potential of such a matrix for synchronization purposes, taking into account the fact that labels obtained from real-world data can be noisy or even missing. We therefore deploy two strategies which mimic the automated extraction of the $M_{\text{PNP}}^{\text{AV}}$ matrices. We generate them: (i) artificially, by producing (noisy) variations of the P/NP matrices derived from MIDI files (Section 3.3.1), and (ii)

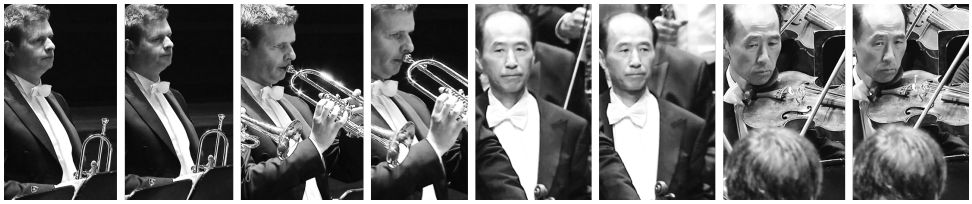


Figure 3.4: Examples of body poses indicating playing/non-playing state of a musician.

more realistically, by deriving the labels directly from the visual channel of a recording in a semi-automatic way (Section 3.3.2).

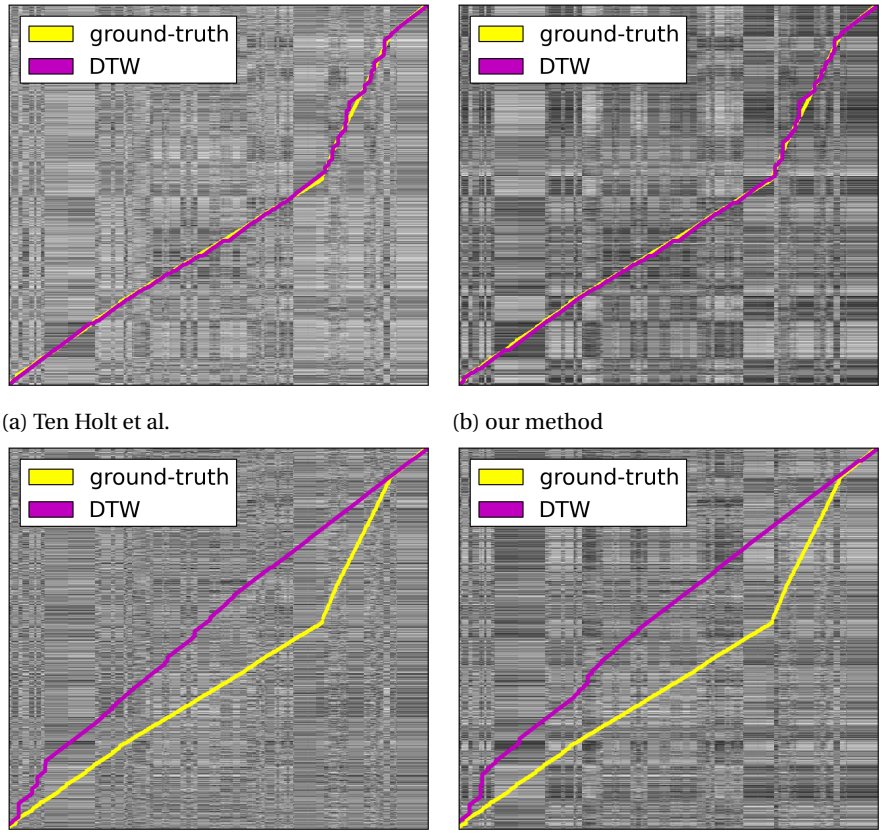
3.3.1. GENERATING SYNTHETIC P/NP MATRICES

The first strategy produces synthetic P/NP matrices by analyzing MIDI files as follows. Similarly to the process of generating a P/NP matrix for the score, we apply a sliding window to the MIDI file and extract labels per instrumental track at each window position. This time, however, time is randomly warped, i.e., the sliding window moves over time with non-constant velocity. More specifically, we generate random time-warping functions by randomly changing slope every 3 minutes and by adding a certain amount of random noise in order to avoid perfect piecewise linear functions. In a real audiovisual recording analysis pipeline, we expect that erroneous and missing P/NP labels will occur. Missing labels may occur if musicians cannot be detected, e.g., because of occlusion or leaving the camera's angle of view in case of camera movement. In order to simulate such sources of noise, we modify the generated P/NP tracks by randomly flipping and/or deleting pre-determined amounts of labels at random positions of the P/NP matrices.

3.3.2. OBTAINING P/NP MATRICES FROM A VIDEO RECORDING

The second strategy more closely mimics the actual video analysis process and involves a simple, but effective method that we introduce for this purpose. In this method, we build on the fact that video recordings of a symphonic music piece are typically characterized by regular close-up shots of different musicians. From the key frames representing these shots, as illustrated by the examples in Figure 3.4, it can be inferred whether they are using their instrument at that time stamp or not, for instance by investigating their body pose [106].

In the first step, a key frame is extracted every second in order to produce one label per second, as in the case of the scores. Faces are detected via off-the-shelf face detectors and upper-body images are extracted by extending the bounding box's areas of face detector outputs. We cluster the obtained images using low-level global features encoding color, shape and texture information. Clustering is done using k-means with the goal to isolate images of different musicians. In order to obtain high precision, we choose a large value for k. As a result, we obtain clusters mostly containing images of the same musician, but also multiple clusters for the same musician. Noisy clusters (those not dominated by a single musician) are discarded, while the remaining are labeled by linking them to the correspondent track of the MIDI file (according to the musician's in-



(a) Ten Holt et al. (b) our method

(c) Ten Holt et al. (d) our method

Figure 3.5: Comparing our distance matrix definition to Ten Holt et al. [90]. The first row corresponds to a noisy M_{PNP} matrix, the second to a very noisy one. By visual inspection, we observe comparable alignment performances. However, the computation of our distance matrix is much faster.

strument and position in the orchestra, i.e., the instrumental part). In order to label the upper-body images as P/NP, we generate sub-clusters using the same features as those extracted in the previous (clustering) step. Using once again k-means, but now with k equal to 3 (one cluster meant for P labels, one for NP and one extra label for possible outliers), we build sub-clusters which we label as either playing (P), non-playing (NP) or undefined (X). Once the labels for every musician are obtained, they are aggregated by instrumental part (e.g., the labels from all the Violin 2 players are combined by majority voting). An example of a P/NP subsequence extracted by visual analysis is given in Figure 3.3.

3.4. SYNCHRONIZATION METHODOLOGY

In this section, we describe the synchronization strategy used in our experiments. The general idea is to compare configurations of P/NP labels for every pair of performance-score time points and produce a distance matrix. The latter can then serve as input into a synchronization algorithm, for which we adopt the well-known dynamic time warping (DTW) principle. This implies we will not be able to handle undefined amounts of repeats of parts of the score. However, this is a general issue for DTW also holding for existing synchronization approaches, which we consider out of the scope of this chapter.

In order to find the time map between performance and score, we need to solve the problem of finding time links between the given $M^{\mathrm{AV}}_{\mathrm{PNP}}$ and $M^{\mathrm{S}}_{\mathrm{PNP}}$ matrices. To this end, we use a state-of-the-art DTW algorithm [90].

3.4.1. COMPUTING THE DISTANCE MATRIX

Ten Holt et al. [90] compute the distance matrix through the following steps: (i) both dimensions of the matrices are normalized to have zero mean and unit variance, (ii) optionally a Gaussian filter is applied, and (iii) pairs of vectors are compared using the city block distance. In our case, we take advantage of the fact that our matrices contain values belonging to the finite set of 3 different integers, namely the set L introduced in Section 3.1. This enables us to propose an alternative, just as effective, but more efficient method to compute the distance matrix.

Let $\boldsymbol{m}^{\mathrm{AV}}_j$ and $\boldsymbol{m}^{\mathrm{S}}_k$ be two column vectors respectively belonging to M^{AV}_{PNP} and M^{S}_{PNP}. To measure how (dis-)similar those two vectors are, we define a *correlation score* s_{jk} as follows:

$$s_{jk} = \mathrm{corr}(\boldsymbol{m}^{\mathrm{AV}}_j, \boldsymbol{m}^{\mathrm{S}}_k) = \sum_{i=1}^{N_{\mathrm{I}}} m^{\mathrm{AV}}_{ij} \cdot m^{\mathrm{S}}_{ik}$$

From such definition, it follows that a pair of observed matching labels add a positive unitary contribution. If the observed labels do not match, the added contribution is unitary and negative. Finally, if one or both labels are not observed (i.e., at least one of them is X), the contribution is 0. Hence, it also holds $-N_{\mathrm{I}} \leq s_{jk} \leq +N_{\mathrm{I}}$. The maximum is reached only if the two vectors are equal. Correlation scores can be efficiently computed as dot-product of the given P/NP matrices, namely as $(M^{\mathrm{AV}}_{\mathrm{PNP}})^{\top} M^{\mathrm{S}}_{\mathrm{PNP}}$.

The distance matrix $D = \{d_{jk}\}$, whose values are zero when the compared vectors are equal, can now be computed as $d_{jk} = N_{\mathrm{I}} - s_{jk}$. As a result, D will have $N^{\mathrm{AV}}_{\mathrm{T}}$ rows and $N^{\mathrm{S}}_{\mathrm{T}}$ columns. When the correlation is the highest, namely equal to N_{I}, the distance will be zero.

Our approach has two properties that make the computation of D fast: D is computed via the dot product and it contains integer values only (as opposed to standard methods based on real-valued distances). As shown in Figure 3.5, both the distance matrix proposed in [90] and using our definition produce comparable results. Since our method allows significantly faster computation (up to 40 times faster), we adopt it in our experiments.

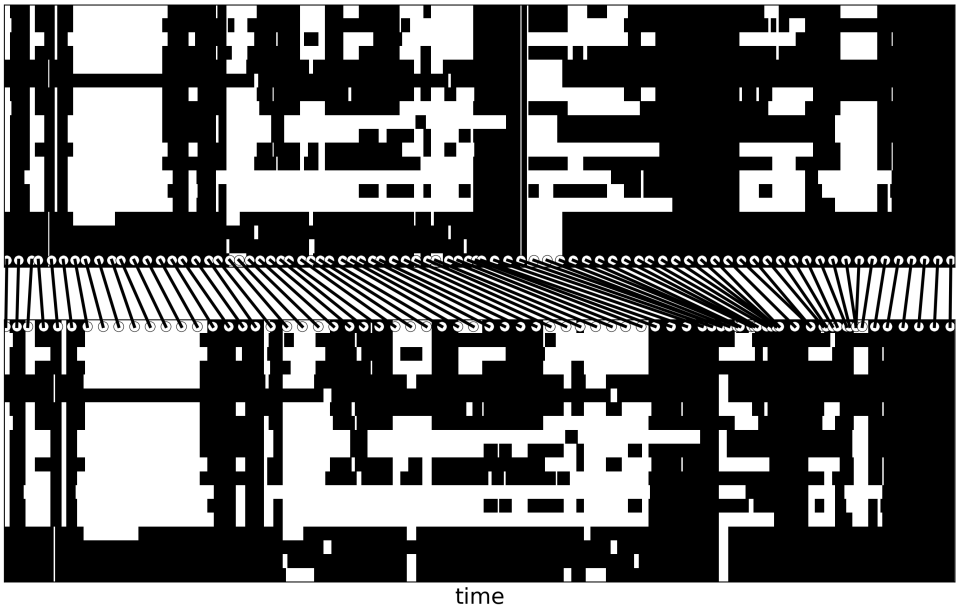


Figure 3.6: Example of produced alignment between two fully-observed M_{PNP} matrices.

3.4.2. DYNAMIC TIME WARPING

Once the distance matrix D is computed, the time map between $M^{\text{AV}}_{\text{PNP}}$ and $M^{\text{S}}_{\text{PNP}}$ is determined by solving the optimization problem: $P^\star = \arg\min_P \text{cost}(D,P)$ where $P = \{(p_\ell \rightsquigarrow p_{\ell+1})\}$ is a path through the items of D having a cost defined by the function $\text{cost}(D,P)$. More specifically, $p_\ell = (i^{\text{AV}}_\ell, i^{\text{S}}_\ell)$ is a coordinate of an element in D. The cost function is defined as $\text{cost}(D,P) = \sum_{\ell=1}^{|P|} d_{i^{\text{AV}}_\ell, i^{\text{S}}_\ell}$ The aforementioned problem is efficiently solved via dynamic programing using the well-known dynamic time warping (DTW) algorithm. Examples of $P^\star$ paths computed via DTW are shown in Figure 3.5.

Once $P^\star$ is found, the time map f_{sync} is computed through the linear interpolation of the correspondences in $P^\star$, i.e., the set of coordinates $\{p^\star_\ell = (i^{\text{AV}}_\ell, i^{\text{S}}_\ell)\}$. This map allows to define correspondences between the two matrices, as shown in the example of Figure 3.6.

3.5. EXPERIMENTAL SETUP

In this section, we describe our experimental setup including details about the dataset. In order to ensure the reproducibility of the experiments, we release the code and share the URLs of the analyzed freely available MIDI files[1].

We evaluate the performances of our method on a set of 29 symphonic pieces composed by Beethoven, Mahler, Mozart and Schubert. The dataset consists of 114 MIDI files. Each MIDI file contains a number of tracks corresponding to different parts per-

[1] http://homepage.tudelft.nl/f8j6a/ISMIR2014baz.zip

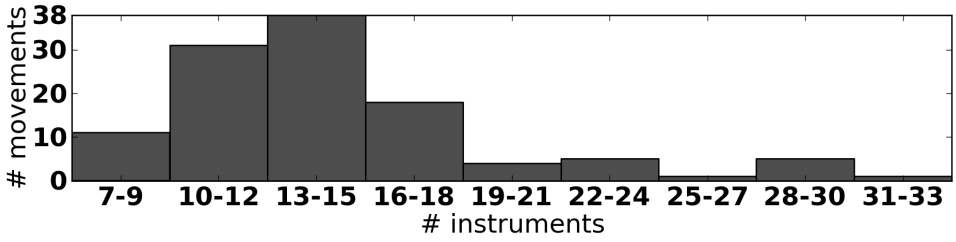


Figure 3.7: Distribution of the number of instrumental parts across performances in the data set.

formed in a symphonic piece. For instance, first and second violins are typically encoded in two different parts (e.g., "Violin 1" and "Violin 2"). In such a case, we keep both tracks separate since musicians in the visual channel can be labeled according to the score which they perform (and not just by their instrument). We ensured that the MIDI files contain tracks which are mutually synchronized (i.e., MIDI files of type 1). The number of instrumental parts, or MIDI tracks, ranges between 7 and 31 and is distributed as shown in Figure 3.7.

For each MIDI file, we perform the following steps. First, we generate one $M^{\mathrm{S}}_{\mathrm{PNP}}$ matrix using a fixed reference tempo of 100 BPM. The reason why we use the same value for every piece is that we evaluate our method on artificial warping paths, hence we do not need to adapt the sliding window parameters to any actual performance. Then we generate one random time-warping function which has two functions: (i) it is used as ground-truth when evaluating the alignment performance, and (ii) it is used to make one time-warped P/NP matrix $M^{\mathrm{AV}}_{\mathrm{PNP}}$. The latter is used as template to build noisy copies of $M^{\mathrm{AV}}_{\mathrm{PNP}}$ and evaluate the robustness of our method. Each template P/NP matrix is used to generate a set of noisy P/NP matrices which are affected by different pre-determined amounts of noise. We consider two sources of noise: mistaken and missing labels. For both sources, we generate the following percentages of noisy labels: 0% (noiseless), 2%, 5%, 10%, 20%, 30%, 40% and 50%. For every pair of noise percentages, e.g., 5% mistaken + 10% missing, we create 5 different noisy versions of the original P/NP matrix[2]. Therefore, for each MIDI file, the final set of matrices has the size $1+(8\times 8-1)\times 5 = 316$. Overall, we evaluate the temporal alignment of $316 \times 114 = 36024$ P/NP sequences.

For each pair of M_{PNP} matrices to be aligned, we compute the matching rate by sampling f_{sync} and measuring the distance from the true alignment. A match occurs when the distance between linked time points is below a threshold. In our experiments, we evaluate the matching rate using three different threshold values: 1, 2 and 5 seconds.

Finally, we apply the video-based P/NP label extraction strategy described in Section 3.3.2 to a multiple camera video recording of the 4th movement of Symphony no. 3 op. 55 of Beethoven performed by the Royal Concertgebouw Orchestra (The Netherlands). For this performance, in which 54 musicians play 19 instrumental parts, we use the MIDI file and the correspondent performance-score temporal alignment file which are shared by the authors of [44]. The latter is used as ground truth when evaluating the synchronization performance.

[2]We do not add extra copies for the pair (0%,0%), i.e., the template matrix.

3.6. RESULTS

In this section, we present the obtained results and provide answers to the research questions posed at the beginning of this chapter. We start by presenting in Figure 3.8 the computed matching rates in 3 distinct matrices, one for each threshold value. Given a threshold, the overall matching rates are reported in an 8×8 matrix since we separately compute the average matching rate for each pair of mistaken-missing noise rates.

Overall, we see two expected effects: (i) the average matching rate decreases for larger amounts of noise, and (ii) the performance increases with the increasing threshold. What was not expected is the fact that the best performance is not obtained in the noiseless case. For instance, when the threshold is 5 seconds, we obtained an average matching rate of 81.7% in the noiseless case and 85.0% in the case of 0% mistaken and 10% missing labels. One possible explanation is that 10% missing labels could give more "freedom" to the DTW algorithm than the noiseless case. Such freedom may lead to a better global optimization. In order to fully understand the reported outcome, however, further investigation is needed, which we leave for future work.

As for our first research question, we conclude that the alignment through P/NP sequences is more robust to missing labels than to mistaken ones. We show this by the fact that the performance for 0% mistaken and 50% missing labels are higher than in the opposite case, namely for 50% mistaken and 0% missing labels. In general the best performance is obtained for up to 10% mistaken and 30% missing labels.

In the second research question we address the behavior at different time resolutions. Since labels are sampled every second, it is clear why acceptable matching rates are only obtained at coarse resolution (namely for a threshold of 5 seconds).

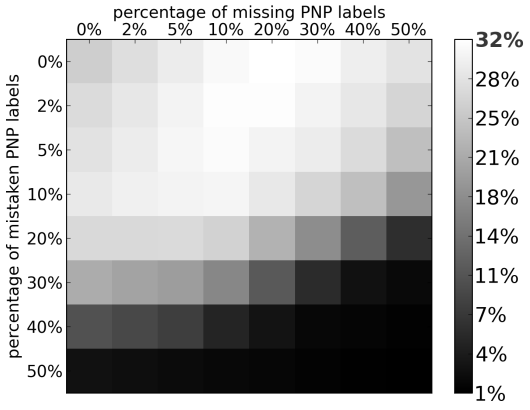
Finally, we comment on the results obtained when synchronizing through the P/NP labels assigned via visual analysis. The P/NP matrix, shown in Figure 3.9a, is affected by noise as follows: there are 53.95% missing and 8.65% mistaken labels.

We immediately notice the large amount of missing labels. This is mainly caused by the inability to infer a P/NP label at those time points when all the musicians of a certain instrumental part are not recorded. Additionally, some of the image clusters generated as described in Section 3.3.2 are not pure and hence labeled as X.

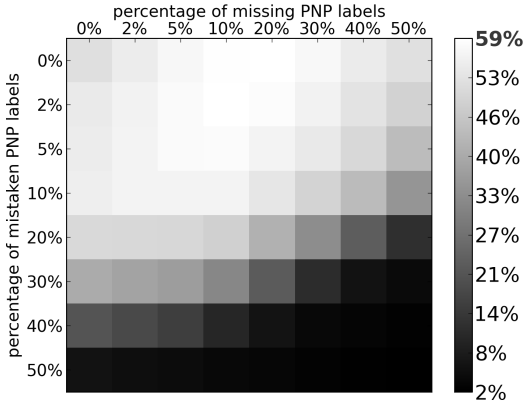
The obtained synchronization performance at 1, 2 and 5 seconds of tolerance are respectively 18.74%, 34.49% and 60.70%. This is in line with the results obtained with synthetic data whose performance at 10% of mistaken labels and 50% of missing for the three different tolerances are 24.3%, 44.2% and 65.9%. Carrying out the second experiment was also useful to get insight about the distribution of missing labels. By inspecting Figure 3.9a, we notice that such a type of noise is not randomly distributed. Some musicians are sparsely observed over time hence leading to missing labels patterns which differ from uniform distributed random noise.

3.7. DISCUSSION

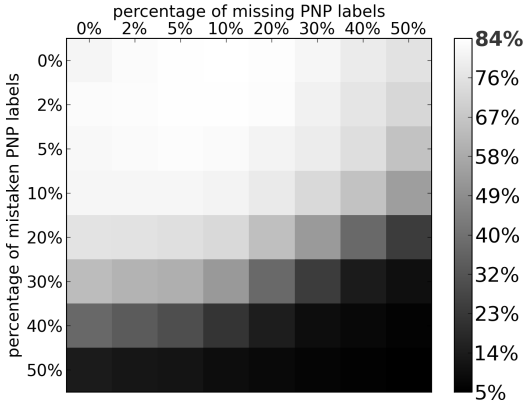
In this chapter, we presented a novel method to synchronize score information of a symphonic piece to a performance of this piece. In doing this, we used a simple feature (the act of playing or not) which trivially is encoded in the score, and feasibly can be obtained from the visual channel of an audiovisual recording of the performance. Unique about



(a) Tolerance: 1 second.

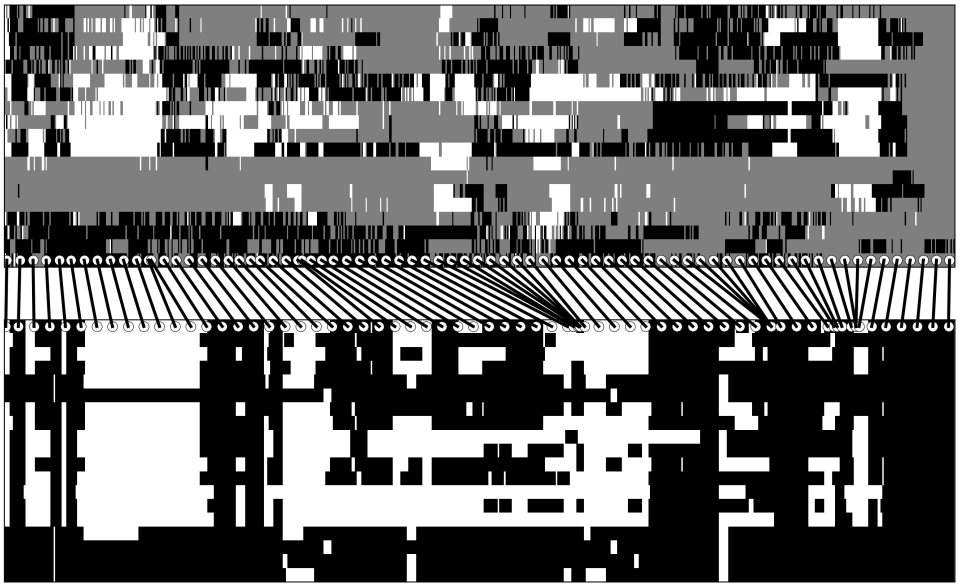


(b) Tolerance: 2 seconds.

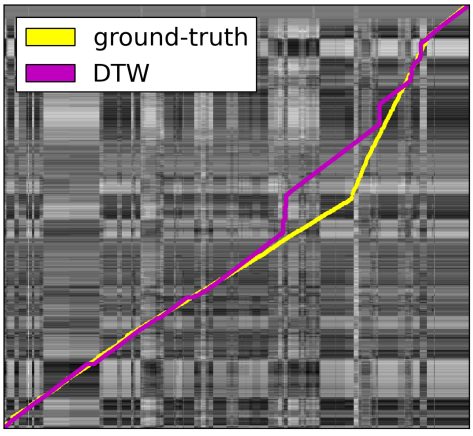


(c) Tolerance: 5 seconds.

Figure 3.8: Average matching rates as a function of the percentage of mistaken and/or missing labels at different tolerance thresholds.



(a) $M^{\mathrm{AV}}_{\mathrm{PNP}}$ and $M^{\mathrm{S}}_{\mathrm{PNP}}$ matrices.



(b) DTW result and ground-truth.

Figure 3.9: Real data example: synchronization results by exploiting P/NP labels derived by a video recording.

our approach is that both for the score and the performance, we start from measuring individual musician contributions, and only then aggregate up to the full ensemble level to perform synchronization. This makes a case for using the visual channel of an audio-visual recording. In the audio channel, which so far has predominantly been considered for score-to-performance synchronization, even if separate microphones are used per instrument, different instruments will never be fully isolated from each other in a realistic playing setting. Furthermore, audio source separation for polyphonic orchestral music is far from being solved. However, in the visual channel, different players are separated by default, up to the point that a first clarinet player can be distinguished from a second clarinet player, and individual contributions can be measured for both.

Our method still works at a rough time resolution, and lacks the temporal sub-second precision of typical audio-score synchronization methods. However, it is computationally inexpensive, and thus can quickly provide a rough synchronization, in which individual instrumental part contributions are automatically marked over time. Consequently, interesting follow-up approaches could be devised, in which cross- or multimodal approaches might lead to stronger solutions, as already argued in [59, 30].

For the problem of score synchronization, a logical next step is to combine our analysis with typical audio-score synchronization approaches, or approaches generally relying on multiple synchronization methods, such as [33], to investigate whether a combination of methods improves the precision and efficiency of the synchronization procedure. Our added visual information layer can further be useful for devising structural performance characteristics, e.g., the occurrence of repeats. Our general synchronization results will also be useful for source separation procedures, since the obtained P/NP annotations indicate active sound-producing sources over time. Furthermore, results of our method can serve applications focusing on studying and learning about musical performances. We can easily output an activity map or multidimensional time-scrolling bar, visualizing which orchestra parts are active over time in a performance. Information about expected musical activity across sections can also help directing the focus of an audience member towards dedicated players or the full ensemble.

Finally, it will be interesting to investigate points where P/NP information in the visual and score channel clearly disagree. For example, in Figure 3.3, some time after the flutist starts playing, there is a moment where the score indicates a non-playing interval, while the flutist keeps a playing pose. We hypothesize that this indicates that, while a (long) rest is notated, the musical discourse actually still continues. While this also will need further investigation, this opens up new possibilities for research in performance analysis and musical phrasing, broadening the potential impact of this work even further.

4

ON DETECTING THE PLAYING/NON-PLAYING ACTIVITY OF MUSICIANS IN SYMPHONIC MUSIC VIDEOS

Chapter 3 indicates that information on whether a musician in a large symphonic orchestra plays her instrument at a given time stamp or not is valuable for performance-to-score synchronization. This result motivates us to address the challenges of devising a vision-based system that extracts sequences of playing/non-playing (P/NP) labels per musician from symphonic orchestra video recordings. Such a system must deal with multiple or single cameras, moving or static ones, occlusions and clutter. Each scene must be segmented, and musicians must be recognized across cameras and viewpoints. Conductor and audience must be excluded from the P/NP labeling step. Even if one were to combine all the most recent breakthroughs in deep learning, designing such a system would still be tricky. Also, following a data-driven approach requires labeled data that is far from being available. We therefore pursue a semi-automatic annotation strategy aiming to efficiently and effectively combine automatic analysis and human annotation. Then, in order to identify the open challenges and the limitations of the proposed method, we carry out a detailed investigation of how different modules of the system affect the overall performance.

This chapter was published as: Alessio Bazzica, Cynthia C. S. Liem, Alan Hanjalic. On detecting the playing/non-playing activity of musicians in symphonic music videos. *Computer Vision and Image Understanding* Vol.144, pp. 188-204, March 2016

Rapidly developing multimedia technology has opened up new possibilities for bringing the full symphonic music concert experience out of the concert hall and into people's homes. New emerging platforms, like *RCO Editions*[1] and the *Berliner Philharmoniker's Digital Concert Hall*[2] are enriching audiovisual recordings of symphonic music performances to make them more informative and accessible offline, in a non-linear fashion and from multiple perspectives. Such platforms rely on the new generation of automatic music data analysis solutions. For instance, loudness and tempo can be estimated continuously over time and visualized as animations [28]. Notes can be detected and analyzed to reveal and visualize repeated parts of a piece [69]. Sheet music scores can be synchronized to the audio recording to allow users to follow the scores while listening to the music [5]. Furthermore, the sound produced by different instruments can be isolated via source separation [34], which could be deployed to zoom in on a particular instrument or instrumental section [43].

While the solutions mentioned above primarily rely on an analysis of the audio channel of the performance recording, the visual channel has remained underexploited. In addition to enabling the development of new functionalities of platforms like *RCO Editions* and *Berliner Philharmoniker's Digital Concert Hall* not covered by audio analysis, the analysis of the visual channel could also help to resolve some of the critical challenges faced by audio analysis. For instance, achieving reliable sound source separation is challenging in the case of large ensembles where the sound produced by many different instruments overlaps both in time and frequency [9].

In this chapter, we focus on the analysis of the visual channel of the audiovisual recording of a symphonic music performance and address the problem of annotating the activity of individual musicians with respect to *whether they play their instruments at a given timestamp or not.* The envisioned output of the solution we propose in this chapter is illustrated in Figure 4.1, where playing and non-playing musicians are isolated as indicated by respectively the green and red rectangles.

Knowing the playing (P) and non-playing (NP) labels for each musician allows the annotations of an audiovisual recording to be enriched in a way that is complementary and supportive to audio-only analysis. For instance, repeats and solo parts could be detected also by analyzing the sequences of P/NP labels to allow novel non-linear browsing functionalities (e.g., skip to solo trumpets, skip to "tutti"). The problem of performance-to-score synchronization, which is typically addressed through audio-to-audio alignment [70], could also be approached in a multimodal fashion by combining state-of-the-art auditory features and P/NP labels [12].

Related methods operating on the visual channel typically deploy a standard classification paradigm and learn visual models for human actions [84, 106]. The disadvantage of this approach in the problem context of symphonic music concert videos is that the models may not be generic enough to cover the wide variety of instruments used and the ways the P/NP activities of individual musicians could be visually recorded. Additionally, a realistic view at the reliability of solving this classification problem reveals the need for manual human intervention in order to correct unavoidable classification errors, in particular in a professional context when high-quality annotation output is required.

[1] `http://www.concertgebouworkest.nl/en/rco-editions/`
[2] `http://www.digitalconcerthall.com/`

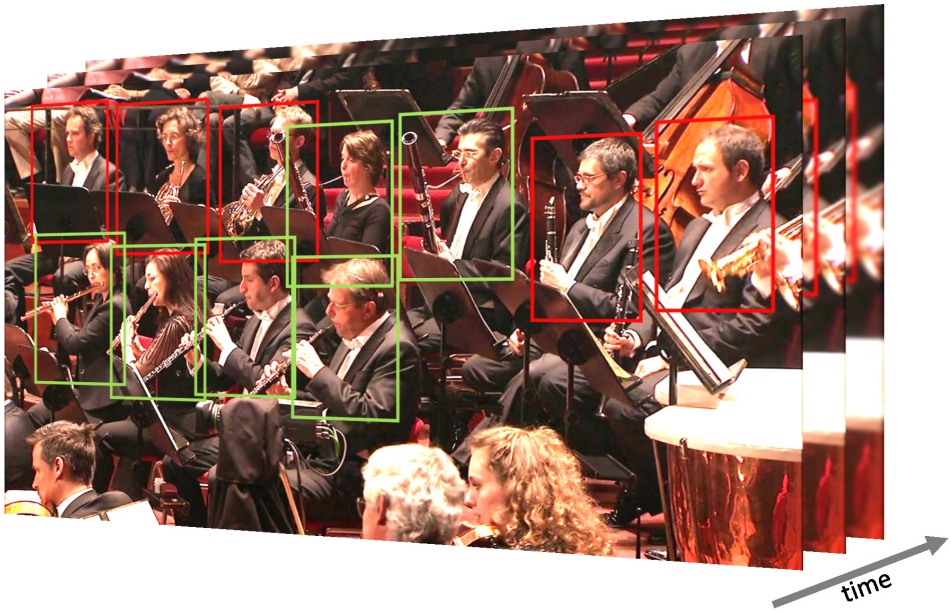


Figure 4.1: Envisioned output of the method proposed in this chapter. Green (red) bounding boxes mark the musicians that play (don't play) their instrument at a given time stamp.

The method we propose in this chapter is geared not only towards neutralizing the disadvantage mentioned above, but also towards incorporating human intervention in the way that is as efficient and effective as possible. We implement our proposed solution to assign P/NP labels per musician to the timeline of a symphonic music performance as a modular framework so that we can provide answers to the following research questions:

- **RQ1**: How reliably can we isolate clusters of images depicting individual musicians from the keyframes extracted from a music video?
- **RQ2**: How accurately can sequences of P/NP labels be generated?
- **RQ3**: What is the tolerance of the proposed framework to errors in different modules?
- **RQ4**: Is a static image informative enough to reveal whether a musician is playing an instrument?
- **RQ5**: What is the relation between the amount of human intervention and the quality of the obtained P/NP label sequences?

The chapter is organized as follows. We start by explaining in Section 4.1 the context in which we operate in this chapter and that characterizes the realization and recording of a typical symphonic music performance. By taking into account the properties

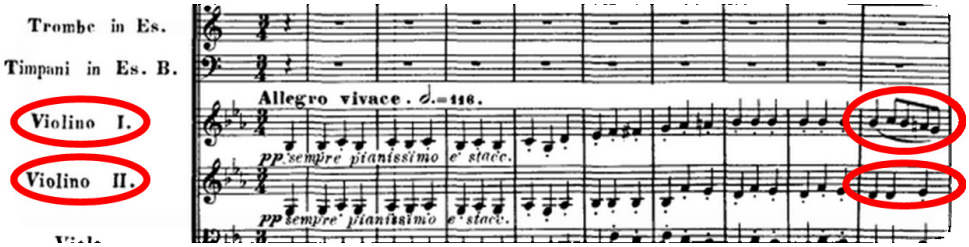


Figure 4.2: Excerpt of a score: same instrument, different instrumental parts.

of the work context and the related limitations, we proceed in Section 4.2 by analyzing the usability of the existing related work and in Section 4.3 by stating our novel contribution and explaining the rationale behind our proposed framework. We introduce the notation, set the goals and make assumptions in Section 4.4. We present our method in Section 4.5 elaborating on the realization of different framework modules. After we explain the experimental setup in Section 4.6, we present our assessment of the framework in Section 4.7 where we also provide answers to the research questions posed above. We conclude with a discussion section in which we also present future research directions (Section 4.8).

4.1. CHARACTERISTICS OF A SYMPHONIC ORCHESTRAL RECORDING

A symphonic orchestra consists of a large number of musicians organized in *sections* (string, brass, woodwind or percussion). Sections are further divided into *instrumental parts*. Each instrumental part consists of a number of musicians playing one particular instrument and following a specific musical score. For instance, in Figure 4.2 the instrumental parts "Violino I" and "Violino II" play different notes even if the instrument is the same (violin). According to the scores, when one musician belonging to one instrumental part is (not) playing, all the other musicians performing the same instrumental part are expected to be (in-)active as well. This usually holds even in the *divisi* case[3].

Performance recordings may differ depending on several factors like, for instance, the type of environment (indoor vs. outdoor), the number of cameras and whether camera motion occurs. In this chapter we focus on the indoor case, and we consider two possible types of recording: single- and multiple-camera recordings. The former is made from a fixed point of view and with a fixed zoom factor. In this way, the whole ensemble is always visible and each musician covers the same region of the video frames throughout the video. The latter typically involves multiple-cameras positioned around and on the stage, with the possibility to zoom and pan. This type of recording typically serves as input to a team of experts in order to create an edited video using a script (e.g., "when the 100th bar of the scores starts, the 3rd camera switches to a close-up on the first clarinet player"). Thereby, the visual channel mainly focuses on (parts of) the orchestra, but can also show the conductor and the audience in the concert hall.

[3] http://en.wikipedia.org/wiki/Unison#Divisi

Figure 4.3: Examples of video frames showing different settings of musicians and their instruments on the stage during the symphonic music performance.

Both in the single- and multiple-camera recordings, depending on the camera position, some musicians appear frontally, some non-frontally, and some even from the back, (fully) occluding their instruments. As illustrated in Figure 4.3, the setting of the orchestra on the stage is rather dense, resulting in significant occlusion of individual musicians and their instruments. A video frame taken from the visual recording of the performance can therefore contain multiple musicians, not necessarily belonging to the same section or instrumental part.

The characteristics of the context in which we operate, as described above, have significant impact on the extent to which we can rely on the existing related work in conceptually developing our proposed solution, but also on the way how we approach the definition and implementation of the modules of our framework. This will be explained in more detail in the following sections.

4.2. RELATED WORK

The problem of extracting the sequence of P/NP labels for each musician continuously over time from an audiovisual recording of a symphonic music performance has not been directly tackled so far. We explore here, however, the usability of a number of related approaches.

4.2.1. DETECTING THE PLAYING/NON-PLAYING ACTIVITY

Regarding the detection of P/NP activity in general, we classify the existing work into hardware-based, score-based, audio-based and vision-based approaches.

APPROACHES BASED ON DEDICATED HARDWARE

Probably the most intuitive approach to inferring the activity of a particular musician is via dedicated hardware [65, 89]. While theoretically effective, the critical deficiency of such an approach is that it requires obtrusive settings, which are unnatural in the work context described in the previous section. For instance, a webcam may need to be mounted above the vibraphone in order to detect which bars are covered by the mallets [89].

SCORE-BASED APPROACHES

An alternative to deploying obtrusive dedicated hardware is to rely on the data from the regular audiovisual recording, possibly in combination with the available supplementary material. For instance, the P/NP states could be inferred by analyzing a synchronized music score, that is, by looking at presence of notes and rests in each bar as done in [12]. Such a method allows to infer P/NP labels for every instrumental part at every time point. However, as pointed out in [34], even if full scores are freely available for many classical pieces, they are rarely aligned to a given audio recording. In order to pursue this strategy, the score and the performance recording need to be synchronized using existing alignment methods [77, 48]. Performing such synchronization can be challenging, especially in presence of structural variations between the score and the recording (e.g., omission of repetitions, insertion of additional parts). However, in practice, even though partial alignment methods exist, likely failures in the structural analysis and subsequent segment matching steps can lead to corrupted synchronization results [68, 91].

AUDIO-BASED APPROACHES

Source separation techniques could be considered to isolate the sound of each instrument and infer P/NP labels by analyzing the isolated instrument-level signals. In view of the context in which we operate, however, this approach is not likely to be successful. Typically, only a limited number of instruments can be recognized with an acceptable accuracy. In [62], the authors address the challenging problem of recognizing musical instruments in multi-instrumental and polyphonic music. Only six timbre models are used, hence this approach has limited utility for symphonic orchestras where more models would be needed. In [9] the number of recognized instruments is 25, but the recognition is performed in those parts of a piece in which a single instrument is played alone. This limits the applicability of this approach in our work context to the rare solo segments only. While it was shown in [34] and [94] that effective audio source separation needs prior information derived by synchronized music scores, such an informed source separation approach would include the limitations of those related to score synchronization, as discussed above.

VISION-BASED APPROACHES

Insufficient applicability of audio-based approaches in our work context makes us investigate the alternatives relying on the visual channel. When video recordings are available, we can see musicians interacting with their instruments. They hold them in a certain way when playing, while they assume different body poses when not playing. In the former case, musicians also move in order to make music (e.g., bowing, pressing keys, opening valves, moving torso to help blowing). Hence, visual appearance and motion information could be potentially useful in inferring whether musicians are playing or resting.

In view of the above, one could explore human-object interaction (HOI) by analyzing visual object appearances in a static image — i.e., a keyframe extracted from a video. For this purpose, investigation of presence of objects of interest (in this case, music instruments), spatial relationships between objects and human body parts has been found promising [105, 106]. Dedicated datasets have been developed for this line of research, a good example of which is the “people playing musical instrument” (PPMI) dataset [105].

Figure 4.4: Examples of the setting of musicians and their instruments as considered by the existing vision-based approaches.

Alternatively, in video action recognition, both visual appearance and motion information are exploited [84, 72, 78]. State-of-the-art performance with popular datasets, like the UCF101 [87], shows that several actions, like "playing violin", can be detected.

The aforementioned methods for HOI detection and video action recognition are based on a supervised classification approach. While such methods are sophisticated and in general have the potential to outperform previously discussed non-visual approaches, they require visual input of a particular type in order to train reliable classifiers. For example, as illustrated in Figure 4.4, the PPMI dataset consists of images containing sufficiently large and well visible regions corresponding to a human and an instrument. This makes the aforementioned methods not applicable to the situations addressed in this chapter and illustrated by the orchestra settings in Figure 4.3.

4.2.2. DETECTING, ISOLATING AND RECOGNIZING MUSICIANS

In order to design a system which yields a sequence of P/NP labels for each musician, we first have to solve the *musician diarization* problem. In other words, we want to understand which musician appears when and where in the video frames. The related literature for this task includes works about detecting, tracking and recognizing people in videos. Then, for each musician appearing in the scene, the regions of the video frames which are informative for the inference of the sought P/NP labels have to be isolated by means of image segmentation.

When the input video consists of a set of fixed-camera recordings, the positions of the musicians in the scene can be manually annotated using a reference video frame from each video (e.g., the first one). Such a manual initialization step is time inexpensive and can be done because the musicians do not change their position throughout the performance. Therefore, the annotated coordinates can be used for the whole recording.

In the case of a video recording consisting of different shots resulting from camera

zoom-in and pan actions, manual-only annotation of musicians becomes too complex and needs to be helped by automatic visual analysis tools. Off-the-shelf face detectors, face clustering and recognition methods can be deployed for this purpose, possibly supported by a face tracking algorithm to collect and verify the evidence from consecutive video frames [54, 83].

Specifically related to face clustering, state-of-the-art solutions are typically based on context-assisted and constrained clustering [107, 102], possibly including human intervention in order to produce high-quality results [108]. For instance, clothing information is exploited to discriminate people with similar faces but dressed differently [107]. *Cannot-link* constraints are used to avoid that two faces detected in the same image fall into the same cluster. People can be tracked and *must-link* constraints can be inferred by the generated face tracks [102]. Face-related visual features can be extracted for every detection, or just when the estimated quality of the face image is good enough to extract reliable information [7]. Finally, to avoid that too many face clusters are generated for the same identity, semi-automatic algorithms can be used to iteratively merge clusters [108].

The existing methods are typically tested only on frontal faces. Alternatively, as done in [7], the detected profile faces are continuously tracked over time, but used at the clustering step only when a switch to a (near-)frontal view occurs. In view of our problem context described in Section 4.1, this focus on (near-)frontal faces makes the methods described above insufficiently suitable as modules of our envisioned framework. This was also revealed by an initial investigation we performed to inform the design choices for this framework, the results of which are reported in Section 4.6.1.

4.3. CONTRIBUTION AND RATIONALE

In view of the fact that the visual channel of the symphonic music recordings is available, and based on the conclusions drawn in Section 4.2.1 regarding the performance-related and practical disadvantages of hardware-, score- and audio-based methods, in our approach we focus on the visual channel to infer the P/NP activity per musician. In order to cope with inevitable errors of automated visual analysis of challenging HOI cases in our application context and to secure high accuracy of the obtained P/NP label sequences, we choose for a semi-automatic approach, where human intervention is efficiently and effectively combined with automated analysis. The value of such hybrid approach for video annotation has already been shown in the past (e.g., [98]).

The proposed method involves two main steps, musician diarization and label assignment per musician and time stamp. Learning from the analysis of the related work, we pursue the development of the solutions for both steps by making the following critical design choices.

Regarding the musician diarization step, as argued in Section 4.2.2, we need a more reliable method for identifying the musicians than what the state-of- the-art in the field currently offers. While we can rely on standard face detection methods, the choice of the face clustering method leaves room for improvement, primarily in view of the requirement to obtain the face clusters that are as pure as possible. This purity is essential because errors in clustering directly propagate to the resulting P/NP label sequences. We have initially considered the approach described in [108], which semi-

automatically merges an initial set of face clusters assuming that all of them are close to being 100% pure. However, our preliminary experiments deploying this method on our concert video data have revealed that only a part of the generated clusters can be obtained as almost 100% pure, while the remaining clusters are too noisy. Moreover, as reported in Section 4.6.1, we found that different features and image regions from those reported in [108] may yield much better face clusters on our data. We therefore investigated alternative ways to increase the number of pure face clusters by strategically employing human annotators. Beside alleviating the impact of unavoidable non-pure clusters, such a semi-automatic strategy can be exploited to efficiently and effectively reject clusters of non-relevant targets — i.e., conductor and audience but also false face detections. Our approach turns out to require significantly simpler visual analysis tools than the complex, sophisticated person identification methods discussed in Section 4.2.2.

Once the musician diarization problem is solved, we infer the P/NP activity per musician. As opposed to the methods discussed in Section 4.2.1, we deploy the information in the visual channel in such a way to better exploit and match the characteristics of the work context we address, however, at the same time, being able to handle the full scope of content generated in such context — i.e., any performance of any symphonic orchestra. Specifically, instead of aiming to develop generic HOI models via a supervised learning approach, we base our solution on the clustering principle. We search for clusters ad-hoc, for a given video of a performance. Thereby, we focus on the following cluster categories in which we group the detected musicians' images: (i) musician identity, (ii) point of view and (iii) playing/non-playing activity. Creating clusters for these categories, labeling them appropriately and propagating the labels to the individual video frames will then automatically result in the targeted P/NP label sequences. The advantage of this approach, as opposed to those based on training the HOI models, is that there is no dependence on the type of instruments nor on the actual way how HOI is represented — i.e., in which way a musician interacting with her instrument is depicted in a particular recording, as long as HOI activity is depicted consistently along the video. In our work context, consistency in general can be assumed due to the following characteristics: (a) the number of musicians is limited, (b) the setting of the orchestra on the stage is constant within one performance, and (c) the variations by which musicians appear in the video are limited by the limited number of camera views.

In view of the above, our proposed approach can now conceptually be summarized as follows. By exploiting the redundancy of each analyzed video recording (e.g., multiple occurrences of the same camera angle), we accumulate information on the dominant appearances of various musicians in terms of their instrument-playing activities. These dominant appearances are then turned into clusters that coincide with P/NP activities to be labeled accordingly, through human intervention. This combination aims to achieve high level of output quality eliminating the need for extensive model training and making the annotation problem more tractable. We refer to Section 4.5 for a detailed explanation of the different steps of our method.

4.4. NOTATION, GOALS AND ASSUMPTIONS

Given a multi-camera video recording of a symphonic music performance, we aim at inferring for each performing musician the P/NP labels over time. A label is assigned at

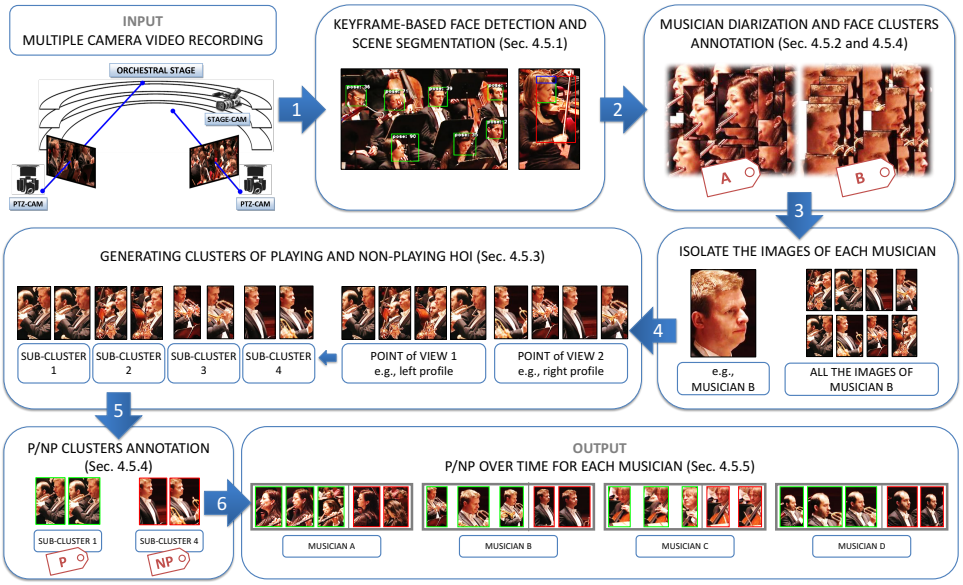


Figure 4.5: Illustration of the modular framework implementing the proposed method for extracting P/NP labels per musician from a video recording.

regular intervals (e.g., every second) at the time point t starting from the first frame in the video. The videos generated by different cameras are denoted as the set $V = \{\mathbf{v}_i(n)\}$ where $n \in \mathcal{N} = \{0 \dots L-1\}$ denotes the frame index and L is the total number of frames. All the videos are synchronized in time and have the same length L. We further denote by M_{GT} the set of performing musicians, where GT stands for "ground truth", and by $|M_{\text{GT}}|$ the set size.

In view of this notation, our goal is to learn the function $\text{PNP}_m(t) : \mathcal{T} \rightarrow \{\text{P}, \text{NP}, \text{X}\}$, which determines the P/NP label at the time points $t \in \mathcal{T}$ for each musician $m \in M_{\text{GT}}$. The additional label X represents the cases when the label is not determined. As discussed in Section 4.6, we evaluate the accuracy of the learned PNP functions as well as the amount of determined P/NP labels.

While we count on multi-camera recordings (see Section 4.1), the minimum required number of cameras for our approach is one and camera motion is allowed (e.g., panning, zoom-in, zoom-out). The method does not require information about which instruments are played during the performance. Furthermore, we do not make any assumption regarding the *timeline coverage*. In other words, while we do not require that every musician is continuously captured by a camera during the performance, it can also happen that at a given time point the same musician is captured by multiple cameras. We only require that for each musician $m \in M_{\text{GT}}$ her corresponding instrumental part is known. This knowledge allows to make a partition of $M_{\text{GT}} = \cup_{h=1}^{H} M_{\text{GT}}^h$ into H mutually disjoint subsets and to recover part of the missing P/NP labels as described in Section 4.5.6.

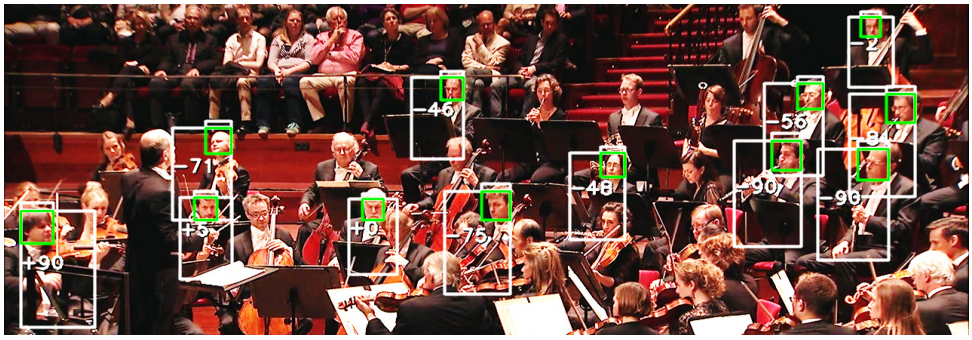


Figure 4.6: Example of keyframe segmentation. For each detected face, the upper body region is determined considering the estimated head pose. In this way, we find the region of the image where the HOI is expected to be visible.

4.5. METHOD DESCRIPTION

In this section, we describe the framework representing the realization of our proposed method and illustrated in Figure 4.5. First, the keyframes are extracted from the given multiple-camera recording and processed to detect and isolate musicians in the scene (details reported in Section 4.5.1). A musician diarization problem is then solved by combining face clustering and human annotation (respectively discussed in Section 4.5.2 and Section 4.5.4). In this way, all the images belonging to each performing musician will be effectively and efficiently isolated and linked to the correspondent musician identity label.

At this point, instead of using pre-trained visual models which independently infer playing and non-playing labels for each single image, we rely on a novel unsupervised method for the reasons discussed in Section 4.2.1. Such method, described in Section 4.5.3, aims at learning ad-hoc discriminative visual patterns for each performing musician to be used for distinguishing playing activities from non-playing ones. This approach produces sub-clusters of P/NP images which will be manually labeled accordingly using the procedure described in Section 4.5.4. Finally, the sought P/NP label sequences are computed as described in Section 4.5.5.

4.5.1. KEYFRAME-BASED FACE DETECTION AND SCENE SEGMENTATION

For every video $\mathbf{v}_i(n) \in V$, one keyframe $\mathbf{f}_i^k$ is extracted at predetermined time points n_i^k (e.g., at regular intervals) where n_i^k is the k-th time point for the i-th video. The set of keyframes extracted from $\mathbf{v}_i(n)$ is denoted as $F_i = \{\mathbf{f}_i^k\}_{k=0}^{K-1}$.

For each keyframe, we detect faces and estimate the head pose angle. Regarding face detection, we rely on standard, off-the-shelf approaches, as described in detail in Section 4.6.3. In this way we build the sets $D_i^k = \{\mathbf{d}_i^{k,l}\}$, where $\mathbf{d}_i^{k,l}$ is the l-th detection in the keyframe $\mathbf{f}_i^k$. Each detection $\mathbf{d}$ is defined as $(\mathbf{b}, \theta)$ where $\mathbf{b} = (x, y, w, h)$ is the vector encoding the face bounding box geometry and $\theta \in [-90, +90]$ is the estimated head pose.

Finally, we exploit the face bounding box geometry using simple but effective heuristics to identify visual information supplementary to the face that can be valuable for the

subsequent clustering steps. Here we focus in particular on the hair and upper body of the musician, and related to the latter, on those regions where the instrument can be expected. Given a face detection $\mathbf{d} = (\mathbf{b}, \theta)$, the two additional bounding boxes are inferred using the Vitruvian man ratios as done in [54]. The hair bounding box is defined as $(x, y - h/4, w, h/4)$. As for the upper body segmentation, we extend the heuristic presented in [54], which is limited to the frontal faces, in order to infer the region of interest for any value of $\theta \in [-90, +90]$. The upper body bounding box is therefore computed as a function of $\mathbf{b}$ and θ. The underlying idea is to look at the region of the image in the direction of the musician's gaze where we expect to see the instrument. If $\theta > \theta^\star (< -\theta^\star)$, we look at the right(left) side of the face bounding box. When $\theta \in [-\theta^\star, +\theta^\star]$, we center the face bounding box horizontally. $\theta^\star$ is the critical angle used to discriminate frontal and profile faces. The upper body region includes the head and the torso. The torso has a height of $2.6 \times h$ and a width of $2.3 \times w$ [54]. An illustration of the results of this segmentation process is given in Figure 4.6.

4.5.2. MUSICIAN DIARIZATION VIA FACE CLUSTERING

Grouping the detected faces into clusters of individual musicians can be performed in different ways. We consider four possibilities that we refer to as (i) *unconstrained*, (ii) *context-assisted*, (iii) *constrained*, and (iv) *context-assisted and constrained*. The *unconstrained* method relies on the visual information only consisting of visual features extracted from the face and hair regions. In addition to visual information, *context-assisted* methods also rely on the *visual context* of the detected face. The upper body region extracted for a face may help discriminating between those musicians whose faces look similar, but who play different instruments. Similarly, a scene descriptor could be deployed to discriminate between similar faces belonging to musicians placed in different parts of the orchestra. In the *constrained* method, we again deploy face- and hair-related visual features, but also exploit the fact that multiple face detections in the same frame should belong to different identities. We build a sparse matrix of cannot-link constraints CL for each pair of faces $(\mathbf{d}_i^{k,l}, \mathbf{d}_i^{k,l'}) | l \neq l'$ detected in the same keyframe. CL is then used to ensure that multiple detections in the same keyframe fall in different face clusters. Another type of constraint which could be deployed is the must-link constraint. During a shot, the detected faces could namely be tracked and therefore linked over time. However, taking this into account would increase the complexity of the system and might not generate exact constraints as in the case of the CL set (e.g., due to the mistakes with crossing face tracks generating wrong must-link constraints). Finally, the *context-assisted and constrained* method exploits both visual context information and the cannot-link constraints.

As for choosing a suitable number of clusters, we consider the following information that can be reasonably defined a priori. The number of musicians $|M_{\text{GT}}|$ may vary, but a typical symphonic ensemble ranges from 50 to 100 players. In addition to the orchestra, some of the frames also show the conductor and the audience. Together, the musicians, conductor and audience form the set E of "entities" to be isolated. Furthermore, the same entity can be recorded from different cameras/viewpoints, and also with variations (e.g., due to camera zoom-in). Therefore, the number of expected clusters can be estimated as $\lceil \alpha \times |E| \rceil$, where $\alpha \geq 1$ is a factor which accounts for the number of cameras

Figure 4.7: Example of labeled sub-clusters generated for a flute player (only a few representative images per cluster are shown).

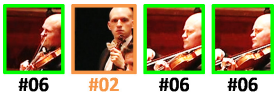
and additional variations in the types of the recorded visual material.

The values for α and $|E|$ can be chosen rather freely, as long as they are large enough. This is due to the subsequent labeling step in which all the detected clusters where musician m appears are merged together into one set S_m containing all the detections $\mathbf{d}_i^{k,l}$ of that musician, independent of the camera viewpoint, HOI activity or other variations. Therefore, while the detected clusters should be sufficiently pure, over-segmentation is not problematic. The labeling step is performed manually and is explained in Section 4.5.4.

4.5.3. GENERATING CLUSTERS OF PLAYING AND NON-PLAYING HOI

Once the set S_m is generated, we follow the hypothesis that the images contained in there can be distinguished from each other using two dominant dimensions: camera viewpoint and performed HOI action. Under this assumption, we divide each set S_m into

(a) Type A1: the dominant musician in S_6^c is #06, the dominant label is P and the image of #02 is also P.



(b) Type A2: the dominant musician in S_6^c is #06, the dominant label is P but the image of #02 is NP.

(c) Type B: the sub-cluster S_{36}^c contains only images of #36, however it is non 100% P/NP pure.

Figure 4.8: Examples of different types of error generated at the face and/or the PNP clustering steps. While the errors of type B have a direct negative impact on the accuracy, an error of type A1 or A2 leads to a P/NP labeling error depending on the timestamp of the detection belonging to the "wrong" musician.

4

sub-clusters. Each sub-cluster should contain the images of the musician m with one specific HOI action recorded from a specific camera viewpoint. This results in a set of C_m mutually disjoint subsets S_m^c such that $S_m = \cup_{c=1}^{C_m} S_m^c$. We estimate the number of sub-clusters C_m by first estimating the number of camera viewpoints $|\text{PoV}_m|$ on the musician m. Then, the number of sub-clusters corresponding to a playing or non-playing HOI is $2 \times |\text{PoV}_m|$.

The number of viewpoints on a musician m is estimated as follows. To maximize the accuracy of the clustering process at this stage, compared to Section 4.5.2, we choose for a more sophisticated method for estimating $|\text{PoV}_m|$. We do this by analyzing how the bounding box geometry $\mathbf{b}$, the head pose θ and the camera/video index i values are distributed. By empirical evaluation, we found that the number of dense regions formed by the set of $(w \times h, i)$ pairs, respectively the face bounding box area and the camera/video index of each detected face $\mathbf{d}_i^{k,l}$ belonging to m, is a suitable and consistent choice.

Then, in order to generate the sub-clusters S_m^c, we follow these steps:

1. for each $\mathbf{d}_i^{k,l} \in S_m$, we extract an image $\mathbf{I}_i^{k,l}$ from the keyframe $\mathbf{f}_i^k$
2. for each image $\mathbf{I}_i^{k,l}$, we extract a vector $\mathbf{x}_i^{k,l}$ of visual appearance features
3. we build a descriptor matrix $\mathbf{X}_m$ having $|S_m|$ rows, where each row is a feature vector $\mathbf{x}_i^{k,l}$
4. we cluster the detections in S_m by running a clustering algorithm taking $\mathbf{X}_m$ as input and with the number of clusters to be generated being set to C_m.

In order to assess the informativeness of different regions of the image, we consider two options for extracting $\mathbf{I}_i^{k,l}$, which capture the face and the upper body regions. As for the way we visually describe the segmented images, we consider global and local features. As for the latter, we aim at exploiting as much as possible the redundancy of the images belonging to each musician. We therefore train one visual word vocabulary for each set S_m instead of training a vocabulary for the whole recording. By training ad hoc vocabularies, we expect that the discriminative power of the trained visual words is optimized for each musician. In Section 4.6.1, we report the details about the used features and the optimal parameters (e.g., number of visual words).

The obtained sub-clusters directly imply the P and NP labels to be assigned to them and therefore the quality of sub-clusters also determines the quality of our P/NP annotation framework. We explain the sub-cluster annotation process in Section 4.5.4. Examples of labeled sub-clusters are shown in Figure 4.7. Unlike in the case of face clusters labeling, non-pure or otherwise ambiguous sub-clusters are not discarded, but annotated using the label X (undetermined).

This clustering step is fundamental to make the subsequent human annotation process efficient. In fact, if every single detection were manually annotated, the complexity of the human annotation task would be $O(|M_{\mathrm{GT}}| \times L)$ — i.e., linear to the number of musicians multiplied with the temporal length of the recording. Since we assumed that the number of points of view $|\mathrm{PoV}_m|$ is limited, the complexity of the human annotation task using our approach becomes $O(|M_{\mathrm{GT}}|)$ — i.e., linear to the number of musicians.

4.5.4. HUMAN ANNOTATION

Our proposed framework illustrated in Figure 4.5 involves two manual labeling steps, the first one annotating the face clusters by the corresponding musician ID and the second one annotating the sub-clusters in terms of P and NP labels.

In general, the annotation process of a cluster of images works as follows. The annotator inspects the content of a given cluster which is rendered, for instance, as a grid of images. Then, the *purity* of the given cluster is evaluated. A cluster is pure if most of the images belong to one class. We call such class *dominant*. If there is a dominant class, it is used as label for the cluster. Conversely, a non-pure cluster is discarded in order to prevent that the labeling accuracy will be low. We assume that: (i) human annotators are able to detect the presence of a dominant class, and (ii) human annotators can recognize the dominant class (if present). More details about the two manual labeling steps are reported below.

FACE CLUSTERS ANNOTATION

The annotator is provided with a reference table of musician IDs like the one in Figure 4.9. The images within a face cluster are shown to the annotator and the annotator decides first whether the cluster is pure enough, that is whether the cluster has a *dominant* musician ID.

If the annotator finds the cluster to be pure enough, then she uses the reference table to check whether the dominant identity belongs to one of the musicians. If a musician is dominant in the face cluster, then the corresponding label is chosen and automatically propagated to all the face detections belonging to the given cluster. In the cases of a non-musician dominant label (conductor, audience or non-face images) and a non-pure cluster, the cluster is discarded and the face detections belonging to it will not be used anymore.

A first type of error that can occur at this step is the error of *type A* (e.g., Figure 4.8a and Figure 4.8b): if a cluster is not discarded and therefore labeled with $m \in M_{\mathrm{GT}}$, any image not belonging to the musician m will generate a musician labeling error. The impact of this error type on the accuracy of P/NP labeling is discussed in more detail below.

Figure 4.9: Example of reference table provided to the face clusters' annotators.

P/NP CLUSTERS ANNOTATION

For this task, the annotator does not need any reference table and we expect that no specific expertise is required in order to distinguish playing and non-playing actions for any musical instrument. We also assume that each sub-cluster can be annotated independently.

Given a sub-cluster S_m^c to be labeled, the annotator once again decides first whether it is sufficiently pure. Differently from the previous annotation task, the purity now has *two* dimensions. The first one is related to the presence of a dominant P/NP class, that is whether the majority of the images show either a playing or non-playing HOI. When a dominant class is chosen, all the images not belonging to that class will generate a P/NP labeling error of *type B* (e.g., Figure 4.8c). The second purity dimension deals with the error of type A since a sub-cluster may contain images of other musicians due to errors at the face clustering phase. Considering these two aspects, we assume that a sub-cluster is discarded if it contains too many errors of type A and/or B.

Finally, regarding the error of type A, we distinguish two cases occurring when a P/NP cluster S_m^c is not discarded and contains images belonging to one or more musicians $m' \neq m$. The error of *type A1* occurs when an image of a different musician m' has the same P/NP label as the one which is dominant in the sub-cluster (e.g., Figure 4.8a). The error of *type A2* occurs when an image of a different musician m' has not the sub-cluster's dominant P/NP label (e.g., Figure 4.8b). The main impact of these types of error is that a spurious observation is added to the musician m and removed from the musician m'. Then, for the musician m, the system may generate an additional and eventually wrong P/NP label according to factors, which depend on the way P/NP sequence are generated as explained in Section 4.5.5.

4.5.5. GENERATING SEQUENCES OF P/NP LABELS

Taking the sub-clusters labeled as either P, NP or X and the keyframe's timestamps associated to the images belonging to the sub-clusters as input, we now proceed with generating the function $\text{PNP}_m(t) : \mathcal{T} \rightarrow \{\text{P}, \text{NP}, \text{X}\}$ that produces the P/NP/X label sequence for each musician $m \in M_{GT}$.

As defined in Section 4.4, we aim at reconstructing the PNP sequence for every musician at regular time intervals (e.g., every second). The reason why we do not extract the labels for every frame lies in the inherent nature of the P/NP labels. As explained in [12], it is not likely that two or more P/NP switches occur in a short period of time, because during short musical rests musicians keep a playing body pose. Hence, we adopt the same sliding window approach of [12] and we derive P/NP labels periodically for every musician. A large window size (e.g., 5 seconds) accounts for the time required to switch from a playing to a non-playing body pose (and vice versa).

For each musician m, each label is generated through a voting process illustrated in Figure 4.10. At every timestamp $t \in \mathcal{T}$, a set $\mathbf{w}$ is built by exploiting the labeled sub-clusters S_m^c associated with the musician m as follows. We look for the images $\mathbf{I}_i^{k,l} \in S_m^c$ extracted within the current sliding window time interval. This search can lead to a variable number of results, depending on how many cameras record m in the considered period of time. For each found image, one P/NP label is added to $\mathbf{w}$, inherited from the sub-cluster the image belongs to. Discarded sub-clusters are ignored. Consequently, $\mathbf{w}$ is either an empty set or contains one or more labels. In the former case, the label assigned at the timestamp t is X because there is no observation of m in the considered time window. In the latter case, P(NP) is assigned if the number of P(NP) labels in $\mathbf{w}$ is greater than the number of NP(P) labels. If the numbers of P and NP labels in $\mathbf{w}$ are equal, the label X is assigned.

4.5.6. DEALING WITH MISSING OBSERVATIONS

As pointed out in Section 4.4, there is no guarantee that each musician is always visible from at least one camera. If a musician does not appear in a keyframe, no P/NP label can be inferred using the procedure explained above. However, the domain knowledge on the orchestral setting (Section 4.1) allows us to infer the labels for individual musicians from all the other musicians playing the same instrumental part and thus belonging to the subset M_{GT}^h. In this case, for each subset M_{GT}^h, the expected sequence of labels is the same for every musician $m \in M_{\text{GT}}^h$.

We propose two different strategies to extrapolate the labels: (i) *highest timeline coverage* (highest TC), and (ii) *merging*. Given M_{GT}^h, the highest TC approach assigns one of the existing PNP functions to all other musicians in M_{GT}^h. The optimal PNP function for a given instrumental part h is that computed for the musician $m^\star$ such that $m^\star = \arg\min_{m \in M_{\text{GT}}^h} |\{t : \text{PNP}_m(t) = \text{X}\}|$. The rationale behind this strategy is to base the extrapolation on the musician for which the number of observations is maximized. Differently, the merging strategy computes a new PNP function for each instrumental part by combining all the labeled sub-clusters S_m^c belonging to the musicians performing the considered instrumental part. As opposed to relying on the strongest evidence as in the previous strategy, here we combine all the available evidence belonging to a certain instrumental part. For this purpose, we deploy a modified version of the majority vot-

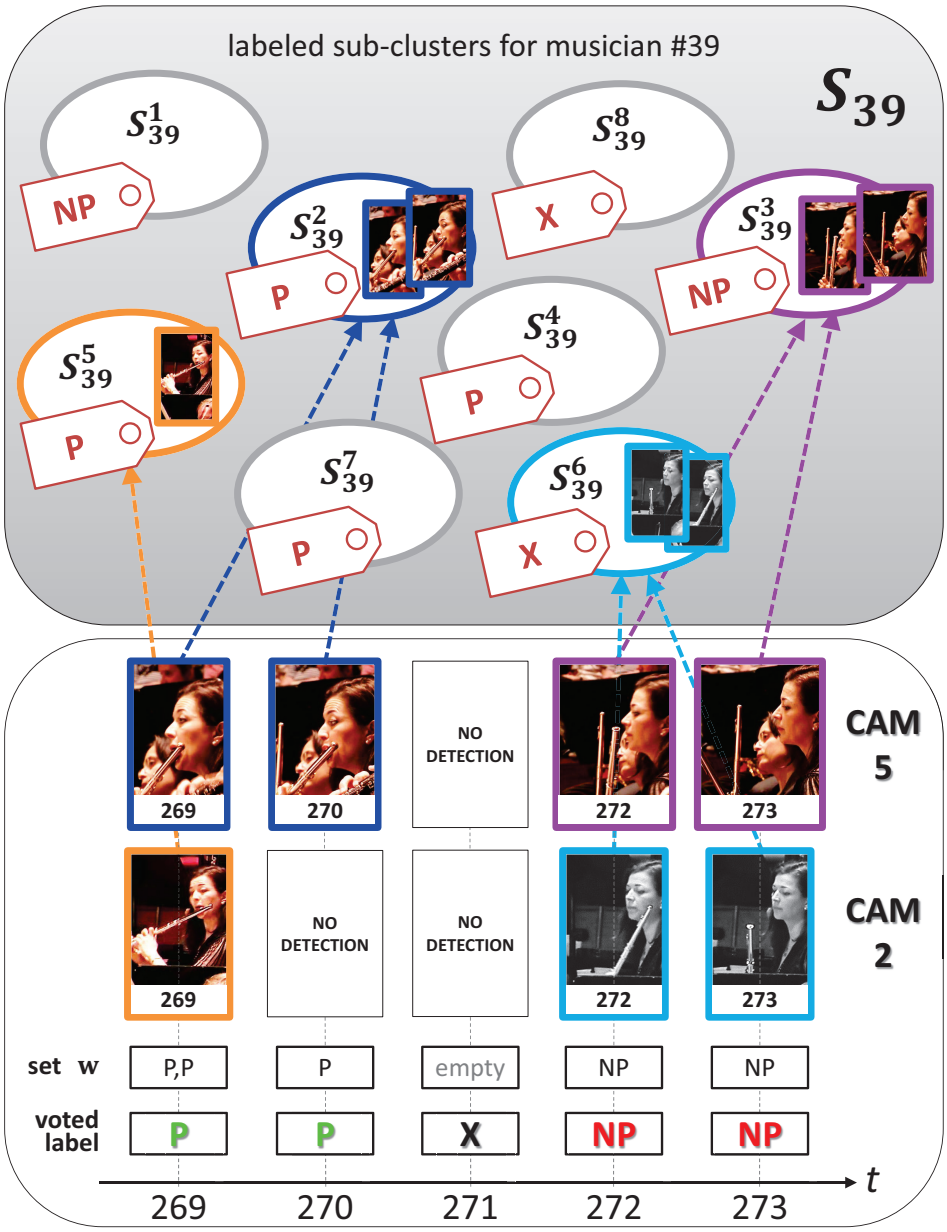
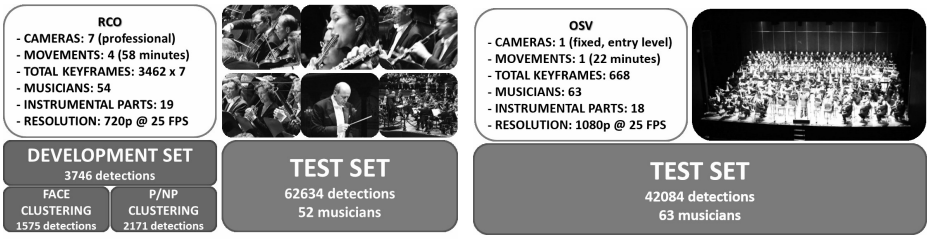


Figure 4.10: Illustration of the process of creating the P/NP label sequences for a musician m via majority voting. In this example, we focus on the case of two cameras recording the musician $m = 39$ and we set the sliding window size to 1 second (for the sake of simplicity). Given 8 labeled sub-clusters S^c_{39}, every second the available P/NP labels are sought in the labeled sub-clusters. The retrieved labels are used to build the sets **w** to which a majority voting is applied to determine the final label.



(a) Properties and organization of the RCO dataset.

(b) Properties and organization of the OSV dataset.

Figure 4.11: Proposed datasets used in this work.

ing approach described in Section 4.5.5. When **w** is populated, instead of considering the sub-clusters S_m^c of a single musician, we consider *all* the sub-clusters S_m^c such that $m \in M_{\text{GT}}^h$.

4.6. EXPERIMENTAL SETUP

In this section we detail how we implemented the proposed framework, present our dataset, and explain how we conducted the experimental evaluation.

4.6.1. FRAMEWORK IMPLEMENTATION

The design choices and the parameter selection underlying the realization of our framework (presented in Section 4.5) were informed following the protocol described in Section 4.6.1 and Sec.4.6.1.

MUSICIANS DIARIZATION

We describe the way we implemented the four face clustering methods introduced in Section 4.5.2 and explain how we selected features and parameters.

The *B-cubed precision/recall* [2] was adopted to assess the quality of the produced clusters. We chose the number of face clusters by approximating the number of entities $|E|$ to the number of musicians. In the case of the development set, $|E|$ was set to 7. A suitable value for factor α taking into account the variations of various types was found by inspecting multiple options, namely 1, 1.5, 2, 2.5, 3, 4, 5, 10, 15 and 20 (generating from 7 to 140 face clusters).

For clustering itself, we used k-means in the unconstrained case and COP k-means [100] in the constrained one. The constrained face clustering methods were not assessed using the development set because COP k-means has no parameters to be tuned and the number of cannot-link constraints generated for the development set was too low.

As for the unconstrained face clustering, we considered two options, both relying on state-of-the-art visual features. In the first one, we deployed Local Binary Patterns (LBP) and Histograms of Oriented Gradients (HOG) from the face bounding box as done in [108]. In the second one, we focused on the face bounding box extended to the hair region from which we extracted Pyramid HOG (PHOG), Joint Composite Descriptor (JCD), Gabor texture (Gabor), Edge Histogram (EdgeHist) and Auto Color Correlogram (ACC)

[61]. In both cases, we evaluated the impact of applying the Principal Component Analysis (PCA) [49] retaining 99% of the total variance.

In the context-assisted case, we included a description of the scene and/or a description of the upper body region. As for the former, given a detection $\mathbf{d}_i^{k,l}$, we extracted the JCD, PHOG and ACC global features from a downsampled copy of the keyframe $\mathbf{f}_i^k$. The upper body region was always described by LBP, PHOG, JCD, Gabor, EdgeHist and ACC. For both scene and upper body descriptors, we assessed the impact of including and excluding this information and also the option of including it by first applying PCA retaining a number of possible ratios of total variance (namely, 50%, 70% and 99%).

By inspecting the results summarized in Figure 4.12, we found that the optimal set of features to assess the face similarity is that extracted from the face-hair region and consisting of PHOG, JCD, Gabor, EdgeHist and ACC applying the PCA (see Figure 4.12a). By comparing the plots in Figure 4.12, we see how different combinations of contextual features affect the performance. The upper body features leads to the strongest improvement and the optimal ratio of retained variance for the PCA is 99% (see Figure 4.12c). The scene features, whose optimal ratio of retained variance for the PCA is 70%, do not add a significant contribution (see Figure 4.12b and Figure 4.12d). Finally, the optimal value of α we chose was 15 because by increasing it to 20 we observe a saturation in the performance.

4

P/NP CLUSTERING

For each set S_m of images belonging to one musician, we estimated the number of points of view (see Section 4.5.3) as follows. The list of $(w \times h, i)$ pairs derived from S_m was first normalized (zero mean, unit variance). Then, we used DBSCAN [31] to automatically estimate the number of formed dense regions. We required that a dense region had at least 10 samples and the dense region radius parameter ϵ was set to 0.4. Pairs not belonging to any dense region were ignored.

As discussed in Section 4.5.3, the P/NP clusters S_m^c were produced considering two possible image regions and two possible types of feature. Evaluating on the dedicated development set, we found the following optimized global feature sets: face images were best described using Gabor, JCD and PHOG without applying the PCA, while upper body images by EdgeHist, Gabor, PHOG and ACC retaining 95% of the total variance. As for the local features, we considered two possible options, namely SIFT and OpponentSIFT [79], aggregating them either via bag-of-words (BoW) [23] or via spatial pyramid (SP) [56]. We also evaluated different visual words vocabulary sizes, namely 200, 400 and 1000 visual words (1000 only used with BoW). For each musician, that is for each set S_m, the visual word vectors were assigned via mini-batch k-means [82] applied to the visual words vocabulary training set, built by randomly sampling 500.000 feature vectors from the images in S_m. Using the development set, we found that the optimal way of describing both face and upper body images was using OpponentSIFT with 200 visual words, but aggregating the former via SP and the latter via BoW.

Image clustering was performed using the k-means algorithm. In order to assess the significance of the obtained results, we also included a random baseline method which simply works by randomly assigning the images in a given set S_m to the sub-clusters S_m^c.

4.6.2. SIMULATING THE HUMAN ANNOTATION

In this work we address a number of research questions for which the experiment has to be repeated several times using different (types of) features and parameters. This is particularly true for the research question RQ3, for which we want to assess the overall impact of errors in different modules. In this context, deploying the two human annotation task presented in Section 4.5.4 for every run is not feasible. In fact, in the full experiment we generate dozens of thousands of image clusters to be annotated. Another reason for not performing human annotation at this stage is that we do not know yet how to instruct human annotators with respect to how tolerant or strict they should be when coming across non-pure image clusters. We therefore made a number of assumptions and simulated human annotation using the available ground-truth information, also quantifying the perceived purity of a cluster of images and assessing the impact of different levels of strictness.

MODELING THE HUMAN ANNOTATOR

Following the annotation process and the assumptions reported in Section 4.5.4, we modeled a human annotator as follows. The core idea is to define a *rejection threshold* with which a cluster is discarded if the frequency of the dominant class is below such threshold. For each cluster, we compute a histogram of frequencies having one bin per class. If the highest frequency is below the rejection threshold, the cluster is discarded, otherwise it is kept and labeled with the dominant label. In our experiments, we used a number of distinct threshold values in order to study the impact on the overall performance. A high threshold corresponds to a *strict* annotator (high precision), while a lower value is a more *tolerant* one (balanced precision and recall).

SIMULATING THE FACE CLUSTERS ANNOTATION

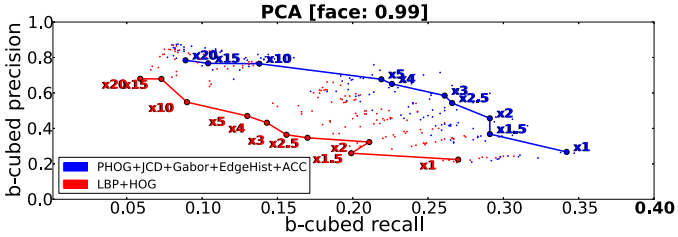
When labeling face clusters, we assigned the histogram bins as follows: one for each musician $m \in M_{\text{GT}}$, one for the conductor, one for the audience, and one for false positive face detections. We considered three types of human annotators by using the values 50%, 70% and 90% for the rejection threshold. When the voted label did not belong to a musician, the face cluster was discarded. In order to understand to what extent face clustering is a critical step, we also used the face clustering ground-truth labels (*ideal* case).

SIMULATING THE P/NP CLUSTERS ANNOTATION

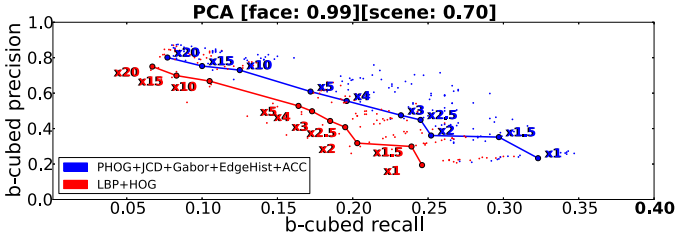
When labeling a sub-cluster S_m^c, we computed the histograms assigning three bins associated to playing, non-playing and outlier images. The latter was used when an image of a different musician occurred, that is when an image belonged to a musician $m\prime \neq m$. We tested the following rejection thresholds: 50%, 60%, 70%, 80% and 90%.

4.6.3. DATASET

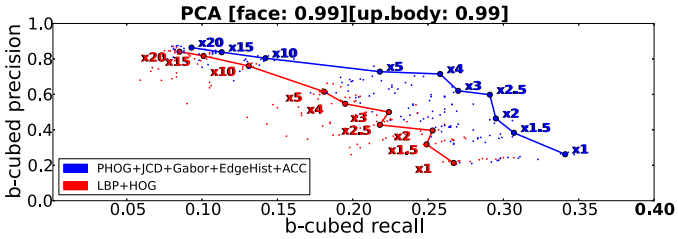
We experimented on a dataset which in total consists of 29 videos (about 7 hours) from which we extracted more than 100,000 detections belonging to 105 different musicians. The dataset was built based on video recordings of two symphonic music concerts performed by two different professional orchestras and is representative for the context in



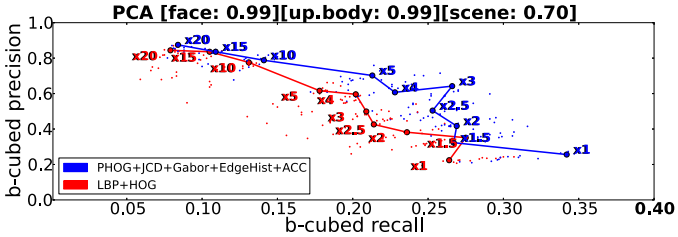
(a) unconstrained



(b) context-assisted (scene only)



(c) context-assisted (upper body only)



(d) context-assisted (scene and upper body)

Figure 4.12: Face clustering evaluation on the dedicated development set. Each dot represented an evaluated combination of types of feature, amounts of retained variance while applying the PCA and factors affecting the number of generated face clusters. Each line corresponds to the selected combination and shows how the performance change when the number of generated cluster varies (e.g., 10x means 10 times the number of musicians in the development set, namely 10×7). The plots show that the richest feature set (namely, PHOG, JCD, Gabor, EdgeHist, and ACC) consistently performs better than the smaller set (namely, LBP and HOG).

which we operate, as described in Section 4.1. The first recording contains the four movements of Beethoven's Symphony No. 3 Op. 55, performed by the Royal Concertgebouw Orchestra (Amsterdam, The Netherlands) and it is a multiple-camera recording. The second one is a fixed, single-camera recording of the fourth movement of Beethoven's Ninth Symphony performed by the Simfònica del Vallès Orchestra (Barcelona, Spain). The two recordings, respectively referred to as "RCO" and "OSV", are available on request. To the best of our knowledge, there is no other available dataset consisting of real world data that we could have used alternatively.

RCO DATASET

The RCO dataset (Figure 4.11a) is organized into 4 sets of 7 synchronized videos where each set represents the multiple-camera recording of a movement (6 hours and 40 minutes in total). The number of performing musicians is 54 and they are organized into 19 instrumental parts and playing 11 different instruments. The recording also captures the audience and the conductor. From each video, we extracted 1 keyframe every second producing 24,234 keyframes in total.

For each keyframe we detected the faces and estimated the head poses. This was done by combining a number of off-the-shelf multi-pose face detectors [113, 97] via non-maximum suppression (NMS). The way we estimated the head pose is an adaptation of the method described in [7]. The adaptation was required in order to integrate the detector from [113] for which we initialized the confidence of its output to the acceptance threshold level (see [7]) in order to maximize the face detection recall. The choice of combining different types of detectors has significantly increased the number of detected faces. Overall, 66,380 face have been found which are distributed as follows: 1,716 belonging to the conductor, 4,539 to the audience, 3,844 are false positives and the remaining 56,281 are distributed across the 54 musicians.

OSV DATASET

The OSV dataset is a fixed, single-camera recording in which the performers appear at the same position throughout the whole event (see Figure 4.11b). Faces approximately cover an area of 20×20 pixels, much smaller compared to those of the RCO dataset. The positions of the faces were manually annotated using a random frame as reference and then the head poses were, again manually, assigned to every face. Therefore, the face clustering step is not necessary for this recording since the musician identity is only a function of the face bounding box position in the reference keyframe. In this case, we extracted a keyframe every 2 seconds because, being the recording a fixed-camera one, oversampling in time would have been unnecessary for the goals of our experiment.

DEVELOPMENT SET

As shown in Figure 4.11a, part of the data extracted from the RCO dataset was used as development set. The reason why we did not include data from the OSV dataset there is twofold. First, we wanted to assess the general applicability of our method to an unseen recording. Hence, we followed a leave-one-recording-out approach while searching for visual features and parameters. Second, we find the RCO concert a more general case than the OSV due to the additional variations caused by panning and zoom-in camera actions.

The face clustering development set was generated by randomly sampling 1,575 face detections belonging to the conductor, audience, 7 musicians performing different instrumental parts and also belonging to the false detections.

The development set was used to inform the design choices and select parameters of our framework. All the remaining data was used at the evaluation step.

4.6.4. GROUND TRUTH

The ground truth for evaluating the face clustering method was created by the authors, by annotating the 66,380 faces detected in the RCO dataset. The true P/NP labels were derived using synchronized symbolic information. As for the RCO dataset, we used four MIDI files synchronized to the video files provided by Grachten et al. [44], from which we extracted the P/NP labels with the method described in [12]. The Music Technology Group (Pompeu Fabra University, Spain) provided us with the video recording and a set of files encoding synchronized note onsets and offsets for each instrumental part. In both cases, each performing musician was bound to the corresponding instrumental part / MIDI track in order to build the corresponding ground truth P/NP sequence.

4.6.5. EVALUATION APPROACH

The goal of the experimental evaluation in this chapter was threefold. First, we assessed the performance of the key-modules of our framework, including P/NP labeling (Section 4.7.2) as well as face labeling — i.e., musician diarization (Section 4.7.1). The quality of P/NP label sequences is the key result serving to demonstrate the effectiveness of our proposed method. However, we also evaluated the face labeling step to understand how inevitable errors there affect the quality of P/NP label sequences.

Second, as reported in Section 4.7, we assessed the quality of the obtained P/NP label sequences also relatively, using a random baseline as a reference. Relying on a random baseline was the only possible choice here, and this for the following reasons. The related literature does not offer a solution for yielding one sequence of P/NP labels for each performing musician. In fact, as discussed in Section 4.2, existing audio-based and visual-based classifiers cannot be directly applied to the type of audiovisual content considered in this chapter. Replacing the semi-automatic framework modules described in Section 4.5.3 and Section 4.5.4 is only theoretically possible. As explained in Section 4.2.1, existing vision-based classifiers require input of a particular type and are instrument-dependent.

Finally, in Section 4.7.6, we compared the efficiency of P/NP labeling using our method with the efficiency of the purely manual P/NP labeling in order to determine how much human annotation can be speeded up, while maintaining the same high quality of the P/NP label sequences.

4.6.6. EVALUATION MEASURES

In this section, we describe the evaluation measures used to assess the quality of the labels produced after the human annotation steps described in Section 4.5.4.

Once the face clusters had been generated and labeled, we jointly evaluated precision, recall and number of labeled (or non-discarded) face detections. The average precision and the average recall were combined together into the average F1-score. The

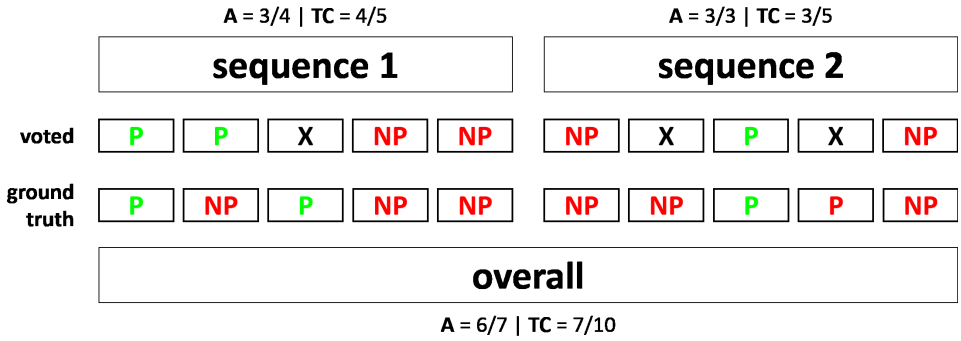


Figure 4.13: Example of how A and TC scores are computed for P/NP label sequence assessment.

percentage of non-discarded face detections was simply determined by counting how many images inherit a label from non-discarded face clusters.

For each musician the system produces a sequence of P/NP/X labels to be compared to the corresponding ground truth sequence. As illustrated in Figure 4.13, we evaluated the labeling performance integrally aggregating the results obtained for all the musicians. The performance with respect to the grond truth was assessed using two scores: *accuracy* (A) and *timeline coverage* (TC). The former is defined as the percentage of matching labels and it is computed only considering the known labels, namely those for which the value is different from X. The TC score is defined as the ratio between the number of non X-valued labels and the ground truth sequence length. It is an indicator of how many detections are used by the system and its upper bound is defined by the percentage of available musician detections.

We recommend to use accuracy instead of other scores, like precision and recall, because we need to assess how well the system produces both playing and non-playing labels. The timeline coverage was chosen to observe how many labels are effectively generated by the system, but also to measure the impact of rejecting non-pure image clusters.

4.7. RESULTS

This section reports the results and provides the reader with the answers to the research questions defined at the beginning of this chapter. First, we addressed RQ1 in Section 4.7.1, where we evaluated different options to solve the musician diarization problem. Then, in Section 4.7.2 we focused on the P/NP labeling problem addressing RQ2 and RQ3. We added a failure analysis section (Section 4.7.3) in which we explained how the system fails. This provides insights about the informativeness of static images (RQ4). The results obtained when adopting the two proposed strategies dealing with missing observations are reported in Section 4.7.4. Then, we qualitatively compared the ground truth and the generated P/NP sequences using the OSV dataset (Section 4.7.5). Finally, we answered RQ5 by measuring the achieved efficiency and effectiveness of the human annotation tasks (Section 4.7.6).

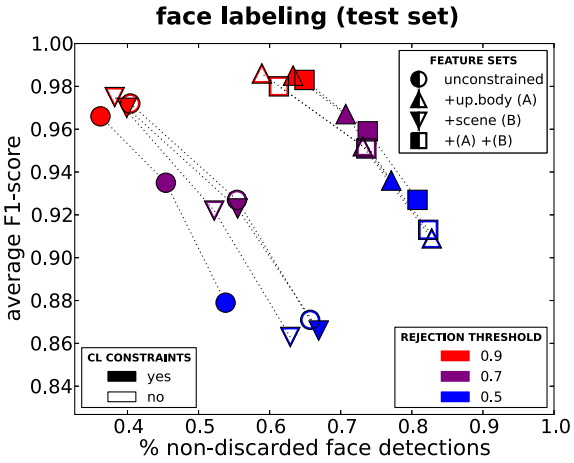


Figure 4.14: We compared four feature sets (represented by different markers) using either the constrained clustering (filled markers) or the unconstrained one (empty markers). We also evaluated three different rejection thresholds (different colors). The plot shows three results. First, combining face and upper body visual information produces the best results. Second, adding scene visual information and/or using the cannot-link constraints does not significantly improve. Third, a higher rejection threshold effectively filters out non-pure clusters.

4.7.1. FACE LABELING

We evaluated the proposed semi-automatic method producing face labels on the RCO test set. This set consists of 64,805 detections belonging to 54 musicians. With these detections we generated 191,745 cannot-link constraints (see Section 4.5.2).

Figure 4.14 shows that the most informative regions are the face and the upper body. Including scene information does not significantly improve the performance and the same holds for the cannot-link constraints. While including scene information did not impact the computation time, running the constrained version of k-means led to a significantly longer execution time. In general, we see that our method generates face labels with high average precision and recall. However, this result was not obtained just via the face clustering step but also using the human annotators' ability to discarding non-pure clusters. In fact, in the best case we already observe that about 20% of the detections fell into discarded face clusters. This means that a part of the clusters was not sufficiently pure.

4.7.2. P/NP LABELING

This section analyzes and compares the results obtained for the RCO and the OSV datasets, addressing the research questions RQ2 and RQ3.

The plots reported in Figure 4.15 and Figure 4.17 show the accuracy and the timeline coverage for the different types of features and regions of the image also including the results obtained with the random baseline method. As for the adopted notation, each point corresponds to the combination of an image region (upper body vs face), of type of features (global vs local) and of rejection threshold used when labeling the P/NP clusters

(50%, 60%, 70%, 80% and 90%). A dedicated marker is used for the random baseline method.

EVALUATION ON THE RCO DATASET

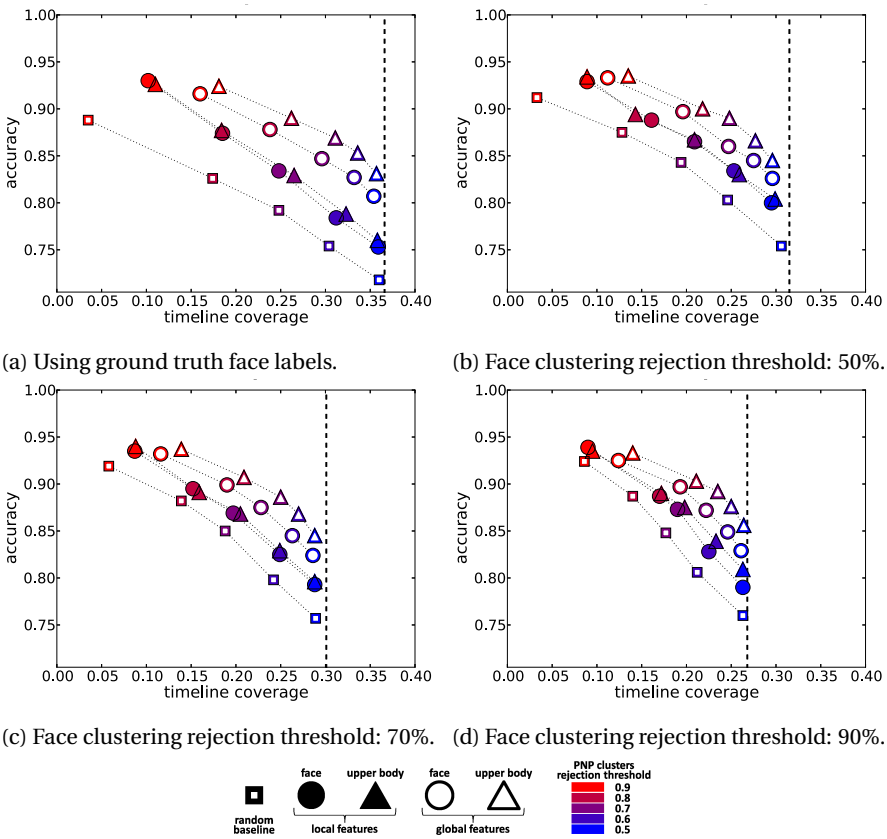
The RCO dataset allowed us to assess the full system that is, we could observe how different ways of generating the face labels affected the performance at the P/NP labeling step. To this end, we evaluated four cases. First, we considered the case of *ideal* input, in which the ground truth face labels were used. Then, we considered three different ways of obtaining the face labels by varying the rejection threshold used to label the generated face clusters. More specifically, we used the RCO test data, which includes the detections of 52 musicians. Setting α to 15 and approximating the entities set size $|E|$ to the number of musicians generated 780 face clusters. Then we simulated the annotation using three different rejection thresholds: 50% (tolerant annotator), 70% and 90% (strict annotator). In this experiment we used the unconstrained context-assisted face clustering method — i.e., we exploited face similarity and context information extracted from the upper body and the scene (see Section 4.5.2). The overall numbers of generated P/NP clusters were 530, 384, 354 and 342 for the face labels input of the types "ideal", 0.5, 0.7 and 0.9 respectively. In Figure 4.15, which summarizes the results, we observe four facts.

First, regardless of the input to the P/NP clustering step, there is a consistent trade-off between accuracy and timeline coverage. The stricter the annotator is (higher P/NP rejection threshold), the lower the number of produced P/NP labels is. More in detail, the figures show that the timeline coverage decrease is much larger than the accuracy increase. This means that quite often the purity of the produced P/NP clusters is below the highest rejection thresholds. In Section 4.7.3 we investigate the reasons why the P/NP clusters are not always pure enough.

Second, global features always outperform local ones and the upper body region is more informative than the face region. What is surprising is that faces are already a good indicator to infer P/NP labels. The advantage of this image region over the upper body one is that occlusions here seldom occurs. When the instrument or the human body parts are not visible, face cues can be always exploited. To show this, we give an example in Figure 4.16. A relaxed, unfocused, or contemplative expression (Figure 4.16a, Figure 4.16b, Figure 4.16c) is likely to be linked to a non-playing action, as opposed to a concentrated one (Figure 4.16d) that is likely to indicate a playing activity.

Third, when the rejection threshold for the sub-cluster annotation is set to 50%, the timeline coverage in Figure 4.15 is always close to its upper boundary (the markers in the four plots are close to the vertical dashed lines). Such boundary is determined by the available face detections and it shows the highest possible timeline coverage. This result was expected because, by setting the rejection threshold to 50% and having only two possible labels (P and NP), no cluster is discarded. Still, a number of additional X labels can be generated by the process explained in Section 4.5.5 due to conflicting cluster labels in case of multiple views on the same musician. However, the plots reveal that this seldom happens.

Finally, by setting again the rejection threshold to 50%, we also observe that the accuracy is always above 75%. This happens because the numbers of P and NP labels in the ground truth are not equal. For this reason, in order to assess whether the proposed



(a) Using ground truth face labels.

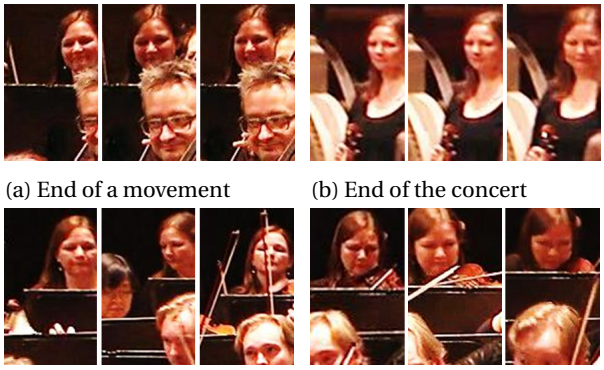
(b) Face clustering rejection threshold: 50%.

(c) Face clustering rejection threshold: 70%.

(d) Face clustering rejection threshold: 90%.

(e) Legend

Figure 4.15: Evaluation of the P/NP labels produced by the system (RCO dataset). The vertical dashed lines show the upper bound for the timeline coverage, which is limited by the availability of face detections. The upper body region described with global features outperforms other combinations. Tuning the system for very high accuracy has a large negative impact on the timeline coverage. This shows that discriminating playing and non-playing HOIs requires information beyond a global description of a static upper body image.



(a) End of a movement (b) End of the concert

(c) Non-playing (d) Playing

Figure 4.16: Informativeness of the face region: even when the torso region is not visible we can guess whether a musician is playing by analyzing the face expression.

method is generating P/NP clusters at all, the random baseline method is included. What we see is that the baseline always performs worse, both in terms of timeline coverage and accuracy. This shows that our method effectively discriminates playing and non-playing actions.

EVALUATION ON THE OSV DATASET

In the OSV dataset, 63 musicians are recorded by a fixed camera. Compared to RCO, there is no point-of-view variability and all the musicians are always visible. The number of P/NP clusters is 126. For this recording we only evaluated local and global features extracted from the upper body region. We made this choice because, as explained in Section 4.6.3, the face region in the OSV dataset is too small. Even we could not use this recording to evaluate the full proposed system, it is an additional test case to also assess whether and to what extent other recordings and recordings of a different type can be exploited for P/NP detection.

The OSV images are challenging because they have a low resolution. However, the system is still able to well discriminate P/NP actions as shown by the results in Figure 4.17. This becomes evident by comparing the random baseline performance with that of our image clustering methods. Increasing the rejection threshold from 50% up to 80%, we see that the number of discarded images decreases linearly at relatively small steps. This means that the majority of the generated P/NP clusters were pure enough. However, when we look at the strictest rejection threshold, we observe that the accuracy increase is small while the number of determined P/NP labels decreases at a much higher rate. Therefore, as we did for the RCO dataset, we conclude that there are additional factors determining the playing/non-playing status of musicians which are not taken into account in our solution.

OVERALL JUDGMENT

By evaluating on the RCO and the OSV datasets, we answered to the research questions RQ2 and RQ3.

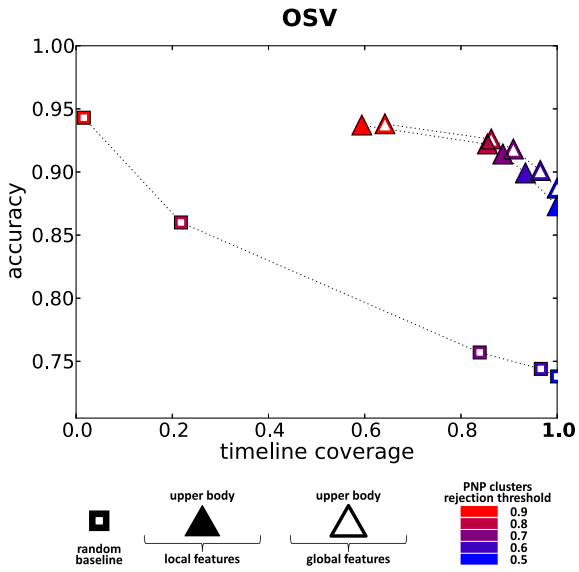


Figure 4.17: Results for the OSV dataset. Even if the video resolution is low, with a fixed camera we accumulate a sufficient number of images for each musician from the same point of view. Due to this, playing and non-playing images can be discriminated with high accuracy.

We conclude that the most P/NP discriminative region is the upper body. However, we remark that faces by themselves are already surprisingly informative. Regarding the accuracy of the system, we see that it ranges between 70% and 94% depending on the strictness of the annotators. However, targeting to a high accuracy has a significant impact on the number of discarded detections especially in the case of a multiple-camera recording in which it is hard to continuously accumulate observations over time for each performing musician.

As for the impact of different modules, we have two conclusions. First, we see that the overall timeline coverage is directly affected by the number of available face detections. This indicates that the face detectors should be tuned to perform with high recall in order to determine as many P/NP labels as possible for each musician. Second, we observe that the musician diarization module has a limited impact on the overall accuracy because most of the face clusters are sufficiently pure.

4.7.3. FAILURE ANALYSIS

As pointed out in Section 4.7.2, a fraction of the produced sub-clusters S_m^c is not sufficiently pure. By inspecting the produced P/NP clusters, we found that subtle discriminative cues in the images sometimes occur. For instance, in Figure 4.18, we see that the mouth region for the French horn player is the discriminative region. However, our method has not been designed to explicitly take into account this part of the image, therefore the images are clustered according to the overall appearance of the upper body.

The aforementioned error belongs to a larger class of errors, namely the false posi-

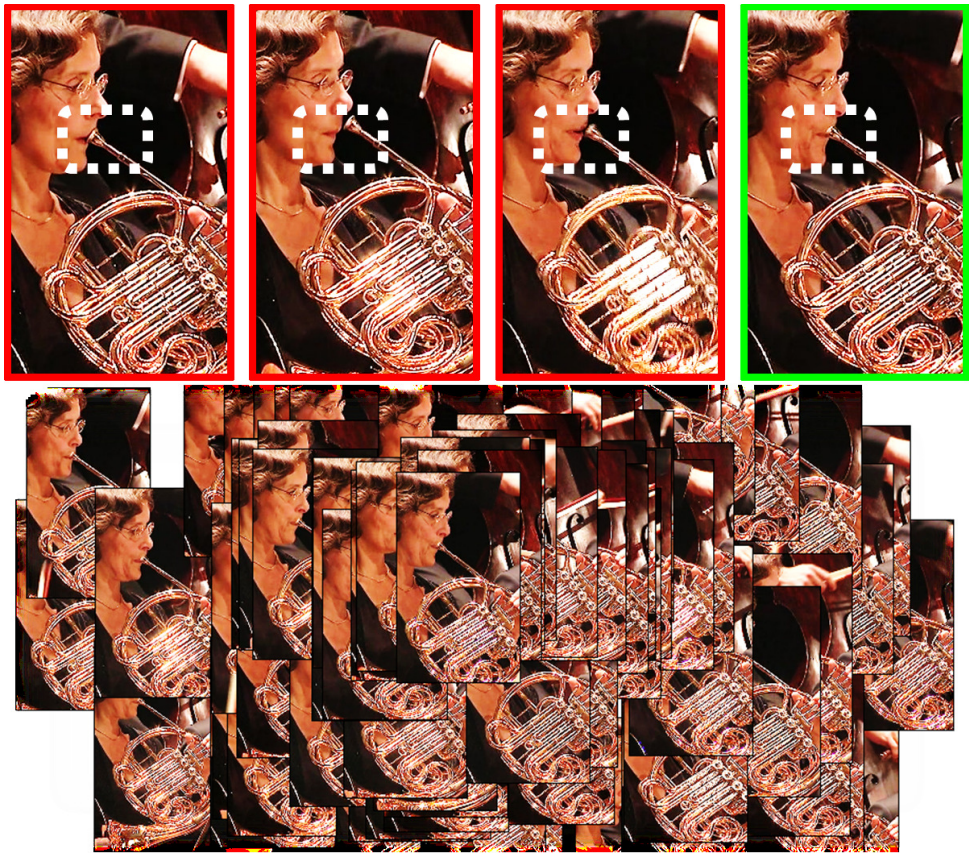


Figure 4.18: Good P cluster containing some NP images, which are included because the differences in the mouth region are not dominant, sufficiently influencing the cluster formation.

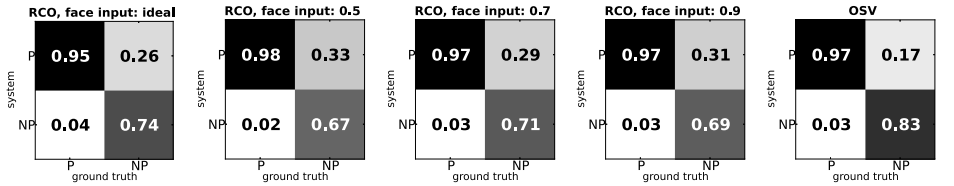


Figure 4.19: The depicted confusion matrices show that the system has a bias towards false positives. Such a bias can be explained by the fact that the musicians usually get ready to play sufficiently in advance.

tives. By inspecting the videos, we observed that they occur for any type of instrument and that they are caused by *anticipation*, which occurs when a musician gets ready to play in advance. This is also supported by the confusion matrices in Figure 4.19. They all show that the amount of false positives (false P labels) is greater than the amount of false negatives (false NP labels). Even if the P/NP ground truth has been generated taking into account anticipation [12], the results reported in Figure 4.19 let us believe that it starts much earlier than expected.

Due to the aforementioned observations, we answer to RQ4 as follows. On the one hand, a more detailed analysis of the images can be performed (e.g., including features extracted by the mouth region) thanks to which a static image could be enough for P/NP labeling. On the other hand, we cannot exclude that an image itself is partially informative. For instance, we expect that musicians' movements could be informative as well. Additionally, timbral features from the audio recording can be used in a multimodal fashion.

4.7.4. EVALUATING THE STRATEGIES FOR MISSING DETECTIONS

In Section 4.5.6 we proposed two ways of dealing with the limited availability of observations (namely, highest TC and merging). We evaluated the two strategies by considering the case of ideal face clustering input, using global features extracted from the upper body region and by setting the P/NP clustering rejection threshold to 80%. The results summarized in Table 4.1 show that both strategies are beneficial. In fact, when nothing is done (standard case), the timeline coverage is always the lowest.

In the highest TC case, the result is a direct consequence of using the labels from the most visible musician. While in the merging strategy, the advantage comes from the availability of multiple P/NP labels obtained by exploiting the musician redundancy within each instrumental part. Due to this redundancy, the voted labels can be inferred with more confidence at the majority voting step (see Section 4.5.5). Overall, the most effective strategy is merging.

4.7.5. QUALITATIVE ASSESSMENT

We also qualitatively assessed the P/NP labeling performance generating a PNP matrix. This matrix shows all the P/NP sequences produced for different instrumental parts. In Figure 4.20 compares the ground truth matrix and the one generated for the OSV dataset. The latter is generated using global features extracted from the upper body region and

strategy	standard		highest TC		merging	
score	A	TC	A	TC	A	TC
RCO	0.890	0.262	0.884	0.369	0.884	**0.429**
OSV	0.926	0.863	0.927	0.867	0.927	**0.873**

Table 4.1: Comparing the standard method with two possible strategies dealing with missing observations. The scores are computed considering the ground truth face labels, global features extracted from the upper body region and adopting 80% as rejection threshold for the PNP clusters. The merging strategy significantly improves the performance in the RCO case, while it has limited benefit in the OSV case. This is expected since the latter is a fixed camera recording and every musician is always visible.

by setting the P/NP clustering rejection threshold to 80%.

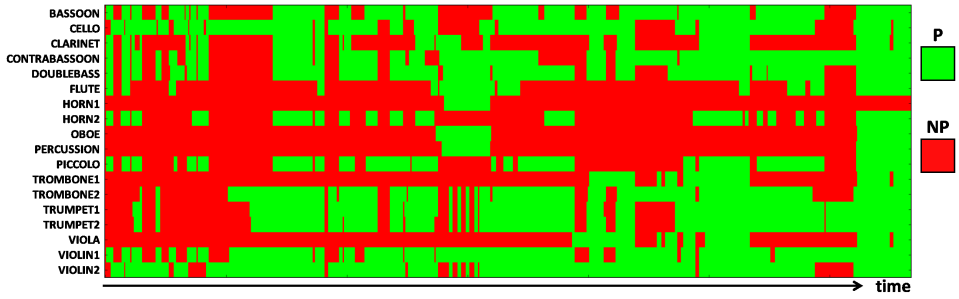
From this example, we observe that the dominant error is indeed caused by the false positives and that for some instrumental parts a significant number of labels are missing (in particular for the clarinet and the horn).

4.7.6. HUMAN ANNOTATION EFFICIENCY

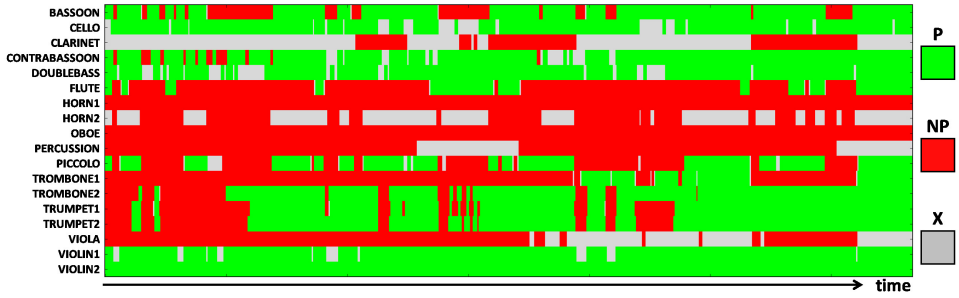
We addressed the last research question (RQ5) by assessing the balance between the efficiency and effectiveness of the human annotation required by our system. We evaluated whether the system generates a close-to-optimal number of P/NP clusters and measured the ratio between the amount of required human annotations and the number of generated P/NP labels. As for the notation used here, we refer to Section 4.5.3.

First, we assessed whether our method produces too many or too few sub-clusters S_m^c. Figure 4.21a and Figure 4.21b report the results for the RCO and the OSV dataset respectively. The plots show how the timeline coverage (TC) and the accuracy (A) change by varying the number of generated P/NP clusters. The results were obtained using the P/NP clustering based on global features extracted from the upper body region. To show the significance of the results, we also evaluated the random baseline's performance. In both cases, the ground truth face labels are used and the P/NP clustering rejection threshold is set to 80%. For each musician we estimated the number of points of view and we considered twice as many sub-clusters (as explained in Section 4.5.3). Then, we used an additional factor β applied to increase (or decrease) the number of sub-clusters per musician. For instance, when $\beta = 5$, the number of sub-clusters is ten times the number of the estimated points of view. When $\beta = 0.5$ the number of sub-clusters is exactly the number of points of view. In summary, the value set for β affects the overall number of sub-clusters $\sum_{m=1}^{|M_{GT}|} C_m$.

In Figure 4.21a, we see that on the left of $\beta = 1$, the performance quickly decreases. By contrast, on its right side the timeline coverage slowly increases. This pattern is even more evident for the OSV concert (Figure 4.21b). In this case there is a sharp transition from the case in which there is only one sub-cluster per musician (namely when $\beta \in \{0.25, 0.50\}$) and a saturation of the performance for values of β bigger than the unity. Both results show that the way the system chooses the number of sub-clusters is optimal to avoid unnecessary over-segmentation. Adding too many clusters would lead to extra manual annotation but with little advantage in terms of accuracy and timeline coverage. Similarly, we see that the system generates the critical number of sub-clusters which are



(a) Ground truth P/NP matrix.



(b) P/NP matrix via P/NP clustering

Figure 4.20: Comparing the P/NP matrices for the OSV performance. The merging strategy has been applied. Therefore both matrices have one row per instrumental part.

necessary to avoid that P and NP images consistently fall together into one cluster.

Finally, we computed the ratios between the overall number of detections and the number of produced sub-clusters. The former is defined as $\sum_{m=1}^{|M_{GT}|} |S_m|$, while the latter is defined as $\sum_{m=1}^{|M_{GT}|} C_m$. For the RCO dataset, the ratio is equal to 52204/530 = 98.5 and for the OSV dataset 42084/126 = 334. This means that on average one human label is propagated to about 100 detections in the RCO dataset and more than 300 in the OSV one.

4.8. DISCUSSION

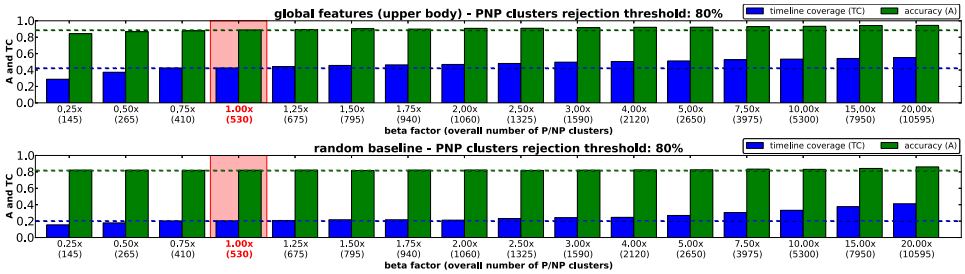
In this final section, we report the limitations we have encountered while deploying a number of state-of-the-art methods hence suggesting possible research directions for the future.

The face detection step is critical for our system since it directly affects the timeline coverage. We found that off-the-shelf detectors are optimized to achieve high precision and that the recall is not satisfying like evident, for instance, from the example of Figure 4.6 in which approximately only one third of the musicians is detected. Our attempt to overcome this problem by combining multiple heterogeneous detectors helped, but it may be useful to investigate more how to improve the face detection recall in videos.

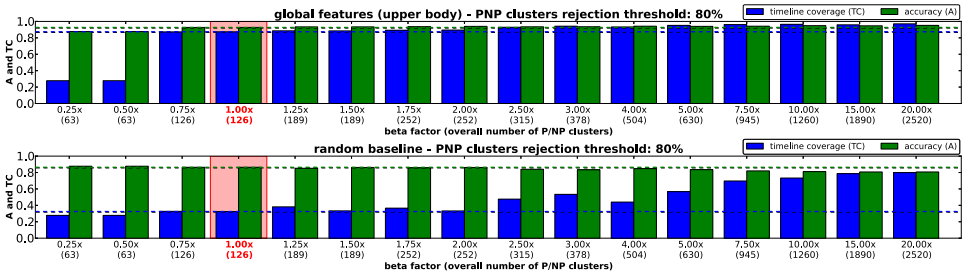
When clustering the faces, it is important to limit the number of generated clusters in order to reduce the amount of human annotation. State-of-the-art face clustering solutions designed to limit the number of produced clusters are available. However, they work assuming that the initial clusters are nearly 100% pure. What we found in our experiments is that this does not always hold. More specifically, we observed either very pure face clusters or fuzzy ones and that the latter usually contain images with lower resolution and/or profile faces. In order to maximize the utility of each detected face, and once again avoid negative impact on the timeline coverage, face clustering methods should be improved so that non-pure clusters are detected and discarded or treated with alternative strategies.

Furthermore, a more detailed analysis of the image segmentation process is needed. The idea of exploiting the head pose to determine the upper body region of a musician seems to be effective. We evince this by inspecting the results obtained at the P/NP clustering step when upper body images are clustered — i.e., empirical evaluation of the segmentation process. However, it may be the case that the optimal size of the upper body bounding box changes for different types of instruments. Hence, a more detailed analysis of the segmentation process should be carried out, eventually measuring the performance in analytical fashion rather than an empirical one.

Finally, by investigating the limitations of our approach, we learned that there are cases in which a non-playing image is very similar to a playing one due to the anticipation before the actual note onsets. What we have observed shows that the playing/non-playing information is not simply encoded in the spatial configuration between the musical instruments and the body parts as assumed by state-of-the-art methods. Additional information has to be extracted by, for instance, exploiting the richness in the face region, the musicians' movements and/or auditory features. A second issue to be considered is how to possibly label the discarded images. For instance, using the non-



(a) RCO dataset (face ground truth labels used). By increasing the number of generated P/NP clusters ($\beta > 1$) both the A and the TC scores slightly increase (saturation of the performance). By contrast, when $\beta < 1$ our method generates much less pure P/NP sub-clusters. This is an indicator that a suitable number of sub-clusters is chosen. Differently, in the case of the random baseline method, increasing β leads to a substantial increase of the TC score. This shows that the number of generated sub-clusters is optimal for the method we propose.



(b) OSV dataset. In this case, there is only one point-of-view on every musician and therefore the estimated number of sub-clusters is 2 for every musician. When $\beta \in \{0.25, 0.50\}$, only one sub-cluster per musician is generated. Due to the P/NP rejection threshold set to 0.8, only those musicians who play for at least 80% of the performance timeline will be labeled as always playing. For this reason, we observe a drop of the TC score and a decrease of the A score. Differently, when $\beta > 1$, the performance slightly improves. This saturation shows once again that the dominant difference in the images is the performed playing/non-playing action and therefore that two sub-clusters for each point-of-view are enough.

Figure 4.21: Assessing whether the amount of required human annotation by our system is optimal. We verify whether the system generates the optimal number of P/NP sub-clusters. Generating too many clusters leads to unnecessary human labor, on the other hand the critical number of P/NP sub-clusters has to be generated in order to avoid too many non-pure sub-clusters. We have added the horizontal dashed lines to compare the performance obtained by different values of β to that obtained when β is 1 — i.e., the default number of generated P/NP clusters.

discarded, and hence labeled, clusters of images, ad hoc classifiers could be trained to relabel the discarded face detections and the images from the discarded sub-clusters. Future work may also be directed towards the exploration of the additional information resources mentioned above and the exploration of relabeling strategies.

5

EXPLOITING SCENE MAPS AND SPATIAL RELATIONSHIPS IN QUASI-STATIC SCENES FOR VIDEO FACE CLUSTERING

In this chapter, we narrow down to a more specific problem that must be solved when videos of large music ensemble scenes are analyzed. As pointed out in the previous chapter, the performance of video face clustering largely affect that of the annotation step. Namely, if we infer that images of two distinct musicians represent the same person, the corresponding annotation sequence (e.g., playing/non-playing labels) is likely to contain errors. For this reason, we further studied the properties of classical music videos and found that people's positions in a scene are (quasi-)stationary. By exploiting this property, which also applies to e.g., talk shows or sitcoms, a more effective and efficient video face clustering strategy can be pursued. This chapter presents a method that automatically builds a map of the scene from an unconstrained video. The map is then exploited to perform sub-graph matching on pairs of face graphs corresponding to visually overlapping scenes. Experimental assessment indicates that the spatial relationships between people can substantially improve the clustering performance compared to the state-of-the-art in the field. Also, thanks to the scene map, our method becomes more efficient than the traditional ones, because it avoids the comparison of each face track pair. Finally, we show that our method becomes even more effective with crowded shots, which are typically challenging for traditional methods due to the lack of visual detail.

This chapter was published as: Alessio Bazzica, Cynthia C. S. Liem, Alan Hanjalic. Exploiting scene maps and spatial relationships in quasi-static scenes for video face clustering. *Image and Vision Computing* Vol.57, pp. 25-43, January 2017

Figure 5.1: Example of quasi-static scenes: the filmed people's spatial configuration is stationary over a particular time interval.

Annotating where and when a person appears in a video over time can be beneficial in many applications including video retrieval [54] and video surveillance [63]. This type of annotation essentially involves two steps, namely detecting a person in a video frame and linking the appearances of the same person across different frames. These operations can be done manually, automatically, or semi-automatically.

Manual annotation can be accurate since a human annotator can easily spot faces in the wild and say whether two faces belong to the same person. However, a fully manual annotation process is typically tedious and time consuming. As an alternative, automatic methods can be applied; they typically work by first detecting people via face detection and tracking, and then by linking the detected faces either by recognition [110] or clustering [99, 22, 76, 54]. Fully automatic methods scale well with long videos, but they may not guarantee a satisfying performance as several types of errors can occur. For instance, (profile) faces may not be detected, leading to lack of *coverage* [85]. Non-face regions can be mistaken for faces and face trackers can drift. Face recognition may also fail, especially when the face images are not detailed enough, and it can only work for known people, for which identity models need to be built using a collection of labeled images. Face clustering solutions are more generally applicable as they do not rely on trained identity models, but they typically suffer from the sub-clustering problem, which occurs when different visual appearances of one identity are recognized as different people. Given the current caveats of fully manual and automatic methods, the most viable approach for person annotation in practice may lie somewhere in-between, in the form of semi-automatic methods that effectively combine automatic computation and human annotation [108, 10]. Such methods can be further optimized by improving, in particular, the reliability of automatic clustering modules in order to significantly reduce the human effort.

In view of the above, automatic face clustering methods are critical for either fully automated or semi-automated face annotation scenarios and there are still numerous challenges to be pursued in order to bring the robustness and reliability of such methods to an acceptable level. In this chapter we provide a novel contribution in this direction.

A logical starting point to develop an automatic face clustering method is the information derived from the visual appearance of a face in a video frame. Relying on this source alone has, however, been shown to be unreliable due to a large degree of varia-

tions of faces' visual appearances across video frames and shots. This is due to the factors like camera viewpoint, face pose (e.g., profile vs. frontal face) and face scale (e.g., close-up vs. group of people) [107]. To cope with these challenges, contextual information could be exploited that characterizes specific categories of video.

In [53], visual features related to clothing proved to be beneficial in the case of TV talk shows. Furthermore, when the audio channel is closely coupled to the visual one, as in the case of a debate with the active speaker being filmed, speech features could strengthen the link between two people's appearances if the visual channel is not sufficiently informative [54]. While these approaches may lead to better performance, they are not applicable to all videos since the required information may not always be available. Methods relying on visual context information (e.g., clothing) may still suffer from the challenges of matching images at multiple scales and from multiple camera angles. Besides, there are cases in which the additional analyzed cues may not be discriminative enough to distinguish different subjects (e.g., similar clothing, similar voice).

Another step towards a more comprehensive solution is including contextual information which still is available in the visual domain, but related to the environment. For instance, people co-occurrence patterns [103, 107] can be used to link face sub-clusters. This approach can help to improve the recall. However, it relies on the assumption that the sub-clusters to merge are nearly 100% pure, which is difficult to guarantee in practice [13].

In this work, we propose a video face clustering method that exploits contextual information derived from the relative position of people appearing in *quasi-static scenes* — i.e., scenes in which the spatial configuration of the objects appearing in a video is (quasi-)stationary over a particular time interval. As explained in more detail in Section 5.2.1, the term "quasi" is used to allow limited variations in the visual appearance of the objects. For instance, the visual appearance of a face may change due to head movement or occlusion and the face object can also slightly move. As shown in Figure 5.1, this scene category applies to a broad range of video genres including, for instance, talk shows and TV debates, TV game shows and symphonic concerts.

In addition to being applicable to a large variety of videos, our proposed solution also removes the need for often tedious and time consuming processes of building person identity models. Therefore, although some quasi-static videos (e.g., talk shows and TV debates) may involve well-known people, for whom facial recognition could be applied, we still pursue a generic clustering solution for this entire category.

Finally, our method can handle complex situations, in which the number of people to be annotated is large. For instance, in an overview shot including dozens of people, the visual detail of each face can be quite poor. However, the relative position between two people can still provide useful information to infer the correct identities.

As shown by the example in Figure 5.2, the proposed method consists of the following steps:

- a **map of the scene** is learned by finding a set of geometrically consistent matches between keyframes (see view 1, 2, and 3);
- the **regions of visual overlap** between matching keyframes are computed (see the black, blue, and red regions and the arrows connecting them);

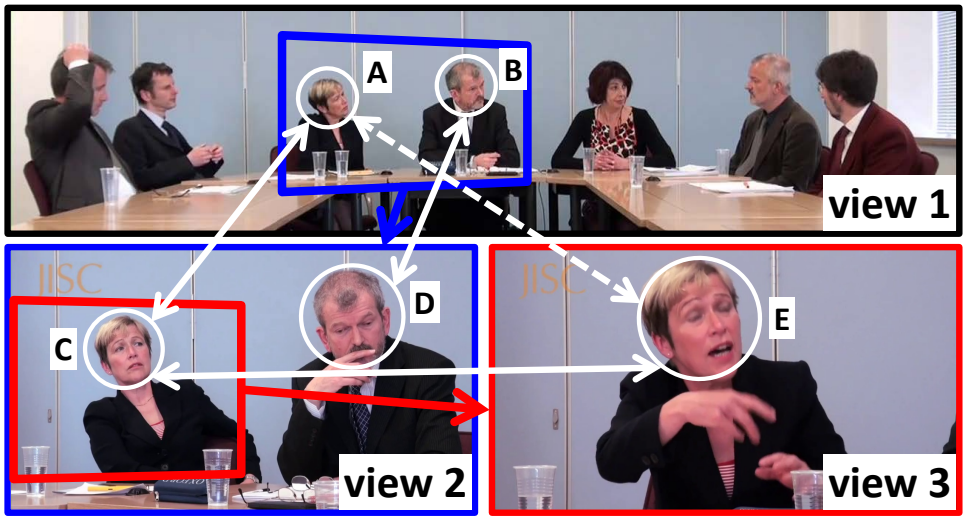


Figure 5.2: An illustration of the proposed method that exploits a map of the scene automatically built from the video keyframes (see the black, red and blue boxes and their mutual visual overlaps). Such map is then used to infer face matches which are propagated across the viewpoints (e.g., face A in view 1 indirectly matches face E in view 3 through face C in view 2).

- the overlapping keyframe pairs are used to efficiently **match faces** via sub-graph matching [111] (see the white arrows connecting faces across keyframes);

- a **face matches graph** is built and used to derive the final face clusters via connected component analysis.

Following this procedure, our algorithm avoids a brute-force comparison between each face pair and exploits the spatial configuration of the filmed people as information for matching their appearances, making it effective when dealing with different viewpoints, different zoom levels and/or varying head poses and expressions.

To the best of our knowledge, this is the first time that the spatial configuration of the filmed people is exploited as context to enable clustering of faces. To encourage further research, we release the code and a fully annotated dataset, referred to as the *QSS dataset*, which has been built using four YouTube videos and a professional symphonic orchestra video.

The chapter is organized as follows. We start by giving an overview of the related work in video face clustering in Section 5.1. Then, in Section 5.2, we explain how we address the limitation of the existing approaches and present our method in Section 5.3. The details about the datasets and the evaluation approach are reported in Section 5.4, while in Section 5.5, we assess our method by comparing it to a number of baselines and existing approaches. The conclusions drawn from this comparison are reported in Section 5.6, together with an outlook towards future research.

5.1. RELATED WORK

Video face clustering can be performed in several ways, depending on the type of information a method exploits. A standard pipeline involves the following steps: (key)frame based face detection, face tracking (typically informed by shot boundaries), visual features extraction and, finally, building face track clusters using similarity scores between faces (or face tracks). A popular paper describing a video face clustering system is [85].

The main distinguishing factor of a method is the type of information and features being exploited. A method can rely on clustering constraints that are automatically generated, and different strategies to compute visual face similarity. Depending on the type of video, contextual information can also be exploited. As anticipated in the example of Figure 5.2, our proposed method also relies on two extra steps: generating a map of the scene and solving a number of sub-graph matching problems.

In this section, we first review the state-of-the-art approaches (Section 5.1.1 — 5.1.3). Section 5.1.1 focuses on the exploitation of automatically generated constraints; Section 5.1.2 reviews how face visual similarity can be computed. Finally, Section 5.1.3 explains how different contextual information can be used. We end this section by reviewing existing algorithms for video scene maps generation and for sub-graph matching (Section 5.1.4 and Section 5.1.5 respectively).

5.1.1. EXPLOITING CONSTRAINTS

Videos can be used to automatically infer *cannot-* and *must-link* constraints [109, 102, 104]. Temporal face sequences, or face tracks, are first computed by means of image tracking. Then, the faces detected in the same keyframe are used to generate cannot-link constraints, whereas those belonging to the same track are used to infer must-link ones.

While this approach is effective, it should not be neglected that it is limited to continuous video shots. When a shot boundary occurs, all active face trackers need to be stopped and re-initialized. As a consequence, inferring constraints across different shots is not possible.

5.1.2. VISUAL FACE MATCHING

In order to link face tracks across different shots, in which the camera angle and the zoom level are typically different, visual appearance information from each detected person can be exploited. Most methods work by extracting face descriptors from which a matrix of similarity scores is computed. Ideally, when two faces belong to the same identity, the computed score is high, regardless of factors affecting the visual appearance (such as head pose, face size, expressions, glasses, lighting conditions, occlusions). Several authors proposed robust visual matching strategies to address the aforementioned challenges [22, 52, 104, 99, 112, 19, 51, 81].

The method presented in [22] relies on deformable face templates used to learn face models and account for variable poses (left-right and up-down). The matching is performed using person-specific templates at near-frontal poses only. While this method is often used to filter out non-reliable images, it cannot recognize a person's face track if the faces never occur frontally.

In [52], the authors proposed a biometric model that handles face variations across

conditions and time. More specifically, their system trains one model for each face track aiming to capture the different appearances of a person. Then, the learned models are compared to measure the similarity for each face track pair. While this approach has the potential to reduce the sub-clustering problem, there is no guarantee that each face track includes enough appearance variations to match the same person from different viewpoints and at different scales.

The method presented in [104] relies on 13 facial landmarks (to be extracted from each face detection) and a novel weighted block-sparse low rank representation. This approach shows superior performance, but it is limited to near-frontal poses only and requires sufficient image quality to extract the facial landmarks.

In [99] the authors combined the following ideas. Since most face detectors identify bounding boxes whose sizes are noisy, a brute-force face pairs comparison is performed by comparing sets of narrower and larger bounding boxes. A mutual information similarity score, based on color histograms, is adopted to compute an initial face similarity matrix. Cannot- and must-link constraints, extracted as explained in Section 5.1.1, are used to replace the scores with 0 and 1 (maximum similarity) respectively. The obtained matrix is then used to derive *global* similarity scores by using rows from the initial matrix as features. The face clusters are finally generated via spectral clustering. While this method effectively exploits the richness of each face track to overcome visual appearance variation issues, it lacks efficiency. In fact, it relies on an affinity matrix of pair-wise face similarity scores with the number of rows (and columns) equal to the sum of the face tracks lengths. The computation of each score is not as efficient as computing the distance between two feature vectors, because it needs a joint histogram to be computed for each face pair. In addition, each score requires a number of face pair comparisons quadratic in the number of bounding box resize factors.

The authors of [112] designed a method for personal photo collections which addresses the need of linking loosely connected face sub-clusters caused by large visual appearance variation of a subject. They do this by relying on a rank-order distance computed by analyzing the neighbors of each face. While this method showed superior performance and efficiency, it does not directly apply to the video face clustering case (e.g., face track constraints would be unexploited). Also, the authors recommend that, when available, information like "social context, body, time, location, event, and torso identification" [112] is included (which is indeed the case we address).

Another promising approach has recently been proposed in [19], where the authors use a number of feature sets to discover and exploit their complementarity. More specifically, by enforcing diversity constraints, their algorithm derives a number of similarity matrices which are mutually less correlated than those independently extracted from each feature set. This approach outperformed state-of-the-art methods, both on personal photo collection and video face clustering datasets. While the authors do not put any requirement on the face pose, the experiments only included (near-)frontal faces.

Spatial relationships derived from the face region were proposed in [51] and [81]. Local face features are first detected within each bounding box and then tracked over time in order to derive spatio-temporal descriptors. As in [104], these methods require sufficient image resolution. Besides, [81] is designed only for frontal faces. Earlier, we claimed to be the first to exploit spatial configuration of filmed people as context to en-

able face clustering. While the methods in [51] and [81] also use spatial relationships, these are extracted from the faces rather than from the scene.

5.1.3. EXPLOITING CONTEXTUAL INFORMATION

Various types of contextual information, like clothing visual features [53] and people co-occurrence patterns [107], were employed to link faces appearing in different shots or different pictures more robustly than done in the methods presented in Section 5.1.2.

In [53], clothing visual features are used as visual context. While such a solution is more robust to appearance variation in the face region, it suffers from appearance differences caused by occlusions and multiple-view and multi-scale shots. In addition, it is not applicable when people change clothing throughout a single video or when they wear similar clothes. The latter case for instance occurs in the case of symphonic music concerts, in which all the musicians typically dress in black.

People co-occurrence patterns have been successfully used in face clustering solutions for personal photo collections. In [107], the authors assign face cluster similarity scores by detecting groups of clusters containing faces which co-occur in the same set of pictures. Together with clothing information, their cluster merging approach significantly reduced the sub-clustering problem. However, the proposed technique is only applicable when the initial face clusters are nearly 100% pure. As observed in [13], this does not occur with faces extracted from videos, whose image quality is too often lower than visually rich portraits in personal photo collections.

5.1.4. GENERATING SCENE MAPS

Our method relies on a map of the scene that has to be automatically learned by analyzing the video keyframes. Two main categories of approaches to solving this learning problem have been proposed, namely (i) for videos consisting of a single continuous shot (e.g., [88]), and (ii) for the cases characterized by frequent viewpoint switches (e.g., [86] adapted to videos). In this work, we focus on the latter category, because it is applicable to videos that consist of multiple shots.

The technique presented in [86] has originally been designed for applications like Microsoft Photo Tourism.[1] Given an unordered collection of images, the spatial relationships between pictures are estimated in order to automatically create a navigable panorama of a specific location. For instance, a large collection of tourists' photos taken around the Trevi Fountain area in Rome can be used to reconstruct, and then explore, that particular area.

The photos are taken in an arbitrarily large but still limited geographical space (e.g., a famous square) and in an uncontrolled way — i.e., from an arbitrary number of arbitrary viewpoints. Also the pictures taken from the same viewpoint may vary in appearance because of several factors. For instance, different bystanders usually appear in the photos, lighting conditions change (e.g., weather, camera flash, night/day), and different cameras are used.

The robustness to the aforementioned factors makes [86] and similar approaches also suitable for quasi-static scene videos. For quasi-static scenes as well, recordings are

[1] http://research.microsoft.com/en-us/um/redmond/groups/ivm/PhotoTours/

made in an arbitrarily large but still limited space (e.g., a TV studio) and using multiple cameras and different zoom levels. Instead of occasional bystanders, our videos include people that move in-place — i.e., displaying movement at approximately fixed positions. Lighting conditions may change throughout the video timeline (e.g., spotlight onto a single person at a specific moment).

As will be detailed in Section 5.3.2, we build a map of the scene by finding a set of geometrically consistent matches between each image pair as done in [86]. Such a map is exploited to match faces between overlapping keyframe pairs. A similar approach is used in [86], where the authors exploit the sparse geometry information to make photo collections annotation more efficient. They state that "if an object is annotated with a set of keywords in one photo, transferring the annotation to other photos enables multiple images to be added to a keyword search database based on a single annotation.". While in [86] the authors refer to objects whose appearance does not change (e.g., a statue), in our case transferring annotations can be more challenging because the face region may be subjected to head pose change, occlusions (e.g., placing a hand on the mouth while yawning, playing a musical instrument) and different facial expressions. In addition, the visual information from a face can be poor (as shown in the example in Figure 5.3).

5.1.5. SUB-GRAPH MATCHING

Our approach includes a step where, given a pair of images I and I', the common identities have to be identified and matched. At this step, we want to include information about the relative position between people. This problem can be seen as a maximum common sub-graph (MCS) matching problem [8], where the optimal match between nodes belonging to two distinct graphs G and G' has to be found, allowing both graphs to have extra nodes and edges. In our case, the two images I and I' are respectively encoded as two graphs G and G' of which the nodes and edges represent the detected faces and their spatial relationships. As we will show in Section 5.3, it is possible to reduce the problem to a more efficient sub-graph matching problem than the MCS problem.

The state-of-the-art solutions for graph matching essentially differ in the way in which the optimization problem is solved and in the type of information that is encoded in the graphs. The Hungarian algorithm [55] works by solving a combinatorial optimization problem using a cost matrix which encodes the cost of linking every possible pair of nodes $u \in G$ and $u' \in G'$. While this is a popular algorithm, it is not suitable for our goals, since there is no straightforward way of including information on the spatial relationships between the detected faces.

We have therefore considered [111], which solves the optimization problem by jointly exploiting two similarity matrices, namely one for the nodes and one for the edges. This algorithm, called *factorized graph matching*, was successfully applied to the task of matching parts of two objects of the same family. For instance, when matching two images of two different cars, the front wheel of a car has to match to the front wheel of the other one. In [111], the spatial relationships between nodes are included by extracting edge descriptors (e.g., apparent distance and angle between the connected nodes) which are finally used to derive an edge similarity matrix. As will be detailed in Section 5.3, we adopted [111] as an off-the-shelf solution for our sub-graph matching problem.

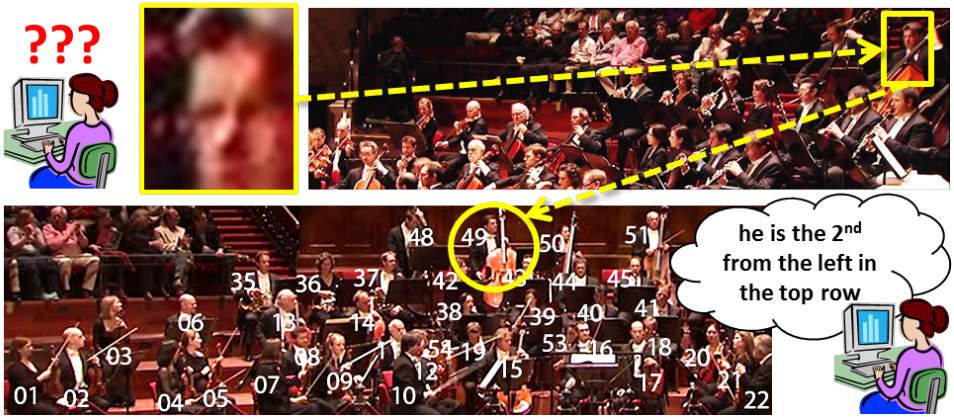


Figure 5.3: Visual information can be poor especially in images of large groups where the face resolution is low. However, a human annotator can still match the same person from different viewpoints by relying on the spatial information.

5.2. RATIONALE AND CONTRIBUTIONS

In this chapter, we address the limitations identified in Section 5.1.1 – 5.1.3 in a novel way, exploiting the properties of a restricted, yet relevant, class of videos. Namely, we focus on quasi-static scene videos, for which we present the properties in Section 5.2.1. We then explain the followed rationale in Section 5.2.2 and highlight our contribution in Section 5.2.3.

5.2.1. QUASI-STATIC SCENES

We define a *quasi-static scene* as a scene featuring objects (including faces) whose positions and visual appearance are quasi-stationary over a particular time interval. The attribute "quasi" is used to allow for slight variations in object positions (like a person moving on the chair while talking) and the variations in the visual appearance of the objects that may result, for instance, from a person turning the head or occluding the face by hand waving. Also, the visual appearance of the entire scene may slightly vary with the changes in lighting conditions (e.g., through the usage of spotlights). Finally, we also allow limited parts of the scene to substantially change their visual appearance, like in the case where there is a screen on the studio wall, of which the visual content changes over time.

A quasi-static scene video is typically recorded in a single place (e.g., a TV studio, a concert hall) and, in general, by multiple cameras placed at different locations allowing for usual recording actions (e.g., through zooming or panning). Examples of quasi-static scenes are political debate programs or TV game shows, which are typically organized in a number of stages during which the debaters or players hold the same general position. A further example is a video of a symphonic concert, in which each musician performs from the same spot. Quasi-static scenes may also occur in movies (e.g., dialog scenes).

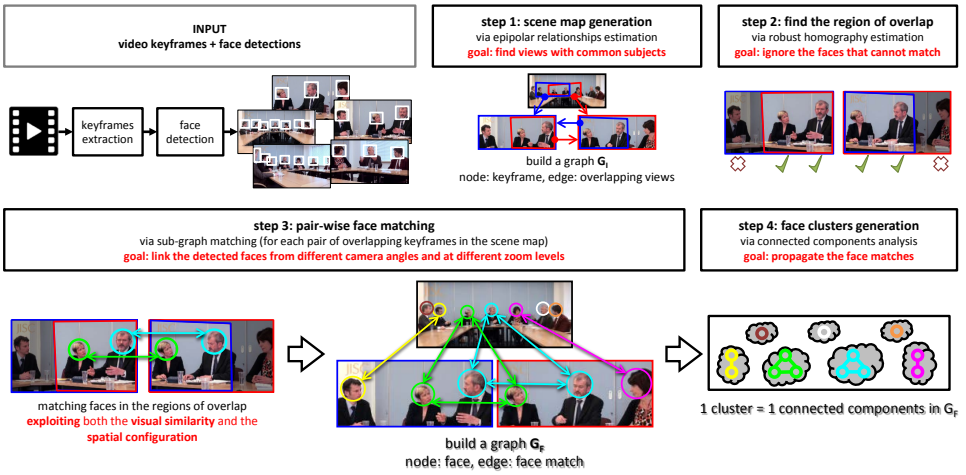


Figure 5.4: An overview of our proposed method designed to link faces across shots of quasi-static scene videos. Given a set of keyframes, we first build a scene map G_I in which overlapping keyframes pairs are linked. The regions of overlap are estimated and a pair-wise face matching problem is solved for each linked keyframe pair. Finally, the face matches are used to build a graph G_F whose connected components will correspond to the sought face clusters.

5

5.2.2. APPROACH RATIONALE

Our method is inspired by the experience acquired while manually annotating symphonic music videos, as illustrated by the example of Figure 5.3. At the top left, we see a face image extracted from a small bounding box. The visual information is extremely poor, making it hard (or even impossible) to recognize a person even for a human annotator. However, if we consider *where* the bounding box is in the image (top right), we can exploit the fact that the positions of the filmed subjects are stationary. As a consequence, we can infer a match between face images from different viewpoints with high confidence. While this is intuitive for a human being, devising an automatic computer vision method that mimics this approach is not trivial.

5.2.3. CONTRIBUTIONS

In this work, we propose an effective and efficient ad-hoc video face clustering algorithm for the class of quasi-static scene videos. The properties of these videos allow for the automatic buildup of a scene map, together with the inference of spatial relationships between the filmed people. As we will show, employing this contextual information is beneficial for tackling the challenging problem of linking face tracks across shots. Leveraging this, the core contributions of this chapter are as follows:

- we propose an efficient method to check if two faces belong to the same person on *selected* face track pairs, hence avoiding a brute-force comparison between each possible face track pair;
- we propose a new agglomerative face clustering strategy based on connected component analysis to efficiently propagate face matches across different viewpoints

Table 5.1: List and descriptions of relevant symbols.

symbol	description
$\mathbf{f}_k$	k-th keyframe
K	number of extracted keyframes
$\mathscr{D}$	set of detections
$\mathbf{d}_{k,l}$	l-th detection in the k-th keyframe
G_I	keyframe connectivity graph
$\mathbf{p}_{k,j}$	region of overlap between keyframes
$\mathscr{G}_k$	temporary face graph for keyframe $\mathbf{f}_k$
G_F	face matches graph
λ	face matches graph pruning factor

and different zoom levels.

5.3. PROPOSED METHOD

Our method, summarized in Figure 5.4, automatically generates clusters of faces detected on a keyframe basis. Face tracking can optionally be used to enrich the visual appearance information extracted from each face image within each keyframe (see Section 5.3.1). In the ideal case, each formed cluster will contain all the faces of a single identity appearing in the given video. Our method consists of four steps, as detailed below.

The goal at the first step is to find good keyframe pair candidates in which we can find people in common (Section 5.3.2). We generate a map of the scene by finding robust geometrically consistent visual matches across the extracted keyframes, like was done in [86]. Knowing that two keyframes cannot match is used to avoid unnecessary comparisons between the faces they contain. The computed map is then encoded as a graph G_I, where each node corresponds to a keyframe. Two nodes in G_I will be connected by an edge, if the corresponding keyframe pair exhibits *visual overlap*. Such overlap occurs when there are one or more visual cues occurring in both keyframes (e.g., overlapping keyframes resulting from a continuous camera panning).

Then, as the second step, for each edge in G_I we find the regions of overlap (Section 5.3.3) in order to focus on the face detections falling in these regions. In this way, we exclude additional face pairs that cannot match, because placed in different parts of the scene. Due to the first and the second step, we do not need to compute a full face affinity matrix, as it is instead done in most of the existing methods.

In the third step, we consider again each edge in G_I in order to jointly match all the faces within the regions of overlap of each found keyframe pair (Section 5.3.4). This is done by solving a sub-graph matching problem via the factorized graph matching algorithm presented in [111]. Here, we do not rely just on visual appearance information, but we also consider the spatial configuration of the faces in the two keyframes. In this way, the matching process becomes robust to variations in viewpoint and appearance (e.g., lighting, different pose, hand on the mouth while yawning). All the inferred face matches are stored in a new graph G_F, in which a node corresponds to a detection and

an edge to a matching face pair.

Finally, we isolate the nodes in G_F into clusters that correspond to each connected component in G_F (Section 5.3.5). As a consequence, all the face detections linked together by a direct edge or by a path will fall into the same cluster.

The relevant notation used in the following sections is presented in Table 5.1 and our video face clustering algorithm is reported in Algorithm 1, also containing the pointers to different steps of the method. In the remainder of this section, we explain all aspects of our method and its different steps illustrated by the modules in Figure 5.4. Each module in the figure has its counterpart in the corresponding part of this section.

Algorithm 1 Quasi-static scene face clustering.

function QSS FACE CLUSTERING
  $G_I \leftarrow$ empty undirected graph ▷ Steps 1 and 2
  for $k \in \{1 \dots K\}$ **do**
    for $j \in \{k+1 \dots K\}$ **do**
      if $\mathbf{f}_k$ matches $\mathbf{f}_j$ **then**
        add edge $k \leftrightarrow j$ in G_I
        $\mathbf{p}_{kj} \leftarrow$ REGION OF OVERLAP($\mathbf{f}_k, \mathbf{f}_j$)
        $\mathbf{p}_{jk} \leftarrow$ REGION OF OVERLAP($\mathbf{f}_j, \mathbf{f}_k$)
      end if
    end for
  end for
  $G_F \leftarrow$ empty undirected graph ▷ Step 3
  for all edge $(k, j) \in G_I$ **do**
    $\mathcal{G}_k \leftarrow$ TEMPORARY FACE GRAPH($k, \mathbf{p}_{j,k}$)
    $\mathcal{G}_j \leftarrow$ TEMPORARY FACE GRAPH($j, \mathbf{p}_{k,j}$)
    for all $(\mathbf{d}_{k,l}, \mathbf{d}_{j,m}) \in$ SUB-GRAPH MATCHING($\mathcal{G}_k, \mathcal{G}_j$) **do**
      add edge $(k, l) \leftrightarrow (j, m)$ in G_I
    end for
  end for
  return CONNECTED COMPONENTS(G_F) ▷ Step 4
end function

function TEMPORARY FACE GRAPH($k, \mathbf{p}$)
  $\mathcal{G} \leftarrow$ empty directed graph
  for all detection $\mathbf{d}_{k,l}$ in $\mathbf{f}_k$ **do**
    if $\mathbf{d}_{k,l}$ falls in $\mathbf{p}$ **then** add node (l) in $\mathcal{G}$
    end if
  end for
  add edges in $\mathcal{G}$ ▷ see Section 5.3.4
  return $\mathcal{G}$
end function

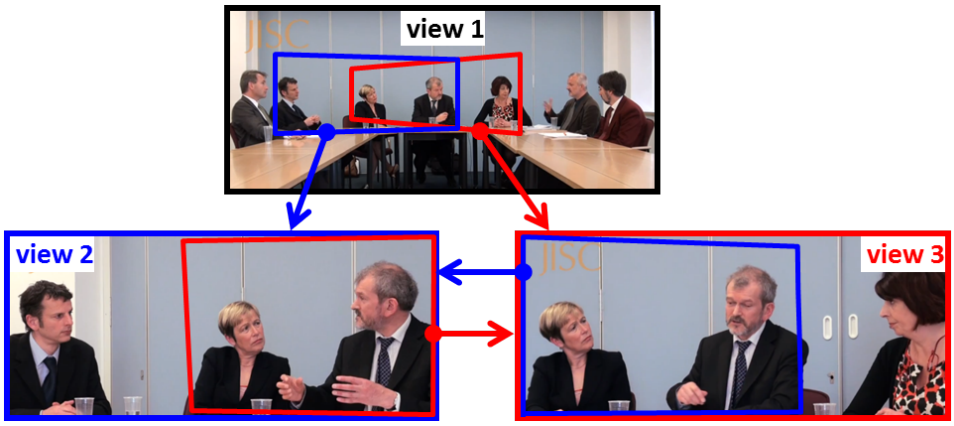


Figure 5.5: The map of the scene is built by identifying keyframe pairs which exhibit visual overlap.

5.3.1. INPUT: KEYFRAMES AND DETECTIONS

We consider the case in which the video face clustering is performed on a keyframe basis — i.e., not continuously over time. The keyframes are extracted from different video shots [3] and denoted as $\mathbf{f}_k$, with $k \in \{1 \dots K\}$ and K is the overall number of extracted keyframes. This case can be extended to the continuous one by means of face tracking, as done in several related works. However, it is out of the scope of this chapter to propose such an extension. We also do not propose a specific keyframe sampling strategy. The details about the way we have selected the keyframes in our proposed dataset are reported in Section 5.4.1.

Once the keyframes are extracted, the presence of one or more subjects in each keyframe $\mathbf{f}_k$ has to be annotated by determining the set of detections $\mathscr{D} = \{\mathbf{d}_{k,l}\}$. Here, $\mathbf{d}_{k,l}$ is the l-th detection and each $\mathbf{d}$ is defined as $\mathbf{d} = (x, y, w, h)$ where (x, y) and (w, h) are the center and size of the face bounding box respectively. Differently from several related methods mentioned in Section 5.1, we impose no restrictions to the accuracy of positioning the face bounding boxes nor on the head visual appearance of the head. Namely, heads can appear at **any** pose, also from the back, and occluded faces are tolerated (e.g., a hand on the face) as well as different facial expressions. Details on how we approached face detection are given in Section 5.4.1.

5.3.2. STEP 1: SCENE MAP GENERATION

At this step, we aim to generate a map of the scene which is encoded as a keyframe connectivity graph G_I. Two nodes in G_I will be connected by an edge if the correspondent keyframe pair exhibits *visual overlap*. Such overlap occurs when there are one or more visual cues occurring in both keyframes (e.g., overlapping keyframes resulting from a continuous camera panning moving towards a constant direction). The common cues can belong to the scene and/or to the filmed people. An example of the output that we expect is reported in Figure 5.5, in which we can see 3 keyframes and the overlap relationships between each keyframe pair.

We first find a set of geometrically consistent matches between each keyframe pair

as done in [86]. Namely, we detect keypoints (e.g., SIFT [60]) and compute the associated local descriptors for every keyframe storing them in a kd-tree. For each keyframe pair, the keypoint descriptors in the corresponding kd-trees are matched via approximate nearest neighbors [4]. The found matches are filtered out by using the ratio test [60] and removing inconsistent ones — i.e., those associated to a keypoint descriptor in one image which is matched more than once in the other image. Then, for each keyframe pair a fundamental matrix is robustly estimated using RANSAC [36] with the eight-point algorithm [46]. The computed matrix is refined via the Levenberg-Marquardt algorithm [73] and a final outliers filtering operation is performed. Finally, for each keyframe pair such that the number of inliers is greater than a given threshold, a weighted edge is added to the keyframe connectivity graph G_I setting the number of inliers as weight.

Depending on the actual visual overlap between keyframes, as well as the effectiveness of the overlap detection method described above, the number of connected components in G_I will range from 1 to K (the number of extracted keyframes). A connected component could result, for instance, from a frequent shot recorded from the same viewpoint, at the same zoom level, and with no visual overlap with any other shot. In this particular example, the corresponding connected component can be strongly connected, hence having a number of edges that is quadratic to the number of nodes. For our face clustering method, this is inefficient, since it would generate unnecessary keyframe comparisons at the pair-wise face matching step (Section 5.3.4). We therefore reassign G_I to its maximum spanning tree (MST step).

5.3.3. STEP 2: REGIONS OF OVERLAP

The goal at this step is identifying the regions of overlap for each keyframe pair in G_I. This output will be used in the subsequent step to reduce the search space when the detected faces have to be matched. An example of the produced output is shown in Figure 5.6, in which the brightest areas correspond to the overlap regions. Faces outside of them — i.e., within darker areas — can be ignored because they belong to different identities (and therefore they cannot match).

Given a keyframe pair $(\mathbf{f}_k, \mathbf{f}_j)$ and the final keypoint matches found during the first step, the two regions of overlap are computed as follows. First, the homography H_{kj} between the $\mathbf{f}_k$ and $\mathbf{f}_j$ is computed using RANSAC. Then, the four vertices of $\mathbf{f}_k$ are projected onto $\mathbf{f}_j$ through $H_{k,j}$ and, analogously, the four vertices of $\mathbf{f}_j$ are projected onto $\mathbf{f}_k$ through $H_{j,k} = H_{k,j}^{-1}$. The resulting polygons, denoted as $\mathbf{p}_{k,j}$ and $\mathbf{p}_{j,k}$ respectively, define the sought region of overlap.

5.3.4. STEP 3: PAIR-WISE FACE MATCHING

Once the map of the scene is generated, we exploit it to match faces between overlapping keyframe pairs. The idea is to jointly match faces by solving an optimization problem that takes into account both the similarity between faces across two keyframes, as well as their relative positions. To this end, we use the factorized graph matching algorithm presented in [111].

Given a keyframe pair $(\mathbf{f}_k, \mathbf{f}_j)$ for which an edge was added in G_I, we build two temporary graphs $\mathcal{G}_k$ and $\mathcal{G}_j$ encoding the presence of faces in the regions of overlap computed for the pair. Each temporary graph is built as follows (example given for the keyframe $\mathbf{f}_k$):

Figure 5.6: Given a keyframe pair $(\mathbf{f}_k, \mathbf{f}_j)$ with visual overlap, we generate two temporary graphs $\mathcal{G}_k$ and $\mathcal{G}_j$ encoding the presence of faces in the regions of overlaps and their spatial relationships.

- we select the face annotations $\mathbf{d}_{k,l}$ whose bounding box center falls in the region of overlap $\mathbf{p}_{j,k}$;
- for each selected face, we add a node in $\mathcal{G}_k$;
- each node is labeled with the corresponding bounding box's center (x, y);
- such coordinates are used with the Delaunay triangulation algorithm to compute a list of undirected edges between the nodes in $\mathcal{G}_k$;
- for each undirected edge $u \leftrightarrow v$, the directed edges $u \rightarrow v$ and $u \leftarrow v$ are added to $\mathcal{G}_k$;
- each directed edge e is labeled with the Euclidean distance $D(e)$ between the nodes it connects as well as the angle $\alpha(e)$ between itself and the horizontal line;
- the number of nodes and directed edges in $\mathcal{G}_k$ is denoted as n_k and m_k respectively;
- the edges of $\mathcal{G}_k$ are encoded as node-edge incidence matrices $\mathbf{U}_k$ and $\mathbf{V}_k$.

$\mathcal{G}_k$ and $\mathcal{G}_j$ are directed because the angle between a directed edge $u \rightarrow v$ and the horizontal line is different from that computed for $u \leftarrow v$. Figure 5.6 shows an example of the generated output.

The two temporary graphs are then used as follows to find the optimal node matches, which will correspond to the sought face matches. We compute two affinity matrices $\mathbf{K}_p \in \mathbb{R}^{n_k \times n_j}$ and $\mathbf{K}_q \in \mathbb{R}^{m_k \times m_j}$ which are the node and the edge similarity matrices respectively. $\mathbf{K}_p$ is computed considering the visual appearances of each face and/or the corresponding visual context. For instance, a set of global features can be extracted from each face bounding box and used to compute similarity scores between the selected faces in $\mathbf{f}_k$ and $\mathbf{f}_j$. More details about the way $\mathbf{K}_p$ was computed in our experiments are reported in Section 5.4.2. The edge affinity matrix $\mathbf{K}_q$ is computed by comparing each pair of directed edges in $\mathcal{G}_k$ and $\mathcal{G}_j$. Given an edge pair (e, e'), we compare the labeled distances and angles as follows. First, for each graph, we divide the computed edge distances $D(e)$ by the largest value. Then we define $\Delta_D = |D(e) - D(e')|$ and Δ_α as

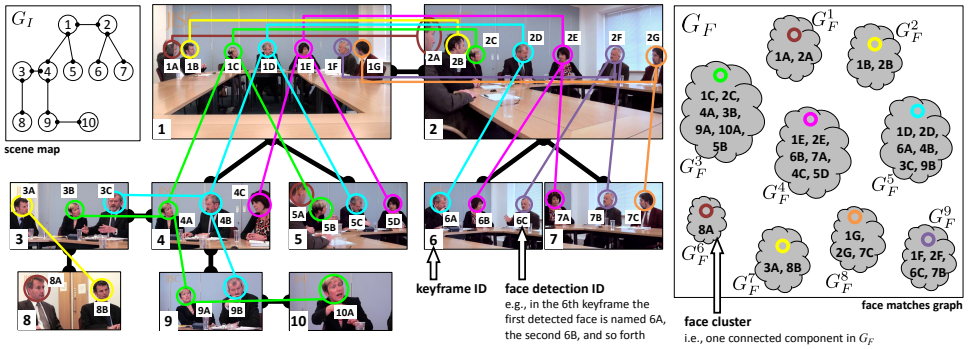


Figure 5.7: Once the face matches graph G_F, which is built on top of the scene map encoded in the G_I graph, is computed, we generate the face clusters by simply isolating the faces belonging to each connected component $G_F^i \subset G_F$. The colored edges connecting the faces across the keyframes represent the found face matches — i.e., edges in G_F.

5

the smallest difference between the angles $\alpha(e)$ and $\alpha(e')$. These two sets of dissimilarity scores are divided by $\max\Delta_D$ and π respectively. Finally we compute the similarity between e and e' as $\exp(-2.5(\Delta_D + \Delta_\alpha))$. Using the node-edge incidence matrices pairs $(\mathbf{U}_k, \mathbf{V}_k)$ and $(\mathbf{U}_j, \mathbf{V}_j)$, and the affinity matrices $\mathbf{K}_p$ and $\mathbf{K}_q$ (both scaled in the interval $[0,1]$), we find the optimal matching $\mathbf{X}$ solving the following problem:

$$\hat{\mathbf{X}} = \arg\max_{\mathbf{X}\in\Pi} J_{gm}(\mathbf{X}) = \mathrm{tr}(\mathbf{K}_p^T\mathbf{X}) + \mathrm{tr}(\mathbf{K}_q^T\mathbf{Y}) \tag{5.1}$$

where Π, the set of partial permutation matrices, and Y are defined as

$$\Pi = \{\mathbf{X} | \mathbf{X} \in \{0,1\}^{n_k \times n_j}, \mathbf{X}\mathbb{1}_{n_k} \leq \mathbb{1}_{n_j}, \mathbf{X}^T\mathbb{1}_{n_k} \leq \mathbb{1}_{n_j}\} \tag{5.2}$$

$$\mathbf{Y} = (\mathbf{U}_k^T\mathbf{X}\mathbf{U}_j \circ \mathbf{V}_k^T\mathbf{X}\mathbf{V}_j) \in \{0,1\}^{m_k \times m_j} \tag{5.3}$$

assuming $n_k \geq n_j$ ($\circ$ in (5.3) is the Hadamard product). The objective $J_{gm}(\mathbf{X})$ is optimized with the path following algorithm described in [111]. $\hat{\mathbf{X}}$ encodes the matches between the faces falling in the regions of overlap of the two analyzed keyframes.

Finally, we build a face matches graph G_F, which is initialized by adding one node for each detected face $\mathbf{d}_{k,l} \in \mathscr{D}$. All the inferred face matches (encoded in the set of computed $\hat{\mathbf{X}}$) are stored in G_F by adding an edge for each matching face pair.

5.3.5. STEP 4: FACE CLUSTERS GENERATION

Once all the visually overlapping keyframe pairs are analyzed, the face matches graph G_F is used to form the sought face clusters. At this final step, we simply isolate the connected components $G_F^i \subset G_F$ and we form one cluster for each sub-graph G_F^i. The faces of the i-th face cluster correspond to the nodes of G_F^i. This results in a simple and lightweight operation which allows to link faces across non-matching keyframe pairs.

Figure 5.7 shows an example of the expected output and summarizes how the G_I and G_F graphs are used to form the face clusters. G_I and G_F are drawn in the leftmost and rightmost boxes respectively. In the middle of Figure 5.7, G_F is visualized on top of G_I. G_I

Table 5.2: QSS dataset properties.

video	annotated length	annotated shots	face tracks	faces	identities
News debate	15 / 17 min.	134 / 144	325	4,268	6
Game show	15 / 19 min.	132 / 183	280	5,266	5
TV series	5 / 5 min.	82 / 82	175	1,507	3
Roundtable	5 / 5 min.	20 / 20	87	3,263	7
Orchestra	4h 35m / 6h 43m	210 / 381	2,066	28,567	55

is always a forest, due to the MST step described in Section 5.3.2. Its nodes are visualized as the corresponding keyframes and the edges as thick black lines. The nodes in G_F are shown as circles and the sought face matches as colored lines (one color per identity was used). In the rightmost box the connected components in G_F are represented as clouds containing the face detection IDs belonging to each cluster. There are seven identities, nine face clusters are formed.

By inspecting Figure 5.7, it is clear that some additional keyframe links would have been unnecessary. For instance, connecting the keyframes 3 and 9 would not have changed the final results. This justifies why we have added the MST step to relax the keyframe connectivity graph G_I (see Section 5.3.2). However, if the keyframes 1 and 3 had been connected in G_I, we would have correctly obtained one face cluster less, because G_F^2 and G_F^7 would have been linked by the face match B1-A3. We decided to always apply the MST relaxation and we left a detailed investigation of alternative strategies for future work.

Optionally, to favor the formation of nearly 100% pure clusters, the following pruning step can be applied to G_F:

- a fraction of the total number of edges m is set, e.g., $m' = \lceil \lambda m \rceil$ where $\lambda \in [0,1]$;
- the edges in G_F are weighted using the node similarity scores computed for the affinity matrices $\mathbf{K}_p$ (see Section 5.3.4);
- the weights are then used to rank the edges (the top edge has the highest weight);
- the top m' edges are retained.

A value of λ close to zero will favor purity over a limited sub-clustering generation.

5.4. EXPERIMENTAL SETUP

In this section we present our dataset (Section 5.4.1), provide details about the implementation of our method (Section 5.4.2), define the evaluation measures (Section 5.4.3), and describe the baseline methods and the comparisons we performed (Section 5.4.4).

5.4.1. DATASET

We propose a novel dataset, namely the *QSS dataset*, that has been annotated by the authors. It consists of five videos, four were downloaded from YouTube ("news debate", "game show", "TV series" and "roundtable" - QSS-YouTube subset) and one is a symphonic orchestra's performance raw recording ("orchestra"). The QSS-YouTube videos

Table 5.3: QSS-YouTube subset video IDs.

video	YouTube ID
News debate	5RFWD7DRkAs
Game show	RsRjP_suxmk
TV series	t0xERWYs55g
Roundtable	YSEl2WJtg8w

consist of broadcast edits with several shots; they have been recorded from multiple viewpoints and at different zoom levels. The "orchestra" video consists of 4 sets of 7 synchronized videos (one set per movement); the videos within each set are synchronized and have been recorded by different cameras. For each video, we provide a set of keyframes, one per shot, annotated in terms of the faces appearing therein and a set of face tracks, also one per shot. We note that we have not processed and annotated all the shots in the videos due to the complexity of the annotation task. We believe, however, that the available amount of data is sufficient to arrive to conclusive results about the method performance. The properties of the videos and the available data per video in the QSS dataset are reported in Table 5.2 and Table 5.3.

Differently from other benchmark datasets, like "Buffy the vampire slayer" [85] and "Notting Hill" [99], our dataset includes manually fixed face annotations (more than 2900 labeled faces in total), including profile poses and also heads filmed from the back. Such annotations can be exploited for a more detailed assessment of people video annotation systems. In particular, *timeline coverage* [13] can now be measured; it relates to how many characters are identified over time and, as stated in [85], it is often neglected.

For each shot, we extracted the central keyframe and semi-automatically annotated the presence of faces using a state-of-the-art face detector [64] and an off-the-shelf annotator[2]. In order to obtain error-less face detections, false detections were manually discarded and missing faces were added including profile faces and even people appearing from the back (which frequently occurs in the orchestra video). For each non-discarded face detection, we ran the OpenTLD tracker[3] [50] to extract a face track from the shot to which the face belongs. Since the keyframes are selected as central frame in the shot, we ran the tracker twice for each face (backward and forward in time) linking the computed tracklet pairs. Similarly as done in [102], we sub-sampled the face tracks by retaining one face image every 10 frames from each face track. For the "orchestra" video only, due to its long duration and large number of filmed people, we instead retained one face image every 30 frames and also limited the face track lengths to 18 images. Note that after this data reduction process, the data size of our experiment is still much larger than that of [99], where about 1,000 face images per video were used (see Tab 5.2 for comparison). The annotations that we produced are publicly available for download[4] and include the reference keyframe and shot boundary timestamps, the face bounding boxes geometry and the ground-truth labels. The original raw orchestra video is available on request.

[2] `http://faint.sourceforge.net/`
[3] `https://github.com/zk00006/OpenTLD`
[4] `http://mmc.tudelft.nl/users/alessio-bazzica#QSS-dataset`

5.4.2. IMPLEMENTATION DETAILS

In order to experiment with variations of face-related visual object appearance, for each face, we also extracted the upper body region — i.e., face and torso — extending the face bounding boxes as done in [13], but ignoring the head pose. We extracted the following sets of global features as done in [13]. Faces were described by Pyramid Histograms of Oriented Gradients, Joint Composite Descriptor, Gabor texture, Edge Histogram, and Auto Color Correlogram [61]. The same features were used for upper body images adding the Local Binary Patterns. The node similarity matrices $\mathbf{K}_p$ defined in Section 5.3.4 were computed exploiting the available face tracks as follows. Given two faces belonging to two distinct keyframes, their similarity score is computed as the similarity between the correspondent face tracks using the "min-min distance" [85] (we used the Euclidean distance). Namely, we build a temporary matrix of pair-wise face (or upper body) similarity scores, and we get its maximum.

At the scene map generation step (Section 5.3.2), we found the geometrically consistent matches between each keyframe pair as done in [86] using the SIFT keypoint detector [60] and the Bundler Structure from Motion Toolkit[5] without changing the default parameters. For the pair-wise face matching step (Section 5.3.4), we used the Factorized Graph Matching Toolbox from [111][6] using the default parameters.

5.4.3. EVALUATION MEASURES

We evaluate and compare different face clustering methods in two ways. In the first one, we assess how each method can balance precision and recall. To this end, we adopt the *B-cubed precision and recall* scores [2] to compute one series of score pairs for each evaluated method by varying the number of generated clusters. In the second one, we rank the evaluated methods using the average B-cubed F-measure.

The B-cubed scores are computed as follows. Let $L(\mathbf{d})$ and $C(\mathbf{d})$ denote the ground-truth face label and the assigned cluster label of a face detection $\mathbf{d}$ respectively. We first define the *correctness* function $M(\mathbf{d},\mathbf{d}')$ as follows:

$$M(\mathbf{d},\mathbf{d}') = \begin{cases} 1 \text{ if } L(\mathbf{d}) = L(\mathbf{d}') \wedge C(\mathbf{d}) = C(\mathbf{d}') \\ 0 \text{ otherwise} \end{cases} \tag{5.4}$$

Then, the B-cubed precision and recall (Pre and Rec) and the weighted F-measure (F_β) are defined as follows:

$$Pre = \underset{\mathbf{d}\in\mathscr{D}}{\text{average}}\left(\underset{\mathbf{d}'\in\mathscr{D}|C(\mathbf{d})=C(\mathbf{d}')}{\text{average}} \left(M(\mathbf{d},\mathbf{d}') \right) \right) \tag{5.5}$$

$$Rec = \underset{\mathbf{d}\in\mathscr{D}}{\text{average}}\left(\underset{\mathbf{d}'\in\mathscr{D}|L(\mathbf{d})=L(\mathbf{d}')}{\text{average}} \left(M(\mathbf{d},\mathbf{d}') \right) \right) \tag{5.6}$$

$$F_\beta = (1+\beta^2) \cdot \frac{Pre \cdot Rec}{(\beta^2 \cdot Pre) + Rec} \tag{5.7}$$

[5] https://github.com/snavely/bundler_sfm
[6] http://www.f-zhou.com/gm_code.html

Table 5.4: Evaluated face clustering methods.

Method	Description
RND	Random baseline.
Bf, Bu	Baseline method, based on face (Bf) or upper body (Bu) visual appearance.
MI	Mutual information visual similarity measure, intra-shot constraints, and spectral clustering [99].
QSSenf, QSSenu	Proposed method, based on face (QSSf) or upper body (QSSu) visual appearance.

In our experiment, we set $\beta = 0.5$ in order to put more emphasis on the precision term. This was done because, for face clustering tasks, precision is typically preferred over recall. In fact, when a human operator has to refine the automatically generated clusters, it is often preferable to merge nearly 100% pure ones instead of cleaning clusters which contain faces belonging to several identities.

5.4.4. EVALUATION APPROACH

We select a number of methods serving as references for assessing our proposed method. These methods, listed in Table 5.4, are representative of the various state-of-the-art solutions reported in Section 5.1.2 – 5.1.3. Briefly, next to a random baseline (*RND*), we include a number of other baselines (*Bf, Bu*) in order to check whether minimum significant performance is achieved. Furthermore, we compare to [99] (*MI*) which relies on face tracks constraints and face visual appearance similarity (Section 5.1.1 – 5.1.2). MI is based on mutual information and a brute-force comparison, which should be more robust to visual appearance variations. Finally, the list includes a method based on visual features extracted from the face (*QSSenf*) or upper body region (*QSSenu*). The latter has been added in order to assess the power of visual information from the face region and the context (more specifically, clothing).

The random baseline (RND) assigns a random cluster to each face track. The other two baseline methods use k-means and represent each face track as a single feature vector extracted either from the face region (Bf method) or the upper body region (Bu method) in the corresponding keyframe — i.e., no additional visual information derived from the available face tracks is exploited. The same face and upper body visual feature sets used for the QSS methods are used (see Section 5.4.2). We also exploit keyframe-based cannot-link constraints to set the similarity score of each face pair in each keyframe to zero — i.e., soft cannot-link constraints. We evaluate our method computing the node similarities for the pair-wise face matching step (Section 5.3.4) considering either face (QSSenf) or upper body (QSSenu) visual appearance (see Section 5.4.2 for the implementation details). The MI method works by applying spectral clustering to an affinity matrix having one row/column per face. Hence, one label per face is assigned. We derive a single label per face track by majority voting as done in [102].

Note that we extract a single label per face track because our objective is assessing how good each evaluated method is at linking face tracks across shots. Namely, we consider the task of labeling each face track and not each face within each face track. In

Table 5.5: Evaluated QSS face clustering methods.

Method	Description
QSSe	QSS with edge similarity only — i.e., $\mathbf{K}_p$ is empty.
QSSnf	QSS with node similarity only (face features) — i.e., $\mathbf{K}_q$ is empty.
QSSnu	QSS with node similarity only (upper body features) — i.e., $\mathbf{K}_q$ is empty.
QSSenf	Full QSS method (face features).
QSSenu	Full QSS method (upper body features).
_co	People co-occurrence analysis refinement step.

this way, long shots containing long face tracks, for which a large number of cannot- and must-link constraints can be automatically derived, will not bias the final result, and we can therefore measure the actual power of making correct links between people appearing across different video shots.

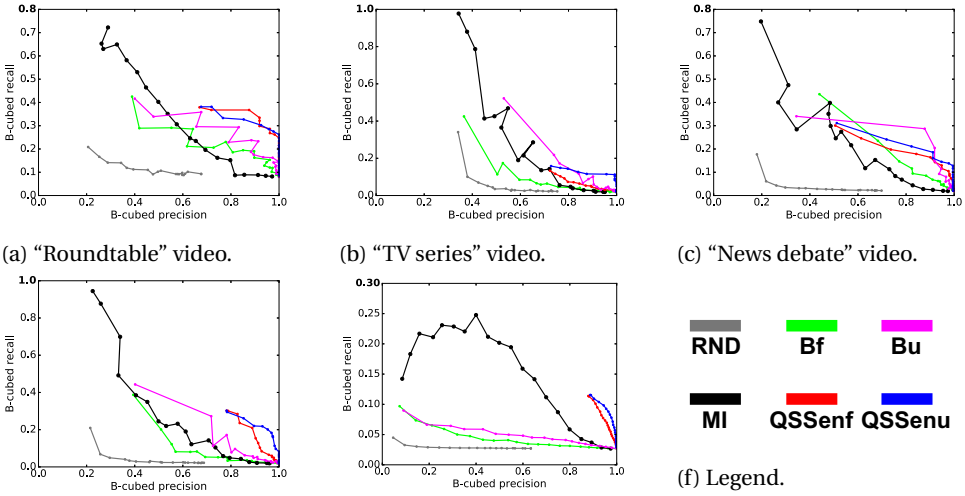
In order to gain more insights on the power of scene and spatial relationship information for video face clustering, we also compare different internal options applied to our method. Namely, we measure the performance of the QSS algorithm using node-only and edge-only similarity information at the pair-wise face matching stage (*QSSnf* / *QSSnu* and *QSSe* respectively). In the latter case, upper body visual similarities are used only at the face matches graph pruning step (see Section 5.3.5). We also add the co-occurrence analysis method used in [107] as optional refinement step. We do this to understand to what extent our method implicitly exploits the information encoded in people co-occurrence patterns. The refinement step is applied iteratively, merging the best face cluster pairs obtained by considering the top N matches where N is approximately one-third of the total number of matching pairs. The loop ends if the maximum number of iterations is reached (we used 10 in our experiments) or when no matching pairs are found. The list of QSS alternative methods is reported in Table 5.5.

We conduct the experiments described above using the error-less face detections (see Section 5.4.1) in order to assess the effectiveness of the proposed face clustering method itself. In particular, we want to assess the idea to exploit spatial relationships between objects in quasi-static scenes as contextual information, with the focus on the upper limit of its performance. Assessing the side effects of real-world (imperfect) face detector errors is beyond the scope of this chapter.

Still, one may wonder how noisy spatial relationships, generated by face detection errors, could affect the final performance of our proposed method. We therefore conduct an additional experiment to simulate how the QSSenu method tolerates the most common type of face detection error [13], namely missing faces. The results are reported in Section 5.5.2.

More in detail, we simulate missing faces by randomly discarding different percentages of error-less annotations from our dataset. We do not compare QSSenu to other methods because a fair comparison would require a more realistic way of discarding faces. In fact, the random process we use does not take into account size or orientation of a head, whereas in practice face detectors more likely miss small faces or profile ones.

In all the experiments, we balance precision and recall in different ways depending on the method. For the QSS algorithm, we control the number of generated clusters



(a) "Roundtable" video.

(b) "TV series" video.

(c) "News debate" video.

(d) "Game show" video.

(e) "Orchestra" video.

(f) Legend.

Figure 5.8: Comparison between baselines (RND, Bf, Bu), the mutual information method (MI) proposed in [99] and our proposed method (QSSenf, QSSenu). Bf and QSSenf rely on face features, whereas Bu and QSSenu rely on upper body visual features. This figure is best seen in color.

by varying λ. As explained in Section 5.3.5, it is a parameter at the pruning step which affects the number of retained edges in the face matches graph. In all the other methods, we pick the desired number of face clusters from the interval between the number of identities in the video and the number of detected faces. We sample 20 points from each B-cubed precision recall curve.

5.5. RESULTS AND DISCUSSION

In this section, we report the results obtained on the QSS dataset along with examples of intermediate output from our algorithm (Section 5.5.1). Additional results showing the extent to which our method is robust to face detection false negatives are reported in Section 5.5.2. We also explain how our approach may fail (Section 5.5.3) and comment on the scalability of the evaluated methods (Section 5.5.4).

5.5.1. FACE CLUSTERING RESULTS

We report two different sets of results. One is used to compare our proposed method with a number of baselines and the mutual information face clustering algorithm from [99] (see Figure 5.8 and Table 5.6). The second result set is used to understand how different internal choices for the QSS method affect the final face clustering result (see Figure 5.9 and Table 5.7). We compute both the B-cubed precision-recall curves (see Figure 5.8 and 5.9) and the average F-measure scores for each video and each method (Table 5.6 and Table 5.7).

Existing methods vs. quasi-static scene clustering. The B-cubed precision-recall curves in Figure 5.8a — 5.8e show us four things. First, the mutual information (MI)

Table 5.6: Face clustering results on the QSS dataset (best method for each video highlighted in bold, average F-measure and standard deviation, $\beta = 0.5$ — i.e., precision-biased). The QSSenu method always outperforms the other methods, except for the TV series video. MI is the second best method, except for the "roundtable" video. These results show that better overall face clustering performance can be achieved when information from a quasi-static scene is exploited.

method	Roundtable	TV series	News debate	Game show	Orchestra
RND	0.273 ± 0.025	0.143 ± 0.059	0.111 ± 0.020	0.116 ± 0.029	0.103 ± 0.018
Bf	0.433 ± 0.065	0.191 ± 0.089	0.232 ± 0.121	0.193 ± 0.094	0.137 ± 0.018
Bu	0.465 ± 0.085	0.232 ± 0.138	0.251 ± 0.150	0.225 ± 0.128	0.156 ± 0.025
MI	0.393 ± 0.073	**0.322 ± 0.159**	0.302 ± 0.125	0.295 ± 0.123	0.259 ± 0.102
QSSenf	**0.537 ± 0.112**	0.200 ± 0.089	0.299 ± 0.140	0.295 ± 0.181	0.259 ± 0.074
QSSenu	0.527 ± 0.107	0.258 ± 0.128	**0.319 ± 0.143**	**0.344 ± 0.187**	**0.263 ± 0.080**

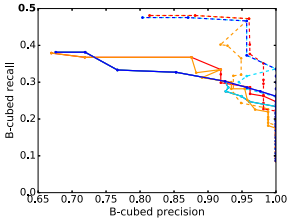
method curves are above the baseline ones (Bf, Bu) only in the "orchestra" video case. By contrast, the QSS method curves exhibit such desired pattern on 3 videos out of 5 and, in the other 2 cases, the same pattern partially occurs (in the high precision region of the plots). Second, the mutual information method (MI) slowly grows in precision, whereas both QSSenf and QSSenu start with a steep increase in recall, while keeping a precision very close to 1.0. Third, the QSS method is biased towards precision and it has limited recall. Finally, we consistently see that when a method exploits visual information beyond the face region (Bu and QSSenu), it produces better performance than the face only option (Bf and QSSenf). However, the performance further improve when information from the quasi-static scene is exploited (Bf and Bu vs. QSSenf and QSSenu).

The performance of each evaluated methods on each video are summarized in Table 5.6. The average F-measure $\beta = 0.5$ scores show that the QSSenu method always performs better than the baselines and the mutual information method with the only exception of the "TV series" video, for which it is the second best method.

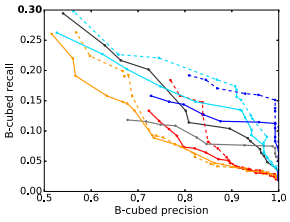
The QSS method's bias towards precision shows that we effectively exploit the combination of scene information and people visual appearance similarity. Recall is limited; however, as it will be explained in Section 5.5.3, this issue can be solved by improving the recall at the scene map generation step. The true power of our proposed method is shown in the "orchestra" video, which includes several crowded shots. In this case, the visual detail typically lacks; hence, the spatial information exploited by our method becomes essential in order to obtain accurate face clustering results.

Different options for the QSS method. We then compare how the face clustering performance change when different combinations of spatial and visual information are used at the pair-wise face matching step (see Section 5.3.4). In addition, we try to combine our method to the people co-occurrence refinement step from [107], which aims to increase the face clustering recall. We do this to verify whether our method implicitly exploits the information about recurrent groups of people.

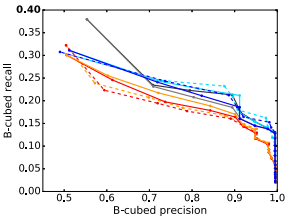
First, Figure 5.9a – 5.9e consistently show that using upper body visual features outperforms face ones (except for the "roundtable" video, where the two feature sets perform on par). When information about the relative positions between people is added (QSSnf/QSSnu vs. QSSenf/QSSenu), the face clustering performance does not always improve (see Figure 5.9c) and it can even decrease, like in the "roundtable" video case



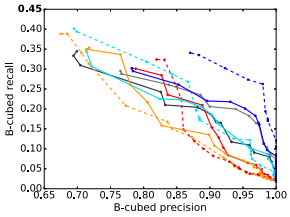
(a) “Roundtable” video.



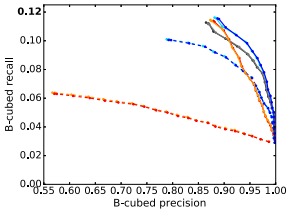
(b) “TV series” video.



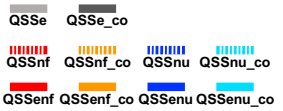
(c) “News debate” video.



(d) “Game show” video.



(e) “Orchestra” video.



(f) Legend.

Figure 5.9: Comparison between the QSS methods. QSSe uses edges similarity alone at the pair-wise face matching step. QSSnf anf QSSnu use nodes similarity alone (with face and upper body visual features respectively). QSSenf and QSSenu use both edges and nodes similarity instead. The “co” suffix is used when the people co-occurrence refinement step is applied. This figure is best seen in color.

Table 5.7: Comparing different options for the QSS method (best method for each video highlighted in bold, average F-measure and standard deviation, $\beta = 0.5$ — i.e., precision-biased). These results show how different similarity functions used at the pair-wise face matching step affect the face clustering performance. We clearly see that including edges similarity is significantly beneficial for the orchestra video only. This means that the information encoded in the relative distances and angles between people can be noisy for videos featuring a few subjects; whereas the same information becomes extremely powerful in the opposite case.

method	Roundtable	TV series	News debate	Game show	Orchestra
QSSe	0.527 ± 0.107	0.220 ± 0.096	0.321 ± 0.145	0.343 ± 0.184	0.259 ± 0.077
QSSe_co	0.525 ± 0.105	0.264 ± 0.137	0.326 ± 0.150	0.342 ± 0.179	0.260 ± 0.077
QSSnf	**0.566 ± 0.142**	0.224 ± 0.125	0.296 ± 0.137	0.277 ± 0.178	0.188 ± 0.028
QSSnf_co	0.565 ± 0.140	0.258 ± 0.134	0.299 ± 0.139	0.294 ± 0.181	0.189 ± 0.029
QSSnu	0.558 ± 0.139	0.297 ± 0.161	0.322 ± 0.147	**0.350 ± 0.211**	0.250 ± 0.068
QSSnu_co	0.556 ± 0.137	**0.309 ± 0.163**	**0.327 ± 0.152**	0.343 ± 0.193	0.251 ± 0.068
QSSenf	0.537 ± 0.112	0.200 ± 0.089	0.299 ± 0.140	0.295 ± 0.181	0.259 ± 0.074
QSSenf_co	0.537 ± 0.112	0.248 ± 0.123	0.303 ± 0.143	0.297 ± 0.180	0.261 ± 0.073
QSSenu	0.527 ± 0.107	0.258 ± 0.128	0.319 ± 0.143	0.344 ± 0.187	0.263 ± 0.080
QSSenu_co	0.525 ± 0.105	0.282 ± 0.142	0.325 ± 0.149	0.342 ± 0.182	**0.264 ± 0.080**

(see Figure 5.9a). However, we observe again that in the "orchestra" video, which is rich of crowded scenes, including the edge similarities at the pair-wise face matching step helps (see Figure 5.9e). Finally, when we also use the people co-occurrence refinement algorithm (QSSe vs. QSSe_co, QSSnf/QSSnu vs. QSSnf_co/QSSnu_co and QSSenf/QSSenu vs. QSSenf_co/QSSenu_co), we do not see any significant improvement. This is particularly true for the "orchestra" video, where the five B-cubed precision-recall curve pairs overlap.

The F-measure scores reported in Table 5.7 also show that edges similarity at the pair-wise face matching step is only beneficial in the "orchestra" video case. For the same video, we also see that when visual information is ignored (QSSe), the performance are already close to the best options (QSSenu and QSSenu_co).

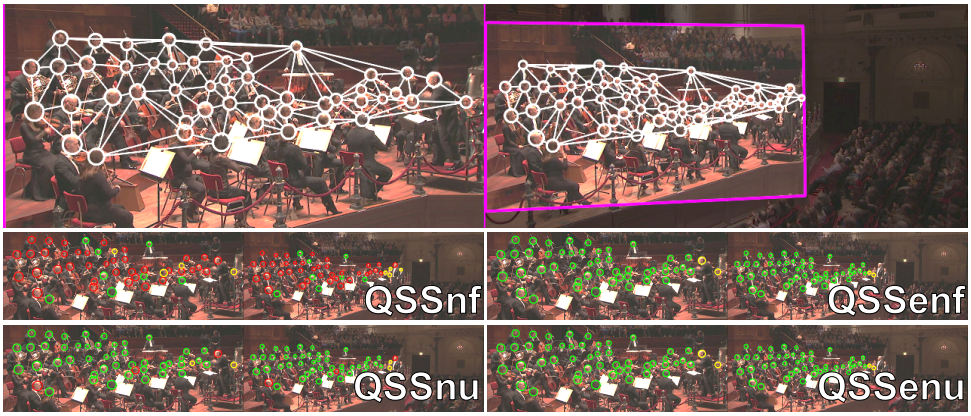
In our method we exploit the following two types of spatial information: one derived by the automatically built scene map, and one by the relative positions between people. From the results presented above, we infer that the former is already sufficient to find good face matches (QSSnf and QSSnu). It also means that edges similarity, which encodes relative distances and angles between people, can be noisy for videos featuring a few subjects; whereas the same information becomes extremely powerful in the opposite case. Finally, the people co-occurrence refinement results prove that the QSS method fully embeds information about recurrent groups of people. Given these conclusions, we recommend that the QSSnu method is used when a few subjects are filmed, whereas the QSSenu method should be used in videos that include crowded shots.

Intermediate results. We also show the behavior of our method by reporting examples of intermediate results (see Figure 5.10 – 5.11). Figure 5.10a shows that good matches are found on overview shots depicting large groups of people whose images lack of visual detail. Figure 5.10b shows correct matches in presence of cluttering. Correct matches are also found for a musician whose face is occluded by an instrument and in the second image by another person (Figure 5.10c). Matching across different camera angles and zoom level is demonstrated in Figure 5.10d and Figure 5.11b. TV programs may include visual effects, like split screens (see Figure 5.11a and Figure 5.11b), and the scene may change throughout the timeline (see Figure 5.11c). In these case, by estimating good regions of overlap, correct face matches can be still found. Finally, in Figure 5.11d and Figure 5.11e, we observe that our method copes with head pose change (including faces filmed from the back) and people that move in place.

5.5.2. ROBUSTNESS ANALYSIS

Since our method relies on spatial relationships that are derived by the available face annotations, one may wonder how errors at the face detection step affect the final video face clustering performance. Two types of error can occur at this step. *False positives* are regions of an image not corresponding to a face; *false negatives* occur instead when a face is missed. Since face detectors are typically tuned towards high-precision [13], we focus on the most frequent type of error that is the false negative.

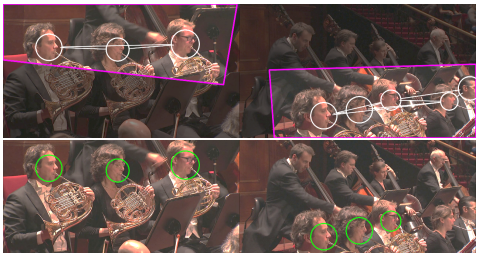
As shown in the examples of Figure 5.12, false negatives may significantly change the temporary graphs generated in our approach (see Section 5.3.4) leading to poor sub-graph matching accuracy. In the first row (figs. 5.12a and 5.12b), we show the temporary graphs and the sub-graph matching result in the ideal case of perfect face detection.



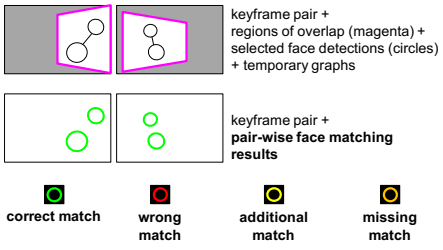
(a) The orchestra video includes overview shots in which people's images lack of visual detail. However, our approach correctly leverages the rich information about the spatial configuration.

(b) Cluttered scenes.

(c) Occlusions and different actions.



(d) Different viewpoints.



(e) Legend.

Figure 5.10: Examples of intermediate results of our proposed method ("orchestra video"). The QSS face clustering algorithm successfully deals with several real world data challenges: lack of visual detail (which usually occurs in crowded scenes), cluttering, occlusions, people appearance variation because of interaction with objects and performed actions, different viewpoints and head poses (including faces filmed from the back). This figure is best seen in color.

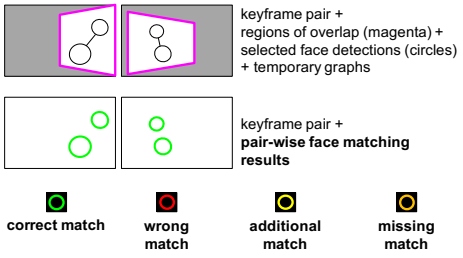
(a) Split screens.

(b) Relative positions change.

(c) Changes in the scene.

(d) Faces from the back.

(e) People moving in place.



(f) Legend.

Figure 5.11: Examples of intermediate results of our proposed method (QSS-YouTube subset). Our method also works well with edited videos which may include, for instance, split screens (see Figure 5.11a). Changes in the scene can be also tolerated (see the background change in Figure 5.11c). If relative positions between people often change, the QSSnu option can be used in place of of QSSenu (see Figure 5.11b); however, if the quasi-static scene assumptions hold, the QSSenu method tolerates people moving in place (see Figure 5.11e). The full set of augmented keyframe pairs can be downloaded from `https://goo.gl/zxRmFA`. This figure is best seen in color.

The other two rows instead show the results when 20% (figs. 5.12c and 5.12d) and 50% (figs. 5.12e and 5.12f) of the faces are randomly discarded. We observe that, as the number of discarded faces increases, the number of correct matches decreases.

In this experiment, we consider different percentages of face annotations to discard (namely 10%, 20%, 30%, 40% and 50%). For each percentage and each keyframe of a video, we generate a random subset of face annotations. We then use the QSSenu method and measure its performance as a function of the percentage of discarded faces. We repeat this process 3 times and average the B-cubed precision-recall scores since discarding faces is a randomized process. The results for each video are reported in fig. 5.13.

The plots in Figure 5.13 show that for every video, the performance drops when missing faces occur. We also observe that the gap between the ideal case (blue curves) and the worst case (discarding rate set to 50%, red curves) is larger for those videos that include a few unique identities (e.g., compare Figure 5.13b to Figure 5.13e). This can be explained by the fact that "crowded" videos contain richer spatial information that makes our method more robust to face detection false negatives.

5.5.3. FAILURE ANALYSIS

When analyzing how our method fails, we found three main factors affecting the accuracy of the produced face clusters.

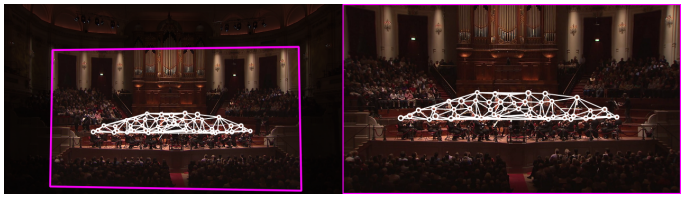
Inexact regions of overlap lower the precision. Sometimes a wrong region of overlap (Section 5.3.3) is estimated and it leads to wrong matches at the pair-wise face matching step (see Figure 5.14). While this seldom happens, it can significantly lower the precision since a single mistake can erroneously link two connected components in G_F that belong to two different identities.

A sparse scene map lowers the recall. At the scene map generation step, good keyframe pairs are always identified. However, several pairs with common identities are missing. More in details, we noticed that pairs only differing in zoom level are correctly found, whereas keyframes taken from viewpoints with large baseline (see Figure 5.15) are not. The lack of intermediate viewpoints leads to missing connections between such pairs. This results in a sparse scene map — i.e., the graph G_I has several connected components — which lowers the overall recall of our method.

On-screen text regions deteriorate the scene map and the region of overlap estimations. By inspecting the intermediate results for the news debate video, we found that several regions of overlap were wrong. We hypothesized that the presence of on-screen text regions affects their estimation, hence we employed an off-the-shelf text detector [71] to mark the extracted keypoints as outliers if detected in a text region. By comparing the face clustering results using all the detected keypoints and then only the inliers, we see that such an inexpensive keyframe pre-processing step leads to increased performance (see Figure 5.16b).

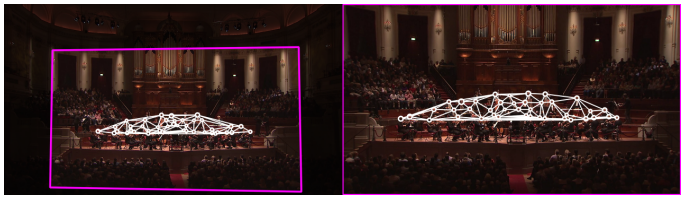
5.5.4. SCALABILITY

In this chapter, we focus on *off-line* video face clustering algorithms; hence, a discussion on efficiency may be less relevant than that of Section 5.5.1 about accuracy. However, the evaluated methods have some fundamental design differences affecting scalability and application scope.



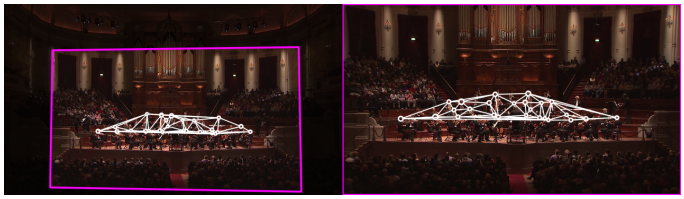
(a) Correct face annotations (temporary graphs).

(b) Correct face annotations (sub-graph matching).



(c) Discarded annotations: 20% (temporary graphs).

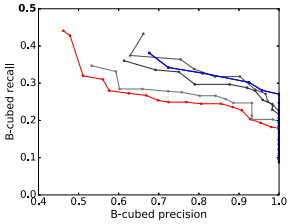
(d) Discarded annotations: 20% (sub-graph matching).



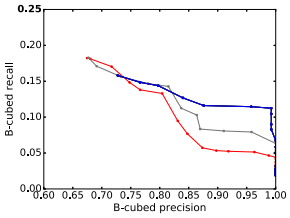
(e) Discarded annotations: 50% (temporary graphs).

(f) Discarded annotations: 50% (sub-graph matching).

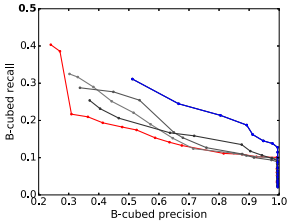
Figure 5.12: False positives at the face detection step reduce the sub-graph matching performance (correct and mistaken matches noted as green and yellow/red circles respectively). Above, we report three different results on the same pair of matching keyframes. As the number of discarded faces increases, the number of correct matches decreases. This figure is best seen in color.



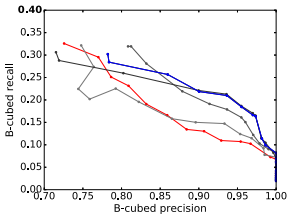
(a) “Roundtable” video.



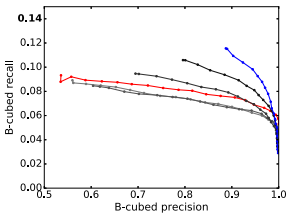
(b) “TV series” video.



(c) “News debate” video.



(d) “Game show” video.



(e) “Orchestra” video.

(f) Different discarding rates.

Figure 5.13: Performance of the “QSSenu” method (edge and node similarity, upper body features) as a function of the percentage of discarded face detections (i.e., false negatives). The blue curves indicate the performance when all the annotations are used, while the red ones correspond to 50% of discarding rate. Some curves overlap because the number of subject of a video are too few. These results show that our method suffers from poor face annotation performance, especially when a video includes a few unique identities (e.g., “TV series”). By contrast, when a video records larger groups, like in the “orchestra” video, our method is more robust to missing face detections. This figure is best seen in color.

Figure 5.14: Inexact estimations of the regions of overlap may cause pair-wise face matching errors.

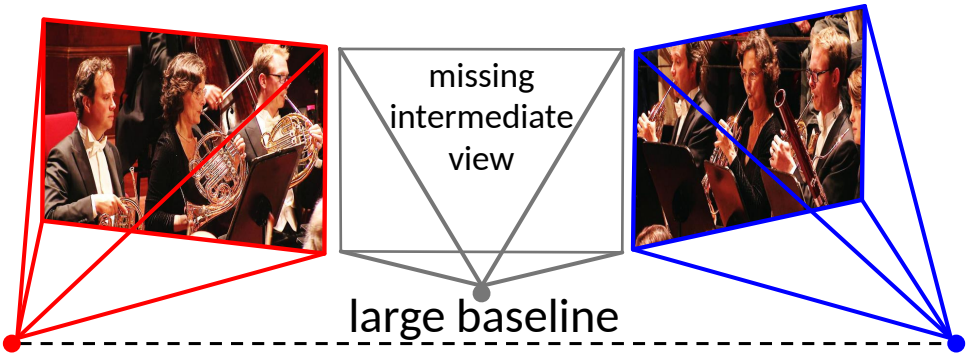
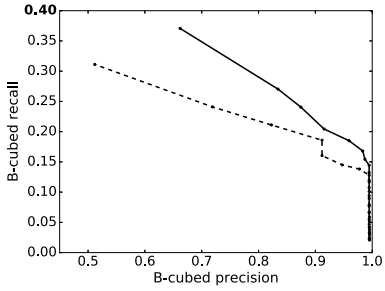


Figure 5.15: Views of the same group of people with large baseline do not connect if there are no intermediate viewpoints.

(a) We filtered out the keypoints falling in on-screen text regions.



(b) When text regions keypoints are filtered (solid line), better face clustering performance are achieved.

Figure 5.16: On-screen text regions affect the scene map generation worsening the face clustering performance. A simple and inexpensive text removal pre-processing step reduces such side-effect.

As already mentioned in Section 5.4.4, a face clustering algorithm can be designed to derive either one label per face, like done in [99], or one label per face track (e.g., our baselines and our proposed method). There are also hybrid approaches using majority voting [102]. Second, a method may require that a full (dis-)similarity matrix is computed, and such matrix can contain either face [99] or face track pair-wise scores. Similarly, constraints can be also encoded as full pair-wise matrices [102].

As shown in Table 5.2, the magnitude of the number of face tracks is less than that of the face track lengths (even if the face tracks have been shortened). Hence, methods that only need to compute face tracks similarities are more efficient than those based on faces similarity scores. One way to reduce the number of face pair comparisons is exploiting must-link and cannot-link constraints automatically derived by the face tracks (like done in [99]). However, thanks to the quasi-static assumptions, our method further reduces the number of computed face track pair scores. This is possible since the scene map and the regions of overlap (see Section 5.3.2 and 5.3.3) tell us which face track pairs are good candidates.

Also note that the authors in [99] originally propose to use **all** the frames to extract rich visual information out of each face track. While this helps a spectral clustering algorithm to generate good clusters, it makes the approach infeasible for videos that are long and/or include several subjects (as in the case of our "orchestra" video).

5.6. CONCLUSIONS

We presented a video face clustering method for quasi-static scene videos, where people maintain the same positions on the scene, but still display movement at their location. In this case, the spatial relationships between people can be used as contextual information to reliably match the detected faces, especially in crowded scenes which typically lack of visual detail.

We compared spatial information to other known sources (e.g., clothing, people co-occurrence patterns, face track constraints). Our results show that the proposed approach effectively and efficiently takes advantage of the quasi-static scenes properties and, when ideal face detection output is used, face clusters are formed with very high precision. By simulating errors at the face detection step, we learned that the performance of our method decreases when a detector misses faces. However, in the challenging case of videos featuring dozens of different people, the drop is not as large as when a video only features a few subjects. Finally, we learned that our method can still be effective when the quasi-static scenes requirements are relaxed, and people change relative positions over time.

These outcomes suggest that our method is well-suited for existing annotation frameworks like [108], in which a human-in-the-loop process is used to obtain high quality output. We therefore plan future research directed towards integrating the QSS algorithm in real-life semi-automatic annotation frameworks, evaluating to what extent it will make the computer-aided annotation more efficient. While the dependency on ideal face detection may limit the applicability of our method to a fully automatic system, we showed that spatial information extracted from the scene and the relative positions between the filmed people is the missing piece for efficient and effective context-based video face clustering.

6

VISION-BASED DETECTION OF ACOUSTIC TIMED EVENTS: A CASE STUDY ON CLARINET NOTE ONSETS

This chapter continues the type of discussion that we started in Chapter 2, but now following a supervised approach. Also, differently from playing/non-playing labels, we consider musically relevant information at a fine temporal resolution. Namely, we switch to note onsets detection and narrow down to the woodwind and brass instruments which, differently from other instruments, require the analysis of finger movements. To this end, we devise a novel method based on deep learning and on multiple regions of interest to be tracked, which makes the method suitable for handling moving cameras. More specifically, we propose a 3D convolutional neural network based on multiple streams and purposely avoiding temporal pooling. We release a fully annotated audiovisual dataset with 4.5 hours of clarinetist videos including about 36,000 onsets, and carry out preliminary experiments showing that our proposed classifier currently performs only on par with a ground-truth informed random baseline. While the gap between vision-based and audio-only note onsets detection (performed on isolated audio tracks) is rather large, this work paves the way to solving the more general problem of acoustic timed events detection.

This chapter was published as: Alessio Bazzica, Jan C. van Gemert, Cynthia C. S. Liem, Alan Hanjalic. Vision-based Detection of Acoustic Timed Events: a Case Study on Clarinet Note Onsets *International Workshop on Deep Learning for Music, in conjunction with the International Joint Conference on Neural Networks* (2017).

Acoustic timed events take place when persons or objects make sound, e.g., when someone speaks or a musician plays a note. Frequently, such events also are visible: a speaker's lips move, and a guitar cord is plucked. Using visual information we can link sounds to items or people and can distinguish between sources when multiple acoustic events have different origins. We then can also interpret our environment in smarter ways: e.g., identifying the current speaker and indicating which instruments are playing in an ensemble performance.

Understanding scenes through sound and vision has both a *multimodal* and a *cross-modal* nature. The former allows us to recognize events using auditory and visual stimuli jointly. But when e.g., observing a door bell button being pushed, we can cross-modally infer that a bell should ring. In this chapter, we focus on the cross-modal case to detect acoustic timed events from video. Through visual segmentation, we can spatially isolate and analyze sound-making sources at the individual player level, which is much harder in the audio domain [13].

As a case study, we tackle the musical note onset detection problem by analyzing clarinetist videos. Our interest in this problem is motivated by the difficulty of detecting onsets in audio recordings of large (symphonic) ensembles. Even for multi-track recordings, microphones will also capture sound from nearby instruments, making it hard to correctly link onsets to the correct instrumental part using audio alone. Knowing where note onsets are and to which part they belong is useful for solving several real-world applications, like audio-to-score alignment, informed source separation, and automatic music transcription.

Recent work on cross-modal lip reading recognition [21] shows the benefit of exploiting video for a task that has traditionally been solved only using audio. In [57], note onset matches between a synchronized score and a video are used to automatically link audio tracks and musicians appearing in a video. The authors show a strong correlation between visual and audio onsets for bow strokes. However, while this type of visual onset is suitable for strings, it does not correlate well to wind instruments. In our work we make an important step towards visual onset detection in realistic multi-instrument settings focusing on visual information from clarinets, which has sound producing interactions (blowing, triggering valves, opening/closing holes) representative for wind instruments in general.

Our contributions are as follows: (i) defining the visual onset detection problem, (ii) building a novel 3D convolutional neural network (CNN) [92] without temporal pooling and with dedicated streams for several regions of interest (ROIs), (iii) introducing a novel audiovisual dataset of 4.5 hours with about 36k annotated events, and (iv) assessing the current gap between vision-based and audio-based onset detection performance.

6.1. RELATED WORK

When a single instrument is recorded in isolation, audio onset detectors can be used. A popular choice is [80], which is based on learning time-frequency filters through a CNN applied to the spectrogram of a single-instrument recording. While state-of-the-art performance is near-perfect, audio-only onset detectors are not trained to handle multiple-instrument cases. To the best of our knowledge, such cases also have not been tackled so far.

A multimodal approach [11] spots independent audio sources, isolate their sounds and is validated on four audiovisual sequences with two independent sources. As the authors state [11], their multimodal strategy is not applicable in crowded scenes with frequent audio onsets. Therefore, it is not suitable when multiple instruments mix down into a single audio track.

A cross-modal approach [18] uses vision to retrieve guitarist fingering gestures. An audiovisual dataset for drum track transcription is presented in [41] and [27] addresses audiovisual multi-pitch analysis for string ensembles. All works devise specific visual analysis methods for each type of instrument, but do not consider transcription or onset detection for clarinets.

Action recognition aims to understand events. Solutions based on 3D convolutions [92] use frame sequences to learn spatio-temporal filters, whereas two-streams networks [35] add a temporal optical flow stream. A recurrent network [29] uses LSTM units on top of 2D convolutional networks. While action recognition is similar to visual-based acoustic timed events detection, there is a fundamental difference: action recognition aims to detect the presence or absence of an action in a video. Instead, we are interested in the exact temporal location of the onset.

In action localization [67] the task is to find what, when, and where an action happens. This is modeled with a "spatio-temporal tube": a list of bounding-boxes over frames. Instead, we are not interested in the spatial location; we aim for the temporal location only, which due to the high-speed nature of onsets reverts to the extreme case of a single temporal point.

6.2. PROPOSED BASELINE METHOD

Together with our dataset, we offer a baseline model for onset detection. The input for our model is a set of sequences generated by tracking a number of oriented ROIs from a video of a single clarinetist (see Figure 6.1). For now, as a baseline, we assume that in case of a multi-player ensemble, segmentation of individual players already took place. The ROIs consider those areas in which the sound producing interactions take place: mouth, left/right hands, and clarinet tip, since they are related to blowing, fingering, and lever movements respectively.

Each sequence is labeled by determining if a note has started during the time span of the *reference frame*. A sequence consists of 5 preceding frames, the reference frame, and 3 succeeding frames, forming a sequence of 9 consecutive frames per ROI. We use a shorter future temporal context because the detector may otherwise get confused by *anticipation* (getting ready for the next note). Examples of onset and not-an-onset inputs are shown in Figure 6.2.

Our model relies on multiple *streams*, one for each ROI. Each stream consists of 5 convolutional layers (CONV1-5), with a fully-connected layer on top (FC1). All the FC1 layers are concatenated and linked to a global fully-connected layer (FC2). All the layers use ReLU units. The output consists of two units ("not-an-onset" and "onset"). Figure 6.3 illustrates our model and, for simplicity, it only shows one stream for the left hand and one for the right one.

To achieve the highest possible temporal resolution, we do not use temporal pooling. We use spatial pooling and padding parameters to achieve slow fusion throughout the 5

Figure 6.1: Raw video frames example.

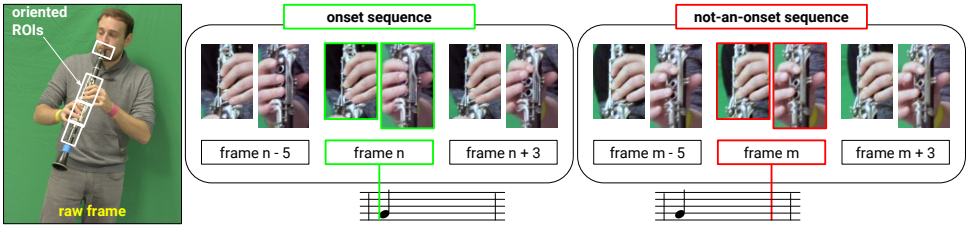


Figure 6.2: Onset and not-an-onset input sequence examples with 2 ROIs from 3 frames.

convolutional layers. We aim to improve convergence and achieve regularization using batch normalization (BN) [47], L2 regularization and dropout. Since we use BN, we omit the bias terms in every layer including the output layer.

We use weighted cross-entropy as loss function to deal with the unbalanced labels (on average, one onset every 15 samples). The loss is minimized using the RMSprop algorithm. While training, we shuffle and balance the mini-batches. Each mini-batch has 24 samples, half of which are not-an-onset ones, 25% onsets and 25% *near-onsets*, where a near-onset is a sample adjacent to an onset. Near-onset targets are set to $(0.75, 0.25)$, i.e., the non-onset probability is 0.75. In this way, a near-onset predicted as onset is penalized less than a false positive. We also use data augmentation (DA) by randomly cropping each ROI from each sequence. By combining DA and balancing, we obtain epochs with about 450,000 samples. Finally, we manually use early-stopping to select the check-point to be evaluated (max. 15 epochs).

6.3. EXPERIMENTAL TESTBED: CLARINETISTS FOR SCIENCE DATASET

We acquired and annotated the new *Clarinetists for Science* (C4S) dataset, released with this chapter.[1] C4S consists of 54 videos from 9 distinct clarinetists, each performing

[1] For details, examples, and downloading see `http://mmc.tudelft.nl/users/alessio-bazzica#C4S-dataset`

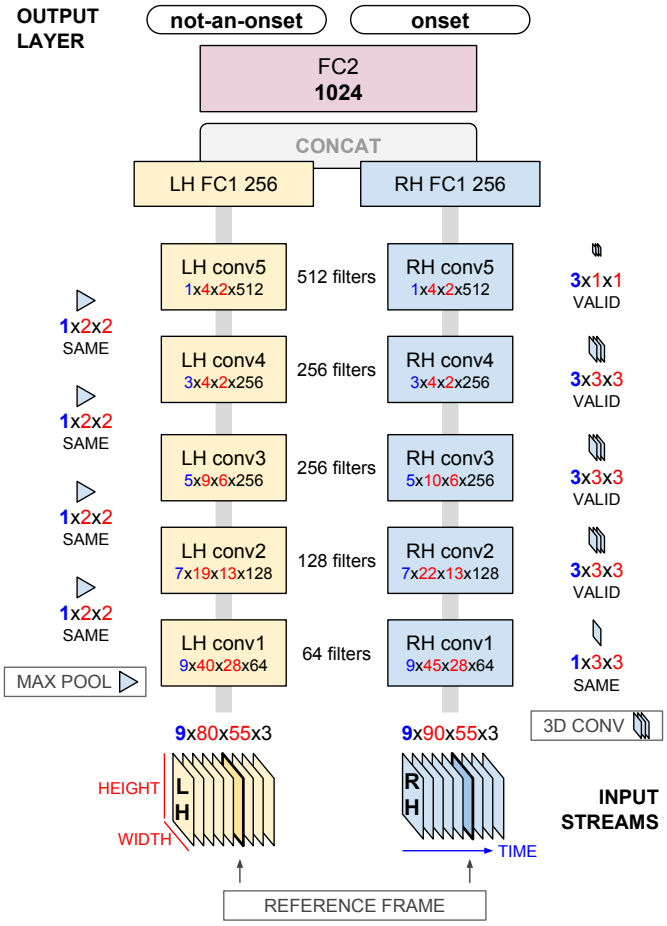


Figure 6.3: Proposed model based on 3D CNNs, slow fusion, and multiple streams (one for each ROI). LH and RH indicate the left and right hand streams respectively.

method	Split 1	Split 2	Average
informed random baseline	27.4	19.6	23.5
audio-only SuperFlux [16]	82.8	81.3	82.1
audio-only CNN [80]	94.3	92.1	93.2
visual-based (proposed)	26.3	25.0	25.7

Table 6.1: F-scores with a temporal tolerance of 50 ms.

3 different classical music pieces twice (4.5h in total). The videos have been recorded at 30 fps, about 36,000 events have been semi-automatically annotated and thoroughly checked. We used a colored marker on the clarinet to facilitate visual annotation, and a green screen to allow for background augmentation in future work. Besides ground-truth onsets, we include coordinates for face landmarks and 4 ROIs: mouth, left hand, right hand, and clarinet tip.

In our experiments, we use leave-one-subject-out cross validation to validate the generalization power across different musicians (9 splits in total). From each split, we derive the training, validation and test sets from 7, 1, and 1 musicians respectively. Hyper-parameters, like decaying learning rate and L2 regularization factors, are manually adjusted looking at f-scores and loss for train and validation sets. We compute the f-scores using 50 ms as temporal tolerance to accept a predicted onset as true positive. We compare to a ground-truth informed random baseline (correct number of onsets known) and to two state-of-the-art audio-only onset detectors (namely, SuperFlux [16] and CNN-based [80]).

6.4. RESULTS AND DISCUSSION

During our preliminary experiments, most of the training trials were used to select optimization algorithm and suitable hyper-parameters. Initially, gradients were vanishing, most of the neurons were inactive, and networks were only learning bias terms. After finding hyper-parameters overcoming the aforementioned issues, we trained our model on 2 splits.

By inspecting the f-scores in Table 6.1, we see that our method only performs slightly better than the baseline, and that the gap between audio-only and visual-based methods is large (60% on average). We investigated why and found that throughout the training, precision and recall often oscillate with a negative correlation. This means that our model struggles with jointly optimizing precision and recall. This issue could be alleviated by different near-onsets options or by formulating a regression problem instead of a binary classification one.

When we train on other splits, we observe initial f-scores not changing throughout the epochs. We also observe different speeds at which the loss function converges. The different behaviors across the splits may indicate that alternative initialization strategies should be considered and that the hyper-parameters are split-dependent.

6.5. CONCLUSIONS

We have presented a novel cross-modal way to solve note onset detection visually. In our preliminary experiments, we faced several challenges and learned that our model is highly sensitive to the optimization algorithm and hyper-parameters. Also, using a binary classification approach may prevent the joint optimization of precision and recall. To allow further research, we release our novel fully-annotated C4S dataset. Beyond visual onset detection, C4S data will also be useful for clarinet tracking, body pose estimation, and ancillary movement analysis.

ACKNOWLEDGMENTS

We thank the C4S clarinetists, Bochen Li, Sara Cazzanelli, Marijke Schaap, Ruud de Jong, dr. Michael Riegler and the SURFsara Dutch National Cluster team for their support in enabling the experiments of this chapter.

7

REFLECTIONS AND RECOMMENDATIONS

The aim of this thesis is to contribute to the continuous innovation of digital music platforms, which have become the primary means for consuming music. As explained in Chapter 1, users can benefit from music recordings that are more informative, accessible in a non-linear fashion and from multiple perspectives [43] [66]. One way of pursuing this objective is looking broader than what audio-only Music Information Retrieval (Music IR) offers. In particular, a way to open up to new applications, and innovate the existing ones, is exploiting the visual channel of a recording as additional source of information.

The main challenge we see when considering visual data for Music IR is the fact that music videos are often *unstructured* and *unconstrained*—i.e., the *real-world data* case. We also believe that, if we do not design from scratch algorithms that address these challenges, we might end up having solutions that will never be used in practice. In this thesis, we therefore expanded the state-of-the-art following two approaches. One approach focused on videos depicting a single musician and aimed to extract musically relevant information for any type of instrument and independently by the recording conditions. The second approach considered instead the case of large ensembles, which requires ways of understanding the scene in order to segment out information belonging to each distinct musician. To this end, we experimented on symphonic orchestra videos, since we found them to be a good representative of the typical real-world data challenges in vision-based Music IR.

We presented our insights and achievements in the five technical core chapters of this thesis. The works in Chapter 2 and Chapter 6 belong to the first approach — i.e., analysis of single musician videos. Chapter 2 described a lightweight feature named *Motion Orientation Histograms* (MOHs), which aims to represent musicians' movements over time. By carrying out an unsupervised analysis experiment on jam session recordings, we showed that MOHs novelty curve peaks correlate well with the structural boundaries of the analyzed piece, even when timbre and instrumentation do not vary much

throughout the piece—a typical case in which audio-only analysis may not be sufficient to reveal the structure of a piece. MOHs are suitable for automatic segmentation of a music piece, potentially allowing non-linear access scenarios (e.g., jump to different sections like intro, solo and chorus). While lightweight and instrument-independent, further investigations have shown that MOHs are unfortunately not robust to camera motion and only focus on global motion aspects. Taking these limitations into account, we moved to a more complex approach, which involves tracking of regions of interest (ROIs) and supervised analysis using a novel deep learning method (Chapter 6) based on purposely removing temporal pooling and on combining multiple convolutional streams (one for each ROI). Differently from MOHs, we focused on a specific instrument family, since different instruments involve different sound producing movements (e.g., plucking, bowing, finger movements) that require a specific solution. More specifically, we aimed to extract note onsets from woodwind and brass player videos, focusing on clarinetist recordings as a case study. We created and publicly shared a new dataset, which has been designed to allow increasingly complex experiments. For instance, the uniform background can be replaced with static images or even with a background video, in order to test the robustness in presence of clutter. Occlusions can be simulated adding virtual (moving) occluders. Markers have been used to extract the ROI positions ground-truth and hence allow the training of human body parts detectors. The results reported in Chapter 6 showed that the proposed method only works on par with a ground-truth informed baseline. However, we are confident that further research can lead to improved performance, which would be relevant since traditional Music IR tasks like sound source separation, automatic music transcription, and performance-to-score synchronization can benefit from knowing when note onsets occur for each individual performing musician.

The second approach followed in this thesis was to consider videos of large ensembles and, to the best of our knowledge, we were the first to address this challenging case. Our main goal was to build a framework that extracts musician-wise annotations from unconstrained videos, spanning multiple moving cameras and a single static one. Such a framework must spot musicians across shots, allow to easily link their identities to the instrumental part they perform, and segment the scenes to allow further analysis (e.g., note onsets detection, detecting active musicians). Before addressing this challenging problem, we first investigated if extracting musician-wise annotations is even worth pursuing. In Chapter 3, we presented an experiment conducted on synthetic data to assess whether sequences of playing/non-playing (P/NP) labels extracted for each performed instrumental part can be used to synchronize a video to a symbolic score. The results showed the power of aggregating annotations over time and across musicians: in fact, we showed that if P/NP sequences are extracted from a performance recording, a coarse temporal alignment can be computed, even in presence of missing or mistaken P/NP labels. Also, we realized that P/NP annotations, regardless of how they are derived (e.g., a synchronized score, a video, or even from isolated audio recordings), can be exploited for non-linear access scenarios, for instance allowing users to easily skip to the next “tutti” or “solo” section. Then, we moved forward proposing a possible framework for musician-wise P/NP detection. In Chapter 4, we first described the characteristics of symphonic video recordings and then presented a semi-automatic annotation sys-

tem that effectively and efficiently combines automatic algorithms and human annotation. More specifically, we presented a possible way to segment the scene by detecting faces and estimating the head poses to extract the corresponding upper body bounding boxes. We studied different state-of-the-art face clustering methods, and then used a manual labeling step to link face clusters to musician identity labels and discard clusters linked to audience and conductor face images. Due to the lack of datasets and the complexity of the data we treated, we used image clustering and again human annotation to efficiently perform P/NP labeling. By simulating the human annotation process, we conducted several experiments to identify the open challenges and the limitations of the proposed method. We learned that by following such a modular system design, it is important to reduce errors in the first steps as much as possible. For this reason, we recommended that both video face detection and clustering must be improved to avoid poor P/NP labeling performance. We therefore narrowed down and in Chapter 5 we presented a method that, by exploiting the *quasi-static scene* (QSS) properties of symphonic orchestra videos, generates face clusters with high precision and more efficiently than the competing methods. Namely, we proposed a novel face clustering approach that benefits from (quasi-)stationary spatial relationships between people. Our method exploits an automatically built scene map, through which overlapping views are selected for a sub-graph matching step that aims to link face tracks across shots. Note that this approach avoids quadratic complexity, typical of existing face clustering solutions, since there is no need to compare all the face track pairs of a video. Finally, we also presented a simple graph pruning strategy, with which one can tune the algorithm towards either face clusters purity or a reduced number of face clusters. By integrating this method in the framework of Chapter 4, the face clustering errors can be reduced making the face labeling step more efficient; we therefore claim that the QSS algorithm can make the musician-wise annotation more accurate. Also, by exploiting the QSS method alone to label musicians' identities, we can already deploy relevant Music IR applications. For instance, the viewpoint can be switched automatically following (and zooming on) a musician selected by the user. Even more interestingly, the musician identity can be linked to an isolated audio track which can be automatically played when a user taps on a musician's face appearing in the video.

In summary, in this thesis we strove to find vision-based Music IR methods that are possibly applicable to any musical instrument and any type of video recordings. Our outcomes show the benefit of tackling the challenges of real-world data: as opposed to oversimplified data, unstructured and unconstrained music videos reveal concrete opportunities to deploy new applications on digital music platforms. Still, the vision-based Music IR field remains a largely unexplored territory. In the next section, we share what is our view on the next steps to be taken.

7.1. THE NEXT STEPS

Throughout the chapters, we have learned that the problem of enriching the user experience on digital music platforms by exploiting vision-based methods must be attacked from different perspectives. Namely, we should think of the general design of a *musician-wise annotation framework* in order to find the best way to link annotations of a single musician to the identity label and/or to the corresponding instrumental part name.

More specifically, we should investigate more the problem of *parsing the scene* by addressing face detection, scene segmentation, and face clustering. In parallel, we should also discover which are the *visual features and annotations* that allow us to solve Music IR tasks and that can be reliably extracted from the visual channel of a single musician's recording.

7.1.1. MUSICIAN-WISE ANNOTATION FRAMEWORK

In Chapter 4, we proposed a possible design for a musician-wise annotation framework. We pursued a semi-automatic strategy that combines face detection, face clustering and manual image clusters labeling. In this way, we reduced the number of errors that propagate throughout the pipeline in an uncontrolled manner, showing that these errors lead to inaccuracies which become hard to fix. This way of proceeding is not the only possible option. For instance, musicians can be detected in different ways, e.g., by combining upper body and face detectors. Face tracks can be extracted independently or jointly. The face clustering algorithm can be tuned either towards precision or recall. Then, regarding the human intervention, we can investigate further where and how it should be integrated in the system. Also, we should study how to possibly perform the correction tasks. How to instruct human annotators? How to optimally design the user interface for the correction task?

Another direction that is worth investigating is the design of fully-automatic annotation systems, which are appealing since they scale well on large music collections. Currently, this type of solutions suffers from two major problems. First, they can only perform on a best-effort basis, hence leading to an overall accuracy that may not be sufficient for commercial applications. Second, data-driven approaches, which are typically exploited in fully-automatic systems, are known to require large amounts of annotated data to perform well on unseen samples. Addressing these issues is relevant, and may lead to better algorithms that can be incorporated in semi-automatic systems as well. This requires that more multimodal music datasets are created and publicly shared, and that the new datasets are as rich as possible—both in terms of annotations and also of recording conditions (e.g., varying camera angles, allowing background replacement).

Finally, note that improving semi-automatic frameworks is also beneficial to improve fully-automatic ones. In fact, manual verification (and correction) of automatic output is applied in several commercial products and it is used to extend datasets. For instance, a popular online translation tool allows users to correct and suggest better translations. Similarly, social network users can correct the automatic face detection output on the pictures they upload.

7.1.2. PARSING THE SCENE

Unstructured music videos require that musicians are found, recognized and tracked. Additional annotations may be needed, for instance to be able to link the identity of a musician to the instrumental part she performs. People that should be excluded from the musician-wise annotation step, like the audience, should be recognized as such. Finally, the extracted musician face tracks should be used to segment out a clip to be used for further analysis. The overall problem remains an open challenge in the computer vision field: it would still be very hard to address it even using recent deep learning meth-

ods.

The first problem we experienced in the context of this challenge is at the musician detection step. The video face detectors we used suffer of limited recall, especially with extreme head poses. This directly affects the *timeline coverage* of the annotation process—i.e., the annotations are only available for a small portion of the recording. Also, for some applications, it can make sense to also find musicians appearing from the back, for which even the most sophisticated face detector would not work. A logical alternative are then human body detectors; however, when we tested them, we found them computationally expensive and performing even worse than face detectors. Our guess is that such detectors cannot cope with classical music players, because musicians typically dress in black and their instruments partially occlude the upper body regions of the video frames. Another option to improve the video face detection recall is to exploit a scene map, like that computed in our QSS algorithm. The map can then be used to infer where a face is expected to be present in a frame given the annotations that are found on other visually overlapping shots.

The second problem that requires further investigation is the scene segmentation. In Chapter 4 we presented a heuristic which calculates size and position of rectangular bounding boxes using the estimated head pose angles. Alternatively, if the played instrument is first recognized, one could adapt the bounding box to only cover relevant areas for that specific musician. Another option is using body pose detectors; however, as explained above, existing ones may not perform sufficiently well on the classical music videos. Finally, our QSS video face clustering can be further improved. For instance, overlapping views with large camera baselines are usually not detected and the lower recall in the computed scene map directly affects the face clustering recall.

7.1.3. VISUAL FEATURES AND ANNOTATIONS

Once the scene is segmented, the annotation framework must analyze multiple regions of the video that belong to each detected musician. The goal at this step is extracting features and annotations that are musically relevant. Throughout this thesis, we have seen two major types of annotations we can extract (low-level and semantic ones), each of which may have positive characteristics and limitations.

In Chapter 2, an instrument-generic and low-level descriptor was presented (namely, MOHs). The main difficulty when thinking of this type of descriptors is that it is hard to make them invariant across viewpoints and at different spatial scales. We also learned that capturing fine spatio-temporal details that matter is challenging, like in the case of fast finger movements in small regions of interest, which are not always sufficiently detailed, especially in crowded overview shots. Still, we find instrument-generic low-level descriptors relevant, since they have shown to correlate well with expressive cues, enabling novel automatic music transcription scenarios (e.g., transcription of accents and phrasing).

To address the aforementioned limitations, we moved to semantic descriptors, as done with P/NP labels in Chapter 3 (instrument-generic) and note onsets in Chapter 6. Identifying other annotations that have a counterpart in (symbolic) music scores has shown to be relevant, since it can potentially improve existing performance-to-score synchronization, sound source separation and automatic music transcription algorithms.

One factor that may limit the identification of relevant visual features and annotations is the fact that *transfer learning* approaches can hardly be used, especially when fine temporal resolution is required (e.g., note onsets detection). Transfer learning would allow us to exploit pre-trained classifiers, hence coping with the lack of annotated datasets for vision-based Music IR. For instance, one could investigate ways to reuse (part of) a deep network trained on large action recognition datasets. Typically, this requires to only (re-)train a few layers—which is faster than training from scratch, since there is no need to learn new low- and mid-level features. Unfortunately, transfer learning requires that the input data shares compatible characteristics across tasks, but this is not the case for popular computer vision and vision-based Music IR problems. In fact, in the former case several methods rely on temporal pooling and reduced video frame rates, which would be harmful for musician videos since the movements performed by musicians have a fine spatio-temporal structure (as discussed in Chapter 6). For these reasons, finding ways to exploit other datasets and other vision tasks is another relevant direction to investigate.

7.2. CONCLUSION

In this chapter, we have summarized contributions and findings reported in this thesis. The unstructured and unconstrained nature of the available music video collections pushed us to think more of how users can benefit from such a resource, which is still a largely unexploited asset. As shown by the applications we envisioned, analysis of music videos can enable novel applications relevant to the digital music market. We also highlighted the limitations of the approaches we proposed, and we suggested the next steps for vision-based Music IR. Our proposed research agenda is rich, and it requires that several multi-disciplinary experiments are carried out. We are confident that the light that this thesis has shed on the topic will help both Computer Vision and Music IR researchers to advance the state-of-the-art of vision-based Music IR.

BIBLIOGRAPHY

[1] Jakob Abeßer, Olivier Lartillot, Christian Dittmar, Tuomas Eerola, and Gerald Schuller. "Modeling musical attributes to characterize ensemble recordings using rhythmic audio features". In: *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2011, May 22-27, 2011, Prague Congress Center, Prague, Czech Republic*. 2011, pp. 189–192. DOI: 10.1109/ICASSP.2011.5946372. URL: https://doi.org/10.1109/ICASSP.2011.5946372.

[2] Enrique Amigó, Julio Gonzalo, Javier Artiles, and Felisa Verdejo. "A comparison of extrinsic clustering evaluation metrics based on formal constraints". In: *Inf. Retr.* 12.4 (2009), pp. 461–486. DOI: 10.1007/s10791-008-9066-8.

[3] Evlampios E. Apostolidis and Vasileios Mezaris. "Fast shot segmentation combining global and local visual descriptors". In: *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2014, Florence, Italy, May 4-9, 2014*. 2014, pp. 6583–6587. DOI: 10.1109/ICASSP.2014.6854873. URL: http://dx.doi.org/10.1109/ICASSP.2014.6854873.

[4] Sunil Arya, David M. Mount, Nathan S. Netanyahu, Ruth Silverman, and Angela Y. Wu. "An Optimal Algorithm for Approximate Nearest Neighbor Searching Fixed Dimensions". In: *J. ACM* 45.6 (1998), pp. 891–923. DOI: 10.1145/293347.293348. URL: http://doi.acm.org/10.1145/293347.293348.

[5] Andreas Arzt, Gerhard Widmer, and Simon Dixon. "Automatic Page Turning for Musicians via Real-Time Machine Listening". In: *ECAI 2008 - 18th European Conference on Artificial Intelligence, Patras, Greece, July 21-25, 2008, Proceedings*. 2008, pp. 241–245. DOI: 10.3233/978-1-58603-891-5-241.

[6] Ricardo A. Baeza-Yates, Paolo Boldi, Berthier A. Ribeiro-Neto, and Berkant Barla Cambazoglu, eds. *Proceedings of the Second International Conference on Web Search and Web Data Mining, WSDM 2009, Barcelona, Spain, February 9-11, 2009*. ACM, 2009. ISBN: 978-1-60558-390-7.

[7] Andrew D. Bagdanov, Alberto Del Bimbo, Fabrizio Dini, Giuseppe Lisanti, and Iacopo Masi. "Posterity Logging of Face Imagery for Video Surveillance". In: *IEEE MultiMedia* 19.4 (2012), pp. 48–59. DOI: 10.1109/MMUL.2012.15.

[8] Xiao Bai, Jian Cheng, and Edwin Hancock. *Graph-Based Methods in Computer Vision: Developments and Applications*. IGI Global Research Collection. Information Science Reference, 2012. ISBN: 9781466618923. URL: https://books.google.nl/books?id=V66eBQAAQBAJ.

[9] Jayme G. A. Barbedo and George Tzanetakis. "Musical Instrument Classification Using Individual Partials". In: *IEEE Transactions on Audio, Speech & Language Processing* 19.1 (2011), pp. 111–122. DOI: 10.1109/TASL.2010.2045186.

[10] Jeremiah R. Barr, Leonardo A. Cament, Kevin W. Bowyer, and Patrick J. Flynn. "Active Clustering with Ensembles for Social structure extraction". In: *IEEE Winter Conference on Applications of Computer Vision, Steamboat Springs, CO, USA, March 24-26, 2014*. 2014, pp. 969–976. DOI: 10.1109/WACV.2014.6835999. URL: http://dx.doi.org/10.1109/WACV.2014.6835999.

[11] Zohar Barzelay and Yoav Y. Schechner. "Onsets Coincidence for Cross-Modal Analysis". In: *IEEE Trans. Multimedia* 12.2 (2010), pp. 108–120. DOI: 10.1109/TMM.2009.2037387. URL: https://doi.org/10.1109/TMM.2009.2037387.

[12] Alessio Bazzica, Cynthia C. S. Liem, and Alan Hanjalic. "Exploiting Instrument-wise Playing/Non-Playing Labels for Score Synchronization of Symphonic Music". In: *Proceedings of the 15th International Society for Music Information Retrieval Conference, ISMIR 2014, Taipei, Taiwan, October 27-31, 2014*. 2014, pp. 201–206. URL: http://www.terasoft.com.tw/conf/ismir2014/proceedings/T037_182_Paper.pdf.

[13] Alessio Bazzica, Cynthia C. S. Liem, and Alan Hanjalic. "On detecting the playing/non-playing activity of musicians in symphonic music videos". In: *Computer Vision and Image Understanding* 144 (2016), pp. 188–204. DOI: 10.1016/j.cviu.2015.09.009. URL: http://dx.doi.org/10.1016/j.cviu.2015.09.009.

[14] Alessio Bazzica, Cynthia C. S. Liem, and Alan Hanjalic. "Exploiting scene maps and spatial relationships in quasi-static scenes for video face clustering". In: *Image and Vision Computing* 57 (2017), pp. 25–43. DOI: 10.1016/j.imavis.2016.11.005. URL: https://doi.org/10.1016/j.imavis.2016.11.005.

[15] Alessio Bazzica, J. C. van Gemert, Cynthia C. S. Liem, and Alan Hanjalic. "Vision-based Detection of Acoustic Timed Events: a Case Study on Clarinet Note Onsets". In: *CoRR* abs/1706.09556 (2017). arXiv: 1706.09556. URL: http://arxiv.org/abs/1706.09556.

[16] Sebastian Böck and Gerhard Widmer. "Maximum filter vibrato suppression for onset detection". In: *Proceedings of the 16th International Conference on Digital Audio Effects (DAFx). Maynooth, Ireland (2013)*. 2013.

[17] G. R. Bradski and J. W. Davis. "Motion segmentation and pose recognition with motion history gradients". In: *Mach. Vis. Appl.* 13.3 (2002), pp. 174–184.

[18] A.M. Burns and M.M. Wanderley. "Visual Methods for the Retrieval of Guitarist Fingering". In: *NIME*. 2006.

[19] Xiaochun Cao, Changqing Zhang, Huazhu Fu, Si Liu, and Hua Zhang. "Diversity-induced Multi-view Subspace Clustering". In: *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*. 2015, pp. 586–594. DOI: 10.1109/CVPR.2015.7298657. URL: http://dx.doi.org/10.1109/CVPR.2015.7298657.

[20] B. Caramiaux, M. M. Wanderley, and F. Bevilacqua. "Segmenting and Parsing Instrumentalist's Gestures". In: *Journal of New Music Research* 41.1 (Apr. 2012), pp. 13–29.

[21] Joon Son Chung, Andrew W. Senior, Oriol Vinyals, and Andrew Zisserman. “Lip Reading Sentences in the Wild”. In: *CoRR* abs/1611.05358 (2016). URL: http://arxiv.org/abs/1611.05358.

[22] Simon Clippingdale and Mahito Fujii. “Skin Region Extraction and Person-Independent Deformable Face Templates for Fast Video Indexing”. In: *2011 IEEE International Symposium on Multimedia, ISM 2011, Dana Point, CA, USA, December 5-7, 2011*. 2011, pp. 416–421. DOI: 10.1109/ISM.2011.75. URL: http://dx.doi.org/10.1109/ISM.2011.75.

[23] G. Csurka, C. Dance, L. Fan, J. Willamowski, and C. Bray. “Visual categorization with bags of keypoints”. In: *Workshop on statistical learning in computer vision, European Conference on Computer Vision (ECCV)*. Vol. 1. 1-22. 2004, pp. 1–2.

[24] Antonello D’Aguanno and Giancarlo Vercellesi. “Automatic Music Synchronization Using Partial Score Representation Based on IEEE 1599”. In: *Journal of Multimedia* 4.1 (2009), pp. 19–24. DOI: 10.4304/jmm.4.1.19-24.

[25] Roger B. Dannenberg and Christopher Raphael. “Music score alignment and computer accompaniment”. In: *Commun. ACM* 49.8 (2006), pp. 38–43. DOI: 10.1145/1145311. URL: http://doi.acm.org/10.1145/1145311.

[26] James Davis. “Recognizing movement using motion histograms”. In: *Technial Report 487, MIT Media Lab* 1.487 (1999), p. 1.

[27] Karthik Dinesh, Bochen Li, Xinzhao Liu, Zhiyao Duan, and Gaurav Sharma. “Visually informed multi-pitch analysis of string ensembles”. In: *2017 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2017, New Orleans, LA, USA, March 5-9, 2017*. 2017, pp. 3021–3025. DOI: 10.1109/ICASSP.2017.7952711. URL: https://doi.org/10.1109/ICASSP.2017.7952711.

[28] Simon Dixon, Werner Goebl, and Gerhard Widmer. “The "Air Worm": an Interface for Real-Time manipulation of Expressive Music Performance”. In: *Proceedings of the 2005 International Computer Music Conference, ICMC 2005, Barcelona, Spain, September 4-10, 2005*. 2005. URL: http://hdl.handle.net/2027/spo.bbp2372.2005.048.

[29] Jeff Donahue, Lisa Anne Hendricks, Sergio Guadarrama, Marcus Rohrbach, Subhashini Venugopalan, Trevor Darrell, and Kate Saenko. “Long-term recurrent convolutional networks for visual recognition and description”. In: *IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2015, Boston, MA, USA, June 7-12, 2015*. 2015, pp. 2625–2634. DOI: 10.1109/CVPR.2015.7298878. URL: https://doi.org/10.1109/CVPR.2015.7298878.

[30] Slim Essid and Gaël Richard. “Fusion of Multimodal Information in Music Content Analysis”. In: *Multimodal Music Processing*. 2012, pp. 37–52. DOI: 10.4230/DFU.Vol3.11041.37. URL: https://doi.org/10.4230/DFU.Vol3.11041.37.

[31] Martin Ester, Hans-Peter Kriegel, Jörg Sander, and Xiaowei Xu. “A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise”. In: *Proceedings of the Second International Conference on Knowledge Discovery and Data Mining (KDD-96), Portland, Oregon, USA*. 1996, pp. 226–231.

[32] Sebastian Ewert and Meinard Müller. "Using score-informed constraints for NMF-based source separation". In: *2012 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2012, Kyoto, Japan, March 25-30, 2012*. 2012, pp. 129–132. DOI: 10.1109/ICASSP.2012.6287834. URL: https://doi.org/10.1109/ICASSP.2012.6287834.

[33] Sebastian Ewert, Meinard Müller, and Roger B. Dannenberg. "Towards Reliable Partial Music Alignments Using Multiple Synchronization Strategies". In: *Adaptive Multimedia Retrieval. Understanding Media and Adapting to the User - 7th International Workshop, AMR 2009, Madrid, Spain, September 24-25, 2009, Revised Selected Papers*. 2009, pp. 35–48. DOI: 10.1007/978-3-642-18449-9_4. URL: https://doi.org/10.1007/978-3-642-18449-9_4.

[34] Sebastian Ewert, Bryan Pardo, Meinard Müller, and Mark D. Plumbley. "Score-Informed Source Separation for Musical Audio Recordings: An overview". In: *IEEE Signal Process. Mag.* 31.3 (2014), pp. 116–124. DOI: 10.1109/MSP.2013.2296076.

[35] Christoph Feichtenhofer, Axel Pinz, and Andrew Zisserman. "Convolutional Two-Stream Network Fusion for Video Action Recognition". In: *2016 IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2016, Las Vegas, NV, USA, June 27-30, 2016*. 2016, pp. 1933–1941. DOI: 10.1109/CVPR.2016.213. URL: https://doi.org/10.1109/CVPR.2016.213.

[36] Martin A. Fischler and Robert C. Bolles. "Random Sample Consensus: A Paradigm for Model Fitting with Applications to Image Analysis and Automated Cartography". In: *Commun. ACM* 24.6 (1981), pp. 381–395. DOI: 10.1145/358669.358692. URL: http://doi.acm.org/10.1145/358669.358692.

[37] Jonathan Foote. "Automatic Audio Segmentation using a Measure of Audio Novelty". In: *2000 IEEE International Conference on Multimedia and Expo, ICME 2000, New York, NY, USA, July 30 - August 2, 2000*. 2000, p. 452.

[38] C. Fremerey, M. Müller, and M. Clausen. "Towards Bridging the Gap between Sheet Music and Audio". In: *Knowledge Representation for Intelligent Music Processing* 09051 (2009).

[39] Christian Fremerey, Michael Clausen, Sebastian Ewert, and Meinard Müller. "Sheet Music-Audio Identification". In: *Proceedings of the 10th International Society for Music Information Retrieval Conference, ISMIR 2009, Kobe International Conference Center, Kobe, Japan, October 26-30, 2009*. 2009, pp. 645–650. URL: http://ismir2009.ismir.net/proceedings/PS4-12.pdf.

[40] Olivier Gillet and Gaël Richard. "Automatic transcription of drum sequences using audiovisual features". In: *2005 IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP '05, Philadelphia, Pennsylvania, USA, March 18-23, 2005*. 2005, pp. 205–208. DOI: 10.1109/ICASSP.2005.1415682. URL: https://doi.org/10.1109/ICASSP.2005.1415682.

[41] Olivier Gillet and Gaël Richard. "ENST-Drums: an extensive audio-visual database for drum signals processing". In: *ISMIR 2006, 7th International Conference on Music Information Retrieval, Victoria, Canada, 8-12 October 2006, Proceedings*. 2006, pp. 156–159.

[42] Rolf Inge Godøy and Alexander Refsum Jensenius. "Body Movement in Music Information Retrieval". In: *Proceedings of the 10th International Society for Music Information Retrieval Conference, ISMIR 2009, Kobe International Conference Center, Kobe, Japan, October 26-30, 2009*. 2009, pp. 45–50. URL: http://ismir2009.ismir.net/proceedings/OS1-4.pdf.

[43] Emilia Gómez, Maarten Grachten, Alan Hanjalic, Jordi Janer, Sergi Jorda, Carles F Julia, Cynthia Liem, Agustin Martorell, Markus Schedl, and Gerhard Widmer. "PHENICX: Performances as Highly Enriched aNd Interactive Concert Experiences". In: *Proceedings of the Sound and Music Computing Conference (Stockholm, 2013)*. 2013.

[44] Maarten Grachten, Martin Gasser, Andreas Arzt, and Gerhard Widmer. "Automatic Alignment of Music Performances with Structural Differences". In: *Proceedings of the 14th International Society for Music Information Retrieval Conference, ISMIR 2013, Curitiba, Brazil, November 4-8, 2013*. 2013, pp. 607–612.

[45] Yushen Han and Christopher Raphael. "Informed source separation of orchestra and soloist using masking and unmasking". In: *ISCA Workshop on Statistical And Perceptual Audition, SAPA 2010, Makuhari, Japan, September 25, 2010*. 2010, pp. 31–36. URL: http://www.isca-speech.org/archive/sapa_2010/sap1_031.html.

[46] Andrew Harltey and Andrew Zisserman. *Multiple view geometry in computer vision (2. ed.)* Cambridge University Press, 2006. ISBN: 978-0-521-54051-3.

[47] Sergey Ioffe and Christian Szegedy. "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift". In: *Proceedings of the 32nd International Conference on Machine Learning, ICML 2015, Lille, France, 6-11 July 2015*. 2015, pp. 448–456. URL: http://jmlr.org/proceedings/papers/v37/ioffe15.html.

[48] Cyril Joder, Slim Essid, and Gaël Richard. "A comparative study of tonal acoustic features for a symbolic level music-to-score alignment". In: *Proceedings of the IEEE International Conference on Acoustics, Speech, and Signal Processing, ICASSP 2010, 14-19 March 2010, Sheraton Dallas Hotel, Dallas, Texas, USA*. 2010, pp. 409–412. DOI: 10.1109/ICASSP.2010.5495784.

[49] Ian Jolliffe. *Principal Component Analysis*. John Wiley & Sons, Ltd, 2005. ISBN: 9780470013199. DOI: 10.1002/0470013192.bsa501. URL: http://dx.doi.org/10.1002/0470013192.bsa501.

[50] Zdenek Kalal, Krystian Mikolajczyk, and Jiri Matas. "Tracking-Learning-Detection". In: *IEEE Trans. Pattern Anal. Mach. Intell.* 34.7 (2012), pp. 1409–1422. DOI: 10.1109/TPAMI.2011.239. URL: http://doi.ieeecomputersociety.org/10.1109/TPAMI.2011.239.

[51] Subhradeep Kayal. "Face clustering in videos: GMM-based hierarchical clustering using Spatio-Temporal data". In: *13th UK Workshop on Computational Intelligence, UKCI 2013, Guildford, United Kingdom, September 9-11, 2013*. 2013, pp. 272–278. DOI: 10.1109/UKCI.2013.6651316. URL: http://dx.doi.org/10.1109/UKCI.2013.6651316.

[52] Elie el Khoury, Paul Gay, and Jean-Marc Odobez. “Fusing matching and biometric similarity measures for face diarization in video”. In: *International Conference on Multimedia Retrieval, ICMR'13, Dallas, TX, USA, April 16-19, 2013*. 2013, pp. 97–104. DOI: 10.1145/2461466.2461484. URL: http://doi.acm.org/10.1145/2461466.2461484.

[53] Elie el Khoury, Christine Sénac, and Philippe Joly. “Face-and-clothing based people clustering in video content”. In: *Proceedings of the 11th ACM SIGMM International Conference on Multimedia Information Retrieval, MIR 2010, Philadelphia, Pennsylvania, USA, March 29-31, 2010*. 2010, pp. 295–304. DOI: 10.1145/1743384.1743435. URL: http://doi.acm.org/10.1145/1743384.1743435.

[54] Elie el Khoury, Christine Sénac, and Philippe Joly. “Audiovisual diarization of people in video content”. In: *Multimedia Tools Appl.* 68.3 (2014), pp. 747–775. DOI: 10.1007/s11042-012-1080-6.

[55] Harold W Kuhn. “The Hungarian method for the assignment problem”. In: *Naval research logistics quarterly* 2.1-2 (1955), pp. 83–97.

[56] Svetlana Lazebnik, Cordelia Schmid, and Jean Ponce. “Beyond Bags of Features: Spatial Pyramid Matching for Recognizing Natural Scene Categories”. In: *2006 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2006), 17-22 June 2006, New York, NY, USA*. 2006, pp. 2169–2178. DOI: 10.1109/CVPR.2006.68.

[57] Bochen Li, Karthik Dinesh, Zhiyao Duan, and Gaurav Sharma. “See and listen: Score-informed association of sound tracks to players in chamber music performance videos”. In: *2017 IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2017, New Orleans, LA, USA, March 5-9, 2017*. 2017, pp. 2906–2910. DOI: 10.1109/ICASSP.2017.7952688. URL: https://doi.org/10.1109/ICASSP.2017.7952688.

[58] Cynthia C. S. Liem, Alessio Bazzica, and Alan Hanjalic. “Looking beyond sound: Unsupervised analysis of musician videos”. In: *14th International Workshop on Image Analysis for Multimedia Interactive Services, WIAMIS 2013, Paris, France, July 3-5, 2013*. 2013, pp. 1–4. DOI: 10.1109/WIAMIS.2013.6616163.

[59] Cynthia C. S. Liem, Meinard Müller, Douglas Eck, George Tzanetakis, and Alan Hanjalic. “The need for music information retrieval with user-centered and multimodal strategies”. In: *Proceedings of the 1st international ACM workshop on Music information retrieval with user-centered and multimodal strategies, Scottsdale, AZ, USA, November 28 - December 01, 2011*. 2011, pp. 1–6. DOI: 10.1145/2072529.2072531. URL: http://doi.acm.org/10.1145/2072529.2072531.

[60] David G. Lowe. “Distinctive Image Features from Scale-Invariant Keypoints”. In: *International Journal of Computer Vision* 60.2 (2004), pp. 91–110. DOI: 10.1023/B:VISI.0000029664.99615.94. URL: http://dx.doi.org/10.1023/B:VISI.0000029664.99615.94.

[61] Mathias Lux and Savvas A. Chatzichristofis. “Lire: lucene image retrieval: an extensible java CBIR library”. In: *Proceedings of the 16th International Conference on Multimedia 2008, Vancouver, British Columbia, Canada, October 26-31, 2008*. 2008, pp. 1085–1088. DOI: 10.1145/1459359.1459577.

[62] Luis Gustavo Martins, Juan José Burred, George Tzanetakis, and Mathieu Lagrange. “Polyphonic Instrument Recognition Using Spectral Clustering”. In: *Proceedings of the 8th International Conference on Music Information Retrieval, ISMIR 2007, Vienna, Austria, September 23-27, 2007*. 2007, pp. 213–218.

[63] Iacopo Masi, Giuseppe Lisanti, Andrew D. Bagdanov, Pietro Pala, and Alberto Del Bimbo. “Using 3D Models to Recognize 2D Faces in the Wild”. In: *IEEE Conference on Computer Vision and Pattern Recognition, CVPR Workshops 2013, Portland, OR, USA, June 23-28, 2013*. 2013, pp. 775–780. DOI: 10.1109/CVPRW.2013.116. URL: http://dx.doi.org/10.1109/CVPRW.2013.116.

[64] Markus Mathias, Rodrigo Benenson, Marco Pedersoli, and Luc Van Gool. “Face Detection without Bells and Whistles”. In: *Computer Vision - ECCV 2014 - 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part IV*. 2014, pp. 720–735. DOI: 10.1007/978-3-319-10593-2_47. URL: http://dx.doi.org/10.1007/978-3-319-10593-2_47.

[65] Kevin McGuinness, Olivier Gillet, Noel E. O’Connor, and Gaël Richard. “Visual analysis for drum sequence transcription”. In: *15th European Signal Processing Conference, EUSIPCO 2007, Poznan, Poland, September 3-7, 2007*. 2007, pp. 312–316. URL: http://ieeexplore.ieee.org/document/7098815/.

[66] Mark S. Melenhorst and Cynthia C. S. Liem. “Put the Concert Attendee in the Spotlight. A User-Centered Design and Development Approach for Classical Concert Applications”. In: *Proceedings of the 16th International Society for Music Information Retrieval Conference, ISMIR 2015, Málaga, Spain, October 26-30, 2015*. 2015, pp. 800–806. URL: http://ismir2015.uma.es/articles/67_Paper.pdf.

[67] Pascal Mettes, Jan C. van Gemert, and Cees G. M. Snoek. “Spot On: Action Localization from Pointly-Supervised Proposals”. In: *Computer Vision - ECCV 2016 - 14th European Conference, Amsterdam, The Netherlands, October 11-14, 2016, Proceedings, Part V*. 2016, pp. 437–453. DOI: 10.1007/978-3-319-46454-1_27. URL: https://doi.org/10.1007/978-3-319-46454-1_27.

[68] Meinard Müller and Sebastian Ewert. “Joint Structure Analysis with Applications to Music Annotation and Synchronization”. In: *ISMIR 2008, 9th International Conference on Music Information Retrieval, Drexel University, Philadelphia, PA, USA, September 14-18, 2008*. 2008, pp. 389–394.

[69] Meinard Müller and Nanzhu Jiang. “A Scape Plot Representation for Visualizing Repetitive Structures of Music Recordings”. In: *Proceedings of the 13th International Society for Music Information Retrieval Conference, ISMIR 2012, Mosteiro S.Bento Da Vitória, Porto, Portugal, October 8-12, 2012*. 2012, pp. 97–102.

[70] Meinard Müller, Frank Kurth, and Tido Röder. “Towards an Efficient Algorithm for Automatic Score-to-Audio Synchronization”. In: *ISMIR 2004, 5th International Conference on Music Information Retrieval, Barcelona, Spain, October 10-14, 2004, Proceedings*. 2004.

[71] Lukas Neumann and Jiri Matas. “Real-time scene text localization and recognition”. In: *2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, June 16-21, 2012*. 2012, pp. 3538–3545. DOI: 10.1109/CVPR.2012.6248097. URL: http://dx.doi.org/10.1109/CVPR.2012.6248097.

[72] Minh Hoai Nguyen, Zhen-Zhong Lan, and Fernando De la Torre. “Joint segmentation and classification of human actions in video”. In: *The 24th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2011, Colorado Springs, CO, USA, 20-25 June 2011*. 2011, pp. 3265–3272. DOI: 10.1109/CVPR.2011.5995470.

[73] Jorge Nocedal and Stephen J Wright. *Springer series in operations research. Numerical optimization*. 1999.

[74] Kristian Nymoen, Baptiste Caramiaux, Mariusz Kozak, and Jim Tørresen. “Analyzing sound tracings: a multimodal approach to music information retrieval”. In: *Proceedings of the 1st international ACM workshop on Music information retrieval with user-centered and multimodal strategies, Scottsdale, AZ, USA, November 28 - December 01, 2011*. 2011, pp. 39–44. DOI: 10.1145/2072529.2072541. URL: http://doi.acm.org/10.1145/2072529.2072541.

[75] Marco Paleari, Benoit Huet, Antony Schutz, and Dirk T. M. Slock. “A multimodal approach to music transcription”. In: *Proceedings of the International Conference on Image Processing, ICIP 2008, October 12-15, 2008, San Diego, California, USA*. 2008, pp. 93–96. DOI: 10.1109/ICIP.2008.4711699. URL: http://dx.doi.org/10.1109/ICIP.2008.4711699.

[76] Johann Poignant, Laurent Besacier, Viet Bac Le, Sophie Rosset, and Georges Quénot. “Unsupervised naming of speakers in broadcast TV: using written names, pronounced names or both?” In: *INTERSPEECH 2013, 14th Annual Conference of the International Speech Communication Association, Lyon, France, August 25-29, 2013*. 2013, pp. 1462–1466. URL: http://www.isca-speech.org/archive/interspeech_2013/i13_1462.html.

[77] Christopher Raphael. “Aligning music audio with symbolic scores using a hybrid graphical model”. In: *Machine Learning* 65.2-3 (2006), pp. 389–409. DOI: 10.1007/s10994-006-8415-3.

[78] Michael S. Ryoo and Jake K. Aggarwal. “Spatio-temporal relationship match: Video structure comparison for recognition of complex human activities”. In: *IEEE 12th International Conference on Computer Vision, ICCV 2009, Kyoto, Japan, September 27 - October 4, 2009*. 2009, pp. 1593–1600. DOI: 10.1109/ICCV.2009.5459361.

[79] Koen E. A. van de Sande, Theo Gevers, and Cees G. M. Snoek. “Evaluating Color Descriptors for Object and Scene Recognition”. In: *IEEE Trans. Pattern Anal. Mach. Intell.* 32.9 (2010), pp. 1582–1596. DOI: 10.1109/TPAMI.2009.154.

[80] Jan Schlüter and Sebastian Böck. “Improved musical onset detection with Convolutional Neural Networks”. In: *IEEE International Conference on Acoustics, Speech and Signal Processing, ICASSP 2014, Florence, Italy, May 4-9, 2014*. 2014, pp. 6979–6983. DOI: 10.1109/ICASSP.2014.6854953. URL: https://doi.org/10.1109/ICASSP.2014.6854953.

[81] Simeon Schwab, Thierry Chateau, Christophe Blanc, and Laurent Trassoudaine. “A multi-cue spatio-temporal framework for automatic frontal face clustering in video sequences”. In: *EURASIP J. Image and Video Processing* 2013 (2013), p. 10. DOI: 10.1186/1687-5281-2013-10. URL: http://dx.doi.org/10.1186/1687-5281-2013-10.

[82] D. Sculley. “Web-scale k-means clustering”. In: *Proceedings of the 19th International Conference on World Wide Web, WWW 2010, Raleigh, North Carolina, USA, April 26-30, 2010*. 2010, pp. 1177–1178. DOI: 10.1145/1772690.1772862.

[83] Caifeng Shan. “Face Recognition and Retrieval in Video”. In: *Video Search and Mining*. 2010, pp. 235–260. DOI: 10.1007/978-3-642-12900-1_9.

[84] Karen Simonyan and Andrew Zisserman. “Two-Stream Convolutional Networks for Action Recognition in Videos”. In: *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*. 2014, pp. 568–576.

[85] Josef Sivic, Mark Everingham, and Andrew Zisserman. “"Who are you?" - Learning person specific classifiers from video”. In: *2009 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2009), 20-25 June 2009, Miami, Florida, USA*. 2009, pp. 1145–1152. DOI: 10.1109/CVPRW.2009.5206513. URL: http://dx.doi.org/10.1109/CVPRW.2009.5206513.

[86] Noah Snavely, Steven M. Seitz, and Richard Szeliski. “Photo tourism: exploring photo collections in 3D”. In: *ACM Trans. Graph.* 25.3 (2006), pp. 835–846. DOI: 10.1145/1141911.1141964. URL: http://doi.acm.org/10.1145/1141911.1141964.

[87] Khurram Soomro, Amir Roshan Zamir, and Mubarak Shah. “UCF101: A Dataset of 101 Human Actions Classes From Videos in The Wild”. In: *CoRR* abs/1212.0402 (2012).

[88] Drew Steedly, Chris Pal, and Richard Szeliski. “Efficiently Registering Video into Panoramic Mosaics”. In: *10th IEEE International Conference on Computer Vision (ICCV 2005), 17-20 October 2005, Beijing, China*. 2005, pp. 1300–1307. DOI: 10.1109/ICCV.2005.86. URL: http://doi.ieeecomputersociety.org/10.1109/ICCV.2005.86.

[89] Tiago Fernandes Tavares, Gabrielle Odowichuck, Sonmaz Zehtabi, and George Tzanetakis. “Audio-visual vibraphone transcription in real time”. In: *14th IEEE International Workshop on Multimedia Signal Processing, MMSP 2012, Banff, AB, Canada, September 17-19, 2012*. 2012, pp. 215–220. DOI: 10.1109/MMSP.2012.6343443. URL: https://doi.org/10.1109/MMSP.2012.6343443.

[90] G.A. Ten Holt, M.J.T. Reinders, and E.A. Hendriks. “Multi-dimensional Dynamic Time Warping for Gesture Recognition”. In: *13th annual conference of the Advanced School for Computing and Imaging*. Vol. 119. 2007.

[91] Verena Thomas, Christian Fremerey, Meinard Müller, and Michael Clausen. “Linking Sheet Music and Audio - Challenges and New Approaches”. In: *Multimodal Music Processing*. 2012, pp. 1–22. DOI: 10.4230/DFU.Vol3.11041.1.

[92] Du Tran, Lubomir D. Bourdev, Rob Fergus, Lorenzo Torresani, and Manohar Paluri. “Learning Spatiotemporal Features with 3D Convolutional Networks”. In: *2015 IEEE International Conference on Computer Vision, ICCV 2015, Santiago, Chile, December 7-13, 2015*. 2015, pp. 4489–4497. DOI: 10.1109/ICCV.2015.510. URL: https://doi.org/10.1109/ICCV.2015.510.

[93] Robert J. Turetsky and Daniel P. W. Ellis. “Ground-truth transcriptions of real music from force-aligned MIDI syntheses”. In: *ISMIR 2003, 4th International Conference on Music Information Retrieval, Baltimore, Maryland, USA, October 27-30, 2003, Proceedings*. 2003. URL: http://ismir2003.ismir.net/papers/Turetsky.PDF.

[94] Emmanuel Vincent, Shoko Araki, Fabian J. Theis, Guido Nolte, Pau Bofill, Hiroshi Sawada, Alexey Ozerov, Vikrham Gowreesunker, Dominik Lutter, and Ngoc Q. K. Duong. “The signal separation evaluation campaign (2007-2010): Achievements and remaining challenges”. In: *Signal Processing* 92.8 (2012), pp. 1928–1936. DOI: 10.1016/j.sigpro.2011.10.007.

[95] Emmanuel Vincent, Nancy Bertin, Rémi Gribonval, and Frédéric Bimbot. “From Blind to Guided Audio Source Separation: How models and side information can improve the separation of sound”. In: *IEEE Signal Process. Mag.* 31.3 (2014), pp. 107–115. DOI: 10.1109/MSP.2013.2297440. URL: http://dx.doi.org/10.1109/MSP.2013.2297440.

[96] Bradley W. Vines, Marcelo M. Wanderley, Carol Krumhansl, Regina L. Nuzzo, and Daniel J. Levitin. “Performance Gestures of Musicians: What Structural and Emotional Information Do They Convey?” In: *Gesture-Based Communication in Human-Computer Interaction, 5th International Gesture Workshop, GW 2003, Genova, Italy, April 15-17, 2003, Selected Revised Papers*. 2003, pp. 468–478. DOI: 10.1007/978-3-540-24598-8_43. URL: https://doi.org/10.1007/978-3-540-24598-8_43.

[97] Paul A. Viola and Michael J. Jones. “Rapid Object Detection using a Boosted Cascade of Simple Features”. In: *2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2001), with CD-ROM, 8-14 December 2001, Kauai, HI, USA*. 2001, pp. 511–518. DOI: 10.1109/CVPR.2001.990517.

[98] Carl Vondrick, Donald J. Patterson, and Deva Ramanan. “Efficiently Scaling up Crowdsourced Video Annotation - A Set of Best Practices for High Quality, Economical Video Labeling”. In: *International Journal of Computer Vision* 101.1 (2013), pp. 184–204. DOI: 10.1007/s11263-012-0564-1. URL: http://dx.doi.org/10.1007/s11263-012-0564-1.

[99] Nicholas Vretos, Vassilios Solachidis, and Ioannis Pitas. "A mutual information based face clustering algorithm for movie content analysis". In: *Image Vision Comput.* 29.10 (2011), pp. 693–705. DOI: 10.1016/j.imavis.2011.07.006. URL: http://dx.doi.org/10.1016/j.imavis.2011.07.006.

[100] Kiri Wagstaff, Claire Cardie, Seth Rogers, and Stefan Schrödl. "Constrained K-means Clustering with Background Knowledge". In: *Proceedings of the Eighteenth International Conference on Machine Learning (ICML 2001), Williams College, Williamstown, MA, USA, June 28 - July 1, 2001*. 2001, pp. 577–584.

[101] Marcelo M. Wanderley. "Quantitative Analysis of Non-obvious Performer Gestures". In: *Gesture and Sign Languages in Human-Computer Interaction, International Gesture Workshop, GW 2001, London, UK, April 18-20, 2001, Revised Papers*. 2001, pp. 241–253. DOI: 10.1007/3-540-47873-6_26. URL: https://doi.org/10.1007/3-540-47873-6_26.

[102] Baoyuan Wu, Yifan Zhang, Baogang Hu, and Qiang Ji. "Constrained Clustering and Its Application to Face Clustering in Videos". In: *2013 IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, June 23-28, 2013*. 2013, pp. 3507–3514. DOI: 10.1109/CVPR.2013.450.

[103] Peng Wu and Feng Tang. "Improving face clustering using social context". In: *Proceedings of the 18th International Conference on Multimedia 2010, Firenze, Italy, October 25-29, 2010*. 2010, pp. 907–910. DOI: 10.1145/1873951.1874110. URL: http://doi.acm.org/10.1145/1873951.1874110.

[104] Shijie Xiao, Mingkui Tan, and Dong Xu. "Weighted Block-Sparse Low Rank Representation for Face Clustering in Videos". In: *Computer Vision - ECCV 2014 - 13th European Conference, Zurich, Switzerland, September 6-12, 2014, Proceedings, Part VI*. 2014, pp. 123–138. DOI: 10.1007/978-3-319-10599-4_9. URL: http://dx.doi.org/10.1007/978-3-319-10599-4_9.

[105] Bangpeng Yao and Fei-Fei Li. "Grouplet: A structured image representation for recognizing human and object interactions". In: *The Twenty-Third IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2010, San Francisco, CA, USA, 13-18 June 2010*. 2010, pp. 9–16. DOI: 10.1109/CVPR.2010.5540234.

[106] Bangpeng Yao, Jiayuan Ma, and Li Fei-Fei. "Discovering Object Functionality". In: *IEEE International Conference on Computer Vision, ICCV 2013, Sydney, Australia, December 1-8, 2013*. 2013, pp. 2512–2519. DOI: 10.1109/ICCV.2013.312.

[107] Liyan Zhang, Dmitri V. Kalashnikov, and Sharad Mehrotra. "A unified framework for context assisted face clustering". In: *International Conference on Multimedia Retrieval, ICMR'13, Dallas, TX, USA, April 16-19, 2013*. 2013, pp. 9–16. DOI: 10.1145/2461466.2461469.

[108] Liyan Zhang, Dmitri V. Kalashnikov, and Sharad Mehrotra. "Context-assisted face clustering framework with human-in-the-loop". In: *IJMIR* 3.2 (2014), pp. 69–88. DOI: 10.1007/s13735-014-0052-1.

[109] Tong Zhang, Di Wen, and Xiaoqing Ding. "Person-based video summarization and retrieval by tracking and clustering temporal face sequences". In: *IS&T/SPIE Electronic Imaging*. International Society for Optics and Photonics. 2013.

[110] Wen-Yi Zhao, Rama Chellappa, P. Jonathon Phillips, and Azriel Rosenfeld. “Face recognition: A literature survey”. In: *ACM Comput. Surv.* 35.4 (2003), pp. 399–458. DOI: 10.1145/954339.954342. URL: http://doi.acm.org/10.1145/954339.954342.

[111] Feng Zhou and Fernando De la Torre. “Deformable Graph Matching”. In: *2013 IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, June 23-28, 2013*. 2013, pp. 2922–2929. DOI: 10.1109/CVPR.2013.376. URL: http://dx.doi.org/10.1109/CVPR.2013.376.

[112] Chunhui Zhu, Fang Wen, and Jian Sun. “A rank-order distance based clustering algorithm for face tagging”. In: *The 24th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2011, Colorado Springs, CO, USA, 20-25 June 2011*. 2011, pp. 481–488. DOI: 10.1109/CVPR.2011.5995680. URL: http://dx.doi.org/10.1109/CVPR.2011.5995680.

[113] Xiangxin Zhu and Deva Ramanan. “Face detection, pose estimation, and landmark localization in the wild”. In: *2012 IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, June 16-21, 2012*. 2012, pp. 2879–2886. DOI: 10.1109/CVPR.2012.6248014.

ACKNOWLEDGEMENTS

Getting a PhD is hard. But as every hard thing one faces, it pays off. In my experience, the PhD has been a trial that began with little available resources: limited knowledge on the research topic, limited networking, limited communication skills. I had to learn how to deal with the different priorities of other people, to communicate effectively, to be polite, to mediate. I then realized that being a researcher requires a lot of skills; and technical ones, though important, are just one among many.

In the beginning, I was focusing only on results and due to such narrow view, I came quite close to feeling burnt out. Luckily, when reality challenges us, it also provides us with everything we need to face it. This final chapter is about the help I received from reality, that is all the people I came across during my Dutch adventure. Thanks to them, I could grow up both professionally and, most importantly, as a person.

First of all, I would like to thank my promotor Alan and my supervisor Cynthia. You were both extremely patient with me, and I will always be thankful for giving me the opportunity to learn and grow step-by-step. Too often I believed that more supervision was needed, but you surely knew that facing some challenges by myself was part of the journey. Now I can say that you have done a great job, because I see myself in such a different way from the beginning, both professionally and as a person. And I am proud of the work we have done together.

Alan, during my first year you told me two things that I needed to hear right at that time. The first is “door de bomen het bos niet meer zien”, which translates to “can’t see the wood for the trees”. Then, when motivating my research was not a skill of mine at all, you asked me the following: “why do you get up in the morning to do what you do?” These words accompanied me for quite some time and were very useful far beyond the paper I was writing at that time. Also, you are very well aware that people in your group have personal needs and you grant flexibility accordingly, which allowed me to deal with being far from my family and my wife. Be proud of having so many PhDs who got married while carrying on their research.

Cynthia you have been a great supervisor and a great friend - I could have never wished for more. I arrived with my Italian attitude to ask for friendship from everyone, including you. While this might have clashed with your position as a supervisor, you did not run away and you embraced the challenge of dealing with this side of me. To show how you have done this greatly, let me report some facts below.

During my first year, you welcomed me at your place to cook tons of pasta for the Christmas lunch of our group. I do not recall for which deadline exactly, but I remember that once you prepared dinner for me and brought it to the office when I was stuck there. For ISMIR 2014, we took some time to visit Taipei and the surroundings, including climbing steep hills[1] and launching lanterns in Pingxi.

[1] Before leaving, Alan asked me to make sure that you were safe.

You came to Italy for my wedding, having accepted the challenge of surviving a crazy trip. You flew to Milan to join friends of mine who were driving to Tuscany; then you went back to Holland from Bologna, getting a car ride from some other friends of mine. When my wife was walking towards the altar, you amazingly played and sang the Hail Mary by Bach-Gounod accepting to use my Casio piano - sorry for that, I still feel ashamed, but I could not provide you with a better piano and avoid taping the pedal on the ground as tight as I could.

When the time to move to Sweden came, my wife and I were quite worried about moving abroad again just after our wedding. You then came to our place for dinner, listened to our fears, shared your experience when you had to make similar choices, and encouraged us to go and check whether the opportunity in Sweden could work for us. And you did not just abandoned us in the middle of a dark and cold Swedish winter. You also came to Stockholm with your mother to visit me at Google, have dinner at our place and go sightseeing when the temperature was almost -20°.

There are many other examples of the great care you took for your first PhD student, but it would take another book to mention all of them. Just let me end by mentioning how fun it was to help with teaching, student projects, and obviously PHENICX. Working with you has been extremely enriching and inspiring. Thanks so much for all of this!

Another key person I would like to thank is Martha. Let me confess my secret wish of having you as an additional supervisor. You were the first person I met from the MIR lab. The first time was in 2010 in Florence, when my English was extremely poor and I could never have imagined leaving my hometown. At the end of your workshop on spontaneous conversational speech retrieval I told you “see you soon”. While something like “have a safe trip back to Holland” would have been more appropriate, it ended up being a prophetic greeting.

I am really thankful for our chats. Even if officially I was not a student of yours, your door was always open for me. When my wife and I had to decide whether to accept the internship opportunity at Google, you helped us not to get stuck with our fears. And I will never forget our trip with Karthik and Jaeyoung to California. Especially when we were in Menlopark and you had to put your whole arm in the sewer, searching in the darkness for the Airbnb owners’ damn cat with Jaeyoung, worried that firemen would have had to destroy the street to bring the beast back home.

Saskia, you welcome people and take care of everyone in the warmest and sweetest possible way. And in my case you have done that surely beyond what TU Delft asks. In fact, you extended your help, and more importantly your friendship, to my wife. We really enjoyed it when you showed us Den Haag and its sorroundings, came over for dinner with your daughter and came to Italy for our wedding. Thanks so much for your friendship and keep helping new employees in this way.

Without Raynor, the journey would have been much much harder. You were always keen to help me with translating and filling in Dutch forms, with my old rusty car, meeting for a coffee and sharing ideas on our experiments, especially all the times I got stuck. I also learned a lot about the Dutch culture through you and moving to Holland was less of a cultural shock.

I am so grateful for having had Karthik as a colleague and, more importantly, as a friend. Our trips abroad have been great occasions to get to know each other better and

every time it was amazingly enriching. Working together helped me face challenges a lot, looking at your patience helped myself become patient. This is why for my PhD defense I asked you to be my paranymph.

Babak and I had the best trip abroad that PhDs can ever have. I keep telling people that the fun (and the fear) started from our very first step on Russian territory, when Babak chose the wrong taxi driver who, in the end, was not the thug we expected. I will not tell here of all our Russian adventures, but just want to express my gratitude to Babak for our chats and the fun we had in Kazan and in Moscow.

My first year as a PhD student was probably the toughest, but the friednship of Christoph, Stevan, Viktoria and Michael also helped me to find my way. And I also want to thank all the other colleagues for the chats, lunches and social events we had. Besides complaining about Dutch food and the canteen (and sharing Italian recipes), it has been a great chance to get to know each other better. So a big thank you to Alessandro Ibba, Nino, Carsten, Xinchao, Peng, Yue, Wen, Bo, Jeroen, Ernestasia, Huijuan, Yi, Christina, Judith, Jaeyoung, Jaehun, Soude, Cunquan, and Xiuxiu.

One bit of luck I had with my work at TU Delft was being part of the European project PHENICX. I would like to thank Marcel from the Royal Concertgebouw of Amsterdam. Your enthusiasm and your help significantly contributed to making this thesis possible and it has been so enriching for me to learn about the beauty of classical music. I am also grateful for the collaboration with Ron and Bauke from VideoDock; your ambition helped me to find relevant problems to work on. Also thanks to Maarten Grachten from OFAI, writing deliverables together was great even if we had to stay up late at night. Finally, a big thank you to the PHENICX colleagues from the Music Technology Group in Barcelona. Being a visiting reasercher was great - in just one month I was able to learn a lot more about Music Information Retrieval. Thanks Emilia and Alba for having welcomed me and thanks to Marius and Alvaro for the time we spent together. Also thanks to all the friends who welcomed me in Barcelona: Alessandra, Ignacio, Virginia, Laura, Gloria, Albert, Mario, and Marcos, who sadly passed away[2].

Back to TU Delft, I now want to thank Rafael. Seeing you in the corridor smiling every day has strengthened my desire to enjoy my job more and more regardless of its difficulties. Even when you have important deadlines and things do not go as planned, you always trust that the circumstances are for a greater good. Thanks for your empathy and for your precious advice.

And last but not least from TU Delft, a big thank you to my piano teacher Marlijn Helder. Your lessons and the events you organized were amazing, but more importantly you taught me that feeling the pressure of not making mistakes while playing is not worth it, and it prevents you from enjoying the moment. This has been a great lesson, and it helped me with my research and it still helps with my new job.

Father Dick from Maria van Jessekerk is one of the greatest gifts I received. You became my spiritual guide, always available for all the questions and burdens I needed to share. You guided me, especially during the preparation of my wedding. I am so happy that many TU Delft students had the chance to meet you, I really miss your guidance.

This whole journey has been accompanied by several dear friends spread all over the Netherlands. Gabriëlle e Thérèse, you were among the first people who I met and

[2] http://www.marcospou.com/

you welcomed me when everything was new. The love you put into preparing delicious dinners, walk and bike tours and many other activities when we all met together was striking and surely made my relocation to Holland smoother. Meeting an Italian who had been living for a long time in Delft was great. Thanks Paolo Tiso for your invitations and your tips; listening to your stories of cultural difference was fun and helped me to deal with some incomprehensible Dutch customs. Isabel, meeting you on campus for lunch every now and then has always been a pleasure; I could learn about architecture and we could share how we were doing. Also thanks for taking rides with me when we had to go all the way to Brabant to meet our friends from the South (yes, I am thankful even for those times you fell asleep). Then I want to thank Cecilia and Tom. Both of you have been a great example for the way you dealt with personal and professional challenges. Thanks so much for your friendship, my wife and I miss you a lot.

The list continues with three key people I met in Holland: Attilio, Luca and Massimo. In different ways, you helped me to become the man I am today, making important choices and remembering that we should never doubt that all challenges are given for a good reason. Without your generosity and your wise guidance, I would not have dared to do as much as I have done both professionally and with my personal wishes. Besides them, there are many other people who accompanied me throughout my Dutch journey: Father Michiel, Pieter, Kenny, Thomas, Remco, Angelo, Anne, Gregorio, Mirka, Luca, Paolo, Aga, Peter and Victoria, Carlo and Anna, Francesco and Bernardette, Paul and Laura, Carlo and Emma, Paolo and Silvia, Nadine and Joshua, Rafael, Jorge and Macarena, Stan and Remco, Manuele and Gloria, Maddalena and Carlo, Marjolein, Martijn and Angelique, Serena and Filippo, Monica and Mattia, Peppe, Patrizio, Gianni and Claudine, Annette and Wim. I hope I did not forget anyone!

Then, my PhD adventure included a parenthesis in Sweden. Contrary to any expectation I might have had, the dark and cold Swedish winter became an enjoyable experience, mainly because of the dear friends we have made. First of all, I would like to thank the WebRTC Audio team for having hosted me and for having allowed me to work on amazing stuff and learn a lot. In particular, I want to thank my host, Minyue. I keep telling everyone that you spoke about the most beautiful thing a colleague can speak about, that is “if you’re only interested in your project, then working together is not so interesting for me; what I care about the most is that we become friends”. You truly wanted that, you advised me well and also invited me to meet in town. Finding colleagues like you is rare and I am so happy that I am now back in Stockholm working with you again. Another great friend I made is Armando, who was the other only Italian in the office at that time. Next to drinking coffee, being called “pappa e ciccia” and playing Rally on playseats like there is no tomorrow, I truly enjoyed our long conversations and chatting every day since the end of my internship and until I came back to Stockholm. I also want to say a big thank you to Julia, who hosted me and Iris in her cozy place in Vällingby and made us truly feel at home. Also, I want to thank Sara, Andrea, Matteo, Bessy, Silvano, Max and Ilaria. You welcomed us and it is also because of all of you that my wife and I eventually opted for Stockholm as our new home.

Even if far away from home, I was so lucky to stay in touch with my dearest friends from Italy still. Marco, the way you take care of your friends living abroad is amazing, our friendship shows that physical distance is not a barrier at all. Thanks for all the calls

and that time when you shared those notes from Father Giussani about the purpose of working. I had to read them every morning for months to truly discover that the value of our job goes far beyond solely scientific findings. Beppone, you have been mentoring me since my Bachelor's thesis and you kept doing that even when I moved to Delft and I was not a student of yours anymore. And as you have done in the past, your supervision was not just scientific, but embraced me as a person. Thanks for your precious friendship throughout all these years. Alex, you have always been next to me, available for sharing joys and trials, reminding me to be patient. You kept me connected to my home land, organising vacations and dinners everytime I came back. Thanks to you, I felt much much less far from home. I want to also thank Don Filippo, you took care of my wife and me, always available for video calls and meeting us everytime I came back. In this way, I did not perceive working in Holland as an obstacle for my vocation.

This part is for my dearest friend Francesco. I do not have siblings, but I guess that the best word to call you is brother. It is a miracle that we happend to move to the Netherlands about the same time. It was our first experience abroad, a cultural shock, being excited, crying, laughing, helping each other to keep looking for beauty, even when the rain was isotropic and we could not understand Dutch people's behavior. The list of things we have done together is endless, but even longer is the list of things I have learned from you by looking at how you did not give up. My Dutch journey, and thus this thesis, would have been much harder to complete without you.

I want to thank my parents. You have been extremely generous and brave. I am thankful because you wish to see me happy much more than having me physically close. Everytime I come back home, I see how much you care by the effort you put into arranging amazing dinners all together. If I ever become a father, I wish to be as brave as you are.

Iris, my wife. You have made the biggest sacrifice for love. I still ask myself whether asking for so much was the right choice. You have generously accepted all of this to see me happy. And without your support, I would have collapsed several times. Whatever we have achieved so far, including this thesis, we both must be proud of it, fifty-fifty. You did not refuse circumstances because they were different from your expectations, you always knew that there was something we could learn. You sustained me continuously.

CURRICULUM VITÆ

Alessio BAZZICA

18-06-1983	Born in Florence, Italy.

ACADEMIA

2013–2017	Ph.D., Computer Science Multimedia Computing Group, Delft University of Technology Delft, The Netherlands
June 2014	Visiting Researcher Music Technology Group, Pompeu Fabra University Barcelona (Spain)
2008-2012	M.Eng., Computer Engineering Università degli Studi di Firenze Florence, Italy
2012	Visiting Student Multimedia Information Retrieval lab, Delft University of Technology Delft, The Netherlands *Thesis:* Automatic Music Video Generation *Promotor:* Prof. dr. A. Del Bimbo
2002-2008	B.Eng., Computer Engineering Università degli Studi di Firenze Florence, Italy
2008	Graduating Student Media Integration and Communication Center Florence, Italy *Thesis:* Sport Videos Audio Classification *Promotor:* Prof. dr. A. Del Bimbo

PROFESSIONAL WORK EXPERIENCE

2017–present	Software Engineer Google Sweden AB Stockholm, Sweden
2016	Ph.D. Intern, Software Engineer Google Sweden AB Stockholm, Sweden
2002-2013	Software Engineer Polimoda Florence, Italy

Music is not just experienced by listening to audio recordings. We also watch it, for instance at a live concert or on YouTube. From a multimedia content analysis perspective, at first video recordings of a music performance look messy and hard to be exploited, because they consist of unconstrained and unstructured visual content. However, if we find ways to understand complex scenes and extract musically relevant cues for each featuring musician, the existing digital music platforms can be further innovated bringing a more informative and non-linearly accessible experience to users which accommodates multiple perspectives on the content.

This thesis sheds light on the challenges of real-world data analysis for vision-based Music Information Retrieval; this is done by presenting a feature named Motion Orientation Histograms for unsupervised musician movements analysis, a score reduction method based on Playing/Non-Playing (P/NP) matrices to be used for performance-to-score synchronization, a semi-automatic musician-wise P/NP annotation system, a face clustering method for quasi-static scene videos, and a supervised approach based on multiple-stream convolutional networks for note onsets detection of clarinetist videos.

Even though the vision-based Music IR field remains a largely unexplored territory, which requires multi-disciplinary research, our outcomes show the benefit of tackling the challenges of real-world data. As opposed to oversimplified data, unstructured and unconstrained music videos reveal concrete opportunities to deploy new applications on digital music platforms.

ISBN 978-94-6299-807-0