UNIVERSIDADE DE LISBOA
FACULDADE DE CIÊNCIAS
DEPARTAMENTO DE INFORMÁTICA

EMOVIDEO: AUTOMATIC PREDICTION OF EMOTIONAL RESPONSES TO VIDEOS USING ITS CINEMATIC FEATURES

Silvana Moreira Graça

Mestrado em Engenharia Informática
Especialização em Engenharia de Software

Dissertação orientada por:
Prof. Doutor Manuel João Caneira Monteiro da Fonseca

2021

Acknowledgments

First, I'd like to acknowledge my supervisor Manuel Joao Caneira Monteiro da Fonseca for being a supportive and patient force through this journey, the completion of this thesis would have been impossible without his constant guidance. Then, I would like to thank Soraia Meneses Alarcão for offering her knowledge and critical advice over our work. I'd also like to thank LASIGE for offering me remote equipment to work. I am grateful for my best friend João Ferreira, who helped me both focus and be less of a hermit during the course of this work. Last but not least, I want to thank my parent who were there for me everyday, encouraging me to continue studying.

Dedicatória.

Resumo

Os filmes conseguem evocar emoções intensas na sua audiência e são comumente usados no campo da psicologia para estudar emoções humanas. Um grande apelo de ver filmes é que estes nos permitem sentir e expressar emoções, sejam positivas ou negativas, num ambiente seguro.

Com surgimento dos serviços de streaming, os anúncios personalizados e algoritmos de recomendação para disponibilizar conteúdos de média a utilizadores, prever a resposta emocional do espectador perante estes conteúdos de média audiovisual tem-se tornado uma área de pesquisa de cada vez maior interesse, já que pode ter extensas aplicações nestes serviços. No entanto, um dos tópicos que raramente é abrangido nesta na previsão de resposta emocional são as técnicas cinemáticas. Normalmente, para provocar uma resposta emocional, os cineastas usam propositadamente uma variedade de técnicas durante a filmagem e edição de um filme. Para além disto, muitos destes trabalhos usam deep learning apenas para extrair e selecionar uma variedade de características do vídeo, o que significa que estes métodos não fornecem informação sobre as características e os específicos de como estas influenciam a emoção.

Com este trabalho pretendemos criar um método que extrapole as respostas emocionais do espectador de um vídeo, tendo como base as suas características cinematográficas. O nosso primeiro foco de estudo foi identificar as principais técnicas cinemáticas usadas por cineastas, dentre destas determinamos o tamanho do shot, iluminação, posição da câmara e movimento da câmara como as mais importantes. Depois estudamos como estas realmente afetam os espectadores emocionalmente e métodos existentes para extrair estas de vídeos, para que possam ser usadas como features na previsão de emoções. Com o nosso estudo descobrimos que características cinemáticas podem mudar para cada shot de filme. Por isso, começamos por dividir o vídeo em shots com um modelo treinado para detetar os frames que limitam cada shot no vídeos, como diferenças substanciais em frames ou frames pretas. Estas shots são depois divididas em frames, key frames, frames de fluxo, subject frames e background frames. As frames mais importantes de cada shot (key frames) foram extraídas dividindo o vídeo em segmentos e selecionando um frame de cada. Frames de fluxo, onde o movimento entre dois frames subsequentes é destacado usando dense optical flow. Por último, obtemos frames onde o objeto no foco da câmara

(subject frames) e o fundo (background frames) foram separados usando salient object detection.

De seguida, extraímos as características cinemáticas de cada shot. Para determinar a iluminação calculamos a média do contraste de todas a key frames na shot. O dataset MovieShots contém 33 mil shots classificadas com o movimento (estático ou uma combinação de movimentos pan, push, pull e vários em simultâneo) e a posição da câmara (extreme close-up, close-up, medium shot, long shot, extreme long shot). Nós treinamos dois tipos de modelos com redes neurais convolucionais (CNN), um que prevê o movimento da câmara no shot e outro que prevê a posição da câmera. Para obter os melhores resultados com estes modelos, treinamos e testamos vários modelos com diferentes configurações de rede, parâmetros e classes. Para a classificação da posição da câmara, a melhor configuração da rede foi uma combinação de ResNet50 e TSN, que alcançou uma precisão geral de 80 % treinada com frames de cor normais. Ainda assim, vimos uma melhoria nos testes de previsão de atividade quando usámos três classes extras, obtidas da combinação das classes extreme close-up com close-up e long shot com extreme long shot. Quanto ao movimento da câmera, a configuração de rede que obteve os melhores resultados foi um combinação de ResNet50 com Slow-Only. Duas destas redes foram treinados, um com flow frames e outro com background frames. A fusão destes modelos alcançou uma precisão de 89 %.

O nosso segundo foco foi criar modelos que prevêem valores de valência e atividade. Com este propósito, o tamanho do shot, iluminação, posição da câmara e movimento da câmara, em conjunto com as durações dos shots, foram usadas como features com métodos de classificação e regressão. Para avaliar a nossa solução, apresentamos os resultados de três tipos modelos de valência e atividade (um que classifica atividade/valência como alta/baixa, um que classifica em quadrantes e um que estima valência e atividade) com seis datasets de referência (DEAP, AFEW, Cognimuse, Media16, EMOVIE e EMDB). Ao treinar cada modelo testamos vários algoritmos de classificação e regressão com grid search e leave-one-out cross validation, para chegar aos melhores resultados.

Para os modelos de classificação existe um grande embalanço entre classes, por isso experimentamos com alguns métodos de oversampling e undersampling para aplanar as classes. Nos modelos alta/baixa, alcançamos precisões entre 60 % e 86 % para atividade e entre 62 % e 79 % para valência. Já os modelos de quadrante obtiveram F1 scores entre 0,79 e 0,38. Para os datasets DEAP e MediaEval16, obtivemos resultados de classificação semelhantes ou superiores aos de trabalhos anteriores que reportaram estes valores.

A seguir, discutimos os resultados dos nossos modelos de regressão. Os valores de PCC que obtivemos estão nos intervalos [0,1, 0,39] e [0,4, 0,69], o que significa uma correlação baixa e moderada com a variável alvo, respetivamente. Estas observações são consistentes com os valores de PCC reportados por outros trabalhos. No entanto, tendo em consideração a relação entre PCC e MSE, notamos a tendência dos nossos valores de

PCC serem menores quando comparados a outros trabalhos, mesmo quando os valores de MSE são melhores do que trabalhos com PCCs semelhantes. Isto sugere que as nossas características têm uma correlação não linear com a valência e atividade.

No geral, encontramos resultados melhores na previsão da atividade do que na valência. Isto é consistente com outros artigos e pode ser porque os sentimentos positivos ou negativos estão mais relacionados à história e aos personagens, enquanto a atividade pode ser mais fácil de prever com features visuais. O nosso método teve resultados comensuráveis com trabalhos anteriores, especialmente ao medir o erro. Dado que usamos apenas features cinemáticas, isto mostra que estas afetam a valência e a atividade dos seus espectadores e podem ser uma ferramenta valiosa na previsão dos seus valores exatos.

Palavras-chave: features cinemáticas, classificação de emoções, estimação de emoções, rede neural convolucional, análise afetiva de conteúdo de vídeo

Abstract

Movies can evoke intense emotions in their audiences and are often used in the field of psychology to study emotion. Predicting how video content affects viewer's emotional response has become a popular area of research over the past years due to its extensive applications. In order to provoke an emotional response, film makers use a variety of techniques while of filming and editing a movie. However, there are not many studies on how these techniques affect viewer's emotional responses and if they can be used solely to predict these responses. With this work we developed a solution that predicts emotional responses to videos using cinematic features. To accomplish this goal, we started by studying cinematic techniques and their effects on viewer's emotions. With this information, we decided to focus on shot length, key lighting, shot type and camera movement in this work. Then, we experimented with different methods to extract these features from videos. First, we used a trained model to segment videos into shots, and then segmented these shots into frames, key frames, dense flow frames and frames with the subject and background isolated. The key lighting was calculated from the contrast of each key frame. We trained and tested models to classify shot type and camera movement, with different Convolutional Neural Networks, parameters, types of frames and labels. The best results achieved were 81% for shot type and 89% for camera movement. Lastly, we created models to predict valence and arousal values with classification and regression algorithms using all the extracted cinematic features. Overall, our method had results close to previous works, especially with error metrics, with only cinematic features. This shows they affect viewers' valence and arousal and can be a tool in predicting exact values, while providing interpretability.

Keywords: cinematic features, emotion classification, emotion estimation, convolutional neural networks, affective video content analysis

Contents

List of Figures

List of Tables

Chapter 1

Introduction

In this chapter we present our motivation, the intended goals, a brief description of our solution to predict the emotional state elicited by films using cinematographic features, and the structure of the document.

1.1 Motivation

Movies can offer us complex emotional experiences, so much that it is common in psychology, to use movies to study human emotions. A large appeal of watching movies is that they allow us to feel and express emotions, either positive or negative, in a safe and even social environment. In order to give audiences the best experience possible, film makers have developed cinematic methods to tell stories in the most immersive way.

Video affective analysis, an area of research on predicting how multimedia content stimulates a viewer's emotional response, has become a popular topic of research over the past years. This is due, for example, to the growth in availability of media content for consumers, the rise of streaming services, and the consequent need for classification and delivery of personalized content.

There have been works on video indexing, movie browsing, movie summarization, personalised recommendations based on mood, and advertising. However, seldom do these studies focus around the cinematic methods purposefully used by film makers, including lighting, camera motion and shot type. Moreover, a lot of these works use deep learning with a wide variety of features, which eliminates the interpretability of their outcomes.

1.2 Goals

The main objective of this work is to create a method that extrapolates the viewer's emotional responses to a video, using its cinematic features. To achieve this, we defined the following secondary goals:

- Identify what movie features are used by film-makers to evoke emotions in their viewers and understand how these features emotionally affect viewers in psychological studies, so these can be used as features in the classification;

- Create methods to detect the relevant cinematography features from a video;

- Develop models that use the extracted features to predict the emotional response of viewers, in the form of values of valence and arousal;

- Compare our model with existing approaches, specifically with those that use visual features as well.

1.3 Developed Solution

The solution developed in the context of this work consists in a system that is able to identify the emotional impact of a video using its content as input. In order to accomplish this we divide the video into shots, extract its cinematic features (shot length, key lighting, shot type and camera movement) and input them on models that predict valence and arousal values.

First, we use a trained model to detect shot limits in videos, like substantial differences in frames or black frames, and extract frames and key frames from each shot. Besides these frames we also extract frames where the subject in camera focus was isolated from the background, obtained using salient object detection. Additionally, we used dense optical flow to obtain flow frames, which are frames where movement in subsequent frames is highlighted.

Then we use deep learning to train two models, one that estimates the probable shot type of the shot and another that estimates camera movement. As to achieve the best results with these models, we trained and tested several models with different network configurations, parameters, labels and types of frames. To determine key lighting we calculate the median contrast on the key frames in the shot. These features along with the shot length were used to create different models to predict valence and arousal.

Since our solution evaluates cinematographic features separately, we can know how the features influence emotional reactions more clearly, and allow film makers to use it as an informational tool.

With the completion of this thesis, we developed a system that achieved good results with benchmark datasets, close to existing works and more interpretable. Plus, we are able to demonstrate that cinematographic features can be used to create a model that predicts emotional response, with a good accuracy.

1.4 Structure of the document

This document is composed by four more chapters. Chapter 2 describes emotional models, essential machine learning algorithms, film concepts and conventions, and studies made on how film affects its audience. Furthermore, it discusses existing works on how to automatically detect the necessary film features and previews attempts to predict emotional responses from video. Chapter 3 describes datasets, algorithms and evaluation metrics used in our solution and experiments. It also presents our solution to predict valence and arousal values from movie segments using only cinematic features. Chapter 4 presents and analyses our experimental results and compare them to existing solutions. Finally, in Chapter 5 we make a summary of the dissertation and present our conclusions.

Chapter 2

Background and Related Work

This chapter discusses film grammar, existing techniques for extracting film features and approaches for predicting emotions from videos. Section 2.1 describes low and high-level features in films and what emotional responses the director intends to convey when using them. Section 2.2 overviews emotion models, specifically those most commonly used. Section 2.3 presents some studies about film features and how they can influence emotions. Section 2.4 describes existing works and frameworks for extracting film features from a video. Finally, section 2.5 discusses previous methods used to estimate the emotions elicited by videos.

2.1 Film Grammar

In a film there are several components to telling a story. The ones that most people remember are dialogue contents and characters' emotional expressions. These are called high-level features, which are hard to define in strictly technical terms and to predict computationally. Nevertheless, there have been some works that tried to use them in affective analysis [49] [40].

The least noticed components are low-level features, which are physical and quantitative aspects that occur in function of the narrative. Throughout cinema history, film makers have slowly established cinematographic rules in order to best tell a story to their audience with what they can show on a screen. This includes framing, lighting, shot length, color, etc.

2.1.1 Low-Level features

In this section, we describe both visual and audio low-level features. When it comes to visual features, we will focus on the following: shot duration, shot scale or type, lighting, and movement. All these features exist in a shot, which is the smallest unit of photographic coverage of a person, action, or event. A scene is a series of shots, usually in one time and place, separated by transitions like cuts, dissolves and wipes.

Shot Length

The length of a shot dictates how well people read and absorb its visual message. The briefer the shot, the less opportunity people have to extract information. Film-makers can use this to manipulate the audience's mood. Usually, action scenes have more short shots, which are suppose to transmit a sense of urgency and distress, while dramatic scenes give the viewer time to completely absorb the character's emotion or to build expectations for the next shots. Lately, the overall average length of shots has been decreasing and there is a tendency towards local clusters of similar shot length, possibly to match the fluctuations in human attention [16].

Shot Type

There are a variety of shot types, but they can be divided into three categories: long, medium and close-up [76].

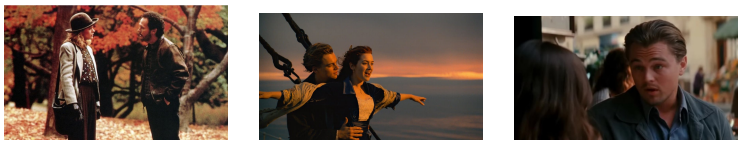
A Long shot (LS) or Wide shot (WS) frames a figure fully visible and its relation to the environment, but can also be close enough to show its clothing, movements, and general facial expressions. Extreme Long Shots (ELS) show a large view of the exterior and are usually used to establish the scenes place and time of day. They might also show very general information about a person or action. Figure 2.1 shows the two types of shots.

(a) (b)

Figure 2.1: Examples of long shots. a) Extreme Long Shot from *Thelma and Loise* b) Long Shot from *No Country For Old Man*

A Medium shot (MS) frames a figure waist up and are the most common since they can capture the character speaking, or performing a small action. Medium Long shots (MLS) are a little wider, but still cut off a body part of the figure. A common iteration of MLSs frames a figure at the knees and is called a "cowboy" shot, since it was used prominently in Western movies. Medium close-ups (MCS) show a figure from the chest up. Figure 2.2 illustrates these types of shots.

A Close-up (CS) shows a magnified view of whatever is being filmed, as shown on Figure 2.3. It can be used to show some important information to the viewer. Close-ups of facial expressions, are meant to force the audience to be intimate and empathize with the character. An Extreme close-up (ECS) only shows one detail and lacks any points of reference to its surroundings.



(a) (b) (c)

Figure 2.2: Examples of medium shots. a) Medium Long Shot from *Harry Met Sally* b) Medium Shot from *Titanic* c) Medium Close-ups *Inception*

(a) (b)

Figure 2.3: Examples of close-up shots. a) Close-Up Shot from *The Shinning* b) Extreme Close-up Shot from *X-Men: First Class*

Camera Motion

Motion in a shot can be divided into the captured on-screen motion and camera movements. The standard camera movements are pans, tilts and zooms. A pan is a rotation of the camera lens horizontally and they can serve to encompass large swaths of landscape. A tilt is a rotation of the camera lens along a horizontal axis. A combination of these creates a diagonal rotation. Most tilts are usually used after an expectation for the movement has been set up, for example if a character looks up slowly, the next shot could be a tilt up to reveal what they're seeing. A zoom is a change in lens' focal lengths to look like the camera is moving closer or further away from the subject.

Color

Color has been quantified in a variety of spaces, but there are a few attributes that affect people. Saturation is the boldness of the color and hue is the color itself. Some directors give particular colors meanings, for example red has been used to denote danger or passion and blue for cold or sterile environments [76].

Lighting

Choices of lighting are often motivated by where light sources would be in the reality of the scene. In a shot, the main source of light is called a key light and it is put near the subject [76]. High Key lighting allows the viewer to clearly see all of the visual space with no shadows, and is often used to generate the lighthearted atmosphere. While, low key lighting has high contrast, dark shadows, and half lit sets and faces. This is used to

create sad or frightening scenes [6].

Audio

There is a large variety of low-level or paralinguistic audio features that have been used in affective analysis, but two categories are prosodic and spectral [22]. Prosodic features like pitch, fundamental frequency (F0), loudness, energy and voice-quality are the most employed. F0 is the frequency at which vocal chords vibrate in voiced sounds, while pitch is used to refer to how the F0 is perceived, low or high [74]. Voice-quality is the breathlessness and tension in a voice [8]. Spectral features include Mel Frequency Cepstral Coefficients (MFCC), chroma, Zero Cross Rate (ZCR) and Linear predictive coding (LPC). MFCC are coefficients that collectively form a short-term power spectrum of a sound, which approximates the human auditory system more closely. The Chroma feature represents sound in relation to the 12 pitches. Since in noise, energy is uniformly distributed in the chroma, and in music it is highly concentrated in the frequencies played, the chroma difference can help classify sound as music or environment. ZCR is the number of times the sound signal crosses the zero axis in a certain period, and it is a good way to distinguish between music and speech or environmental sounds. LPC is a widely used technique in audio signal processing, especially in speech signal processing [36].

2.1.2 High-Level features

The main high-level audio features used for classification are lexical features. These include the semantic meaning of words, as well as disfluencies and non-verbal cues like pauses (time the speaker is silent), filled pauses, fillers (verbal filled pauses), laughter, stutters, and audible breaths. As for visual high-level features, many studies have been made. Yi and Wang [85] kept track of on-screen motion information, and some information about objects. Given previous studies on semantic analysis of images, it could be possible to extract emotional information from each key-frame and use it along with the other features.

2.2 Emotion Classification

Since, current emotion models vary considerably in number of emotions they represent and principles used [63], in this document we only describe discrete and dimensional models.

Discrete basic emotion models theorize that there are a limited number of fundamental emotions, anywhere from 3 to 11 [53]. A lot of these theories are based on studies by Tomkins, Ekman and Izard who found some facial expressions and corresponding emotions that seemed universal. Ekman proved that happiness, sadness, anger, fear, disgust

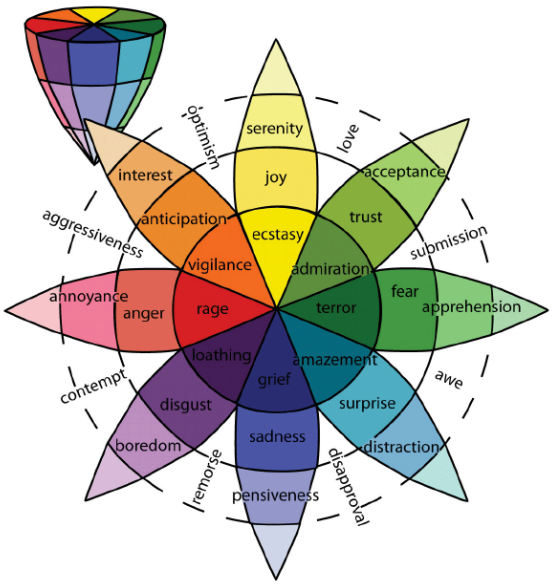


Figure 2.4: Plutchik's wheel of emotions

and surprise were expressed the same by the Japanese and Americans [25]. Plutchik created a model analogous to a color wheel placing similar emotions next to each other and opposites at 180 degrees, and other emotions are a mixture of the 8 primary emotions [53], as shown in Figure 2.4 .

Unidimensional models assume that one dimension of emotion is enough for emotional analysis. For example, knowing if an emotion is negative or positive is still one of the most important ways to distinguish feelings [23] [32]. On the other hand, multidimensional models consider that emotional states rest on two or three dimensions. Wundt was the first to propose a model with three dimensions (evaluation, activity and potency) [45]. Osgood et al. developed a semantic space to measure the meaning of words [51]. Here pairs of opposite emotions like happy/sad and lighthearted/dark were associated with the Evaluation dimension, serious/humorous for Potency, exciting/relaxing and fast-paced/slow-paced for Activity. Mehrabian and Russell proposed the PAD emotional Cartesian space [47] [59] [46] where each axis represents one of the dimensions valence or pleasure, arousal and dominance. Valence indicates the degree of pleasantness of a certain feeling and varies from positive to negative. Arousal measure the excitement of the feeling and ranges from excited to calm. Finally, dominance defines the level of attention of a feeling from dominant to submissive. Since, dominance is the least comprehended dimension, some researchers use spaces with only the dimensions arousal and valence[58] [42] [1]. Dimensional models have the advantage of being easier to track than words when there are variation overtime and they quantify a wider range of emotions. However, none of these models account for the possibility of people feeling two emotions at the same time.

The hourglass model [7], depicted in Figure 2.4 is a reinterpretation of Plutchik's model where primary emotions are organized around the dimensions (Pleasantness, Attention, Sensitivity and Aptitude), with intensity varying on vertical values and activation on radius. This model allows classification in a discrete and dimensional way, and for four emotions to be expressed at the same time.

2.3 Emotional Effects of Film Features

Various psychological investigations have found that movies can have a specific emotional impact in most individuals, to the point that they have become a useful tool in psychological studies of emotion [50]. Smith tried to explain the emotions people derive from film [69], and postulated that emotional associations provided by cinematic features are crucial to film emotions, and are consistent across a range of audiences, despite individual variations.

2.3.1 Shot Length

Viewers have the tendency to look at the center of the screen, and after a cut between shots, they reorient their gaze to the center [77], where most of the action happens. Hochberg and Brooks found that people paid more attention when watching abstract or meaningful pictures changing at faster paces (1 to 0.5 seconds per picture). They related this to the concept of "visual momentum", an impulse to learn more visual information [28]. Lang et al. made a related study with one minute TV clips with fewer than three scenes. They classified the number of cuts between shots, in a scene, as: 0-7 cuts is slow, 8-15 is medium, 16-23 is fast, more than 24 is very fast paced. Results showed that both shot and scene cuts increased arousal, but changing scenes quickly impaired recognition of information, while fast paced scenes cause an increase in recognition memory [41].

2.3.2 Motion

Viewers can integrate some on-screen motion and camera movement to obtain seamless visual input, so in some papers the combinations of these get called visual activity. There has been a connection found between motion and arousal, while it does not alter valence much, [21]. However, camera zooms are alien to the visual processing, so it stands out to the viewer [76]. Overall, camera movement seems like it might better capture and sustain attention.

2.3.3 Shot Type

Canini et al. [9] annotated the response of participants to shots of types LS, MS and CS. They found that LSs are associated with high arousal and CSs with low arousal sectors

and corresponded with dramatic scenes. However, there was no connection to valence.

2.3.4 Lighting

In regard to lighting, there have been quite a few empirical studies about its impact on people, on a variety of ways like concentration, memory, performance and mood [37] [61]. Huang et al. [31] studied the effects of lighting and found data that partially confirmed that higher levels of low-key lighting increased arousal and negative valence, but the effect sizes of the positive correlations were small.

2.3.5 Color

Mehrabian and Valdez used the PAD model to quantify emotional responses from 76 colors of the Munsell Color System (seven samples of brightness and saturation for each of ten hues) [78]. They concluded that more saturation elicited more arousal and brighter colors cause higher values in valence, and lower values in arousal and dominance, while dark colors are associated with anger, hostility, or aggression. This when colors were used in reasonable and probable situations. Later, Wilms and Oberfeld measured the emotional effects of hue, saturation, and brightness of the CIELAB space, as a factorial experiment [83]. For 30 different colors (all combinations of low, medium and high for saturation and brightness in the hues blue, green, red and gray), they asked participants to fill out Self-Assessment Manikin (SAM) [5] scales with values of 1 to 9 for valence and arousal. These results were complemented with skin conductance responses and heart rate. They concluded that the effects of emotion were dependent on the combination of all dimensions. For example, hue and brightness only affected arousal (the latter in a positive way), while high saturation and chromatic colors led to more positive ratings on the valence scale with high brightness, while achromatic on average caused negative values.

2.3.6 Audio

There have been a few studies on how well humans could infer someone emotional state just from hearing them talk, and all obtained accuracies of around 50%, ignoring possible correct guesses [62], being that sadness and anger are best recognized, and disgust is the worst. Researchers have found pitch and energy related features to be enough to recognize acted emotions [79]. Energy has been positively connected with arousal [71]. In general, works have had success in using all kinds of audio features from all mentioned categories [22],

2.4 Detection of Film Features

This section presents some methods for extracting the previously mentioned features from videos. Most of these methods are proposed in the affective analysis works described in the next section, others are existing software tools or studies.

2.4.1 Shot Length

The fundamental step in determining shot length and all other features is dividing the movie into smaller segments, shots. There have been a variety of algorithms implemented to segment videos into shots.

One simple and common method is to create three histograms, one for each RGB color, with the frequency of color intensities in a frame and compare it with the graphs of consecutive frames, the changed areas are computed into a new image. There are a few variations of this process, some use histogram difference, the squared difference, the difference after equalization and other the intersection of histograms. Teixeira et al. [74] summed these images by creating a Motion History Image (MHI), and used it to estimate the amount of motion of a shot. This amount is measured as a ratio between the number of pixels inside the MHI by the number of pixels outside.

Some methods define effects that mark a cut, like transitions and calculate when they happen. Other methods calculate, for each pixel between consecutive frames, changes like noise, motion, lighting, scale or a transition. Another method compares the edges in two consecutive frames, based on how during a shot transition new intensity edges appear far from the locations of old edges, and old edges disappear far from the location of new edges. Dailianas et al. [17] compared all of these methods. They used four test videos (three with around 30 minutes and one with 60 minutes). Based on each method, a certain threshold was specified for a metric. Whenever the values cross the threshold from below to above, a shot transition is identified. Several methods identify correctly 94% to 95% of the real transitions, but also had a very high percentage of false positives, because of quick motions, sudden variations of the luminance of the image or variations in the value of the metric of dissimilarity between consecutive frames in the case of fades, dissolves, and wipes, causing the upward crossing of the threshold more than once for a single shot transition. None of these algorithms addressed how to distinguish between a fast change of the image in the same shot caused by movement of the camera and a cut or dissolve, which might require object recognition. Another problem is the identification of semantic transitions not associated with shot transitions.

Smeaton et al. tested all of these methods plus works that used machine learning algorithms [68]. They found SVMs to be the most effective with a precision of over 90% for simple cuts and over 75% for gradual transitions. At this point shot segmentation is considered to reach an accuracy that is sufficient for any practical application and MovieNet

tools [29] has an implemented solution to this problem.

Souček and Lokoč [70] used a deep learning approach, with a 3D convolutional architecture named TransNet, to find shot boundaries. After having calculated each shot's boundaries obtaining its length is trivial.

2.4.2 Motion

Kang extracts camera motion by dividing the frame into nine regions and then computes motions using optical flow. The motion phase is quantized into 8 directions and classified into "pan", "tilt", "zoom", and "no camera motion". The motion intensity is also computed, but not many details were given [34].

Wang and Cheong calculate visual excitement based on the average number of pixels that according to human perception change between corresponding key frames [80][85]. Frames are divided into 20 blocks and are considered changed if their CIELUV color space histograms are more different than a certain threshold. Three subjects manually annotated eighty-two 15s video clips of one type or degree of motion, with a scale of 0 to 10 of excitation. The plots of the two measures were pretty close, but was only a good indicator since the clips feature explosions, large occlusions and special effects.

Yi and Wang used two unique features that kept track of on-screen motion [85]. For global motion, the SURF detector was used to detect key-points. Then for each frame, the dense optical flow is calculated. These points are individually tracked with optical flow at different spatial scales. They use the algorithm vector field consensus to estimate the perspective transformation matrix between frames and then use it to remove camera motion from each of them. To describe trajectories the descriptors Histogram of Optical Flow (HOF), Motion Boundary Histogram (MBH) and Trajectory- Based Covariance (TBC) are calculated into vectors which are individually encoded into a signature vector, and combined in the MKT vector. For local motion, they calculated two-stream Convolutional Networks (ConvNets), one to capture appearance and the other motion clues, using the Caffe toolbox[1] (a deep learning framework)

2.4.3 Shot Type

Rao et al. define the system SGNet to predict a movie shot type, more specifically its movement (static, pan, zoom in and zoom out) and scale (long, full, medium, close-up and extreme close-up), and can generate edited shots [57]. In the SGNet pipeline, a saliency map generator separates the object in focus from the background of several clips from the same shot. These images are input to a ResNet50 network where their features from different stages are fused into a whole image pathway, and in the case of the background, a variance map as well. Then, the classification results are fused for the final prediction.

[1] https://caffe.berkeleyvision.org/

SGNet obtained an accuracy of 87.77% for scale and 83.72% for movement. The greatest advantage of this paper is being clear and easy to implement. In the course of their work, Rao et al. also made available the dataset they created, MovieShots.

2.4.4 Lighting

The features lighting and color can be extracted only from key-frames[2]. Some approaches apply a saliency map to the frame, to help determine the regions that would most attract the attention of the viewer, and only extract lighting and/or color features from these regions. Lighting key as a feature can be calculated by the median level of the brightness histogram of the shot and the proportion of shadows, meaning pixels with lightness below a certain threshold [74].

2.4.5 Audio features

Sun and Yu [71] implemented software that extracted audio features, related to timbre, melody and tempo, based on MPEG-7 audio content descriptions. Wang and Cheong [80] used SVMs with the features chroma difference and low short time energy ratio to classify the audio as music, speech, environment or silence.

Teixeira et al. calculated several low-level audio features [74], like irregularity of the spectrum, which is the sum of the spectral bins minus the average of all the bins, or the spectral roll-off that is the frequency below which 85% of the sum of magnitudes is contained. They also detected F0 using the Component Frequency Ratio technique, where the spectrum peaks with the highest amplitudes are compared two by two to find a harmonic proportion among them, and the MFCC, by applying a Fourier transform and the Mel scale to the signal, and then obtaining the cosine of the logarithm of the power of the amplitudes.

OpenSMILE is a toolkit for audio feature extraction and classification of speech and music signals, used in several of the works studied [85]. It computes relevant prosodic and spectral features, as well as a few visual features based on OpenCV (HSV color histograms, Local binary patterns, Optical flow, face detection).

Yi and Wang used OpenSMILE to calculate the 1,582-dimensional vector EmoBase10 [85], proposed by the same authors, that is obtained by applying statistical functions to low-level features and corresponding delta coefficients. In another work, they used VGGish, a state-of-the-art audio feature extractor that converts audio input into semantically meaningful 128-dimensional vectors [86].

Muszynski et al. extracted the lexical features crowd-sourced annotations (CSA) and Disfluency and Non-verbal Vocalization (DIS-NV). For DIS-NV, the dialogue was annotated into six categories (filled pauses, fillers, stutters, laughter, and audible breath). For

[2]The FFmpeg tool has implemented functions that can parse a video into key-frames

CSA, they removed stop words from the movie transcript and lemmatized the remaining words using the Natural Language Toolkit. Then a dictionary annotated with valence, arousal, and dominance ratings was searched for the all lemmas [49]

2.5 Prediction of Emotional Effect of a Video

Video affective content analysis is a topic that has drawn a lot of attention in recent years, due to its applications in film making and advertising. In this section we describe the annotated dataset used by the community, and works from two types of approaches: traditional and using deep learning techniques.

2.5.1 Datasets

Xu et al. made the Ekman dataset [35], a collection of 1637 videos from websites like Youtube and Flickr, all annotated with Ekman's basic emotions (happiness, sadness, anger, fear, disgust and surprise). For the VideoEmotion dataset, Jiang et al. downloaded 4,486 videos from YouTube and 3,215 from Flickr, with an average duration of 107 seconds, and annotated all of them according to Plutchik's wheel of emotions [33].

Baveye et al. created the Discrete LIRIS-ACCEDE dataset [3], made up of 9800 clips lasting from 8 to 12 seconds, from 160 movies under Creative Commons licenses, thus allowing the database to be shared publicly without copyright issues. Each clip was rated and ranked by emotional scores of valence and arousal (within the 1 to 5 range). Additionally, the order of excerpts within a film was kept, allowing the study of temporal transitions of emotions. Later, the same authors created the Continuous LIRIS-ACCEDE dataset [2]. They selected 30 feature movies from the dataset and had participants rate each second of their entire length in arousal and valence (from -1 to 1). Over the years, additions to LIRIS-ACCEDE dataset were made for MediaEval workshop tasks.

In MediaEval 15 [67], Wang et al. added an extra 1100 movie clips for the Discrete LIRIS-ACCEDE (for a total of 10,900) and annotated all clips with a binary value to indicate the presence of violence and the class for induced arousal (calm, neutral or active) and valence (negative, neutral or positive). In MediaEval 16 [18], 1200 short clips were added to the Discrete LIRIS-ACCEDE dataset annotated with a valence and arousal (from 1 to 5). While, 10 long movies were added to the Continuous LIRIS-ACCEDE, annotated with a valence and arousal (from -1 to 1).

DEAP [38] has 40 one-minute long excerpts of music videos annotated with valence and arousal ratings from 1 to 9 by 32 participants. AFEW [39] consists of 600 videos extracted from features films, for a total of more than 30.000 frames, each annotated with valence and arousal in the range of -10 to 10. Cognimuse [87] contains 30 minute clips from 12 movies, with continuous annotations of intended arousal and valence, of range

from -1 to 1. EMDB [12] has 52 clips, 40 seconds each, edited from 100 commercial films, each annotated by 32 volunteers and then averaged. The EMovie [44] dataset is composed of 39 excerpts, with approximately two minutes, characterized in terms of valence and arousal from 1 to 9, by 174 participants and averaged by gender.

Dataset	**Videos**	**VA Scope**	**VA Range**
Ekman [35]	1637 online videos	Video	Ekman's basic emotions
VideoEmotion [33]	7701 short online videos	Video	Plutchik's wheel of emotion
Discrete LIRIS-ACCEDE [3]	9800 short clips from movies	Video	1 to 5
Continuous LIRIS-ACCEDE [3]	30 short and feature films	Every second	-1 to +1
MediaEval15 [67]	10900 clips from movies	Every second	high, neutral or low
MediaEval16 Global [18]	10900 clips from movies	Video	1 to 5
MediaEval16 Continuous [18]	40 short and feature films	Every second	-1 to +1
AFEW [39]	600 clips from features films	Shot	-10 to 10
Cognimuse [87]	30 minute clips from 12 movies	Frame	-1 to +1
DEAP [38]	40 one minute clips from music videos	Video	1 to 9
EMDB [12]	52 clips from 100 films	Video	1 to 9
EMovie [44]	39 clips from movies and TV-shows	Video	1 to 9

Table 2.1: Description of existing VA datasets

2.5.2 Traditional Methods

Bayesian-based methods can be used to integrate the temporal dynamics of multimodal data to resemblance the way a movie being watched is interpreted, how a scene feels is linked to the previous feelings that have been elicited. Dynamic Bayesian Networks (DBN) are probabilistic graphical models, where the nodes represent different modalities and the edges denote their probabilistic dependencies. The most used and simple DBN is the Hidden Markov Model (HMM). Autoregressive HMM (ArHMM) is an extension of the HMM, where each node has dependency on two previous observed variables as well as an hidden state. A few older works used HMMs and ArHMMs on affective content analysis of videos.

Kang [34] performed an empirical study and related visual features (shot length, motion type and intensity, color hue and saturation) to the basic emotions (fear, sadness and joy) using HMMs. They made two different topologies: a circular HMM model with one state per emotion plus ”normal”; and a model with four HMMs (one per emotion plus ”normal”). The best result overall was achieved with all features and 4 individual HMMs.

Sun and Yu presented a similar framework that also classified anger and used audio features [71]. They collected a set of 10 movies annotated with arousal curves. These curves are used to detect highlights of emotional content at several time granularities of the video and rate their intensity (strong, mild or weak). Then all highlights are mapped out in a tree and their audiovisual features are extracted and turned into observation vectors. Then 4HMMs (one per emotion) are trained and tested.

Multimodal fusion is the process of combining data collected from various modalities for analysis tasks. There are two commonly used fusion methods: feature-level and decision-level. Feature-level or early fusion, the different extracted features are fused into a single vector that is then analyzed. On one hand, the correlation of features at an early stage can potentially provide better task accomplishment. On the other hand, since the features obtained can differ widely in many aspects they must be put in the same format. Decision-level or late fusion, the features of each modality are classified independently and those results are fused as a decision vector to obtain the final decision. This method has the advantage that each modality decision can use the most optimal classifier model. A hybrid fusion uses both feature and decision-level fusion to get the advantages of both fusion strategies without their disadvantages [55].

Teixeira et al. made several experiments with both visual (shot length, colour, motion, lighting) and audio (prosodic and spectral) features [74]. Their experiments did not result in any method that was better for every emotion, but the most consistently good and effective method used ArHMM models, saliency maps to highlight relevant regions of frames and decision-level fusion.

Support Vector Machines (SVMs) can map vectors from an input space into a possibly infinite dimensional feature space and find the best separating hyperplane therein. Wang and Cheong [80] trained SVMs individually for each class pair of six basic emotions. To classify movies, a take-one-movie-out approach was used, where all scenes from one movies is reserved for testing and the rest is for training. Unlike other works, they found audio features to be more helpful.

Eggink and Bland [24] classified mood of Tv Shows into two dimensions: serious/humorous and slow/fast. The SVMs were trained either for classification or regression, using radial-basis function (rbf) kernels, with the features of genre, spectrum audio features, presence of full-frontal faces, lighting and motion.

Kossaifi et. al [39] collected facial expressions and emotional ratings per frames of videos from a few films. And then tested the efficiency of using these features in pre-

dicting valence and arousal values, using an SVM for regression (SVR). They then made available the valence and arousal ratings for each video in the dataset AFEW.

2.5.3 Deep Learning Methods

Cao et. al [10] used the EEG readings from audiences watching videos from the DEAP dataset. They tried to verify the correlation between EEG signals and the classifications of valence and arousal, using a custom CNN. They found better results with CNNs than with traditional classifiers.

Sivaprasad et. al [66] extracted several features (shot length, histogram of optical flow, histogram of 3D HSV, video compressibility, histogram of facial area, plus audio features) from 5-second segments of videos in the COGNIMUSE dataset. They then trained Long short-term memory (LSTM) based models with all of these features. Authors found that these models have more difficulty in predicting valence, given that it involves more cognitive thinking related to plot and characters.

There is a difference between emotions conveyed by the movie (perceived emotions) and emotions that are actually evoked in the audience (induced emotions). Muszynski et al. [49] were the first to investigate the relationship between perceived and induced emotions. For this purpose, they used existing LIRIS features and collected annotations about Disfluency, Non-verbal Vocalisations in the dialogue, and aesthetic features. They then induced emotions of 8 movies from the dataset. These features and perceived emotions were feed to a LSTM and DBN networks. They found that watching too many exciting, pleasant and dominating scenes may evoke boredom and scenes in which main characters are dominated by dramatic events can cause displeasure. Experiments also showed that including features for the past 3-time steps gives optimal performance, and so LSTM was the best algorithm.

The MediaEval 2016 Challenge "Emotional Impact of Movies" proposed the task of emotion prediction of a short video clip (around 10 seconds) using their LIRIS-ACEDE dataset and the MediaEval 2016 dataset, that together contain 11000 short clips annotated with valence and arousal, from 0 to 5. Out of all the works submitted 2 had the best results. Ma et al. [43] extracted visual features with a CNN and audio features with the Geneva Minimalistic Acoustic Parameter Set. Predictions were calculated with a SVR. Chen and Ji [14] extracted visual features related to color, key objects, object movement and actions, plus MFCC audio features. Regression was calculated with SVRs and Random Forests. They achieved the best results with a late fusion of audio, image and motion modalities.

Thao et. al [75] used Cognimuse and the dataset given in MediaEval 2016, divided their videos into five second clips or segments were the valence and arousal is stable and extracted its features. They used a ResNet50 based network to extract static appearance features, a I3D model for spatio-temporal features, optical flow to obtain low-level motion information, plus OpenSMILE and VGGish for audio features. All feature vectors

are fused through a fully connected layer, and the fused vector is input for the AttendAffectNet model. This is a two-stream model based on self-attention, which is allows the model to recognize the most relevant features and the relations among each other. Their method didn't manage to improve on the results of other state of the art works.

Ou et. al [52] tried to emulate the human brain processing multimodal information. With this purpose, their model four features (visual appearance, motion, audio and tone) are extracted individually with CNNs and LSTMs, then self-attention mechanisms are used select key features and to fuse the modalities. They evaluate their method with the DEAP and MediaEval16 datasets, and demonstrated their method improved the results of other methods that use self-attention and state of the art methods.

extracted several features (shot length, histogram of optical flow, histogram of 3D HSV, video compressibility, histogram of facial area, plus audio features) from 5-second segments of videos in the COGNIMUSE dataset. They then trained Long short-term memory (LSTM) based models with all of these features. Authors found that these models have more difficulty in predicting valence, given that it involves more cognitive thinking related to plot and characters.

Yi and Wang [85] also used the dataset given in MediaEval 2016, and presented a multi-modal learning framework that classified videos according to three features: Motion Keypoint Trajectory (MKT) that depicts long motion information; ConvNets based on Temporal Segment Networks, with one spatial stream that describes static information about scenes and objects and one temporal stream that captures the short-term motion information; and EmoBase10 which contains audio information. Then cross validation is used to select features and an early fusion strategy is used to combine the vectors of these features. They then used a different regression algorithm, and got the best results with an SVM. Also, they found that MKT and ConvNet obtain better results than other visual features and higher performance than the audio feature.

The same authors also proposed an Adaptive Fusion Recurrent Network (AFRN) [86], a network that receives audio, color and optical flow features collected with three CNNs, and tracks previous scenes influence on current scenes to predict the emotions of the viewer. The AFRN has one statistical layer that uses arithmetic means and the standard deviation to reduce noise signals from the input video features. It also includes, two other layers based on recurrent networks and adaptive weights. One combines temporal inputs and the other the multiple features. Overall, the best results they achieved used the recurring layers LSTM.

2.6 Discussion

In this section we discuss the methods to predict the emotional effect of a video described previously, by comparing the methods' features, output type and performance (see Table

2.6).

Several older papers performed studies with low-level features (shot length, motion in frame, color, lighting and audio), which coincide with our focus and are simple and effective [34][71] [74][80]. Eggink and Bland [24] study of the BBC archive was too specific to their data and using genre as a feature requires pre-existent knowledge of the input. All these papers use datasets that are not available to the public and only report results in the dimensions of basic emotions.

Kossaifi et. al [39] and Cao et. al [10] both created and reported some of the best results for the AFEW and DEAP datasets, respectively, so they're relevant for the next chapters, even though EEG readings and facial features used are not in our scope.

The goal of the Muszynski et al. [49] study was to compare perceived and induced emotions, and so the only features studied, not in the original dataset and other works are related to dialogue which is outside of the purview of our study.

The state of the art methods in this area use several properties of frames, like color, movement of the objects in frame and facial expressions, some are extracted and selected with deep learning methods [43][14][52] and others with traditional methods [66][86] [75][85]. None of these works focus only on cinematography features, but we can compare our results to theirs.

Several works find slightly better results when merging audio and visual features, rather than just using visual features, therefore we have given some context about the type of audio features that exist and how they are captured. However, audio features are outside of the goal of this work, but might be explored in future works.

2.7 Summary

In this chapter, we presented some cinematography tools used in film making, both high and low level. We saw that high level features include semantic meanings of speech and images, and can be hard to computerize. While low level features are shot type, camera motion, color, lighting and some descriptors of audio. We also exemplified in what scenarios film makers could use low level features.

Next we explained how low level features have been proven to emotionally impact its viewers. Scenes with a smaller shot length tend to increase arousal and are found mostly in action scenes. There is not a lot of information about how camera movement impacted emotion, but it seems that movement causes an increase in arousal in general. Long shots are associated with high arousal and close-up shots with low arousal. In lighting, a higher contrast increases arousal and lowered valence. Color dimensions affect emotions differently in different combinations. As for audio, pitch and energy are enough to recognize acted emotions, and energy elevates arousal.

Then we described several methods to detect or extract these features. The first task

is to divide the video clip in shots, we presented a few methods that can detect frames that limit shots. To identify the camera motion of a shot, and for that there were a few methods that calculate changes in between frames and calculate trajectories. For shot types, we found a dataset annotated that can be used with CNNs to predict shot type. Lighting can be calculated by median brightness of the pixels in the frame.

Lastly we discussed existing methods to predict emotion from video, with a variety of features and machine learning methods. Some of the older methods use datasets that are not available and output classifications for basic emotions. A few methods use shot length and lighting as visual features, but most use features outside of the focus of this work, for example color, facial expressions, movements or descriptions of objects in frame, dialogue and low level audio features. To extract the features, some methods use the means mentioned, while others simply use deep learning methods. Then, the features are used as input for HMMs, SVMs or a deep learning network with custom configurations.

Paper	Dataset	Feature	Classifiers	Classes	Success Rate
Kang [34]	six movies segmented into scenes	shot length, motion type and intensity, color hue and saturation	4 HMMs	fear, joy and sadness	detention rate 81,3% for fear, 76,5% sadness and 78,4% joy
Sun and Yu [71]	ten movies segmented into shots	visual (shot length, motion, color) and audio (prosodic)	4 HMMs	fear, joy, anger and sadness	precision of 80,9% for joy, 59,1% for anger, 72,5% sadness, 60,0% fear
Teixeira et al. [74]	346 clips with average 112s in length from 24 movies	visual (shot length, color, motion, lighting) and audio (irregularity, ZCR, frequency, roll-off, Tristimulus features, MFCC)	2 ArHMM	anger, disgust, fear, happiness, sadness and surprise	accuracy of 57% for joy, 73% trust, 65% fear, 69% surprise, 66% sadness and 62% disgust
Eggink and Bland [24]	544 tv show clips	genre, visual (presence of faces, lighting and motion) and spectrum audio	SVM	serious/humorous and slow/fast-paced	accuracy of 97% for serious/humorous and 93,5% for slow/fast-paced
Wang and Cheong [80]	36 feature films segmented into 2040 scenes	Visual (lighting, color, motion, shot duration) and audio (silence, environment, speech and music proportion, and sound characteristics)	SVMs	6 basic emotions (Happy, Surprise, Anger, Sad, Fear, and Neutral)	74,69% correct classification rate
Kossaifi et. al [39]	AFEW	visual (facial expressions)	SVR	arousal and valence (both between -10 and 10)	arousal: 4,97 MSE and 0,45 PCC; valence: 6,97 MSE and 0,40 PCC
Cao et. al [10]	DEAP	EEG readings	custom CNN	arousal and valence (both between 1 and 9)	arousal: 1,217 MSE and 0,367 PCC; valence: 1,281 MSE and 0,300 PCC
Ma et al. [43]	MediaEval 2016	visual (extracted with CNN)) and audio (Geneva Minimalistic Acoustic Parameter Set)	SVR	arousal and valence (both between -1 and 1)	arousal: 1,467 MSE and 0,344 PCC; valence: 0,214 MSE and 0,296 PCC
Chen and Ji [14]	MediaEval 2016	visual (Hue Saturation Histogram, Dense SIFT, Dense Trajectory and C3D features) and audio (probabilistic, MFCC)	SVR and Random Forests	arousal and valence (both between -1 and 1)	arousal: 1,479 MSE and 0,201 PCC; valence: 0,201 MSE and 0,419 PCC
Ou et. al [52]	Cognimuse and MediaEval 2016	visual (static appearance and motion) and audio and tone,	LSTMs	arousal and valence (DEAP) between 1 and 9; (MediaEval16) between -1 and 1	(DEAP) arousal: 1,22 MSE and 0,37 PCC; valence: 1,28 MSE and 0,30 PCC (MediaEval16) arousal: 1,40 MSE and 0,30 PCC; valence: 0,20 MSE and 0,42 PCC
Thao et. al [75]	MediaEval 2016 and Cognimuse	visual (static appearance, spatio-temporal and low-level movement) and audio	Custom CNN	arousal and valence (-1 to 1)	(MediaEval16) arousal: 0,93 MSE and 0,35 PCC; valence: 0,76 MSE and 0,34 PCC (Cognimuse) arousal: 0,15 MSE and 0,52 PCC; valence: 0,20 MSE and 0,48 PCC
Muszynski et al. [49]	LIRIS-ACCEDE	special effects, lighting, music, main characters' emotional responses, dialogue	LSTMs	arousal and valence (both between -1 and 1)	MSE of 0,055 for arousal and 0,070 for valence
Sivaprasad et. al [66]	COGNIMUSE	visual (shot length, optical flow, color, video compressibility, facial expressions) and audio	LSTM	arousal and valence (both between -1 and 1)	arousal: 0,14 MSE and 0,70 PCC; valence: 0,25 MSE and 0,40 PCC
Yi and Wang [85]	MediaEval 2016	MKT, ConvNet and EmoBase10	SVM	arousal and valence (from -1 to 1)	arousal: 1,25 MSE; valence: 0,22 MSE
Yi and Wang [86]	MediaEval16	audio, color and optical flow	2 LSTMs	arousal and valence (from -1 to 1)	arousal: 0,73 MSE and 0,44 PCC; valence: 0,20 MSE and 0,42 PCC

Table 2.2: Comparison of previous video affective analysis works

Chapter 3

Prediction Of Emotional Responses To Videos Using Cinematic Features

In this chapter we describe all aspects related to our solution. We start with a brief overview of the whole solution, then we describe the datasets, methods and metrics used in the solution. Lastly, we present in detail each module of the solution.

3.1 Overview

As mentioned in Chapter 1, the main goal of our solution is to predict valence and arousal responses to videos using only their cinematic features. To achieve that, we first segment the input video into shots by detecting frames that mark shot boundaries with a trained model. Then, the shot's frames and key frames are used to extract the shot's cinematic features studied previously: shot length, key lighting, shot type and camera movement. Shot length corresponds to the number of frames. The key lighting is obtained by calculating the standard deviation of brightness in the key frames. Shot type (ELS, LS, MS, CS and ECS) and camera movement (static or not) are predicted using trained CNN models. Then, a vector with these features is passed to classification and regression methods

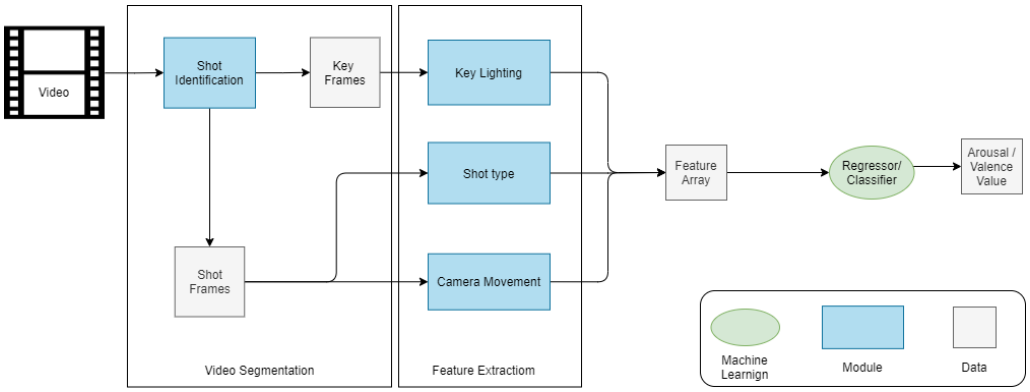


Figure 3.1: Diagram of our solution

to create models that output valence and arousal estimations. Figure 3.1 illustrates these steps.

3.2 Materials and Methods

In this section we present the material and methods used to create our solution and to perform the experiments, including datasets, machine learning algorithms and evaluation metrics.

3.2.1 Datasets

For training models to predict a video's shot type we found two datasets containing the necessary information. MovieShots [57] contains 33 thousand shots trimmed from over 10 thousand movie trailers and annotated by professionals with the shot scale (ELS, LS, MS, CS and ECS) and movement (static, pan, zoom in, zoom out and multiple movements). CineScale [60] is a large dataset of 792 thousand shot frames collected from 124 different whole movies, and then annotated with shot scale (ELS, LS, MS, CS and ECS).

As for Valence and Arousal (VA) prediction, we used six different benchmark datasets (see Table 3.1): DEAP, AFEW, Cognimuse, EMDB, EMovie and MediaEval 2016. DEAP has 40 one-minute long excerpts of music videos annotated with valence and arousal ratings from 1 to 9 by 32 participants. AFEW consists of 600 videos extracted from features films, for a total of more than 30.000 frames, each annotated with valence and arousal in the range of -10 to 10. Cognimuse contains 30 minute clips from 12 movies, with continuous annotations of intended arousal and valence, of range from -1 to 1. EMDB has 52 clips, 40 seconds each, edited from 100 commercial films, each annotated by 32 volunteers and then averaged. The EMovie dataset is composed of 39 excerpts, with approximately two minutes, characterized in terms of valence and arousal from 1 to 9, by 174 participants and averaged by gender. MediaEval 2016 [18] has 11000 short clips each annotated with valence and arousal, with range from 0 to 5.

Table 3.1: Description of VA datasets

Name	Videos	VA Range
AFEW	600 clips from features films	-10 to 10
Cognimuse	30 minute clips from 12 movies	-1 to +1
DEAP	40 one-minute clips from music videos	1 to 9
EMDB	52 clips from 100 films	1 to 9
EMovie	39 clips from movies and TV-shows	1 to 9
MediaEval16	11000 clips from movies	0 to 5

3.2.2 Deep Learning Methods

Recently there has been a lot of success using convolutional neural networks (CNNs) for action recognition in benchmark datasets. In our work we used some of these networks to train models for identifying cinematographic features. In this section, we provide some relevant concepts related to deep learning and CNNs.

A shot type can be identified from just a few frames that represent different points of the video, for example, Rao. et al [57] trained models to predict shot type using only key frames. Contrarily, camera movement requires using all frames to detect movement from successive frames.

Since our goal is to extract features for each shot, we used input data types that group frames by video during both kinds of tests. MMaction2 [15] is an open-source toolbox that offers different frameworks to work action recognition on video. It offers a few options. We opted by the "RawframeDataset", a subclass of the torch Dataset. This class loads raw frames, applies transformations to the images, like resizing them to 224x224, and returns a dictionary containing the frames in Tensors, which is a multi-dimensional matrix that represents different data types for the GPU. The RawframeDataset accepts a text file as input, where each line has the relative path of the folder with the shot frames, the total of frames and the label, split by a white space. For all datasets mentioned, we split the shots randomly in a 70 to 30 ratio for training and testing, and created a text file for each subset.

There are a few of strategies to handle video input. In uniform sampling, videos are divided into multiple segments of equal length, and then one frame is randomly selected from each one. In another strategy we choose frames of each sampled output clip, temporal interval of adjacent sampled frames and number of clips to be sampled. In Dense sampling, all the possible segments in a fixed-size sliding window is selected. We experimented with different sample rates with different models.

The convolution kernel of a CNN is a matrix of weights to extract relevant features from a part of input. In CNNs with a two dimensional kernel (2D CNNs), it moves in two directions and outputs a two dimensional matrix, while in 3D CNNs kernels move in three directions and output a three dimensional matrix.

The term backbone, in deep learning, denotes the feature extractor part of a CNN. It is a model, usually pre-trained, that outputs a feature map representation of the input. The most popular backbones are the VGGNet, Inception and ResNet. VGGNet [65] is a stack structure of small convolution and pooling layers, that ends in fully connected and softmax layers (see Figure 3.2). ResNet [27] has a similar structure, but with a skip connection from some layers to the output, thereby regularizing layers with bad performance (see Figure 3.3). Inception [73] consists of inception layers (see Figure 3.4), which are a combination of convolution layer with different sizes with their output concatenated into a vector forming the input of the next stage. Out of used backbones in action recognition,

independent of other settings, ResNet50 tends to be the strongest [13].

Figure 3.2: VGG19 architecture (source: [27])

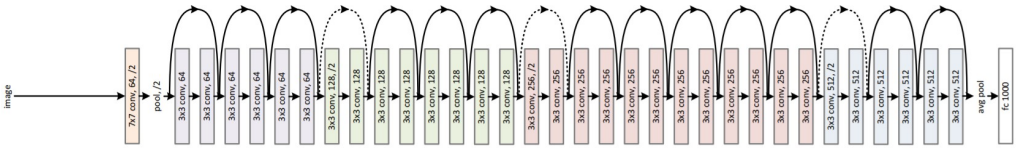


Figure 3.3: ResNet34 architecture (source: [27])

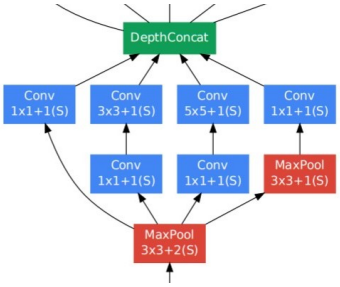


Figure 3.4: Inception Layer (source: [73])

Meanwhile, the term ”head” refers to the rest of the layers on top of the backbone. Some successful heads in action recognition are Temporal Segment Network (TSN), I3D and SlowFast. TSN [81] is a two-stream CNN framework able to model long-range temporal structures on videos, by combining a sparse input sampling of frames and video-level consensus. I3D [11] is a stack of inception layers, plus a few convolution and pool layers, with 3D kernels. SlowFast [26] uses two processes in parallel (see Figure 3.5): the Slow-Only path that trains a I3D head with low frame rate to capture spatial semantics and the Fast-Only path that analyses the frames at a high rate to easily detect swift changes in content. The Slow-Only path can be used as a head separately of the SlowFast path.

MMaction2 offers implementations for all of the CNNs mentioned above, and implementations for training and testing models that we used in several of our experiments.

3.2.3 Regression/Classification Methods

Besides CNNs, we also used some standard classification and regression methods to develop our solution. For regression we used Ridge Regressor, K-nearest Neighbours (KNN), Random Forest and Support Vector Machine for regression (SVR). As for classification we used Ridge Classifier, which scales the targets into a [-1, 1] range and applies

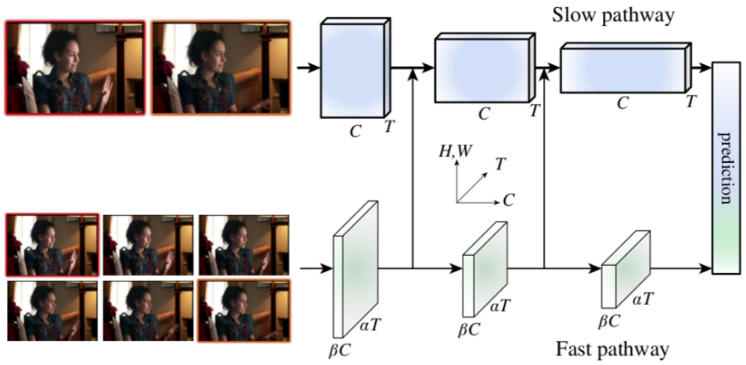


Figure 3.5: High-level view of SlowFast network (source: [26])

Ridge Regression on the problem, KNN, Random Forest and Support Vector Machine (SVM).

We tested the parameters of each algorithm using grid search and 10-fold cross-validation. Ridge classification and regression has the parameter alpha with range 1e-15 to 100.

In the case of KNN, we tested the number of neighbours with values 1, 3, 5, 7, 11, 21, 31 and 41, the algorithm used to compute the nearest neighbors: BallTree, KDTree or brute force. Weights either uniform or based on distance. Furthermore, tree algorithms can have euclidean, manhattan or minkowski metrics, with p 1 or 2 and leaf size from 1 to 80.

For Random Forest parameters, we tested the number of trees: 10, 50, 100, 500 or 1000. Max depth from none to 100. The minimum number of samples required to split an internal node: 2, 5 or 10. The minimum number of samples required to be at a leaf node 1, 2 or 4. And the square root of the number of features was used as the number of features to consider when looking for the best split.

Finally, SVR and SVM were used with the kernel 'rbf'. Regularization parameter C was tested with values between 1e-4 and 5, and epsilon between 1e-4 and 1e-1.

3.2.4 Evaluation metrics

In order to evaluate our experiment results, we used accuracy (Acc), precision, recall and the F1 score (F1) for classification tasks. An important step to calculating these metrics is the confusion matrix. In binary confusion matrices, when the predicted class is positive, if the actual class is positive then it's called a true positive (TP), otherwise it's a false positive (FP). When the predicted class is negative, and the actual class is negative, it's called a true negative (TN), otherwise it's a false negative (FN). Accuracy tells us how often a model is correct (see Equation 3.1), it can give a good idea of how well it is trained and how it performs. It is the most common metric used in existing classification works. However, accuracy is not reliable in cases of a big class imbalance, which is the

case in some of our datasets.

Precision are the TP out of the predicted positives (see Equation 3.2), and can be important when false positives have a high cost. Meanwhile, Recall are the TP out of TP and FN (see Equation 3.3), which helps when the costs of false negatives are high. F1 Score combines precision and recall (see Equation 3.4) for optimization. F1 scores closer to 1, have lower FP and FN and mean a better model. For problems with multiple classes, the metrics can be calculated using a few ways. We used the weighted method, which calculate metrics for each label and then their average by number of true instances for each label.

As for regression tasks, we used Mean Square Error (MSE) and Pearson's Correlation Coefficient (PCC). MSE is the average squared difference between the predicted values and actual numbers (see Equation 3.5), the closer to 0 the better the value. PCC measures the strength of a linear association between two variables, and is calculated by drawing a line that best fits the data of two variables, and then measuring the distance of all data points to this line (see Equation 3.6). The value is between -1 and 1, where -1 means there is a strong negative correlation, 0 means that there is no correlation and +1 means that there is a strong positive correlation. PCC values in the interval [0, 0.1] mean a negligible correlation, [0.1, 0.39] a weak correlation, [0.4, 0.69] a moderate correlation, [0.7, 0.89] a strong correlation and [0.9, 1] a very strong correlation [64].

$$ACC = \frac{TP + TN}{TP + FP + FN + TN} \tag{3.1}$$

$$Precision = \frac{TP}{TP + FP} \tag{3.2}$$

$$Recalll = \frac{TP}{TP + FN} \tag{3.3}$$

$$F1Score = \frac{2 * (Recall * Precision)}{Recall + Precision} \tag{3.4}$$

$$MSE = \frac{\sum_{i=1}^{N} (\hat{y}_i - y_i)^2}{N} \tag{3.5}$$

$$PCC = \frac{N \sum_{i=1}^{N} (\hat{y}_i y_i) - \sum_{i=1}^{N} \hat{y}_i \sum_{i=1}^{N} y_i}{\sqrt{N \sum_{i=1}^{N} \hat{y}_i^2 - (\sum_{i=1}^{N} \hat{y}_i)^2)} \sqrt{N \sum_{i=1}^{N} y_i^2 - (\sum_{i=1}^{N} y_i)^2)}} \tag{3.6}$$

3.3 Video Segmentation

One of the first decisions we had to make was the granularity at which our solution extracts features and outputs VA values. Shot length, shot type and camera motion are features that are only meaningful in the context of a shot. We also established that key lighting and VA ratings did not vary much during a shot. So even though other works use video

segments of one or five seconds, in the ambit of our goal it makes sense to work with shots.

In order to segment the videos into shots, we tried a couple of existing methods to detect shot boundaries. The first came from the MovieNet toolset [30], which is based on the PySceneDetect library. This toolset provides implementations of a couple of algorithms for shot detection. Unfortunately, this method did not always work and several resulting shots had errant frames.

Next, we tried the code in the repository TransNet V2 [70]. It uses 3D convolutional architectures trained with a large dataset to detect shots and output a list of boundary frames. We then used ffmpeg tools to extract all video frames and organize them into shots. This method managed better results, and so we used it with all datasets.

Aiming to extract keyframes from shots, we attempted to use the library Katna. It allows for the extraction of specific number of key frames, by dividing the video into a given number of segments and then select a key frame from each one. Since most shots have a small length, less than three seconds, and do not have a standard shot length, we can not predict a maximum length. So, we determined that three key frames should offer more than enough information about the whole shot. This process first requires combining the shot frames in a video using the moviepy library.

In sum, this first module of our solution, as depicted in Figure 3.1, receives a video as input, decomposes it into shots and outputs the shots' frames and key frames. We performed this process for all videos in the datasets used, except for the MovieShots dataset that already came divided into shots and the AFEW dataset that was divided in shot frames.

3.4 Feature Extraction

The feature extraction module includes sub-modules that receive segmented videos and determine the cinematic features key lighting, frame scale and camera movement, as shown in Figure 3.1. Shot length is easily obtained from the number of frames in the shot.

3.4.1 Key Lighting

As seen in Chapter 2, a low-key lighting, a lighting that creates a high contrast and dark shadows, increase arousal and decrease valence. To obtain a numerical value for the amount of contrast in a frame, we first convert the color image into a grayscale image. We do this because a grayscale image only captures the intensity of light in pixels and we wish to ignore color variations. Then we compute the standard deviation of brightness in all the pixels in the image. For this we used the Python Imaging Library. The final value of the shot is the median of the contrast in each key frame, and represents the key lighting.

3.4.2 Shot Type

There are two previous works on how to automatically classify shot scale. One uses video input and ResNet based CNNs [57], and the other uses frames as input and a VGG

network [60]. With this in mind, we trained our own models using datasets divided in shots

In our experiments we trained networks with ResNet50 and VGG-19 backbones, pre-trained with ImageNet, and with TSN and Slow-Only heads. For optimizers we tried SGD with commonly used parameters, learning rate 0.01 and momentum 0.9, and Adam with the default learning rate 0.001. The sample rates were chosen based on other reported works and computing limitations. We used 1x1x3 and 8x8x1, for TSN and Slow-Only respectively, where the order is frames of each sampled output clip, temporal interval of adjacent sampled frames and number of clips to be sampled. As for the number of epochs, we tested values from five to 150, and noticed that the accuracy results plateaued around 60 to 80, and decided to put the training maximum at 100 epochs.

The first model we trained used the MovieShots dataset labeled with five shot types (ELS, LS, MS, CS and ECS). Since we were unsure of how exactly shot type classes would impact valence and arousal prediction, we combined the labels ECS with CS and ELS with LS, and trained another model with the three classes (ELS + LS, MS, CS + ECS), using the best network from the previous test.

In the case of the CineScale dataset, labels were attributed on every frame so we reduced the labels to one per shot, by choosing the most common label in the shot's frames. Before creating our own models, we tested the model trained with the CineScale dataset, provided by the dataset authors, that only outputs three labels (CS, MS and LS), with the MovieShots dataset with three labels as ground truth. We also tested our model trained with the MovieShot dataset on a CineScale ground truth. Next, we created our own model using CineScale and five labels. Finally, we trained a model using both datasets combined, labelled with five classes.

With the aim of improving the results, we tried to isolate the subject from the background on frames. According to previous works [74] [57], isolating the subject in frames increases the accuracy of identifying shot type, since it highlights the distance of the camera to the subject.

The first step was to look for other authors' works on salient object detection. The most state of the art we found was U-2-Net [56], which uses a custom made deep learning network to capture great detail, and produced very good results. After running this algorithm on the key frames of all the shots of MovieShots, we got images with the objects in focus in white and the rest in black (salient frames). Then we obtained subject only images by using the method multiply, which performs a multiplication between the current frame and mask frame, so pixels in black (0) in the salient image turn pixels black in current frame as well, and the detected object is highlighted. For this we used the OpenCV library. After obtaining all frames with the subject isolated from the MovieShot dataset, we used them to train a model to predict shot type.

3.4.3 Camera Movement

When it comes to automatically predicting the camera movement in a video, the only previous work found uses video as input and ResNet based CNNs. So, we proceeded with experiments similar to the ones mentioned in the section above. We trained models with the MovieShots dataset, same backbones, same optimizers, but with TSN, Slow-Only, I3D and SlowFast heads. The sample rates used were 1x1x3 and 8x8x1, 4x16x1

and 32x2x1 for TSN, Slow-Only, SlowFast and I3D, respectively.

In relation to the labels used, the MovieShots dataset offers five labels (static, pan, zoom in, zoom out and multiple movements). However, training results were unsatisfactory. Since, there are no evidence that people react differently to different kinds of camera movement, we combined the camera motion labels into two labels: static and motion.

Similarly to when predicting the shot type, isolating the background from the subject on all frames of a video is said to improve accuracy in predicting camera movement [57], since frames with only the background can make movement in the background be more noticeable.

Once again, we used U-2-Net to create salient images for all frames in the MovieShot dataset. Then we obtained background only images by subtracting the salient frame from the current frame, so pixels in white in the salient image turn pixels black in current frame. Then, we used these frames to train a model to predict camera movement.

Since, identifying a camera movement requires tracking motion temporal information, we also trained a model with flow frames. The flow frames were obtained with a dense optical flow algorithm, implemented in the code of another author, called denseflow [82]. These frames show exclusively the vectors (with a maximum of one vector per pixel) of objects between consecutive frames of sequence, caused by the relative movement between the object and camera. This model and the model trained with background frames were then fused with a weighted average.

3.5 Regression and Classification

After extracted from the shots, all cinematic features are given to classification and regression methods in order to create five models, which outputs types are illustrated in Figure 3.6. Two models output estimations for arousal/valence values (Figure 3.6 a and b), two models output high/low classes for arousal/valence (Figure 3.6 d), and one model outputs the arousal/valence quadrant (Figure 3.6 c), where the arousal is high in the first and third quadrants (and vice versa), and valence is high in the first and forth quadrants (and vice versa).

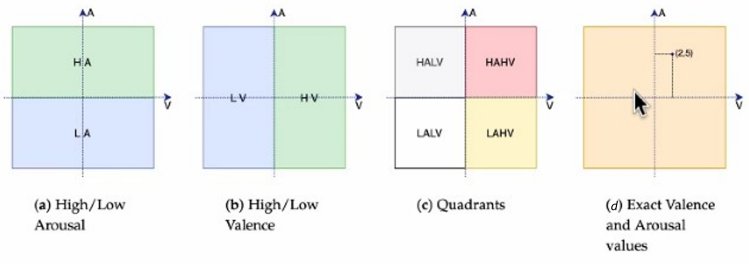


Figure 3.6: Depiction of the different VA models' outputs

The first models receive and output an estimation for an arousal/valence value. None of our benchmark datasets are labelled by shots, the original provided labels either apply to one video, in which case we use the same label for the shots contained in the video, or they apply to a timestamp. The latter is a problem, since using all labels within the shots with repeated features skews the model's results and makes it difficult to compare our

results between datasets and with other works. Therefore, we tried a couple of methods to merge ratings so that each shot has one label for valence and one for arousal. We tried using the average or median for all labels that apply to the shot, and using the value with the largest distance from the average, after removing outlier. Ultimately, we found using the median had the best results.

Still, different datasets required different approaches: DEAP and EMovie have one VA rating for the whole video, so the median of all participants ratings is used as the rating of all the shots in the video; for AFEW and Media16 we used the median of the ratings of all the frames in each shot. In Cognimuse, we used the median of all participant ratings for all the frames in the shot. EMDB has one VA rating for each video, so all shots in a video use the same rating.

The second model outputs high/low classification and receives each shot labelled as high, if the merged rating is ≥ 0, and low otherwise. The third model outputs quadrant classification, here the shots are labelled according to the four quadrants in the valence-arousal plane.

To choose the best algorithms for training each model, we experimented with the methods explained in Section 3.2.3, Ridge Regressor, Ridge Classifier, KNN, Random Forest and Support Vector Machine. All parameters were chosen using a random grid search. Then a leave one video out cross-validation was conducted, to best reproduce a real life situation, where the trained model would receive one new shot and output an emotional prediction.

3.6 Summary

In this chapter, we presented our solution to predicting emotional reactions to videos based on their cinematic features. We explained why it was important to divide a video into shots to extract its features, how using the TransNet V2 trained model was the most successful method in finding frames that limit shots and that we obtained keyframes by dividing a video into segments and selecting one frame from each. On the subject of datasets, we expounded on the contents of each dataset used and how they were homogenized.

Next, we described how the key lighting, shot type and camera movement features were extracted. For key lighting, we turned all key frames in a shot to greyscale and used the median contrast calculated in each one. The other features required a lot of tests to train deep learning networks, with different parameters and labels. As an attempt to improve results, we used salient object detection to find the object focused in the frame and used frames with only the subject to train a shot type model. Dense optical flow algorithms were used to highlight movement when compared to the previous frame.

Finally, we explained that datasets provide multiple labels that apply to just one shot and why this was a problem. We used the median as the best method to merge labels. Lastly, we mentioned the tests run to train high/low classification, quadrant classification and regression models for valence and arousal prediction.

Chapter 4

Experimental Evaluation

In this chapter, we present the results from the experimental procedure described in Chapter 3. First we present the results achieved with the models trained to predict the type and camera movement of a shot (cinematic features). Then, we present the results for the models trained with benchmark datasets to predict the emotional state that the videos could elicit in the viewers. We show results for high/low, quadrant and exact values of valence and arousal provoked by a shot.

4.1 Cinematic Feature Extraction

Throughout this section, we present the results from tests made with various trained CNNs for identifying shot types and camera movement.

One of our first tests, compared the optimizers SGD and Adam. The optimizer Adam with the default parameter learning rate 0.001 achieved the best results. For the TSN, Slow-Only, SlowFast and I3D heads we used different sample rates, 1x1x3 and 8x8x1, 4x16x1 and 32x2x1, respectively, where the order is frames of each sampled output clip, temporal interval of adjacent sampled frames and number of clips to be sampled.These models were all trained on a NVIDIA GeForce RTX 2060 GPU and with the datasets divided in a 70 to 30 train and test ratio.

When evaluating trained models, we present accuracy and F1-scores values for each class, calculated considering only the samples in the class, and overall, calculated with all samples and not the average of the values for each class.

4.1.1 Shot Type Estimation

In this section, we present the results of the models created to predict shot type. For the first set of tests, we compared different networks configurations to see which had the best results. We trained networks with ResNet50 and VGG-19 backbones, and TSN and Slow-Only heads, using the MovieShots dataset.

The results show that the TSN with a ResNet50 backbone is the best network, achieving an overall accuracy of 79,7%. In Table 4.1, we can see that TSN achieves a better accuracy than Slow-Only for all types of shots, except when classifying ELS. In the models that use VGG-19 as a backbone, the networks failed to learn the data and output the classes with most of the samples, despite there were no class imbalance. Since RestNet50

Table 4.1: Accuracy (%) for 5 classes of shot type, using the MovieShots dataset

Algorithm	ECS	CS	MS	LS	ELS	Overall
ResNet50 + TSN	86,7	68,4	75,8	80,4	90,7	79,7
ResNet50 + Slow-Only	73,6	62,3	56,2	48,7	94,7	65,8

did not have this problem, we think that, this might be due to VGG-19 simply not having enough depth to learn the MovieShots dataset.

In the second set of tests, we intended to see if the results of using a smaller number of shot type classes would affect the results, To that end, we combined the labels ECS with CS and ELS with LS, in the MovieShots dataset, defining only three classes. Then, we trained a ResNet50 + TSN model with these three labels and compared it with the model with the model with five labels (see Table 4.2). As we can see, the model with three labels achieved better results.

When comparing the results of using three and five classes as features on the models to estimate VA, we noted that arousal results slightly improved with 3 classes, while valence declined. More details are presented in Section 4.2.

Table 4.2: Comparison of the accuracy for 3 and 5 classes of shot type, using the MovieShot dataset.

Model	Metric	ECS	CS	MS	LS	ELS	Overall
5 classes	Accuracy	86,7	68,4	75,8	80,4	90,7	79,7
	F1-score	0,80	0,74	0,78	0,79	0,89	0,80
3 classes	Accuracy	-	80,7	80,9	89,9	-	84,2
	F1-score	-	0,86	0,72	0,90	-	0,85

Next, we ran tests using the CineScale dataset. In Table 4.3 we present the main results. First, we used a model trained with the CineScale dataset provided by the dataset's publisher, which is trained with three classes (CS, MS and LS), and tested it's accuracy when predicting the shot type of the MovieShot dataset, in particular, the same test subset used in the other experiment with three classes. When compared to the MovieShots model, the pre-trained CineScale managed very high results for the CS class, but low results for the other two classes, so it overall didn't have as good results as the other model.

Next, we tested the MovieShots trained model with five classes on CineScale as ground truth, which had results not much higher than chance accuracy (20%). Then, we tried to train our own model with the CineScale dataset with five classes, but the model always output the class with the most samples. In order to fix lack of samples, we trained a model with both datasets combined with five labels, however this only worsened the results from the MovieShot dataset model. This might be due to how the dataset originally was labelled in every frames instead of shot. Given these results, we abandoned the idea of using CineScale to create shot type models.

Lastly, we did a set of test using frames where their subject is isolated (Subject model) and compared it to the model trained with normal frames (RGB model). The results are shown in Table 4.4. While salient frames managed some good results, it did not improve the previous results.

Table 4.3: Accuracy (%) of models with dataset CineScale

Training data	Ground Truth	ECS	CS	MS	LS	ELS	Overall
Pre-trained CineScale	MovieShot (3 classes)	-	96,9	35,7	34,1	-	56,0
MovieShot	CineScale	22,0	21,0	37,0	25,0	10,0	29,0
CineScale	CineScale	0,0	0,0	100,0	0,0	0,0	45,3
MovieShot + CineScale	MovieShot + CineScale	68,7	41,0	75,7	38,5	82,7	59,6

Table 4.4: Comparison of models RGB and Subject models

Model	Metric	ECS	CS	MS	LS	ELS	Overall
RGB	Accuracy	86,7	68,4	75,8	80,4	90,7	79,7
	F1-score	0,80	0,74	0,78	0,79	0,89	0,80
Subject	Accuracy	57,3	77,2	54,9	59,8	89,4	67,0
	F1-score	0,65	0,72	0,67	0,61	0,69	0,67

4.1.2 Camera Movement Estimation

In this section we describe the results of predicting camera movement. We started by running tests to determine which networks configurations model camera movement best. We trained networks with ResNet50 and VGG-19 backbones, and I3D, TSN, Slow-Only and SlowFast heads, with the MovieShots dataset.

Just like in the test for shot type, experiences with VGG-19 failed for the same reasons discussed in the previous section.

When running tests with SlowFast (and ResNet50), we saw early on that it was computationally too expensive for the quality of results that it was returning, so we did not consider it a viable option. While I3D was less expensive, we were not able to find the correct parameters for the algorithm to learn our training data without exhausting resources. Thus, we only achieved results using TSN and Slow-Only, as we can see in Table 4.5. Both TSN and Slow-Only achieved great results predicting lack of movement, but Slow-Only is much better at predicting the existence of camera movement, than TSN.

Table 4.5: Accuracy (%) of movement models using the MovieShots dataset

Algorithm	Static	Moving	Overall
ResNet50 + Slow-Only	88,4	67,8	81,8
ResNet50 + TSN	89,0	24,0	67,8

In the next set of tests, we wanted to see if the best model (ResNet50 + Slow-Only) was able to identify several camera movement classes. We trained one ResNet50 + Slow-Only model with all available classes in the MovieShot dataset, which are pan, push, pull, multiple movements and static, and another with only three classes, by combining zooms and multiple movement labels. The results from all different models (in Table 4.6) show that the zoom and multiple movement classes were not learned by any model, possibly

due to class imbalance. So, we decided to only consider the detention of movement or static.

Table 4.6: Comparison of models for 2, 3 and 5 classes of camera movement using the MovieShot dataset

Model	Metric	Pan	Push	Pull	Multiple	Static	Overall
5 classes	Accuracy	66,8	0	0	0	89,9	78,9
	F1-score	0,68	0,00	0,00	0,00	0,86	0,77
3 classes	Accuracy	75,9		0,75		75,5	72,6
	F1-score	0,67		0,07		0,81	0,73
2 classes	Accuracy			67,8		88,4	81,8
	F1-score			0,71		0,87	0,81

For the last set of tests, we trained one ResNet50 + Slow-Only model with dense-flow frames and one with frames where the background is isolated from the foreground (salient). The denseflow model had worse results than the RGB frames overall, and even though it got a better accuracy for the Static class, the F1-score shows a lot of false positives and negatives. On the other hand, the background model got overall better results than the model with normal frames (RGB).

Then, we studied the fusion of the background and flow approaches. To that end we chose the distribution of weights to a total of one, we attempted all combinations in intervals of 0,05 from 0 to 1, and concluded that the best results were achieved when giving a weight of 0,1 to the flow model and 0,9 to the background model. This fusion improved the results, as we can see in Table 4.7

Table 4.7: Comparison of the original model (RGB) with models using flow, background and a combination of both, using the MovieShots dataset

Model	Metric	Movement	Static	Overall
RGB	Accuracy	67,8	88,4	81,8
	F1-score	0,71	0,87	0,81
Flow	Accuracy	54,8	92,6	80,3
	F1-score	0,64	0,86	0,79
Background	Accuracy	76,3	91,0	86,9
	F1-score	0,77	0,91	0,87
Background + Flow	Accuracy	74,5	95,3	88,5
	F1-score	0,81	0,92	0,88

4.1.3 Discussion

From our tests we could decide on what models to use to extract cinematic features the models for emotion estimation.

In regard to the shot type, we experimented with a couple of network configurations and found that the ResNet50 was the only backbone that could extract features from the

training data and that TSN was the head with the best performance, with an overall accuracy of 80% and F1-scores between 0,74 and 0,89 for each class.

We also tested a model with three classes by combining similar labels, ECS with CS and ELS with LS, and saw an improvement in overall accuracy to 84%. However, when testing the three class in addition to the five original classes as features in predicting VA, we only saw a slight overall improvement on arousal, so they seem to contain around the same information. Nevertheless, this improvement was consistent in all benchmark datasets, so we decided to use this combination.

The CineScale dataset was very recently released to the public and the only reported results use three classes of shot types, takes as input the frames, image input and uses a VGG network to extract features. In our experiments, we didn't manage to achieve as good results as we achieved with the MovieShots dataset. Perhaps this dataset could be used in future experiments.

In view of these observations, the models used in the rest of the tests to identify shot type have a network consisting of a TSN head and a ResNet50 backbone, trained with RGB frames from the MovieShots dataset. In valence experiments only the model that outputs five classes is used, while in arousal experiments, we use this model in combination with a model that outputs three classes.

As for camera movement, we tried a few network configurations and found that the ResNet50 was the only backbone that could learn the training data, while the Slow-Only head had the best results, with an overall accuracy of 82%, with a lower score for the Moving class than for the Static one.

Comparing models trained with different numbers of classes, made it clear that the model didn't learn any of the different classes within movement (pan, push, pull and multiple) and since we didn't find any evidence that different camera movements provoked different emotions it made sense to combine them in a single class (movement).

Finally, a model trained with frames with an isolated background salient and a model trained with denseflow frames were fused. Just as expected both these methods helped the algorithm focus in the camera movement and improved the overall accuracy from 81,8% to 88,5%.

Given all of this, the rest of the tests extract camera movement with two fused models, both trained with a Resnet50 + Slow-Only network, one trained with RGB frames with the background isolated (background) and one trained with denseflow frames (flow) using MovieShots dataset.

4.2 Prediction of Emotional Responses

In this section, we present the results of our method to predict the emotional response elicited by videos. We report values for VA classification (High/Low Arousal/Valence and quadrants) and VA estimation (exact values for valence and arousal). We compare our results with the best published works across four benchmark datasets, namely DEAP, AFEW, Cognimuse and Media16. To that end, we trained different regression and classification algorithms using the cinematic features. Shot length and key lighting are extracted from metadata and key frames, respectively, using traditional methods and normalized between 0 and 1. Shot type and camera movement are extracted using the trained models identified in the previous section and then the probability of each class is used as a feature.

The results are separated by classification and estimations of arousal/valence, and then by dataset. In all tables, '-' indicates that the values were not reported by the author.

4.2.1 VA classification

In this section, we present the classification results for high/low arousal/valence and for quadrant models. All results where obtained by running the Ridge Regressor, K-nearest Neighbours (KNN), Random Forest (RF) and Support Vector Machine (SVM) with grid search to obtain the best results, and using the leave one out cross validation to simulate a real live situation, of receiving one shot as input and producing a result.

Before running all classification tests, all the datasets have labels skewed to one class, especially in the case of the quadrant models. In order to solve the class imbalance problem, we tried oversampling methods (Random, Synthetic Minority Oversampling Technique (SMOTE), Borderline-SMOTE, Adaptive Synthetic Sampling (ADASYN)), undersampling methods (Random and Near Miss) and a combination of both (SMOTETomek). We found SMOTE to be the best fit for high/low models, while on quadrant models oversampling caused overfitting, and Near Miss achieved the best results with the shortest learning time.

Table 4.8 shows our results for the high/low arousal/valence classification for all datasets. On arousal, the datasets DEAP, EMOVIE and EMDB have over 80% accuracy on arousal, while Media16Eval has the lowest accuracy and F1-score, 60% and 0,54, respectively. As for valence, Media16, EMOVIE and EMDB have the accuracies over 70%, although the Media16 F1-score being 60% indicates some false positives. The lowest accuracy is 62% and the lowest F1-score is 0,55, both on the Cognimuse dataset. Overall, we achieved good results.

Table 4.8: Our classification results for high/low arousal/valence

Dataset	Alg	Arousal		Valence	
		ACC	F1	ACC	F1
DEAP	SVM	**86,30**	0,80	67,12	0,67
AFEW	Ridge	64,17	0,62	66,67	0,56
Cognimuse	Ridge	77,20	0,69	62,15	0,55
Media16	SVM	60,02	0,54	70,98	0,59
EMOVIE	SVM	83,74	**0,83**	73,89	0,74
EMDB	SVM	80,90	0,81	**78,65**	**0,79**

Meanwhile, Table 4.9 shows the results for quadrant models for all datasets. All accuracies are between 79% and 39% and the F1-scores are between 0,79 and 0,38. Figure 4.1 contains the confusion matrices for all quadrant models. We can see that despite classes being balanced and Random Forest algorithms being given balanced class weights, the models still show bias towards most common classes. Datasets EMOVIE and EMDB achieve the best results across classes, confirming the F1-scores.

Table 4.10 compares the classification results with DEAP and MediaEval16 dataset using our method and the Ou et. al [52] method, which extracts multiple visual features using a CNN and uses LSTMs to predict.

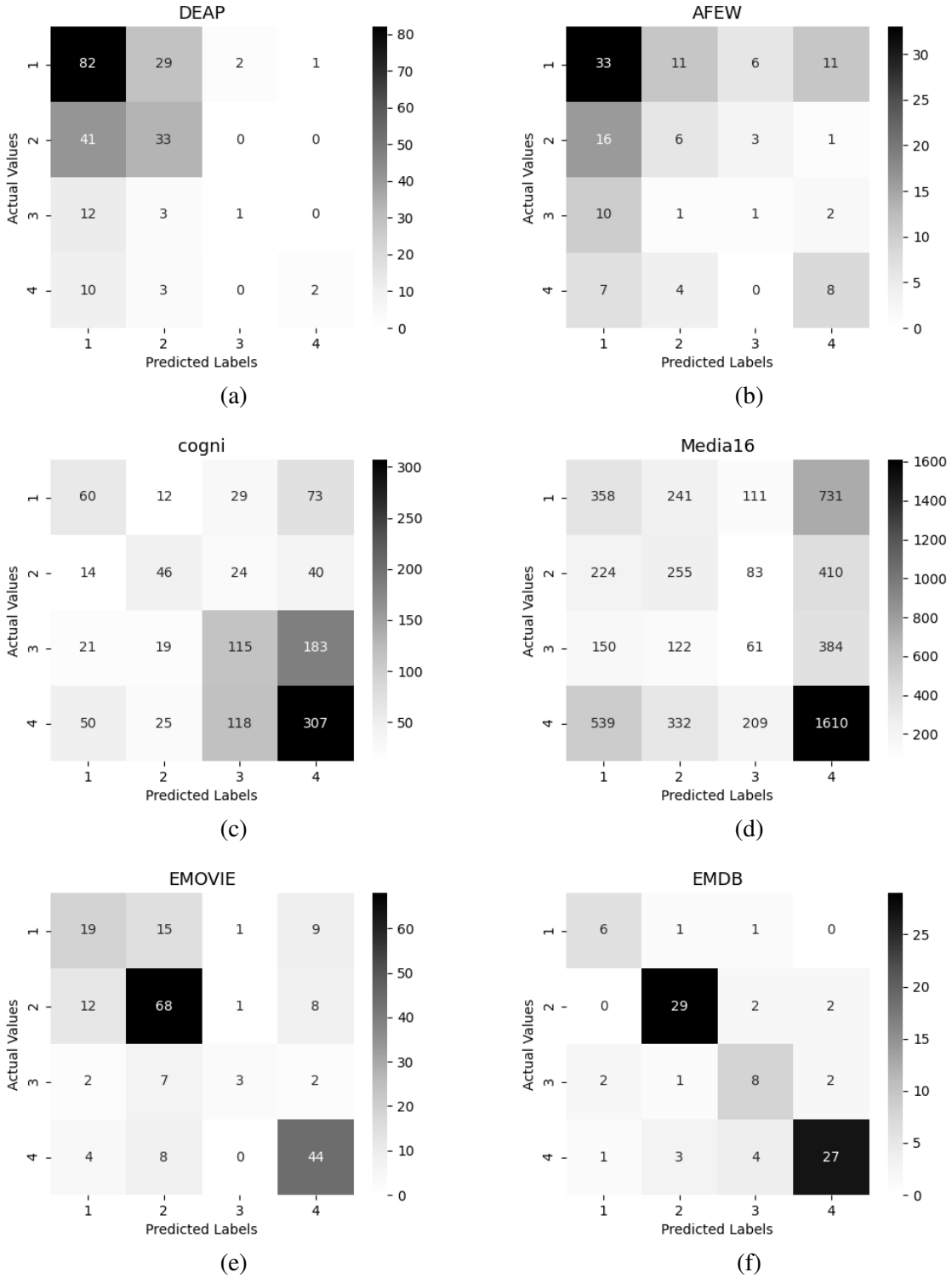


Figure 4.1: Confusion matrices for quadrants. a) DEAP b) AFEW c) Cognimuse d) Media16 e) EMOVIE f) EMDB

With the DEAP dataset, our method achieved the best results on arousal and even surpassed the results with the Ou et. al method from 60% to 86%. While, on valence the

Table 4.9: Our classification results for quadrants

Dataset	Alg	ACC	F1
DEAP	Random Forest	53,88	0,51
AFEW	KNN	40,00	0,39
Cognimuse	Random Forest	46,30	0,46
Media16	Random Forest	39,24	0,38
EMOVIE	KNN	66,00	0,65
EMDB	Random Forest	**78,65**	**0,79**

Ou et. al method attained results higher than our method by 4%.

In the case of the MediaEval16 dataset results, our method improved on the Ou et. al method significantly, especially for valence accuracy, from 45% to 71%.

Basically, we achieved similar or higher classification results in both datasets, with our handcrafted features, against features extracted and selected without allowing for interpretability.

Table 4.10: High/Low Arousal/Valence classification comparison using the DEAP and MediaEval16 dataset

Dataset	Work	Alg	Arousal		Valence	
			ACC	F1	ACC	F1
DEAP	Ou et. al [52]	LSTMs	60,29	0,60	**70,59**	**0,71**
	Ours	SVM	**86,30**	**0,80**	67,12	0,67
MediaEval16	Ou et. al [52]	LSTMs	56,25	0,29	44,85	0,41
	Ours	SVM	**60,02**	**0,54**	**70,98**	**0,59**

4.2.2 VA Estimation

In this section, we present the results for our valence and arousal regression models. All models where obtained by testing the Ridge Classifier, K-nearest Neighbours, Random Forest and Support Vector Machine for regression (SVR) algorithms with grid search and leave one out cross validation.

In Table 4.11 we compare our experiments for the DEAP dataset with Ou et. al [52], which uses a CNN to extract visual features and several LSTMs to predict exact valence and arousal values. We can see that our method achieved better results on valence (MSE) and arousal (PCC). Our MSE for arousal is better than the valence MSE, while theirs are around the same values.

In the case of the AFEW dataset our method struggled more. There were no tests for this dataset that used visual features. Therefore, we can't compare our results to other works. Nevertheless, in Table 4.12 we include results from the Kossaifi et. al [39] method, which collected EEG readings and facial features and used them on a SVR algorithm, only as a point of reference. Our valence model achieved a high negative PCC value, meaning our features had a strong negative correlation to the target. However, both valence and

Table 4.11: Comparison of regression results using the DEAP dataset

Work	Alg	VA Range	Arousal		Valence	
			MSE	PCC	MSE	PCC
Ou et. al [52]	LSTMs	[1, 9]	1,22	**0,37**	**1,28**	0,30
Ours	Random Forest	[1, 9]	**0,66**	0,30	2,43	**0,37**

arousal MSE values over seven in an interval of -10 to 10, too high to consider it a good model.

Table 4.12: Comparison of regression results using the AFEW dataset

Work	Alg	VA Range	Arousal		Valence	
			MSE	PCC	MSE	PCC
Kossaifi et. al [39]	SVR	$[-10, 10]$	**4,97**	**0,45**	**6,97**	**0,40**
Ours	Ridge	$[-10, 10]$	7,00	0,21	7,65	-0,70

When comparing our experiments using the COGNIMUSE dataset, we used two other works. The Sivaprasad et. al [66] method which trains LSTMs with a few visual features extracted with traditional methods and the Thao et. al [75] method which trains a custom CNN with visual features extracted with a ResNet50 and I3D network. The results for the comparison are shown in Table 4.13. We see that we can better predict exact valence and arousal than previous works, since our MSE is the lowest. However, our PCC has the lowest values.

Table 4.13: Comparison of regression results using the Cognimuse dataset

Work	Alg	VA Range	Arousal		Valence	
			MSE	PCC	MSE	PCC
Sivaprasad et. al [66]	LSTM	$[-1, +1]$	0,14	**0,70**	0,25	0,40
Thao et. al [75]	Custom CNN	$[-1, +1]$	0,15	0,52	0,20	**0,48**
Ours	Random Forest	$[-1, +1]$	**0,08**	0,44	**0,08**	0,33

The comparison between our results and reported results for the MediaEval16 dataset are shown in the Table 4.14. Both Yi and Wang methods use CNNs to extract visual features, while one [85] uses SVM to predict VA, the other [86] uses a custom CNN based on LSTMs. We can see that our experiments held up well against other reported results, but our PCC results are significantly lower than other reported works. All results were surpassed by Yi and Wang's work [86]. In here, we can again see that our valence model had better results than the arousal one.

4.2.3 Discussion

In this section, we discuss the results presented in this section for all three types of models trained to predict valence and arousal.

Table 4.14: Comparison of regression results using the MediaEval16 dataset

Work	Alg	VA Range	Arousal		Valence	
			MSE	PCC	MSE	PCC
Yi and Wang [85]	SVM	[0, 5]	1,25	-	0,22	-
Yi and Wang [86]	LSTMs	[0, 5]	**0,73**	**0,44**	**0,20**	**0,42**
Thao et. al [75]	Custom CNN	[0, 5]	0,93	0,35	0,76	0,34
Ou et. al [52]	LSTMs	[0, 5]	1,40	0,30	**0,20**	**0,42**
Ours	Random Forest	[0, 5]	0,96	0,22	0,40	0,16

Our test with high/low arousal/valence models showed that all reached accuracies between 59% and 86%. DEAP had the best results and when compared to other works, one using a more specialized feature, EEG, and another using a bigger variety of visual features, our method still had a better accuracy and F1-score for arousal. Our results for Media16, despite having our worst high/low score, still improved upon a previous work.

The quadrant models got F1-scores between 0,79 and 0,38, all higher than change. It should also be taken into account that some datasets have heavy class imbalances, and even after using SMOTE to balance the training data, this seemed to have affected the results.

With the regression models our method got results that are similar to the best VA prediction works and sometimes even better, using only cinematic features. On the DEAP dataset, our method performed better on the arousal MSE, with a value almost half lower, than a method that uses several visual features, but the valence model didn't fit the values.

Our method had a worse performance with the AFEW dataset, even getting MSEs equal and greater than seven in a range from -10 to 10, despite, the PCC for valence being high. This might be due to the nature of the clips in the dataset, since they mostly include shots with people, often in medium or close-up shots, so the method using facial expression had a better performance. Plus, the best method that used AFEW, with EEG readings and facial features as features, didn't report good results either.

As for the Cognimuse dataset, our MSEs are lower than 0,1 in an interval of -1 to 1, better than other methods which use a variety visual features, and deep learning techniques.

In almost all our results, the PCC values are in the intervals [0.1, 0.39] and [0.4, 0.69], which signifies a low and moderate correlation to the target variable, respectively. These observations are consistent with other work's reported PCC values. However, taking into consideration the relation between PCC and MSE, we noticed the tendency for our PCC values to be lower when compared to other works, while MSE values are better than works with similar PCCs. This suggests that our features have a non-linear correlation to valence and arousal.

As expected we found mostly better results predicting arousal than valence. This is consistent with other reports, and can be because positive or negative feelings are more related to plot and characters, while excitement can be more easy to predict with visual cues [66].

Overall, our method had results commensurable with previous works, especially when measuring error. Given that we only used cinematic features, this shows that they could

affect viewers valence and arousal and can be a valuable tool in predicting exact values.

4.3 Summary

In this chapter, we started by describing the results of extracting features using CNNs trained with the MovieShots dataset. For shot type estimation, we identified that the best network configurations was a ResNet50 backbone and a TSN head. It achieved an overall accuracy of 80% trained with normal RGB frames of the MovieShot dataset. We also saw an improvement in arousal prediction tests, when using three extra features from combining labels ECS with CS and ELS with LS.

As for camera movement, the network configuration that got best results was a ResNet50 backbone with a Slow-Only head. Two of these models were trained, one with denseflow frames and the other with frames with the background isolated, and two labels, static and movements (a combination of pan, push, pull and multiple camera movements). The fusion of these models achieved an overall accuracy of 89%.

Next, we presented the results of training three valence and arousal models (one for high/low arousal/valence classification, one for quadrant classification and one for valence and arousal regression) with four benchmark datasets (DEAP, AFEW, Cognimuse and Media16). For all models, we tested several classification and regression algorithms with grid search and leave one out cross validation, and reported the best results.

On the high/low models we achieved accuracies between 60% and 86% for arousal and between 62% and 79% for valence. While the quadrant models got F1-scores between 0,79 and 0,38. For the DEAP and MediaEval16 datasets, we achieved similar or higher classification results than the previous works that reported values for the datasets.

Finally, we discussed results for our regression models. Overall, our method using cinematic features achieved results close and sometimes better to other works that extract a bigger variety of features using deep learning. Given this, we concluded that our method can be relevant in predicting valence and arousal provoked by videos.

Chapter 5

Conclusion

In this chapter, we present the final conclusion and the contributions of our work. We also name the limitations of our solution and suggest ways to expand and improve it in future works.

5.1 Summary of Dissertation

In this work we presented our solution to predicting viewers' emotional response to videos using cinematic features. We explored methods to extract cinematic features from videos, created models to predict valence and arousal, and compared results to other reported methods.

In chapter 2, we described the main cinematic features, studies on how they emotionally impact its viewers and existing methods to detect or extract these features. We then discussed approaches to predict emotion with different visual features. In the end, we decided to use the features shot length, key lighting, shot type and camera movement in our work.

In chapter 3, we presented our solution, which receives one video as input, extracts its cinematic features and uses them to predict emotional response to the video. First, we detail how we decomposed videos into shots and key frames. Then, we present the traditional methods we used to calculate shot length and key lighting, and then we describe the CNN models we trained and tested to detect shot type and camera movement. Lastly, we talked about the high/low arousal/valence, quadrant and estimation of arousal/valence models we created with the extracted features.

In chapter 4, we conducted our experimental evaluation. First we compared the models trained to predict shot type and achieved an overall accuracy of 80% with five classes and 86% with three classes. In models that predict arousal, we decided to use the features obtained with both these model, while in models with valence, we only used the model with five classes. For camera movement, we achieved an accuracy of 89% with the fusion of a model trained with flow frames and the other with frames with the background isolated. Finally, we compared results for the VA prediction models with other author's works for various datasets. Overall, we achieved results close and sometimes better than other works that used deep learning methods to extract various visual features.

5.2 Contributions

In the end of this dissertation, we successfully developed a solution that uses cinematic features to predict emotional responses to videos. We trained models that classify shots with shot type and camera movement, present the results of extensive tests with different networks, parameters and input and describe the combination that produced the best models. Lastly, we created models that use cinematic features as input and output emotional response in three forms: high/low arousal/valence classification, quadrant classification and arousal/valence regression. These models achieve results comparable to methods that use a variety of visual features extracted with deep learning, while also being able to provide information about how the cinematic features of the video impact the emotional response to videos.

5.3 Limitations

Despite the success of our solution, we did face some limitations while training our models to detect camera movement, not only due to some computation power limitations, but also in the information available to use. Since the only dataset labelled with camera movement had a large class imbalance between static and movement labels, and consequently, we could only use two classes.

5.4 Future Work

For future works, we propose a study on how cinematic features affects emotional responses, meaning what features elicit what emotional state, if there are any combinations of features with a particular emotional response and with what intensity features impact emotional responses.

Further works could study additional cinematic features that impact emotional response to video. One option is extracting color from key frames, some research would have to be done to find which color space to use and how to calculate its values. Another option is extracting sound features. Some sounds purposefully used by film makers to induce emotion, for example music cues, and it is already proven that sound features improve valence and arousal prediction. To predict emotion, new models could be trained individually for each feature and then merged with a late fusion or one new model could be trained with all the features combined.

Bibliography

[1] Magda B Arnold. The emotions: Facts, theories and a new model, 1964.

[2] Yoann Baveye, Emmanuel Dellandréa, Christel Chamaret, and Liming Chen. Deep learning vs. kernel methods: Performance for emotion prediction in videos. In *2015 international conference on affective computing and intelligent interaction (acii)*, pages 77–83. IEEE, 2015.

[3] Yoann Baveye, Emmanuel Dellandrea, Christel Chamaret, and Liming Chen. Liris-accede: A video database for affective content analysis. *IEEE Transactions on Affective Computing*, 6(1):43–55, 2015.

[4] S. Benini, M. Savardi, K. Bálint, A. B. Kovács, and A. Signoroni. On the influence of shot scale on film mood and narrative engagement in film viewers. *IEEE Transactions on Affective Computing*, pages 1–1, 2019.

[5] Margaret M Bradley and Peter J Lang. Measuring emotion: the self-assessment manikin and the semantic differential. *Journal of behavior therapy and experimental psychiatry*, 25(1):49–59, 1994.

[6] Kaitlin L Brunick, James E Cutting, and Jordan E DeLong. Low-level features of film: What they are and why we would be lost without them. 2013.

[7] Erik Cambria, Andrew Livingstone, and Amir Hussain. The hourglass of emotions. In *Cognitive behavioural systems*, pages 144–157. Springer, 2012.

[8] Nick Campbell and Parham Mokhtari. Voice quality: the 4th prosodic dimension. In *15th ICPhS*, pages 2417–2420, 2003.

[9] Luca Canini, Sergio Benini, and Riccardo Leonardi. Affective analysis on patterns of shot types in movies. In *2011 7th International Symposium on Image and Signal Processing and Analysis (ISPA)*, pages 253–258. IEEE, 2011.

[10] Guolu Cao, Yuliang Ma, Xiaofei Meng, Yunyuan Gao, and Ming Meng. Emotion recognition based on cnn. In *2019 Chinese Control Conference (CCC)*, pages 8627–8630. IEEE, 2019.

[11] Joao Carreira and Andrew Zisserman. Quo vadis, action recognition? a new model and the kinetics dataset. In *proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 6299–6308, 2017.

[12] Sandra Carvalho, Jorge Leite, Santiago Galdo-Álvarez, and Oscar F Gonçalves. The emotional movie database (emdb): A self-report and psychophysiological study. *Applied psychophysiology and biofeedback*, 37(4):279–294, 2012.

[13] Chun-Fu Chen, Rameswar Panda, Kandan Ramakrishnan, Rogerio Feris, John Cohn, Aude Oliva, and Quanfu Fan. Deep analysis of cnn-based spatio-temporal representations for action recognition, 2021.

[14] Shizhe Chen and Qin Jin. Ruc at mediaeval 2016 emotional impact of movies task: Fusion of multimodal features. In *MediaEval*, 2016.

[15] MMAction2 Contributors. Openmmlab's next generation video understanding toolbox and benchmark. https://github.com/open-mmlab/mmaction2, 2020.

[16] James E Cutting, Jordan E DeLong, and Christine E Nothelfer. Attention and the evolution of hollywood film. *Psychological science*, 21(3):432–439, 2010.

[17] Apostolos Dailianas, Robert B Allen, and Paul England. Comparison of automatic video segmentation algorithms. In *Integration Issues in Large Commercial Media Delivery Systems*, volume 2615, pages 2–16. International Society for Optics and Photonics, 1996.

[18] Emmanuel Dellandréa, Liming Chen, Yoann Baveye, Mats Viktor Sjöberg, Christel Chamaret, et al. The mediaeval 2016 emotional impact of movies task. In *CEUR Workshop Proceedings*, 2016.

[19] Emmanuel Dellandréa, Liming Chen, Yoann Baveye, Mats Viktor Sjöberg, Martijn Huigsloot, et al. The mediaeval 2017 emotional impact of movies task. In *MediaEval 2017 Workshop*, 2017.

[20] Emmanuel Dellandréa, Liming Chen, Yoann Baveye, Mats Viktor Sjöberg, Martijn Huigsloot, and Zhongzhe Xiao. The mediaeval 2018 emotional impact of movies task. In *MediaEval 2018 Workshop*, 2018.

[21] Benjamin H Detenber, Robert F Simons, and Gary G Bennett Jr. Roll 'em!: The effects of picture motion on emotional responses. *Journal of Broadcasting & Electronic Media*, 42(1):113–127, 1998.

[22] Laurence Devillers, Laurence Vidrascu, and Lori Lamel. Challenges in real-life emotion annotation and machine learning based detection. *Neural Networks*, 18(4):407–422, 2005.

[23] Ed Diener and Asghar Iran-Nejad. The relationship in experience between various types of affect. *Journal of Personality and Social Psychology*, 50(5):1031, 1986.

[24] Jana Eggink and Denise Bland. A large scale experiment for mood-based classification of tv programmes. In *2012 IEEE International Conference on Multimedia and Expo*, pages 140–145. IEEE, 2012.

[25] P Ekman. Nebraska symposium on motivation, 1971, 1972.

[26] Christoph Feichtenhofer, Haoqi Fan, Jitendra Malik, and Kaiming He. Slowfast networks for video recognition. In *Proceedings of the IEEE international conference on computer vision*, pages 6202–6211, 2019.

[27] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition, 2015.

[28] Julian Hochberg and Virginia Brooks. Film cutting and visual momentum. *Mary A. Peterson, Barbara Gillam, and HA Sedgwick*, 2007.

[29] Qingqiu Huang, Yu Xiong, Anyi Rao, Jiaze Wang, and Dahua Lin. Movienet: A holistic dataset for movie understanding. In *The European Conference on Computer Vision (ECCV)*, 2020.

[30] Qingqiu Huang, Yu Xiong, Anyi Rao, Jiaze Wang, and Dahua Lin. Movienet: A holistic dataset for movie understanding. In *Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

[31] Yunting Huang. Investigating how film lighting techniques influence viewers' emotional arousal, emotional valence and state empathy. 2018.

[32] Alice M Isen, Paula M Niedenthal, and Nancy Cantor. An influence of positive affect on social categorization. *Motivation and Emotion*, 16(1):65–78, 1992.

[33] Yu-Gang Jiang, Baohan Xu, and Xiangyang Xue. Predicting emotions in user-generated videos. In *Twenty-Eighth AAAI Conference on Artificial Intelligence*, 2014.

[34] Hang-Bong Kang. Affective content detection using hmms. In *Proceedings of the eleventh ACM international conference on Multimedia*, pages 259–262, 2003.

[35] John R Kender, John R Smith, Jiebo Luo, Susanne Boll, and Winston Hsu. *ICMR'16: proceedings of the 2016 ACM on International Conference on Multimedia Retrieval: June 6-9, 2016, New York, NY, USA*. ACM, 2016.

[36] Hyung-Suk Kim. Linear predictive coding is all-pole resonance modeling.

[37] Igor Knez. Effects of indoor lighting on mood and cognition. *Journal of environmental psychology*, 15(1):39–51, 1995.

[38] Sander Koelstra, Christian Muhl, Mohammad Soleymani, Jong-Seok Lee, Ashkan Yazdani, Touradj Ebrahimi, Thierry Pun, Anton Nijholt, and Ioannis Patras. Deap: A database for emotion analysis; using physiological signals. *IEEE transactions on affective computing*, 3(1):18–31, 2011.

[39] Jean Kossaifi, Georgios Tzimiropoulos, Sinisa Todorovic, and Maja Pantic. Afew-va database for valence and arousal estimation in-the-wild. *Image and Vision Computing*, 65:23–36, 2017.

[40] Sarah Kozloff. *Overhearing film dialogue*. Univ of California Press, 2000.

[41] Annie Lang, Shuhua Zhou, Nancy Schwartz, Paul D Bolls, and Robert F Potter. The effects of edits on arousal, attention, and memory for television messages: When an edit is an edit can an edit be too much? *Journal of Broadcasting & Electronic Media*, 44(1):94–109, 2000.

[42] Peter J Lang. The emotion probe: studies of motivation and attention. *American psychologist*, 50(5):372, 1995.

[43] Ye Ma, Zipeng Ye, and Mingxing Xu. Thu-hcsi at mediaeval 2016: Emotional impact of movies task. In *MediaEval*, 2016.

[44] Antonio Maffei and Alessandro Angrilli. E-movie-experimental movies for induction of emotions in neuroscience: An innovative film database with normative data and sex differences. *Plos one*, 14(10):e0223124, 2019.

[45] W MCD. Grundzüge der physiologischen psychologie principles of physiological psychology. *Nature*, 71(1849):529–530, 1905.

[46] Albert Mehrabian. Pleasure-arousal-dominance: A general framework for describing and measuring individual differences in temperament. *Current Psychology*, 14(4):261–292, 1996.

[47] Albert Mehrabian and James A Russell. *An approach to environmental psychology.* the MIT Press, 1974.

[48] William C Miller. Film movement and affective response and the effect on learning and attitude formation. *AV communication review*, 17(2):172–181, 1969.

[49] Michal Muszynski, Leimin Tian, Catherine Lai, Johanna Moore, Theodoros Kostoulas, Patrizia Lombardo, Thierry Pun, and Guillaume Chanel. Recognizing induced emotions of movie audiences from multimodal information. *IEEE Transactions on Affective Computing*, 2019.

[50] Paula M Niedenthal and François Ric. *Psychology of emotion*. Psychology Press, 2017.

[51] Charles Egerton Osgood, George J Suci, and Percy H Tannenbaum. *The measurement of meaning*. Number 47. University of Illinois press, 1957.

[52] Yangjun Ou, Zhenzhong Chen, and Feng Wu. Multimodal local-global attention network for affective video content analysis. *IEEE Transactions on Circuits and Systems for Video Technology*, 31(5):1901–1914, 2020.

[53] Robert Plutchik. The nature of emotions: Human emotions have deep evolutionary roots, a fact that may explain their complexity and provide tools for clinical practice. *American scientist*, 89(4):344–350, 2001.

[54] Jennifer Lee Poland. Lights, camera, emotion!: An examination on film lighting and its impact on audiences' emotional response. 2015.

[55] Soujanya Poria, Erik Cambria, Rajiv Bajpai, and Amir Hussain. A review of affective computing: From unimodal analysis to multimodal fusion. *Information Fusion*, 37:98–125, 2017.

[56] Xuebin Qin, Zichen Zhang, Chenyang Huang, Masood Dehghan, Osmar Zaiane, and Martin Jagersand. U2-net: Going deeper with nested u-structure for salient object detection. volume 106, page 107404, 2020.

[57] Anyi Rao, Jiaze Wang, Linning Xu, Xuekun Jiang, Qingqiu Huang, Bolei Zhou, and Dahua Lin. A unified framework for shot type classification based on subject centric lens. *arXiv preprint arXiv:2008.03548*, 2020.

[58] James A Russell. A circumplex model of affect. *Journal of personality and social psychology*, 39(6):1161, 1980.

[59] James A Russell and Albert Mehrabian. Evidence for a three-factor theory of emotions. *Journal of research in Personality*, 11(3):273–294, 1977.

[60] M. Savardi, A. Signoroni, P. Migliorati, and S. Benini. Shot scale analysis in movies by convolutional neural networks. In *2018 25th IEEE International Conference on Image Processing (ICIP)*, pages 2620–2624, Oct 2018.

[61] Andreea Ioana Sburlea and M Poel. The effects of light, priming and positive reinforcement on cognitive performance. *RETRIEVED on*, 10:14, 2015.

[62] Klaus R Scherer. Expression of emotion in voice and music. *Journal of voice*, 9(3):235–248, 1995.

[63] Klaus R Scherer et al. Psychological models of emotion. *The neuropsychology of emotion*, 137(3):137–162, 2000.

[64] Patrick Schober, Christa Boer, and Lothar A Schwarte. Correlation coefficients: appropriate use and interpretation. *Anesthesia & Analgesia*, 126(5):1763–1768, 2018.

[65] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition, 2015.

[66] Sarath Sivaprasad, Tanmayee Joshi, Rishabh Agrawal, and Niranjan Pedanekar. Multimodal continuous prediction of emotions in movies using long short-term memory networks. In *Proceedings of the 2018 ACM on International Conference on Multimedia Retrieval*, pages 413–419, 2018.

[67] Mats Sjöberg, Yoann Baveye, Hanli Wang, Vu Lam Quang, Bogdan Ionescu, Emmanuel Dellandréa, Markus Schedl, Claire-Hélène Demarty, and Liming Chen. The mediaeval 2015 affective impact of movies task. In *MediaEval*, 2015.

[68] Alan F Smeaton, Paul Over, and Aiden R Doherty. Video shot boundary detection: Seven years of trecvid activity. *Computer Vision and Image Understanding*, 114(4):411–418, 2010.

[69] Greg M Smith. *Film structure and the emotion system*. Cambridge University Press, 2003.

[70] Tomáš Souček and Jakub Lokoč. Transnet v2: An effective deep network architecture for fast shot transition detection. *arXiv preprint arXiv:2008.04838*, 2020.

[71] Kai Sun and Junqing Yu. Video affective content representation and recognition using video affective tree and hidden markov models. In *International Conference on Affective Computing and Intelligent Interaction*, pages 594–605. Springer, 2007.

[72] M. Svanera, M. Savardi, A. Signoroni, A. B. Kovács, and S. Benini. Who is the film's director? authorship recognition based on shot features. *IEEE MultiMedia*, 26(4):43–54, 2019.

[73] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Computer Vision and Pattern Recognition (CVPR)*, 2015.

[74] René Marcelino Abritta Teixeira, Toshihiko Yamasaki, and Kiyoharu Aizawa. Determination of emotional content of video clips by low-level audiovisual features. *Multimedia Tools and Applications*, 61(1):21–49, 2012.

[75] Ha Thi Phuong Thao, BT Balamurali, Dorien Herremans, and Gemma Roig. Attendaffectnet: Self-attention based networks for predicting affective responses from movies. In *2020 25th International Conference on Pattern Recognition (ICPR)*, pages 8719–8726. IEEE, 2021.

[76] Roy Thompson and Christopher Bowen. *Grammar of the Shot*. Taylor & Francis, 2009.

[77] Po-He Tseng, Ran Carmi, Ian GM Cameron, Douglas P Munoz, and Laurent Itti. Quantifying center bias of observers in free viewing of dynamic natural scenes. *Journal of vision*, 9(7):4–4, 2009.

[78] Patricia Valdez and Albert Mehrabian. Effects of color on emotions. *Journal of experimental psychology: General*, 123(4):394, 1994.

[79] Thurid Vogt and Elisabeth André. Comparing feature sets for acted and spontaneous speech in view of automatic emotion recognition. In *2005 IEEE International Conference on Multimedia and Expo*, pages 474–477. IEEE, 2005.

[80] Hee Lin Wang and Loong-Fah Cheong. Affective understanding in film. *IEEE Transactions on circuits and systems for video technology*, 16(6):689–704, 2006.

[81] Limin Wang, Yuanjun Xiong, Zhe Wang, Yu Qiao, Dahua Lin, Xiaoou Tang, and Luc Van Gool. Temporal segment networks: Towards good practices for deep action recognition, 2016.

[82] Shiguang* Wang, Zhizhong* Li, Yue Zhao, Yuanjun Xiong, Limin Wang, and Dahua Lin. denseflow. https://github.com/open-mmlab/denseflow, 2020.

[83] Lisa Wilms and Daniel Oberfeld. Color and emotion: effects of hue, saturation, and brightness. *Psychological research*, 82(5):896–914, 2018.

[84] Chung-Hsien Wu, Jui-Feng Yeh, and Ze-Jing Chuang. Emotion perception and recognition from speech. In *Affective Information Processing*, pages 93–110. Springer, 2009.

[85] Yun Yi and Hanli Wang. Multi-modal learning for affective content analysis in movies. *Multimedia Tools and Applications*, 78(10):13331–13350, 2019.

[86] Yun Yi, Hanli Wang, and Qinyu Li. Affective video content analysis with adaptive fusion recurrent network. *IEEE Transactions on Multimedia*, 2019.

[87] Athanasia Zlatintsi, Petros Koutras, Georgios Evangelopoulos, Nikolaos Malandrakis, Niki Efthymiou, Katerina Pastra, Alexandros Potamianos, and Petros Maragos. Cognimuse: A multimodal video database annotated with saliency, events, semantics and emotion with application to summarization. *EURASIP Journal on Image and Video Processing*, 2017(1):1–24, 2017.