Bárbara Maria Assunção da Silveira

Bachelor in Computer Science and Engineering

Semantic Video Quality Assessment

Dissertation submitted in partial fulfillment
of the requirements for the degree of

Master of Science in
Computer Science and Informatics Engineering

Adviser:	Nuno Manuel Robalo Correia, Full Professor, NOVA University of Lisbon
Co-adviser:	Rui Jesus, Adjunct Professor, Instituto Superior de Engenharia de Lisboa

Examination Committee

Chairperson:	Professor Doutor José Augusto Legatheaux Martins
Raporteur:	Professor Doutor Manuel João Carneira Monteiro da Fonseca
Member:	Professor Doutor Nuno Manuel Robalo Correia

FACULDADE DE
CIÊNCIAS E TECNOLOGIA
UNIVERSIDADE NOVA DE LISBOA

November, 2018

Acknowledgements

First and foremost I want to thank the Faculdade de Ciências and Tecnologia, NOVA that helped me to grow professionally. I would like to thank professor Nuno Correia - NOVA University of Lisbon and professor Rui Jesus - Instituto Superior de Engenharia de Lisboa for their continuous guidance, patience, availability and support throughout the realization of this work.

I also want to express my gratitude for the friends that accompanied me throughout my academic journey, that went through the moments of hardships and happiness with me.

Last but not least, I would like to express my deepest grattitude to my family that supported me on my journey, with their support and love. I thank my father and grandmother for their encouragement, my sister Lorena and Maria Clara for their willingness to help me and especially my mother for her love and understanding and for encouraging me and my decisions. I would also like to thank my dog, Mel, for being my source of joy.

Abstract

The increasing availability of high-speed internet connections, the increase in smartphone usability and also the ubiquity of social networking, all combined, help to create a great diversity of User-Generated Content (UGC). Along with this expansion, Ultra High Definition (UHD) broadcast technology has been developing rapidly since its beginning. This created the need to distinguish between good and bad quality videos.

The best way to assess the quality of a video is through the human eye. However, given the amount of content it becomes quite impractical. Therefore, computational methods are used. These methods try to assess it as close as possible to what would be assessed by the human vision.

The semantics of a video is the meaning of the video itself and using this information, an idea of what the video is about can be provided, helping even in the assessment of a video. Having that in mind, this thesis uses a video collection and a news articles collection in order to extract the information regarding the objects in the scene and the terms in the news. The similarity between both information is taken into consideration to assess the quality o the videos. In this way, the assessment is done using semantic information.

The main contributions of this work are the video quality assessment based on semantic information and an evaluation of a set of object detection algorithms used for semantic extraction in videos.

Keywords: User Generated Content, Video Quality Assessment, Semantic, Video Processing, Object Detection.

Resumo

O aumento da disponibilidade de conexões de internet de alta velocidade, o aumento da usabilidade do smartphone e a ubiquidade das redes sociais combinadas, criam uma grande diversidade de conteúdo gerado pelo utilizador. Acompanhando esta expansão, a tecnologia de transmissão de alta definição tem sido rapidamente desenvolvida. Isso criou a necessidade de distinguir entre vídeos de boa ou má qualidade.

O melhor método para avaliar a qualidade de um vídeo é através do olho humano. Contudo, dada a quantidade de conteúdo, tal método torna-se impraticável. Por isso, métodos computacionais são utilizados. Estes tentam avaliar o mais próximo possível do que seria avaliado pela visão humana.

A semântica de um vídeo representa o seu significado e ao utilizar essa informação, pode-se ter uma ideia sobre o seu conteúdo. Tendo isso em mente, esta tese usa uma coleção de vídeos e de artigos de notícia para extrair as informações sobre os objetos na cena e os termos encontados na notícia. A similaridade entre ambas as informações são consideradas para avaliar a qualidade de vídeos. Desta forma, a avaliação é feita utilizando informação semântica.

As principais contribuições deste trabalho são uma avaliação da qualidade de vídeos, baseando-se em informação semântica, e uma avaliação de um conjunto de algoritmos de deteção de conceitos que são usados para a extração semântica em vídeos.

Palavras-chave: Conteúdo Gerado pelo Utilizador, Avaliação da Qualidade de Vídeo, Semântica, Processamento de Vídeo, Detecção de Objetos.

Contents

List of Figures

List of Tables

Glossary

C4.5

C4.5 is an algorithm introduced by J.R. Quinlan which produces reasonable decision tree by using a set of training set or data set.

Data Mining

Data mining is the computing process of discovering patterns in large data sets involving converging methods from machine learning, statistics, and database systems.

Domain Knowledge

Domain Knowledge is a particular area of work or specific area to be worked upon.

Max-pooling layer

It is a function to progressively reduce the spatial size (using the MAX operation) of the representation to reduce the number of parameters and computation in the network.

SoftMax Function

Softmax function calculates the probabilities distribution of the event over 'n' different events.

Acronyms

CNN	Convolution Neural Networks.
DNN	Deep Neural Network.
FR	Full Reference.
GME	Global Mention Extension.
HVS	Human Visual System.
ILSVRC	ImageNet Large Scale Visual Recognition Challenge.
MPQM	Moving Pictures Quality Metric.
MSE	Mean Square Error.
NQM	Noise Quality Measure.
NR	No-Reference.
PSNR	Peak Signal-to-Noise Ratio.
RR	Reduced Reference.
SBF	Snipped-Based Features.
SSIM	Structural Similarity Index Metric.
SVM	Support Vector Machine.
TRECVid	TREC Video Retrieval Evaluation.
UGC	User-Generated Content.

UHD	Ultra High Definition.
VQM	Video Quality Metrics.

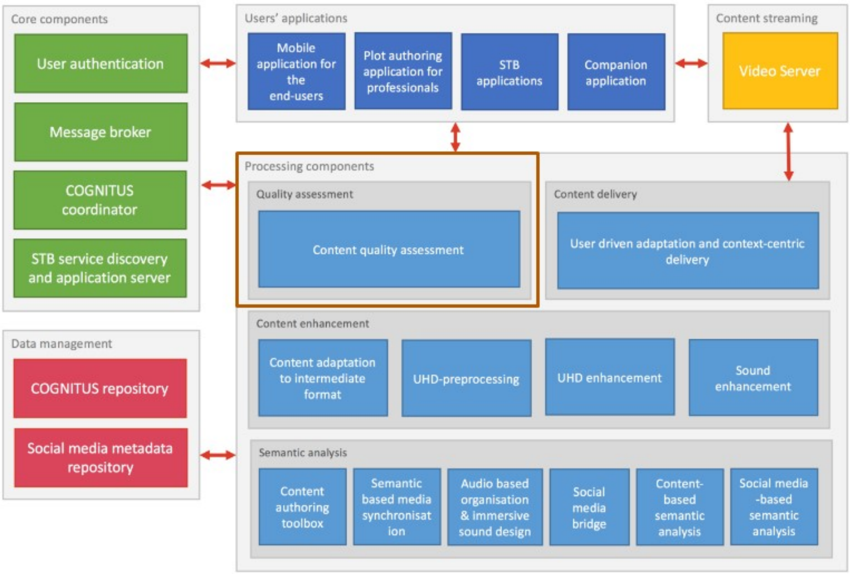
Chapter 1

Introduction

The purpose of this chapter is to give a description of the context, the motivation and the definition of the problem on which this thesis aims to solve. Later the contributions and the document structure are presented.

1.1 Context

This thesis is integrated into the Cognitus project, a joint research project comprising eight European participants, led by BBC Research & Development, being University



Core components
User authentication
Message broker
COGNITUS coordinator
STB service discovery and application server
Data management
COGNITUS repository
Social media metadata repository
Users' applications
Mobile application for the end-users
Plot authoring application for professionals
STB applications
Companion application
Content streaming
Video Server
Processing components
Quality assessment
Content quality assessment
Content delivery
User driven adaptation and context-centric delivery
Content enhancement
Content adaptation to intermediate format
UHD-preprocessing
UHD enhancement
Sound enhancement
Semantic analysis
Content authoring toolbox
Semantic based media synchronisat ion
Audio based organisation & immersive sound design
Social media bridge
Content-based semantic analysis
Social media -based semantic analysis

Figure 1.1: Cognitus Architecture

Nova Lisboa one of its partners. Having an almost 3 years of development, from January 2016 to December 2018, being integrated into the European Horizon 2020 initiative, so far the largest innovation and research program funded by the European Commission. This project has the intention to merge the advances in UHD broadcasting technologies with the massive amount of content generated by users for the sake of new interactive, immersive modes of production. It has the purpose to optimize how UHD content is produced and distributed, by exploiting the knowledge of professional producers, the universality of UGC and also including the power of social creativity from interactive networks. The requirements for the system to be implemented in this thesis result directly from use cases related to the topic Quality of Experience Enhancement (QOE) whose positioning can be seen in figure 1.1 [26][8].

1.2 Motivation and Problem Definition

The growth of technology evolved in a way that the access to a mobile device is available to almost everybody. This growth and the desire to share events with others in social media made the number of self-produced videos increase drastically. Despite the importance of these videos, since they capture the event in a reality that is closer to the user, each video has different quality, either by the way it was filmed or by the devices' brand. The videos filmed by professionals, although they have a better quality than self-produced video, the content might not be to the like of users. And so, it arose the need to combine both ways, so that the videos selected could have a good quality and also have a content that matters to the user. In order to make that true, both the semantics and the quality of the videos need to be taken into account. This can be achieved through many processes being one of them through the use of machine learning.

The human eyes are the best quality assessor, however, is quite impractical to ask a human being to assess the huge amount of videos available. Apart from the fact that this assessment is highly subjective and does not give a standard measure. Having that in mind, the objective measurement is used to give some standardization and produce results in useful time.

Given all the UGC, the access to videos related to events became vulgar. Considering the number of videos is safe to say that not all of them have either a good quality or a relevant content. The semantics of a video helps to identify objects and concepts in a video. Knowing the concepts or the objects of a video beforehand can be helpful since the user will be able to know if the particular video is of its interest or not. Besides, through its semantic, the user would be able to infer if the video follows some kind of a storyline or not. And so the problem here depicted, relates to the possibility to extract the semantics of the videos in order to help evaluate its quality.

1.3 Contributions

The use of semantics for assessing video quality is the main focus of this work.

In order to achieve this goal, the objects of the videos are detected and these detections are compared to the information found in the news articles, and combining the information of both, the videos are assessed as *good quality videos* or *bad quality videos* through the use of semantics.

Another aspect that was studied consists in verifying if a video tells a ”story” through time, which can imply that the content is relevant and that the video is structured.

Summarizing, as the result of the development of this thesis, here are the following contributions:

- Study of video semantic properties, considering temporal aspects;
- Evaluation of existing methods and algorithms to assess video semantics including object detection and classification;
- Assessment of the quality of videos using semantic information;

1.4 Document Structure

The presented document is structured into three chapters, which are following described.

Introduction In this chapter the context of this thesis is presented along with a clarification of the motivation and the problem definition. The contributions of this study and the structure of the document is also referred.

Related Work The second chapter is regarded to the related work. This chapter establishes a connection between the main goals of this dissertation and the related work previously published. The definitions of the concepts are presented and also some techniques regarding video assessment are made clear. It is elucidated how to assess a video and how the semantics can contribute for the evaluation of the video quality. The extraction of features from a video and their classification is also explained. At last, systems that use semantic to identify objects are presented along with some of the most used datasets.

Semantic Quality Assessment In this chapter, the model of the architecture followed in this thesis work is presented, being each component of it explained.

Implementation In the implementation chapter, each component of the model previously mentioned is thoroughly explained. Every detail of the implementation done in order to achieve the objective of this work is exposed. The algorithms used and how the classification process works is presented along with the explanation of the metrics used to assess the video.

Evaluation In the evaluation chapter, the results and an analysis of the tests done throughout the development of this thesis are presented. The data used for the tests are made clear and what were the motivations for each test. An observation of the results and a more profound analysis is also presented.

Conclusion This chapter presents what can be concluded with the work done, along with also some of the problems faced. Some improvement possibilities are presented with some research opportunities for future works.

Chapter 2

Related Work

The work of this thesis uses computer vision techniques. Its goal is to create autonomous systems to perform some of the tasks that the human visual system can perform and even try to surpass it [17]. Sub-domains of this include event detection, video tracking, object recognition, among others. The most relevant previous approaches will be presented in the following sections.

2.1 Video Assessment

Nowadays, digital videos are in our everyday life through different types of video applications, providing us videos with all kinds of quality. Although the human visual system is the one that performs the best when it comes to assessing the quality of a video (subjective assessment), it is not the most reliable since it is subjective to each person [42]. This created the need to develop and improve an objective assessment (which is accomplished by use of measurements), being its best characteristics the fact that they can be repeatable, standardized and can be performed quickly and easily using portable equipment, making easier to evaluate in a more standard way. The goal of the objective methods is to give results that correlate closely with results obtained through human perception [56].

2.1.1 Objective Assessment Methods

According to Shahid et al. [43], the objective assessment methods are divided into Full Reference (FR), Reduced Reference (RR), No-Reference (NR) depending on the amount of information that is available from the original video as a reference in the quality assessment:

- **Full Reference Methods:** This method computes the quality difference by comparing the signals of the original video and the received video. Typically, every pixel from the source is compared against the corresponding pixel at the received video, with no knowledge about the encoding or transmission process in between. FR metrics are usually the most accurate at the expense of higher computational effort.

- **Reduced Reference Methods:** This metric extracts some representative features of the original video in order to compare it with the corresponding information from the distorted video. This is done in order to give a quality score, being the later provided as input for RR methods. This method is used mostly when the original video is not available or when is impractical to do so. This makes them more efficient than FR metrics.

- **No-Reference Methods:** NR metrics tries to assess the quality of a distorted video without any reference to the original signal. Instead it searches for artifacts concerning the pixel domain of the video or uses information ingrained in the bitstream of the video. It can even perform quality assessment using a combination of the two approaches previously mentioned. Due to the absence of an original signal, this method is more efficient to compute, however they may be less accurate than the methods previously mentioned.

2.1.2 Video Quality Metrics

As observed by Wang et al. [55], nowadays the most used FR objective image and video distortion/quality metrics are the Mean Square Error (MSE) and Peak Signal-to-Noise Ratio (PSNR). These are the most used because of the simplicity in their calculations, the clear physical meanings and are also mathematically easy to deal with optimization purpose, however they are also criticized for not correlating well with observed quality measurement. A great effort has been made, for the past 3 or 4 decades, in objective image and video quality assessment methods (mostly for FR quality assessment) in order to incorporate perceptual quality measures by also considering Human Visual System (HVS) characteristics.

A survey made by Wang [54] was put together in order to provide a better understanding regarding each video quality metric. It was also his intention to compare each of the presented metric with PSNR in order to see if there were any advantages of using that metric over PSNR. In the end of the survey, a comparison of these metrics in terms of computational complexity was provided, as well as the correlation with subjective video quality measurement, along with the accessibility of each metric. The results can be seen in table 2.1. The metrics compared by Wang et al. were:

- **Peak-Signal-to-Noise-Ratio -** is derived by setting the MSE in relation to the maximum possible value of the luminance. The result is a single number in decibels, ranging from medium to high-quality video. PSNR is still the most popular metric to evaluate the quality difference among pictures, despite the fact that many models have already been developed.

- **Video Quality Metrics -** it measures the perceptual effects of video impairments including blurring, jerky/unnatural motion, global noise, block distortion and color distortion, combining them into a single metric. The testing shows that this metric has a high correlation with subjective video quality assessment and has been adopted by ANSI[1] as an objective video quality standard. It processes the video as input and computes it by following these steps:

  1. **Calibration:** In this step, the sample video is calibrated in order to prepare for feature extraction. The spatial and temporal shift, as well as the contrast and brightness offset of the processed video sequence with respect to the original video sequence is estimated and corrected.
  2. **Quality Features Extraction:** This step is responsible to extract a set of quality features that characterize perceptual changes in the spatial, temporal, and chrominance properties from spatial-temporal sub-regions of video streams using a mathematical function.
  3. **Quality Parameters Calculation:** In this step, a set of quality parameters is computed that describes perceptual changes in video quality by comparing features extracted from both the processed video and the original video.
  4. **VQM Calculation:** In the last step, the VQM is computed using a linear combination of the parameters calculated from the previous steps.

  Video Quality Metrics (VQM) can be computed using various models based on certain optimization criteria. These models include: Television; Videoconferencing; General; Developer and PSNR. One of the problems associated with PSNR is that it does not take the visual masking phenomenon into consideration, meaning that every single pixel error contributes to the decrease of the PSNR, even if is not perceived. This matter is addressed by means of incorporating some modeling of the Human Visual System. In particular, two key human perception phenomenon that has been intensively studied are **contrast sensitivity** and **masking**. The first phenomenon takes into consideration the fact that a signal is detected by the eye only if its contrast is greater than some threshold. The sensitivity of the eye varies as a function of spatial frequency, orientation, and temporal frequency. The masking phenomenon takes into account the human vision response to the combination of several signals. A stimulus consists of two types of signals - foreground and background. The detection threshold of the foreground will be modified as a function of the contrast of the background.

- **Moving Pictures Quality Metric -** is a metric for moving picture which incorporates two human vision characteristics (contrast sensitivity and masking). A decomposition of the original sequence and the distorted version of it into perceptual

[1] American National Standards Institute, https://www.ansi.org/

channels is the first step. Then, a channel-based distortion measure is computed taking into account the two human vision characteristics. In the end, the data is merged over all the channels in order to compute the quality rating which is scaled numerically from 1 to 5, being 1 considered bad and 5 excellent. MPQM does not take into consideration the chrominance and that is why the method Color MPQM (CMPQM) has been introduced. This metric represents the typical image quality assessment models based on the error sensitivity. The widely adopted assumption of these models is corresponding to direct relation between the loss of perceptual and the visibility of the error signal.

- **Structural Similarity Index Metric -** This metric took a different approach regarding the ones previously mentioned, since they were all error based, while this uses structural distortion measurement. The idea behind this is related to how the human vision system is highly specialized in extracting structural information from the viewing field instead of being specialized in extracting the errors. Therefore, this kind of measurement should provide a better correlation to the subjective impression.

- **Noise Quality Measure -** In this quality measurement metric, a degraded image is first modeled as an original image being that also modeled as an original image that has been subjective to linear frequency distortion and additive noise injection. These two sources of degradation are decoupled into two different quality measures being these measures: a distortion measure (DM) of the effect of frequency distortion, and a noise quality measure (NQM) of the effect of additive noise. The last one takes into account:

  1. Variation in contrast sensitivity with distance, image dimensions;
  2. Variation in the local luminance mean;
  3. Contrast interaction between spatial frequencies;
  4. Contrast masking effects.

The first measure is computed in three steps. The first one consists of finding the frequency distortion in the degraded image. The second is relative to compute the deviation of the found frequency distortion from an all-pass response unity gain. Finally, the deviation is weighted by a model of the frequency response of the human visual system.

An important step to a successful video quality assessment is validation. Therefore it is essential to build an image and video database with subjective evaluation scores associated with each of the images and videos sequences in the database. Then this database can be used to assess the performance of the objective quality measurement algorithms. The quality metrics produces a video quality score and these scores have to correlate with the subjective assessments given by the human evaluators. The entity

Comparison			
Quality Metric	**Mathematical Complexity**	**Correlation with Subj. Methods**	**Accessibility**
PSNR	Simple	Poor	Easy
Moving Pictures Quality Metric (MPQM)	Complex	Varying	Not Available
VQM	Very Complex	Good	Not Available
Structural Similarity Index Metric (SSIM)	Complex	Fairly	Available
Noise Quality Measure (NQM)	Complex	Unknown	Not Available

Table 2.1: Metric comparison

responsible to validate objective video quality metric models is the Video Quality Experts Group (VQEG) [5].

2.2 Video Semantic

The proliferation of the availability of video data is helping to create demand for methods that understands and manages videos at the semantic level [2]. So far, most of the information found regarding event detection frameworks was advanced towards videos with loose structures or without story units, e.g. sports videos, surveillance videos, medical videos. In opposition, the concept-extraction schemes were largely carried out on the news video, since those have content structures.

Most of these studies are operated in a procedure composed of two stages. The first is named as **video content processing** which consists of segmenting the video clip into certain analysis units and also extracting their representative features (usually keyframes). The following stage is **decision-making process** where the extraction of the semantic index from the feature descriptors occurs concerning the improvement of the framework robustness [46]. The algorithms used for the decision-making include *hidden Markov Model* and *controlled Markov chain*, since those model temporal relations among frames or shots of an event [4]. Hidden Markov Model scheme achieves promising results because it automatically discovers the statistical descriptions on high-level structures and also achieves slightly better accuracy in detecting discovered structures in unlabelled videos than a supervised approach [60]. The controlled Markov chain models, after adequately trained, provides a list of video segments that can be extracted to be able to represent a specific event of interest using maximum likelihood criteria [22]. Another type of heuristic method uses a set of heuristic rules, being those derived from the domain knowledge to map the feature descriptors to events. A data mining approach is also possible since it helps to mine the high-level semantics and patterns from a large amount of multimedia data [4].

According to Shyu et al. [46] the semantic analysis of videos related to sports usually involves two types of features: **cinematic** and **object-based**. Cinematic features refer to those that result from common video composition and production rules, for instance, shot types and replays. The objects are characterized according to their spatial features (color, texture, and shape) as well as for the spatio-temporal features (object motions, and interactions). Regarding object-based features, since they permit high-level domain analysis, their extraction can become computationally costly for real-time implementation while cinematic features balance well the computational requirements and the resulting semantics [12].

The analysis and detection of sports events detection generated a lot of attention given its great commercial potentials. Regarding the processing of video content, many of the literature adopted **unimodal** approaches that use only the visual, auditory, or textual modality.

Concerning the algorithms for detection of events and concept extraction, SVM is the most adopted one. Despite the fact that SVM presents an encouraging performance, the training process represents a problem since the scale and the increase of the data is not proportional. Another example of sports video analysis is C4.5 with regard to data mining method. The decision tree learning algorithm is mathematically less complex making it useful to deduce the mappings from low-level features to high-level concepts, being that way, able to select the representative feature items automatically [4].

As observed by Shyu et al. [46] there are different measures to construct the data mining procedure such as **distance-based**, **rule-based**, **instance-based**, **statistic-based** and some others procedures. The two most used are distance-based and rule-based. Since the detection of an event or a concept is considered a difficult task, the measurement of individual data mining cannot conclude well alone, requiring some support from certain artifacts. Through the results found from previous experiences it was clear that the detection capability is limited due to the **semantic gap** and **rare event/concept** detection issue.

Semantic gap (considered one of the main challenges) is regarded to mapping high-level semantic concepts into low-level spatio-temporal features that can be automatically extracted from video data. To deal with that, the rules to map is usually written into the code causing the inflexibility of the existing approaches and systems. Therefore, the use of domain knowledge is imperative to enable higher level semantics in favor of those being integrated into the techniques that capture the semantic through automatic parsing [2].

Rare event/concept (also known as imbalance dataset) issue occurs when there is a very small percentage of positive instances while a large number of negative instances dominate the detection model training process. This causes an undesirable degradation of the detection performance [46].

What was proposed by Shyu et al. [46] was a new framework that tries to offer a solution for the problems previously mentioned. It utilizes the multimodal content analysis and the distance-based and rule-based data mining techniques, being one of the best attributes the fact that it is automatic and does not need the domain knowledge. Considering this fact, this framework can be easily extended to various applications domains. The approach consists of using an improvement of a previously proposed distance-based RSPM algorithm [36] in order to perform the rough classification which includes the feature combination and selection. Then for a further classification, the rule-based algorithm C4.5 decision tree is employed. The relaxation of the domain knowledge was possible because many distance-based data mining schemes were adopted in order to ease the class imbalance issue and to also reconstruct and reduce the feature dimension. The particular detail of the framework developed is the ability to mitigate the rare event/concept detection and semantic gap problems without the need to rely on the artifacts or the domain knowledge.

Wu et al. [59] proposed a three-layer near-real-time event inference scheme for sports event recognition. In the first layer, a Global Mention Extension (GME) algorithm is used to separate the frame's foreground from the background. Then, low-level features are extracted from both. And so the system automatically segments the sequences of frames into clip as basic semantic inference units. Thereafter, the semantic concepts of these clips are extracted in favor of giving a semantic description. Lastly, in the third-layer, rule-based finite-state machines are designed for event inference. The results of this experiment are good since it can recognize sports events precisely.

To assess a video through the use of semantics, a combination between the subjective and objective evaluation techniques are used so that it is possible to compare the performance of both methods. The subjective assessment includes the comparison of frames and frames details.

In some cases, the importance of the content can be compared with the importance of the perceptual quality for the user satisfaction. And so the quality assessment of a multimedia signal, done by the subjective tests, that focus only on the quality scales can fail in relating to the higher cognitive processes of a human perception [20]. Some studies show that there is a substantial correlation between the subjective rating of video quality and the content of the video. Therefore the content of a video given by the semantics needs to be considered in a video quality assessment [44].

2.2.1 Video and Text Aligment

Understanding the story of videos is not an easy task. The alignment of the video with texts is one way to do it.

One of the analysed case of alignment between video and text was the work done by Tapaswi. In this work, the author used mainly two sources of text in order to understand the video while focusing on the aspects of the story, being those *Plot synopses* and *Books*.

Both sources of texts come with problems when it comes to alignment, given that the plot synopses is too summarized containing very little detail while, the book, on the other hand, contain too much detail that it is not portrayed in the video. The solution for both problems passes by selecting parts of the texts that have a direct link with the video.

The alignment of texts and videos requires that the difference in the data representation must be taken care of. The author does the alignment by finding the characteristics of the data that can be found in both the text and the video. They use the notion that stories are strongly character-centric. And so for a good alignment, the crucial component is the detection and analysis of the character names in the text and the identification of characters in the video.

The similarity $\Phi(t,v)$ is defined using each segment in the video, $v \in V$, and chunks of text in the document, $t \in T$, computed at every unit of the alignment. Being that the case, the alignment problem is transformed into an optimization problem since it maximizes the similarity between the text and the video while respecting some story progression constraints.

For the plot synopses, they proposed a fine-grained approach. Which means that they tried to align individual shots of the video with each sentence from the plot synopsis. For the books, a coarse grained alignment was done - the video scenes which correspond to particular chapters was found and used for the alignment [49].

2.3 Features Extraction and Classification

According to Motoda and Liu [29] after some experiments and research, researchers realized that a pre-processing stage is a fundamental part in data-mining. The pre-processing consists of the processing the data before it is used as input for an algorithm. The main ambition of feature extraction, selection and construction is divided into three main points:

1. reducing the amount of data;
2. focusing in relevant data;
3. improving the quality of data and hence the performance of data mining algorithms (learning time, predictive accuracy).

2.3.1 Feature Selection

The work done by Motoda and Liu [29] provided a better understanding regarding the selection of features. According to some evaluation criteria, this process is a search problem that chooses from the original set, a certain subset of features so that the feature space is reduced given some criteria.

- *Feature Subset generation* - The easiest way to generate subsets of features is sequentially. It can be either *sequential forward selection* or *sequential backward selection*. Another possibility is to randomly generate the subsets.

- *Feature evaluation* - The optimity of a subset is always based on a particular criterion of evaluation. The evaluation criteria are either an *independent* or a *dependent* criteria being that defined on how they depend or not on the learning algorithm. Independent criteria are **distance measure**, **information measure**, **dependency measure** and **consistency measure**. Dependent criteria are evaluated through the performance of the algorithm. In supervised learning, the objective is to maximize the **predictive accuracy** whereas for the unsupervised learning there are several heuristic criteria that estimate the quality of clustering results (**cluster compactness**, **scatter separability** and **maximum likelihood**)

Algorithms/Methods [19]

1. *Chi-squared* - most common that measures divergence from the expected distribution;

2. *Euclidian Distance* - examines the root square differences between coordinates of pair of objects;

3. *T-test* - assesses if the average of two groups are statistically different from one another;

4. *Information gain* - measures the increase in entropy when the feature is given in contrast to when is absent;

5. *Correlation-Based Feature Selection* - searches feature subset given the degree of redundancy amidst the features.

2.3.2 Feature Extraction

Motoda and Liu [29] clarify the concept of feature extraction. This process subsists in the extraction of a set of new features through some functional mapping from the original set, being the new set smaller than the original set. The main target of it, is to search for a smaller set of new features after the original has been transformed according to some performance measure.

- *Performance measure* - It selects what is the most suitable in order to evaluate the extracted features. For classification, the predictive accuracy can be used to determine the set of features while for the clustering the measure used is for example inter-cluster/intra-cluster similarity, variance among data.

- *Transformation* - It takes into consideration the ways of mapping original attributes to new features being the mapping either a linear transformation or a nonlinear transformation. The transformation can be both linear and nonlinear and labeled and non-labeled. The most used techniques are K-Means, K-Medoids, Multi-layer Perceptrons.

- *Number of new features* - Is regarding the minimum number of new features after the transformation that can fully represent the original set. The characteristics of the data can be seen as a critical constraint regarding the aspects previously mentioned. In addition, data attributes can be of many types: continuous, nominal, binary, mixed.

  Feature extraction can have many usages: dimensionality reduction for further processing, visualization, compound features used to booster some data mining algorithms.

Algorithms/Methods [19]

1. *Independent Component Analysis* - Linear transformation in which the wanted representation is regarding the minimization of the statistical dependence of the components of the representation.

2. *Principal Component Analysis* - Is an orthogonal transformation that converts samples from correlated variables into samples of linearly uncorrelated features.

2.3.3 Feature Construction

Researchers Motoda and Liu [29] also explained about feature construction. This process discovers the missing information regarding the relationships between features and amplifies the feature space supported by the derivation or the creation of additional features. The feature construction main goal is to increase the expressive power of the original features.

- *Construction of new features* - The approaches can be divided into four groups: data-driven, hypothesis-driven, knowledge-driven and hybrid.

- *Choice and design operators for feature construction* - The most frequently used constructive operators for nominal features are conjunction, disjunction, and negation. Regarding numerical features, the operators are the algebraic operators.

- *Use of operators to construct new features efficiently* - It studies the connection between data mining tasks, data characteristics, and other operators considered effective.

- *Measurement and selection of useful new features* - In order to avoid too many features a selection technique is applied to remove abundant and irrelevant features.

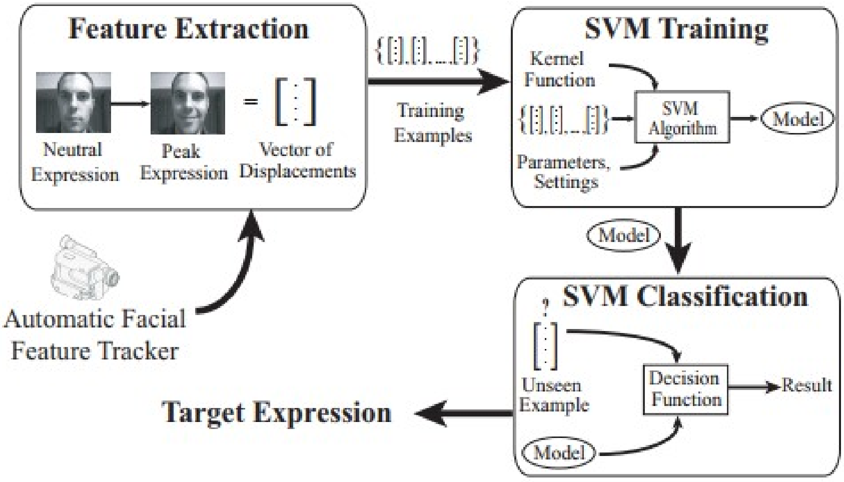


Figure 2.1: The stages of SVM-based automated expression recognition taken from [28]

2.3.4 Classification by Support Vector Machine

Support Vector Machine is a supervised machine learning algorithm which can be addressed for problems regarding classification and regression. The idea of this algorithm is that the input vectors are mapped to high-dimensional feature space in a non-linearly way being each data plotted as a point in the n-dimensional space (being n, the number of features) where the value of each feature is the value of a certain coordinate. Thereafter, the classification is performed by finding the hyperplane that helps to differentiate the two classes precisely [53]. This algorithm was originally implemented for separable training data however in the work done by Cortes and Vapnik in [10] it was also extended to non-separable training data.

Michael and Kaliouby [28] did work regarding the classification of real-time facial feature by SVM. In their paper they used a real time facial feature tracker in order to deal with the obstacle of face location and the feature extraction in spontaneous expressions. Their tracker extracted the location of 22 facial features from the video stream and also used a filter that tracks their position on the following frames. The displacements for each feature is calculated between the neutral and the frame that is representative of each one of the expressions. The scheme on the figure 2.1 presents the structure used in this work.

The algorithm (SVM) receives the input in its training phase, then builds a model of it and afterward gives the output being this one a hypothesis function that can be used to predict some future data. One of the SVM stronger points regarding previous algorithms is that it allows some intuition and human understanding. Another point is that they deal better with noisy data and overfitting.

The implementation followed by Michael and Kaliouby [28] uses *libsvm*[2] as the underlying SVM classifier. It encapsulates its stateless functionality in an object-based manner in favor of working in an incrementally trained interactive environment. The user has to request the training examples to be gathered at non-contiguous time intervals and also provide a label separately. Then is further combined with the displacements output that came out of the feature extraction phase and then added as a new example to the training set, being the SVM retrained.

The results that were obtained from this shows that the properties of the algorithm associates well with the restrains related to recognition accuracy and also regarding the speed by a real-time environment.

A problem faced was related to the inaccuracy when it comes to the movement of the head such as nodding and head tilting. That was dealt with the normalization of all the feature displacements concerning the root feature, which gives an approximation to head motion in the video stream.

SVM is a highly used algorithm for image retrieval. The algorithm selects the most informative images to query the user and quickly learns a boundary that separates the images that satisfy the concept chosen by the user from the rest of the dataset. The results provided by the experiments using SVM instead of the conventional query refinement schemes, has shown a significant improvement in the search accuracy [52].

2.3.5 Classification by DNN

An Artificial Neuron (AN) is basically an engineering approach of a biological neuron. It has *n* inputs and one output. An Artificial Neuron Network consists of a large number of simple processing elements (AN) that are interconnected with each other [45].

A deep neural network is an artificial neural network that has more than two layers, adding a certain level of complexity to it. This type of network uses sophisticated mathematical modelling in order to process data in complex ways.

The learning method used by the neural net learning algorithm consists of processing several examples with the ”answers” given and, using the answer, it learns what characteristics of the input are needed to construct the correct output. After a sufficient number of examples have been processed, the processing of new information can begin in order to successfully give the right results. The accuracy is proportional to the numbers of examples previously seen, which means that, the more examples and new inputs the program sees, the more accurate the results are given that the program learns with experience [30].

CireşAn et al. did a work using a multi-column deep neural network to classify traffic signs. The work developed won the final phase of the German traffic sign recognition benchmark. They used a fast, fully parametrized GPU implementation of a Deep Neural Network and combined various DNNs, each trained on differently preprocessed data, into

[2] Popular open source machine learning library.

a Multi-Column DNN (MCDNN). The MCDNN improves the performance of the recognition and also makes the system indifferent to variations of contrast and illumination.

For the basic building block, a deep hierarchical neural network that alternates convolutional with max-pooling layers was used. The last layer of the classification is a fully connected layer with one output unit per class in the recognition task. The softmax activation function is used in order to be able to interpret the output as a probability of a certain image belonging to a certain class.

The first step to train a single DNN is to pre-process a given dataset and then during the training, continuously distort it. The preprocessing of the data consists of: cropping the images and processing only the images inside the bounding boxes; visually inspect it; resize the images to 48x48 pixels, and do the contrast normalization.

The MCDNN is formed by averaging the output of several DNN columns.

The MCDNN developed by them had a recognition accuracy rate of 99.46% which it was better than the recognition done by humans (98.31%). They came to conclude that the Multi-Column Deep Neural Network they developed in their work had a recognition rate of 98.52%–99.46%. And also that combining the preprocessing methods into a MCDNN, increases robustness to various types of noise and gets more traffic signs recognized [6].

2.3.6 Classification by Convolutional Neural Network

CNN is a class of deep, feed-forward artificial neural networks being composed of one or more convolutional layers followed by one or more fully connected layers being the last one standard multilayer neural network. Its design provides minimal pre-processing. Its architecture takes advantages of the 2D structure of the input image, being that carried out by local connections and tied wights followed by some form of pooling which results in translation invariant features. Another advantage of this algorithm is that is easier to train along with the fact that requires fewer parameters when compared with fully connected networks that have the same number of hidden units [9]. The CNN structure can be seen in Figure 2.2.

A study in video classification using CNN was made by Karpathy et al. [18]. In order to speed up the runtime performance of CNN, its structure was modified in order to contain two separate streams of processing: context stream and a high-resolution *fovea*[3]

[3] In the eye, a tiny pit located in the macula of the retina that provides the clearest vision of all

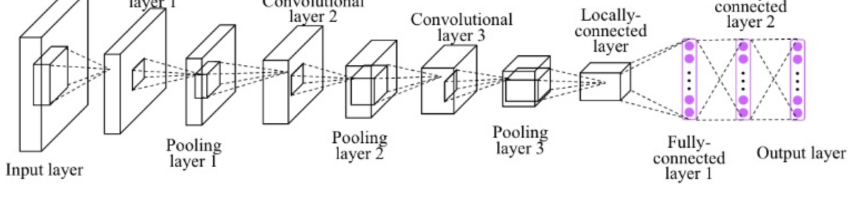


Figure 2.2: CNN Structure taken from [11]

stream. The first one learns features in low-resolution frames while the second stream only operates on the middle portion of the frame.

Dissimilar to images, videos show a great variation in the temporal extent and so cannot be easily processed with a fixed-size architecture. Having that in mind, this work treated the videos as a set of short, fixed-sized clips and since each clip contains many contiguous frames in time, they could extend the connectivity of the network in the time dimension in favor of learning spatio-temporal features.

Their model was trained for over a month, with models processing approximately, per second, 5 clips for full-networks and up to 20 clips per second regarding the multi-resolution networks on a single model replica. In order to produce predictions for the entire video, each one of the 20 randomly selected clip was individually presented to the network and propagated through it four times. Then the average of the network class predictions was calculated so that it could produce a more robust estimate of the class probabilities.

The features are of two types: *local features* and *global features*.

In their work, they found that a multi-layer network performs consistently and significantly better than linear models regarding separated validation experiments.

Karpathy et al. concluded with their work, that CNN architectures are capable of learning powerful features from weakly-labeled data that outperforms the feature based methods in performance. Notably, it was found that a single-frame model already displays very strong performance, which suggests that local motion cues may not be critically important, even for a dynamic dataset such as Sports.

Since the results for object detection with CNN were good, even better ways to detect were created such as Region CNN (R-CNN), Fast R-CNN and Faster R-CNN. *R-CNN* improves CNN through the addition of a bounding box to exactly identify the location of the main object in the image [1]. The *Fast R-CNN* solved the R-CNN main problem, which was that even though it works well, it is considerably slow. This problem was solved by the use of Region of Interest Pooling (RoIPool). The core of RoIPool consists of sharing the first step of CNN across its subregions, meaning that it combines all models into one network [1, 15]. Despite the advances made, there was still a bottleneck, being that, the region proposer. And so the *Faster R-CNN* solved it after taking into consideration that the region proposals depended on features of the image that were already calculated in the first step of the classification. Considering that, the idea was to reuse those results for region proposals instead of running a separate selective search algorithm [1, 39].

2.4 Object Detection Systems

Object detection is part of computer vision that detects objects using semantics. The systems created for this purpose use datasets in order to be able to identify the objects. In this section, some datasets will be presented along with some of the developed systems.

2.4.1 Datasets

Many datasets were created regarding detection and recognition of objects, being the ones that follows the most known and used.

2.4.1.1 Pascal Visual Object Classes

According to Everingham et al. [13], the Pascal Visual Object Classes Challenge is a benchmark in visual object category recognition and detection, providing both the vision and machine learning communities with a standard dataset of images and annotation, along with standard evaluation procedures. This dataset became accepted as a benchmark for object detection.

The challenge tasks are **classification** and **detection**.

- *Classification* - The goal of this task is to predict the presence as well as the absence of at least one object of a certain class in a test image. The classification methods consist mostly of variations of the bag-of-words method. This consists of computing local features using, for instance, SIFT[4] descriptors. The vector is estimated usually by k-means into a visual vocabulary and then each image is represented by a histogram that shows how often these local features are assigned to each one of the visual words. The classifier is normally a SVM or an Earth Mover's Distance[5] kernel.

- *Detection* - This task predicts the bounding boxes of each object of the same class in a test image, with associated real-value confidence. The method used for detection was the *sliding window*. This consists of taking a rectangular window of the image, posteriorly features are extracted from it to later be classified as either containing an instance of a certain class or not. The classifier then runs over the image at different location and scale.

2.4.1.2 ImageNet

Russakovsky et al. [40] presented an overview on ImageNet Large Scale Visual Recognition Challenge (ILSVRC). ILSVRC is a benchmark that started to run in 2010 in object category classification and detection on several of objects categories and images. This benchmark follows from the PASCAL VOC which sets the standardized evaluation of recognition algorithms.

ILSVRC is divided into two components: **publicly available dataset** and an **annual competition** (which is not going to be focused here). The dataset allows the development and the comparison of categorical object recognition algorithms.

[4] Scale Invariant Feature Transformation - is an algorithm in computer vision to detect and describe local features in images.

[5] Is a method to evaluate dissimilarity between two multi-dimensional distributions in some feature space where a distance measure between single features is given.

The annotation of ILSVRC can be discriminated into two categories:

- image-level annotation of a binary label -> either 0 or 1 in case of presence or absence in the image of a certain object class.
- object-level annotation of a bounding box and the class label around an object instance.

The main motivation for the development of algorithms that can distinguish classes that are visually very similar, was the huge amount and diversification of object categories.

Regarding the construction of a large-scale object recognition image datasets, it has three main steps:

1. *Image Classification* - This step defines the set of target object categories, which is selected from the existing ImageNet categories. Through the use of WordNet as a foundation, the ImageNet deals with ambiguous word meanings and the combination of synonyms into the same object category. The combination of an automatic heuristics with a manual post-processing helps to create a list of target categories adapted to each task.
2. *Single-Object Localization* - It collects varies set of possible images so that it can represent the elected categories. The strategies used on many search engine to do the image collection are both automatic and manual. The process modifies according to the ILSVRC tasks.
3. *Object Detection* - This last step is the annotation of the millions of the collected images in order to obtain a clean dataset, being for each individual task, a crowd-sourcing strategy designed. For object detection this dataset consists of 465.567 images for training and 20.121 images used for validation for 200 different classes of varied types [14].

The paper of Russakovsky et al. also presents some of the criticisms associated with it, being those related to the insufficiently challenging dataset given that the objects are usually large and centred in the images. Another problem faced is associated with the fact that the datasets grow larger in scale, which can become impossible to fully annotate them manually.

Russakovsky et al. also presents possible future works in the area. For instance, the growth of unlabeled or partially labelled largescale datasets implies two essential things: the algorithms will have to rely more on weakly supervised training data and will also have to first make a prediction in order to be evaluated later. Which means that instead of evaluating accuracy[6] or recall[7], these algorithms will focus more on predictions[8].

[6] How many of the test images or objects does the algorithm get it right.

[7] How many of the desired images or objects does the algorithm manage to find.

[8] How many of the predictions made by the algorithm were assumed correct by humans.

Since the field of machine learning started to grow, it opened new ways to several things, which included ImageNet. In order to improve performance, larger datasets need to be collected, more powerful models need to be learnt and better techniques to prevent overfitting should also be used. To learn about an considerable amount of objects from an even bigger number of images, a model with a big learning capacity is required. So the model used should have a lot of prior knowledge to compensate for the lack of data, which is why CNN are used. Comparing the usually taken approach with CNN, it was possible to conclude that it has fewer connections and parameters making it easier to train. The results that came out from this work provides the best result on these datasets [21].

2.4.1.3 Microsoft Common Objects in Context

The main goal of the presented dataset (COCO) [23] is to address the three fundamental problems in scene understanding, being those the detection of non-iconic views of objects; the contextual reasoning between objects; and the precise 2D location of the objects. So that the database could take care of the three mentioned problems, the authors employed a novel pipeline to gather data with the use of Amazon Mechanical Turk. The first step of the creation lies in yield a large set of images that encloses contextual relationship and non-iconic object views. This was accomplished by an effective technique, that queries for pairs of objects together with images, retrieved via scene-based queries. The next step involved that each image was labelled as containing certain object categories through the use of a hierarchical labelling approach. For each category found, the individual instances were labelled, verified and in the end, segmented. Considering the ambiguity that labelling is subject to, each of the stages had many tradeoffs.

The biggest distinction between this dataset and the other (PASCAL VOC and SUN) lies in the number of labelled instances per image which can assist in learning contextual information. Another important property is that the images on the dataset are non-iconic images that contain objects in their natural context, being the amount of contextual information in each image estimated through the examination of the average number of object categories and instances per image.

Bounding-box detection - The experiment took a subset of 55,000 images and did tight-fitting bounding boxes from the annotated segmentation masks. Two different model were evaluated: **DPMv5-P** and **DPMv5-C**. The first being the last implementation trained on PASCAL VOC 2012 while the later used the same implementation but trained on COCO. The comparison of the average performance of DPMv5-P on PASCAL VOC and MS COCO showed that the performance of the later is inferior, which suggest that this dataset includes more difficult images of objects. The same happened with the second model.

Generating segmentations from detections - Following from previous works, a simple method for generating object bounding boxes and segmentation masks object is achieved through learning the aspect-specific pixel-level segmentation masks for different categories. These are learned by averaging together segmentation masks from aligned training instances.

Detection evaluated by segmentation - Even after the assumption of a detector that reports correct results, segmentation can be considered a difficult task since it requires a fine localization of object part boundaries. The criteria established were demanded the standard requirement which requires that the intersection of the union between predicted and ground truth boxes needs to be at least 0.5.

2.4.1.4 TRECVid

The TREC Video Retrieval Evaluation (TRECVid) is an international benchmarking activity to help promote a content-based use of digital video via open, metrics-based evaluation. To make it possible, TRECVid provides a large test collection, uniform scoring procedures, and a forum so that interested organisations can compare their results. This benchmark is involved with both interactive and automatic/manual search for shots. These shots can come from within a video corpus, automatic detection of a diversity of semantic and low-level video features, shot boundary detection and also through the detection of story boundaries in broadcast TV news. TRECVid is funded by the NIST[9], being also supported from other US government agencies [47] [34]. TRECVid 2014 continued a total of five tasks from the previous year, being those: **Semantic indexing**, **Instance search**, **Multimedia event detection**, **Multimedia event recounting**, **Surveillance event detection** [34].

For the semantic indexing, the data came from short videos from the **Internet Archive under Creative Common** licences. Those videos were included since it has a wide variety of content style and source device. The **BBC EastEnders** was used for the instance search task. The **Heterogeneous Audio Visual Internet Corpus** collection of Internet videos was used for development and testing in multimedia event detection task. The tasks were judged by the NIST assessors [34].

2.4.2 Systems

Several systems were developed to detect objects in real-time using the datasets previously mentioned.

2.4.2.1 YOLO

YOLO9000 is a real-time object detection system [37]. They propose a new method to tame the considerable amount of classification data that they already have and use it to expand the scope of current detection systems. The method used consists of a hierarchical

[9] The National Institute of Standards and Technology is a measurement standards laboratory, and a non-regulatory agency of the United States Department of Commerce.

view of object classification that makes it possible to combine distinct datasets together. The method used is the WordTree that combines data from different sources. The paper also proposes a joint training algorithm that concedes the chance to train object detectors for detection and also for classification data. The proposed method takes advantage of the labelled detection images so that it can learn to precisely localize objects while it uses classification images to increase both the vocabulary and the robustness.

The approach taken by YOLO consists of applying a single neural network to the full image instead of hard negatives. The image is divided into regions by the network to later predict the bounding boxes (which are weighted by the predicted probabilities) and the probabilities for each region.

The model looks, during test time, at the whole image and so the predictions it makes are related to the global context of the image. And since it makes predictions with a single network, unlike the other systems, it results in a faster system [61].

The limitations of this system consist of the detection regarding small objects or unusual aspects ratios [31].

A result of this system detection can be seen in figure 2.3.

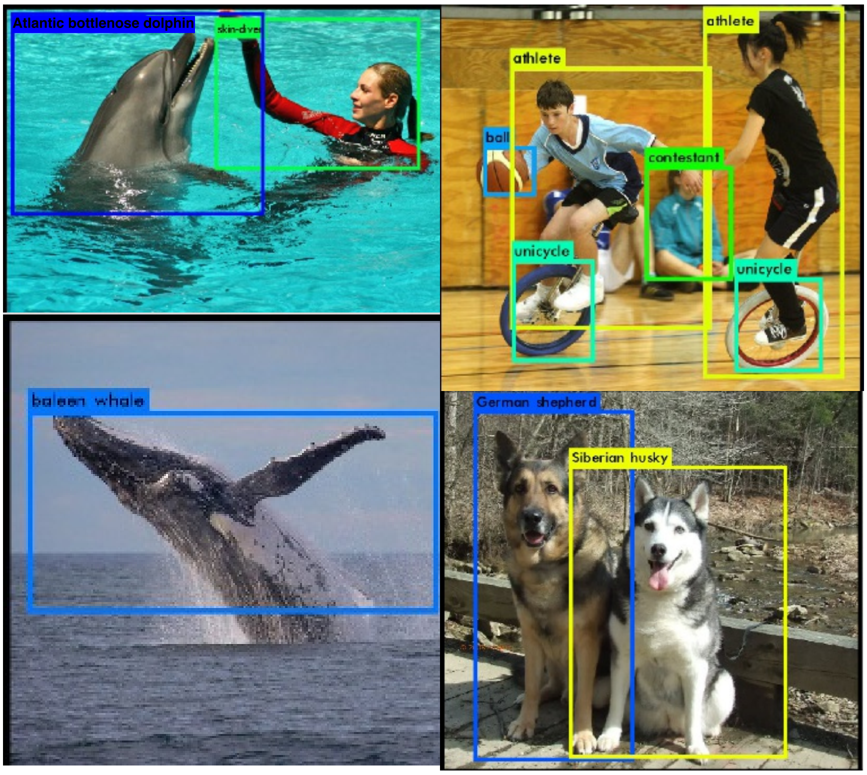
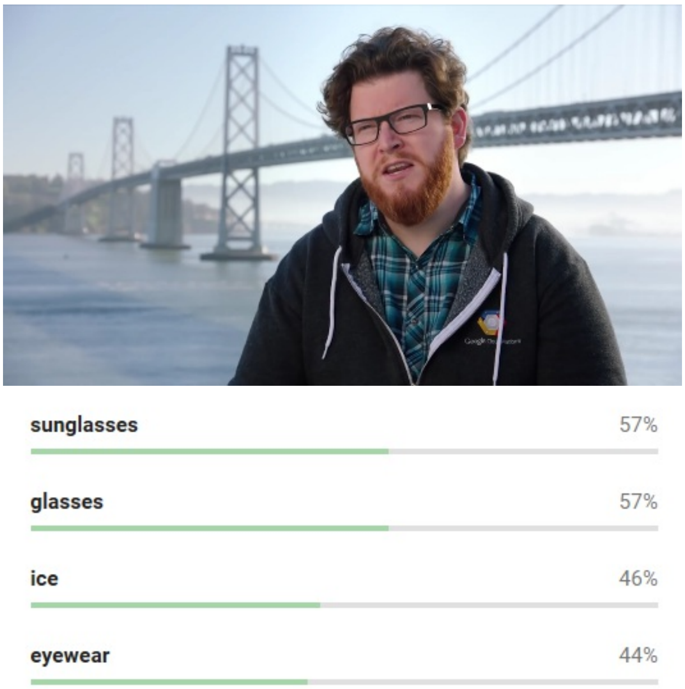


Figure 2.3: Detection of objects of different classes using YOLO, adapted from [37]

2.4.2.2 Video Intelligence API

Cloud Video Intelligence API for video analysis was recently introduced by Google. The system can separate signal from noise, through the information retrieved from the video, shot or frame. The API uses deep-learning models and works by selecting a video for annotation, then it detects the objects within the video (labels), scene changes and also the description of the video events over time (shot labels). Since its a Google API, it is available to developers so they can build an application that search, automatically, within the videos [16].

A demonstration of the process of detection made by the API can be seen in figure 2.4.



sunglasses 57%

glasses 57%

ice 46%

eyewear 44%

Figure 2.4: Detection of objects via Video Intelligence API, taken from [7]

2.4.2.3 OpenCV 3 Tracking API

OpenCV 3 incorporated a new tracking API that has implementations of many object tracking algorithms, being those BOOSTING, MIL, KCF, TLD, MEDIANFLOW, and GOTURN [32]. The process of tracking consists of firstly opening a video and selecting a frame, then defining a bounding box containing the object for the first frame and initializing the tracker with both the frame and the bounding box. The last step is to read the frames from the video and update in a loop in order to obtain a new bounding box for the current frame.

The goal of tracking is to find an object in the current frame given that the object has been successfully tracked in all (or most) of the previous frames. Considering that the object has been tracked it is possible to know the parameters of the **motion model** (location and velocity - speed and direction of motion) of the object in question. Besides the motion model, an **appearance model** is also built that shows how the object looks. This model also helps the previous model to predict more accurately the location of the object. This model is a classifier that is trained in an online manner, meaning that it trains at runtime.

2.5 Previous work

This thesis follows the previous work done in another thesis by Pedro Martins, entitled *Sistema para Avaliação Semiautomática de Vídeo* [27]. The structure followed by this work, is shown in figure 2.5.

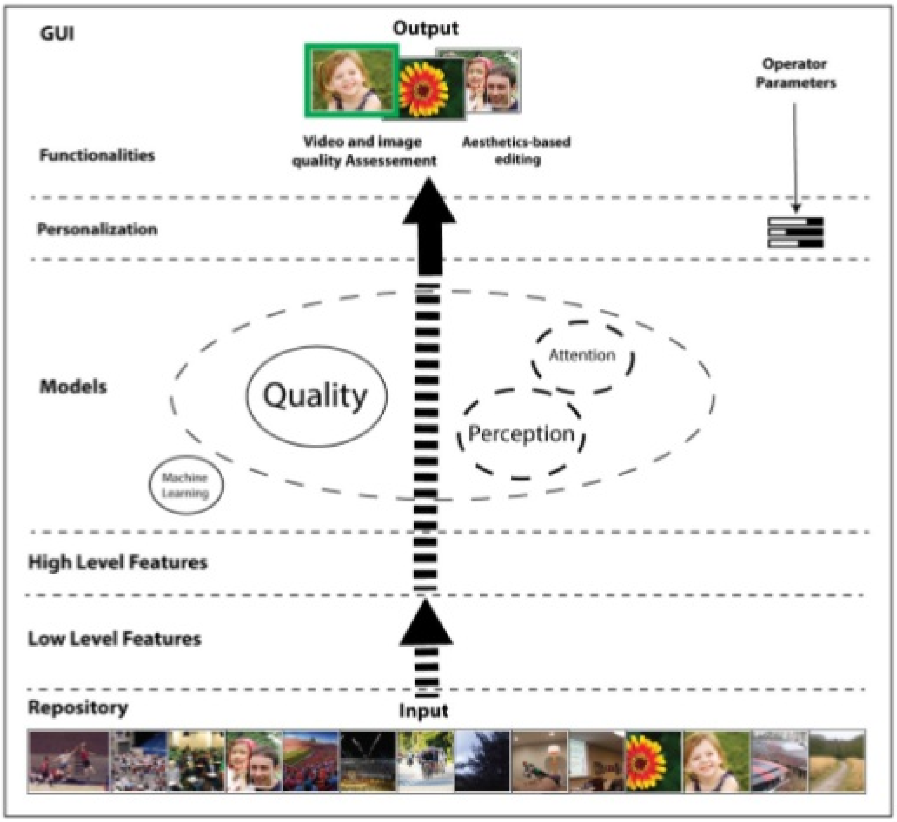


Figure 2.5: Previous thesis application diagram [27]

The approach tries to combine on a single graphical interface, a balanced set of recuperation video tools that gives an efficient visual discrimination. Tools that are simple to filter and order, based on a big variety of properties and easy concepts that are easy to understand by humans. It includes another tool that searches for color and texture similarity and another one that provides an automatic binary prediction about aesthetic and visual interest.

Besides the graphical interface, there are two other aplications, one for the extractions and computation of features and the other one for the training and testing of the learning algorithms through the use of machine learning.

Graphical interface The graphical interface offers a simple way to look for files in a video repository and starts with the load of the metadata generated by the feature extraction application. The menu permits three different natures of video recuperation tools. The most basic feature relates to the filtering and organization of the videos given the representative values of the visual characteristics extracted from the video repository. The second category of features permits, through the pre-extracted indexation values that are also included in the metadata, to organize videos given its color and texture. At last, it gives the possibility to select or exclude videos automatically according to a binary criteria related to aesthetic and visual interest.

Feature extraction and computation The feature extraction is parted in three different categories: General features (e.g. luminance, focus, texture, face aerea, etc.); Optional features (e.g. shakiness, foreground ratio, dynamic saliency, etc.); and Global Settings (e.g. frame resize; sampling factor; optical flow settings).

To process the videos the size of the sliding window adjusts according to the category of the feature it wants to extract. The result is a set of numeric values that describes interesting visual properties.

Classification through machine learning The experimental values are agregated with the values from the features extraction and computation phase in a machine learning procedure that uses SVM to create the binary classifiers regarding aesthetics and visual interest.

The technologies used in this are:

- *OpenCV* for the feature extraction and computation;
- *3.5.1 CERTH-ITI-VAQ700* regarding the dataset.

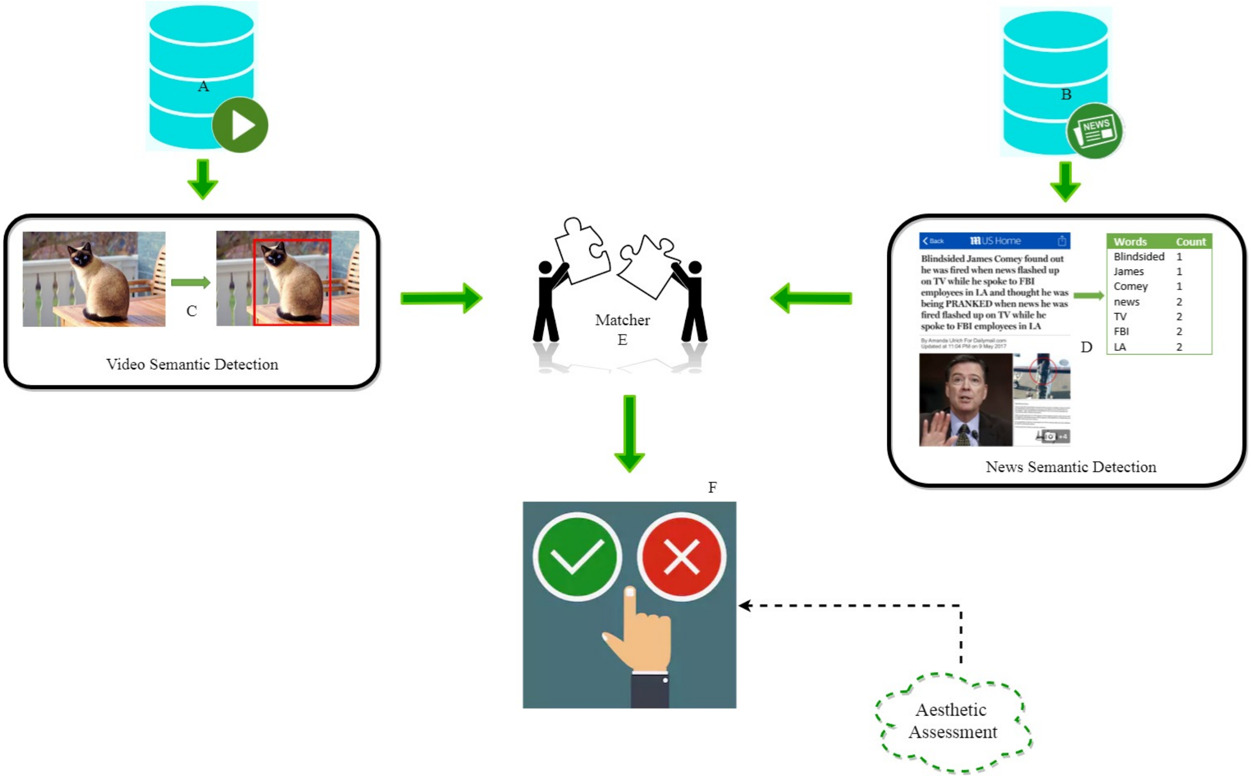
The results from this work proved to be very effective in basic tasks of visual discrimination.

Chapter 3

Semantic Quality Assessment

As it was concluded in the previous chapters, assessing the quality of a video given its semantic it is not an easy task and might not be as easy as most people think. Having that in mind, a model was created that given a video, the user would know its quality according to its semantic features.

The model can be seen in figure 3.1.



Blindsided James Comey found out he was fired when news flashed up on TV while he spoke to FBI employees in LA and thought he was being PRANKED when news he was fired flashed up on TV while he spoke to FBI employees in LA

By Amanda Ulrich For Dailymail.com
Updated at 11:04 PM on 9 May 2017

Words	Count
Blindsided	1
James	1
Comey	1
news	2
TV	2
FBI	2
LA	2

Figure 3.1: Architecture Model

As we can see, the model is divided into six different components, (**A**, **B**, **C**, **D**, **E** and **F**) each one corresponding to a different element that is essential to achieve the semantic assessment of the video quality.

The component signalized as **A** is the **Video Collection**. This component is, as the name suggests, a collection of videos. Considering that the main goal is to assess videos, a collection of videos was needed in order to provide the data for the videos assessment.

The element flagged as **B** is the **News Collection**. Much like the first collection, this one aggregates news. The news are needed in order to compare the content of the news with the content of the videos.

The **Video Semantic Detection** is represented by the element **C** in the model. This component is one of the most important elements of this model. It is here that the videos collected are used for object detection using the object detection algorithms/classifiers. As an example of these algorithms we have the classifier that uses the OpenCV DNN module, the CNN classifier, the SVM algorithm, among others. From the detections, we can see which algorithm provides the best detection (accurate detection of the objects + high confidence level of the detections) for a certain video. The detections are also important in order to later be compared with the content of the news.

The **D** element is the **News Semantic Detection**. This component is related to the extraction of the concepts of the news in the form of the wordcount. This element is another one of the most important parts of this model since it provides information to assess the video. The news article is processed and its wordcount is retrieved, being each one used individually or with the aggregation of others wordcounts from the news related to it.

The element identified as **E** is the **Matcher**. The Matcher retrieves the information from both the **Video Semantic Detection** and the **News Semantic Detection** components. The matching is done visually, which means that for each object detected, we see which term of the news relates to it the most. The object detected can match one or more terms. The results are normalized using a variation of the TF-IDF algorithm that followed from the work done by Tapaswi and Makarand [49].

The final element is the **Video Quality Assessment - F**. This element represents the assessment of the video given its semantic using the values retrieved and properly normalized from the previous phase. Considering that both the concepts detected in the video and the terms of the news that match with those objects are vectors, two distance metric are used to measure the distance between those two vectors. As an example of these distance metrics we have: **Cosine Similarity**, **Euclidean Distance** and **Manhattan distance**. From the results of the metrics used, we can come to a conclusion about the semantic quality of the video. Which means that, if both values of the metrics are close to it optimal value, then the video is assessed as a good quality video.

The element in dotted line from the model designed **Aesthetic Assessment**, refers to the assessment of images/videos previously done by Pedro Martins [27] in the thesis ”Sistema para Avaliação Semiautomática de Vídeo”, in which this masters thesis initially was based. The aesthetic assessment given by the tool developed by Pedro Martins is presented alongside with the semantic assessment provided by this.

The details from each element of the model are going to be explained and specified in the next chapter.

Chapter 4

Implementation

Following the model presented in the previous chapter, this chapter will explain throughly each one of the elements represented in the model and how the implementation of each one was achieved.

4.1 Video Semantic Detection

Like it was said in the Semantic Video Assessment chapter, this component corresponds to the detection of the objects in the videos by the different algorithms.

4.1.1 PySceneDetect

As it is well known not all frames have a significant relevance when comes to analysing a video and the videos are usually composed of different shots/scenes. Having that in mind a tool was used so that given a video, which scene could be individually retrieved.

The tool used was **PySeceneDetect**, an open source command line application and a Python library for detecting scene changes in videos, and automatically splitting the video into separate clips. It has available diverse detection methods. PySeceneDetect is written in python and requires the Numpy and OpenCV software libraries [35].

The methods used in this thesis were:

- **detect-content (-d content)** - This detection method compares each frame sequentially looking for changes in content, being useful in the detection of fast cuts between video scenes, although this method is slower to process. The break of the scene happens when the defined threshold is exceeded. The **threshold (-t)** by default is 30, but this value was changed for each video in order to better adjust to it.

- **output (-o)** - Specifying this command (-o output_file.mkv) will automatically split the input video, which will generate a new video clip for each detected scene in sequence, starting with output_file-001.mkv. Since each scene is now separated, the extraction of the keyframe for each scene is clearer.

- **save-images (-si)** - This command saves the first frame and last frame of each scene, before the cut.

4.1.2 Caffe Model Zoo

The implementation of this algorithm used the *opencv_dnn* module. This module is trained using GoogLeNet network from Caffe Model Zoo [24].

The implementation of the algorithm followed the tutorial provided by OpenCV. However, modifications were made in order to meet the need of this work. The main aspect of this implementation is related to the use of the network.

Three different implementations were done in order to chose the frames that were going to be classified.

The first implementation consisted in reading the video file and choosing one frame per second to be used in the classification. This was achieved using a condition that only if the number of the frame divided by 30 was zero, then this frame would be processed. The number 30 was chosen because is usually the number of frames per second of the videos (framerate of the videos).

The second one used the opencv property **CAP_PROP_FRAME_COUNT**, which counts the number of frames in the video file. Dividing the total number of frames by two, we were able to capture the main frame (keyframe). This keyframe was used for the detection to represent the whole scene. This implementation was done in order to reduce the processing of meaningless frames since, within a scene, there is almost no variation of the content. Therefore, processing the mainframe of the scene is the same as processing all frames of the same scene.

The last implementation came because there was a need to see if the results from the tests done using only the mainframe could be improved. Having that in mind, the frame count used for the last implementation was divided by four in order to divide the shot into three parts and then retrieve the frames from each division. This allowed us to retrieve more than one frame for the classification, namely three frames, and so, have more detection material.

Regarding the network, this one is initialized using the Caffe model file. The video is read and according to the implementation, either one frame per second or the mainframe or three frames of the shot is used for the processing part. The processing begins with the transformation of the image into a three-dimensional array with a 224x224x3 shape to later be converted to the four-dimensional blob with a 1x3x224x224 shape that is accepted by the GoogLeNet. The blob is then passed to the network and in the forward pass, the output of the ”prob” layer is computed. The best class is then determined by the output of the ”prob” layer, that contains the probabilities for each of 1000 ILSVRC2012 image classes. The index of the element with the maximum value is found and corresponds to the class of the image.

Each one of the captured frames were saved as an image file in order to be used on the others classifiers, that required not a video but an image. The results of the classification were also saved in a text file, so that it could be more practical to deal with it.

The 1000 ILSVRC2012 image classes can be seen in appendix A.

4.1.3 TensorFlow

This algorithm trains the classifier (MobileNet - CNN) with TensorFlow [1].

MobileNets are a new family of convolutional neural networks designed by Google which is small, fast and provide a good resource/accuracy trade-off.

Since the algorithm uses the code from the Google codelab [50], the code itself takes care of setting up and training the neural network. However in order to complete the training for the image classification, is necessary to run the scripts that were provided with certains parameters and also gather the images from the category that is meant for training.

The training parameters consisted of:

- Input image resolution: 224px. Although using a high-resolution image takes more processing time, the classification is more accurate;

- Architecture: mobilenet_0.50_%IMAGE_SIZE%. This represents the relative size of the model.

- Training steps: 4000 (default). This parameter was not specified in order to use the default (4000), considering that by training the longer, the accuracy of the classification can be enhanced.

The training script downloads the pre-trained model and adds a new final layer and is this final layer that is trained on the images that we provided. This means that the classifier was not trained from scratch, instead, it uses the transfer learning technique, meaning that some of the parameters MobileNet has learned are reused to create a new high accuracy classifier with a lot less training data, which result in a much faster learning time.

The classifier is just a function $f(x) = y$ being **f(x)** a two-dimensional array of pixels from the image and **y** being the label.

The model used was trained on the ImageNet Large Visual Recognition Challenge dataset since these models can differentiate between 1,000 different classes.

The script that gives the result was modified in order to only present the object detected if its confidence percentage is above a certain threshold (15% - 0.15)

[1] Open source machine learning framework especially useful for working with deep learning.

The classes trained with this classifier were: **Bycicle**, **Car**, **Daisy**, **Dog**, **Drum**, **Flags**, **Goal**, **Messi**, **Mic**, **Musician**, **Ronaldo**, **Scoreboard**, **Soccer field**, **SoccerBall**, **Stage**, **Street concert**, **Volcano**.

Images selection A python script were created in order to download the images that are used for the TensorFlow training.

The script uses the **icrawler** [57], a mini framework for web crawlers that is small and flexible. A keyword and a maximum number of iterations is provided in order to specify the amount of images we want along with its content.

After all images are saved, a manual process of selecting the images were made, so that the images could provide a good dataset and hence a good classifier.

The selection of the training classes were mostly based on the content found in the *Cognitus* videos, which majorly means they are related to concerts, street concerts and football.

4.1.4 Yolo

The YOLO system was directly used in this thesis, meaning that no alteration to the code was made.

A single neural network is applied to the full image. The network is then divided into regions and bounding boxes and probabilities for each region is predicted.

The network predicts four coordinates for each bounding box. The width and the height of the box are according to the offsets from the cluster centroids. The centre coordinates are predicted relative to the location of filter application using a sigmoid function.

The bounding boxes are weighted by the predicted probabilities.

The classes that the bounding box may contain is predicted for each box using multilabel classification, applying an independent logistic classifier. A binary cross-entropy loss is used for the class predictions during training time.

The system extracts features from three different scales using a similar concept to feature pyramids networks. From the base feature extractor, multiples convolutional layers are added (53 layers). The last of the layers predicts a three-dimensional tensor encoding bounding box, objectness and class predictions.

The complete image is looked at during test time so the global context of the image is taken into consideration for its predictions. The predictions are made using a single network evaluation.

This system is consistently being modified in order to improve the existent shortback [38, 61].

4.1.5 SVM

The implementation of the SVM algorithm used was developed in Python using some properties of the OpenCV and also the classification of the scikit-learn[2].

One image of the ”right” class was read with the **imread** method and this image were then transformed into a grey image (opencv enumerator - COLOR_BGR2GRAY). The descriptor and the keypoints of the image were retrieved using the SURF[3] algorithm.

One image of the ”wrong” class followed the same process and it descriptor was stored.

Two vectors were created, one with de descriptor for each one of the images and the other with the matching label of each of the descriptors. The labels represented either ”right” or ”not_right”. Later the estimator (**svm.SVC**) was defined and the **fit()** of the vectors was done. The estimator was set with the default parameters except for the kernel that it was set to ’linear’ instead of ’rbf’.

Then, the image in need to be classified went through the same process as the others images and using the **predict()** method of the estimator it was possible to classify the image given its computed descriptor.

This implementation contained several flaws apart from the fact that it could be biased since the images used as ”right” or ”wrong” were handpicked. This was made it evident from the results gathered from the first trial of tests.

Therefore another implementation was required in order to suppress these flaws. The second implementation consists of an implementation of the Bag-of-Features/Bag-of-Visual-Words.

The Bag-of-Features/Bag-of-Visual-Words Model (an adaptation of the Bag-of-Words Model) was used in here in order to classify the images, by treating image features as words.

The images used for the training were the same as the one used to feed the TensorFlow algorithm.

The images were retrieved using the **glob** and read using the opencv function **imread**.

The class **BOWKMeansTrainer**[4] is initialized with the size of the dictionary (defined according to the number of images classes).

The next step consists of converting each image, inside each set of images, to a grey image using the opencv enumerator **COLOR_BGR2GRAY**. Using the transformed image, its descriptor were computed using the SIFT algorithm. The descriptor is then added to the training set through the method **add** from the BOWTrainer Class.

After all the images from every set of images have been added to the training set, the train descriptors stored are clustered (method **cluster()**). The method returns the vocabulary, which means the cluster centers.

[2] Simple and efficient tools for data mining and data analysis [41]

[3] Speeded-Up Robust Features

[4] Kmeans-based class to train visual vocabulary using the bag of visual words approach [33].

The FLANN - Fast Library for Approximate Nearest Neighbors was used since it has optimized algorithms for high dimensional features. So first the parameters were set in order to create the objects with the specified parameters.

The following step consists of creating the bag-of-visual-words. For that, the class **BOWImgDescriptorExtractor** from OpenCV was used, since it normalizes the histogram of visual vocabulary words. The parameters given to create the object of the class was: the descriptor extractor (*SIFT*) and the descriptor matcher (*BFMatcher*). Then the visual vocabulary was set (method **setVocabulary()**) using the vocabulary previously computed.

Then a loop goes through each one of the images from a set of images. The image is transformed into a grey image, then it computes the image descriptor using the set of visual vocabulary using the BOWImgDescriptorExtractor class. The keypoints of the grey image was retrieved using the SIFT algorithm. Then the result (descriptors of the image) is stored in a list (train_desc) and a numerical label is appended to another list (train_labels), with that, each image is assigned to a label. That process is made for every set of images.

After the dictionary is created and the descriptor for each image is computed and has its label, the **svm.SVC** estimator does the fit of the data (train_des and train_label).

At last, the image that needs to be classified passes through the same process of being transformed into a grey image, computing the image descriptor using the set of visual words. Next, the estimator does a prediction given the computed descriptor (**predict()** method).

4.2 News Semantic Detection

This section will explain in further detail what the **D** component - News Semantic Detection - of the model 3.1 represents and how it was implemented.

The implementation for this part is divided into two in order to cover all the possible cases. The first implementation counts the words of the news as a whole. While for the second implementation, the wordcount is related to each one of the paragraph/sections of the news and stored separatedly.

Having the wordcount of the news articles we can have an idea of the content of the news. This information is then stored in order to latter assess the video.

4.2.1 News wordcount - whole

The wordcount for the news was achieved using a python script.

This script uses the python module **requests** in order to get the webpage given the url from the news article (method **get()**). An object of the class BeautifulSoup[5] is created using as parameter, the bytes of the response body of the request (method **content()**).

[5] Beautiful Soup is a Python library for pulling data out of HTML and XML files [3]

The words from the paragraphs are acquired through the **findAll()** BeautifulSoup method using **’p’** as parameters. Then using the **Counter** dict subclass for each word from each paragraph found, the words are counted.

The same was done for each *div* of the HTML and then both sets of words are added.

Later using the **most_common()** method it is possible to the know common elements and their counts from the most common to the least.

To reduce the words that are meaningless, the final set of words were filtered, which consisted in removing words that have less than two letters and more than fifteen letters and also replace the characters from the words (e.g. The word ”hello€” would be after the replacement ”hello”).

After the script run, a manual cleaning was done in order to eliminate words that were not meaningfull, reducing the set of words.

4.2.2 News wordcount - paragraph

This implementation also consisted of a python script. The news selected are separated into different files by sections/paragraphs. From each one of these files, the script starts by opening and reading the text file.

For each word read from the file, the same filtering process regarding the characters previously done in section 4.2.1 is done. Then if the word does not exist in the dictionary, the word is added. Otherwise, it increases the count.

The dictionary is used as a parameter to create a new object of the **Counter** subclass. Later, the **most_common()** method is used to ordain the words.

At last the file is closed.

Much like in the previous section 4.2.1, a manual cleaning was done reducing the set of words, eliminating the meaningless words.

4.3 Matcher

The **Matcher** element of the model uses the information previously gathered in the **Video Semantic Detection** phase and in the **News Semantic Detection** phase.

In order to properly use the gathered data, the results had to be normalized. The normalization comes from the study done by Tapaswi [49] since their study resembles our case given that much like them, we want to combine video and text. Therefore we used a variation of the Term Frequency-Inverse Document Frequency (Tf-Idf) algorithm (the idf part is not taken into consideration).

This algorithm gives a weight that is often used in information retrieval and text mining. The weight is a statistical measure used to evaluate how important a word is to a document in a collection [51].

The normalization is divided into **news normalization** and **video normalization**.

4.3.1 News Normalization

In order to normalize the values retrieved by the news, both the words and their respective counter were analyzed.

In the cases where the same term appeared in more than one article, the value of the counter of that term was divided by the number of articles in which the term was found. E.g. if the term **Goal** is present in five different article news and it appeared five times is each article the value of the counter of this term is $5 + 5 + 5 + 5 + 5 = 25/5 = \mathbf{5}$.

Following the algorithm (Tf-Idf), each counter was later normalized given the term frequency equation with a slight modification:

TF(t) = Counter of term t / Sum of the counters of every term

Applying this equation to every term, we normalized the terms of the news.

The results that came from the paragraph/section of the news, followed the same procedure, except that, the term did not have to be divided by the number of news in which the term appeared because it was considered only one article.

4.3.2 Video Normalization

The normalization of the video is similar to the normalization of the news. All the detection algorithms used provides which object it could detect in the scene but it also provides the confidence level in which the algorithm believes that the object in the scene is really the one the algorithm said it is.

Alike to what was done in the news normalization, for each object detected, the confidence percentage was summed up each time the object was detected throughout the whole video. Then this value was divided by the sum of the confidence level of each object detected considering that each one had the maximum value of confidence level ($100\% = 1$). E.g. if the TensorFlow algorithm detected the object **Stage** six times in the video one with 90% of confidence level each time and it detected fifteen objects in the video, then the normalized value of the **Stage** object for this algorithm would be $(0.90 * 6) = 5.4/15 = \mathbf{0.36}$

This process was done for every object detected for every video by all classifiers.

4.3.3 Match

The match between both the video and the news is done visually and separated by video and classifier. This means that considering a video, we take the normalized value of each object detected and try to match each one with one or more terms found in the news and then pair them up. The terms, and its respective normalized value, and the object detected that matches the terms, and its respective normalized value, is placed on a single table to later be assessed.

This process is repeated for each video and for each classifier.

4.4 Video Quality Assessment

The element **Video Quality Assessment** is the element responsible for assessing the video. In order to do that, two distance metric were used, namely **Cosine Similarity** and **Euclidean Distance**.

The Cosine Similarity metric was chosen because it is one of the most well-known measures used to calculate the similarity between different documents, used mostly for information retrieval and text mining.

The Euclidean Distance was used in order to measure more precisely how far away the vectors really are from each other, having, that way, a better notion of both vectors considering its weight and magnitude. Which is different from the other metric that only considers the angles between the vectors.

The figure 4.1 represents both of the metrics chosen in this thesis.

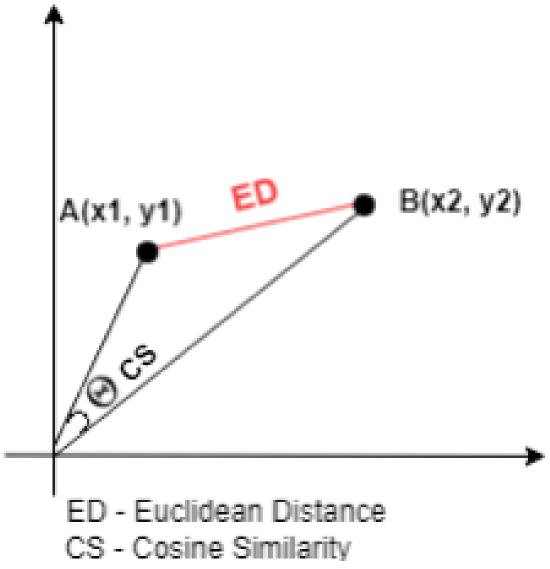


Figure 4.1: Cosine Similarity and Euclidean Distance representation

4.4.1 Cosine Similarity

- The Cosine Similarity[6] is used in order to compare the values from the news with the values from the object detected. The Cosine Similarity equation is as follows:

$$cos(\Theta) = \frac{\sum_{i=1}^{n} X_i * Y_i}{\sqrt{\sum_{i=1}^{n} X_i^2} * \sqrt{\sum_{i=1}^{n} Y_i^2}} \tag{4.1}$$

being **X** the vector representing the normalized values of the counter of the news and **Y** the vector representing the normalized values from the objects detected.

[6] The cosine similarity between two vectors (or two documents on the Vector Space) is a measure that calculates the cosine of the angle between them [25].

This process is done for each video and for every classifier.

The result given by the metric goes from -1 (completely opposite) to 1 (exactly the same), therefore it is possible to know the level of similarity between the news and the video and which classifier provides the best detection.

The representation of what the metrics represents can be seen in figure 4.2.

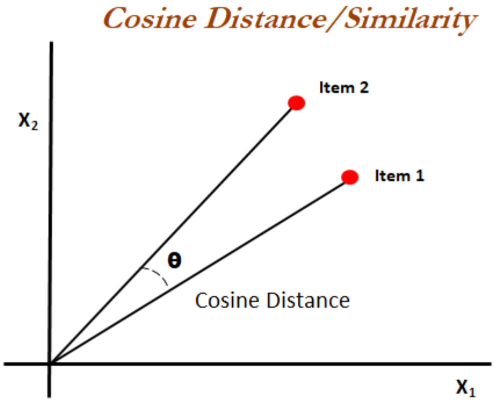


Figure 4.2: Cosine Similarity representation taken from [48].

4.4.2 Euclidian Distance

The Euclidean Distance is the function that assigns to any two vectors in Euclidean n-space a number and also giving the ”standard” distance between any two vectors in the Euclidean space [58].

Following the equation of this metric:

$$d(x,y) = d(y,x) = \sqrt{\sum_{i=1}^{n} (y_i - x_i)^2} \tag{4.2}$$

being **x** the term from the news and **y** the detected object, it is possible to know if both vectors are related to each other. The closer the result is from 0, the more the terms are related.

A distance was also generated for the correlation between each term with its correspondence from the object detected.

4.4.3 Assessment

Given the results obtained from both the metrics, we can assess the video as a good quality video or as a bad quality video. If for a given video, the value of the metrics are close to its optmal value, then the video is assessed as ”good”. But, if the contrary happens, the video is assessed as ”bad”.

4.5 Aesthetic Assessment

The **Aesthetic Assessment** represented in the model refers to the work done by Pedro Martins [27] in the thesis ”Sistema para Avaliação Semiautomática de Vídeo”. Using the tool developed by the colleague, it was possible to aestheticly assess the same videos used for the semantic assessment.

The videos were fed into the tool and the values of the assessment of those videos were retrieved. These values were stored and helped to complement the results from the semantic quality assessment.

4.6 Interface

A simple interface was created to show the videos and the objects that were detected from each one of the classifiers along with its confidence level. The assessment of the video (semantic and aesthetic wise) is also presented, meaning that the value of the metrics for each video will be shown.

The interface was created using HTML, XML and JavaScript. The visual part of the interface was done using Bootstrap elements such as modals, containers, hovers, and so on. The events created are obtained through the use of JavaScript.

The main point of the interface is to present the values from the classifier and that was achieved with the use of XML and JavaScript. For each video, four XML files were created with the objects detected and its confidence level. The values of the metrics are also in the XML file. The values presented in the interface are retrieved using JavaScript. So the page won't have to reload with each update, an XMLHttpRequest is created, then the request is initialized (method **open()**), and later the request is sent to the server (method **send()**). Using the property **responseXML()** it is possible to fetch the document that contains the XML and then retrieve the values that we need from it (object, confidence level and metrics values).

The value of the aesthetic assessment is provided by the tool developed by Pedro Martins [27] is also presented, being possible to know not only the semantic assessment of each video but also the aesthetic assessment of each video.

The design of the interface can be seen in image 4.3. Once a video is selected the details of the video will be shown, much like is represented in figure 4.4.

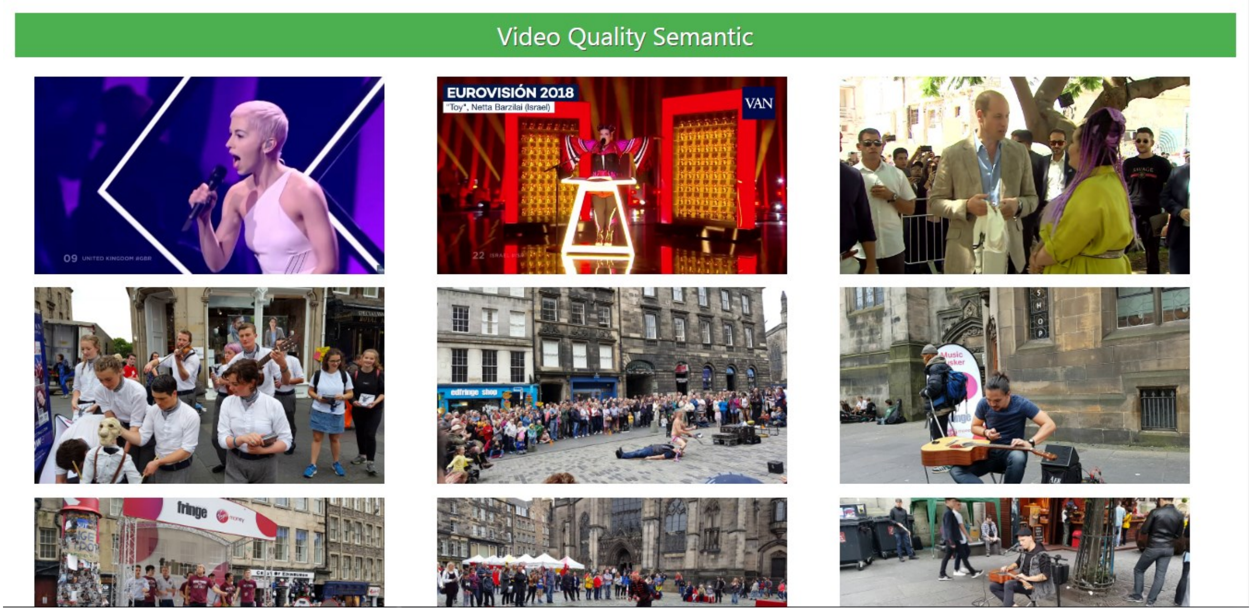
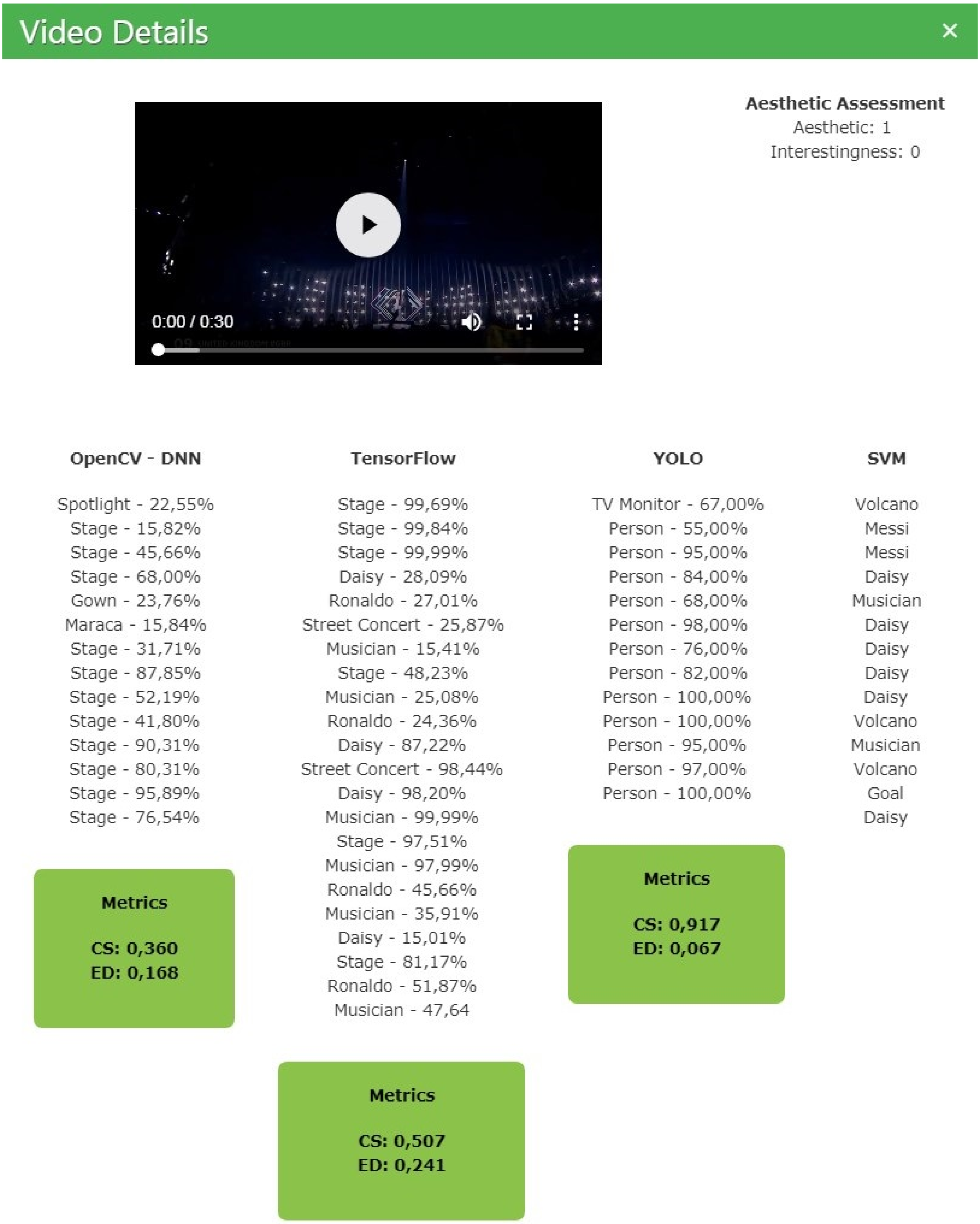


Figure 4.3: Interface Design



Video Details

Aesthetic Assessment
Aesthetic: 1
Interestingness: 0

OpenCV - DNN

Spotlight - 22,55%
Stage - 15,82%
Stage - 45,66%
Stage - 68,00%
Gown - 23,76%
Maraca - 15,84%
Stage - 31,71%
Stage - 87,85%
Stage - 52,19%
Stage - 41,80%
Stage - 90,31%
Stage - 80,31%
Stage - 95,89%
Stage - 76,54%

Metrics

CS: 0,360
ED: 0,168

TensorFlow

Stage - 99,69%
Stage - 99,84%
Stage - 99,99%
Daisy - 28,09%
Ronaldo - 27,01%
Street Concert - 25,87%
Musician - 15,41%
Stage - 48,23%
Musician - 25,08%
Ronaldo - 24,36%
Daisy - 87,22%
Street Concert - 98,44%
Daisy - 98,20%
Musician - 99,99%
Stage - 97,51%
Musician - 97,99%
Ronaldo - 45,66%
Musician - 35,91%
Daisy - 15,01%
Stage - 81,17%
Ronaldo - 51,87%
Musician - 47,64

Metrics

CS: 0,507
ED: 0,241

YOLO

TV Monitor - 67,00%
Person - 55,00%
Person - 95,00%
Person - 84,00%
Person - 68,00%
Person - 98,00%
Person - 76,00%
Person - 82,00%
Person - 100,00%
Person - 100,00%
Person - 95,00%
Person - 97,00%
Person - 100,00%

Metrics

CS: 0,917
ED: 0,067

SVM

Volcano
Messi
Messi
Daisy
Musician
Daisy
Daisy
Daisy
Daisy
Volcano
Musician
Volcano
Goal
Daisy

Figure 4.4: Video details design

Chapter 5

Evaluation

This chapter presents the evaluation of the proposed model to assess the quality of videos in terms of semantics. It describes the tests performed and discusses the results obtained.

The first part of the model is based on semantic detection on videos. Four methods to detect semantic concepts on images were tested. Following this, it was evaluated the semantic detection over the time. Finally, this chapter describes the evaluation of the algorithm to assess the semantic quality of videos.

The materials used for the tests subsists mostly on videos and articles news.

5.1 Videos

The video used are mainly related to the topic of the main focus of the Cognitus project (Edinburg festival - street concerts and football events). The main video sources were the Cognitus database (`https://cognitus-mobile.virt.ch.bbc.co.uk/`) and from Youtube. Different set of videos were used during the test phase. The common characteristic between the videos is that they are short duration videos of the original videos.

For the initial test (YOLO vs OpenCV (DNN)), a video about a match of a famous football player was used. Images of the video can be seen in 5.1.

The second test - Detection in Videos of Events, like it will be later explained, involves some popular news, therefore the videos are related to it, being about volcano eruptions, Eurovision contest and football. Three news were selected and two videos for each news was used. Some of the frames retrieved from these videos can be seen in 5.2.

For the following test (Semantic Detection with Scene Detection), the videos were retrieved from the Cognitus dataset. Seventeen videos were selected from the database mostly about street concerts but also about some football matches. Examples of frames can be seen in 5.3.

The videos used for the video assessment consisted of three videos, all of them related to the Eurovision contest. The first video is about the intrusion of a man during the stage of the UK's singer. The second one shows the performance of the winner of the contest. The last video is related to the meeting of the Eurovision contest winner with the prince of England at his visit to Israel. Some of the frames of these videos can be seen in 5.4.

Figure 5.1: Frames from the video used in the first test

Figure 5.2: Frames from the videos related to popular news

Figure 5.3: Frames from the videos retrieved from the Cognitus database

Figure 5.4: Frames from the videos used in the video quality assessment

5.2 News Articles

The news articles chosen are related to events, such as Eurovision, football and volcano eruptions (popular news). The news selected are mostly written in English, being some written in Portuguese. The main sources of the news were from online news websites such as https://edition.cnn.com/ (CNN), https://www.bbc.com/news (BBC) and https://www.independent.co.uk/ (Independent).

The news article used for the video assessment are exclusively related to the Eurovision contest. Ten news were chosen whose titles are: ”Prince William meets Eurovision winner Netta”; ”9 Eurovision moments to inspire Will Ferrel’s new Netflix comedy”; ”Israel complains over Dutch TV Eurovision parody”; ”Eurovision: Surie left ’bruised’ after stage invasion”; ”Stage stormed during UK’s Eurovision song”; ”Netta Barzilai of Israel wins Eurovision Song Contest”; ”Eurovision song contest could make you happier, study suggests”; ”Eurovision pulls plug on China after censorship of LGBT act”; ”Eurovision bosses explain how stage invader managed to past pass security”; ”Eurovision 2018: The real romance behind Spain’s entry by Alfred and Amaia”

5.3 Assessment

The main focus of this thesis work is related to the assessment of the video quality given its semantic. The assessment was done considering the level of correlation between the data retrieved from the news articles (wordcount) and the data regarding the detection of the objects by the different classifiers. In order to achieve that, two distance metric were used - Cosine Similarity and Euclidean Distance. Considering that the metric chosen gives how close two vectors are, the data retrieved from the news articles and the detections need to be transformed into vectors for the metrics to make sense. The assessment of the video consisted in:

- **Good Quality Video** if the value of the metrics are close to its optimal value. Being this value *zero* for the Euclidean Distance and *one* for the Cosine Similarity.

- **Bad Quality Video** if the values of the metrics are far from its optimal value.

5.4 Semantic Detection Techniques

Having in mind that the detection of the objects is the core of the assessment of the videos, understanding how the algorithms behave and what are its strong and weak points is essential. The hit rate for each classifier used in each test was retrieved in order to be able to properly evaluate them.

5.4.1 YOLO vs OpenCV (DNN)

After implementing the DNN algorithm, a test was done to compare the objects detected by the DNN algorithm and by the YOLO classifier.

A video from a famous football player (Lionel Messi) was selected and using the implementation of the algorithm that consisted in classifying one frame per second (explained in section 4.1.2), both the frames and its classification were stored. The same images were given to the YOLO classifier so that the algorithm could also classify them.

Some frames used for detection can be seen in 5.5, while the results from the OpenCV (DNN) detection can be seen in 5.1.

Figure 5.5: YOLO vs DNN - Frames examples

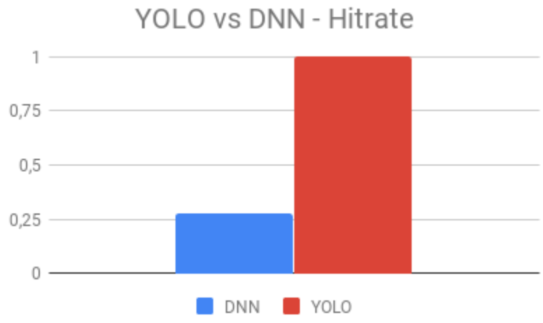


Figure 5.6: YOLO vs DNN - Hit rate

For the same images, the results given by YOLO can be seen in 5.2. The figure 5.6 shows the hit rate for each classifier (DNN and YOLO).

As it is possible to see, the classifier that detects the objects with more precision is the YOLO classifier, detecting the object *person* with high precision. However, there is not

Frames:	Name:	Probability:
1	ping-pong ball	17.01%
2	racket	6.80%
3	scoreboard	10.07%
4	ballplayer, baseball player	23.04%
5	prison, prison house	57.51%
6	capuchin, ringtail, Cebus capucinus	36.99%
7	theater curtain, theatre curtain	7.57%
8	racer, race car, racing car	32.87%
9	ballplayer, baseball player	34.27%
10	scoreboard	29.50%
11	scoreboard	28.24%
12	ping-pong ball	74.58%
13	balance beam, beam	26.42%
14	torch	7.98%
15	mortarboard	18.75%
16	torch	13.58%
17	neck brace	18.02%
18	comic book	18.16%

Table 5.1: DNN algorithm detection

Frames:	Name:	Probability:
1	person	100.00%
2	person	92.00%
3	person	100.00%
4	person	100.00%
5	person	100.00%
6	person	100.00%
7	person	89.00%
8	person / tv monitor	93.00% / 70.00%
9	tv monitor / person	85.00% / 95.00%
10	tv monitor / person	76.00 / 97.00%
11	person	99.00%
12	person	100.00%
13	person	100.00%
14	person	99.00%
15	person	99.00%
16	person	90.00%
17	person	74.00%
18	person	100.00%

Table 5.2: YOLO detection

much of variation of the detected objects (only two objects detected).

While the YOLO classifier detects with high precision, the DNN algorithm from OpenCV, detects with a wide range of precision, going from 7% to 75%. The spectrum of objects detected is also large and despite having some detections that at first sight are not directly related to the image, it also detects some objects that can be seen in the images such as *scoreboard* and *ballplayer*.

Regarding the hit rate of the classifers, we can see that there is a substancial difference between the number of hits of the YOLO classifier and the DNN classifier, and using this measure we can say that the YOLO classifer was better in detecting the right objects than the DNN classifier.

In conclusion, both classifiers have its flaws and margin error, but the YOLO classifier could be seen as a better classifier, especially if the video/images contains people.

5.4.2 Detection in Videos of Events

The CNN classifier was introduced in order to provide another source of comparison concerning the objects detected. The classes trained by this classifier were at this point only a few and the training data were also not thoroughly polished which could induce some errors in the classification. Therefore, this classifier and the other two classifiers (DNN and YOLO) were evaluated in a context of detection of semantic concepts in event videos (e.g., Eurovision Contest). This test consisted of:

- selecting an news article and retrieving its wordcount. Three different events were selected having, each one, two articles. Being those events: Eurovision contest, a volcanic eruption in Hawaii and current news in football;

- choosing a video that correlates the most with the news;

- detecting the objects and its confidence level using DNN and also retrieving the images (one per second);

- detecting the objects and its confidence level using YOLO;

- detecting the objects and its confidence level using the CNN classifier.

The wordcount of the news was only used as a visual comparison, which means that given the results obtained from the classifiers, a visual comparison was made to see if the video/objects detected could really correlate to what it is said in an news article.

The result of the wordcount from one of the news concerning the volcanic eruption in Hawaii event can be seen in table 5.3.

Some of the frames used for the classification of the video (volcanic eruption) can be seen in figure 5.7.

The results from this test, regarding the news from the volcanic eruption, can be seen in figure 5.8. The hit rate of each classsifer used is displayed in figure 5.9.

Word:	Worcount:
lava	4
homes	3
volcano	3
island	2
kilauea	2
eruption	2

Table 5.3: Wordcount from news article

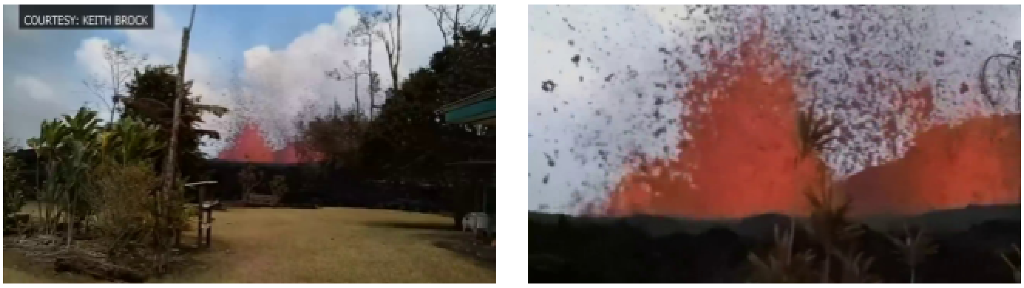
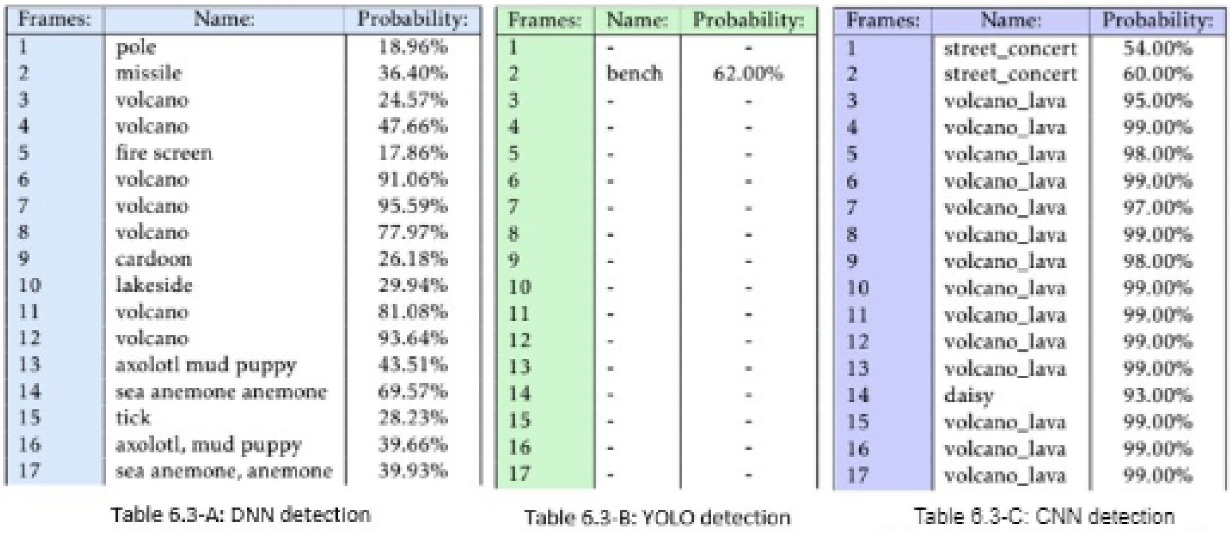


Figure 5.7: Detection in Videos of Events - Frames examples



Frames:	Name:	Probability:
1	pole	18.96%
2	missile	36.40%
3	volcano	24.57%
4	volcano	47.66%
5	fire screen	17.86%
6	volcano	91.06%
7	volcano	95.59%
8	volcano	77.97%
9	cardoon	26.18%
10	lakeside	29.94%
11	volcano	81.08%
12	volcano	93.64%
13	axolotl mud puppy	43.51%
14	sea anemone anemone	69.57%
15	tick	28.23%
16	axolotl, mud puppy	39.66%
17	sea anemone, anemone	39.93%

Table 6.3-A: DNN detection

Frames:	Name:	Probability:
1	-	-
2	bench	62.00%
3	-	-
4	-	-
5	-	-
6	-	-
7	-	-
8	-	-
9	-	-
10	-	-
11	-	-
12	-	-
13	-	-
14	-	-
15	-	-
16	-	-
17	-	-

Table 6.3-B: YOLO detection

Frames:	Name:	Probability:
1	street_concert	54.00%
2	street_concert	60.00%
3	volcano_lava	95.00%
4	volcano_lava	99.00%
5	volcano_lava	98.00%
6	volcano_lava	99.00%
7	volcano_lava	97.00%
8	volcano_lava	99.00%
9	volcano_lava	98.00%
10	volcano_lava	99.00%
11	volcano_lava	99.00%
12	volcano_lava	99.00%
13	volcano_lava	99.00%
14	daisy	93.00%
15	volcano_lava	99.00%
16	volcano_lava	99.00%
17	volcano_lava	99.00%

Table 6.3-C: CNN detection

Figure 5.8: Results from test - Detection in Videos of Events

From the results, we can see that the YOLO classifier could not classify properly the video. The DNN classifier had an average performance given that out of 17 frames, it could detect in 7 of them the object *volcano* and in one of them the object *fire sceen* which is in some level correct. And the confidence level from these detections was mostly high, having an average of 78.08%. The CNN classifier had a high performance. It could detect the object *volcano_lava* for almost every frame (14 out of 17) with a high confidence level (average of 98.43%).

Comparing with the wordcount table, we can see that indeed the words with the highest counters, had the objects corresponding to it, detected by the classifiers. From the

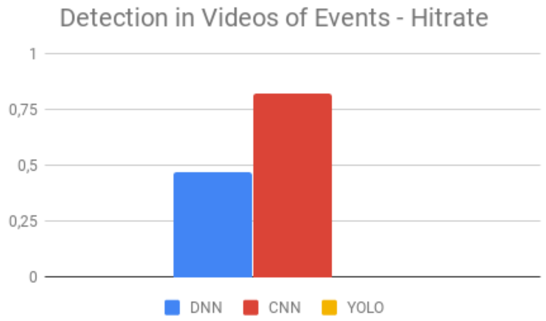


Figure 5.9: Detection in Videos of Events - Hit rate

graph of the hit rate we can see that the CNN classifier had the most successful detection of the right objects in the scene in comparison to the others classifiers. This test showed that it was possible to compare the news articles with the video at the content level.

The results from the others events (Eurovision Contest and Football) can be found in Appendix B.

Similarly to the results from the volcanic eruption, both the CNN and the DNN classifier performed better than the YOLO classifier considering the variety of the detected objects for the remain videos. Although it had detected *person* with a high confidence, it was the only concept detected apart from the fact that the concept is too general.

The DNN classifier could, for the football video, predict *ballplayer* and *scoreboard* and also some other kind of balls although not the right kind. The confidence level for the *ballplayer* was above average. The same classifier had a better performance on the next event (Eurovision contest) detecting with a high confidence the concepts *stage*, *microphone* and *spotlight*.

The CNN classifier for the football related event could detect *soccer field*, *goal*, *scoreboard*, *ronaldo* and *messi* with a high confidence. Although it could recognize the football players in some frames, most of the detection between those two players were not correct. However, the video was rightfully represented by its detected objects, which means that even if a person that did not see the video, it could tell what the video was about. Regarding the Eurovision video, the concepts detected were *stage* with an average confidence level around 83% and also *street concert*, that even though it does not properly match with what it is presented in the video, it has some sort of similarity to it.

However, when the focus is only on the hit rate measure, the classifier that gave the best results was the YOLO classifier. The second best alternates between the other two classifers, sometimes being the DNN and other times being the CNN.

The interpretation of the results can be seen in two different lights. We can say that the best classifier is the one that have the highest hit rate, despite identifying only a fairly general concept. Or the best classifier is the one that provides an average hit rate but it detects more objects correctly in which these objects are relevant to the content of the video.

5.4.3 Semantic Detection with Scene Detection

Another classifier was introduced in order to have another source of comparison and also another way of detecting concepts from the videos. The classifier used in this phase consisted in its first implementation (4.1.5), which consisted in a SVM trained with a set of positive and negatives images. The results from this implementation only displayed if the image belongs to a certain class or not, which means that an image could only be classified as *name of the class* (e.g. street_concert) or *not_name of the class* (e.g. not_street_concert).

The PySceneDetect was used in this test. The main goal in using this tool to separate the scenes of the videos was to reduce the number of processed frames per scene, given that the objects in it were the same.

This set of tests used the videos from the Cognitus dataset. The dataset contained a huge amount of videos, however, there were a lot of irrelevant videos and so a selection was made in order to choose the videos that had more relevance. In order to test the tool, videos with different scenes were also chosen. This dataset contains in its majority, videos from the Edinburgh festival and so they were mostly related to street concerts. There were also two football-related videos and one of a stage performance.

The tables 5.4, 5.5, 5.6, 5.7 presents the results for one of the videos out of the ones that were classified and some of the frames from this video is displayed in 5.10.

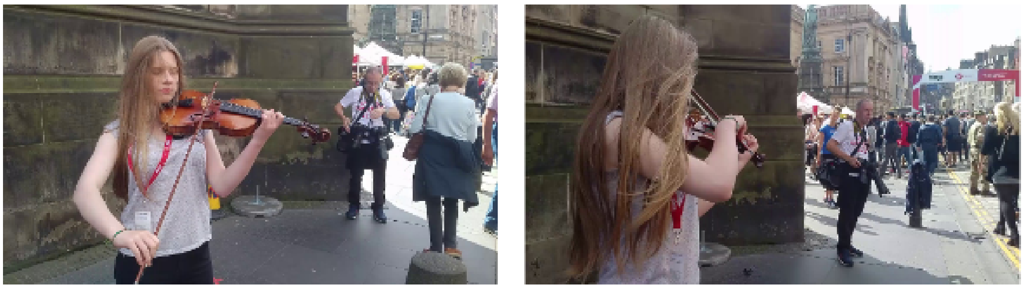


Figure 5.10: Semantic Detection with Scene Detection - Frames examples

The graph representing the hit rate of the classifiers for this test can be seen in figure 5.11.

From the results of the tables we can see that:

- The DNN algorithm could detect rightfully the instrument present in the video - *violin*, although the confidence level of the detection is small. The object was only detected twice, which shows that the detection is lacking.

- Once again, the YOLO classifier could identify *person* with a high confidence level but in this test, it could also identify some of the others objects that were present in the scene, such as *handbag* and some others. Despite of the detection being accurate, it is not possible to understand what is the content of the video through its detection.

Frames:	Name:	Probability:
1	flute	31.485143%
2	violin	29.569679%
3	rifle	48.199138%
4	rifle	32.236749%
5	wig	21.606749%
6	wig	14.265956%
7	bubble	39.387673%
8	wig	21.245395%
9	trombone	43.178338%
10	violin	29.831415%
11	rifle	39.946914%
12	bubble	17.513129%
13	bubble	18.610345%
14	bubble	28.058845%

Table 5.4: DNN detection

Frames:	Name:
1	musician
2	musician
3	musician
4	musician
5	musician
6	musician
7	musician
8	musician
9	musician
10	musician
11	musician
12	musician
13	musician
14	musician

Table 5.5: SVM detection

Frames:	Name:	Probability:
1	musician	99.00%
2	musician	87.00%
3	musician	97.00%
4	musician / street concert	68.00% / 32.00%
5	street concert / musician	58.00% / 40.00%
6	musician	92.00%
7	street concert	95.00%
8	musician	89.00%
9	street concert	99.00%
10	street concert	99.00%
11	street concert / musician	81.00% / 18.00%
12	street concert	99.00%
13	street concert/musician	84/16
14	street concert	99.00%

Table 5.6: CNN detection

- The SVM could correctly classify the images, however, these results could not be considered as valid since each image was hand-picked (the one for the right class and the one for the wrong class) and so the classification could be considered *biased*. The performance of the SVM in this test helped to enlight that the initial implementation was not meeting its purpose.

- The CNN algorithm could detect *street concert*, *musician* in this video. The confidence level of the detection was also relatively high. With its detection we can infer the content of the video.

From the hit rate of the classifiers, we can see that it is not possible to diferentiate between the CNN, the YOLO and the SVM classifiers. We can only conclude that between

Frames:	Name:	Probability:
1	person / nadbag	100.00 / 89.00%
2	person / nadbag	100.00% / 53.00%
3	person	100.00%
4	person / handbag	99.00% / 54.00%
5	person / umbrella / train	99.00% / 67.00% / 74.00%
6	person / umbrella	100.00% / 50.00%
7	person / handbag / umbrella	100.00% / 51.00% / 63.00%
8	person / cellphone / handbag / backpack	100.00% / 62.00% / 52.00% / 90.00%
9	person / handbag / backpack	100.00% / 53.00% / 94.00%
10	person	100.00%
11	person	100.00%
12	person	100.00%
13	person	100.00%
14	person / handbag	100.00% / 60.00%

Table 5.7: YOLO detection

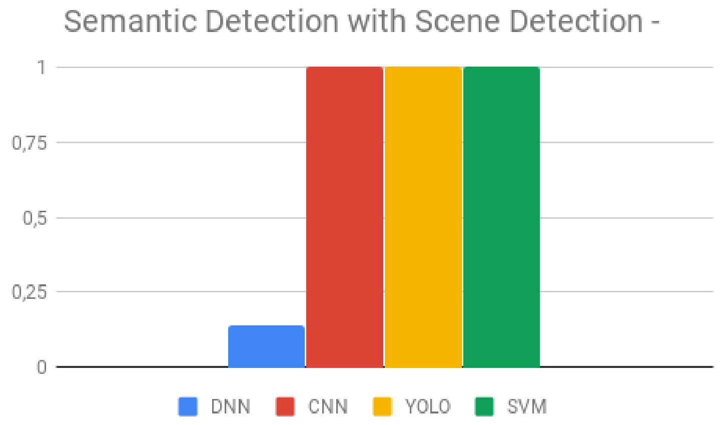


Figure 5.11: Semantic Detection with Scene Detection - Hit rate

all four classifiers, the one with the worst hit rate is the DNN classifier. This means that out of the considered frames, this classifier was the one to have the highest amount of objects mistakenly detected.

Given both the tables and the graphs presented, we can say that the CNN, besides having a high accuracy in the detection, it can also properly represent the videos given its detected objects.

Through the detection of the concepts from all the classifier it was not clear if the detection of the scene did actually make a difference in the detection or not or if the detection of the scene was correct, nevertheless, a manual check was done in order to secure that the scenes were cut accordingly.

5.4.4 Temporal semantic

One of the main focus of this thesis is to evaluate video quality in terms of semantic. Therefore, it is important to analyse the evolution of the semantic over the entire video.

The method that was used, consisted in how the distribution of a certain object follows during the length of the video using a graphic. Since the detection of the concepts is done throughout the video in an ordered manner, it is possible to figure out the timeline of the video using the detections. The detection error also played an important factor since the result deviated from the reality.

This test used the detection given by the DNN algorithm of a video from the collection of videos retrieved from the Cognitus dataset. Out of the detection results, a graph was generated. The graph from the temporal semantic can be found in the figure 5.12.

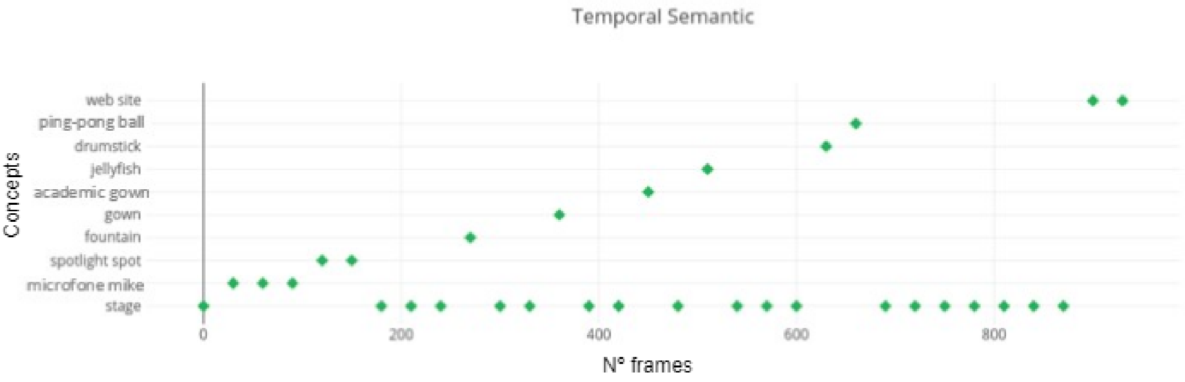


Figure 5.12: Temporal semantic

From the results, it is possible to see that the concept **stage** was the most detected and it also made it clear at what time/frame the object was present. Comparing it with the video, we could see that the detection was almost accurate, matching the detection of the concept with a **stage** on the video. Having that in mind we could, through the detections, create a timeline of the video, at least regarding the object **stage**.

5.5 Video Assessment

The test for the assessment of the video quality followed the subsequent steps:

1. **News selection and wordcount** - Ten news articles were selected and the wordcount of each one of them were extract. The wordcount of all the articles were aggregated, adding to the counter of the words found. Like it was explained in the 4.2.1, a cleaning was also done in order to remove the words that were insignificant, which means that the words that reveal emotions, are adjectives, and so on, were removed, remaing only the words that represents objects or are nouns.

2. **News wordcount normalization** - Like it was explained in the section 4.3.1, the values of the wordcount of the news went through a normalization process so that the data could be compared to the data of the detection given by the object detection algorithms. The normalization process, as it was explained before, followed the TF ideology. Given this fact, the counter for each word was divided by the number of news that term appeared in. The same is applied to every word and then added to the value of the previous term in order to produce a final value that is the sum of every counter of all terms. To normalize the values, each term was divided by the value of the sum and each one of these values became one position of the array. For example, the term *eurovision* appeared in every article *(10)* and the total number of times it appeared in the sum of all news articles was *941*. So we did the operation *941/10 = 94.10*. In order to normalize the value, we used the given result previously obtained *(94.10)* and divided by the sum of frequencies of the terms, which was *(2858.20)*. The result of the normalization is *94.10 / 2858.20 = 0.03292281856*. This value *(0.03292281856)* was the number inserted into the array for the position related to the term *eurovision*.

3. **Objects Detection** - In this step we used the videos in order to extract the objects detected in it. Much like it was said in the 4.1 section, the videos were first cut into scenes by the **PySceneDetect** tool. The videos that resulted from this cut were fed into the DNN algorithm. In this phase, the code used was the one that only extracted the middle frame of the videos. The frames used for classification were saved along with the results of the detection. These frames were fed into the others classifiers and the classification of the videos and its confidence levels were displayed in an excel file along with the terms and its normalization.

4. **Classification normalization** - In order to compare both the detections's data and the news's data, the values of the detection was normalized. The normalization followed what was described in the section 4.3.2. For the same concept, the normalized value is the sum of every confidence level of this concept found in the detection of the video and divided by the sum of the maximum value of the confidence level (1) of every concept found by this classifier in the video. In case of different concepts detected correlates to the same terms, the values of the confidence level of the concepts are aggregated. The same is applied to every concept detected throughout the video and for each classifier.

5. **Metrics** - In order to compare the videos with the news, a table was made for every video for every classifier. An example of the tables can be seen in 5.8.

   The table has the columns: *words*, *w_values*, *detection*, *c_values*, *sqr words values*, *sqr detection values*, *cosine simiarity*, *euclidean distance*. The column *words* corresponds to the terms found and considered in the news articles. The *w_values* column corresponds to the normalized values of the terms that were calculated in the second

words	w_values	detection	c_values	sqr of words values	sqr detection values	cosine similarity	euclidean distance
stage	0,0146	stage	0,002	0,0146	0,002	**1**	**0,0126**

Table 5.8: Metrics table example

step - *News wordcount normalization*. The objects detected that matches the terms of the news are displayed in the column *detection*, but it also includes the caracter "-" when there is no relation between the objects detected and the news. The *c_values* column has the values that were calculated in the fourth step (*Classification normalization*), along with the value **0** whenever there is no correlation between the arrays. Following the calculus equation for the cosine similarity the square roots were calculated in advance, being these represented in the colums *sqr of words values* and *sqr detection values*. The calculus for this metric uses the values from the others columns. So the value presented in the column *cosine similarity* comes from adding the multiplication of the both *values* columns for every concept (i.e. every row). Then dividing by the multiplication of the both *square root* colums. The equation can be translated as:

$$\frac{\sum_{i=1}^{n} w\_values(i) * c\_values(i)}{sqr\_of\_words\_values * sqr\_detection\_values} \tag{5.1}$$

The *euclidean distance* is calculated using the columns *w_values*, *c_values*. The distance is calculated for each concept $\sqrt{(w\_values - c\_values)^2}$ so that we can see the difference between the term and the concept but is also calculated between both arrays in order to see if they match overall. The equation for the whole array is the square root of the sum of the distance of all concepts - $\sqrt{\sum_{i=1}^{n}(w\_values(i) - c\_values(i))^2}$.

The results from the DNN classifier for the videos can be seen in the table 5.9, 5.10, 5.11. Complementary information can be found in C.1 and in C.2.

From this tables we can see that the value of the Cosine Similarity of the first video is 0.4448. The value of this metric for the second video is approximately 0.4438 while the value for the third one is 0.1904. As it was said before, the closer the value of the Cosine Similarity is to 1, more the vectors are similar to each other. Having that in mind, the best video to represent acordding to this metric and this classifier is the first video. The result given by the Euclidean Distance metric differs from the previous metric, since the value of the distance for the second video is closer to zero (0.254) than the remaining videos (1.016 and 0.3248). And unlike the previous metric, the closer to zero the value of the distance is, the closer to the optimal value the result is.

Tables 5.12, 5.13, 5.14 displays the result from the CNN classifier. Complementary information can be found in C.3 and in C.4.

Deriving out of tables we can see that for the Cosine Similarity metric, the first video was the best one out of the ones considered, since the result of this metric for this video is

words	w_values	detection	c_values	sqr of words values	sqr detection values	cosine similarity	euclidean distance
performance	0,167937863	stage	0,49010267	0,2241531654	1,096034398	**0,4447866847**	**0,322164807**
stage	0,01460709537	stage	0,49010267				**0,4754955746**
music	0,009796375341	stage	0,49010267				**0,4803062947**
microphone	0,01434469246	stage	0,49010267				**0,4757579775**
show	0,01574417466	stage	0,49010267				**0,4743584953**
dress	0,01539430411	gown	0,01697838286				**0,00158407875**
song	0,05709887342	-	0				**0,05709887342**
photos	0,06297669862	-	0				**0,06297669862**
ferrell	0,04618291232	-	0				**0,04618291232**
eurovision	0,03292281856	-	0				**0,03292281856**
she	0,03148834931	-	0				**0,03148834931**
film	0,02728990274	-	0				**0,02728990274**
singer	0,02662514869	-	0				**0,02662514869**
contest	0,02632775873	-	0				**0,02632775873**
parody	0,02554055	-	0				**0,02554055**
gaza	0,02449093835	-	0				**0,02449093835**
prince	0,02344132671	-	0				**0,02344132671**
report	0,02029249178	-	0				**0,02029249178**
star	0,01994262123	-	0				**0,01994262123**
video	0,01952277657	-	0				**0,01952277657**
netta	0,01877638607	-	0				**0,01877638607**
israel	0,01830989201	-	0				**0,01830989201**
surie	0,01644391575	-	0				**0,01644391575**
message	0,01574417466	-	0				**0,01574417466**
israeli	0,01574417466	-	0				**0,01574417466**
ryan	0,01574417466	-	0				**0,01574417466**
media	0,01539430411	-	0				**0,01539430411**
crowd	0,01539430411	-	0				**0,01539430411**
countries	0,01521936883	-	0				**0,01521936883**
man	0,01504443356	-	0				**0,01504443356**
palestinian	0,01504443356	-	0				**0,01504443356**
visit	0,01504443356	-	0				**0,01504443356**
flag	0,01434469246	-	0				**0,01434469246**
band	0,01399482192	-	0				**0,01399482192**
protester	0,01399482192	-	0				**0,01399482192**
home	0,01364495137	-	0				**0,01364495137**
violence	0,01224546918	-	0				**0,01224546918**
world	0,01049611644	-	0				**0,01049611644**
barzilai	0,009796375341	-	0				**0,009796375341**
chinese	0,008047022602	-	0				**0,008047022602**
netherlands	0,006997410958	-	0				**0,006997410958**
secutity	0,006297669862	-	0				**0,006297669862**
europe	0,005597928766	-	0				**0,005597928766**
lisbon	0,005422993492	-	0				**0,005422993492**
albania	0,005248058218	-	0				**0,005248058218**
						total:	**1,016245247**

Table 5.9: Video 1 - DNN classification

cosine similarity	euclidean distance
0,4437760658	0,2540671298

Table 5.10: Video 2 - DNN classification

cosine similarity	euclidean distance
0,1904242877	0,324803232

Table 5.11: Video 3 - DNN classification

closer to the optimal value than the rest of the videos. Considering the Euclidean Distance metric, the chosen video is the second video (smaller value out of the ones calculated).

The results from the classification by the YOLO classifier for the videos can be seen in the tables 5.15, 5.16, 5.17. Complementary information can be found in C.5 and in C.6.

Taking into account only the Cosine Similarity metric, it is not possible to select a single video, given that the values of this metric for the three videos were the same (0.3204). Regarding the Euclidean Distance metric, the third video was for once considered the best video given that it has the lowest value (1.7764).

words	w_values	detection	c_values	sqr of words values	sqr detection values	cosine similarity	euclidean distance
performance	0,167937863	stage/street_concert	0,2958066973	0,2241531654	0,7116023404	**0,4756386144**	**0,1278688343**
stage	0,01460709537	stage/street_concert	0,2958066973				**0,2811996019**
music	0,009796375341	stage/street_concert	0,2958066973				**0,2860103219**
microphone	0,01434469246	stage/street_concert	0,2958066973				**0,2814620048**
show	0,01574417466	stage/street_concert	0,2958066973				**0,2800625226**
singer	0,02662514869	musician	0,1463869923				**0,1197618436**
netta	0,01877638607	musician	0,1463869923				**0,1276106062**
surie	0,01644391575	musician	0,1463869923				**0,1299430765**
man	0,01504443356	ronaldo	0,06769361591				**0,05264918235**
song	0,05709887342	-	0				**0,05709887342**
photos	0,06297669862	-	0				**0,06297669862**
ferrell	0,04618291232	-	0				**0,04618291232**
eurovision	0,03292281856	-	0				**0,03292281856**
she	0,03148834931	-	0				**0,03148834931**
film	0,02728990274	-	0				**0,02728990274**
contest	0,02632775873	-	0				**0,02632775873**
parody	0,02554055	-	0				**0,02554055**
gaza	0,02449093835	-	0				**0,02449093835**
prince	0,02344132671	-	0				**0,02344132671**
report	0,02029249178	-	0				**0,02029249178**
star	0,01994262123		0				**0,01994262123**
video	0,01952277657	-	0				**0,01952277657**
israel	0,01830989201		0				**0,01830989201**
message	0,01574417466	-	0				**0,01574417466**
israeli	0,01574417466	-	0				**0,01574417466**
ryan	0,01574417466	-	0				**0,01574417466**
media	0,01539430411	-	0				**0,01539430411**
crowd	0,01539430411	-	0				**0,01539430411**
dress	0,01539430411	-	0				**0,01539430411**
countries	0,01521936883		0				**0,01521936883**
palestinian	0,01504443356	-	0				**0,01504443356**
visit	0,01504443356	-	0				**0,01504443356**
flag	0,01434469246	-	0				**0,01434469246**
band	0,01399482192	-	0				**0,01399482192**
protester	0,01399482192	-	0				**0,01399482192**
home	0,01364495137	-	0				**0,01364495137**
violence	0,01224546918	-	0				**0,01224546918**
world	0,01049611644	-	0				**0,01049611644**
barzilai	0,009796375341	-	0				**0,009796375341**
chinese	0,008047022602	-	0				**0,008047022602**
netherlands	0,006997410958	-	0				**0,006997410958**
secutity	0,006297669862	-	0				**0,006297669862**
europe	0,005597928766	-	0				**0,005597928766**
lisbon	0,005422993492	-	0				**0,005422993492**
albania	0,005248058218	-	0				**0,005248058218**
						total:	**0,6361205063**

Table 5.12: Video 1 - CNN classification

cosine similarity	euclidean distance
0,4728443308	0,6305818987

Table 5.13: Video 2 - CNN classification

cosine similarity	euclidean distance
0,465843023	1,275460046

Table 5.14: Video 3 - CNN classification

Results analysis Having in mind that this test intended to assess the videos using the news article related to it and given the results obtained from both metrics but also from each classifier, it is not possible to say that a single video has the best quality given its semantic since the results of the metrics points at disticts directions. However if we consider only the DNN and the CNN classifiers, we can say that for the Cosine Similarity metric the best video is the first video, while for the Euclidean Distance the selected one was the second video.

Analysing the results it is possible to see that the algorithm that gave the best value for the Cosine Similarity metric was the CNN in the first video. The variance of the this

words	w_values	detection	c_values	sqr of words values	sqr detection values	cosine similarity	euclidean distance
performance	0,167937863	-	0	0,2241531654	2,678812331	**0,3203793007**	**0,167937863**
stage	0,01460709537	-	0				**0,01460709537**
music	0,009796375341	-	0				**0,009796375341**
microphone	0,01434469246	-	0				**0,01434469246**
show	0,01574417466	-	0				**0,01574417466**
singer	0,02662514869	person	0,8076923077				**0,781067159**
netta	0,01877638607	person	0,8076923077				**0,7889159216**
surie	0,01644391575	person	0,8076923077				**0,7912483919**
man	0,01504443356	person	0,8076923077				**0,7926478741**
song	0,05709887342	-	0				**0,05709887342**
photos	0,06297669862	-	0				**0,06297669862**
ferrell	0,04618291232	person	0,8076923077				**0,7615093954**
eurovision	0,03292281856	-	0				**0,03292281856**
she	0,03148834931	person	0,8076923077				**0,7762039584**
film	0,02728990274	-	0				**0,02728990274**
contest	0,02632775873	-	0				**0,02632775873**
parody	0,02554055	-	0				**0,02554055**
gaza	0,02449093835	-	0				**0,02449093835**
prince	0,02344132671	person	0,8076923077				**0,784250981**
report	0,02029249178	-	0				**0,02029249178**
star	0,01994262123		0				**0,01994262123**
video	0,01952277657	-	0				**0,01952277657**
israel	0,01830989201		0				**0,01830989201**
message	0,01574417466	-	0				**0,01574417466**
israeli	0,01574417466	-	0				**0,01574417466**
ryan	0,01574417466	person	0,8076923077				**0,791948133**
media	0,01539430411	-	0				**0,01539430411**
crowd	0,01539430411	person	0,8076923077				**0,7922980036**
dress	0,01539430411	-	0				**0,01539430411**
countries	0,01521936883	-	0				**0,01521936883**
palestinian	0,01504443356	person	0,8076923077				**0,7926478741**
visit	0,01504443356	-	0				**0,01504443356**
flag	0,01434469246	-	0				**0,01434469246**
band	0,01399482192	-	0				**0,01399482192**
protester	0,01399482192	person	0,8076923077				**0,7936974858**
home	0,01364495137	-	0				**0,01364495137**
violence	0,01224546918	-	0				**0,01224546918**
world	0,01049611644	-	0				**0,01049611644**
barzilai	0,009796375341	-	0				**0,009796375341**
chinese	0,008047022602	-	0				**0,008047022602**
netherlands	0,006997410958	-	0				**0,006997410958**
secutity	0,006297669862	-	0				**0,006297669862**
europe	0,005597928766	-	0				**0,005597928766**
lisbon	0,005422993492	-	0				**0,005422993492**
albania	0,005248058218	-	0				**0,005248058218**
						total:	**2,492747756**

Table 5.15: Video 1 - YOLO classification

cosine similarity	euclidean distance
0,3203793007	1,849673065

Table 5.16: Video 2 - YOLO classification

cosine similarity	euclidean distance
0,3203793007	1,776394103

Table 5.17: Video 3 - YOLO classification

algorithm for this metric is small (0.0098) and through the analysis, we could see that the values were located in the middle of the scale. The DNN algorithm gave the worst value for the third video considering this metric and it also has the biggest variance between the different videos (0.2544). On the other hand, the values for the YOLO algorithm are the same for every video, being the value closer to zero than to one, meaning that the videos do not represent well the news articles.

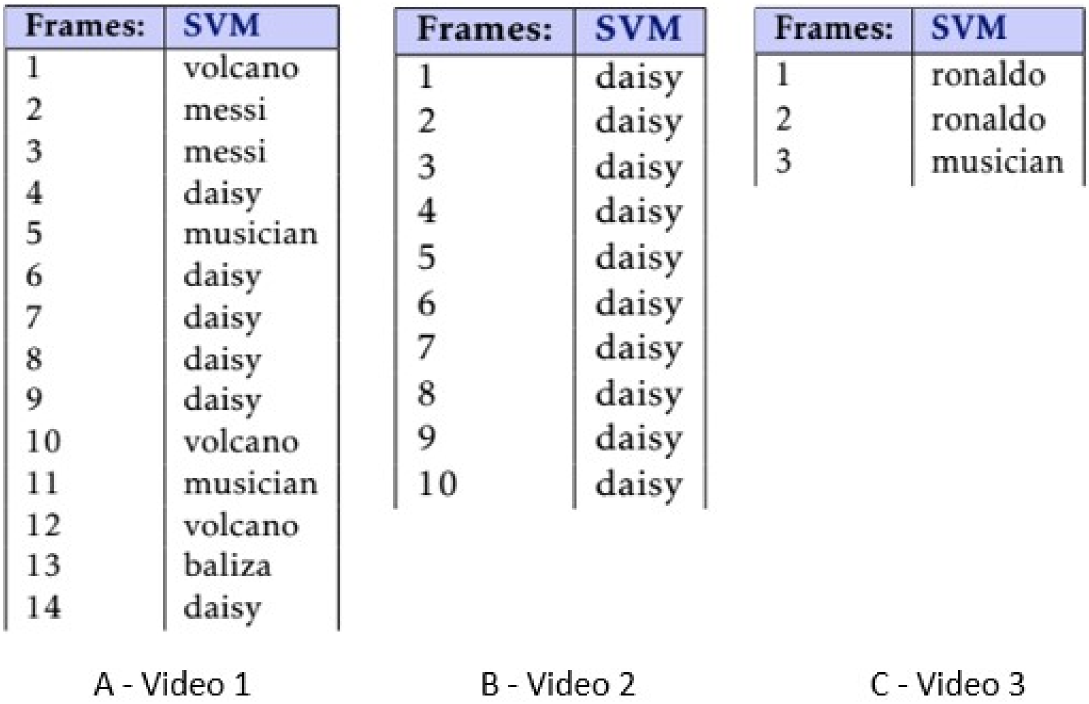
The algorithm that performed the best regarding the Euclidean Distance was the DNN algorithm. This algorithm, out of the entirety of the algorithms, had the value (0.2540) closest to the optimal value and had, overall, the best performance. However, unlike the other metric, the algorithm that had the worst result for this metric was the YOLO algorithm with the furthest value (2.4927) from the optimal one. This algorithm had the worst performance, being the values obtained, the worst in every video. Regarding the CNN algorithm, the values were in between the results form the others classifiers, neither the best nor the worst.

5.5.1 SVM classification

The suggestion to use the SVM algorithm was only to see how this algorithm behaved in the classification of complex scenes. Therefore, the results of its classification were not used to assess the videos.

This test used the new implementation of the SVM algorithm - the bag-of-visual-words. The set of images used to create the dictionary of visual words, were the same set used in the training of the CNN algorithm. The images used for detection, were the same ones that were classified by the others classifiers - the main frames of the scenes of the videos.

The results can be found in figure 5.13 where it shows the detection of the frames by the SVM algorithm.



Frames:	SVM
1	volcano
2	messi
3	messi
4	daisy
5	musician
6	daisy
7	daisy
8	daisy
9	daisy
10	volcano
11	musician
12	volcano
13	baliza
14	daisy

A - Video 1

Frames:	SVM
1	daisy
2	daisy
3	daisy
4	daisy
5	daisy
6	daisy
7	daisy
8	daisy
9	daisy
10	daisy

B - Video 2

Frames:	SVM
1	ronaldo
2	ronaldo
3	musician

C - Video 3

Figure 5.13: SVM classification

Like it was previously mentioned, the videos are related to the set of news on some level, therefore it was expected that the objects detected were related to the content of the news. From the image set provided to train the SVM, the objects that matched the content of the news and that were supposed to be found in the videos should be *musician*, *stage* or *street concert*.

From the results, we can see that this only happened in the first video. The objects detected in the first video can somewhat elude to a video that is related to a concert since it detects the object *musician*. It also detects the object *messi* which can be seen as the algorithm identified a person. The others objects that were detected have no relation to the video.

The classification for the second video misses completely the content of the video.

Regarding the results from the last video, the objects detected were *musician* and *ronaldo*. These detections are not accurate but it is not completely wrong since the video contains persons.

Results analysis After seeing the results, we can conclude that the SVM algorithm can identify some objects correctly. Therefore, it could be another source of complex image classification. Some of the problems faced in this test were related to the implementation of the algorithm, which was lacking, especially since it does not show the percentage of the objects detected.

5.5.2 Video Assessment - Joined classifiers

Seeing that the previous test helped to assess the videos considering each isolated classifier, we also wanted to assess the videos joining all the classifiers (DNN, CNN, YOLO).

This experience takes the results already computed from the previous test regarding the detection values. So for every object detected from each classifier, its values are combined on a single table. In case of term repetition, the term is only referenced one time but its value is the sum of the normalized value of the object detected related to the term in question by each algorithm. For example the term **performance** was related to the objects detected **stage** from DNN, and **stage/street_concert** from CNN. So the value for the object was for example: 0,4901 (value from **stage** DNN) + 0,2958(value from **stage** CNN) = 0,7859093673 (value for joined classifiers).

The results from this test can be found in tables 5.18, 5.19, 5.20. Complementary information can be found in C.7 and in C.8.

From the results displayed in the tables, we could see the different metrics giving different results for the best quality video given its semantic. For the Cosine Similarity metric the chosen video was the third video and for the Euclidean Distance metric the selected video was the second one. We can also see that the values for the Euclidean Distance metric is quite far from the optimal value, meaning that the difference between the arrays is significant.

words	w_values	detection	c_values	sqr of words values	sqr detection values	cosine similarity	euclidean distance
performance	0,167937863	stage/street_concert	0,7859093673	0,2241531654	3,339494936	**0,5043286817**	**0,6179715043**
stage	0,01460709537	stage/street_concert	0,7859093673				**0,7713022719**
music	0,009796375341	stage/street_concert	0,7859093673				**0,7761129919**
microphone	0,01434469246	stage/street_concert	0,7859093673				**0,7715646748**
show	0,01574417466	stage/street_concert	0,7859093673				**0,7701651926**
singer	0,02662514869	person/musician	0,9540793				**0,9274541513**
netta	0,01877638607	person/musician	0,9540793				**0,9353029139**
surie	0,01644391575	person/musician	0,9540793				**0,9376353842**
man	0,01504443356	person/ronaldo	0,8753859236				**0,86034149**
song	0,05709887342	-	0				**0,05709887342**
photos	0,06297669862	-	0				**0,06297669862**
ferrell	0,04618291232	person	0,8076923077				**0,7615093954**
eurovision	0,03292281856	-	0				**0,03292281856**
she	0,03148834931	person	0,8076923077				**0,7762039584**
film	0,02728990274	-	0				**0,02728990274**
contest	0,02632775873	-	0				**0,02632775873**
parody	0,02554055	-	0				**0,02554055**
gaza	0,02449093835	-	0				**0,02449093835**
prince	0,02344132671	person	0,8076923077				**0,784250981**
report	0,02029249178	-	0				**0,02029249178**
star	0,01994262123		0				**0,01994262123**
video	0,01952277657	-	0				**0,01952277657**
israel	0,01830989201		0				**0,01830989201**
message	0,01574417466	-	0				**0,01574417466**
israeli	0,01574417466	-	0				**0,01574417466**
ryan	0,01574417466	person	0,8076923077				**0,791948133**
media	0,01539430411	-	0				**0,01539430411**
crowd	0,01539430411	person	0,8076923077				**0,7922980036**
dress	0,01539430411	gown	0,01697838286				**0,00158407875**
countries	0,01521936883	-	0				**0,01521936883**
palestinian	0,01504443356	person	0,8076923077				**0,7926478741**
visit	0,01504443356	-	0				**0,01504443356**
flag	0,01434469246	-	0				**0,01434469246**
band	0,01399482192	-	0				**0,01399482192**
protester	0,01399482192	person	0,8076923077				**0,7936974858**
home	0,01364495137	-	0				**0,01364495137**
violence	0,01224546918	-	0				**0,01224546918**
world	0,01049611644	-	0				**0,01049611644**
barzilai	0,009796375341	-	0				**0,009796375341**
chinese	0,008047022602	-	0				**0,008047022602**
netherlands	0,006997410958	-	0				**0,006997410958**
secutity	0,006297669862	-	0				**0,006297669862**
europe	0,005597928766	-	0				**0,005597928766**
lisbon	0,005422993492	-	0				**0,005422993492**
albania	0,005248058218	-	0				**0,005248058218**
						total:	**3,133639424**

Table 5.18: Video 1 - Joined classifiers

cosine similarity	euclidean distance
0,470438735	2,136654725

Table 5.19: Video 2 - Joined classifiers

cosine similarity	euclidean distance
0,5063363807	2,407331127

Table 5.20: Video 3 - Joined classifiers

5.5.2.1 Joined classifiers - 100% probability

Like it was done in the previous test, this test combines the results from all the classifiers. The difference between the previous test and this one, is that the values for the confidence level of the detected objects from each classifier were changed to 100%. The main goal of this test was to see which video correlates the most with the news articles disregarding the accuracy of the classification and only focusing on the detected concepts.

The results from this test can be seen in the tables 5.21, 5.22 e 5.23. Complementary information can be found in C.9 and in C.10.

From the results of this test we can verify that regarding the Euclidean Distance metric

words	w_values	detection	c_values	sqr of words values	sqr detection values	cosine similarity	euclidean distance
performance	0,167937863	stage/street_concert	1,149350649	0,2241531654	4,252677965	**0,5205457068**	**0,9814127864**
stage	0,01460709537	stage/street_concert	1,149350649				**1,134743554**
music	0,009796375341	stage/street_concert	1,149350649				**1,139554274**
microphone	0,01434469246	stage/street_concert	1,149350649				**1,135005957**
show	0,01574417466	stage/street_concert	1,149350649				**1,133606475**
singer	0,02662514869	person/musician	1,195804196				**1,169179047**
netta	0,01877638607	person/musician	1,195804196				**1,17702781**
surie	0,01644391575	person/musician	1,195804196				**1,17936028**
man	0,01504443356	person/ronaldo	1,104895105				**1,089850671**
song	0,05709887342	-	0				**0,05709887342**
photos	0,06297669862	-	0				**0,06297669862**
ferrell	0,04618291232	person	0,9230769231				**0,8768940108**
eurovision	0,03292281856	-	0				**0,03292281856**
she	0,03148834931	person	0,9230769231				**0,8915885738**
film	0,02728990274	-	0				**0,02728990274**
contest	0,02632775873	-	0				**0,02632775873**
parody	0,02554055	-	0				**0,02554055**
gaza	0,02449093835	-	0				**0,02449093835**
prince	0,02344132671	person	0,9230769231				**0,8996355964**
report	0,02029249178	-	0				**0,02029249178**
star	0,01994262123		0				**0,01994262123**
video	0,01952277657	-	0				**0,01952277657**
israel	0,01830989201		0				**0,01830989201**
message	0,01574417466	-	0				**0,01574417466**
israeli	0,01574417466	-	0				**0,01574417466**
ryan	0,01574417466	person	0,9230769231				**0,9073327484**
media	0,01539430411	-	0				**0,01539430411**
crowd	0,01539430411	person	0,9230769231				**0,907682619**
dress	0,01539430411	gown	0,07142857143				**0,05603426732**
countries	0,01521936883	-	0				**0,01521936883**
palestinian	0,01504443356	person	0,9230769231				**0,9080324895**
visit	0,01504443356	-	0				**0,01504443356**
flag	0,01434469246	-	0				**0,01434469246**
band	0,01399482192	-	0				**0,01399482192**
protester	0,01399482192	person	0,9230769231				**0,9090821012**
home	0,01364495137	-	0				**0,01364495137**
violence	0,01224546918	-	0				**0,01224546918**
world	0,01049611644	-	0				**0,01049611644**
barzilai	0,009796375341	-	0				**0,009796375341**
chinese	0,008047022602	-	0				**0,008047022602**
netherlands	0,006997410958	-	0				**0,006997410958**
secutity	0,006297669862	-	0				**0,006297669862**
europe	0,005597928766	-	0				**0,005597928766**
lisbon	0,005422993492	-	0				**0,005422993492**
albania	0,005248058218	-	0				**0,005248058218**
						total:	**4,039703592**

Table 5.21: Video 1 - Joined classifiers - 100%

cosine similarity	euclidean distance
0,499982425	2,685117945

Table 5.22: Video 2 - Joined classifiers - 100%

cosine similarity	euclidean distance
0,4798523507	3,710150132

Table 5.23: Video 3 - Joined classifiers - 100%

the results matches the one we obtained from the previous test (joined classifiers). For the Cosine Similarity metric the best video was the first, while for the other metric, the chosen video was the second. And even more than before the values for the Euclidean Distance are even further of the optimal value.

Results analysis Taking under consideration the graph presented in 5.14 we can see that, the values of the detection considering the joined classifier is indeed better than the values of each individual classifier. And if we remove the confidence level of each classifier and then aggregate the detections of each one of them (joined classifier - 100%), we can see that the values are even better. This happens because, as expected, if we join the objects detected from each classifier, we will have more objects detected in a single video and therefore the value of the normalization of the array will be better than if we consider a single classifier, since it will have more data to represent the video. The results considering that each confidence level is 100%, provides an even better result, because in that case only the objects will be taken into account, which means that more objects direct implies better results.

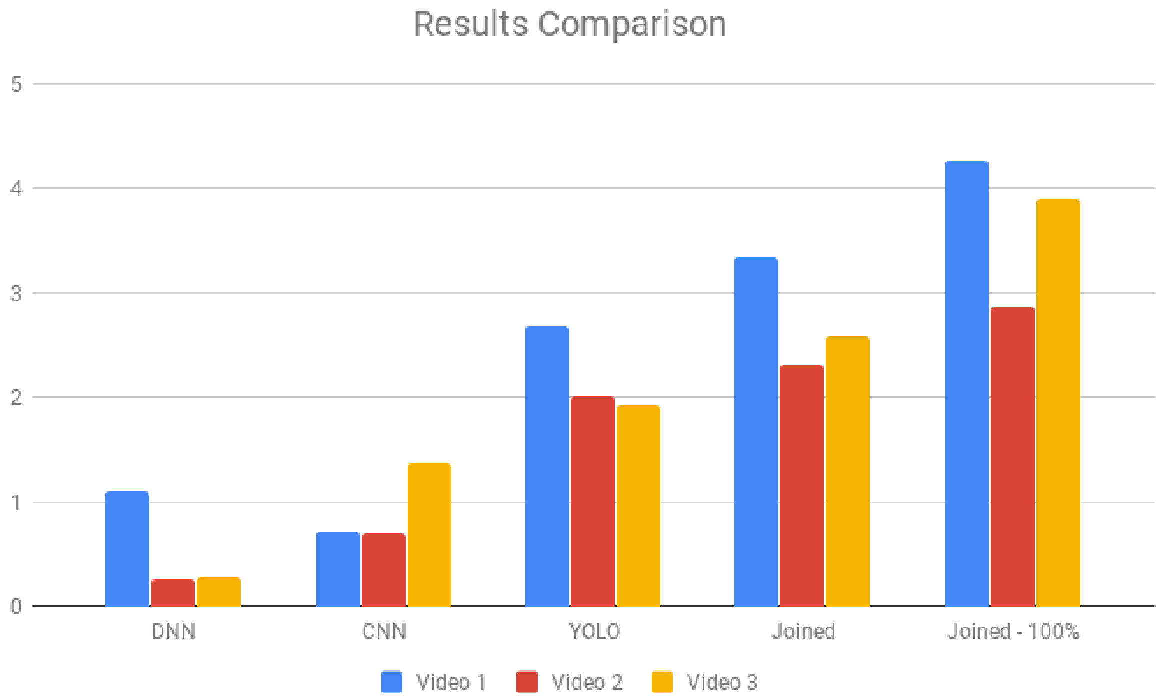


Figure 5.14: Comparison of the detection's normalized values

Regarding the assessment of the video quality using the news information, the results of the joined classifiers were different from the detection results. The comparison of the values of the metrics from both the joined classifiers and the individual classifier, can be seen in tables 5.24, 5.25, 5.26, 5.27, 5.28, 5.29. Analysing the results we can see that, the values for the Euclidean Distance metric were worst for the joined classifiers than for the individual classifiers. This it is due to the fact that even if the values of the detection are enhanced, it does not directly imply that the correlation between the news and the video will be improved. Considering, from the start, that an array with only a few objects detected is only slighted related to the news article, increasing the number of objects detected and hence increasing the elements in the array (removing the places in the array that were not taken into account in the equation because it had the value zero), could make the difference between the arrays even bigger.

For the Cosine Similarity metric we can see that for the entirety of the videos, either the joined classifiers or the joined classifiers with the confidence level of 100%, had better values than the individual classifier. Improving the amount of match of the arrays, resulted in improving the value of this metric. The ponctaul cases where the improvement does not occur can be explained to the fact previously mentioned.

DNN	CNN	YOLO	Joined	Joined - 100%
0,4447866847	0,4756386144	0,3203793007	0,5043286817	0,5205457068

Table 5.24: Comparison of the values for the Cosine Similarity metric - video 1

DNN	CNN	YOLO	Joined	Joined - 100%
0, 4437760658	0,4728443308	0,3203793007	0,470438735	0,499982425

Table 5.25: Comparison of the values for the Cosine Similarity metric - video 2

DNN	CNN	YOLO	Joined	Joined - 100%
0,1904242877	0,465843023	0,3203793007	0,5063363807	0,4798523507

Table 5.26: Comparison of the values for the Cosine Similarity metric - video 3

DNN	CNN	YOLO	Joined	Joined - 100%
1,016245247	0,6361205063	2,492747756	3,133639424	4,039703592

Table 5.27: Comparison of the values for the Euclidean Distance metric - video 1

5.6 Video Assessment - more frames

After analysing the first tests regarding the assessment of the video, a need to have more frames which would result in more concepts arose. Thus, a new test was made. This test used the third implementation of the DNN algorithm that consisted in classifying not the main frame of the shot as before, but three frames of the shot. Likely to what was done before, these frames were fed into the other two classifiers in order to detect the objects presents in it. The same process done in 5.5 was reproduced here, but instead of only one frame per scene, three frames and its classification were used.

The new results from the DNN can be found in tables 5.30, 5.31, 5.32, but also in C.11, C.12.

From the tables, we can see that the best quality video for both metrics is different. While for the Cosine Similarity metric the selected video is the first one, for the Euclidean Distance metric is the second one.

The results from the CNN classifier can be seen in tables 5.33, 5.34, 5.35, but also in C.13, C.14.

Observing the results, we can see that the best quality video for both metrics is the second video. The values of the Cosine Similarity metric are similar between the videos,

DNN	CNN	YOLO	Joined	Joined - 100%
0,2540671298	0,6305818987	1,849673065	2,136654725	2,685117945

Table 5.28: Comparison of the values for the Euclidean Distance metric - video 2

DNN	CNN	YOLO	Joined	Joined - 100%
0,324803232	1,275460046	1,776394103	2,407331127	3,710150132

Table 5.29: Comparison of the values for the Euclidean Distance metric - video 3

words	w_values	detection	c_values	sqr of words values	sqr detection values	cosine similarity	euclidean distance
performance	0,167937863	stage	0,3804771178	0,2241531654	0,8508138492	**0,444368646**	**0,2125392548**
stage	0,01460709537	stage	0,3804771178				**0,3658700224**
music	0,009796375341	stage	0,3804771178				**0,3706807425**
microphone	0,01434469246	stage	0,3804771178				**0,3661324253**
show	0,01574417466	stage	0,3804771178				**0,3647329431**
dress	0,01539430411	-	0				**0,01539430411**
song	0,05709887342	-	0				**0,05709887342**
photos	0,06297669862	-	0				**0,06297669862**
ferrell	0,04618291232	-	0				**0,04618291232**
eurovision	0,03292281856	-	0				**0,03292281856**
she	0,03148834931	-	0				**0,03148834931**
film	0,02728990274	-	0				**0,02728990274**
singer	0,02662514869	-	0				**0,02662514869**
contest	0,02632775873	-	0				**0,02632775873**
parody	0,02554055	-	0				**0,02554055**
gaza	0,02449093835	-	0				**0,02449093835**
prince	0,02344132671	-	0				**0,02344132671**
report	0,02029249178	-	0				**0,02029249178**
star	0,01994262123	-	0				**0,01994262123**
video	0,01952277657	-	0				**0,01952277657**
netta	0,01877638607	-	0				**0,01877638607**
israel	0,01830989201	-	0				**0,01830989201**
surie	0,01644391575	-	0				**0,01644391575**
message	0,01574417466	-	0				**0,01574417466**
israeli	0,01574417466	-	0				**0,01574417466**
ryan	0,01574417466	-	0				**0,01574417466**
media	0,01539430411	-	0				**0,01539430411**
crowd	0,01539430411	-	0				**0,01539430411**
countries	0,01521936883	-	0				**0,01521936883**
man	0,01504443356	-	0				**0,01504443356**
palestinian	0,01504443356	-	0				**0,01504443356**
visit	0,01504443356	-	0				**0,01504443356**
flag	0,01434469246	-	0				**0,01434469246**
band	0,01399482192	accordion	0,00836779878				**0,005627023135**
protester	0,01399482192	-	0				**0,01399482192**
home	0,01364495137	-	0				**0,01364495137**
violence	0,01224546918	-	0				**0,01224546918**
world	0,01049611644	-	0				**0,01049611644**
barzilai	0,009796375341	-	0				**0,009796375341**
chinese	0,008047022602	-	0				**0,008047022602**
netherlands	0,006997410958	-	0				**0,006997410958**
secutity	0,006297669862	-	0				**0,006297669862**
europe	0,005597928766	-	0				**0,005597928766**
lisbon	0,005422993492	-	0				**0,005422993492**
albania	0,005248058218	-	0				**0,005248058218**
						total:	**0,777434115**

Table 5.30: Video 1 - DNN classification - more frames

having a small variance. The values for the other metric have a large variance, 1.5509. The worst video for this classifier, considering both metrics, is the third video.

In the tables 5.36, 5.37, 5.38 and also in the tables C.15, C.16 found in the appendix C the results from the YOLO classifier can be seen.

The results from the Cosine Similarity metric are the same for all the videos, making it unable to choose a single video as the best one. The same could not be said for the Euclidean Distance metric since this metric selected the third video as the best. The values for this metric are quite far from the optimal value, being the value for the first

cosine similarity	euclidean distance
0,4437760658	0,2036728787

Table 5.31: Video 2 - DNN classification - more frames

cosine similarity	euclidean distance
0,1904242877	0,2405606299

Table 5.32: Video 3 - DNN classification - more frames

words	w_values	detection	c_values	sqr of words values	sqr detection values	cosine similarity	euclidean distance
performance	0,167937863	stage/street_concert	0,3289175477	0,2241531654	0,8173230046	0,4725441282	0,1609796847
stage	0,01460709537	stage/street_concert	0,3289175477				0,3143104523
music	0,009796375341	stage/street_concert	0,3289175477				0,3191211723
microphone	0,01434469246	stage/street_concert	0,3289175477				0,3145728552
show	0,01574417466	stage/street_concert	0,3289175477				0,313173373
singer	0,02662514869	musician	0,2032807871				0,1766556384
netta	0,01877638607	musician	0,2032807871				0,1845044011
surie	0,01644391575	musician	0,2032807871				0,1868368714
man	0,01504443356	ronaldo	0,055802265				0,04075783144
song	0,05709887342	-	0				0,05709887342
photos	0,06297669862	-	0				0,06297669862
ferrell	0,04618291232	-	0				0,04618291232
eurovision	0,03292281856	-	0				0,03292281856
she	0,03148834931	-	0				0,03148834931
film	0,02728990274	-	0				0,02728990274
contest	0,02632775873	-	0				0,02632775873
parody	0,02554055	-	0				0,02554055
gaza	0,02449093835	-	0				0,02449093835
prince	0,02344132671	-	0				0,02344132671
report	0,02029249178	-	0				0,02029249178
star	0,01994262123		0				0,01994262123
video	0,01952277657	-	0				0,01952277657
israel	0,01830989201		0				0,01830989201
message	0,01574417466	-	0				0,01574417466
israeli	0,01574417466	-	0				0,01574417466
ryan	0,01574417466	-	0				0,01574417466
media	0,01539430411	-	0				0,01539430411
crowd	0,01539430411	-	0				0,01539430411
dress	0,01539430411	-	0				0,01539430411
countries	0,01521936883		0				0,01521936883
palestinian	0,01504443356	-	0				0,01504443356
visit	0,01504443356	-	0				0,01504443356
flag	0,01434469246	-	0				0,01434469246
band	0,01399482192	-	0				0,01399482192
protester	0,01399482192	-	0				0,01399482192
home	0,01364495137	-	0				0,01364495137
violence	0,01224546918	-	0				0,01224546918
world	0,01049611644	-	0				0,01049611644
barzilai	0,009796375341	-	0				0,009796375341
chinese	0,008047022602	-	0				0,008047022602
netherlands	0,006997410958	-	0				0,006997410958
secutity	0,006297669862	-	0				0,006297669862
europe	0,005597928766	-	0				0,005597928766
lisbon	0,005422993492	-	0				0,005422993492
albania	0,005248058218	-	0				0,005248058218
						total:	**0,738159297**

Table 5.33: Video 1 - CNN classification - more frames

video the worst one.

Results analysis Having in mind that this test followed the one presented in section 5.5, it is important to compare the results obtained in this test with the previous one. This way, we will be able to see if classifying more frames, improves the results. To make it simples to describe, the test that consider only the main frame will be reffered as **testMain** and the test considering more frames will be reffered as **testMore**.

For the DNN algorithm, the results in testMore matches the results previously found in testMain given that the same videos were selected. The difference between the tests

cosine similarity	euclidean distance
0,4747816774	0,5443146709

Table 5.34: Video 2 - CNN classification - more frames

cosine similarity	euclidean distance
0,4437760658	2,095233529

Table 5.35: Video 3 - CNN classification - more frames

words	w_values	detection	c_values	sqr of words values	sqr detection values	cosine similarity	euclidean distance
performance	0,167937863	-	0	0,2241531654	2,677686779	**0,3203793007**	**0,167937863**
stage	0,01460709537	-	0				**0,01460709537**
music	0,009796375341	-	0				**0,009796375341**
microphone	0,01434469246	-	0				**0,01434469246**
show	0,01574417466	-	0				**0,01574417466**
singer	0,02662514869	person	0,8073529412				**0,7807277925**
netta	0,01877638607	person	0,8073529412				**0,7885765551**
surie	0,01644391575	person	0,8073529412				**0,7909090254**
man	0,01504443356	person	0,8073529412				**0,7923085076**
song	0,05709887342	-	0				**0,05709887342**
photos	0,06297669862	-	0				**0,06297669862**
ferrell	0,04618291232	person	0,8073529412				**0,7611700289**
eurovision	0,03292281856	-	0				**0,03292281856**
she	0,03148834931	person	0,8073529412				**0,7758645919**
film	0,02728990274	-	0				**0,02728990274**
contest	0,02632775873	-	0				**0,02632775873**
parody	0,02554055	-	0				**0,02554055**
gaza	0,02449093835	-	0				**0,02449093835**
prince	0,02344132671	person	0,8073529412				**0,7839116145**
report	0,02029249178	-	0				**0,02029249178**
star	0,01994262123		0				**0,01994262123**
video	0,01952277657	-	0				**0,01952277657**
israel	0,01830989201		0				**0,01830989201**
message	0,01574417466	-	0				**0,01574417466**
israeli	0,01574417466	-	0				**0,01574417466**
ryan	0,01574417466	person	0,8073529412				**0,7916087665**
media	0,01539430411	-	0				**0,01539430411**
crowd	0,01539430411	person	0,8073529412				**0,7919586371**
dress	0,01539430411	-	0				**0,01539430411**
countries	0,01521936883	-	0				**0,01521936883**
palestinian	0,01504443356	person	0,8073529412				**0,7923085076**
visit	0,01504443356	-	0				**0,01504443356**
flag	0,01434469246	-	0				**0,01434469246**
band	0,01399482192	-	0				**0,01399482192**
protester	0,01399482192	person	0,8073529412				**0,7933581193**
home	0,01364495137	-	0				**0,01364495137**
violence	0,01224546918	-	0				**0,01224546918**
world	0,01049611644	-	0				**0,01049611644**
barzilai	0,009796375341	-	0				**0,009796375341**
chinese	0,008047022602	-	0				**0,008047022602**
netherlands	0,006997410958	-	0				**0,006997410958**
secutity	0,006297669862	-	0				**0,006297669862**
europe	0,005597928766	-	0				**0,005597928766**
lisbon	0,005422993492	-	0				**0,005422993492**
albania	0,005248058218	-	0				**0,005248058218**
						total:	**2,491678484**

Table 5.36: Video 1 - YOLO classification - more frames

is that, for the Euclidean Distance metric, the value for the testMore for the best video was better (0.2037 instead of 0.2541) while for the Cosine Similarity was slightly worst (0.4448 instead of 0.4444). Regarding the others videos, the values from both tests were similar.

Considering the CNN classifier, the results presented in testMore does not confirm the results found in the testMain, because in the former test we can see that the second video was the chosen one for both metrics. The difference is that, for the Cosine Similarity metric, the best quality video was the second one and not the first one like in testMain,

cosine similarity	euclidean distance
0,3203793007	2,030648533

Table 5.37: Video 2 - YOLO classification - more frames

cosine similarity	euclidean distance
0,3203793007	1,721463336

Table 5.38: Video 3 - YOLO classification - more frames

although the difference between the values for the first video and the second video in testMore is relatively small (0.0022). Considering all the videos, the values for this metric are overall worst in testMore. For the Euclidean distance metric, the values for the videos that were not chosen were also worst, however the value for the best video was better (0.5443 instead of 0.6306).

The results for the YOLO classifier also gave the same result for both tests. The impossibility to choose a single video considering only the Cosine Similarity metric, given that the results for every video are the same. And also that the third video was the one selected as the best video given the Euclidean Distance metric. The values of the Cosine Similarity for the videos in both tests were the same (0.3204) which is closer to the worst value than the optimal value. The Euclidean Distance, except for the third video, have bigger values in comparison to the values from the testMain, indicating that the vectors are further away from each other. Regarding the third video, its value is better in the testMore than it is in the testMain.

From the analysis of both the testMain and testMore, we can see that the testMain had overall better results than the testMore. This happened because even though more frames were used for detection, the objects detected were either the same or did not have a match with news article, therefore did not increase the detection array. As the array of detections go through a normalization process that involves the division by the total number of detections of the classifier and because the confidence level of these detections (numerator of the equation) did not change much from the ones in the testMain, the bigger the number of detections, the smaller is the normalized value of each object. Which explains why the values of the testMore is slightly worse than the results found in testMain.

5.6.1 Video Assessment - more frames - joined classifiers

Like it was done previously for the test using only the main frames, this test combined the results from every classifier in order to see if the values of the metrics could be improved. This test followed the same methodology as the one done in section 5.5.2.

The tables 5.39, 5.40, 5.41 and tables C.17, C.18 from appendix C presents the result from the test that combined all the classifiers using more frames for classification.

Through the results, we can see that the second video, considering the Euclidean

words	w_values	detection	c_values	sqr of words values	sqr detection values	cosine similarity	euclidean distance
performance	0,167937863	stage/street_concert	0,7093946655	0,2241531654	3,29972482	**0,4916081465**	**0,5414568025**
stage	0,01460709537	stage/street_concert	0,7093946655				**0,6947875701**
music	0,009796375341	stage/street_concert	0,7093946655				**0,6995982901**
microphone	0,01434469246	stage/street_concert	0,7093946655				**0,695049973**
show	0,01574417466	stage/street_concert	0,7093946655				**0,6936504908**
singer	0,02662514869	person/musician	1,010633728				**0,9840085796**
netta	0,01877638607	person/musician	1,010633728				**0,9918573422**
surie	0,01644391575	person/musician	1,010633728				**0,9941898126**
man	0,01504443356	person/ronaldo	0,8631552062				**0,8481107726**
song	0,05709887342	-	0				**0,05709887342**
photos	0,06297669862	-	0				**0,06297669862**
ferrell	0,04618291232	person	0,8073529412				**0,7611700289**
eurovision	0,03292281856	-	0				**0,03292281856**
she	0,03148834931	person	0,8073529412				**0,7758645919**
film	0,02728990274	-	0				**0,02728990274**
contest	0,02632775873	-	0				**0,02632775873**
parody	0,02554055	-	0				**0,02554055**
gaza	0,02449093835	-	0				**0,02449093835**
prince	0,02344132671	person	0,8073529412				**0,7839116145**
report	0,02029249178	-	0				**0,02029249178**
star	0,01994262123		0				**0,01994262123**
video	0,01952277657	-	0				**0,01952277657**
israel	0,01830989201		0				**0,01830989201**
message	0,01574417466	-	0				**0,01574417466**
israeli	0,01574417466	-	0				**0,01574417466**
ryan	0,01574417466	person	0,8073529412				**0,7916087665**
media	0,01539430411	-	0				**0,01539430411**
crowd	0,01539430411	person	0,8073529412				**0,7919586371**
dress	0,01539430411	-	0				**0,01539430411**
countries	0,01521936883	-	0				**0,01521936883**
palestinian	0,01504443356	person	0,8073529412				**0,7923085076**
visit	0,01504443356	-	0				**0,01504443356**
flag	0,01434469246	-	0				**0,01434469246**
band	0,01399482192	accordion	0,00836779878				**0,005627023135**
protester	0,01399482192	person	0,8073529412				**0,7933581193**
home	0,01364495137	-	0				**0,01364495137**
violence	0,01224546918	-	0				**0,01224546918**
world	0,01049611644	-	0				**0,01049611644**
barzilai	0,009796375341	-	0				**0,009796375341**
chinese	0,008047022602	-	0				**0,008047022602**
netherlands	0,006997410958	-	0				**0,006997410958**
secutity	0,006297669862	-	0				**0,006297669862**
europe	0,005597928766	-	0				**0,005597928766**
lisbon	0,005422993492	-	0				**0,005422993492**
albania	0,005248058218	-	0				**0,005248058218**
						total:	**3,095803651**

Table 5.39: Video 1 - Joined classifiers - more frames

cosine similarity	euclidean distance
0,4255588613	2,515783944

Table 5.40: Video 2 - Joined classifiers - more frames

cosine similarity	euclidean distance
0,5464057543	2,747604732

Table 5.41: Video 3 - Joined classifiers - more frames

Distance, was assessed as the best quality video given its semantic. The values for this metric, are quite far away from its optimal value. A different result was achieved using the Cosine Similarity metric, since it selected the third video as the best one. The values for this metric are close to the middle of the scale of possible values for this metric.

5.6.1.1 Video Assessment - more frames - all classifiers - 100%

Much like it was done in section 5.5.2.1, this test took the values of the confidence level of every object detected of each classifier for every frame and set it to 1 (maximum value). The results can be seen in tables 5.42, 5.43, 5.44. The detailed results of the video 2 and video 3 can be seen in C.19, C.20

words	w_values	detection	c_values	sqr of words values	sqr detection values	cosine similarity	euclidean distance
performance	0,167937863	stage/street_concert	1,129355401	0,2241531654	4,225274695	**0,5184252156**	**0,9614175377**
stage	0,01460709537	stage/street_concert	1,129355401				**1,114748305**
music	0,009796375341	stage/street_concert	1,129355401				**1,119559025**
microphone	0,01434469246	stage/street_concert	1,129355401				**1,115010708**
show	0,01574417466	stage/street_concert	1,129355401				**1,113611226**
singer	0,02662514869	person/musician	1,195804196				**1,169179047**
netta	0,01877638607	person/musician	1,195804196				**1,17702781**
surie	0,01644391575	person/musician	1,195804196				**1,17936028**
man	0,01504443356	person/ronaldo	1,104895105				**1,089850671**
song	0,05709887342	-	0				**0,05709887342**
photos	0,06297669862	-	0				**0,06297669862**
ferrell	0,04618291232	person	0,9230769231				**0,8768940108**
eurovision	0,03292281856	-	0				**0,03292281856**
she	0,03148834931	person	0,9230769231				**0,8915885738**
film	0,02728990274	-	0				**0,02728990274**
contest	0,02632775873	-	0				**0,02632775873**
parody	0,02554055	-	0				**0,02554055**
gaza	0,02449093835	-	0				**0,02449093835**
prince	0,02344132671	person	0,9230769231				**0,8996355964**
report	0,02029249178	-	0				**0,02029249178**
star	0,01994262123		0				**0,01994262123**
video	0,01952277657	-	0				**0,01952277657**
israel	0,01830989201		0				**0,01830989201**
message	0,01574417466	-	0				**0,01574417466**
israeli	0,01574417466	-	0				**0,01574417466**
ryan	0,01574417466	person	0,9230769231				**0,9073327484**
media	0,01539430411	-	0				**0,01539430411**
crowd	0,01539430411	person	0,9230769231				**0,907682619**
dress	0,01539430411	-	0				**0,01539430411**
countries	0,01521936883	-	0				**0,01521936883**
palestinian	0,01504443356	person	0,9230769231				**0,9080324895**
visit	0,01504443356	-	0				**0,01504443356**
flag	0,01434469246	-	0				**0,01434469246**
band	0,01399482192	accordion	0,0243902439				**0,01039542199**
protester	0,01399482192	person	0,9230769231				**0,9090821012**
home	0,01364495137	-	0				**0,01364495137**
violence	0,01224546918	-	0				**0,01224546918**
world	0,01049611644	-	0				**0,01049611644**
barzilai	0,009796375341	-	0				**0,009796375341**
chinese	0,008047022602	-	0				**0,008047022602**
netherlands	0,006997410958	-	0				**0,006997410958**
secutity	0,006297669862	-	0				**0,006297669862**
europe	0,005597928766	-	0				**0,005597928766**
lisbon	0,005422993492	-	0				**0,005422993492**
albania	0,005248058218	-	0				**0,005248058218**
						total:	**4,012143204**

Table 5.42: Video 1 - Joined classifiers - more frames - 100%

cosine similarity	euclidean distance
0,4603525751	2,97641379

Table 5.43: Video 2 - Joined classifiers - more frames - 100%

cosine similarity	euclidean distance
0,5416014228	3,005180158

Table 5.44: Video 3 - Joined classifiers - more frames - 100%

Analyzing the tables, it is possible to see that, for the Cosine Similarity metric, the video that correlates the most with the news articles is the third. And the one that correlates the least is the second, although the difference of values between both of them is quite small (0.0812). For the Euclidean Distance metric, the chosen video is the second one because it has the closest value from the optimal one, although all of them are quite far away from optimal value.

Results Analysis Like it was done in the section 5.5, the idea to join the classifiers and also to join them while setting all of the objects's confidence level to one, arose in order to see if the results from the assessment could be improved. The summarization of the results can be seen in the graph 5.15 and tables 5.45, 5.46, 5.47, 5.48, 5.49, 5.50.

From the graph we can see that, much like it was concluded in the previous section 5.5, the normalized value of the detection for the joined classifier with every confidence level's value at 100% is bigger than the values for the joined classifier which in turn is bigger than the values for each individual classifier. This proves that aggregating the objects detected from each classifier does improve the overall value of the detection.

Concerning the results from the tables, we can see that the results were rather similar to the ones found in section 5.5. The values from the Euclidean Distance are worst for every video if we consider the joined classifiers. The reason explained in the mentioned section can be once again applied to this case. Even if the detection values are better it does not imply that the correlation between video and news will improve. The Cosine Similarity metric had similar results, except for the second video. The values for the joined classifiers were better in two out of three cases. The case where the results were not the expected, can be explained by the not implication between increasing the detection values and the improvement of the correlation, like it was already explained in the section 5.5.

Once again, we can conclude that classifying more frames does not improve the semantic video quality assessment.

DNN	CNN	YOLO	Joined	Joined - 100%
0,444368646	0,4725441282	0,3203793007	0,4916081465	0,5184252156

Table 5.45: Comparison of the values for the Cosine Similarity metric - video 1 - more frames

DNN	CNN	YOLO	Joined	Joined - 100%
0,4437760658	0,4747816774	0,3203793007	0,4255588613	0,4603525751

Table 5.46: Comparison of the values for the Cosine Similarity metric - video 2 - more frame

DNN	CNN	YOLO	Joined	Joined - 100%
0,1904242877	0,4437760658	0,3203793007	0,5464057543	0,5416014228

Table 5.47: Comparison of the values for the Cosine Similarity metric - video 3 - more frame

The table 5.51 summarizes the results of the video assessment obtained through the tests that used only the main frame of each shot. While the table 5.52 summarizes the result from the test that considered more frames for the classification.

DNN	CNN	YOLO	Joined	Joined - 100%
0,777434115	0,738159297	2,491678484	3,095803651	4,012143204

Table 5.48: Comparison of the values for the Euclidean Distance metric - video 1 - more frame

DNN	CNN	YOLO	Joined	Joined - 100%
0,2036728787	0,5443146709	2,030648533	2,515783944	2,97641379

Table 5.49: Comparison of the values for the Euclidean Distance metric - video 2 - more frame

DNN	CNN	YOLO	Joined	Joined - 100%
0,2405606299	2,095233529	1,721463336	2,747604732	3,005180158

Table 5.50: Comparison of the values for the Euclidean Distance metric - video 3 - more frame

5.7 Video Assessment by paragraph

Apart from the detection, one of the main goal of this thesis was to assess a video considering its story. The goal was to see if the video follows the same ”story” or the same structure of the news. If so, then the video was semantically accepted.

The assessment of the video considering its ”story” followed the same logic used in 5.5.

The steps followed by this test were:

- **Paragraph wordcount** - Using the same news dataset that was previously used for the video assessment, the news article that was directly related to a video was chosen. From each news chosen, the wordcount of each paragraph was done using its respective implementation (4.2.2).

- **Wordcount normalization** - The values of the wordcount were placed in an excel sheet. Using the **sum** operation of excel, we were able to obtain the sum of the wordcount of all terms. The normalization of the values consisted in dividing each term by the sum of the terms. This procedure was done for every paragraph of the news article.

- **Objects detection** - Considering that the videos used in this test were the ones previously used in the other test, the detection had already been done. The values of it were reused for this test.

- **Classification normalization** - In order to normalize the classification, each value of the confidence level of each frame was divided by the total amount of objects detected by the classifier. This action was done for each classifier. Which means that for every frame, a different value of the classification normalization was obtained as long as the confidence level of the objects were different.

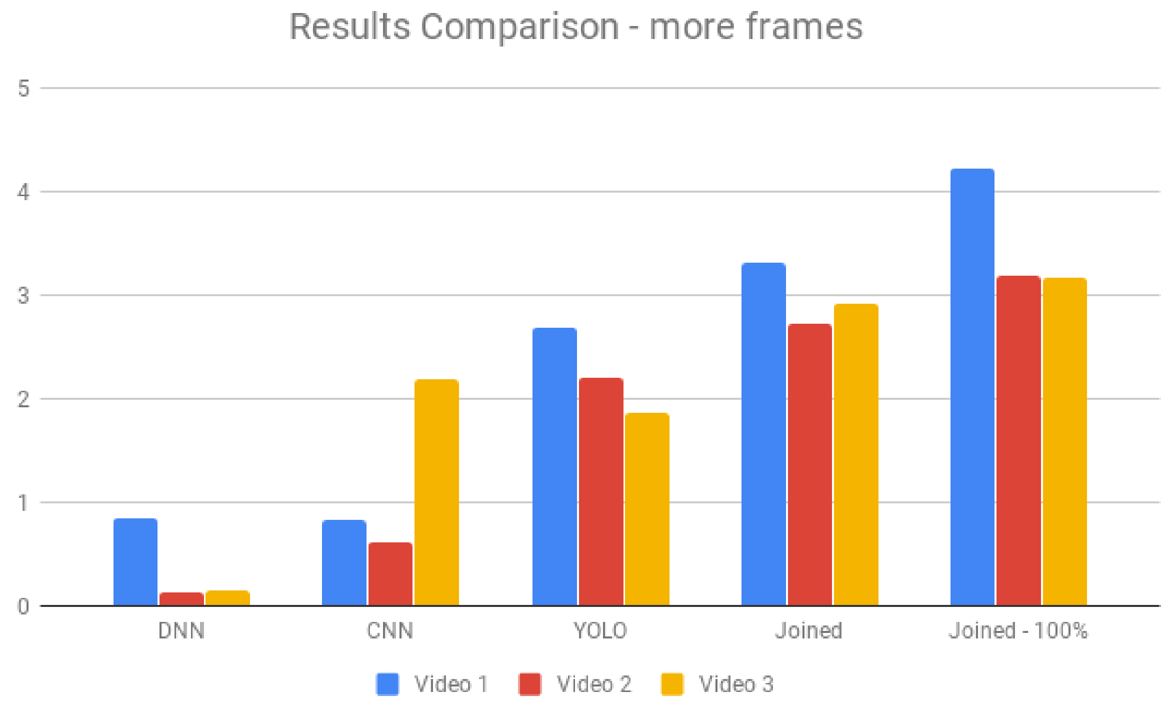


Figure 5.15: Comparison of the detection's normalized values - more frames

	DNN		CNN		YOLO		All classifiers		All classifiers - 100%	
	Cosine Similarity	Euclidean Distance	Cosine Similarity	Euclidean Distance	Cosine Similarity	Euclidean Distance	Cosine Similarity	Euclidean Distance	Cosine Similarity	Euclidean Distance
Video 1	X		X		X				X	
Video 2		X		X	X			X		X
Video 3					X	X	X			

Table 5.51: Resuts summary

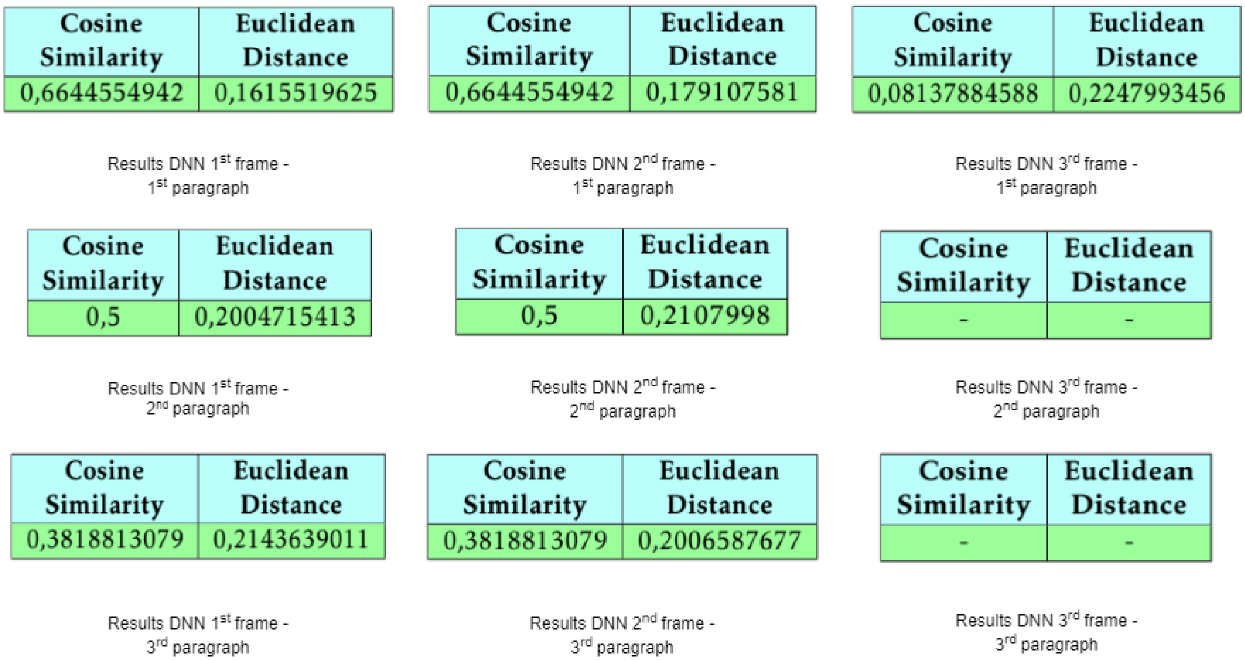
	DNN		CNN		YOLO		All classifiers		All classifiers - 100%	
	Cosine Similarity	Euclidean Distance	Cosine Similarity	Euclidean Distance	Cosine Similarity	Euclidean Distance	Cosine Similarity	Euclidean Distance	Cosine Similarity	Euclidean Distance
Video 1	X				X					
Video 2		X	X	X	X			X		X
Video 3					X	X	X		X	

Table 5.52: Resuts summary - more frames

- **Metrics** - Much like what was performed in 5.5, a table was done with the values that were previously calculated and the values of the metrics respective of each case. Having that in mind, a table was designed for each frame for each paragraph and for each classifier. This means that giving a article with three paragraph, and the video related to it had five main frames that were classified by each one of the three algorithms, we would have 15 tables for each one of the classifiers. Therefore, considering all classifiers, we would have 45 tables.

The reason for having a value of the metric for each frame, is because we wanted to see which frame(s) could relate the most for each paragraph and then from that, figure out the structure of the video and see if it matches the one from the news article.

The resuts from this test regarding one News can be seen in figure 5.16, 5.17, 5.18.



Cosine Similarity	Euclidean Distance
0,6644554942	0,1615519625

Results DNN 1st frame - 1st paragraph

Cosine Similarity	Euclidean Distance
0,6644554942	0,179107581

Results DNN 2nd frame - 1st paragraph

Cosine Similarity	Euclidean Distance
0,08137884588	0,2247993456

Results DNN 3rd frame - 1st paragraph

Cosine Similarity	Euclidean Distance
0,5	0,2004715413

Results DNN 1st frame - 2nd paragraph

Cosine Similarity	Euclidean Distance
0,5	0,2107998

Results DNN 2nd frame - 2nd paragraph

Cosine Similarity	Euclidean Distance
-	-

Results DNN 3rd frame - 2nd paragraph

Cosine Similarity	Euclidean Distance
0,3818813079	0,2143639011

Results DNN 1st frame - 3rd paragraph

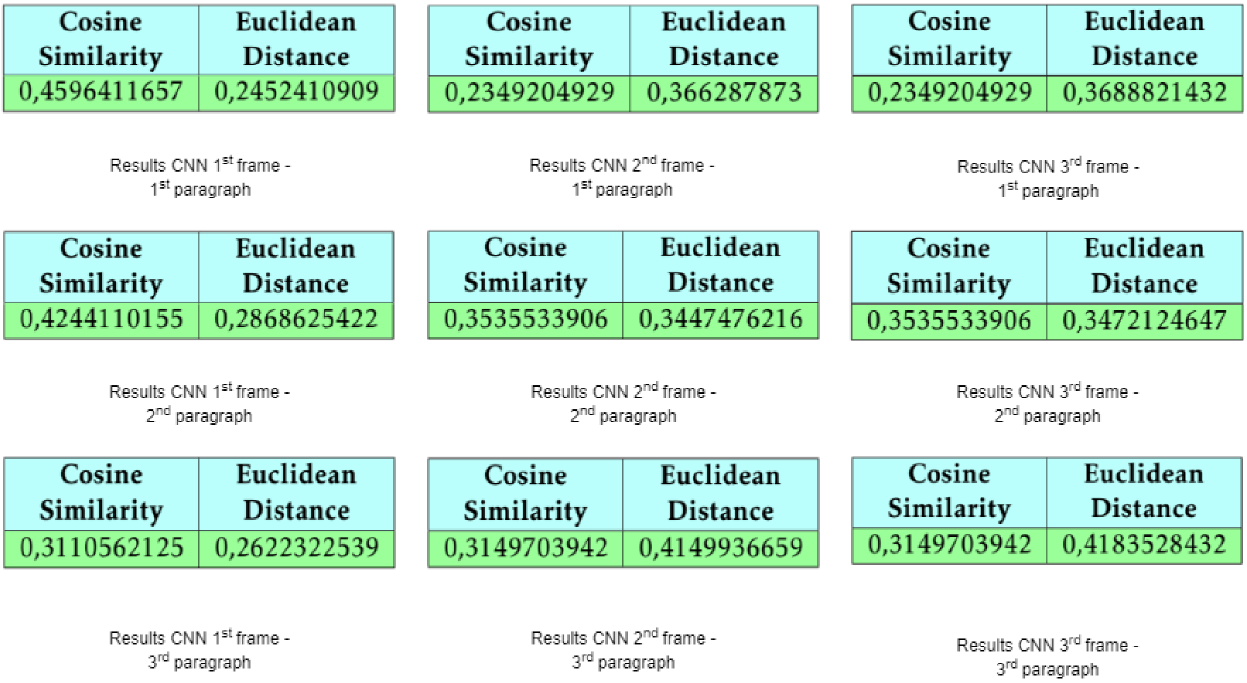
Cosine Similarity	Euclidean Distance
0,3818813079	0,2006587677

Results DNN 2nd frame - 3rd paragraph

Cosine Similarity	Euclidean Distance
-	-

Results DNN 3rd frame - 3rd paragraph

Figure 5.16: DNN results



Cosine Similarity	Euclidean Distance
0,4596411657	0,2452410909

Results CNN 1st frame - 1st paragraph

Cosine Similarity	Euclidean Distance
0,2349204929	0,366287873

Results CNN 2nd frame - 1st paragraph

Cosine Similarity	Euclidean Distance
0,2349204929	0,3688821432

Results CNN 3rd frame - 1st paragraph

Cosine Similarity	Euclidean Distance
0,4244110155	0,2868625422

Results CNN 1st frame - 2nd paragraph

Cosine Similarity	Euclidean Distance
0,3535533906	0,3447476216

Results CNN 2nd frame - 2nd paragraph

Cosine Similarity	Euclidean Distance
0,3535533906	0,3472124647

Results CNN 3rd frame - 2nd paragraph

Cosine Similarity	Euclidean Distance
0,3110562125	0,2622322539

Results CNN 1st frame - 3rd paragraph

Cosine Similarity	Euclidean Distance
0,3149703942	0,4149936659

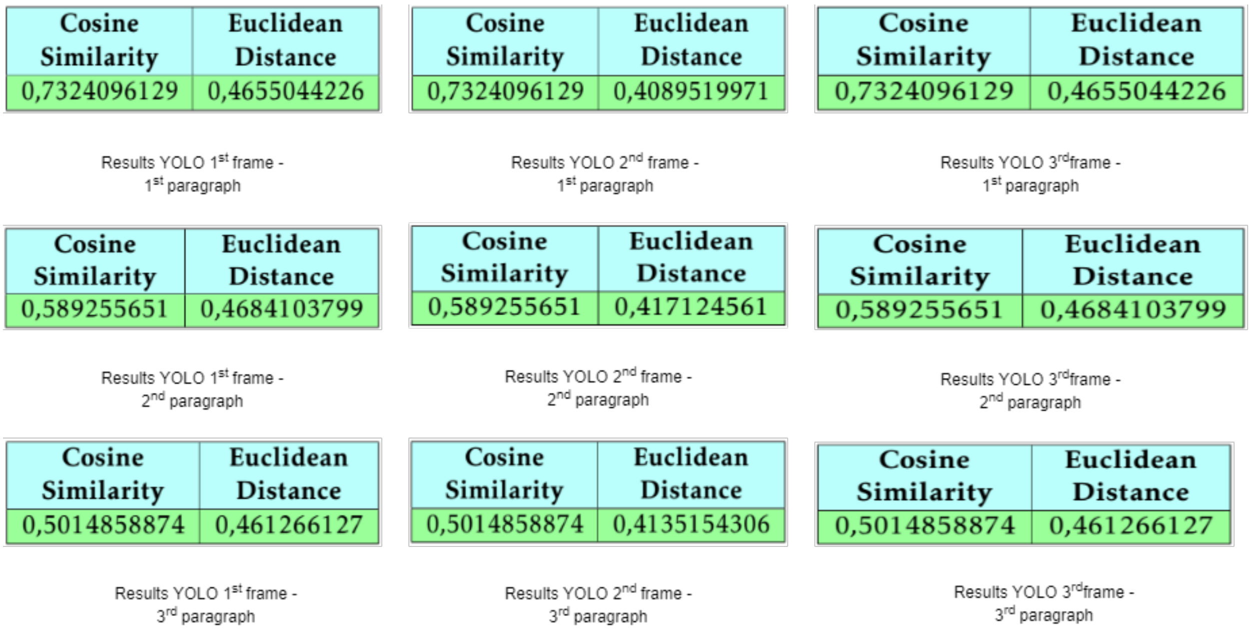
Results CNN 2nd frame - 3rd paragraph

Cosine Similarity	Euclidean Distance
0,3149703942	0,4183528432

Results CNN 3rd frame - 3rd paragraph

Figure 5.17: CNN results

After the analysis of the results, we can see that for the DNN algorithm, the values of the Cosine Similarity metric were always the same for the three frames for each paragraph,



Cosine Similarity	Euclidean Distance
0,7324096129	0,4655044226

Results YOLO 1st frame - 1st paragraph

Cosine Similarity	Euclidean Distance
0,7324096129	0,4089519971

Results YOLO 2nd frame - 1st paragraph

Cosine Similarity	Euclidean Distance
0,7324096129	0,4655044226

Results YOLO 3rdframe - 1st paragraph

Cosine Similarity	Euclidean Distance
0,589255651	0,4684103799

Results YOLO 1st frame - 2nd paragraph

Cosine Similarity	Euclidean Distance
0,589255651	0,417124561

Results YOLO 2nd frame - 2nd paragraph

Cosine Similarity	Euclidean Distance
0,589255651	0,4684103799

Results YOLO 3rdframe - 2nd paragraph

Cosine Similarity	Euclidean Distance
0,5014858874	0,461266127

Results YOLO 1st frame - 3rd paragraph

Cosine Similarity	Euclidean Distance
0,5014858874	0,4135154306

Results YOLO 2nd frame - 3rd paragraph

Cosine Similarity	Euclidean Distance
0,5014858874	0,461266127

Results YOLO 3rdframe - 3rd paragraph

Figure 5.18: YOLO results

except for the first paragraph. This means that the values of the metric for the frames 1, 2 and 3 were the same for the 2nd paragraph, the same being applied for the 3rd paragraph. Except for the 1st paragraph that had the value for the 3rd frame different from the values of frame 1 and 2. However the best value for this metric can be found in the first paragraph for both the first and the second frames (0.6645), while the worst is related to the third paragraph. Using the Euclidean Distance metric we can better differenciate the videos. For the first paragraph, the frame which array had the smaller distance from the news's array was the first. The same was verified for the second paragraph. For the third paragraph, the best value is related to the second frame. The third frame could not be matched with the second and the third paragraph.

For the CNN algorithm, the best value for the Cosine Similarity metric was achieved in the first frame of the first paragraph, although the value it is not close to the optimal value. The same happened for the Euclidean Distance. The frame that represents the best each paragraph is the first, for both the metrics.

Analysing the results from the YOLO algorithm, it is possible to see that the value for the Cosine Similarity is the same for the three frames for each paragraph. And the highest value was achieved in the first paragraph. The Euclidean Distance metric has the best value for the second frame of the second paragraph. The second frame was the one selected as the best in all the paragraphs.

Results analysis Through the analysis of the results previously presented we can see that using the DNN algorithm, we can say that for the first and the second paragraph, the frame that correlates the most with its content it is the first frame. While for the third paragraph the stronger correlation is with the second frame. This could indicate that the video followed at some point the structure of the news article.

On the other hand, using the CNN algorithm, we could not reach a strong conclusion regarding the structure of the video, given that from the results obtained, we could see that the first frame is the one that matches the most the entire news article. Therefore, we can not conclude anything related to the video's ”story”.

Much like what happened with the previously mentioned algorithm, we can not came to a conclusion regarding the video's ”story” considering the YOLO algorithm, because now it is the second frame that was chosen as the best out of the frames for every paragraph.

As a conclusion, we can see that although the results were not the best, we could still see that using these metrics, we can deduce the relationship between the video and the articles news's structure. Therefore, inferring the ”story” of the video.

Chapter 6

Conclusions

This chapter consists in presenting the conclusion of the thesis work but also offer some possibilities of system improvement and some research opportunities.

6.1 Conclusions

We started our work with the idea that the quality of a video is not only related to the aesthetic of the video but also related to the video's content. And a way to find out the content of the video is through the detection of the objects in the scene. Using the information from the objects detected we assess the quality of the video.

With the work we did, it was possible to evaluate the existing object detection and classification methods and algorithms. And also assess the quality of the video using both the news and the video's information. Another aspect of the assessment of the videos relates to the possibility to verify its ”story”.

After the testing phase, it is important to highlight some of the results.

- The evaluation of the object detection algorithms shows that the content of the video influences which algorithm can represent it the best considering the objects that it detects.

- The methodology used to assess the quality of the videos using the information of the algorithms and the content of the news provided some interesting results. From the results, we can see that combining the algorithms does not necessarily improve the values of the metrics, not even if we discard the confidence level of the detection. However, it does improve the value that represents the detections on a video. And also that different algorithms do not always assess the video the same way. The same can be said about the metrics used. Another conclusion is that classifying more frames does not imply better results.

- Regarding the ”story” of the video, although the results were not the best, we can see that it is indeed possible to align the videos with the structure followed by the news.

Considering that we are trying to assess the videos by correlating it with the information found in news articles, one of the main components is the detection of the objects given by the classifiers. However, the detections of the objects were for the most part either inaccurate or not enough to classify the whole scene. And even if it could identify the objects, the confidence level in which it did was not high. The problems with the detection made it difficult to assess with high reliability the videos.

Regarding the ”story”, one of the problems associated with it is that the news articles do not follow the same structure for every news, and therefore makes it harder to define what structure should be followed by the video. Another problem is that the videos used also lack temporal structured, making it even harder to define its ”story”.

In chapter 1, we have a section with contributions 1.3 that we wanted to achieve in our work and we achieved all of them.

The work done considers the semantic properties of the videos and provides some evaluation on the existing methods and algorithms that assess the video semantics, namely DNN, CNN (TensorFlow), YOLO and SVM. The assessment of the quality of the video was also achieved using semantic information.

6.2 Future Work and Research Opportunities

The main focus of this thesis was related to the investigation regarding the topic of semantic quality and the evaluation of the object detection and classification algorithms. Although that was achieved in this thesis, we believe that there are some aspects that could be improved either by adding functionalities or in terms of research enrichment.

One of the problems of the assessment can be explained by the simplicity of the metrics used. More sophisticated metrics would consider more corner cases and therefore provide the possibility of better assessing the video.

Another point worth improving is related to the matches done between the objects detected on the videos and the terms found in the news articles. As how it is done now, the match is done visually, which is very subjective and not very reliable. A possibility is to create an ontology of the terms, specifying what are the possible words related to a certain term and given these words, match it with the objects detected. Another possibility is to use the family of the words. Given a certain term, the object detected that could match with it would have to be part of the family of the term.

For the ”story” of the video, another way of aligning both the news and the videos can be achieved by the presence of certain character name in the news and the same character in the video. Using this information, the alignment is made more cleary, since we can identify exactly where in the written document, the video is referring to.

The system as it is now, requires that the detection of the objects must be done separately. Each classifier runs its own program in order to classify the objects, which leads to unnecessary loss of time. One of the suggestion is to make this process automatic by creating a tool that given the video as an argument, each classifier would classify the video and display the objects that were detected along with its confidence level.

One of the points discussed in this thesis was the ”story” of the video. The way we found to address this matter was to used each paragraph of the news article and compare it with each frame of the video. This method has some obvious flaws since the news articles does not follow the same structure when it comes to addressing a certain matter and there is also not a clear definition of how to part the video. Therefore, the synchronization between the news and the video can be a good bet in term of research.

The possibility of writing and submitting a paper of the work developed in a conference is being considered.

Bibliography

[1] *A Brief History of CNNs in Image Segmentation: From R-CNN to Mask R-CNN*. https://blog.athelas.com/a-brief-history-of-cnns-in-image-segmentation-from-r-cnn-to-mask-r-cnn-34ea83205de4. Accessed em: 06-02-2018.

[2] L. Bai, S. Lao, G. J. Jones, and A. F. Smeaton. "Video semantic content analysis based on ontology." In: *Machine Vision and Image Processing Conference, 2007. IMVIP 2007. International*. IEEE. 2007, pp. 117–124.

[3] *Beautiful Soup Documentation*. Accessed em: 15-05-2018. URL: https://www.crummy.com/software/BeautifulSoup/bs4/doc/.

[4] M. Chen, S.-C. Chen, M.-L. Shyu, and K. Wickramaratna. "Semantic event detection via multimodal data mining." In: *IEEE Signal Processing Magazine* 23.2 (2006), pp. 38–46.

[5] S. Chikkerur, V. Sundaram, M. Reisslein, and L. J. Karam. "Objective video quality assessment methods: A classification, review, and performance comparison." In: *IEEE transactions on broadcasting* 57.2 (2011), pp. 165–182.

[6] D. CireşAn, U. Meier, J. Masci, and J. Schmidhuber. "Multi-column deep neural network for traffic sign classification." In: *Neural networks* 32 (2012), pp. 333–338.

[7] *CLOUD VIDEO INTELLIGENCE*. Accessed em: 05-02-2018. URL: https://cloud.google.com/video-intelligence/#demo.

[8] *Cognitus*. http://cognitus-h2020.eu/. Accessed em: 10-01-2018.

[9] *Convolutional Neural Network*. http://ufldl.stanford.edu/tutorial/supervised/ConvolutionalNeuralNetwork/. Accessed em: 26-01-2018.

[10] C. Cortes and V. Vapnik. "Support-vector networks." In: *Machine learning* 20.3 (1995), pp. 273–297.

[11] *Deep Representation Learning with Target Coding*. Accessed em: 05-02-2018. URL: http://personal.ie.cuhk.edu.hk/~ccloy/project_target_code/index.html.

[12] A. Ekin, A. M. Tekalp, and R. Mehrotra. "Automatic soccer video analysis and summarization." In: *IEEE Transactions on Image processing* 12.7 (2003), pp. 796–807.

[13] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman. "The pascal visual object classes (voc) challenge." In: *International journal of computer vision* 88.2 (2010), pp. 303–338.

[14] K. Gauen, R. Dailey, J. Laiman, Y. Zi, N. Asokan, Y.-H. Lu, G. K. Thiruvathukal, M.-L. Shyu, and S.-C. Chen. "Comparison of Visual Datasets for Machine Learning." In: (2017).

[15] R. Girshick. "Fast r-cnn." In: *arXiv preprint arXiv:1504.08083* (2015).

[16] H. Hosseini, B. Xiao, and R. Poovendran. "Deceiving google's cloud video intelligence api built for summarizing videos." In: *arXiv preprint* (2017).

[17] T Huang. "Computer vision: Evolution and promise." In: (1996). Accessed em: 10-02-2018. URL: http://cds.cern.ch/record/400313/files/p21.pdf.

[18] A. Karpathy, G. Toderici, S. Shetty, T. Leung, R. Sukthankar, and L. Fei-Fei. "Large-scale video classification with convolutional neural networks." In: *Proceedings of the IEEE conference on Computer Vision and Pattern Recognition*. 2014, pp. 1725–1732.

[19] S. Khalid, T. Khalil, and S. Nasreen. "A survey of feature selection and feature extraction techniques in machine learning." In: *Science and Information Conference (SAI), 2014*. IEEE. 2014, pp. 372–378.

[20] P. Kortum and M. Sullivan. "The effect of content desirability on subjective video quality ratings." In: *Human factors* 52.1 (2010), pp. 105–118.

[21] A. Krizhevsky, I. Sutskever, and G. E. Hinton. "Imagenet classification with deep convolutional neural networks." In: *Advances in neural information processing systems*. 2012, pp. 1097–1105.

[22] R. Leonardi, P. Migliorati, and M. Prandini. "Semantic indexing of soccer audio-visual sequences: a multimodal approach based on controlled Markov chains." In: *IEEE Transactions on Circuits and Systems for Video Technology* 14.5 (2004), pp. 634–643.

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

[24] *Load Caffe framework models*. Accessed em: 10-03-2018. URL: https://docs.opencv.org/3.3.0/d5/de7/tutorial_dnn_googlenet.html.

[25] *Machine Learning :: Cosine Similarity for Vector Space Models (Part III)*. Accessed em: 21-05-2018. URL: http://blog.christianperone.com/2013/09/machine-learning-cosine-similarity-for-vector-space-models-part-iii/.

[26] P. C. P. Martins. "Sistema para Avaliação Semiautomática de Vídeo." Master's thesis. NOVA University of Lisbon, 2016.

[27] P. C. P. Martins. “Sistema para Avaliação Semiautomática de Vídeo.” Master’s thesis. NOVA University of Lisbon, 2017.

[28] P. Michel and R. El Kaliouby. “Real time facial expression recognition in video using support vector machines.” In: *Proceedings of the 5th international conference on Multimodal interfaces*. ACM. 2003, pp. 258–264.

[29] H. Motoda and H. Liu. “Feature selection, extraction and construction.” In: *Communication of IICM (Institute of Information and Computing Machinery, Taiwan) Vol* 5 (2002), pp. 67–72.

[30] *Neural Network*. Accessed em: 02-04-2018. URL: https://deepai.org/machine-learning-glossary-and-terms/neural-network.

[31] *Object Detection*. http://slazebni.cs.illinois.edu/spring17/lec07_detection.pdf. Accessed em: 02-02-2018.

[32] *Object Tracking using OpenCV (C++/Python)*. https://www.learnopencv.com/object-tracking-using-opencv-cpp-python/. Accessed em: 06-02-2018.

[33] *OpenCV*. Accessed em: 15-05-2018. URL: https://docs.opencv.org/3.4.0/d4/d72/classcv_1_1BOWKMeansTrainer.html#details.

[34] P. Over, J. Fiscus, G. Sanders, D. Joy, M. Michel, G. Awad, A. Smeaton, W. Kraaij, and G. Quénot. “Trecvid 2014–an overview of the goals, tasks, data, evaluation mechanisms and metrics.” In: *Proceedings of TRECVID*. 2014, p. 52.

[35] *PySceneDetect*. Accessed em: 21-05-2018. URL: https://pyscenedetect.readthedocs.io/en/latest/.

[36] T. Quirino, Z. Xie, M.-L. Shyu, S.-C. Chen, and L. Chang. “Collateral representative subspace projection modeling for supervised classification.” In: *Tools with Artificial Intelligence, 2006. ICTAI’06. 18th IEEE International Conference on*. IEEE. 2006, pp. 98–105.

[37] J. Redmon and A. Farhadi. “YOLO9000: better, faster, stronger.” In: *arXiv preprint* 1612 (2016).

[38] J. Redmon and A. Farhadi. “Yolov3: An incremental improvement.” In: *arXiv preprint arXiv:1804.02767* (2018).

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

[40] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, et al. “Imagenet large scale visual recognition challenge.” In: *International Journal of Computer Vision* 115.3 (2015), pp. 211–252.

[41] *scikit-learn*. Accessed em: 03-06-2018. URL: https://scikit-learn.org/stable/.

[42] K. Seshadrinathan, R. Soundararajan, A. C. Bovik, and L. K. Cormack. "Study of subjective and objective quality assessment of video." In: *IEEE transactions on Image Processing* 19.6 (2010), pp. 1427–1441.

[43] M. Shahid, A. Rossholm, B. Lövström, and H.-J. Zepernick. "No-reference image and video quality assessment: a classification and review of recent approaches." In: *EURASIP Journal on image and Video Processing* 2014.1 (2014), p. 40.

[44] M. Shahid, S. Khatibi, and Y. Tuemay. "Popularity index through video semantic quality assessment." In: *Signal and Information Processing (ChinaSIP), 2014 IEEE China Summit & International Conference on*. IEEE. 2014, pp. 344–348.

[45] V. Sharma, S. Rai, and A. Dev. "A comprehensive study of artificial neural networks." In: *International Journal of Advanced research in computer science and software engineering* 2.10 (2012).

[46] M.-L. Shyu, Z. Xie, M. Chen, and S.-C. Chen. "Video semantic event/concept detection using a subspace-based multimedia data mining framework." In: *IEEE Transactions on Multimedia* 10.2 (2008), pp. 252–259.

[47] A. F. Smeaton, P. Over, and W. Kraaij. "Evaluation campaigns and TRECVid." In: *Proceedings of the 8th ACM international workshop on Multimedia information retrieval*. ACM. 2006, pp. 321–330.

[48] *Statistics for Machine Learning by Pratap Dangeti*. Accessed em: 16-05-2018. URL: https://www.oreilly.com/library/view/statistics-for-machine/9781788295758/eb9cd609-e44a-40a2-9c3a-f16fc4f5289a.xhtml.

[49] M. Tapaswi. "Story Understanding through Semantic Analysis and Automatic Alignment of Text and Video." In: (2016).

[50] *TensorFlow For Poets*. Accessed em: 20-03-2018. URL: https://codelabs.developers.google.com/codelabs/tensorflow-for-poets/.

[51] *Tf-idf*. Accessed em: 20-05-2018. URL: http://www.tfidf.com/.

[52] S. Tong and E. Chang. "Support vector machine active learning for image retrieval." In: *Proceedings of the ninth ACM international conference on Multimedia*. ACM. 2001, pp. 107–118.

[53] *Understanding Support Vector Machine algorithm from examples (along with code)*. https://www.analyticsvidhya.com/blog/2017/09/understaing-support-vector-machine-example-code/. Accessed em: 23-01-2018.

[54] Y. Wang. "Survey of objective video quality measurements." In: (2006). Accessed em: 08-01-2018. URL: https://digitalcommons.wpi.edu/cgi/viewcontent.cgi?referer=https://scholar.google.pt/&httpsredir=1&article=1043&context=computerscience-pubs.

[55] Z. Wang, H. R. Sheikh, A. C. Bovik, et al. "Objective video quality assessment." In: *The handbook of video databases: design and applications* 41 (2003), pp. 1041–1078.

[56] A. A. Webster, C. T. Jones, M. H. Pinson, S. D. Voran, and S. Wolf. "An objective video quality assessment system based on human perception." In: *SPIE human vision, visual processing, and digital display IV*. Vol. 1913. 1993, pp. 15–26.

[57] *Welcome to icrawler*. Accessed em: 09-03-2018. URL: https://icrawler.readthedocs.io/en/latest/.

[58] *WolframMathWorld*. Accessed em: 21-05-2018. URL: http://mathworld.wolfram.com/EuclideanMetric.html.

[59] C. Wu, Y.-F. Ma, H.-J. Zhan, and Y.-Z. Zhong. "Events recognition by semantic inference for sports video." In: *Multimedia and Expo, 2002. ICME'02. Proceedings. 2002 IEEE International Conference on*. Vol. 1. IEEE. 2002, pp. 805–808.

[60] L. Xie, S.-F. Chang, A. Divakaran, and H. Sun. "Unsupervised discovery of multilevel statistical video structures using hierarchical hidden Markov models." In: *Multimedia and Expo, 2003. ICME'03. Proceedings. 2003 International Conference on*. Vol. 3. IEEE. 2003, pp. III–29.

[61] *YOLO: Real-Time Object Detection*. https://pjreddie.com/darknet/yolo/. Accessed em: 01-02-2018.

Appendix A

1000 ILSVRC2012 image classes

1 tench, Tinca tinca

...

41 common iguana, iguana, Iguana iguana

...

81 centipede

...

121 rock crab, Cancer irroratus

...

161 Rhodesian ridgeback

...

201 Tibetan terrier, chrysanthemum dog

...

241 EntleBucher

...

281 tiger cat

...

321 monarch, monarch butterfly, milkweed butterfly, Danaus plexippus

...

361 three-toed sloth, ai, Bradypus tridactylus

...

401 airship, dirigible

...

441 binder, ring-binder

...

481 cellular telephone, cellular phone, cellphone, cell, mobile phone

...

521 dial telephone, dial phone

...

561 garbage truck, dustcart

...

601 jersey, T-shirt, tee shirt

...

641 microphone, mike

...

681 packet

...

721 pole

...

761 poncho

...

801 spider web, spider's web

...

841 thimble

...

881 washbasin, handbasin, washbowl, lavabo, wash-hand basin

...

921 broccoli

...

961 seashore, coast, seacoast, sea-coast

...

1000 stage

Appendix B

Appendix - Detection in Videos of Events Test

B.1 Results from test - Detection in Videos of Events

The tables B.1, B.2, B.3 shows the objects detected along with its confidence level from a football related video, being one of the events chosen for this test. The graph representing the hit rates of each classifier can be seen in figure B.1.

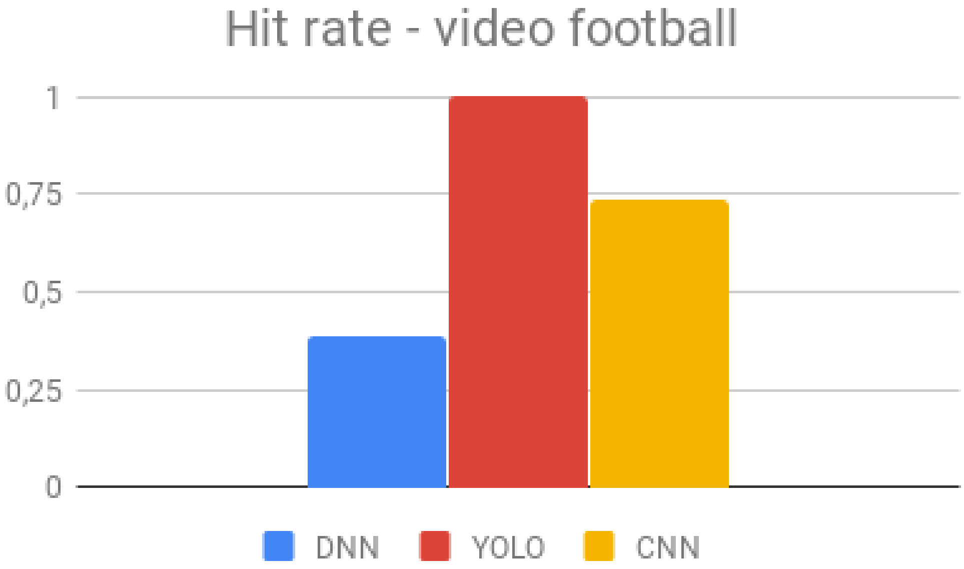


Figure B.1: Detection in Videos of Events - hi rate video football

The tables B.4, B.5, B.6 shows the objects detected and its confidence level from a eurovision contest video. In figure B.2 is it possible to see the graph representing the hit rates of each classifier.

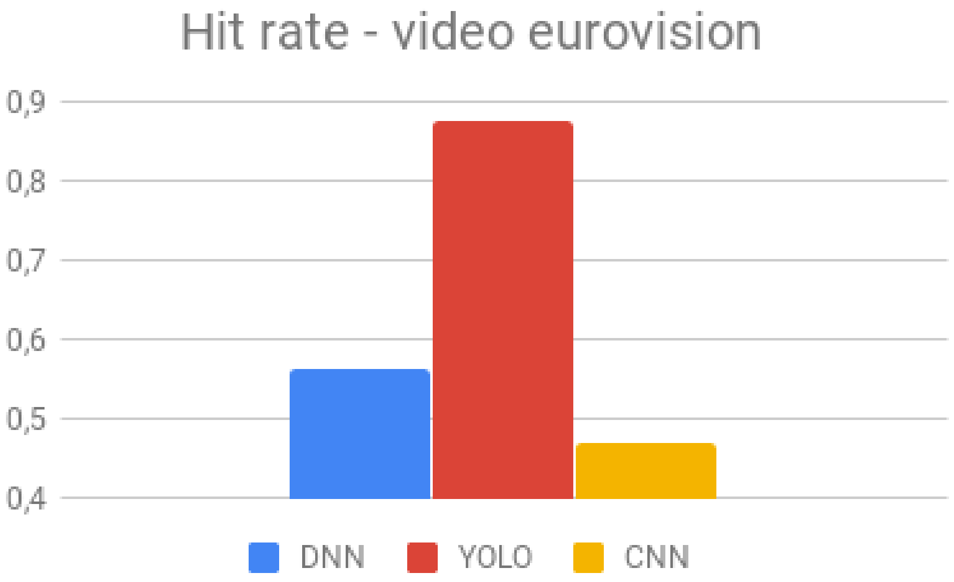


Figure B.2: Detection in Videos of Events - hi rate video eurovision

Frames:	Name:	Probability:
1	racket, racquet	59.61%
2	parachute chute	41.82%
3	ballplayer, baseball player	46.77%
4	ballplayer, baseball player	79.61%
5	ant, emmet, pismire	9.25%
6	scoreboard	23.41%
7	ballplayer, baseball player	27.35%
8	ping-pong ball	63.44%
9	racket, racquet	34.32%
10	ballplayer, baseball player	80.17%
11	ballplayer, baseball player	20.43%
12	ballplayer, baseball player	95.35%
13	ping-pong ball	39.47%
14	tennis ball	33.56%
15	pool table, billiard table, snooker table	33.70%
16	ballplayer, baseball player	85.70%
17	ballplayer, baseball player	98.50%
18	ballplayer, baseball player	87.92%
19	ballplayer, baseball player	92.93%
20	ballplayer, baseball player	99.20%
21	ballplayer, baseball player	95.81%
22	ballplayer, baseball player	45.79%
23	military uniform	13.90%
24	bathing cap, swimming cap	35.23%
25	cloak	22.14%
26	tennis ball	64.42%
27	ballplayer, baseball player	43.14%
28	ballplayer, baseball player	92.37%
29	bathing cap, swimming cap	17.83%
30	ballplayer, baseball player	46.65%
31	hair slide	21.87%
32	web site, website internet site, site	20.26%
33	volleyball	99.72%
34	punching bag, punch bag, punching ball punchball	31.58%
35	jersey, T-shirt, tee shirt	36.89%
36	neck brace	13.86%
37	racket, racquet	46.89%
38	sports car, sport car	27.68%
39	hair slide	18.36%
40	web site, website internet site, site	21.95%
41	maraca	18.60%
42	neck brace	15.31%
43	ballplayer, baseball player	31.02%
44	ballplayer, baseball player	19.53%
45	ballplayer, baseball player	60.11%
46	jersey, T-shirt, tee shirt	27.93%
47	neck brace	12.29%
48	horizontal bar, high bar	20.08%
49	bubble	10.22%

Table B.1: DNN detection - football

Frames:	Name:	Probability:
1	person	100.00%
2	person	99.00%
3	person	100.00%
4	person	100.00%
5	person	95.00%
6	person	98.00%
7	person	97.00%
8	person	97.00%
9	person	95.00%
10	person	99.00%
11	person	99.00%
12	person	100.00%
13	person	99.00%
14	person	99.00%
15	person	99.00%
16	person	100.00%
17	person	100.00%
18	person	99.00%
19	person	98.00%
20	person	100.00%
21	person	100.00%
22	person	100.00%
23	person	99.00%
24	person	99.00%
25	person	87.00%
26	person	88.00%
27	person	100.00%
28	person	100.00%
29	person	91.00%
30	person	99.00%
31	person	88.00%
32	person	85.00%
33	person	90.00%
34	person	93.00%
35	person	96.00%
36	person	97.00%
37	person	100.00%
38	person	99.00%
39	person	84.00%
40	person	92.00%
41	person	95.00%
42	person	97.00%
43	person	99.00%
44	person	99.00%
45	person	100.00%
46	person	100.00%
47	person	96.00%
48	person	96.00%
49	person	99.00%

Table B.2: YOLO detection - football

Frames:	Name:	Probability:
1	soccer_field	99.00%
2	-	-
3	soccer_field	99.00%
4	soccer_field	95.00%
5	goal	83.00%
6	goal	99.00%
7	ronaldo	99.00%
8	ronaldo	99.00%
9	soccer_field	99.00%
10	soccer_field	99.00%
11	soccer_field	99.00%
12	soccer_field	99.00%
13	scoreboard	86.00%
14	-	-
15	street_concert	85.00%
16	soccer_field	99.00%
17	soccer_field	99.00%
18	soccer_field	99.00%
19	soccer_field	99.00%
20	soccer_field	99.00%
21	soccer_field	99.00%
22	soccer_field	99.00%
23	ronaldo	99.00%
24	ronaldo	99.00%
25	ronaldo	99.00%
26	soccer_field	51.00%
27	goal	99.00%
28	goal	98.00%
29	ronaldo	66.00%
30	soccer_field	98.00%
31	soccer_field	99.00%
32	soccer_field	99.00%
33	goal	91.00%
34	street_concert	69.00%
35	messi	99.00%
36	ronaldo	88.00%
37	scoreboard	79.00%
38	soccer_field	94.00%
39	soccer_field	99.00%
40	soccer_field	98.00%
41	messi	70.00%
42	street_concert	71.00%
43	soccer_field	99.00%
44	soccer_field	99.00%
45	soccer_field	99.00%
46	messi	94.00%
47	ronaldo	66.00%
48	goal	58.00%
49	soccer_field	99.00%

Table B.3: CNN detection - football

Frames:	Name:	Probability:
1	stage	90.66%
2	microphone mike	32.21%
3	microphone mike	76.12%
4	microphone mike	72.13%
5	spotlight spot	57.25%
6	spotlight spot	13.83%
7	stage	85.57%
8	stage	52.15%
9	stage	25.02%
10	fountain	47.32%
11	stage	55.54%
12	stage	13.38%
13	gown	11.60%
14	stage	35.12%
15	stage	9.39%
16	academic gown , judge's robe	6.98%
17	stage	20.45%
18	jellyfish	33.47%
19	stage	28.64%
20	stage	79.11%
21	stage	55.90%
22	drumstick	23.59%
23	ping-pong ball	35.04%
24	stage	98.10%
25	stage	92.43%
26	stage	61.37%
27	stage	98.41%
28	stage	93.70%
29	stage	85.50%
30	stage	64.06%
31	web site, website internet site, site	17.24%
32	web site, website internet site, site	17.27%

Table B.4: DNN detection - eurovision

Frames:	Name:	Probability:
1	person	99.00%
2	person	100.00%
3	person	100.00%
4	-	-
5	person	74.00%
6	person	90.00%
7	-	-
8	person	97.00%
9	person	96.00%
10	person	100.00%
11	person	75.00%
12	person	80.00%
13	person	89.00%
14	person	85.00%
15	person	83.00%
16	person	92.00%
17	person	95.00%
18	person	91.00%
19	person	90.00%
20	person	93.00%
21	person	84.00%
22	person	86.00%
23	person	100.00%
24	person	99.00%
25	person	98.00%
26	person	96.00%
27	person	90.00%
28	person	96.00%
29	person	89.00%
30	person	97.00%
31	person	100.00%
32	person	94.00%

Table B.5: YOLO detection - eurovision

Frames:	Name:	Probability:
1	stage	77.00%
2	ronaldo	98.00%
3	messi	70.00%
4	ronaldo	94.00%
5	stage	63.00%
6	stage	99.00%
7	stage	99.00%
8	stage	76.00%
9	stage	99.00%
10	stage	99.00%
11	stage	99.00%
12	daisy	89.00%
13	stage	83.00%
14	daisy	76.00%
15	street_concert / daisy	43.00% / 35.00%
16	street_concert	47.00%
17	ronaldo	71.00%
18	street_concert	92.00%
19	daisy	68.00%
20	street_concert	76.00%
21	ronaldo	97.00%
22	messi / stage	60.00% / 30%
23	stage	68.00%
24	street_concert / ronaldo	58.00% / 38.00%
25	ronaldo	98.00%
26	stage / ronaldo	49.00% / 46.00%
27	stage	99.00%
28	stage	99.00%
29	stage	99.00%
30	ronaldo	99.00%
31	-	-
32	-	-

Table B.6: CNN detection - eurovision

Appendix

C

Appendix - Video Assessment Tests

C.1 Results from DNN classifier

C.2 Results from CNN classifier

C.3 Results from YOLO classifier

C.4 Results from joined classifiers

C.5 Results from joined classifiers - 100%

C.6 Results from DNN classifier - more frames

C.7 Results from CNN classifier - more frames

C.8 Results from YOLO classifier - more frames

C.9 Results from joined classifiers - more frames

C.10 Results from joined classifiers - 100%

words	w_values	detection	c_values	sqr of words values	sqr detection values	cosine similarity	euclidean distance
performance	0,167937863	stage	0,11438902	0,2241531654	0,2557816246	**0,4437760658**	**0,05354884299**
stage	0,01460709537	stage	0,11438902				**0,09978192463**
music	0,009796375341	stage	0,11438902				**0,1045926447**
microphone	0,01434469246	stage	0,11438902				**0,1000443275**
show	0,01574417466	stage	0,11438902				**0,09864484534**
dress	0,01539430411	-	0				**0,01539430411**
song	0,05709887342	-	0				**0,05709887342**
photos	0,06297669862	-	0				**0,06297669862**
ferrell	0,04618291232	-	0				**0,04618291232**
eurovision	0,03292281856	-	0				**0,03292281856**
she	0,03148834931	-	0				**0,03148834931**
film	0,02728990274	-	0				**0,02728990274**
singer	0,02662514869	-	0				**0,02662514869**
contest	0,02632775873	-	0				**0,02632775873**
parody	0,02554055	-	0				**0,02554055**
gaza	0,02449093835	-	0				**0,02449093835**
prince	0,02344132671	-	0				**0,02344132671**
report	0,02029249178	-	0				**0,02029249178**
star	0,01994262123	-	0				**0,01994262123**
video	0,01952277657	-	0				**0,01952277657**
netta	0,01877638607	-	0				**0,01877638607**
israel	0,01830989201	-	0				**0,01830989201**
surie	0,01644391575	-	0				**0,01644391575**
message	0,01574417466	-	0				**0,01574417466**
israeli	0,01574417466	-	0				**0,01574417466**
ryan	0,01574417466	-	0				**0,01574417466**
media	0,01539430411	-	0				**0,01539430411**
crowd	0,01539430411	-	0				**0,01539430411**
countries	0,01521936883	-	0				**0,01521936883**
man	0,01504443356	-	0				**0,01504443356**
palestinian	0,01504443356	-	0				**0,01504443356**
visit	0,01504443356	-	0				**0,01504443356**
flag	0,01434469246	-	0				**0,01434469246**
band	0,01399482192	-	0				**0,01399482192**
protester	0,01399482192	-	0				**0,01399482192**
home	0,01364495137	-	0				**0,01364495137**
violence	0,01224546918	-	0				**0,01224546918**
world	0,01049611644	-	0				**0,01049611644**
barzilai	0,009796375341	-	0				**0,009796375341**
chinese	0,008047022602	-	0				**0,008047022602**
netherlands	0,006997410958	-	0				**0,006997410958**
secutity	0,006297669862	-	0				**0,006297669862**
europe	0,005597928766	-	0				**0,005597928766**
lisbon	0,005422993492	-	0				**0,005422993492**
albania	0,005248058218	-	0				**0,005248058218**
						total:	**0,2540671298**

Table C.1: Video 2 - DNN classification

words	w_values	detection	c_values	sqr of words values	sqr detection values	cosine similarity	euclidean distance
performance	0,167937863	-	0	0,2241531654	0,2820710667	**0,1904242877**	**0,167937863**
stage	0,01460709537	-	0				**0,01460709537**
music	0,009796375341	-	0				**0,009796375341**
microphone	0,01434469246	-	0				**0,01434469246**
show	0,01574417466	-	0				**0,01574417466**
dress	0,01539430411	groom/bridegroom	0,1410355333				**0,1256412292**
song	0,05709887342	-	0				**0,05709887342**
photos	0,06297669862	-	0				**0,06297669862**
ferrell	0,04618291232	-	0				**0,04618291232**
eurovision	0,03292281856	-	0				**0,03292281856**
she	0,03148834931	groom/bridegroom	0,1410355333				**0,109547184**
film	0,02728990274	-	0				**0,02728990274**
singer	0,02662514869	-	0				**0,02662514869**
contest	0,02632775873	-	0				**0,02632775873**
parody	0,02554055	-	0				**0,02554055**
gaza	0,02449093835	-	0				**0,02449093835**
prince	0,02344132671	groom/bridegroom	0,1410355333				**0,1175942066**
report	0,02029249178	-	0				**0,02029249178**
star	0,01994262123	-	0				**0,01994262123**
video	0,01952277657	-	0				**0,01952277657**
netta	0,01877638607	-	0				**0,01877638607**
israel	0,01830989201	-	0				**0,01830989201**
surie	0,01644391575	-	0				**0,01644391575**
message	0,01574417466	-	0				**0,01574417466**
israeli	0,01574417466	-	0				**0,01574417466**
ryan	0,01574417466	-	0				**0,01574417466**
media	0,01539430411	-	0				**0,01539430411**
crowd	0,01539430411	-	0				**0,01539430411**
countries	0,01521936883	-	0				**0,01521936883**
man	0,01504443356	groom/bridegroom	0,1410355333				**0,1259910998**
palestinian	0,01504443356	-	0				**0,01504443356**
visit	0,01504443356	-	0				**0,01504443356**
flag	0,01434469246	-	0				**0,01434469246**
band	0,01399482192	-	0				**0,01399482192**
protester	0,01399482192	-	0				**0,01399482192**
home	0,01364495137	-	0				**0,01364495137**
violence	0,01224546918	-	0				**0,01224546918**
world	0,01049611644	-	0				**0,01049611644**
barzilai	0,009796375341	-	0				**0,009796375341**
chinese	0,008047022602	-	0				**0,008047022602**
netherlands	0,006997410958	-	0				**0,006997410958**
secutity	0,006297669862	-	0				**0,006297669862**
europe	0,005597928766	-	0				**0,005597928766**
lisbon	0,005422993492	-	0				**0,005422993492**
albania	0,005248058218	-	0				**0,005248058218**
						total:	0,324803232

Table C.2: Video 3 - DNN classification

words	w_values	detection	c_values	sqr of words values	sqr detection values	cosine similarity	euclidean distance
performance	0,167937863	stage/stret_concert	0,304919678	0,2241531654	0,705038516	**0,4728443308**	**0,136981815**
stage	0,01460709537	stage/stret_concert	0,304919678				**0,2903125826**
music	0,009796375341	stage/stret_concert	0,304919678				**0,2951233027**
microphone	0,01434469246	stage/stret_concert	0,304919678				**0,2905749855**
show	0,01574417466	stage/stret_concert	0,304919678				**0,2891755033**
singer	0,02662514869	musician	0,09001028733				**0,06338513864**
netta	0,01877638607	musician	0,09001028733				**0,07123390126**
surie	0,01644391575	musician	0,09001028733				**0,07356637158**
man	0,01504443356	messi	0,088846516				**0,07380208244**
song	0,05709887342	-	0				**0,05709887342**
photos	0,06297669862	-	0				**0,06297669862**
ferrell	0,04618291232	-	0				**0,04618291232**
eurovision	0,03292281856	-	0				**0,03292281856**
she	0,03148834931	-	0				**0,03148834931**
film	0,02728990274	-	0				**0,02728990274**
contest	0,02632775873	-	0				**0,02632775873**
parody	0,02554055	-	0				**0,02554055**
gaza	0,02449093835	-	0				**0,02449093835**
prince	0,02344132671	-	0				**0,02344132671**
report	0,02029249178	-	0				**0,02029249178**
star	0,01994262123		0				**0,01994262123**
video	0,01952277657	-	0				**0,01952277657**
israel	0,01830989201		0				**0,01830989201**
message	0,01574417466	-	0				**0,01574417466**
israeli	0,01574417466	-	0				**0,01574417466**
ryan	0,01574417466	-	0				**0,01574417466**
media	0,01539430411	-	0				**0,01539430411**
crowd	0,01539430411	-	0				**0,01539430411**
dress	0,01539430411	-	0				**0,01539430411**
countries	0,01521936883		0				**0,01521936883**
palestinian	0,01504443356	-	0				**0,01504443356**
visit	0,01504443356	-	0				**0,01504443356**
flag	0,01434469246	-	0				**0,01434469246**
band	0,01399482192	-	0				**0,01399482192**
protester	0,01399482192	-	0				**0,01399482192**
home	0,01364495137	-	0				**0,01364495137**
violence	0,01224546918	-	0				**0,01224546918**
world	0,01049611644	-	0				**0,01049611644**
barzilai	0,009796375341	-	0				**0,009796375341**
chinese	0,008047022602	-	0				**0,008047022602**
netherlands	0,006997410958	-	0				**0,006997410958**
secutity	0,006297669862	-	0				**0,006297669862**
europe	0,005597928766	-	0				**0,005597928766**
lisbon	0,005422993492	-	0				**0,005422993492**
albania	0,005248058218	-	0				**0,005248058218**
						total:	**0,6305818987**

Table C.3: Video 2 - CNN classification

words	w_values	detection	c_values	sqr of words values	sqr detection values	cosine similarity	euclidean distance
performance	0,167937863	street_concert	0,5992723475	0,2241531654	1,364457588	**0,465843023**	**0,4313344845**
stage	0,01460709537	street_concert	0,5992723475				**0,5846652521**
music	0,009796375341	street_concert	0,5992723475				**0,5894759722**
microphone	0,01434469246	street_concert	0,5992723475				**0,584927655**
show	0,01574417466	street_concert	0,5992723475				**0,5835281728**
singer	0,02662514869	musician	0,148445025				**0,1218198763**
netta	0,01877638607	musician	0,148445025				**0,1296686389**
surie	0,01644391575	musician	0,148445025				**0,1320011092**
man	0,01504443356	-	0				**0,01504443356**
song	0,05709887342	-	0				**0,05709887342**
photos	0,06297669862	-	0				**0,06297669862**
ferrell	0,04618291232	-	0				**0,04618291232**
eurovision	0,03292281856	-	0				**0,03292281856**
she	0,03148834931	-	0				**0,03148834931**
film	0,02728990274	-	0				**0,02728990274**
contest	0,02632775873	-	0				**0,02632775873**
parody	0,02554055	-	0				**0,02554055**
gaza	0,02449093835	-	0				**0,02449093835**
prince	0,02344132671	-	0				**0,02344132671**
report	0,02029249178	-	0				**0,02029249178**
star	0,01994262123		0				**0,01994262123**
video	0,01952277657	-	0				**0,01952277657**
israel	0,01830989201		0				**0,01830989201**
message	0,01574417466	-	0				**0,01574417466**
israeli	0,01574417466	-	0				**0,01574417466**
ryan	0,01574417466	-	0				**0,01574417466**
media	0,01539430411	-	0				**0,01539430411**
crowd	0,01539430411	-	0				**0,01539430411**
dress	0,01539430411	-	0				**0,01539430411**
countries	0,01521936883		0				**0,01521936883**
palestinian	0,01504443356	-	0				**0,01504443356**
visit	0,01504443356	-	0				**0,01504443356**
flag	0,01434469246	-	0				**0,01434469246**
band	0,01399482192	-	0				**0,01399482192**
protester	0,01399482192	-	0				**0,01399482192**
home	0,01364495137	-	0				**0,01364495137**
violence	0,01224546918	-	0				**0,01224546918**
world	0,01049611644	-	0				**0,01049611644**
barzilai	0,009796375341	-	0				**0,009796375341**
chinese	0,008047022602	-	0				**0,008047022602**
netherlands	0,006997410958	-	0				**0,006997410958**
secutity	0,006297669862	-	0				**0,006297669862**
europe	0,005597928766	-	0				**0,005597928766**
lisbon	0,005422993492	-	0				**0,005422993492**
albania	0,005248058218	-	0				**0,005248058218**
						total:	**1,275460046**

Table C.4: Video 3 - CNN classification

words	w_values	detection	c_values	sqr of words values	sqr detection values	cosine similarity	euclidean distance
performance	0,167937863	-	0	0,2241531654	2,00103029	**0,3203793007**	**0,167937863**
stage	0,01460709537	-	0				**0,01460709537**
music	0,009796375341	-	0				**0,009796375341**
microphone	0,01434469246	-	0				**0,01434469246**
show	0,01574417466	-	0				**0,01574417466**
singer	0,02662514869	person	0,6033333333				**0,5767081846**
netta	0,01877638607	person	0,6033333333				**0,5845569473**
surie	0,01644391575	person	0,6033333333				**0,5868894176**
man	0,01504443356	person	0,6033333333				**0,5882888998**
song	0,05709887342	-	0				**0,05709887342**
photos	0,06297669862	-	0				**0,06297669862**
ferrell	0,04618291232	person	0,6033333333				**0,557150421**
eurovision	0,03292281856	-	0				**0,03292281856**
she	0,03148834931	person	0,6033333333				**0,571844984**
film	0,02728990274	-	0				**0,02728990274**
contest	0,02632775873	-	0				**0,02632775873**
parody	0,02554055	-	0				**0,02554055**
gaza	0,02449093835	-	0				**0,02449093835**
prince	0,02344132671	person	0,6033333333				**0,5798920066**
report	0,02029249178	-	0				**0,02029249178**
star	0,01994262123	-	0				**0,01994262123**
video	0,01952277657	-	0				**0,01952277657**
israel	0,01830989201	-	0				**0,01830989201**
message	0,01574417466	-	0				**0,01574417466**
israeli	0,01574417466	-	0				**0,01574417466**
ryan	0,01574417466	person	0,6033333333				**0,5875891587**
media	0,01539430411	-	0				**0,01539430411**
crowd	0,01539430411	person	0,6033333333				**0,5879390292**
dress	0,01539430411	-	0				**0,01539430411**
countries	0,01521936883	-	0				**0,01521936883**
palestinian	0,01504443356	person	0,6033333333				**0,5882888998**
visit	0,01504443356	-	0				**0,01504443356**
flag	0,01434469246	-	0				**0,01434469246**
band	0,01399482192	-	0				**0,01399482192**
protester	0,01399482192	person	0,6033333333				**0,5893385114**
home	0,01364495137	-	0				**0,01364495137**
violence	0,01224546918	-	0				**0,01224546918**
world	0,01049611644	-	0				**0,01049611644**
barzilai	0,009796375341	-	0				**0,009796375341**
chinese	0,008047022602	-	0				**0,008047022602**
netherlands	0,006997410958	-	0				**0,006997410958**
secutity	0,006297669862	-	0				**0,006297669862**
europe	0,005597928766	-	0				**0,005597928766**
lisbon	0,005422993492	-	0				**0,005422993492**
albania	0,005248058218	-	0				**0,005248058218**
						total:	1,849673065

Table C.5: Video 2 - YOLO classification

words	w_values	detection	c_values	sqr of words values	sqr detection values	cosine similarity	euclidean distance
performance	0,167937863	-	0	0,2241531654	1,923642378	**0,3203793007**	**0,167937863**
stage	0,01460709537	-	0				**0,01460709537**
music	0,009796375341	-	0				**0,009796375341**
microphone	0,01434469246	-	0				**0,01434469246**
show	0,01574417466	-	0				**0,01574417466**
singer	0,02662514869	person	0,58				**0,5533748513**
netta	0,01877638607	person	0,58				**0,5612236139**
surie	0,01644391575	person	0,58				**0,5635560842**
man	0,01504443356	person	0,58				**0,5649555664**
song	0,05709887342	-	0				**0,05709887342**
photos	0,06297669862	-	0				**0,06297669862**
ferrell	0,04618291232	person	0,58				**0,5338170877**
eurovision	0,03292281856	-	0				**0,03292281856**
she	0,03148834931	person	0,58				**0,5485116507**
film	0,02728990274	-	0				**0,02728990274**
contest	0,02632775873	-	0				**0,02632775873**
parody	0,02554055	-	0				**0,02554055**
gaza	0,02449093835	-	0				**0,02449093835**
prince	0,02344132671	person	0,58				**0,5565586733**
report	0,02029249178	-	0				**0,02029249178**
star	0,01994262123	-	0				**0,01994262123**
video	0,01952277657	-	0				**0,01952277657**
israel	0,01830989201	-	0				**0,01830989201**
message	0,01574417466	-	0				**0,01574417466**
israeli	0,01574417466	-	0				**0,01574417466**
ryan	0,01574417466	person	0,58				**0,5642558253**
media	0,01539430411	-	0				**0,01539430411**
crowd	0,01539430411	person	0,58				**0,5646056959**
dress	0,01539430411	-	0				**0,01539430411**
countries	0,01521936883	-	0				**0,01521936883**
palestinian	0,01504443356	person	0,58				**0,5649555664**
visit	0,01504443356	-	0				**0,01504443356**
flag	0,01434469246	-	0				**0,01434469246**
band	0,01399482192	-	0				**0,01399482192**
protester	0,01399482192	person	0,58				**0,5660051781**
home	0,01364495137	-	0				**0,01364495137**
violence	0,01224546918	-	0				**0,01224546918**
world	0,01049611644	-	0				**0,01049611644**
barzilai	0,009796375341	-	0				**0,009796375341**
chinese	0,008047022602	-	0				**0,008047022602**
netherlands	0,006997410958	-	0				**0,006997410958**
secutity	0,006297669862	-	0				**0,006297669862**
europe	0,005597928766	-	0				**0,005597928766**
lisbon	0,005422993492	-	0				**0,005422993492**
albania	0,005248058218	-	0				**0,005248058218**
						total:	**1,776394103**

Table C.6: Video 3 - YOLO classification

words	w_values	detection	c_values	sqr of words values	sqr detection values	cosine similarity	euclidean distance
performance	0,167937863	stage/street_concert	0,419308698	0,2241531654	2,312675025	**0,470438735**	**0,251370835**
stage	0,01460709537	stage/street_concert	0,419308698				**0,4047016026**
music	0,009796375341	stage/street_concert	0,419308698				**0,4095123227**
microphone	0,01434469246	stage/street_concert	0,419308698				**0,4049640055**
show	0,01574417466	stage/street_concert	0,419308698				**0,4035645233**
singer	0,02662514869	person/musician	0,6933436207				**0,666718472**
netta	0,01877638607	person/musician	0,6933436207				**0,6745672346**
surie	0,01644391575	person/musician	0,6933436207				**0,6768997049**
man	0,01504443356	person/messi	0,6921798493				**0,6771354158**
song	0,05709887342	-	0				**0,05709887342**
photos	0,06297669862	-	0				**0,06297669862**
ferrell	0,04618291232	person	0,6033333333				**0,557150421**
eurovision	0,03292281856	-	0				**0,03292281856**
she	0,03148834931	person	0,6033333333				**0,571844984**
film	0,02728990274	-	0				**0,02728990274**
contest	0,02632775873	-	0				**0,02632775873**
parody	0,02554055	-	0				**0,02554055**
gaza	0,02449093835	-	0				**0,02449093835**
prince	0,02344132671	person	0,6033333333				**0,5798920066**
report	0,02029249178	-	0				**0,02029249178**
star	0,01994262123		0				**0,01994262123**
video	0,01952277657	-	0				**0,01952277657**
israel	0,01830989201		0				**0,01830989201**
message	0,01574417466	-	0				**0,01574417466**
israeli	0,01574417466	-	0				**0,01574417466**
ryan	0,01574417466	person	0,6033333333				**0,5875891587**
media	0,01539430411	-	0				**0,01539430411**
crowd	0,01539430411	person	0,6033333333				**0,5879390292**
dress	0,01539430411	-	0				**0,01539430411**
countries	0,01521936883	-	0				**0,01521936883**
palestinian	0,01504443356	person	0,6033333333				**0,5882888998**
visit	0,01504443356	-	0				**0,01504443356**
flag	0,01434469246	-	0				**0,01434469246**
band	0,01399482192	-	0				**0,01399482192**
protester	0,01399482192	person	0,6033333333				**0,5893385114**
home	0,01364495137	-	0				**0,01364495137**
violence	0,01224546918	-	0				**0,01224546918**
world	0,01049611644	-	0				**0,01049611644**
barzilai	0,009796375341	-	0				**0,009796375341**
chinese	0,008047022602	-	0				**0,008047022602**
netherlands	0,006997410958	-	0				**0,006997410958**
secutity	0,006297669862	-	0				**0,006297669862**
europe	0,005597928766	-	0				**0,005597928766**
lisbon	0,005422993492	-	0				**0,005422993492**
albania	0,005248058218	-	0				**0,005248058218**
						total:	**2,136654725**

Table C.7: Video 2 - Joined classification

words	w_values	detection	c_values	sqr of words values	sqr detection values	cosine similarity	euclidean distance
performance	0,167937863	street_concert	0,5992723475	0,2241531654	2,578585065	**0,5063363807**	**0,4313344845**
stage	0,01460709537	street_concert	0,5992723475				**0,5846652521**
music	0,009796375341	street_concert	0,5992723475				**0,5894759722**
microphone	0,01434469246	street_concert	0,5992723475				**0,584927655**
show	0,01574417466	street_concert	0,5992723475				**0,5835281728**
singer	0,02662514869	musician/person	0,728445025				**0,7018198763**
netta	0,01877638607	musician/person	0,728445025				**0,7096686389**
surie	0,01644391575	musician/person	0,728445025				**0,7120011092**
man	0,01504443356	person/groom/bridegroom	0,7210355333				**0,7059910998**
song	0,05709887342	-	0				**0,05709887342**
photos	0,06297669862	-	0				**0,06297669862**
ferrell	0,04618291232	person	0,58				**0,5338170877**
eurovision	0,03292281856	-	0				**0,03292281856**
she	0,03148834931	person/groom/bridegroom	0,7210355333				**0,689547184**
film	0,02728990274	-	0				**0,02728990274**
contest	0,02632775873	-	0				**0,02632775873**
parody	0,02554055	-	0				**0,02554055**
gaza	0,02449093835	-	0				**0,02449093835**
prince	0,02344132671	person/groom/bridegroom	0,7210355333				**0,6975942066**
report	0,02029249178	-	0				**0,02029249178**
star	0,01994262123		0				**0,01994262123**
video	0,01952277657	-	0				**0,01952277657**
israel	0,01830989201		0				**0,01830989201**
message	0,01574417466	-	0				**0,01574417466**
israeli	0,01574417466	-	0				**0,01574417466**
ryan	0,01574417466	person	0,58				**0,5642558253**
media	0,01539430411	-	0				**0,01539430411**
crowd	0,01539430411	person	0,58				**0,5646056959**
dress	0,01539430411	groom/brigegroom	0,1410355333				**0,1256412292**
countries	0,01521936883	-	0				**0,01521936883**
palestinian	0,01504443356	person	0,58				**0,5649555664**
visit	0,01504443356	-	0				**0,01504443356**
flag	0,01434469246	-	0				**0,01434469246**
band	0,01399482192	-	0				**0,01399482192**
protester	0,01399482192	person	0,58				**0,5660051781**
home	0,01364495137	-	0				**0,01364495137**
violence	0,01224546918	-	0				**0,01224546918**
world	0,01049611644	-	0				**0,01049611644**
barzilai	0,009796375341	-	0				**0,009796375341**
chinese	0,008047022602	-	0				**0,008047022602**
netherlands	0,006997410958	-	0				**0,006997410958**
secutity	0,006297669862	-	0				**0,006297669862**
europe	0,005597928766	-	0				**0,005597928766**
lisbon	0,005422993492	-	0				**0,005422993492**
albania	0,005248058218	-	0				**0,005248058218**
						total:	**2,407331127**

Table C.8: Video 3 - Joined classification

words	w_values	detection	c_values	sqr of words values	sqr detection values	cosine similarity	euclidean distance
performance	0,167937863	stage/street_concert	0,6666666667	0,2241531654	2,868216635	**0,499982425**	**0,4987288037**
stage	0,01460709537	stage/street_concert	0,6666666667				**0,6520595713**
music	0,009796375341	stage/street_concert	0,6666666667				**0,6568702913**
microphone	0,01434469246	stage/street_concert	0,6666666667				**0,6523219742**
show	0,01574417466	stage/street_concert	0,6666666667				**0,650922492**
singer	0,02662514869	person/musician	0,8666666667				**0,840041518**
netta	0,01877638607	person/musician	0,8666666667				**0,8478902806**
surie	0,01644391575	person/musician	0,8666666667				**0,8502227509**
man	0,01504443356	person/messi	0,8				**0,7849555664**
song	0,05709887342	-	0				**0,05709887342**
photos	0,06297669862	-	0				**0,06297669862**
ferrell	0,04618291232	person	0,6666666667				**0,6204837543**
eurovision	0,03292281856	-	0				**0,03292281856**
she	0,03148834931	person	0,6666666667				**0,6351783174**
film	0,02728990274	-	0				**0,02728990274**
contest	0,02632775873	-	0				**0,02632775873**
parody	0,02554055	-	0				**0,02554055**
gaza	0,02449093835	-	0				**0,02449093835**
prince	0,02344132671	person	0,6666666667				**0,64322534**
report	0,02029249178	-	0				**0,02029249178**
star	0,01994262123		0				**0,01994262123**
video	0,01952277657	-	0				**0,01952277657**
israel	0,01830989201		0				**0,01830989201**
message	0,01574417466	-	0				**0,01574417466**
israeli	0,01574417466	-	0				**0,01574417466**
ryan	0,01574417466	person	0,6666666667				**0,650922492**
media	0,01539430411	-	0				**0,01539430411**
crowd	0,01539430411	person	0,6666666667				**0,6512723626**
dress	0,01539430411	-	0				**0,01539430411**
countries	0,01521936883	-	0				**0,01521936883**
palestinian	0,01504443356	person	0,6666666667				**0,6516222331**
visit	0,01504443356	-	0				**0,01504443356**
flag	0,01434469246	-	0				**0,01434469246**
band	0,01399482192	-	0				**0,01399482192**
protester	0,01399482192	person	0,6666666667				**0,6526718448**
home	0,01364495137	-	0				**0,01364495137**
violence	0,01224546918	-	0				**0,01224546918**
world	0,01049611644	-	0				**0,01049611644**
barzilai	0,009796375341	-	0				**0,009796375341**
chinese	0,008047022602	-	0				**0,008047022602**
netherlands	0,006997410958	-	0				**0,006997410958**
secutity	0,006297669862	-	0				**0,006297669862**
europe	0,005597928766	-	0				**0,005597928766**
lisbon	0,005422993492	-	0				**0,005422993492**
albania	0,005248058218	-	0				**0,005248058218**
						total:	**2,685117945**

Table C.9: Video 2 - Joined classification - 100%

words	**w_values**	**detection**	**c_values**	**sqr of words values**	**sqr detection values**	**cosine similarity**	**euclidean distance**
performance	0,167937863	street_concert	0,75	0,2241531654	3,884620674	**0,4798523507**	**0,582062137**
stage	0,01460709537	street_concert	0,75				**0,7353929046**
music	0,009796375341	street_concert	0,75				**0,7402036247**
microphone	0,01434469246	street_concert	0,75				**0,7356553075**
show	0,01574417466	street_concert	0,75				**0,7342558253**
singer	0,02662514869	musician/person	1				**0,9733748513**
netta	0,01877638607	musician/person	1				**0,9812236139**
surie	0,01644391575	musician/person	1				**0,9835560842**
man	0,01504443356	person/groom/bridegroom	1,416666667				**1,401622233**
song	0,05709887342	-	0				**0,05709887342**
photos	0,06297669862	-	0				**0,06297669862**
ferrell	0,04618291232	person	0,75				**0,7038170877**
eurovision	0,03292281856	-	0				**0,03292281856**
she	0,03148834931	person/groom/bridegroom	1,416666667				**1,385178317**
film	0,02728990274	-	0				**0,02728990274**
contest	0,02632775873	-	0				**0,02632775873**
parody	0,02554055	-	0				**0,02554055**
gaza	0,02449093835	-	0				**0,02449093835**
prince	0,02344132671	person/groom/bridegroom	1,416666667				**1,39322534**
report	0,02029249178	-	0				**0,02029249178**
star	0,01994262123		0				**0,01994262123**
video	0,01952277657	-	0				**0,01952277657**
israel	0,01830989201		0				**0,01830989201**
message	0,01574417466	-	0				**0,01574417466**
israeli	0,01574417466	-	0				**0,01574417466**
ryan	0,01574417466	person	0,75				**0,7342558253**
media	0,01539430411	-	0				**0,01539430411**
crowd	0,01539430411	person	0,75				**0,7346056959**
dress	0,01539430411	groom/bridegroom	0,6666666667				**0,6512723626**
countries	0,01521936883	-	0				**0,01521936883**
palestinian	0,01504443356	person	0,75				**0,7349555664**
visit	0,01504443356	-	0				**0,01504443356**
flag	0,01434469246	-	0				**0,01434469246**
band	0,01399482192	-	0				**0,01399482192**
protester	0,01399482192	person	0,75				**0,7360051781**
home	0,01364495137	-	0				**0,01364495137**
violence	0,01224546918	-	0				**0,01224546918**
world	0,01049611644	-	0				**0,01049611644**
barzilai	0,009796375341	-	0				**0,009796375341**
chinese	0,008047022602	-	0				**0,008047022602**
netherlands	0,006997410958	-	0				**0,006997410958**
secutity	0,006297669862	-	0				**0,006297669862**
europe	0,005597928766	-	0				**0,005597928766**
lisbon	0,005422993492	-	0				**0,005422993492**
albania	0,005248058218	-	0				**0,005248058218**
						total:	**3,710150132**

Table C.10: Video 3 - Joined classification - 100%

words	w_values	detection	c_values	sqr of words values	sqr detection values	cosine similarity	euclidean distance
performance	0,167937863	stage/microphone	0,06100674233	0,2241531654	0,1364152229	**0,4437760658**	**0,1069311207**
stage	0,01460709537	stage/microphone	0,06100674233				**0,04639964696**
music	0,009796375341	stage/microphone	0,06100674233				**0,05121036699**
microphone	0,01434469246	stage/microphone	0,06100674233				**0,04666204987**
show	0,01574417466	stage/microphone	0,06100674233				**0,04526256768**
dress	0,01539430411	-	0				**0,01539430411**
song	0,05709887342	-	0				**0,05709887342**
photos	0,06297669862	-	0				**0,06297669862**
ferrell	0,04618291232	-	0				**0,04618291232**
eurovision	0,03292281856	-	0				**0,03292281856**
she	0,03148834931	-	0				**0,03148834931**
film	0,02728990274	-	0				**0,02728990274**
singer	0,02662514869	-	0				**0,02662514869**
contest	0,02632775873	-	0				**0,02632775873**
parody	0,02554055	-	0				**0,02554055**
gaza	0,02449093835	-	0				**0,02449093835**
prince	0,02344132671	-	0				**0,02344132671**
report	0,02029249178	-	0				**0,02029249178**
star	0,01994262123	-	0				**0,01994262123**
video	0,01952277657	-	0				**0,01952277657**
netta	0,01877638607	-	0				**0,01877638607**
israel	0,01830989201	-	0				**0,01830989201**
surie	0,01644391575	-	0				**0,01644391575**
message	0,01574417466	-	0				**0,01574417466**
israeli	0,01574417466	-	0				**0,01574417466**
ryan	0,01574417466	-	0				**0,01574417466**
media	0,01539430411	-	0				**0,01539430411**
crowd	0,01539430411	-	0				**0,01539430411**
countries	0,01521936883	-	0				**0,01521936883**
man	0,01504443356	-	0				**0,01504443356**
palestinian	0,01504443356	-	0				**0,01504443356**
visit	0,01504443356	-	0				**0,01504443356**
flag	0,01434469246	-	0				**0,01434469246**
band	0,01399482192	-	0				**0,01399482192**
protester	0,01399482192	-	0				**0,01399482192**
home	0,01364495137	-	0				**0,01364495137**
violence	0,01224546918	-	0				**0,01224546918**
world	0,01049611644	-	0				**0,01049611644**
barzilai	0,009796375341	-	0				**0,009796375341**
chinese	0,008047022602	-	0				**0,008047022602**
netherlands	0,006997410958	-	0				**0,006997410958**
secutity	0,006297669862	-	0				**0,006297669862**
europe	0,005597928766	-	0				**0,005597928766**
lisbon	0,005422993492	-	0				**0,005422993492**
albania	0,005248058218	-	0				**0,005248058218**
						total:	**0,2036728787**

Table C.11: Video 2 - DNN classification - more frames

words	w_values	detection	c_values	sqr of words values	sqr detection values	cosine similarity	euclidean distance
performance	0,167937863	-	0	0,2241531654	0,1410627911	**0,1904242877**	**0,167937863**
stage	0,01460709537	-	0				**0,01460709537**
music	0,009796375341	-	0				**0,009796375341**
microphone	0,01434469246	-	0				**0,01434469246**
show	0,01574417466	-	0				**0,01574417466**
dress	0,01539430411	groom/bridegroom	0,07053139556				**0,05513709145**
song	0,05709887342	-	0				**0,05709887342**
photos	0,06297669862	-	0				**0,06297669862**
ferrell	0,04618291232	-	0				**0,04618291232**
eurovision	0,03292281856	-	0				**0,03292281856**
she	0,03148834931	groom/bridegroom	0,07053139556				**0,03904304624**
film	0,02728990274	-	0				**0,02728990274**
singer	0,02662514869	-	0				**0,02662514869**
contest	0,02632775873	-	0				**0,02632775873**
parody	0,02554055	-	0				**0,02554055**
gaza	0,02449093835	-	0				**0,02449093835**
prince	0,02344132671	groom/bridegroom	0,07053139556				**0,04709006885**
report	0,02029249178	-	0				**0,02029249178**
star	0,01994262123	-	0				**0,01994262123**
video	0,01952277657	-	0				**0,01952277657**
netta	0,01877638607	-	0				**0,01877638607**
israel	0,01830989201	-	0				**0,01830989201**
surie	0,01644391575	-	0				**0,01644391575**
message	0,01574417466	-	0				**0,01574417466**
israeli	0,01574417466	-	0				**0,01574417466**
ryan	0,01574417466	-	0				**0,01574417466**
media	0,01539430411	-	0				**0,01539430411**
crowd	0,01539430411	-	0				**0,01539430411**
countries	0,01521936883	-	0				**0,01521936883**
man	0,01504443356	groom/bridegroom	0,07053139556				**0,055486962**
palestinian	0,01504443356	-	0				**0,01504443356**
visit	0,01504443356	-	0				**0,01504443356**
flag	0,01434469246	-	0				**0,01434469246**
band	0,01399482192	-	0				**0,01399482192**
protester	0,01399482192	-	0				**0,01399482192**
home	0,01364495137	-	0				**0,01364495137**
violence	0,01224546918	-	0				**0,01224546918**
world	0,01049611644	-	0				**0,01049611644**
barzilai	0,009796375341	-	0				**0,009796375341**
chinese	0,008047022602	-	0				**0,008047022602**
netherlands	0,006997410958	-	0				**0,006997410958**
secutity	0,006297669862	-	0				**0,006297669862**
europe	0,005597928766	-	0				**0,005597928766**
lisbon	0,005422993492	-	0				**0,005422993492**
albania	0,005248058218	-	0				**0,005248058218**
						total:	**0,2405606299**

Table C.12: Video 3 - DNN classification - more frames

words	w_values	detection	c_values	sqr of words values	sqr detection values	cosine similarity	euclidean distance
performance	0,167937863	stage/street_concert	0,2573418026	0,2241531654	0,6139637971	**0,4747816774**	**0,08940393963**
stage	0,01460709537	stage/street_concert	0,2573418026				**0,2427347072**
music	0,009796375341	stage/street_concert	0,2573418026				**0,2475454273**
microphone	0,01434469246	stage/street_concert	0,2573418026				**0,2429971102**
show	0,01574417466	stage/street_concert	0,2573418026				**0,241597628**
singer	0,02662514869	musician	0,1008835386				**0,07425838988**
netta	0,01877638607	musician	0,1008835386				**0,0821071525**
surie	0,01644391575	musician	0,1008835386				**0,08443962282**
man	0,01504443356	messi	0,1236732074				**0,1086287738**
song	0,05709887342	-	0				**0,05709887342**
photos	0,06297669862	-	0				**0,06297669862**
ferrell	0,04618291232	-	0				**0,04618291232**
eurovision	0,03292281856	-	0				**0,03292281856**
she	0,03148834931	-	0				**0,03148834931**
film	0,02728990274	-	0				**0,02728990274**
contest	0,02632775873	-	0				**0,02632775873**
parody	0,02554055	-	0				**0,02554055**
gaza	0,02449093835	-	0				**0,02449093835**
prince	0,02344132671	-	0				**0,02344132671**
report	0,02029249178	-	0				**0,02029249178**
star	0,01994262123		0				**0,01994262123**
video	0,01952277657	-	0				**0,01952277657**
israel	0,01830989201		0				**0,01830989201**
message	0,01574417466	-	0				**0,01574417466**
israeli	0,01574417466	-	0				**0,01574417466**
ryan	0,01574417466	-	0				**0,01574417466**
media	0,01539430411	-	0				**0,01539430411**
crowd	0,01539430411	-	0				**0,01539430411**
dress	0,01539430411	-	0				**0,01539430411**
countries	0,01521936883		0				**0,01521936883**
palestinian	0,01504443356	-	0				**0,01504443356**
visit	0,01504443356	-	0				**0,01504443356**
flag	0,01434469246	-	0				**0,01434469246**
band	0,01399482192	-	0				**0,01399482192**
protester	0,01399482192	-	0				**0,01399482192**
home	0,01364495137	-	0				**0,01364495137**
violence	0,01224546918	-	0				**0,01224546918**
world	0,01049611644	-	0				**0,01049611644**
barzilai	0,009796375341	-	0				**0,009796375341**
chinese	0,008047022602	-	0				**0,008047022602**
netherlands	0,006997410958	-	0				**0,006997410958**
secutity	0,006297669862	-	0				**0,006297669862**
europe	0,005597928766	-	0				**0,005597928766**
lisbon	0,005422993492	-	0				**0,005422993492**
albania	0,005248058218	-	0				**0,005248058218**
						total:	**0,5443146709**

Table C.13: Video 2 - CNN classification - more frames

words	w_values	detection	c_values	sqr of words values	sqr detection values	cosine similarity	euclidean distance
performance	0,167937863	street_concert	0,9772122511	0,2241531654	2,185113022	**0,4437760658**	**0,8092743881**
stage	0,01460709537	street_concert	0,9772122511				**0,9626051557**
music	0,009796375341	street_concert	0,9772122511				**0,9674158758**
microphone	0,01434469246	street_concert	0,9772122511				**0,9628675586**
show	0,01574417466	street_concert	0,9772122511				**0,9614680765**
singer	0,02662514869	-	0				**0,02662514869**
netta	0,01877638607	-	0				**0,01877638607**
surie	0,01644391575	-	0				**0,01644391575**
man	0,01504443356	-	0				**0,01504443356**
song	0,05709887342	-	0				**0,05709887342**
photos	0,06297669862	-	0				**0,06297669862**
ferrell	0,04618291232	-	0				**0,04618291232**
eurovision	0,03292281856	-	0				**0,03292281856**
she	0,03148834931	-	0				**0,03148834931**
film	0,02728990274	-	0				**0,02728990274**
contest	0,02632775873	-	0				**0,02632775873**
parody	0,02554055	-	0				**0,02554055**
gaza	0,02449093835	-	0				**0,02449093835**
prince	0,02344132671	-	0				**0,02344132671**
report	0,02029249178	-	0				**0,02029249178**
star	0,01994262123		0				**0,01994262123**
video	0,01952277657	-	0				**0,01952277657**
israel	0,01830989201		0				**0,01830989201**
message	0,01574417466	-	0				**0,01574417466**
israeli	0,01574417466	-	0				**0,01574417466**
ryan	0,01574417466	-	0				**0,01574417466**
media	0,01539430411	-	0				**0,01539430411**
crowd	0,01539430411	-	0				**0,01539430411**
dress	0,01539430411	-	0				**0,01539430411**
countries	0,01521936883		0				**0,01521936883**
palestinian	0,01504443356	-	0				**0,01504443356**
visit	0,01504443356	-	0				**0,01504443356**
flag	0,01434469246	-	0				**0,01434469246**
band	0,01399482192	-	0				**0,01399482192**
protester	0,01399482192	-	0				**0,01399482192**
home	0,01364495137	-	0				**0,01364495137**
violence	0,01224546918	-	0				**0,01224546918**
world	0,01049611644	-	0				**0,01049611644**
barzilai	0,009796375341	-	0				**0,009796375341**
chinese	0,008047022602	-	0				**0,008047022602**
netherlands	0,006997410958	-	0				**0,006997410958**
secutity	0,006297669862	-	0				**0,006297669862**
europe	0,005597928766	-	0				**0,005597928766**
lisbon	0,005422993492	-	0				**0,005422993492**
albania	0,005248058218	-	0				**0,005248058218**
						total:	**2,095233529**

Table C.14: Video 3 - CNN classification - more frames

words	w_values	detection	c_values	sqr of words values	sqr detection values	cosine similarity	euclidean distance
performance	0,167937863	-	0	0,2241531654	2,191987475	**0,3203793007**	**0,167937863**
stage	0,01460709537	-	0				**0,01460709537**
music	0,009796375341	-	0				**0,009796375341**
microphone	0,01434469246	-	0				**0,01434469246**
show	0,01574417466	-	0				**0,01574417466**
singer	0,02662514869	person	0,6609090909				**0,6342839422**
netta	0,01877638607	person	0,6609090909				**0,6421327048**
surie	0,01644391575	person	0,6609090909				**0,6444651752**
man	0,01504443356	person	0,6609090909				**0,6458646573**
song	0,05709887342	-	0				**0,05709887342**
photos	0,06297669862	-	0				**0,06297669862**
ferrell	0,04618291232	person	0,6609090909				**0,6147261786**
eurovision	0,03292281856	-	0				**0,03292281856**
she	0,03148834931	person	0,6609090909				**0,6294207416**
film	0,02728990274	-	0				**0,02728990274**
contest	0,02632775873	-	0				**0,02632775873**
parody	0,02554055	-	0				**0,02554055**
gaza	0,02449093835	-	0				**0,02449093835**
prince	0,02344132671	person	0,6609090909				**0,6374677642**
report	0,02029249178	-	0				**0,02029249178**
star	0,01994262123		0				**0,01994262123**
video	0,01952277657	-	0				**0,01952277657**
israel	0,01830989201		0				**0,01830989201**
message	0,01574417466	-	0				**0,01574417466**
israeli	0,01574417466	-	0				**0,01574417466**
ryan	0,01574417466	person	0,6609090909				**0,6451649163**
media	0,01539430411	-	0				**0,01539430411**
crowd	0,01539430411	person	0,6609090909				**0,6455147868**
dress	0,01539430411	-	0				**0,01539430411**
countries	0,01521936883	-	0				**0,01521936883**
palestinian	0,01504443356	person	0,6609090909				**0,6458646573**
visit	0,01504443356	-	0				**0,01504443356**
flag	0,01434469246	-	0				**0,01434469246**
band	0,01399482192	-	0				**0,01399482192**
protester	0,01399482192	person	0,6609090909				**0,646914269**
home	0,01364495137	-	0				**0,01364495137**
violence	0,01224546918	-	0				**0,01224546918**
world	0,01049611644	-	0				**0,01049611644**
barzilai	0,009796375341	-	0				**0,009796375341**
chinese	0,008047022602	-	0				**0,008047022602**
netherlands	0,006997410958	-	0				**0,006997410958**
secutity	0,006297669862	-	0				**0,006297669862**
europe	0,005597928766	-	0				**0,005597928766**
lisbon	0,005422993492	-	0				**0,005422993492**
albania	0,005248058218	-	0				**0,005248058218**
						total:	2,030648533

Table C.15: Video 2 - YOLO classification - more frames

words	w_values	detection	c_values	sqr of words values	sqr detection values	cosine similarity	euclidean distance
performance	0,167937863	-	0	0,2241531654	1,865601445	**0,3203793007**	**0,167937863**
stage	0,01460709537	-	0				**0,01460709537**
music	0,009796375341	-	0				**0,009796375341**
microphone	0,01434469246	-	0				**0,01434469246**
show	0,01574417466	-	0				**0,01574417466**
singer	0,02662514869	person	0,5625				**0,5358748513**
netta	0,01877638607	person	0,5625				**0,5437236139**
surie	0,01644391575	person	0,5625				**0,5460560842**
man	0,01504443356	person	0,5625				**0,5474555664**
song	0,05709887342	-	0				**0,05709887342**
photos	0,06297669862	-	0				**0,06297669862**
ferrell	0,04618291232	person	0,5625				**0,5163170877**
eurovision	0,03292281856	-	0				**0,03292281856**
she	0,03148834931	person	0,5625				**0,5310116507**
film	0,02728990274	-	0				**0,02728990274**
contest	0,02632775873	-	0				**0,02632775873**
parody	0,02554055	-	0				**0,02554055**
gaza	0,02449093835	-	0				**0,02449093835**
prince	0,02344132671	person	0,5625				**0,5390586733**
report	0,02029249178	-	0				**0,02029249178**
star	0,01994262123	-	0				**0,01994262123**
video	0,01952277657	-	0				**0,01952277657**
israel	0,01830989201	-	0				**0,01830989201**
message	0,01574417466	-	0				**0,01574417466**
israeli	0,01574417466	-	0				**0,01574417466**
ryan	0,01574417466	person	0,5625				**0,5467558253**
media	0,01539430411	-	0				**0,01539430411**
crowd	0,01539430411	person	0,5625				**0,5471056959**
dress	0,01539430411	-	0				**0,01539430411**
countries	0,01521936883	-	0				**0,01521936883**
palestinian	0,01504443356	person	0,5625				**0,5474555664**
visit	0,01504443356	-	0				**0,01504443356**
flag	0,01434469246	-	0				**0,01434469246**
band	0,01399482192	-	0				**0,01399482192**
protester	0,01399482192	person	0,5625				**0,5485051781**
home	0,01364495137	-	0				**0,01364495137**
violence	0,01224546918	-	0				**0,01224546918**
world	0,01049611644	-	0				**0,01049611644**
barzilai	0,009796375341	-	0				**0,009796375341**
chinese	0,008047022602	-	0				**0,008047022602**
netherlands	0,006997410958	-	0				**0,006997410958**
secutity	0,006297669862	-	0				**0,006297669862**
europe	0,005597928766	-	0				**0,005597928766**
lisbon	0,005422993492	-	0				**0,005422993492**
albania	0,005248058218	-	0				**0,005248058218**
						total:	**1,721463336**

Table C.16: Video 3 - YOLO classification - more frames

words	w_values	detection	c_values	sqr of words values	sqr detection values	cosine similarity	euclidean distance
performance	0,167937863	stage/street_concert/microphone	0,318348545	0,2241531654	2,725071338	**0,4255588613**	**0,150410682**
stage	0,01460709537	stage/street_concert/microphone	0,318348545				**0,3037414496**
music	0,009796375341	stage/street_concert/microphone	0,318348545				**0,3085521696**
microphone	0,01434469246	stage/street_concert/microphone	0,318348545				**0,3040038525**
show	0,01574417466	stage/street_concert/microphone	0,318348545				**0,3026043703**
singer	0,02662514869	person/musician	0,7617926295				**0,7351674808**
netta	0,01877638607	person/musician	0,7617926295				**0,7430162434**
surie	0,01644391575	person/musician	0,7617926295				**0,7453487137**
man	0,01504443356	person/messi	0,7845822983				**0,7695378647**
song	0,05709887342	-	0				**0,05709887342**
photos	0,06297669862	-	0				**0,06297669862**
ferrell	0,04618291232	person	0,8073529412				**0,7611700289**
eurovision	0,03292281856	-	0				**0,03292281856**
she	0,03148834931	person	0,8073529412				**0,7758645919**
film	0,02728990274	-	0				**0,02728990274**
contest	0,02632775873	-	0				**0,02632775873**
parody	0,02554055	-	0				**0,02554055**
gaza	0,02449093835	-	0				**0,02449093835**
prince	0,02344132671	person	0,8073529412				**0,7839116145**
report	0,02029249178	-	0				**0,02029249178**
star	0,01994262123		0				**0,01994262123**
video	0,01952277657	-	0				**0,01952277657**
israel	0,01830989201		0				**0,01830989201**
message	0,01574417466	-	0				**0,01574417466**
israeli	0,01574417466	-	0				**0,01574417466**
ryan	0,01574417466	person	0,8073529412				**0,7916087665**
media	0,01539430411	-	0				**0,01539430411**
crowd	0,01539430411	person	0,8073529412				**0,7919586371**
dress	0,01539430411	-	0				**0,01539430411**
countries	0,01521936883	-	0				**0,01521936883**
palestinian	0,01504443356	person	0,8073529412				**0,7923085076**
visit	0,01504443356	-	0				**0,01504443356**
flag	0,01434469246	-	0				**0,01434469246**
band	0,01399482192	-	0				**0,01399482192**
protester	0,01399482192	person	0,8073529412				**0,7933581193**
home	0,01364495137	-	0				**0,01364495137**
violence	0,01224546918	-	0				**0,01224546918**
world	0,01049611644	-	0				**0,01049611644**
barzilai	0,009796375341	-	0				**0,009796375341**
chinese	0,008047022602	-	0				**0,008047022602**
netherlands	0,006997410958	-	0				**0,006997410958**
secutity	0,006297669862	-	0				**0,006297669862**
europe	0,005597928766	-	0				**0,005597928766**
lisbon	0,005422993492	-	0				**0,005422993492**
albania	0,005248058218	-	0				**0,005248058218**
						total:	**2,515783944**

Table C.17: Video 2 - Joined classifiers - more frames

words	w_values	detection	c_values	sqr of words values	sqr detection values	cosine similarity	euclidean distance
performance	0,167937863	street_concert	0,9772122511	0,2241531654	2,917726827	**0,5464057543**	**0,8092743881**
stage	0,01460709537	street_concert	0,9772122511				**0,9626051557**
music	0,009796375341	street_concert	0,9772122511				**0,9674158758**
microphone	0,01434469246	street_concert	0,9772122511				**0,9628675586**
show	0,01574417466	street_concert	0,9772122511				**0,9614680765**
singer	0,02662514869	person	0,5625				**0,5358748513**
netta	0,01877638607	person	0,5625				**0,5437236139**
surie	0,01644391575	person	0,5625				**0,5460560842**
man	0,01504443356	person/groom/bridegroom	0,6330313956				**0,617986962**
song	0,05709887342	-	0				**0,05709887342**
photos	0,06297669862	-	0				**0,06297669862**
ferrell	0,04618291232	person	0,5625				**0,5163170877**
eurovision	0,03292281856	-	0				**0,03292281856**
she	0,03148834931	person/groom/bridegroom	0,6330313956				**0,6015430462**
film	0,02728990274	-	0				**0,02728990274**
contest	0,02632775873	-	0				**0,02632775873**
parody	0,02554055	-	0				**0,02554055**
gaza	0,02449093835	-	0				**0,02449093835**
prince	0,02344132671	person/groom/bridegroom	0,6330313956				**0,6095900689**
report	0,02029249178	-	0				**0,02029249178**
star	0,01994262123		0				**0,01994262123**
video	0,01952277657	-	0				**0,01952277657**
israel	0,01830989201		0				**0,01830989201**
message	0,01574417466	-	0				**0,01574417466**
israeli	0,01574417466	-	0				**0,01574417466**
ryan	0,01574417466	person	0,5625				**0,5467558253**
media	0,01539430411	-	0				**0,01539430411**
crowd	0,01539430411	person	0,5625				**0,5471056959**
dress	0,01539430411	groom/brigegroom	0,07053139556				**0,05513709145**
countries	0,01521936883	-	0				**0,01521936883**
palestinian	0,01504443356	person	0,5625				**0,5474555664**
visit	0,01504443356	-	0				**0,01504443356**
flag	0,01434469246	-	0				**0,01434469246**
band	0,01399482192	-	0				**0,01399482192**
protester	0,01399482192	person	0,5625				**0,5485051781**
home	0,01364495137	-	0				**0,01364495137**
violence	0,01224546918	-	0				**0,01224546918**
world	0,01049611644	-	0				**0,01049611644**
barzilai	0,009796375341	-	0				**0,009796375341**
chinese	0,008047022602	-	0				**0,008047022602**
netherlands	0,006997410958	-	0				**0,006997410958**
secutity	0,006297669862	-	0				**0,006297669862**
europe	0,005597928766	-	0				**0,005597928766**
lisbon	0,005422993492	-	0				**0,005422993492**
albania	0,005248058218	-	0				**0,005248058218**
						total:	**2,747604732**

Table C.18: Video 3 - Joined classifiers - more frames

words	w_values	detection	c_values	sqr of words values	sqr detection values	cosine similarity	euclidean distance
performance	0,167937863	stage/street_concert/microphone	0,5380952381	0,2241531654	3,179536391	**0,4603525751**	**0,3701573751**
stage	0,01460709537	stage/street_concert/microphone	0,5380952381				**0,5234881427**
music	0,009796375341	stage/street_concert/microphone	0,5380952381				**0,5282988628**
microphone	0,01434469246	stage/street_concert/microphone	0,5380952381				**0,5237505456**
show	0,01574417466	stage/street_concert/microphone	0,5380952381				**0,5223510634**
singer	0,02662514869	person/musician	0,9848484848				**0,9582233362**
netta	0,01877638607	person/musician	0,9848484848				**0,9660720988**
surie	0,01644391575	person/musician	0,9848484848				**0,9684045691**
man	0,01504443356	person/messi	1,032467532				**1,017423099**
song	0,05709887342	-	0				**0,05709887342**
photos	0,06297669862	-	0				**0,06297669862**
ferrell	0,04618291232	person	0,8181818182				**0,7719989059**
eurovision	0,03292281856	-	0				**0,03292281856**
she	0,03148834931	person	0,8181818182				**0,7866934689**
film	0,02728990274	-	0				**0,02728990274**
contest	0,02632775873	-	0				**0,02632775873**
parody	0,02554055	-	0				**0,02554055**
gaza	0,02449093835	-	0				**0,02449093835**
prince	0,02344132671	person	0,8181818182				**0,7947404915**
report	0,02029249178	-	0				**0,02029249178**
star	0,01994262123		0				**0,01994262123**
video	0,01952277657	-	0				**0,01952277657**
israel	0,01830989201		0				**0,01830989201**
message	0,01574417466	-	0				**0,01574417466**
israeli	0,01574417466	-	0				**0,01574417466**
ryan	0,01574417466	person	0,8181818182				**0,8024376435**
media	0,01539430411	-	0				**0,01539430411**
crowd	0,01539430411	person	0,8181818182				**0,8027875141**
dress	0,01539430411	-	0				**0,01539430411**
countries	0,01521936883	-	0				**0,01521936883**
palestinian	0,01504443356	person	0,8181818182				**0,8031373846**
visit	0,01504443356	-	0				**0,01504443356**
flag	0,01434469246	-	0				**0,01434469246**
band	0,01399482192	-	0				**0,01399482192**
protester	0,01399482192	person	0,8181818182				**0,8041869963**
home	0,01364495137	-	0				**0,01364495137**
violence	0,01224546918	-	0				**0,01224546918**
world	0,01049611644	-	0				**0,01049611644**
barzilai	0,009796375341	-	0				**0,009796375341**
chinese	0,008047022602	-	0				**0,008047022602**
netherlands	0,006997410958	-	0				**0,006997410958**
secutity	0,006297669862	-	0				**0,006297669862**
europe	0,005597928766	-	0				**0,005597928766**
lisbon	0,005422993492	-	0				**0,005422993492**
albania	0,005248058218	-	0				**0,005248058218**
						total:	**2,97641379**

Table C.19: Video 2 - Joined classifiers - more frames - 100%

words	w_values	detection	c_values	sqr of words values	sqr detection values	cosine similarity	euclidean distance
performance	0,167937863	street_concert	1	0,2241531654	3,170159806	**0,5416014228**	**0,832062137**
stage	0,01460709537	street_concert	1				**0,9853929046**
music	0,009796375341	street_concert	1				**0,9902036247**
microphone	0,01434469246	street_concert	1				**0,9856553075**
show	0,01574417466	street_concert	1				**0,9842558253**
singer	0,02662514869	person	0,5625				**0,5358748513**
netta	0,01877638607	person	0,5625				**0,5437236139**
surie	0,01644391575	person	0,5625				**0,5460560842**
man	0,01504443356	person/groom/bridegroom	0,8958333333				**0,8807888998**
song	0,05709887342	-	0				**0,05709887342**
photos	0,06297669862	-	0				**0,06297669862**
ferrell	0,04618291232	person	0,5625				**0,5163170877**
eurovision	0,03292281856	-	0				**0,03292281856**
she	0,03148834931	person/groom/bridegroom	0,8958333333				**0,864344984**
film	0,02728990274	-	0				**0,02728990274**
contest	0,02632775873	-	0				**0,02632775873**
parody	0,02554055	-	0				**0,02554055**
gaza	0,02449093835	-	0				**0,02449093835**
prince	0,02344132671	person/groom/bridegroom	0,8958333333				**0,8723920066**
report	0,02029249178	-	0				**0,02029249178**
star	0,01994262123		0				**0,01994262123**
video	0,01952277657	-	0				**0,01952277657**
israel	0,01830989201		0				**0,01830989201**
message	0,01574417466	-	0				**0,01574417466**
israeli	0,01574417466	-	0				**0,01574417466**
ryan	0,01574417466	person	0,5625				**0,5467558253**
media	0,01539430411	-	0				**0,01539430411**
crowd	0,01539430411	person	0,5625				**0,5471056959**
dress	0,01539430411	groom/brigegroom	0,3333333333				**0,3179390292**
countries	0,01521936883	-	0				**0,01521936883**
palestinian	0,01504443356	person	0,5625				**0,5474555664**
visit	0,01504443356	-	0				**0,01504443356**
flag	0,01434469246	-	0				**0,01434469246**
band	0,01399482192	-	0				**0,01399482192**
protester	0,01399482192	person	0,5625				**0,5485051781**
home	0,01364495137	-	0				**0,01364495137**
violence	0,01224546918	-	0				**0,01224546918**
world	0,01049611644	-	0				**0,01049611644**
barzilai	0,009796375341	-	0				**0,009796375341**
chinese	0,008047022602	-	0				**0,008047022602**
netherlands	0,006997410958	-	0				**0,006997410958**
secutity	0,006297669862	-	0				**0,006297669862**
europe	0,005597928766	-	0				**0,005597928766**
lisbon	0,005422993492	-	0				**0,005422993492**
albania	0,005248058218	-	0				**0,005248058218**
						total:	**3,005180158**

Table C.20: Video 3 - Joined classifiers - more frames - 100%