· U C ·
FCTUC FACULDADE DE CIÊNCIAS
E TECNOLOGIA
UNIVERSIDADE DE COIMBRA

Aesthetic Analysis of Images

Intermediate report

David Manuel Agostinho Nunes de Rebelo e Pascoal
pascoal@student.dei.uc.pt 2006130129
July 6, 2015

Advisors:
Penousal Machado
João Correia
Pedro Martins

Juri Arguente: Jorge Henriques (jh@dei.uc.pt)
Juri Vogal: Carlos Nuno Laranjeiro (cnl@dei.uc.pt)

Abstract

Aesthetic judgement is a highly subjective topic, however that has not stopped researchers from trying to reach a consensus on key elements that may prove themselves very valuable in its quantification. In this dissertation we will take a comprehensive look at multiple state-of-the-art image features regarding image quality and aesthetics, for the purpose of image analysis. The goal is to find and create a competitive general purpose set of features for image analysis. We will be exploring and expanding a previously built feature extractor by adding new feature extraction methods, either from studied features or by proposing new ones. We will be testing the efficiency of our features on a widely used data set in multiple image classification scenarios. The results obtained are described and analyzed, and the performance of our new feature extractor is compared with the old and other works.

Keywords: "Aesthetics", "Feature Extraction", "Image Classification", "Computer Vision"

Resumo

Opinião estética é um tópico muito subjetivo, no entanto isso não impede investigadores de tentarem reunir um consenso sobre elementos chave que poderão prover ser bastante importantes na sua quantificação. Nesta dissertação, vamos tentar oferecer uma visão abrangente sobre varias características de imagens existentes no estado-da-arte pertencentes à qualidade e estética de imagens, com o propósito de análise das mesmas. Nós iremos explorar e expandir um extrator de características previamente criado, ao adicionar novos métodos de extração de caraterísticas, não só através de caraterísticas existentes como também de caraterísticas propostas por nós. Iremos testar a eficiência das nossas caraterísticas num conjunto de dados amplamente utilizados em vários cenários de classificação de imagens. Os resultados obtidos são descritos e analisados, e a performance do nosso extrator de caraterísticas é comparado com o antigo e outros trabalhos.

Palavras-chave: “Estética”, “Extração de Caraterísticas”, “Classificação de imagem”, “Visão Computacional”

My special thanks to my family for encouraging me over the years.

I would also like to thank my supervisors Pedro Martins, João Correia and Penousal Machado for all the help and patience provided during my thesis period, as well as the opportunity to develop it.

Abbreviations

CGI Computer generated image(s)

CV Cross Validation

DCT Discrete Cosine Transform

DFT Discrete Fourier Transform

FE Feature Extractor

FFT Fast Fourier Transform

JPEG Joint Photographic Experts Group

RBCS Region-based Contrast Saliency

RGB Red green blue

ROI Region of Interest

STD Standard Deviation

Contents

1 Introduction **8**

1.1 Objectives 10

1.2 Contribution 10

1.3 Structure 11

2 State of the Art **12**

2.1 Photography Related Features 13

2.2 Artwork Related Features 16

2.3 Saliency Detection Algorithms 18

2.4 Features 19

2.4.1 Color 19

2.4.2 Texture 24

2.4.3 Composition 26

2.4.4 Saliency 33

2.4.5 Compression 37

2.5 Feature Application Tasks 39

2.5.1 Classification and Regression 40

2.5.2 Clustering 41

3 Ongoing Work **43**

3.1 Framework 43

3.2 Feature Extractor 43

3.3 Approach 44

4 Planning **46**

4.1 First Semester 46

4.2 Second Semester 46

5 Image Classification **48**

5.1 Dataset and Data Analysis 48

5.2 Parameter Variation 50

5.3 All Features 51

5.3.1 Naive Bayes 51

5.3.2 Random Forest 53

5.3.3 AdaBoost 54

5.3.4 Real AdaBoost 56

5.3.5 Support Vector Machine 57

5.4 Feature Selection 59

5.4.1 Naive Bayes 61

5.4.2 Random Forest 65

5.4.3 AdaBoost 69

5.4.4 Real AdaBoost 73

5.4.5 Support Vector Machine 76

5.5 Results Discussions 79

5.5.1 Full Feature Set 80

5.5.2 Feature Selection 83

6 Conclusions **90**

A Accuracy, precision, recall and AUROC values for all feature sets with the Naive Bayes classifier, highlighting the highest accuracy data set. **97**

B Accuracy, precision, recall and AUROC values for all feature sets with the Random Forest classifier, highlighting the highest accuracy data set. **99**

C Accuracy, precision, recall and AUROC values for all feature sets with the AdaBoost classifier, highlighting the highest accuracy data set. **101**

D Accuracy, precision, recall and AUROC values for all feature sets with the Real AdaBoost classifier, highlighting the highest accuracy data set. 103

E Accuracy, precision, recall and AUROC values for all feature sets with the SVM classifier, highlighting the highest accuracy data set. 105

F Features Selected for the New FE. 107

G Features Selected for the Old FE. 114

Chapter 1

Introduction

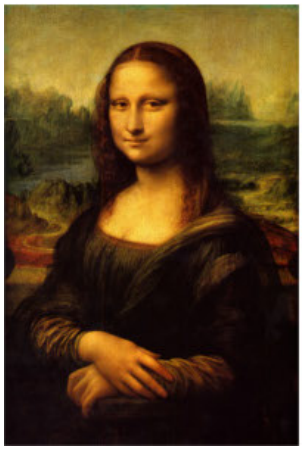
Aesthetic is usually associated with an adjective regarding the appreciation of beauty, but in a broader sense, *Aesthetics* is defined as "critical reflection on art, culture and nature". Aesthetics also encompasses style, which is usually associated with the distinctive appearance each artist conveys to his work. However different people have a different definition for it, which makes it very difficult to reach a consensus on aesthetic judgement. One of the reasons leading to this conflict in aesthetic judgement is each country's cultural differences, which originate from the different legacies built over the years. Meaning our past and environment shape us and our tastes making the world unique through each person's eyes. We can look as fashion as an example of people's ever changing perception of beauty, showing us that trends can affect the judgement of the same photograph of a model in different eras. Likewise the history of art shows us how painting styles have evolved over the years, from the enchanting stare of "Mona Lisa" by Leonardo Da Vinci to the unorthodox "Guernica" of Pablo Picasso we can attest how varied the artwork style is. This contrast can be seen in Fig. 1.1.

The interest in aesthetics is something that has followed mankind for a long time, the first major contributions come from ancient Greece as a branch of philosophy, some texts such as Plato's Republic, Aristotle's Poetics or the work On Sublimity attributed to Longinus are often indicated as precursors of modern aesthetic thought (Sheppard, Oleg V.Bychkov, 2010), and their influence can be felt up to the present. Greek philosophers had different opinions on the matter, Plato in particular is very well known for thinking the highest beauty of proportion is the combination of a beautiful mind with a beautiful body.

The first attempting to create a link between philosophical and computational aesthetics was the mathematician George David Birkhoff, who suggested the formula $M = \frac{O}{C}$, where he tries to measure the aesthetic value of an object by the ratio between order and complexity (Birkhoff, 1933). The order tries to measure the amount of regularities the image has and complexity refers to the number of elements in the image. This formula, along its premise had multiple applications ranging from music to poetry.

Drawing inspiration from Birkhoff's aesthetic measure, Bense (1969) developed the information aesthetic theory, where he converted the measure proposed by Birkhoff into an informational measure: redundancy divided by Shannon entropy (Shannon, 1948). As indicated by Bense, there is a set of elements (such as a palette of colors, sounds and phonemes) that are instrumental to the artistic invention process that carries over to the resulting product. Despite the distribution of elements

(a)



(b)

Figure 1.1: Two different artstyles. (a): Guernica (Pablo Picasso, 1937). (b): Mona Lisa (Leonardo da Vinci, 1503-1506).

of an aesthetic state having a determined order, the repertoire also demonstrates some complexity. Bense also made a distinction between global complexity, which spawns from partial complexities and a global order, which is created from partial orders. His contemporaries, Moles and Cohen (1968), considered order expressed not only as redundancy but also as the degree of predictability.

Another measure proposed with the intent of quantifying aesthetics is the one proposed by Koshelev et al. (1998). They defined their aesthetic measure, based in Kolmogorov complexity (Kolmogorov, 1963), as the combination of the length of the shortest program required to generate a given visual design and the running time of this program. Machado and Cardoso (1998) defined an aesthetic visual measure as a dependence on the ratio between image complexity and processing complexity. Utilizing both JPEG and fractal compression, they estimated both dependencies, concluding that images that are visually complex but easy to process have a higher aesthetic value.

Despite the innate difficulty of being able to quantify aesthetics, this is not a hopeless pursuit. Denis Dutton says: “art will itself have predictable content identifiable cross-culturally”(Berys Gout, 2005), implying that humans share characteristics at the most fundamental level simply by being human. So with this in mind we set out to try and find generic features that can be extracted from any image regardless of whether its a photograph, a painting or a CGI, with the intention of learning more on aesthetics quantification.

Still, it is hard to find a good balance between the right features and the different type of image, what can be considered as a solid feature for photographs can have no impact when judging paintings, such as brush strokes. This does not mean that all features have to be different between photographs, paintings or CGI, Datta et al. (2006) describes how important some techniques are important to images in general, an example is the "Rule of Thirds", which will be described further on.

The purpose of this dissertation is to find and create a suitable set of features for image analysis. The choice of the majority of features is based on image aesthetic background. This can have multiple practical applications, such as software that can sort images by aesthetic preference, software included in digital cameras that can provide feedback to improve our photographs or image searching software that can filter less aesthetic ones. It is easy to understand that in a society driven by

the ever growing social media, understanding aesthetics can prove to be very valuable. This work will be done atop an already existent Feature Extractor (FE) (Correia, 2011). We aim to make our FE a competitive entry into the image aesthetic judgement space, seeking a performance on par or better than the already existing ones in different image analysis scenarios. The difficulty presented by building a FE with the right features to correctly classify images based on aesthetics, means that there is always some room for improvement in this area. This situation presents a great opportunity to experiment with different feature combinations in order to obtain better classification solutions, improving the already existent FE.

1.1 Objectives

The goal of this dissertation is to explore and create a set of multipurpose features. These features can be used for many tasks, however, this thesis will focus on classification tasks, more specifically, separating images between high and low quality. These classification tests focus on validating the feature set. In order to achieve this goal, several objectives are proposed:

- Review of state-of-the-art methods on image analysis.
- Review the features of the previous FE.
- Implement feature extracting methods from existing state-of-the-art.
- Develop and implement new feature extracting methods.
- Improve the current FE implementation.
- Utilize the FE to conduct tests on different scenarios for image quality assessment.
- Evaluate and readjust accordingly the existing set of features.

1.2 Contribution

The following points illustrate what this dissertation aims to contribute to the field of research:

- Development of a new FE: A new FE will be built from the ground up. The core concept will be based on its previous iteration. It will include features from the previous FE as well as features from the current state-of-the-art.
- Exploration of new features: New features will be considered, including ones which have not been published yet.
- New feature set combinations: Some features may prove to have a bigger impact than others during the machine learning tasks, we will be experimenting with new feature combinations with the objective of optimizing the results.

1.3 Structure

This dissertation will be comprised of five additional chapters. Chapter 2 is the review of the literature, which aims to identify different researches about aesthetic assessment and how they can be relevant to our work. It will also consist of introductory sections and detailed descriptions on different types of features (section 2.4) and its applications (section 2.5). Chapter 3 introduces the previous FE (section 3.1), its structure (section 3.2) and how we will improve upon it (section 3.3). Chapter 4 will consist of first and second semesters work plan. 5 will present the results we obtained with our features, as well as the discussion of these results. And chapter 6 will conclude this document with the success of this dissertation, as well as what could be done to improve it.

Chapter 2

State of the Art

This chapter will be comprised of three parts. The first part analyzes studies done on the subject of image evaluation, mostly aesthetic. The focus here is to understand what the features implemented intended to accomplish, and the logic that makes them interesting. These articles are explained in a way that reveal the intention of the research when it comes to feature selection, as well as the type of images targeted. Finally, it is also important to understand the philosophy that makes these features work. We know that the researchers are trying to design features targeting human aesthetics and overall image quality, therefore it is interesting to understand the logical conclusion used to present those features. We will also be including studies regarding saliency maps. More information about what is saliency and why it is relevant will be further explained in the section 2.4.4. In the second part, we will cover a wide range of features and their purpose, with the objective of incorporating them into the current FE. These features are grouped into the following categories: color, texture, composition, saliency and compression.

The choice process took influences from recurring features, meaning that certain features appeared consistently often when investigating the current state-of-the-art which undoubtedly outlines their importance. Also during this research some features were more influential in the classification and regression processes, performing far better than the others, which was a strong motivation for their consideration.

Lastly we will finish with an overview of different classification, regression and clustering methods, explaining why they were chosen, how they fared against others and the performance obtained when judging images for its aesthetic value given some well known datasets. When conducting our study, we settled by the taxonomy presented in Fig. 2.1. This structure was created with the objective of organizing the features into the categories we defined, and also allows us to indicate which features are more ambiguous and overlap in categories.

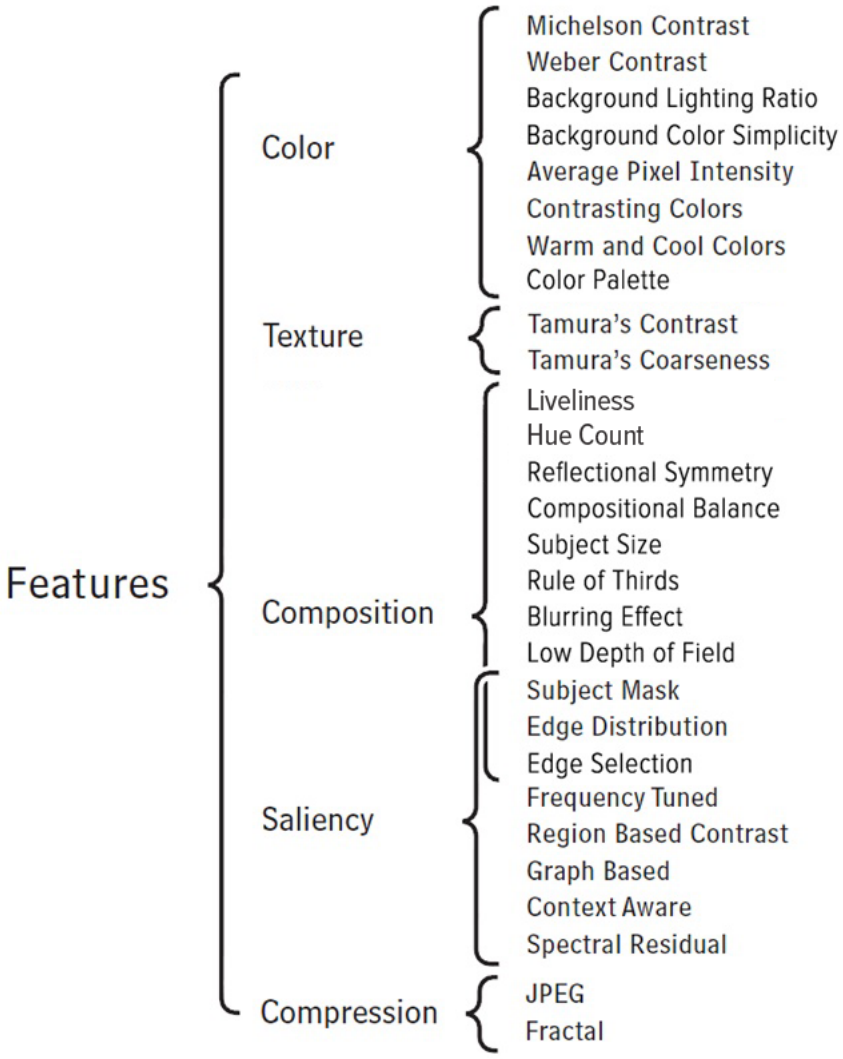


Figure 2.1: Our feature taxonomy is divided into 5 categories: color, texture, composition, saliency, compression and each category is comprised of different features.

2.1 Photography Related Features

Faria conducted some research in aesthetics, focused primarily on photographs, his work attempts to identify the elements that can make an image more aesthetically appealing (de Faria, 2012). Faria believes that the features described are similar to the way humans perceive the quality of a photograph and could be used by software to automatically identify a picture as being of high or low quality. The research begins by developing or adapting features from multiple areas that compose an image. The areas were divided into saliency map, subject mask, global features, subject features, background features and subject/background relation features. Faria also presented a more recent work, which aimed to improve the classification results previously obtained in the thesis described above. This work utilized the features presented in the thesis, in particular the ones who provided

the best results in classification studies. The objective is to compare the results from a newly introduced classification method, with the ones used previously (João Faria, Stanislav Bagley, Stefan Rüger, 2013). Also by using different data sets provided by multiple photography websites (both professional and amateur), Faria et al concludes that certain images provide better classification results with different sets of features, once again highlighting the complexity of creating a generic set of features that can discriminate any type of image. This work showed a classification accuracy up to 90%.

Datta is also known for his extensive research of aesthetics in images. In his work (Datta et al., 2006), focused on photography, the goal was to turn the aesthetic quality of images into a machine learning problem. By resorting to a statistical learning approach, the objective was to prove that there are features that can deem a picture, overall, more aesthetically pleasing. The choice the feature set was based on 3 principles, namely rules of thumb in photography, common intuition and observed trends in photography. Not only are the features used based on artistic intuition, but it is also mentioned the psychology underneath some trends observed in ratings, and as such his work looks for emotions that some pictures can elicit in people. An example would be an image with low saturation being considered dull. In the final part of his research, the best features are selected and then it is built a classification/regression models in order to separate images into high and low quality. This work achieved a classification accuracy of 70.12%.

Ke et al. presented a research in photography aesthetics which became very popular (Ke et al., 2006). This work attempts to distinguish between high professional photographs and low quality snapshots. The method developed to achieve this purpose relied on a top-down approach, building high level semantic features to help evaluate aesthetic quality. This is done to differentiate itself from other methodologies, such as grouping low level features and feeding them to classifiers for feature selection. The first step seeks to identify the criteria that people use to rate photos. After interpreting these key elements, feature development was the next step. These features attempted to extract the differences which distinguish photos from high or low quality. This was done by studying the techniques employed by professional photographers to achieve the best results. It was found that there are three differentiating factors between high and low quality photos, these factors are simplicity, realism and basic photographic technique. The features were chosen from these factors. This work achieved a classification accuracy of 72%.

Dhar et al. conducted a study inspired by the still increasing popularity of digital cameras (Dhar et al., 2011). In our modern world, where social portals make sharing pictures an easy task, if we consider the fact that any person has the possibility of having his/her pictures seen by big amounts of people, it makes sense that the pictures we share should be as best as possible. In this work, the aesthetic quality estimation is done by interpreting image properties that the author describes as *high level describable attributes*. The authors try to demonstrate that using high level attribute prediction is more effective for aesthetic image classification than using purely low level features, concluding that the most optimal setup for image classification is a mixture of both. High level attributes are based on visual cues that humans use to evaluate an image. The work makes use of two data sets in a distinct way to achieve its results. It uses pictures from DPChallenge[1], which promotes artistic content, to estimate the aesthetic quality of an image, differentiating them into high and low quality. On the other hand, Flickr's[2] different type of content is used, more specifically the social interaction

[1]http://www.dpchallenge.com/
[2]https://www.flickr.com/

aspect of photography to predict interestingness in images. This work is compared to the previously mentioned Ke et al. (2006), the authors show precision-recall curves comparing both works, which reveals improvements in aesthetic prediction.

In their research, Wang et al. (2013), made an attempt to understand images on the affective level. While this is a very subjective research, the idea behind it is that artists utilize certain elements such as color patterns or shapes to express emotions through their creations. The authors, inspired by art theories, developed a set of features which relate images to emotions. They refer to their image classification system as *affective image classification*, the features considered, differ a bit from standard low level features, since they have a more semantic purpose. The affective image classification is done by dividing the feature set into 8 emotional categories (amusement, anger, awe, contentment, disgust, excitement, fear and sad). And to make use of these categories, an automatic interpreter was developed which determined where an image will fit into its specific affective category. This work reveals true positive (proportion of actual positives which are correctly identified as such) ranges between 0.66 to 0.78.

Nishiyama et al. focuses on the assessment of color harmony. The authors believe that a big part of how people perceive the quality of a photograph is influenced by its color (Nishiyama et al., 2011). This work defends that color features like color histograms are incomplete, that it is needed to look at the bigger picture and consider how color harmony can improve the aesthetic quality of an image. In this research, it was found out that very often an image with high aesthetic quality had the quantity of local color patterns directly related to color harmony scores, (the higher the better. In order to assess the color harmony of photographs, it was proposed an automatic method to evaluate it, which was named *bags-of-color-patterns*. This method assumes that photos can be characterized by a group of simple color patterns, making the classification of the aesthetic quality of photos supported by the frequency of representation of these color patterns. In addition to the color features, some other features, such as blur, edges and saliency were also utilized for aesthetic quality classification. This work achieved a classification accuracy of 77.6%.

Utilizing a different approach, Lo et al. decided to select a set of features that are describable, discriminative, and computationally efficient (Lo et al., 2012). The main goal of this approach is to be able to apply their method to applications that need to provide fast aesthetic feedback, which could be very useful for portable devices. This way, it is necessary to avoid techniques with high processing power, such as subject detection or image segmentation. While restricting the set of possible features might not be optimal, this work aims to prove that you can reach high efficiency with low computational costs. The feature selection was done on the basis that color and composition are the most important elements which help determine the quality of a photo. This logic is further extended by connecting these choices to human perception of the same photo quality. The results obtained show an increase in true positive rates when compared with other works (one of them being the previously mentioned Ke et al. (2006)), demonstrating these features are not only efficient but accurate as well.

Khan et al. defend the idea that the quality of a photograph can be improved by crucial composition principles (Khan and Vogel, 2012). While many researchers attempt to quantify the aesthetic quality of an image as an independent whole, the authors promote a top-down comprehensive study of certain composition key elements in portrait photographs. The goal is to prove that adjusting a limited set of features to very particular image types is more advantageous than using a more standard

approach, where the goal is to find correlation between a big set of global statistical features and aesthetic classification. This methodology also leads the authors to consider practical applications such as mobile software, that provides instant feedback regarding the aesthetic quality of portrait photographs. Here, it is done a selection of 7 features that make use of knowledge present in salient face and background regions. These features focus on spatial composition and color-space elements, in order to aid highlight and shadow composition. The classification testing was not done extensively. Despite this consideration, this work showed a classification accuracy of 73.3% in a dataset composed of 145 photographs with a single person, from the flickr public data.

(Yeh et al., 2010) also developed a work focused on photographs, but this time the assessment is not done on professional photos but amateur ones, with special emphasis on accounting for individual preference. In this work it was developed two ranking interfaces which allow users to customize their personal photo ranking. One interface allows more knowledgeable users to manually define the weight each feature has on image selection. The other interface lets users select photographs they like more, these photos are available in the database, and the user selection will lead the system to update the weighting according to the selected photos. The logic behind this methodology is that the judgment of aesthetics is heavily dependent on feelings and personal taste, which leads to everyone having a personal ranking list. The feature selection was divided into two major groups, which were composition and color distribution. This choice is based on the belief that aesthetics in photography are reflected by the way that different visual elements are organized inside an image frame. This work achieved a binary classification accuracy of 93%.

Aesthetics are not only present in photos and paintings. In a study developed by Ciesielski et al., both photographs and abstract images are used for aesthetic assessment (Ciesielski et al., 2013). Utilizing a computer to judge and create images of high aesthetic value is a difficult task to achieve. The approach taken in this work involved analyzing images that people enjoyed with the intuit to obtain these key elements that make the images above all the others. To achieve this, this work contemplates two image databases, and both of them have been previously rated by people. The photograph database is the same used in (Datta et al., 2006)[3], in order to compare results. The other database is composed by abstract images generated from an evolutionary art system, developed in Xu et al. (2007). For the purpose of parity, the abstract images were also rated on a 7 point scale. By using data sets rated by humans and machine learning techniques, this work aimed to examine what separates high from low rated images, with the added objective of trying to find meaningful similarities between both types of images. The feature selection was based on two factors, being somewhat easy to compute and the capacity of being able to separate good from bad photos. The results show that for photographs, human raters tended for wavelet/texture features. Unlike abstract images, where human raters focused mostly on color features of the whole image. The classification results achieved 70% accuracy for photographs and 90% for abstract images.

2.2 Artwork Related Features

With the help of digital images, Li et al. presented a research study on aesthetic assessment of paintings (Li and Chen, 2009). In order to tackle this challenge, the authors turned it into a machine learning problem, presenting a set of methods to extract features that bring both global

[3]http://www.photo.net

and local elements of a painting. This is motivated by the need to bring together computer vision, art works and human perception, with the ultimate goal of utilizing a computer to correctly assess if a painting is appealing through human eyes. Being successful means being able to better understand how humans perceive aesthetics, and also finding patterns that can express human vision correctly. The data set for this research consisted on 100 pictures of paintings, these were manually scored by human subjects with the help of a survey developed for this sole purpose. The results allowed pictures to be separated into high and low quality. With the help of the questionnaire as well as intuition and some widely acknowledge rules in art, a group of features was extracted and later evaluated as efficient or not. The features extracted represented three major elements present in paintings, they were color, brightness and composition. The results focused mainly on interpreting the highest impact features. These were brightness contrast, complementary colors, blurring, symmetry and saturation.

Inspired by the research presented by Birkhoff, which introduced aesthetics as a quantifiable measure (Birkhoff, 1933), Rigau et al. present us with a work that follows Van Gogh through six periods, where even thou Van Gogh's art style might evolve, some aesthetic measures maintain consistency throughout these periods (Rigau et al., 2008). The goal here is to study the work that the artist created over the years, and find important elements present in the paintings (or its style) that can go unnoticed by the observer. To do so, the researchers analyzed the paintings with the help of three tools, the entropy of the palette, the compressibility of the image and the information channel which was used to seize the basic structure of the painting. Two new measures were also proposed in this work. These measures aim to quantify the information present in each color and region of a painting, allowing us to identify which colors and elements (objects or regions) are most salient in a painting, which the artist purposely chose to outline. The conclusion of the work does not translate into classification results or better performing features, since the objective is to interpret the artist path throughout the years.

Aesthetics and artwork go hand in hand, Machado and Cardoso present us a study, which occurred while partaking in the development of software capable of creating artworks (Machado and Cardoso, 1998). In this work, it is said that there are two factors which are crucial for the assessment of an artwork. These are its *content*, which is the visual interpretation of the artwork's meaning. As well as its *form*, which represents the visual aesthetic value of the artwork, such as color combination or composition, and it is ultimately what our research attempts to obtain with feature extraction. In this work it is proposed a formula which associates aesthetic value to the relation between image complexity and processing complexity. To obtain results a TDA(Test of Drawing Appreciation) was made, comparing correct answers between the program in this work and people. The results of the program showed an average score at 58.6, compared to the human average of 45.680 and fine arts graduates (55.6897).

Supported by his own previous work on unsupervised evolutionary art, den Heijer, developed his own set of aesthetic measures (den Heijer, 2012). The theme chosen is symmetry, because it is something present in our daily lives. It has also been often associated with aesthetic preference, marking it as a key feature in personal attractiveness. Symmetry has many angles, in this work it is being utilized bilateral symmetry and balance. Bilateral symmetry is most common in design, architecture and nature. We can see it present in the human body, as well as animals, or even buildings and objects. The feature developed from bilateral symmetry was named symmetry, and consists on an axis dividing two equal parts, horizontally, vertically and diagonally. The other

considered type of symmetry is balance, which is more prevalent in design and visual arts. It differs from bilateral symmetry in the sense that two halves of the image should present similar *weigt* without being necessarily equal. The feature developed, was named compositional balance and derives from balance, it is being computed by calculating the distance between two halves. The end result concluded with symmetry being efficient at assessing aesthetics, to a certain degree, since too much symmetry can turn to be boring.

There are multiple works based on the previously developed FE, which is described in section 3.1. This tool was utilized to extract feature values, which were used to assess and classify images based on aesthetic criteria (Romero et al., 2011) (Romero et al., 2012a) (Romero et al., 2012b). In Romero et al. (2012b), 20 low level features are utilized. These derive from the use of two filters (Canny and Sobel), two compression methods (JPEG and fractal) and two statistical metrics, Average and Standard Deviation(STD). The compression methods are computed for three levels of detail: low, medium and high. This process is used to compute a complexity measure, which is believed by the researchers to be one of the key features that has a high impact on the aesthetic assessment of the viewer. In order to be able to measure the effectiveness and directly compare the classification results obtained, the dataset used was similar to the one used by Datta, and the classification technologies were also adopted from other studies. This work showed successful classification results up to 74.5%.

2.3 Saliency Detection Algorithms

There are multiple ways to develop saliency detection algorithms. Saliency is very important to aesthetic assessment, as we seen earlier there are certain elements such as backgrounds or faces that are important to be singled out to compute certain features. We will be describing the saliency algorithms more in-depth in the section 2.4.4. In this work, Achanta et al. developed a saliency detection algorithm which utilizes an approach named *Frequency-Tuned* (Achanta et al., 2009). This method utilizes color and luminance features to estimate center-surround contrast. This algorithm was developed with three advantages in mind, well defined boundaries, full resolution and computation efficiency.

Cheng et al. proposed two global *Contrast Based* saliency algorithms (Cheng et al., 2011). The first relies on a histogram-based contrast method (HC) to measure saliency, these HC-maps assign values to each pixel based on the color differences present in the image. However, due to the heavy computing processing of this algorithm, a second one was proposed. This second algorithm instead of relying on spatial relationships to compute contrast at pixel-level, it utilizes a contrast analysis approach. This approach, named as region contrast (RC), combines spatial relationships into region-level contrast computation. Meaning the original image is first split into regions, then the color contrast is computed at region level, assigning different weights for region contrasts, depending on the region proximity.

Utilizing human fixation as a benchmark, Harel et al. developed their *Graph-Based* saliency algorithm (Harel et al., 2006). The approach used, seeks to fully utilize the computational power, topographical structure and parallel nature of graph algorithms in order to obtain efficient saliency computations. The logic behind the algorithm is to highlight the most relevant locations of the image, dictated by a defined criterion such as human fixation. This is done in three steps, computing feature maps, computing activation maps (which derive from node pairwise contrast), and the normalization

and combination of the activation maps.

Goferman et al. propose a different type of saliency, more specifically *Context-Aware* saliency, which highlights not only the single salient objects, but also the sections of the background that provide the context (Goferman et al., 2012). The main concept behind this algorithm, is making use of the distinction between the salient regions both locally and globally towards their surroundings. Therefore, the algorithm aims to mark the unique parts of the background. However, the algorithm also prioritizes regions close to the center of attention, attempting to keep background texture if interesting. This method is inspired by principles of human visual attention.

In a completely opposite approach, Hou et al. present a *Spectral Residual* saliency computing algorithm (Hou and Zhang, 2007). This method does not require any kind of prior features, categories or relations between the image elements. It draws inspiration from natural image statistics, attempting to simulate the behavior of pre-attentive visual search. The algorithm seeks to analyze the properties of the backgrounds. This is done by examining the log spectrum of each image in order to extract the spectral residual. Then it converts the spectral residual into spatial domain, in order to obtain the saliency map, which indicates the presence of visual indexes or proto objects. These are elements that can be promptly identified as objects, but these objects are not specific enough to be classified by their name or type. One of the biggest advantages of this algorithm is its generality.

Zhang et al. propose the *Boolean Map* based Saliency model (Zhang and Sclaroff, 2013). This approach searches for key indicators present in regions, utilizing them to better understand the division between figures and backgrounds. Based on psychological studies, the authors realized that some factors could make it more likely the achieve the aforementioned division. This work included the surroundedness key element to achieve saliency detection. This element signifies the relationship between the figure and the ground, which tends to be clear and well defined throughout images. As the name implies, this algorithm defines an image by a group of Boolean maps. The biggest advantage of this algorithm is its performance in achieving saliency detection.

2.4 Features

A feature is a measurable attribute, either physical or abstract. Thus we can extract certain features from images such as average pixel intensity and use the resulting data to compare two distinct images. By developing metrics that quantify every feature, it is possible to identify a set of properties that describes the target being measured. In this section a multitude of features will be described for the purpose of obtaining information on the aesthetic value of an image.

2.4.1 Color

Arguably, one of the categories with most impact in images is color. Its importance reaches beyond simple feature classification but also into semantic territory, colors evoke emotional responses such as black representing grief and sorrow in Western cultures, while yellow is used to express happiness or pleasant settings in China. Colors can also have associated meanings such as pink and purple symbolizing romance and royalty, respectively, adding an extra layer of complexity to this category,

Figure 2.2: A simple example of contrast. [4]

which cannot be quantified (Aslam, 2006).

Ciesielski et al. (2013) show that color is a major feature when identifying good from bad abstract images. Color features, such as Weber and Michelson contrast, can be used as means of making a particular section of an image stand out. Other features such as illumination can render an image unappealing if low values are observed, thus finding the perfect balance becomes essential. The following features are utilized for the quantification of the aesthetic value in any image observed.

Weber and Michelson Contrast

Contrast is a common technique applied to increase the aesthetic value of an image. It is normally used to make some intended part of the image to stand out, or to simply enhance a specific section (Fig. 2.2). de Faria (2012) discusses two different measures to obtain contrast, both measures offer similar data with the images tested. The first one is Weber contrast:

$$f_{WC} = \frac{1}{WH} \sum_{x=1}^{W} \sum_{y=1}^{H} \frac{I(x,y) - I_{avg}}{I_{avg}}, \tag{2.1}$$

where $I(x,y)$ is the intensity of the pixels of the image at position (x,y), and I_{avg} is the average intensity observed.

The second measure is known as Michelson Contrast. Unlike the Weber contrast, it only considers the extrema of the luminance values to calculate the luminance range:

$$f_{MC} = \frac{L_{max} - L_{min}}{L_{max} + L_{min}}, \tag{2.2}$$

where L_{max} and L_{min} are the highest and lowest of the luminance values, respectively.

[4] `http://jishytheobserver.files.wordpress.com/2012/04/contrast-wallpaper.jpg`

Background lighting ratio

The lighting in both subject and background also provides an important contrast. Luo and Tang (2008) highlight the importance of having a noticeable distinction in the brightness between subject and background. In his research, the most aesthetically pleasing images are the ones with a high subject/background contrast. This contrast is measured as:

$$Lighting = |\log \frac{B_{subject}}{B_{background}}|, \tag{2.3}$$

where $B_{subject}$ and $B_{background}$ are the mean brightness values for the subject and the background areas, respectively. The subject and background are obtained by creating a saliency map, further information on saliency maps is available in section 2.4.4.

Background color simplicity

When considering backgrounds in images, it is important to take color distribution into account. This feature evaluates the background area in order to obtain its color simplicity. Images possessing a background with high color complexity will concentrate high amounts of attention, thus distracting the observer from the subject which the picture aims to highlight (de Faria, 2012). When computing the feature, the first step is to first reduce each component of the RGB color space to a 16-values scale in order to reduce the computational complexity. This will result in a 4096-histogram (16x16x16), where each color is associated with a different bin. The background color simplicity feature is given by

$$S = \{i|H(i) >= \alpha m\}, \tag{2.4}$$

$$Bg_{simp} = ||S||/4096. \tag{2.5}$$

Where H is the histogram computed earlier, i is the histogram bin, m is the highest value obtained in H and α is the sensitivity parameter (this parameter was set as $\alpha = 0.01$). The result of this feature is the number of histogram bins which are above a certain relative value.

Average pixel intensity

Datta et al. (2006) used the average pixel intensity to characterize the presence of light. This way it can be judged whether or not there is over/under exposure or a good balance between high and low lighting (fig. 2.3). The average pixel intensity is given by

$$Avg_{PI} = \frac{1}{WH} \sum_{x=1}^{W} \sum_{y=1}^{H} I_V(x, y), \tag{2.6}$$

where I_V is the value channel of the image in the HSV color space.

Figure 2.3: Different light exposures.

Contrasting colors

Contrasting colors can also impact significantly the aesthetics of an image, Datta et al. (2006) proposed a method to compute relative color distribution, with the intention of distinguishing between different tiers of colors in images. Images with high values will be more interesting, thus captivating the viewers, while images with low values convey emotions of negativity.

Utilizing the Earth Mover's Distance (EMD) proposed by Rubner et al. (2000), which measures the distance between any two probability distributions, the color space is split into 64 cubic blocks with four similar partitions for each dimension where every one of these cubes is used as a sample point:

$$C_{colors} = emd(D_1, D_2, \{d(a,b)|0 \leq a, b \leq 63\}), \tag{2.7}$$

where the $D1$ distribution is produced as the color distribution of a possible image, where for all the 64 sample points, the frequency is $\frac{1}{64}$. The distribution $D2$ is calculated from an image by discovering how often there is color for each of the 64 cubes. In order to compute the EMD it is required a pairwise distance between any sampling point in both distributions to be given. As such, the pairwise Euclidean distance converted into the LUV space is computed:

$$d(a,b) = ||rgb2luv(C_a) - rgb2luv(C_b)|| \tag{2.8}$$

where $rgb2luv$ is the conversion from RGB to LUV and C_aC_b are the geometric centers of two cubes.

Warm and cool colors

Warm and cool colors are defined based on human feelings stimulated by colors (e.g., warm colors like red, yellow can arouse excitement, and cool colors like blue, green and purple make people calm (Fig. 2.4) (Wang et al., 2013). Cool colors are the ones with the hue values between 30 and 100 in the HSV space ([0-360]) while warm colors are outside the [30-100] values. This feature is computed

Figure 2.4: Warm and Cool colors in the hue channel.

by first counting how many pixels are located in the warm colors range and dividing by the total number of pixels, obtaining its ratio.

Color Palette

This feature analysis the color palette of an image, more specifically it analyses the contrast between colors in an image. The result of this feature is a vector with multiple features. This feature is computed with the following steps. (i) Reduction of the total number of colors by utilizing a quantization algorithm; (ii) sorting all the colors that exist in the image in descending order of occurrence; (iii) compute the Euclidean distance between one color and the next in the ordered list, this step will begin with the most frequent colors, and if the computed distance is lower than a predetermined threshold that color will be removed; (iv) compute the mean and STD of differences in occurrences; (v) compute the maximum and minimum values for each of the colors, as well as the distance between colors, for each of the channels HSV; (vi) compute the average and STD values of each of the colors, as well as the distance between colors, for the Value channel; (vii) compute the biggest and smallest distance between all the colors; (viii) compute the linear regression slope and square error.

Gif Quantization This method uses the median cut algorithm from the GIF[5] compression algorithm. The biggest advantage it presents is the ability to maintain the most important colors when reducing the number of colors in an image, presenting little degradation from the original (fig. 2.5). The algorithm works by continuously splitting the RGB color space into groups of equal size, where the color count is the sum of the histogram counts enclosed by each of its region.

[5]http://www.w3.org/Graphics/GIF/spec-gif87.txt

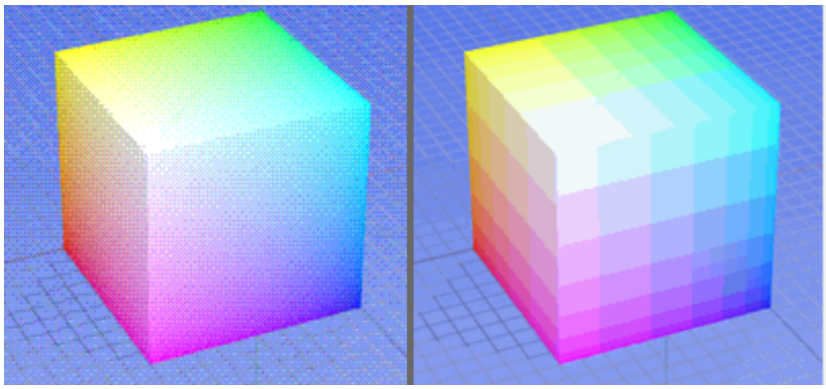


Figure 2.5: An example of the median cut algorithm applied to an image.[6]

2.4.2 Texture

Ciesielski et al. (2013) describe texture as the perceived surface quality of a work of art. The importance of texture is directly tied to the identification of objects or regions of interest in an image. Since it is an innate property of virtually all surfaces, it makes it easier to understand the relationship between the elements in the environment. The various aesthetic elements in texture includes components of pattern, granularity, and orientation among others. Pattern is often described as a repeating unit of shape or form, which proves to be very useful for extracting saliencies in an image. Granularity is an objective measure of graininess which indicates the presence/absence and nature of texture within the image. It can be used to try to identify certain elements, such as the sky. In this context, it is related to the spatial frequency of a texture.

Diagonal orientation

It has been observed that the presence of diagonal elements in the texture improves the aesthetic rating for viewers (Jacobs, 2011). Therefore, the objective is to search for these elements with various orientation angles. Tamura et al. (1978) propose a method that quantifies the directionality of a section by first obtaining edge strength and direction with the help of the Sobel edge detector. A histogram H(ϕ) is built with direction values obtained by quantizing ϕ and counting the pixels with magnitude larger than a predefined threshold. The directionality can then be quantified utilizing the sharpness of the peaks obtained in the histogram H, where acute peaks indicate that the image has high dimensionality in contrast with flatter peaks indicating the opposite:

$$Diag_{ori} = l - r \cdot n_p \cdot \sum_{p} \sum_{\phi \in W \cdot p} (\phi - \phi_p)^2 \cdot H(\phi), \tag{2.9}$$

where the number of peaks is represented by n_p, ϕ_p is the peak position of the histogram H, $W \cdot p$ is the set of bins distributed for every peak p, r is the normalizing factor connected to quantizing levels of ϕ and l is the value of the pixels.

Tamura roughness

Tamura et al. (1978) describe multiple features and their correlation. The Tamura roughness feature indicates high aesthetic value if it has a high value itself, however this feature is composed by two distinct features (contrast and coarseness), meaning both of those features must have a high value for Tamura roughness to score high.

Contrast

Contrast measures the variance of the gray levels in the image and quantifies the tendency of the distribution towards black or white:

$$contrast = \frac{\sigma}{(\alpha_4)^n} \tag{2.10}$$

$$\alpha_4 = \frac{\mu_4}{\sigma^4} \tag{2.11}$$

Tamura experimented multiple values for n, concluding 0.25 is the closest agreement to human measurements. σ is the standard deviation of gray levels, μ_4 is the fourth moment about the mean and α from the Kurtosis distribution, representing its shape which can have peaks or perhaps be a bit flatter.

Coarseness

Considered by Tamura et al. (1978) as the the most fundamental texture feature, its main purpose is to identify the largest size of the primitive elements, known as texels, forming the texture. A texture with higher coarseness is the one which has a smaller amount of texture elements for a fixed image size, as we can see in Fig. 2.6.

To obtain a coarseness measure, the first step is to compute multiple averages for every point over neighborhoods by powers of 2. For a given point (x, y) the average over the neighborhood of size $2^k \times 2^k$ is:

$$A_k(x,y) = \sum_{i=x-2^{k-1}}^{x+2^{k-1}-1} \sum_{j=y-2^{k-1}}^{y+2^{k-1}-1} \frac{I(i,j)}{2^{2k}}. \tag{2.12}$$

The next step is to compute the differences between pairs of averages in the neighborhoods which do not overlap and will also be on the opposite sides of the point in both horizontal and vertical orientations. The horizontal and vertical orientations are respectively given by

$$E_{k,h}(x,y) = |A_k(x+2^{k-1},y) - A_k(x-2^{k-1},y)|, \tag{2.13}$$

and

$$E_{k,v}(x,y) = |A_k(x,y+2^{k-1}) - A_k(x,y-2^{k-1})|. \tag{2.14}$$

At each point, it is picked the size corresponding to the highest output value for a maximized E regardless of direction. Finally the coarseness is computed by averaging $S_{opt}(x,y) = 2^{k_{opt}}$ for the whole image.

[7] http://www.micc.unifi.it/delbimbo/wp-content/uploads/2011/10/slide_corso/A12_texture_detectors.pdf

Figure 2.6: Coarseness increases from left to right.[7]

2.4.3 Composition

In art, composition refers to the arrangement of the components or visual elements of a scene to produce a visible artistic concept. As such, its importance comes from directing the viewer into seeing determined sections of a photograph or painting as well as placing focal elements in aesthetically correct places. In this subsection, the features considered will explore these points and the techniques used for achieving so.

Edge Detection

There are multiple algorithms to detect edges. This feature will apply two different edge detection methods to an image for all the channels in the HSV color space, and extract its results. The two edge detection methods used are the Canny algorithm (Canny, 1986) and the application of a Gabor filter. Both methods are part of the OpenCV library, making the integration simple. The Canny algorithm for edge detection is comprised of the following steps: (i) a Gaussian filter is applied to smooth the image with the purpose of removing the noise; (ii) by using the magnitude of the gradient, the most salient edge of the previously smoothed image is found; (iii) the edges are tracked by their hysterisis, i.e., of all the previously detected edges, the ones that are not connected to strong edges are suppressed. The Gabor method consists in first obtaining the Gaussian kernel and then apply a filter which will convolute the kernel with an image.

Symmetry

Symmetry is present in our everyday lives. It is undoubtedly a concept that has been related to aesthetic favoritism. den Heijer (2012) analyses the psychological intricacies of symmetry, revealing that humans are predisposed to ascertain symmetry within 100ms (Locher and Nodine, 1989). This shows how ingrained this perception is in our bodies. Other studies also reveal that non symmetrical faces greatly reduce personal attractiveness (Dutton, 2009). In order to compute this feature, two concepts of symmetry were considered. Reflectional symmetry, a specifying type of symmetry which separates two halves around an axis, one halve mirroring the other. The other concept is compositional balance, which is similar to reflectional symmetry in the sense that two halves should have equal *weight*, but at the same time it does not imply that one halve mirrors the other.

Reflectional Symmetry In order to compute reflectional symmetry the following method is used. We start by dividing the image into four quarters, making it so each half is comprised of two quarters,

as seen in fig. 2.7 . Thus we define the following halves as $A_{left} = A_1 + A_3$, $A_{right} = A_2 + A_4$, $A_{top} = A_1 + A_2$ and $A_{bottom} = A_3 + A_4$. The next step is to compute the following reflectional symmetries.

Horizontal symmetry

$$S_h(I) = s(A_{left}, A_{right}), \tag{2.15}$$

vertical symmetry

$$S_v = s(A_{top}, A_{bottom}), \tag{2.16}$$

diagonal symmetry

$$S_d = \frac{s(A_1, A_4) + s(A_2, A_3)}{2}. \tag{2.17}$$

The similarity between two areas is computed in the following way

$$s(A_i, A_j) = \frac{\sum_{x=0}^{w} \sum_{y=0}^{h} (sim(A_i(x, y), A_j^m(x, y)))}{w \cdot h}, \tag{2.18}$$

where the coordinates of the pixel are x and y, w is the width, h is the height of the area and A_j^m represents the mirrored area of A_j. The mirroring is done by assigning the vertical axis for the horizontal symmetry, the horizontal axis for the vertical symmetry and both axis for the diagonal symmetry. Then, the similarity between two opposing pixels can be defined as

$$sim(A_i(x, y), A_j(x, y)) = \begin{cases} 1 & if |I(A_i(x, y)) - I(A_j^m(x, y))| < \alpha, \\ 0 & otherwise \end{cases} \tag{2.19}$$

where $I(A_i(x, y))$ corresponds to the intensity value of a pixel with the coordinates (x, y) in the area A_i, and α is a difference threshold, set to 0.05. The intensity of a RGB pixel $I(x, y)$ is defined as the average of the value of its three components red, green and blue.

$$I(x, y) = \frac{r(x, y) + g(x, y) + b(x, y)}{3}. \tag{2.20}$$

The aesthetic measure for symmetry is extracted directly from the symmetries.

$$AM_{sym}(I) = S_m(I), \tag{2.21}$$

where m is horizontal, vertical or diagonal.

Compositional Balance To obtain the compositional balance, we split the image into two halves, the axis chosen are vertical, horizontal and diagonal. We then measure the distance between each half with the Stricker and Orengo image distance function (Stricker and Orengo, 1995). We compute the distance in the following way:

$$d_{SO}(I_a, I_b) = \frac{\sum_{i=0}^{i<N} w_i \cdot |v_{ai} - v_{bi}|}{\sum_{i=0}^{i<N} w_i}, \tag{2.22}$$

the function d_{SO} obtains the distance between the two image halves I_a and I_b, by computing the distance between two feature vectors v_a and v_b. N is the total number of image features and

Figure 2.7: We split the image into four quadrants to compute reflectional symmetry

w is the weight attributed to each of the features. We are using 12 features to build the feature vector, with each feature having a corresponding weight as seen in table. 2.1. Finally we compute the compositional balance measure:

$$CB(I) = 1 - d_{SO}(I_a, I_b), \tag{2.23}$$

where I is the image we are analyzing and d_{SO} is the previously computed distance between the two halves.

Image feature	Weight
Hue (avg)	4
Hue (sd)	4
Hue (skewness)	4
Saturation (avg)	1
Saturation (sd)	1
Saturation (skewness)	1
Intensity (avg)	2
Intensity (sd)	2
Intensity (skewness)	2
Colourfulness (avg)	2
Colourfulness (sd)	2
Colourfulness (skewness)	2

Table 2.1: These are the 12 features and its corresponding weights we used to compute the distance between two image halves. Source: (den Heijer, 2012)

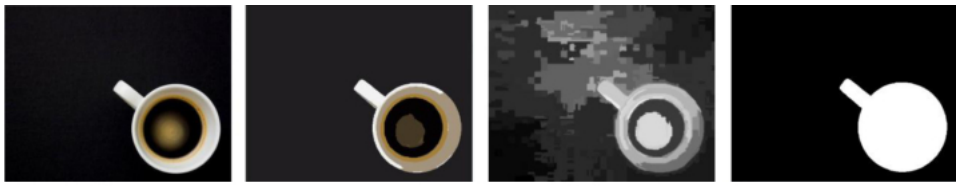


Figure 2.8: Components used for the subject mask computation. (a) original image; (b) meanshift image; (c) RBCS map; (d) subject mask. Source: de Faria (2012).

Liveliness

den Heijer (2012) observed that different intensities presented in an image can often be deemed interesting or lively. In order to measure whether an image is uninteresting or lively, we will be calculating the entropy of the intensity of the pixels presented in the image. We will be able to distinguish an image as uninteresting if it has little to no variation in its pixel intensity as well as low entropy, contrasting with a lively image which has many different pixel intensity values and high entropy. To compute this feature we will calculate the entropy for all available intensity values (256):

$$Liveliness(I) = -\sum_{i=1}^{n} p(x_i) log(p(x_i)), \tag{2.24}$$

where $x_i \in [0, .., 255]$ which is the intensity value of the pixels, and $\mathrm{p}(x_i)$ is the probability of the intensity value x_i.

Subject Mask

The distinction between subject and background is important for the development of other features such as subject size (subsection 2.4.3). To make this distinction we have a subject mask which is a binary image. The subject region tends to have a higher saliency compared to the background, meaning saliency maps can be used to predict the subject and background corresponding image pixels. There are multiple methods for creating saliency maps, further information on saliency maps is available in section 2.4.4. For subject mask extraction, de Faria (2012) uses both the frequency-tuned method which is explained in more detail in the subsection 2.4.4 and the Region-based contrast saliency (RBCS) method which can also be seen in the subsection 2.4.4. The frequency-tuned method uses a mean-shift algorithm (Fukunaga and Hostetler, 1975) which segments the image. Such segmentation is important since the subject is considered as an area instead of independent pixels. This way similar pixels might be assembled and interpreted as related pixels inside the image. The RCBS map is utilized to compute the average saliency value for every segment. The subject is then composed by all the segments which have the average saliency doubling the average image saliency. Figure 2.8 shows a visual description of the sequence of events.

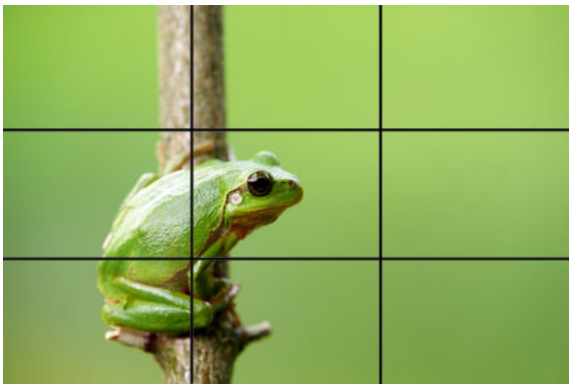


Figure 2.9: An example of the rule of thirds in a photograph. [8]

Subject size

de Faria (2012) proposes a feature which takes into account the subject importance in an image, more specifically its presence proportion, and its relevance to aesthetic judgements. He measures which percentage of the image that is considered part of the subject as:

$$SubjectSize = \frac{S_{area}}{T_{area}}, \tag{2.25}$$

where S_{area} stands for size of the area the subject occupies (subsection 2.4.3) and T_{area} for the whole image size.

Rule of thirds

A very popular rule in photography is the Rule of Thirds. As described by Datta et al. (2006), it consists on dividing a picture into 9 equal parts by two imaginary lines both horizontal and vertical, creating four intersection points in the middle of the image (Fig. 2.9). Those points define the Region Of Interest (ROI) and indicate where the object of interest should be placed. In theory this rule states that the ROI should be located at one of the four intersection points. Thus concluding that the principal object of an image should be located either inside or in the periphery of a ROI. The ROI will later on be retained for processing.

Blurring Effect

Blurring on an image can indicate multiple events, it can mean the image has poor compression, therefore making it look like a poor quality picture degrading it's aesthetic value. However it can also have other meanings attached to the image, such as the sense of speed, depth perception, generating background effects and shadows, smoothing edges in images eliminating "jaggies", conveying highlighting effects among others.

In order to obtain the blurring effect, Li and Chen (2009) utilized the following method to model

[8] `http://digitalphotographylive.com/wp-content/uploads/2012/09/Rule-Of-Thirds-Example.jpg`

the blurred image I^b which results from applying a Gaussian smoothing filter G_σ on a expected sharp image I^s, i.e., $I^b = G_\sigma * I^s$, where $*$ is the convolution operator. Both the parameter σ of the Gaussian kernel as well as the sharp image parameter I^s are unknown. By assuming that the frequency distribution of I^s is approximately the same, the smoothing parameter σ represents the degree of blurring. After using the Fourier transform on I^b, this method seeks the highest frequency whose power is greater than a certain threshold and assumes it as inverse-proportioned to the σ. After observing the highest frequency, if the frequency value obtained is small, this can be attributed to having a large σ blurring it. The blurring feature can be measured as:

$$Blur = \max(\frac{2(m - \lfloor \frac{M}{2} \rfloor)}{M}, \frac{2(n - \lfloor \frac{N}{2} \rfloor)}{N}) \propto \frac{1}{\sigma} \tag{2.26}$$

where (m,n) satisfies $|\zeta(m,n)| = |FFT(I^b)| > \varepsilon$, where FFT is the Fast Fourier Transform and ε was considered to perform best when set to 4.

Low Depth of Field

For higher aesthetic purposes, in macro shots the subject of interest can be enhanced by displaying it in sharp contrast with the out of focus surroundings. Figure 2.10 demonstrates this. We can identify this event with high values of the low Depth of Field indicator. Datta et al. (2006) extracts this feature by first dividing the image into 16 rectangular blocks of equal proportions denoted as $M_1, ..., M_{16}$, numbered in row-major order. After applying a three-level Daubechies wavelet transform, a set of level 3 wavelet coefficients in the high-frequency (denoted as $w_3 = w_3^{lh}, w_3^{hl}, w_3^{hh}$) of the hue image is utilized to compute the low depth of field indicator:

$$LowDOF = \frac{\sum_{(x,y) \in M_6 \cup M_7 \cup M_{10} \cup M_{11} \cup} w_3(x,y)}{\sum_{i=1}^{16} \sum_{(x,y) \in M_i} w_3(x,y)}, \tag{2.27}$$

where x and y correspond to the pixel position. A macro shot tends to capture the object of interest at the center, contrasting with the out of focus surrounding area. This indicates that a high value for this feature is consistent with the purpose of a macro shot.

Edge Distribution

When considering an image, we must take into account its complexity. Images with a high degree of complexity can confuse the viewer, since his attention gets split between various different areas in the image which in turn makes the image harder to judge and appreciate. Giving something the viewer can focus on may sometimes be more pleasing (Datta et al., 2006). One way to grasp simplicity of an image is by measuring the density of the spatial distribution of edges. Ke et al. (2006) computes this feature by first applying a 3×3 Laplacian filter to the image. The gradients directions are going to be discarded, thus only the absolute value is considered. If the image has color, it is applied a filter for each of the RGB channels separately while still computing a mean for all the channels. The Laplacian image is then resized to 100×100 and its image sum normalized to 1. This step ensures a easier time computing the edge spatial distribution. Then a bounding box which holds the

[9] `http://posterjackcanada.files.wordpress.com/2012/03/shallow-depth-of-field.jpg`

Figure 2.10: A macro shot emphasizing the low depth of field technique. [9]

top 96.04% of the edge energy is computed. The logic behind it is that crowded backgrounds will result in a larger bounding box, while concise subjects will result in a smaller bounding box. The area of the bounding box is obtained by projecting the Laplacian image L onto both axis x and y independently, so that

$$P_x(i) = \sum_y L(i, y), \tag{2.28}$$

$$P_y(j) = \sum_x L(x, j). \tag{2.29}$$

Hue Count

Just like we mentioned in the previous feature, measuring the complexity of an image can provide to be an excellent asset in being able to understand whether the image contains too much distractions. Another way to measure an image complexity is by counting the number of existing hues (Ke et al., 2006). The reasoning behind this, relies on the principle that most professional photographers attempt to mix different tones of a few hue values for their photos. Unlike low quality photos, which contain multiple objects with different distinctive colors. This feature is built by creating a 20-bin histogram of the hue component, where the pixels with brightness values ranging from [0,15; 0,95] and saturation greater than 0.2 are sought, or it would be inaccurate otherwise. We compute the set of bins:

$$N = \{i | H(i) > \alpha m\}, \tag{2.30}$$

where H is the computed histogram, m is the maximum value of the histogram and α is the sensitivity parameter set to 0,05, since it was reported to produce good results. Finally we compute

the hue count feature:

$$HC = 20 - ||N||. \tag{2.31}$$

2.4.4 Saliency

All the events that occur very few times are identified as a saliency. In photographs is what most people describe as the most significant element (Goferman et al., 2012). Studies show that elements that are interesting and manage to captivate the attention of people can contribute positively to visual aesthetics (Lind, 1980). The opposite can also be considered, where aesthetics can divert the attention of the viewer towards the saliency, be it an object or a person Coe (1992). In Wong and Low (2009) it is considered the quality of the interaction between subject/background as well as the overall quality of the subject in a photograph to be very discriminative features. The result of saliency methods is a saliency map, these display the relation between pixels in an image by assigning them a value between 0 and 1, for the minimum (black) and maximum (white) saliency values respectively. There are multiple ways to extract saliency maps from an image, this section will describe the different approaches and what they achieve.

Frequency-tuned Saliency

When Achanta et al. (2009) developed the salience map for their algorithm, a specific criteria were devised which the saliency map meets. These are enumerated directly from their work:

- Emphasize the largest salient objects.
- Uniformly highlight whole salient regions.
- Establish well-defined boundaries of salient objects.
- Disregard high frequencies arising from texture, noise and blocking artifacts.
- Efficiently output full resolution saliency maps.

To achieve the first criterion, very low frequencies derived from the original images were considered, which aids in highlighting salient objects uniformly as well, meeting the second criterion. For the third high frequencies obtained from the original image were retained, but in order to avoid noise, coding artifacts and texture patterns the frequencies with the highest value have to be disregarded as specified in the fourth criterion. The saliency map is then computed:

$$Map(x, y) = |I_\mu - I_{w_{hc}}(x, y)|, \tag{2.32}$$

where I_μ represents the arithmetic mean pixel value of the image I and $I_{w_{hc}}$ is the Gaussian blurred version of the original image, this is done with the use of a small Gaussian kernel and aims at removing fine texture details and also noise and coding artefacts (fourth criterion). Finally the image is processed without the use of any downsampling resulting in a saliency map with full resolution fulfilling the last criterion. An example of the resulting saliency map can be seen in Fig 2.11.

[10] `http://ivrgwww.epfl.ch/supplementary_material/RK_CVPR09/`

(a) (b)

Figure 2.11: (a) Original image; (b) Frequency-tuned saliency map.[10]

Region-based Contrast Saliency

The defining characteristic of this method is that relies on a region-based approach (Cheng et al., 2011). It groups similar pixels, this way the saliency map is comprised by saliency regions. Utilizing a graph-based method (subsection 2.4.4) the image is divided into segments, and for every segment the saliency is computed by comparing the contrast between colors of the regions in the image, so that

$$S(r_k) = \sum_{r_k \neq r_i} exp(-D_s(r_k, r_i)).w(r_i).D_r(r_k, r_i), \tag{2.33}$$

where D_s represents the Euclidean spatial distance between the centroids in the two regions, w is the relative region size and D_r is the Euclidean color distance for the CIELAB color space. The objective is to consider the color differences between regions as well as their proximity and size.

Graph-based Saliency

This method relies on handling pixels individually, which in turn will result in a saliency map without a strong separation region. Originally proposed by Harel et al. (2006). It follows three main steps which will be described in a brief overview. The first is to extract specific features, namely four orientation maps which correspond to the orientations $\phi = \{0°, 45°, 90°, 135°\}$ utilizing Gabor filters, a contrast map from the luminance variance and the luminance map. The second step is to create activation maps by using the extracted feature channels so that each obtained feature has a correspondent activation map, which in turn makes the entire map able to be interpreted as a fully-

(a) (b)

Figure 2.12: (a) Original image; (b) Graph-based saliency map.[11]

connected graph. The final step is to combine and normalize the activation maps. The computed activation maps are combined with the purpose of considering the whole feature set, but to do so it is first necessary to normalize every one of them. This is done since after combining all the maps, the resulting map may prove to be too uniform which does not provide much information, therefore it is crucial to concentrate the activation process in some key locations. In the end some different activation maps can be combined so the final image saliency is achieved.

Context-aware Saliency

When considering salient regions, its context can provide valuable information. Goferman et al. (2012) proposed a method and the main idea behind it is that salient regions differ between both local and global surroundings. Therefore the unique sections of the background as well as the dominant objects would be considered salient by this algorithm. Goferman et al. (2012), inspired by four basic principles of human visual attention, developed the following criteria for his algorithm. These are the local low-level considerations with the inclusion of some elements like color and contrast. Global considerations whose purpose is to suppress frequently-ocurring features, while at the same time keeping features that deviate from the norm. Visual organization rules affirm that that visual forms are organized through centers of gravity, and each form may be constituted by one or more of these centers. Also high-level factors, more specifically priors on the salient object location and object detection. The structure of the the saliency algorithm starts by first defining a single-scale local-global saliency in regards to criteria 1 and 3. After this it further enhances the saliency through the use of multiple scales. It continues by adjusting the saliency which falls in line with the third criterion. As for the final step, the fourth criterion is introduced as a post-processing event. An example of a saliency map created with this method can be seen in Fig. 2.13.

Local-global single-scale saliency What makes a pixel salient is its unique aspect, as is related to both the first and second criteria. This does not mean single pixels should only be considered, but instead it is necessary to look at its surroundings. This is done by computing the euclidean distances between vectorized patches (background pixels). The pixel is then considered salient when the value of the computed distance is high, more specifically in the interval $[1, 64]$. Another important factor is the positional distance between patches, falling in line with the third criterion, background patches have a higher propensity of sharing similar patches. Not only near but far-away as well, in the image.

[11] http://4.bp.blogspot.com/-lNyRB56FTPY/UYk7NcQ773I/AAAAAAAAANc/ZFKW36nMVUw/s640/flowerGBVS.png

(a) (b)

Figure 2.13: (a) Original image; (b) Context-aware saliency map.

Contrasting with salient patches who are usually grouped together. This is done by computing a dissimilarity measure between patches. This measure is proportional to the difference in appearance and inversely proportional to the positional distance.

Multi-scale saliency enhancement When multiple scales are used, patches are more probable to be similar in large homogeneous or blurred regions. Unlike the most salient pixels which might have more similar patches at some of the scales but not on every scale. Logically, multiple scales are included in order to reduce the saliency of background pixels. The way that multiple scales are used is by representing each pixel by the set of multi-scale image patches at its center.

Including the immediate context Visual forms are constituted by one or more centers of gravity about which the form is organized (Koffka, 1955). As such, it makes sense to take more into consideration areas close to the foci of attention rather than the furthest regions. This makes regions around the foci that give context to draw attention, making them salient.

High-level factors Refinements to the saliency map could be done through the addition of high level factors such as object recognition or face detection algorithms.

Spectral Residual Saliency

The spectral residual approach was proposed by Hou and Zhang (2007). Saliency detection of spectral residual is built on the concept of detecting the redundant portions of the log spectrum in an image. Spectral residual is the salient information and the average log spectrum is the general shape. This approach can be described by first applying the Fourier Transform to the selected image and obtaining its spectrum, then computing the amplitude and phase parts. Next by subtracting the average log spectrum, spectral residual is acquired. And finally by applying the Inverse Fourier Transform on the spectral residual the saliency map can be built, where the high value pixels indicate the salient regions. Due to the fact that only the Fourier Transform is used, this method becomes computationally fast and fairly efficient for target region localization. An example of this saliency detection can be seen in Fig. 2.14.

(a) (b)

Figure 2.14: (a) Original image; (b) Spectral saliency map.

2.4.5 Compression

Image compression serves the purpose of reducing the file size of digital images efficiently by reducing redundant information available in an image. Compression methods can either be lossy, where data is discarded during the compression process, or lossless which permits the reconstruction of the original data after the compression process. Lossy compression methods may display compression artifacts which are image distortion segments presented due to the data compression process. Some loss of fidelity is still acceptable, especially in photographs since it is an acceptable trade-off considering the largely imperceptible loss of quality for the big gain in stored data reduction. Both the fractal and JPEG compression methods, which will be described in this subsection, have proven to be valuable in image classification tasks according to stylistic criteria, such as, utilizing the author or aesthetic to be able to classify an image (Graves, 1946).

In the following works Machado et al. (2007) and Correia (2011), an estimate of image complexity is computed using the following formula:

$$Complexity(I) = RMSE(I, C(I)) \times \frac{s(C(I))}{s(I)}, \tag{2.34}$$

where $RMSE$ is the root mean square error, C is the compression rate for either the JPEG or Fractal methods of the image I, and s is the file size function. This equation can be applied to any $C(I)$. In the following subsubsections two compression examples will be introduced. These are JPEG (subsubsection 2.4.5) and Fractal (subsubsection 2.4.5).

JPEG compression

JPEG is a common compression method, normally used by digital photographs. There exists both lossy and lossless (JPEG2000) implementations, we are using the lossy version. Lossy means it looses information by compressing the image. Its advantage is the flexible tradeoff it provides between storage size and image quality, achieving good compress rates with minimal visible loss in image quality. For image perception, one of its main steps requires the transformation of a digital image into some internal representation. To achieve this transformation variable amounts of detail are lost, only retaining a few aspects of the image. This is similar to lossy image compression methods, where the image is coded with some error. By specifying the amount of detail kept, the error that results from the compression and its ratio is specified as well. For the JPEG implementation, a

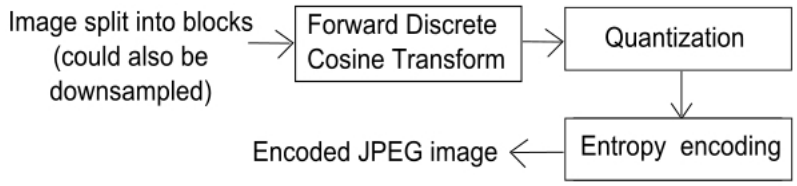


Figure 2.15: The JPEG enconding process[12].

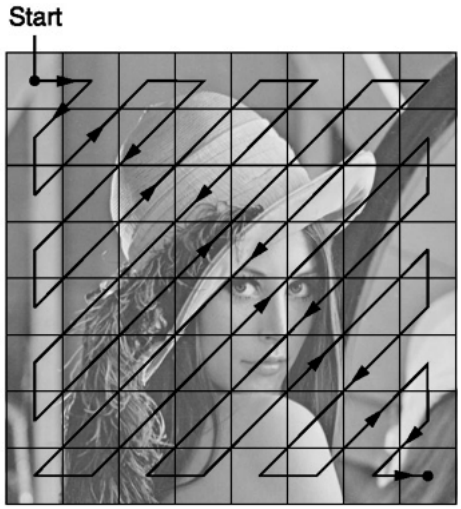


Figure 2.16: Entropy coding consists in arranging the image components in a "zigzag" order.

two-dimensional discrete cosine transformation (DCT) is applied. This DCT is based on the Fourier transform (DFT). These transformations are applied to a series of finite numbers, however the DCT works exclusively with the cosines unlike the DFT which only uses complex exponentials. The JPEG compression process (Fig. 2.15) is comprised by the following four steps:

1. The image is split in different blocks with a fixed size (normally containing 8x8 pixels).
2. A DCT is applied for every block.
3. The number of the coefficient matrix is obtained from the outcome of the DCT implementation.
4. The image is then compressed with the Huffman method followed by a Run Length Encoding procedure (entropy coding, Fig. 2.16).

[12] `http://upload.wikimedia.org/wikipedia/commons/6/68/JPEG_process.svg`

(a) (b)

Figure 2.17: (a) Original image; (b) Similarity. Source: Correia (2011)

One of the JPEG particularities that is present in the human eye is the sensitivity to changes in luminance (brightness changes). These become more obvious in small areas, where the brightness is uniform, contrary to areas with a great variation. This compression method does not consider relationships between different elements of the image, basically making it a local method.

Fractal compression

Fractal compression is an alternate method to JPEG. It differs from JPEG compression because it benefits from resemblances in the image (Fig. 2.17), while JPEG only considers local information. The properties of this compression method, as explained by Machado and Cardoso (1998) consist in displaying better image quality after the compression process over JPEG, even though there is no mathematical evidence to support it. Images encoded by this method do not have a fixed size, which allows them to be reproduced in any dimension. If the reproduced image has a greater size than the original, it is called a "fractal zoom", meaning these images have higher quality than images resulting from conventional zoom techniques. Finally, it revolves around exploring similar sections within the image. In contrast with other compression methods, these properties permit fractal compression to resemble the way the human eye processes an image. By sharing the same mathematical principals found in JPEG compression, the complexity estimate resulting from fractal compression can be obtained in terms of error and compression ratio.

2.5 Feature Application Tasks

Features have many applications, ranging from gathering data to grouping them due to their similar nature. Despite having data obtained from features, it is still important to know how to utilize these results. In this section both classification and regression will be described and the associated results they provided for the researchers that used them, along with the clustering methods employed and their purpose.

2.5.1 Classification and Regression

There are two viable paradigms, the classification paradigm distributes the data into a finite number of classes or categories, and its goal is to delineate the limits for the classes presented by the data. The second paradigm is regression, utilizing a different approach, it connects the data with real numbers, akin to giving an aesthetic rating. Its goal is to make that connection between the data and the real numbers through the discovery of a mathematical function. Providing results or classification is not the sole purpose of feature extraction, it can also be used for clustering, so groups can be made for similar categories.

AdaBoost Luo and Tang (2008) conducted a study on photo quality evaluation, with the unique characteristic of focusing on the subject. They utilized gentle adaboost (Torralba et al., 2004), which is a adaboost variant that uses a specific way of weighting its data, applying less weight to outliers. The success rate obtained was 96%. Khan and Vogel (2012) utilizing their proposed set of features for photographic portraiture aesthetic classification achieved an accuracy of 59.14% with the multiboosting variant (multi-class version) of adaboost (Benbouzid et al., 2012).

K-Nearest Neighbors Khan and Vogel (2012) utilizing their proposed set of features for photographic portraiture aesthetic classification achieved an accuracy of 59.33% with the K-NN algorithm (Altman, 1992).

Linear Regression Datta et al. (2006) estimated the effectiveness of regression by making use of the residual sum-of-squares error method, which measures the general difference between the obtained data and the values predicted. He obtained a 28% reduction from the variance, which was described as not very high but satisfactory due to the complexity of the task at hand.

Naive Bayes Li and Chen (2009) utilized the Naive Bayes classifier to aesthetically classify paintings in which the results were described as robust. Luo and Tang (2008) conducted a study on photo quality evaluation, with the unique characteristic of focusing on the subject. The success rate achieved utilizing a Bayesian classifier was 94%. Ke et al. (2006) study in photograph assessment through the use of high-level features achieved an accuracy of 72%.

Neural Network Machado et al. (2007), Romero et al. (2012b) used compression based features and an Artificial Neural Network (ANN) (Rumelhart et al., 1986) architecture as a classifier. In the first work it was tested the classifier capacity to identify the author of a set of paintings, reporting a success rate ranging from 90.9% to 96.7%. It was also executed a test with the purpose of measuring the degree of recognition and reaction to aesthetic principles, attaining an average success rate of 74.49%. In the second work the ANN classifier was used to predict the aesthetic merit of photographs, a success rate of 73.27% was attained.

Random Forest Khan and Vogel (2012) utilizing their proposed set of features for photographic portraiture aesthetic classification, achieved an accuracy of 59.79% by making use of random forests (Statis-

tics and Breiman, 2001). Ciesielski et al. (2013) achieved 73% accuracy when tackling photograph aesthetic assessment.

C4.5 Ciesielski et al. (2013) utilized the J48 algorithm which is an open source Java implementation of the C4.5 algorithm (Karimi and Hamilton, 2002), achieving 68% accuracy when tackling photograph aesthetic assessment.

Support Vector Machine In Datta et al. (2006), it was studied the correlation between a defined set of features and their aesthetic value, by using a previously rated set of photographs. For overall values Datta obtained up to 76% accuracy when varying the number of unique aesthetic ratings per photograph and up to 75% when increasing the gap between the high and low categories, utilizing the Support Vector Machine classifier (Cortes and Vapnik, 1995). Nishiyama et al. (2011) did a research on aesthetic classification of photographs based on color harmony, his approach on considering photographs as a collection of its local regions obtained an accuracy rating of 77.6%. Lo et al. (2012) study on photograph aesthetics assessment achieved an accuracy of 86%. Li and Chen (2009) utilized this classifier to aesthetically classify paintings in which the results were described as robust. Ke et al. (2006) research on aesthetic features for photo assessment reported an accuracy value of 72%. Wong and Low (2009) study on classification of professional photos and snapshots achieved an accuracy of 78.8%. Khan and Vogel (2012) utilizing their proposed set of features for photographic portraiture aesthetic classification achieved an accuracy of 61.10%. Luo and Tang (2008) conducted a study on photo quality evaluation, with the unique characteristic of focusing on the subject. The success rate achieved was 95%. Ciesielski et al. (2013) achieved 71% accuracy when tackling photograph aesthetic assessment. In Romero et al. (2012b) a SVM classifier was used to predict the aesthetic merit of photographs, a success rate of 74.59% was attained.

2.5.2 Clustering

Clustering consists in finding clusters or groups of data who share similar properties. Image clustering is a very usual unsupervised learning technique. By grouping sets of image data in a particular way, it maximizes the similarity within a cluster, while the similarity between clusters is minimized.

K-Means Clustering Datta et al. (2006) used k-means clustering to compute two features, these features would combine in order to measure how many distinct color blobs and how many disconnected significantly large regions are present in a photograph. Li and Chen (2009) utilized this method to initialize color clusters for image segmentation. Lo et al. (2012) utilized this method to find dominant colors in an image.

Fuzzy Clustering Felci Rajam and Valli (2011) utilized Fuzzy C-Mean clustering (FCM) (Bezdek, 1981) for the multi-class classification of images. Celia and Felci Rajam (2012) utilized FCM clustering for effective image categorization and retrieval.

Spectral Clustering Zakariya et al. (2010) utilized a spectral clustering technique named normalized cuts (Ncut) (Shi and Malik, 2000) to organize images with similar feature values.

Chapter 3

Ongoing Work

In this chapter we will be starting by first introducing the previous FE framework and its structure. Then we will introduce the new FE.

3.1 Framework

This work will expand upon the Feature Extractor, a previous project developed by Correia (2011). The goal is to study the structure of the previous FE and understand how it was implemented. We will also be analyzing the set of features existent in the FE, by doing so their relevance will be considered in order to decide which features will remain and which will be added. During development of the new FE, since we will be improving a previous tool, we will consider beneficial modifications regarding its structure and add them accordingly. Like the previous iteration this work will also utilize very popular API Open Source Computer Vision library (OPENCV)[1].In the following sections a brief overview of the previous FE will be given. Then we will be discussing the plans regarding the future of the new FE.

3.2 Feature Extractor

A feature can be defined as a measurable characteristic either concrete or abstract. An example of a concrete characteristic is the amount of lighting present in an image 2.4.1, while an abstract concept such as excitement an image provokes, can be attempted to be measured by the type of colors present 2.4.1. The resulting features are assessed by metrics, which will result in a quantified value for each feature. This approach allows us to gather a set of properties that characterizes the element we wish to measure. In order to obtain these measures a tool was built, primarily developed by Machado, and denominated as feature extractor. This initial build, developed with the graphic library "IMAGEMAGICK" [2], focused on estimating image complexity. However it presented some

[1] http://opencv.org/

[2] http://www.imagemagick.org/script/index.php

problems, namely portability and software stability. Moving forward, Correia developed a new version of the FE, this time in C++ and utilizing the very popular OPENCV API,

The previous Feature Extractor is structured in: (i) color channels, (ii) filters and (iii) metrics. This approach allowed the possibility of adding a color channel or a filter easily. It possesses a configuration file where all these elements could be turned on or off independently, therefore making it possible to turn off one color channel in the whole FE. By turning an element off, it automatically excludes it from analysis and feature extraction, thus we can chose which features are extracted depending on which context or situations they could be more suitable for analysis. This reduces the execution time if needed be or allow us to discard irrelevant features.

The feature extracting process can be explained through these steps (Fig. 3.1):

1. Image pre-processing, this includes all the transformation operations, as well as the partitioning and normalization which is applied to the input image.
2. Filters application, different image filters and their respective variants are applied to the resulting pre-processed images.
3. Metrics application, the utilization of determined methods based in measurements and statistical estimates provided by the filtered image.
4. Features building, resulting metrics are extracted, with the objective of acquiring the group of values characteristic of the input image.

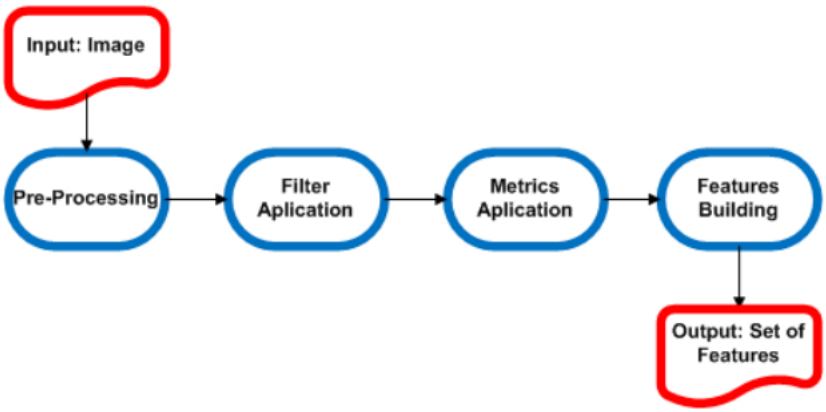


Figure 3.1: Overview of the original Feature Extractor. Source: Correia (2011).

3.3 Approach

Moving forward, our objective is to improve the existing FE. We did this by rebuilding the application while still keeping the overall flow of execution that has been described (section 3.2). The main improvement that we aimed for is the existing feature set. We kept some features from the existing FE, these are the color palette (section 2.4.1), edge detection (section 2.4.3), and both JPEG and fractal compression features (section 2.4.5).

We also focused on introducing all of the features described in our taxonomy (Fig. 2.1). We focused on having a functional version of the FE as soon as possible, so we could start testing the performance of our feature set. In order to test our feature set, we will be performing classification tasks which we will describe in chapter 5.

Chapter 4

Planning

This dissertation is divided in two semesters. As such it is necessary to create a well balanced work plan that correctly distributes the available work hours with the necessary work to be done. One Gantt chart will be presented, this chart will merge both the first and second semesters. Gantt charts are typically used to illustrate a project's schedule (Fig. 4.1).

4.1 First Semester

During the first semester, the main focus laid on investigating which features could be added to the existent FE and implementing them. This research will persist throughout the whole dissertation, as we will always be exploring the possibility of adding different features. The research during the first semester consisted in the evaluation of existent studies of aesthetic analysis, and which features performed the best. The next logical step was to implement them. This allowed us to better select our feature set, since the documentation on some features proved to be too vague, and other features proved to be redundant, leading us to discard some previous choices. As mentioned before this is a continuous process, and there are already plans to include more features which are not documented here. Finally it was done a review on the previous FE, where we included previous existing features in our own FE.

4.2 Second Semester

For the second semester, we finalized the implementation of all the features mentioned in the section 2.4. Despite having the possibility to implement even more features later along the road, we transitioned into the tests next and ended up not having the time to add more content to the FE. We then started to experiment with multiple data sets and quickly settled with one. Naturally, we then started testing classifiers and data validation methods. Meanwhile the report was being written, until we concluded the tests and finalized the data analysis.

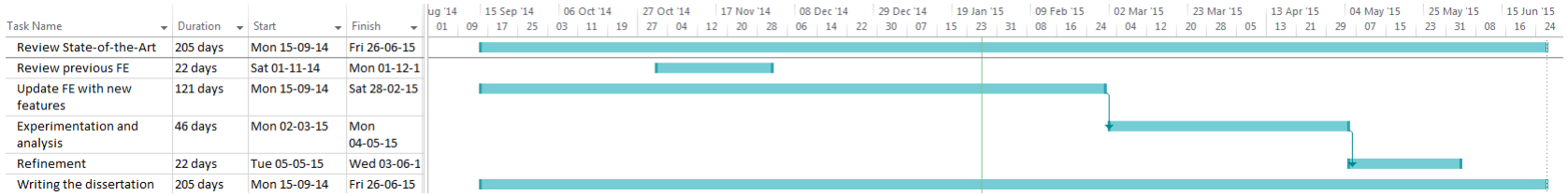


Figure 4.1: Gantt chart planning for the 2014/2015 year.

Chapter 5

Image Classification

In this chapter we will report the results for the different classification methods employed, as well as explain the choice of data set and the feature selection process. These steps will allow us to compare the performance of our FE relatively with the old FE as well as state-of-the-art studies. Data normalization as a pre-processing stage should be taken into account, since it can make a difference when considering classification performance (Graf et al., 2003). With this in mind we decided to normalize the data whenever possible, this means the feature set values are set between [0,1]. The classification methods we used were Support Vector Machine (SVM), Naive Bayes, Random Forest, AdaBoost and Real AdaBoost. We utilized these methods based on the papers who also studied this dataset, therefore we could directly compare our results with them.

5.1 Dataset and Data Analysis

The choice of the dataset is a very important step in order to to able to compare the performance of our features relative to others. There were multiple datasets available, however there was one who seems to have gathered the interest of the scientific community by featuring in multiple studies. This dataset originates from the DPChallenge website. Its origin is a study between high and low quality photographs (Ke et al., 2006). It was also utilized in many other studies, (de Faria, 2012) (Gadde and Karlapalem, 2011) (Luo and Tang, 2008) (Yeh et al., 2010) (Yeh et al., 2010) (Su et al., 2011). By including this dataset in our work, we are looking to have a stronger base of comparison, as well as contributing to the pool of studies in the community with our own. The test process is very long, with some tests having execution times of almost a week. The process was also constantly iterated, and thus repeated multiple times. It would not be feasible to have multiple datasets being tested, especially if some of those datasets only have one study as base of comparison.

The chosen dataset was made possible thanks to the DPChallenge website[1]. This is a digital photo contest website, where users submit their photographs competing against each other. Submitted photos are rated by other users in a scale from 1 to 10. The construction of the dataset began by first crawling and extracting 60000 photos submitted by more than 40000 different users. Also one of the quality control criteria required each photo to be rated by at least 100 different users, this is

[1] http://www.dpchallenge.com/

done in order to reinforce the validity of each photo evaluation. The value attributed to each photo consisted in its average rating. From there, 10% from both the top and bottom rated photos were selected and labeled as high quality professional photos and low quality snapshots. This distinction was made to clearly separate both ends of the spectrum, since distinguishing between high and low quality photos is a very subjective matter. But by eliminating the middle 80%, it is made an attempt to reduce ambiguous opinions and include the photos in which people have stronger and more unanimous impressions of their quality. The resulting dataset consists in a total of 12000 photos, separated into two 6000 photo subsets (low and high quality). Furthermore, each subset is split in half, one half assigned as a training set and the other as a testing set (Table 5.1).

	Test	Train
Low Quality Snapshots	3000	3000
High Quality Professional Photos	3000	3000

Table 5.1: Table illustrating the dataset distribution.

As mentioned before, the data set is divided into train and test sets equally 5.1. We will respect this predefined set up, since it will allow us not only to have consistent tests to compare with the state-of-the-art, but it is also crucial to not be utilizing the same set we used in the training stage to avoid overfitting. Overfitting happens when a model is very good at predicting data similar to the one that was used to train, but it cannot predict unseen data as accurately. We will also be including 10-fold Cross-Validation (CV) for every classifier except for AdaBoost which will use 5-fold CV.

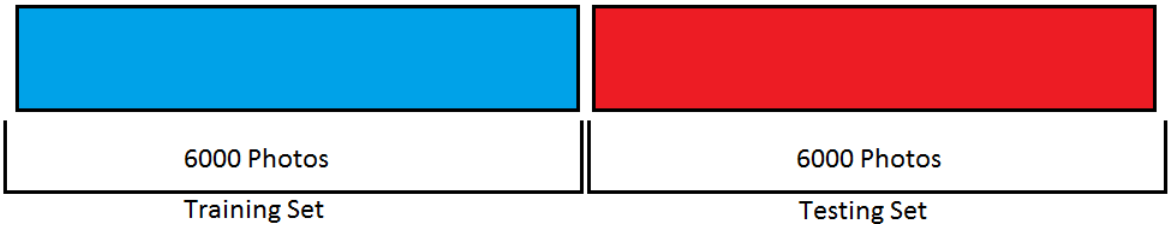


Figure 5.1: The Train/Test distribution of the dataset.

Cross-Validation consists in splitting the training set into smaller sets. This means a 10-fold CV method will split the set into 10 subsets. Out of these 10 sets, 9 will be used to train the model and the remaining set will be used to validate the process. This process repeats itself by the number of folds, which is 10 in this case, and we obtain the result by averaging the 10 results obtained in order to have a final value (Fig. 5.2).

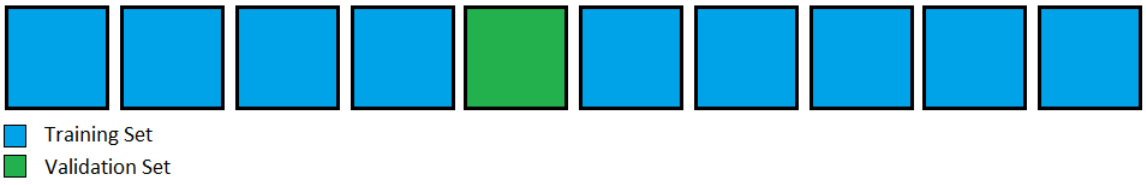


Figure 5.2: Example of a fold during the CV method.

We will be using the 10-fold CV and the Train/Test model validation techniques when training classifiers. We found out that the studies we encountered could either have one of these techniques or even both. For all intents and purposes, we decided to test with both. This way we can further compare the results between both new and old FE, while at the same time we ensure we can compare with other state-of-the-art studies as well.

To further helps us analyze the data, we will be making use of confusion matrices. These contain information about the elements that were correctly classified as well as the incorrect ones, for both the positive and negative classes (Table 5.2). The fields that comprise this matrix are True Negatives (TN), False Positives (FP), True Positives (TP) and False Negatives (FN). These values can help us obtain even more relevant information. These are:

Precision: The proportion of the correctly predicted positive cases.

$$Precision = \frac{TP}{TP + FP}, \tag{5.1}$$

Recall (True Positive Rate): The proportion of the correctly identified positive cases.

$$Recall = \frac{TP}{TP + FN}, \tag{5.2}$$

Fall-out (False Positive Rate): The proportion of negative cases that are mistakenly classified as positive.

$$Fall\text{-}out = \frac{FP}{FP + TN}, \tag{5.3}$$

Area Under the Receiver Operating Characteristic Curve (AUROC): The ROC curve is the plotting of the recall and fall-out metrics for every possible classification threshold. The AUC also a very relevant statistical property, it is equal to the probability that a classifier will rank a randomly chosen positive case higher than a negative one (Fawcett, 2006). By computing the area under this curve we can obtain a measure of how well the classifier is doing, a perfect score would be 1.

		Predicted	
		Negative	Positive
Actual	Negative	TN	FP
	Positive	FN	TP

Table 5.2: The confusion matrix

5.2 Parameter Variation

In order to efficiently train the selected classifiers and try to obtain the highest accuracy possible, it is important to experiment with different values to the corresponding parameters of each classifier.

For this purpose we will be doing a grid search on a predefined domain equally for all datasets. The parameters used are the following:

	Max Depth
Domain	[0, 50]
Increment	1

Table 5.3: Random Forest parameter grid search.

	Number of Iterations	Seed	Weight Threshold
Domain	[1, 10]	[0, 10]	[50, 150]
Increment	1	1	1

Table 5.4: AdaBoost and Real AdaBoost parameter grid search.

The reason we experiment with multiple parameters is because the defaults may not provide the best results. Our choice of domain is based on encompassing the edges of the default values. For example, the default value of the weight threshold parameter in AdaBoost was set to 100, therefore we varied it from 0 to 150.

5.3 All Features

We first start our classification procedure with the feature set that contains all the implemented features.

5.3.1 Naive Bayes

It should be noted that the Naive Bayes classifier we are using does not have configurable parameters. The Accuracy values can be seen in the Table 5.5.

	New FE	Old FE
10-Fold CV Accuracy	68,22%	64,83%
Train/Test Accuracy	84,93%	50,02%

Table 5.5: Accuracy Values for the old and new FE

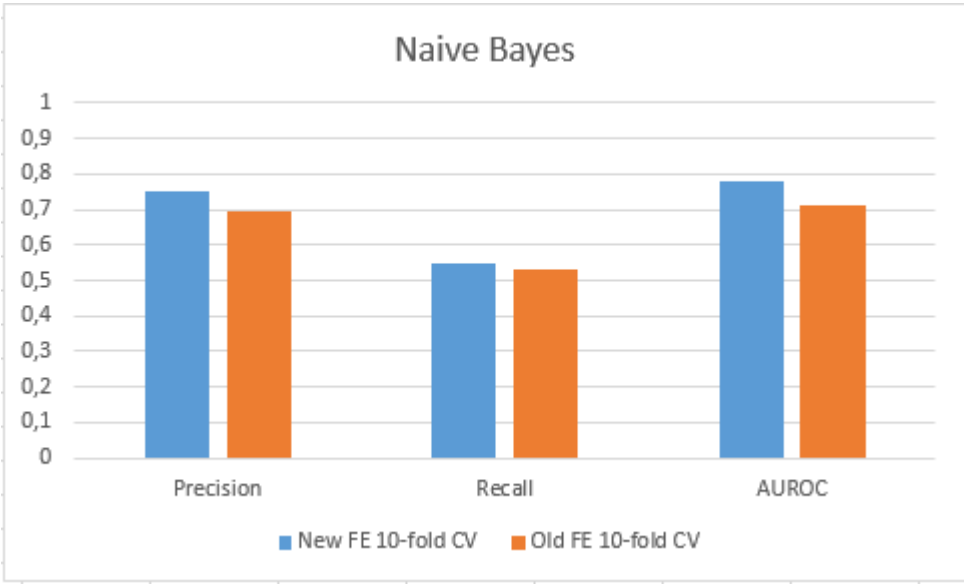


Figure 5.3: Precision, recall and AUROC statistics for the Naive Bayes classifier using 10-fold CV.

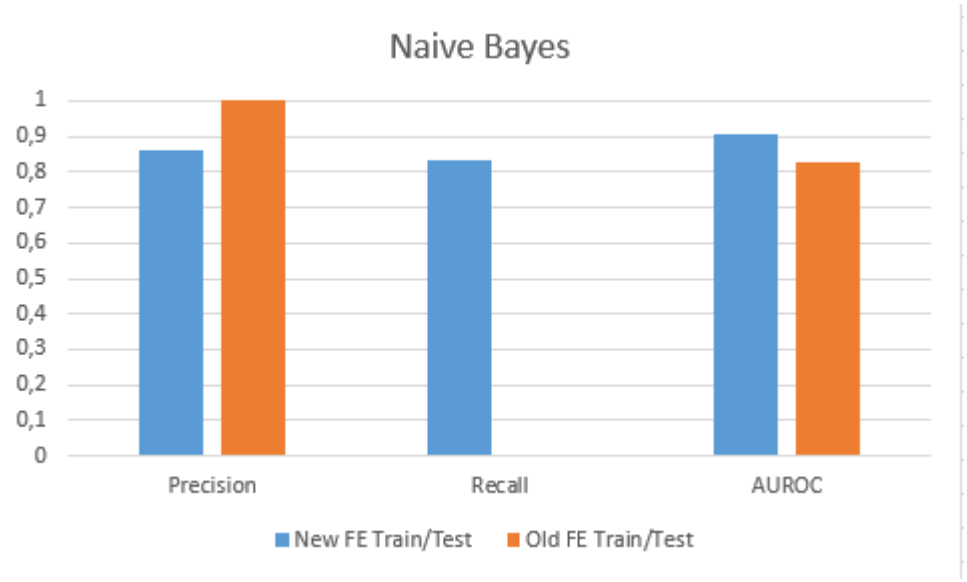


Figure 5.4: Precision, recall and AUROC statistics for the Naive Bayes classifier using Train/Test.

Accuracy wise, the new FE outperforms the old in every category, albeit by a small margin(Table 5.5). Also when comparing both new and old FE we can conclude that the new FE not only is it more accurate, but it is also stronger at separating both classes correctly(Fig. 5.4). When comparing using Train/Test we notice that the precision is one. However the recall is almost zero, making the precision value very misleading (Table A.4). In order to understand this we have to look at the respective confusion matrix (Table 5.6). The reason for this being the classifier only predicted 1 positive case out of 3000 total cases. The only positive prediction was correct, thus making its

precision value one. But it should have predicted 2999 more cases making the recall value almost zero.

		Predicted	
		Negative	Positive
Actual	Negative	3000	0
	Positive	2999	1

Table 5.6: Confusion matrix for the Old FE using Naive Bayes classifier and Train/Test.

5.3.2 Random Forest

After running the grid search the following parameters were obtained with the highest rating of success:

	New FE	Old FE
Max Depth	9	9
10-fold CV Accuracy	76,97%	75,73%

Table 5.7: Parameters determined for the Random Forest classifier using 10-fold CV.

	New FE	Old FE
Max Depth	2	4
Train/Test Accuracy	72.9%	68.4%

Table 5.8: Parameters determined for the Random Forest classifier using Train/Test.

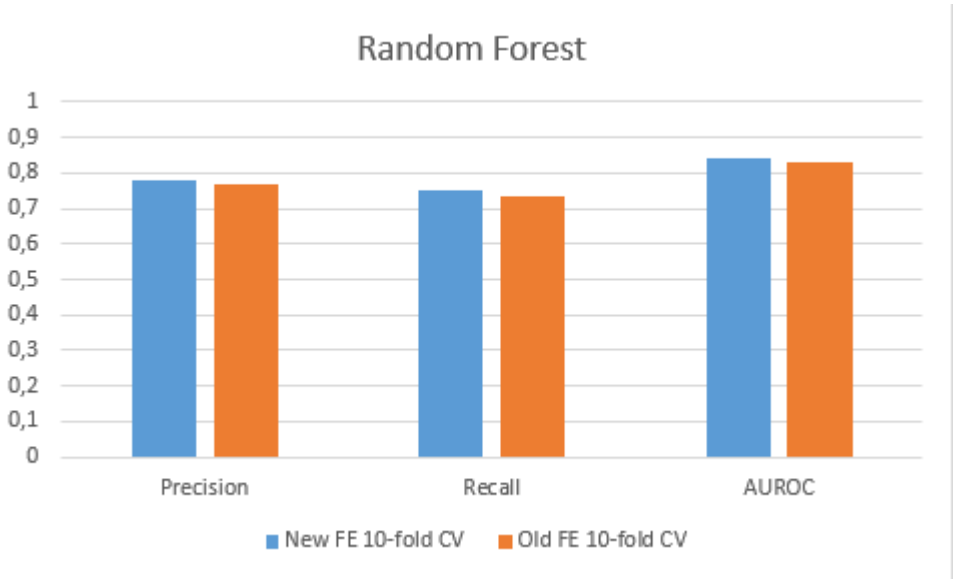


Figure 5.5: Precision, recall and AUROC statistics for the Random Forest classifier using 10-fold CV.

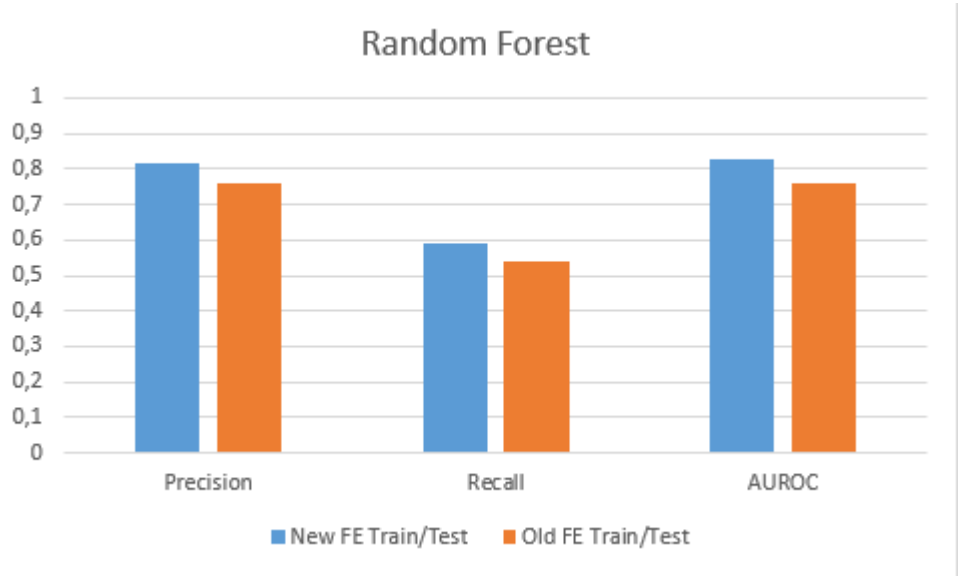


Figure 5.6: Precision, recall and AUROC statistics for the Random Forest classifier using Train/Test.

By comparing both the 10-fold CV results as well as the Train/Test, we can see that the new FE outperforms the old by a small margin (Table 5.7, 5.8). We can also see that the new FE obtained a bigger margin using Train/Test even though it did not need to go as far in the trees to obtain the best accuracy result. The precision, recall and AUROC results also show the new FE faring better overall, consolidating its advantage other than accuracy (Fig. 5.5, 5.6).

5.3.3 AdaBoost

	New FE	Old FE
Number of Iterations	10	9
Seed	0	0
Weight Threshold	90	95
10-fold CV accuray	75.69%	74.54%

Table 5.9: Parameters determined for the AdaBoost classifier using 10-fold CV.

	New FE	Old FE
Number of Iterations	8	8
Seed	0	0
Weight Threshold	93	95
Train/Test accuracy	78.22%	74.02%

Table 5.10: Parameters determined for the AdaBoost classifier using Train/Test.

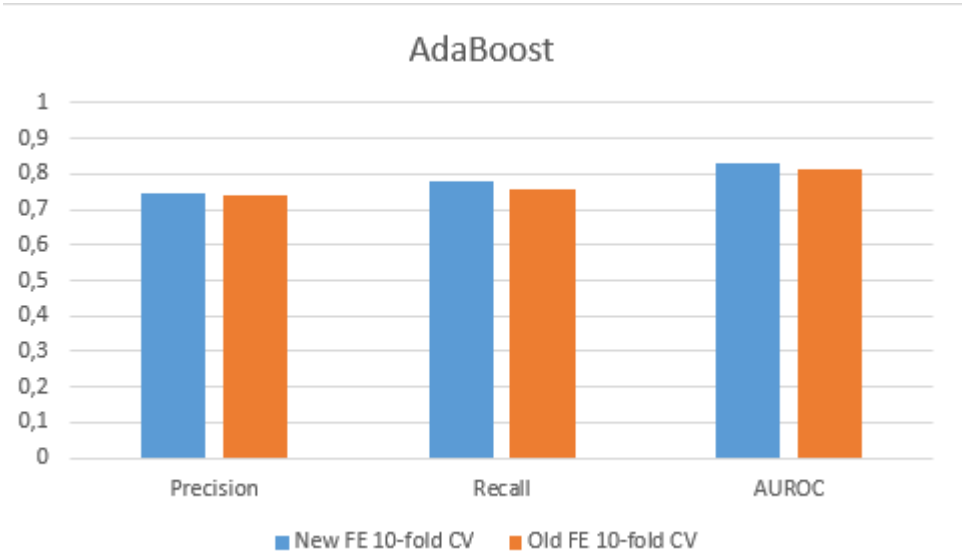


Figure 5.7: Precision, recall and AUROC statistics for the AdaBoost classifier using 10-fold CV.

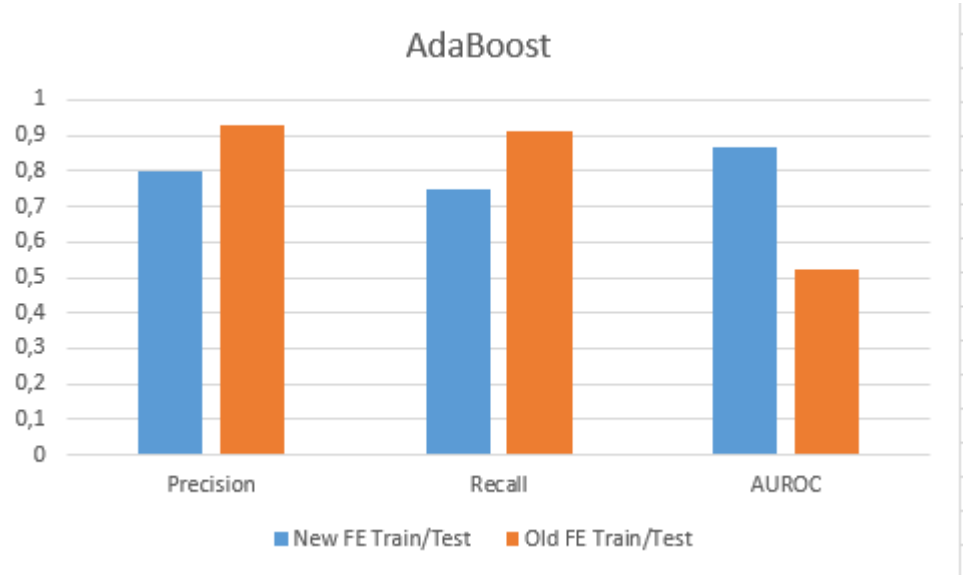


Figure 5.8: Precision, recall and AUROC statistics for the AdaBoost classifier using Train/Test.

Comparing both feature extractors by accuracy we can see that the new FE is overall superior using 10-fold CV (Table 5.9). This superiority is also maintained through precision, recall and AUROC comparison metrics (Fig. 5.7). Looking at Train/Test we can see that accuracy wise the new FE is also superior to the old (Table 5.10). However both precision and recall are higher for the old FE (Fig. 5.8), meaning that from all the positive cases it predicts, it does so more precisely. But a higher AUROC value from the new FE tells us that overall, despite having more false positives, it still predicts more positive cases correctly.

5.3.4 Real AdaBoost

	New FE	Old FE
Number of Iterations	8	10
Seed	0	0
Weight Threshold	98	99
10-fold CV accuray	75.55%	75.54%

Table 5.11: Parameters determined for the Real AdaBoost classifier using 10-fold CV.

	New FE	Old FE
Number of Iterations	6	6
Seed	0	0
Weight Threshold	100	95
Train/Test accuracy	81.05%	66.7%

Table 5.12: Parameters determined for the Real AdaBoost classifier using Train/Test.

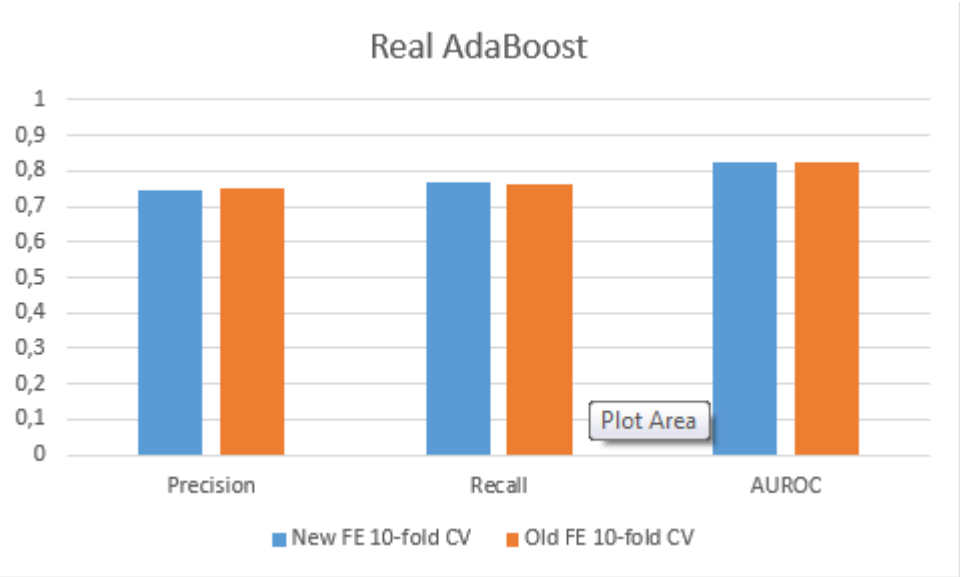


Figure 5.9: Precision, recall and AUROC statistics for the Real AdaBoost classifier using 10-fold CV.

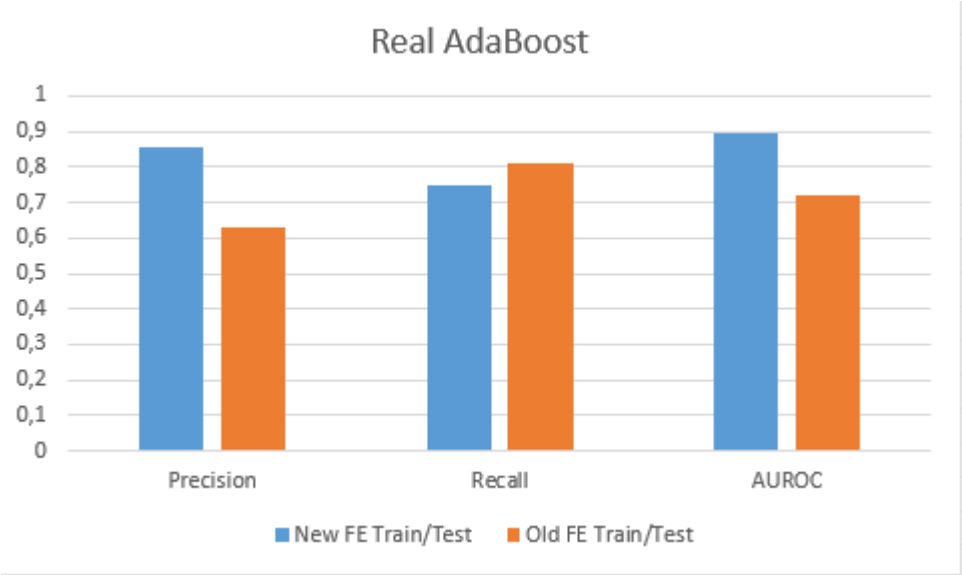


Figure 5.10: Precision, recall and AUROC statistics for the Real AdaBoost classifier using Train/Test.

Starting with 10-fold CV we can see that the new FE outperforms the old not only in accuracy (Table 5.11), but also in precision, recall and AUROC metrics (Fig. 5.9) by a very small margin. However, looking at the Train/Test performance we can see that the new FE has a big margin over the old FE regarding accuracy (Table 5.12). The precision and AUROC are also bigger for the new FE unlike the recall which is better for the old (Fig. 5.10). Meaning that the new FE is better at predicting positive cases, while the old FE managed to correctly identify more positive cases but incorrectly identified more negative instances.

5.3.5 Support Vector Machine

We will be varying only the kernel, the existing options are: Poly, Normalized Poly, Puk and Radial Basis Function. There were more variable parameters, but unfortunately due to some issues we ended up only varying the kernel and using default values for the rest. We believe it should be mentioned that we have multiple results with different parameters. But due to the lengthy execution time of this classifier, we could not obtain the classification results for all data sets. In particular, the data set which contains all the features of the old FE. So in order to maintain the consistency between the results, we decided to present the results with no parameter variation for all data sets.

	New FE	Old FE
Kernel	Poly	Poly
Accuracy	78,74	81,29

Table 5.13: Parameters determined for the SVM classifier using 10-fold CV.

	New FE	Old FE
Kernel	Poly	Poly
Accuracy	83,32	85,33

Table 5.14: Parameters determined for the SVM classifier using Train/Test.

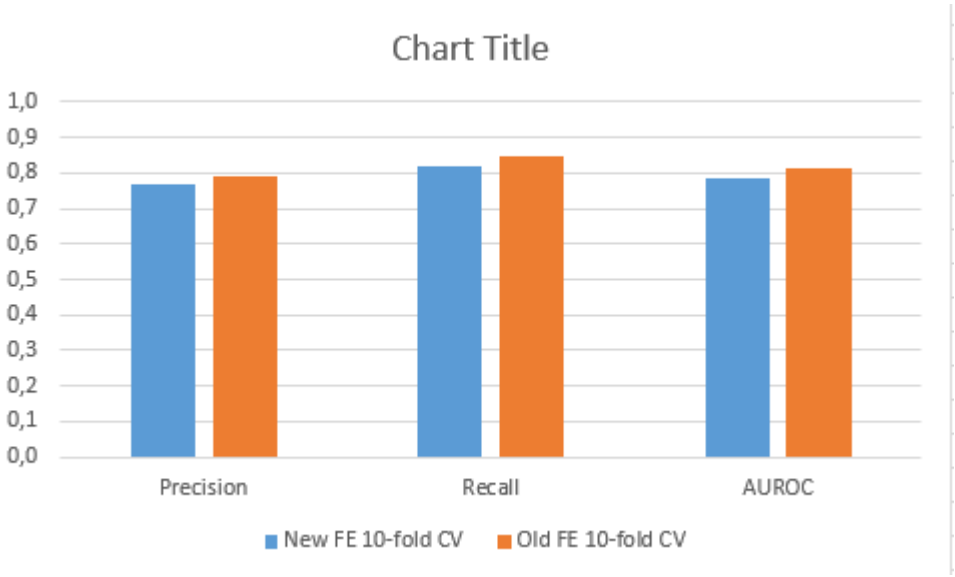


Figure 5.11: Precision, recall and AUROC statistics for the SVM classifier using 10-fold CV.

Figure 5.12: Precision, recall and AUROC statistics for the SVM classifier using Train/Test.

Using the settings described above, we can clearly see that the best performing kernel is poly. When

comparing the accuracy values, the old FE surpasses the new with a slight lead using 10-fold CV. When using Train/Test the old FE also performs better than the new. As for precision, recall and AUROC metrics using 10-fold CV, the old FE edges out the new by a small margin (Fig. 5.11). This indicates the old FE is better at predicting any class against the new FE. When looking at the Train/Test method, the same also happens, with the old FE outperforming the new by a slight margin in all the metrics.

5.4 Feature Selection

Having a big feature set is not always the better option. It is important to initially have a large feature set to try and include as many image discrimination options as possible. However this methodology may present some problems which we aim to reduce with feature selection. These consist in reducing overfitting, since reducing redundant data leads to making fewer decisions based on noise(curse of dimensionality[2]). Less noise may help to improve the performance of learning models. Improving accuracy. And finally, significantly improve processing time, once again, by reducing the data being processed, the classification algorithms will train faster.

Our selection processed was achieved with the help of Weka[3]. Weka is a data mining software developed in Java, it has multiple feature selection algorithms which we used in our feature set. These algorithms are split into two categories, the attribute evaluator and the search method. The attribute evaluator is the technique by which the features are assessed. For example, by considering the individual predictive ability of each feature along with the degree of redundancy between them. The search method is how the space of possible features is searched. For example, ranking features by their individual evaluations. We combine both of these categories to achieve a reduced feature set.

We tried to have parity between both old and new FE feature sets, unfortunately some of the methods couldn't be applied to both, since the old FE feature set presented some errors when running some particular feature selection methods in Weka. This means the new FE will have a few more selection methods available.

We also decided to experiment more with the feature selection process, after putting together all the selected features we combined them into two different datasets. These are the *most selected*, which consist on the features that were picked more often by the feature selection methods. And also the *all selected*, which consists of all the selected features by the multiple feature selection methods. Our goal with the most selected features is simple. All the feature selection methods have a unique logic, but if there are some features that are picked more often than others perhaps we can obtain interesting classification results with them. The combination of all the selected features is to expand our dataset with more features, this way we can observe if having more features is beneficial.

The attribute evaluators used are the following:

[2]The curse of dimensionality refers to the fact that as the dimensionality of data increases, its volume increases as well. This leads to the data becoming sparse and dissimilar. This is an obstacle to machine learning algorithms, since they rely on analyzing groups with similar properties.

[3]http://www.cs.waikato.ac.nz/ml/weka/

- CorrelationAttributeEval(CAE): Evaluates the quality of an attribute by measuring the correlation between the attribute and the respective class.
- CfsSubsetVal: Seeks subsets that have a higher correlation with the class value and low correlation with each other (Hall, 1998).
- GainRatioAttributeEval(GRAE): Evaluates the quality of an attribute by measuring the gain ratio relatively to the class.
- InfoGainAttributeEval(IGAE): Evaluates the quality of an attribute by measuring the information gain relatively to the class.
- OneRAttributeEval(ORAE): Evaluates the quality of an attribute by using the OneR classifier (Holte, 1993).
- PrincipalComponents(PC): Utilizes the Principal component analysis (PCA) to analyze and transform the data. PCA is a technique utilized which highlights variation and finds strong patterns in a dataset.
- ReliefAttributeEval(RFAE): Evaluates the quality of an attribute by doing multiple samplings of an instance and considering the value of the given attribute for the nearest instance of both the same and opposite class (Kira and Rendell, 1992), (Kononenko, 1994), (Robnik-Sikonja and Kononenko, 1997).
- SymmetricaluncertAttributeEVal(SUAE): Evaluates the quality of an attribute by measuring the symmetrical uncertainty regarding the class.
- WrapperSubsetEval(WSE): Assesses subsets using a self specified classifier with a n-fold CV. We used the default ZeroR with a 5-fold CV (Kohavi and John, 1997).

The search methods used are the following:

- Ranker: Ranks attributes by their performances.
- BestFirst: Navigates the attribute subsets by greedy hillclimbing enhanced with a backtracking facility.

We utilized these methods to obtain reduced feature sets, up to 15 features per set (Table 5.15). Most of the attribute evaluators are restricted to a searching method, which explains why they are mapped one to one. The cases where there were more than one option of searching method for one attribute evaluator, resulted in the same features selected, therefore we discarded them.

Attribute Evaluator	Searching Method
CorrelationAttributeEval	Ranker
GainRatioAttributeEval	Ranker
InfoGainAttributeEval	Ranker
OneRAttributeEval	Ranker
Principal components	Ranker
ReliefAttributeEval	Ranker
SymmetricaluncertAttributeEVal	Ranker
CfsSubsetVal	BestFirst
WrapperSubsetEval	BestFirst

Table 5.15: The combination of attribute evaluators and searching methods we used to create the selection feature sets.

5.4.1 Naive Bayes

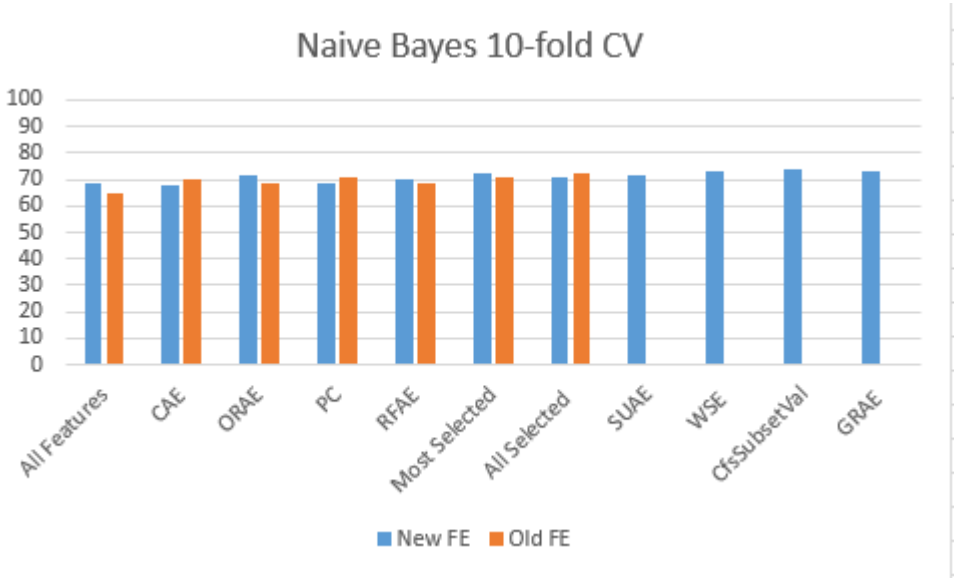


Figure 5.13: Accuracy for all datasets with the Naive Bayes classifier using 10-fold CV.

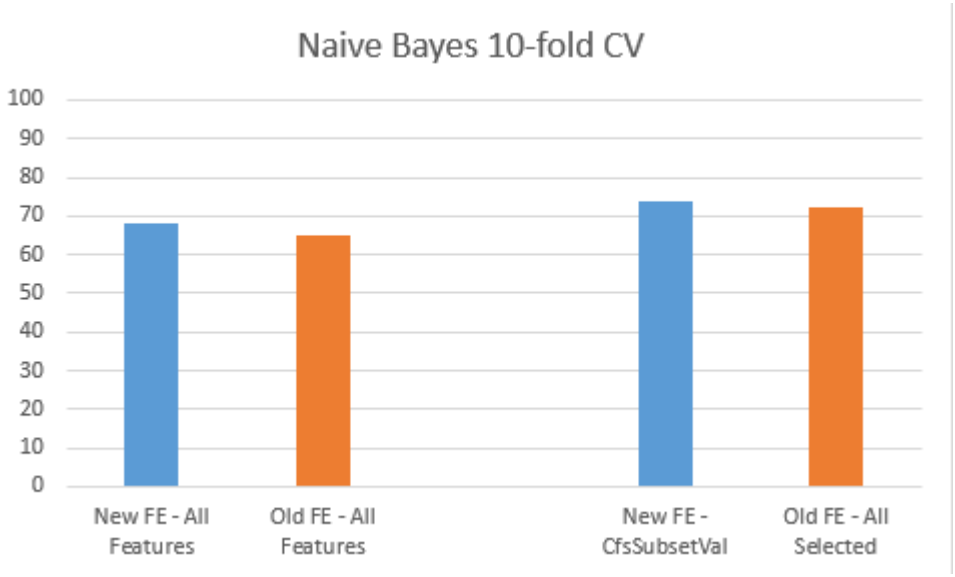


Figure 5.14: Accuracy comparison including all the features and the best feature selection method for this classifier using 10-fold CV.

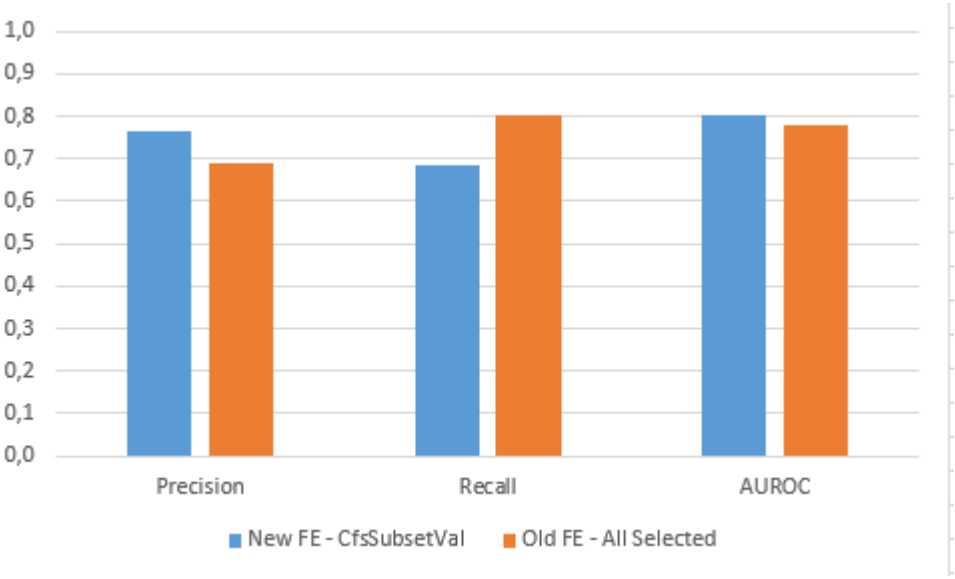


Figure 5.15: Precision, recall and AUROC statistics for the Naive Bayes classifier using 10-fold CV between the best feature sets found by the feature selection process.

As we can see, for the 10-fold CV we have an evenly distribution (Table A.1). From the 6 feature selection methods we can compare between old and new FE, 3 have higher accuracy for the old and 3 have higher accuracy for the new FE (all results can be seen in the appendix A). However if we compare the best feature sets between both old and new FE, we can see that the new FE comes on top (Fig. 5.14). Also, both selected feature sets outperform the feature combined ones of

each respective FE. Comparing the precision, recall and AUROC values between both top feature selection sets, we can see that once again the new FE predicts less false positives than the old FE, but the old FE identifies more positive instances correctly than the new (Fig. 5.15).

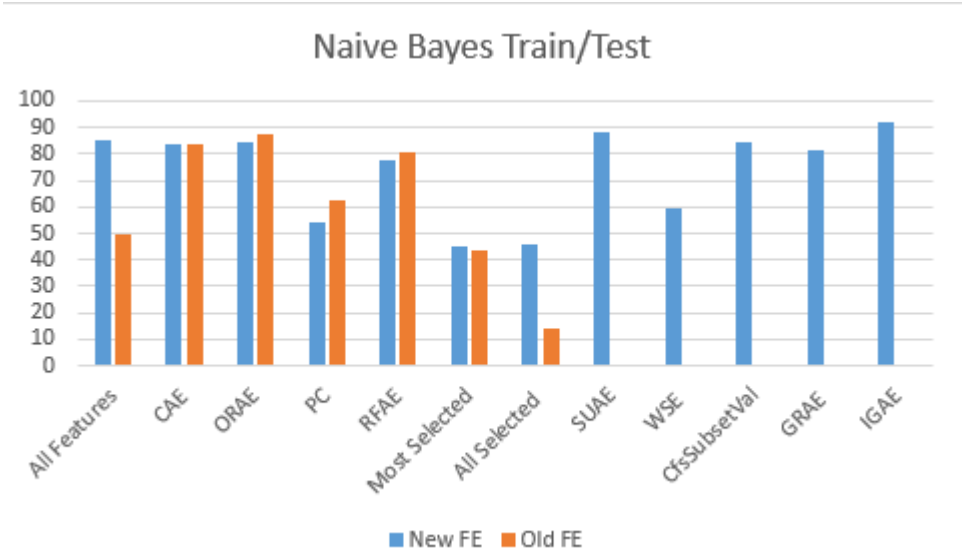


Figure 5.16: Accuracy for all datasets with the Naive Bayes classifier using Train/Test.

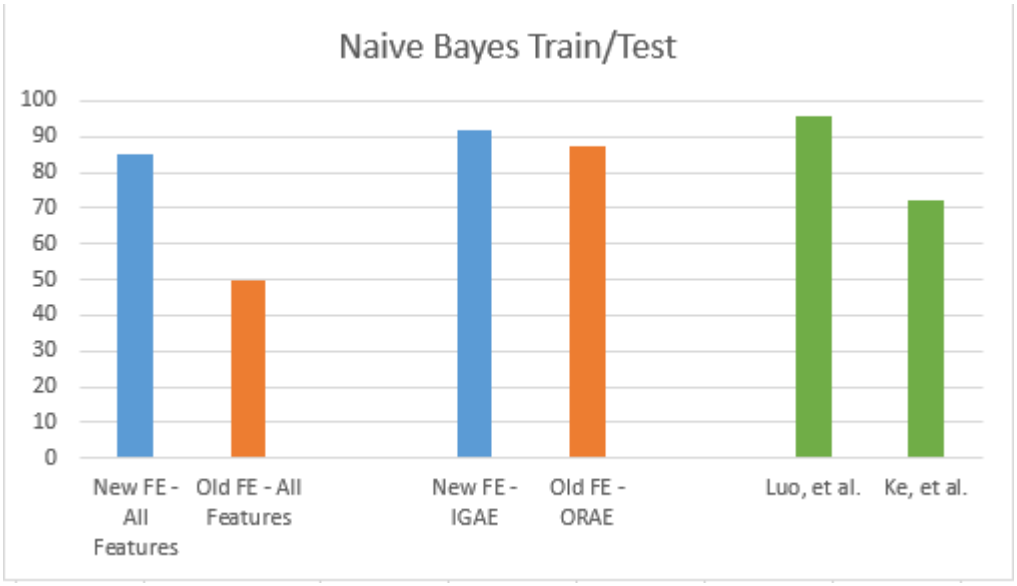


Figure 5.17: Accuracy comparison including all the features, the best feature selection method and similar studies for this classifier using 10-fold CV.

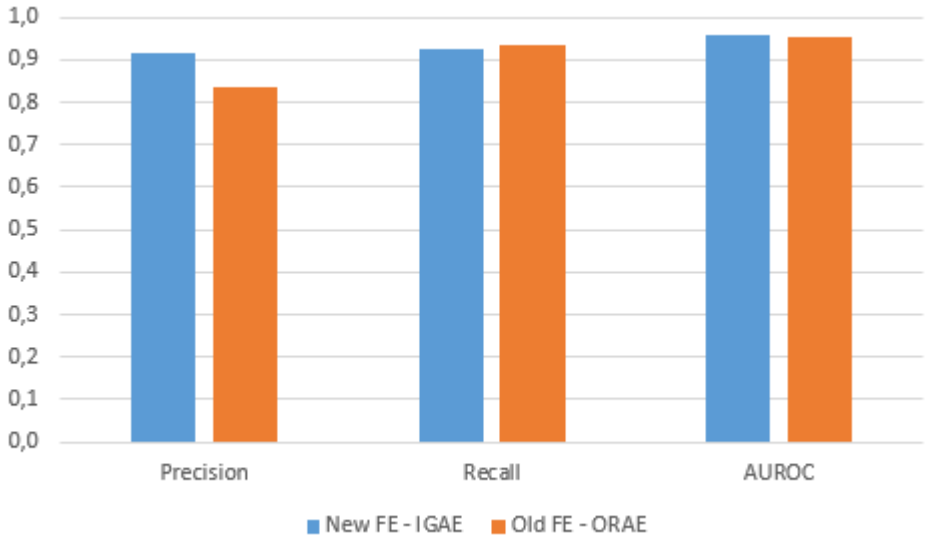


Figure 5.18: Precision, recall and AUROC statistics for the Naive Bayes classifier using Train/Test between the best feature sets found by the feature selection process.

When comparing both new and old FE when using the Train/Test data validation method, we can see that the accuracy distribution has bigger gaps between selection sets(Fig. 5.16). Comparing the methods directly, we see that once again the new FE has 3 methods that classify better than the old FE, and vice versa. When comparing the feature selection sets between the combined features set, we can observe that the new FE feature selection set performs slightly better (Fig. 5.17). Meanwhile the old FE has a bigger improvement from the feature selection set to the combined features set. When comparing with other studies which use the same approach, we can see that both new and old FE, are outperformed by Luo and Tang (2008), which achieve an overall accuracy of 96%. But when comparing with Ke et al. (2006), with an accuracy score of 72,2%, we can observe that the new FE outperforms it with both the combined features and best selection method. While the old FE outperforms it only with the best feature selection method. Looking at the precision, recall and AUROC metrics we can see once again the new FE with a lead over the old for the precision and AUROC metrics, while the old FE has a slight better performance for the recall metric.

5.4.2 Random Forest

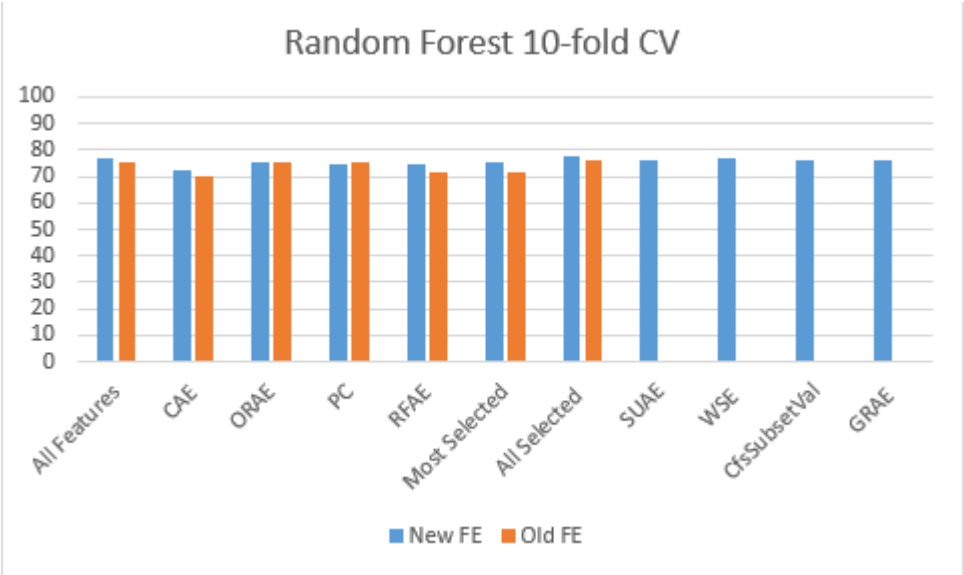


Figure 5.19: Accuracy for all datasets with the Random Forest classifier using 10-fold CV.

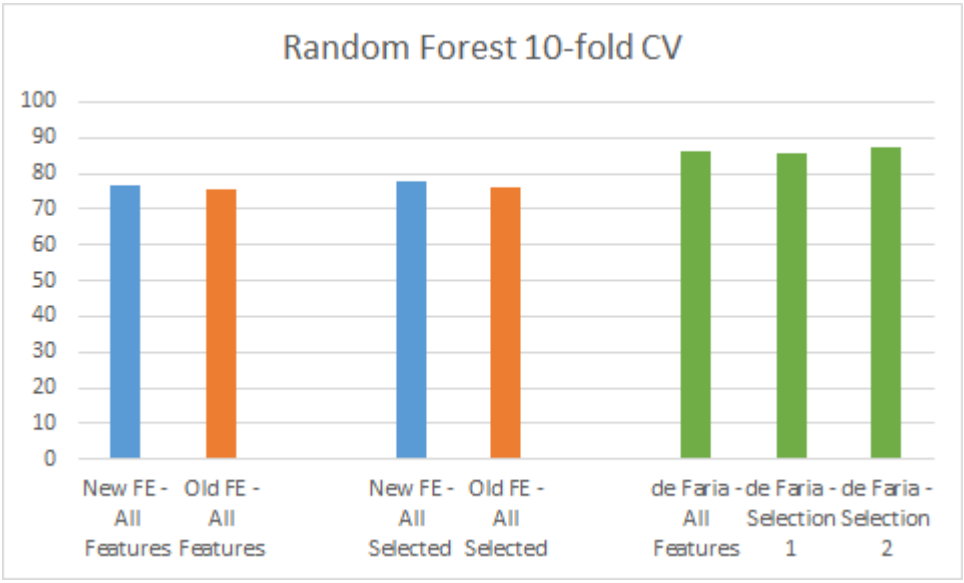


Figure 5.20: Accuracy comparison including all the features, the best feature selection method and similar studies for this classifier using 10-fold CV.

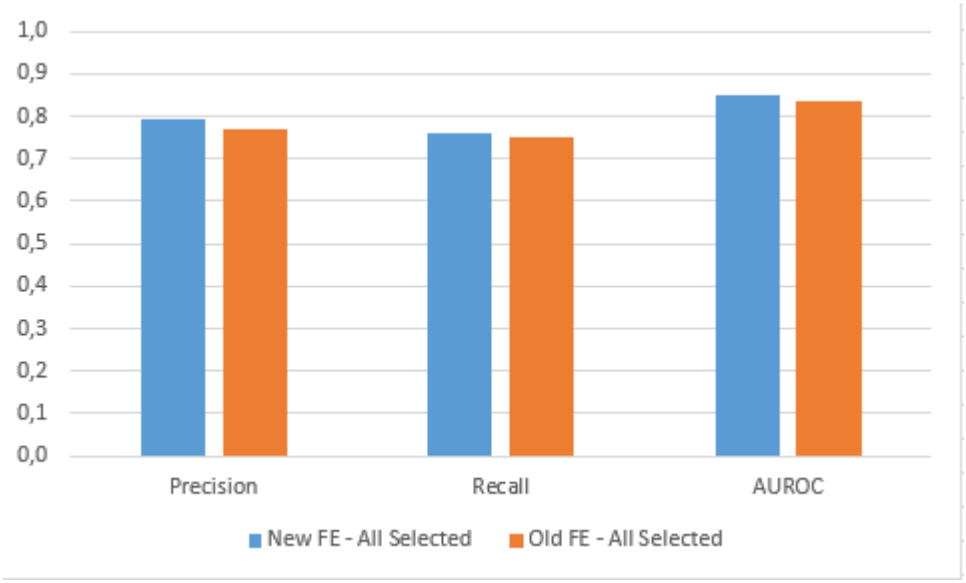


Figure 5.21: Precision, recall and AUROC statistics for the Random Forest classifier using 10-fold CV between the best feature sets found by the feature selection process.

Starting with the 10-fold CV data validation method, we compare all features sets between both new and old FE (Fig. 5.19). As we can see, the distribution is not very different, but favors the new FE overall with the exception being the principal components feature set (all the results can be seen in the appendix B). When comparing the new and old FE with other similar studies (Fig. 5.20), we can observe that both feature extractors are outperformed, our comparing study (de Faria, 2012) presents accuracy values of 86,33% for all the features combined, 85,4% for the first feature selection method and 87,1% for the second feature selection method. When looking at the precision, recall and AUROC values (fig 5.21) we can see that the new FE leads on all metrics by a slight margin, presenting no downsides.

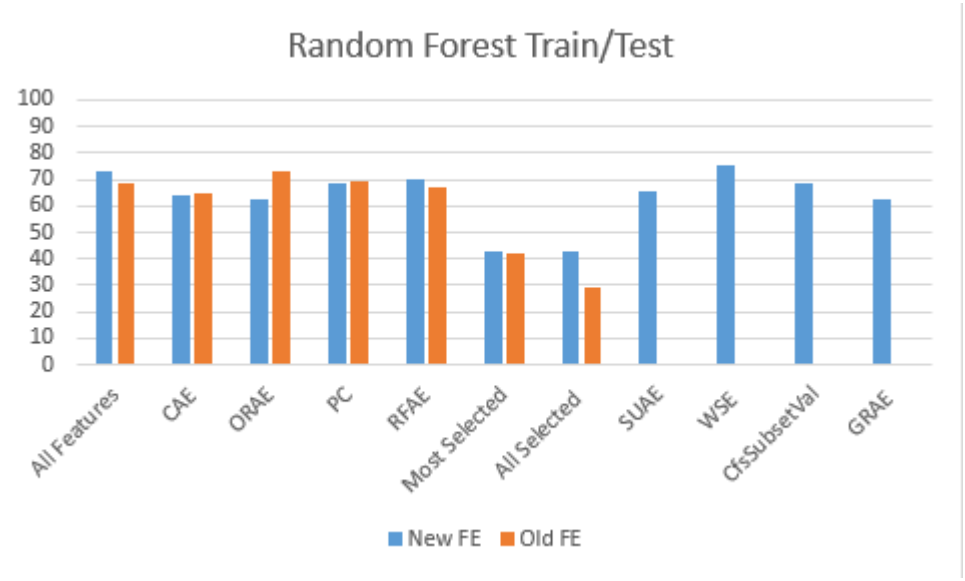


Figure 5.22: Accuracy for all datasets with the Random Forest classifier using Train/Test.

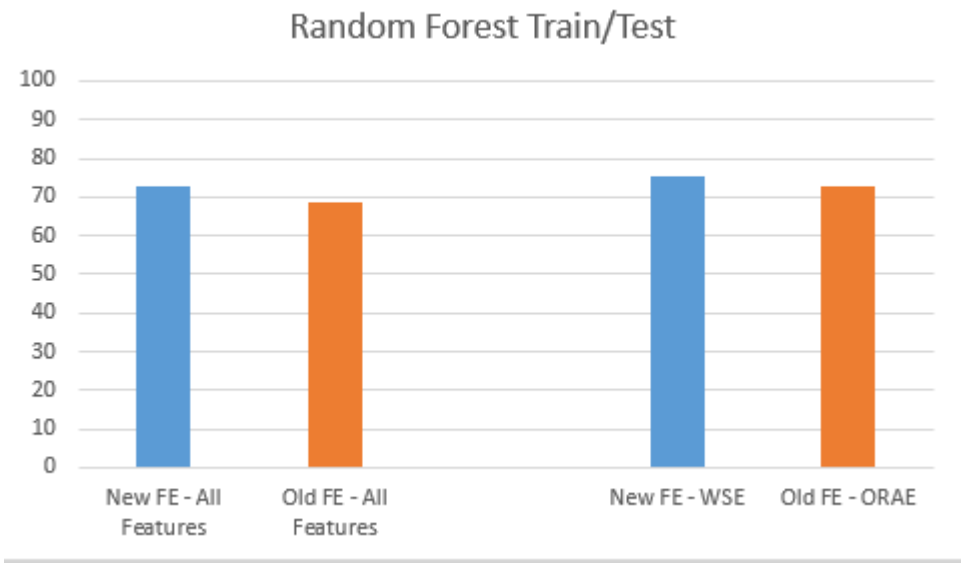


Figure 5.23: Accuracy comparison including all the features and the best feature selection method for this classifier using Train/Test.

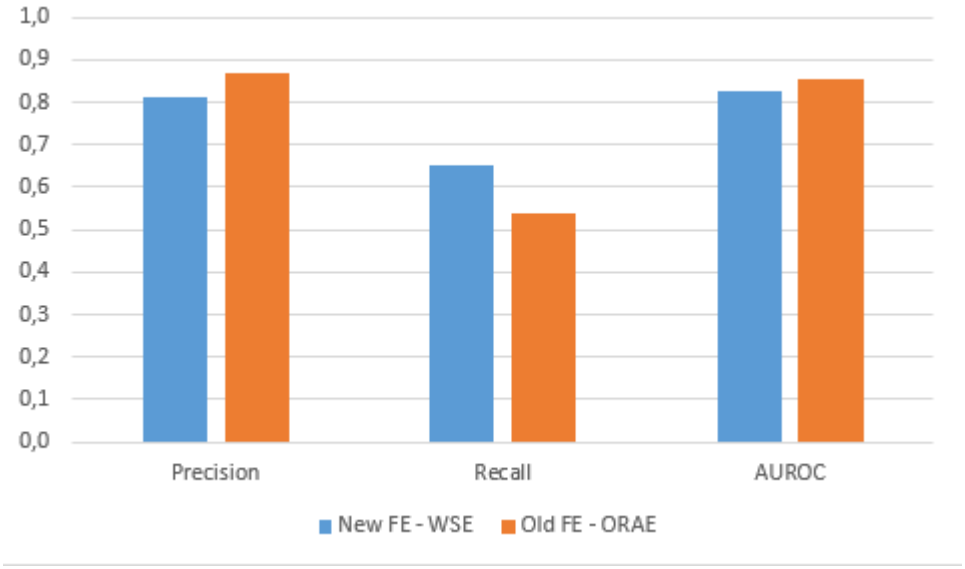


Figure 5.24: Precision, recall and AUROC statistics for the Random Forest classifier using Train/Test between the best feature sets found by the feature selection process.

Moving on to the Train/Test data validation method, we can observe the disparity between results is bigger (Fig. 5.22). Comparing the results between both feature extractors, we have 3 methods for each FE having better classification results. And once we compare the best selection method for each FE we come to the conclusion, once again, that the new FE comes out on top (Fig 5.23). However, when comparing the precision, recall and AUROC metrics we come to realize that the best selection method for the old FE is better at predicting positive instances, although it does not predict as many overall as the method from the new FE (Fig. 5.24).

5.4.3 AdaBoost

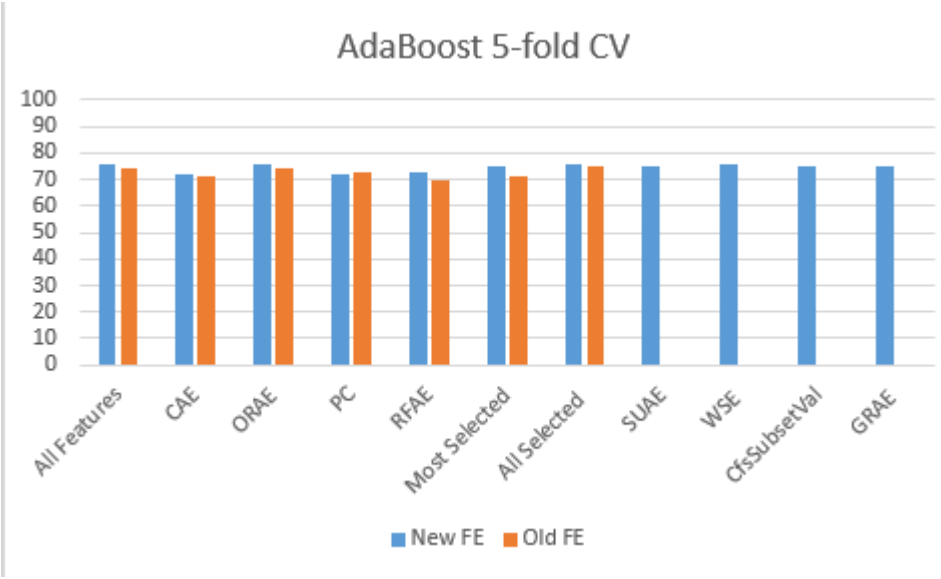


Figure 5.25: Accuracy for all datasets with the AdaBoost classifier using 10-fold CV.

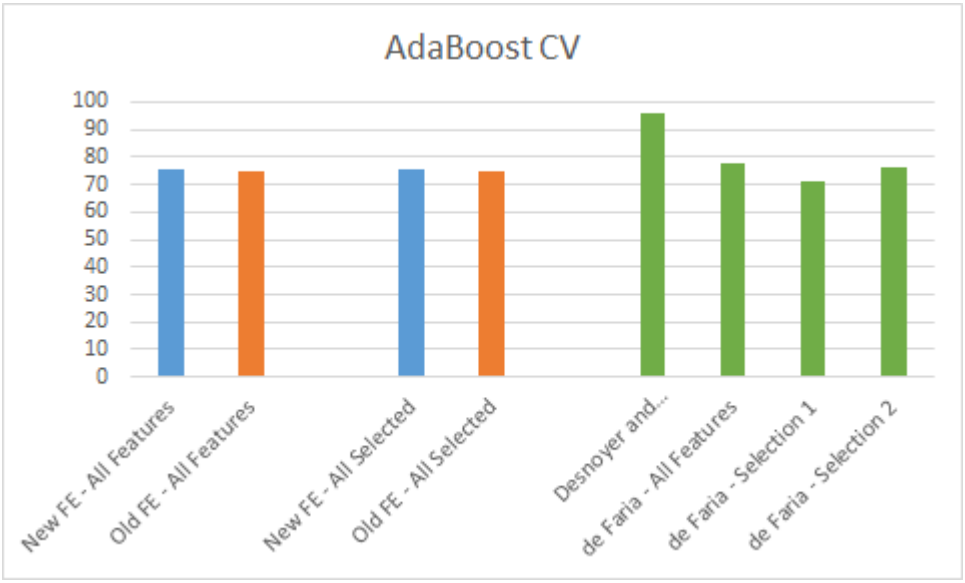


Figure 5.26: Accuracy comparison including all the features, the best feature selection method and similar studies for this classifier using CV.

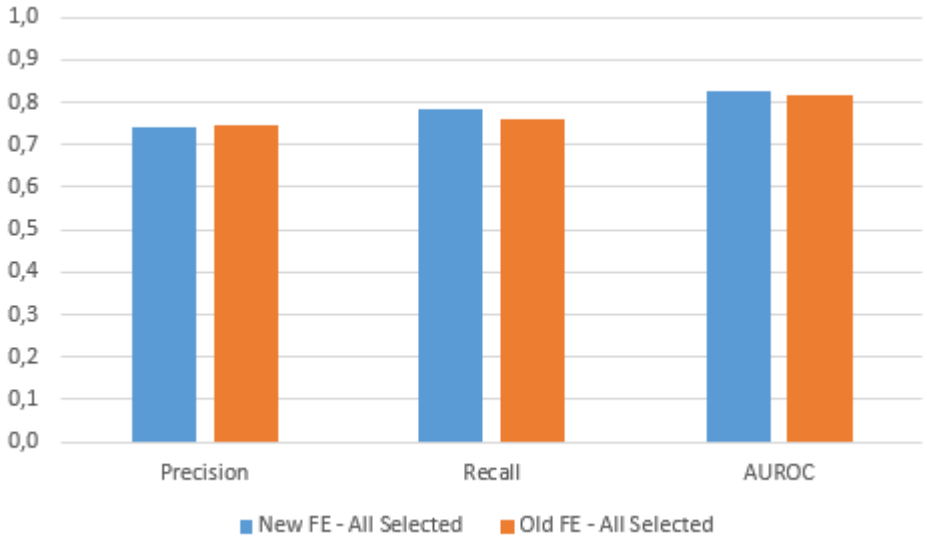


Figure 5.27: Precision, recall and AUROC statistics for the AdaBoost classifier using 5-fold CV between the best feature sets found by the feature selection process.

Looking at the overall comparison between both new and old feature extractors (Fig. 5.25), we can see that the new FE performs better in all comparison methods. The only exception being principal components (all results can be seen in the appendix C). Comparing both of the best feature selection sets, we can see that the new FE performs better than the old. But when comparing with similar works, we can observe that Desnoyer and Wettergreen (2010) performs the best with 96% accuracy, followed by de Faria (2012) with different data sets. The combined features present 77,6% accuracy, then two different feature selection sets, with 71,4% and 76,2% accuracy respectively. Meaning the new and old feature extractors only outperform the first feature selection set presented by Faria. When comparing precision, recall and AUROC metrics between the best feature sets we can see that precision is the same for both, while recall and AUROC slightly favoring the new FE. Meaning that the new FE predicts correctly more positive instances.

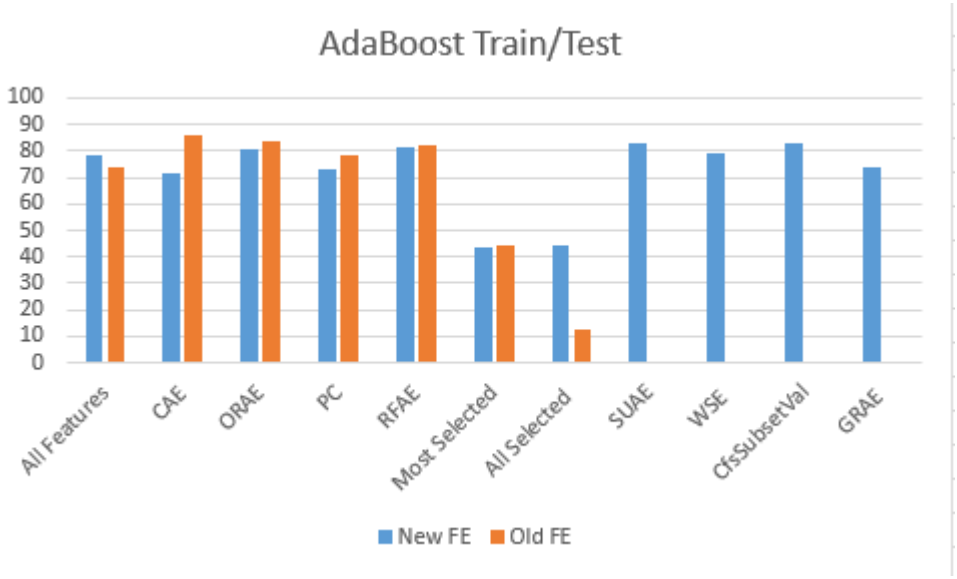


Figure 5.28: Accuracy for all datasets with the AdaBoost classifier using Train/Test.

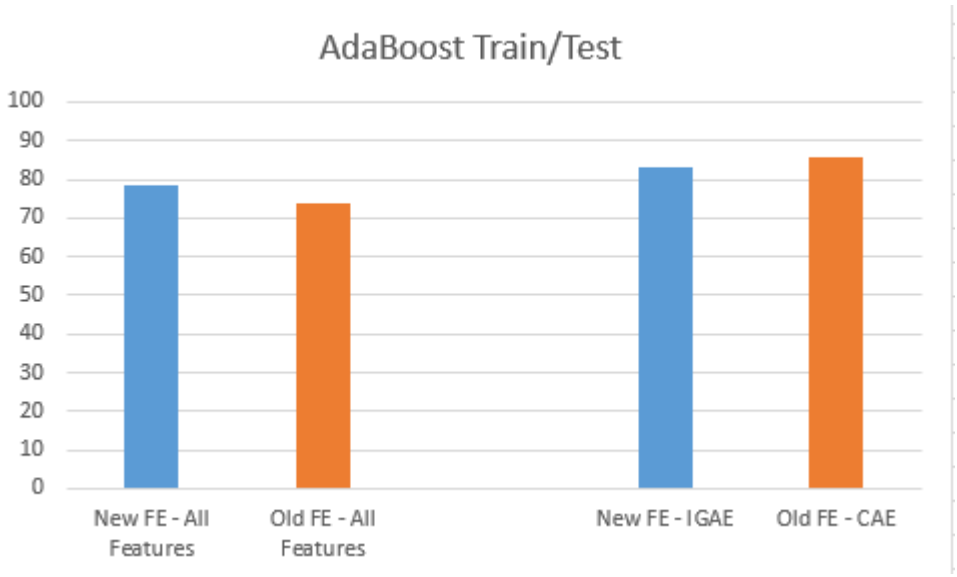


Figure 5.29: Accuracy comparison including all the features and the best feature selection method for this classifier using Train/Test.

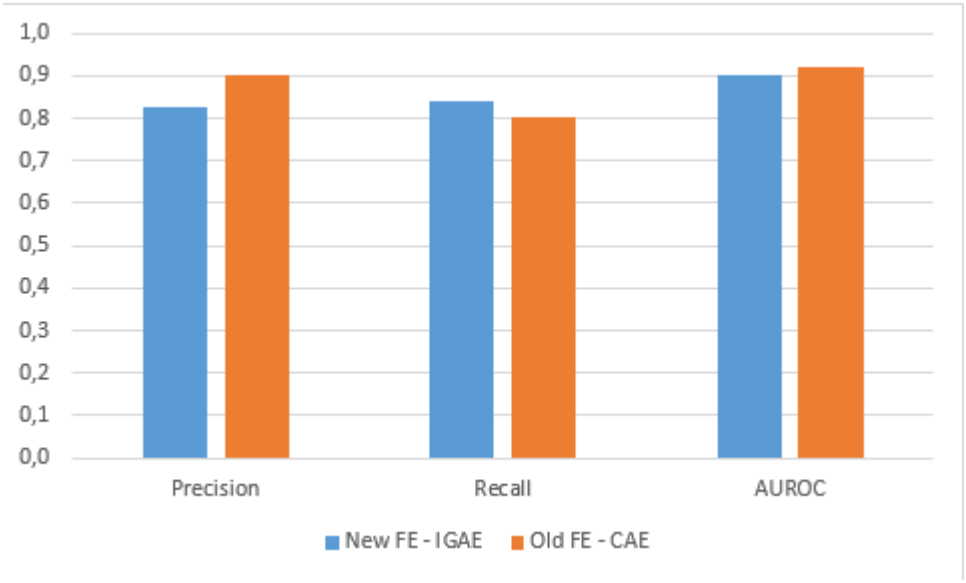


Figure 5.30: Precision, recall and AUROC statistics for the Random Forest classifier using Train/Test between the best feature sets found by the feature selection process.

When using Train/Test though, the results are completely different (5.28). We can observe that the old FE performs better overall with their selected feature sets. The only exception being the all selected feature set, where the new FE performs better. When comparing the best performing sets from each FE, we can observe that although the old FE performs worse with its combined features, when comparing the selected sets it actually performs better than the new FE (Fig. 5.29). Looking at the precision, recall and AUROC metrics we can see that both the precision and AUROC are higher for the old FE while the recall is higher for the new FE. Meaning that the best performing feature selection method for the old FE is more accurate at predicting positive instances, while the method for the new FE predicts more positive instances overall.

5.4.4 Real AdaBoost

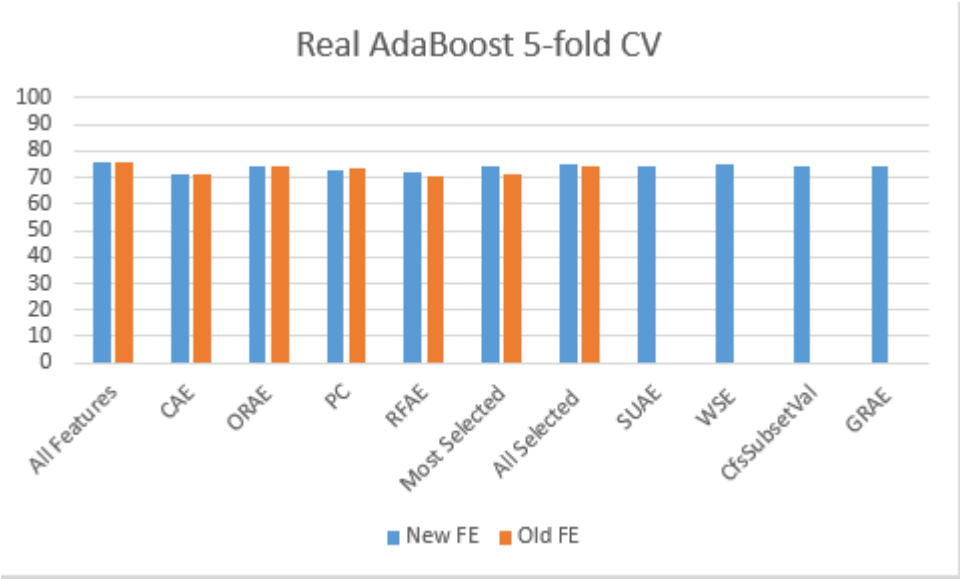


Figure 5.31: Accuracy for all datasets with the Real AdaBoost classifier using 10-fold CV.

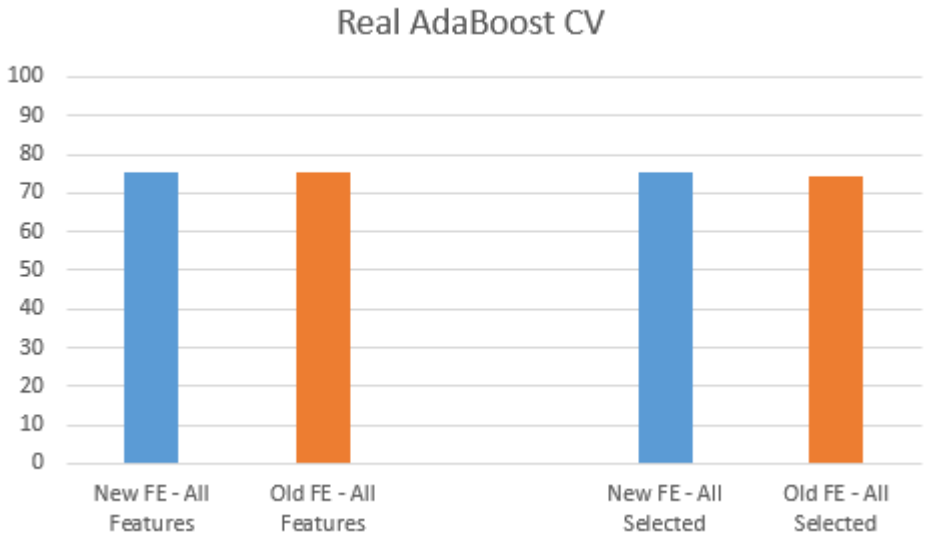


Figure 5.32: Accuracy comparison including all the features and the best feature selection method for this classifier using 5-fold CV.

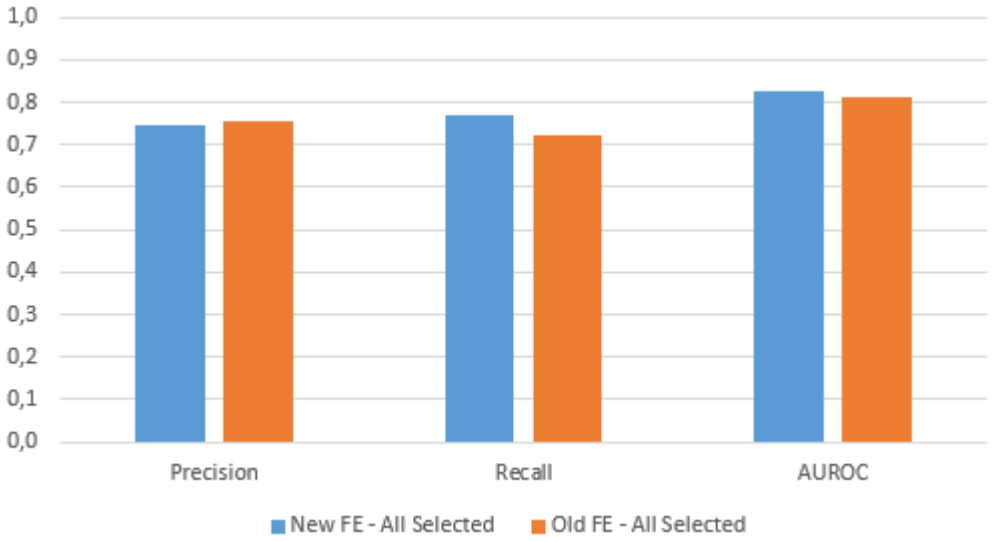


Figure 5.33: Precision, recall and AUROC statistics for the Real AdaBoost classifier using 5-fold CV between the best feature sets found by the feature selection process.

Analyzing the Real AdaBoost classifier for all the feature selection data sets, we can see there is an even distribution (Fig. 5.31). A direct comparison shows us 3 selection methods are performing better for each FE (all results can be seen in the appendix D). When comparing the best feature selection set for each FE we can see that the new FE comes out on top (Fig. 5.32). When comparing the precision, recall and AUROC metrics, the conclusion is that the new FE is not as accurate at predicting positive classes but identifies more of them.

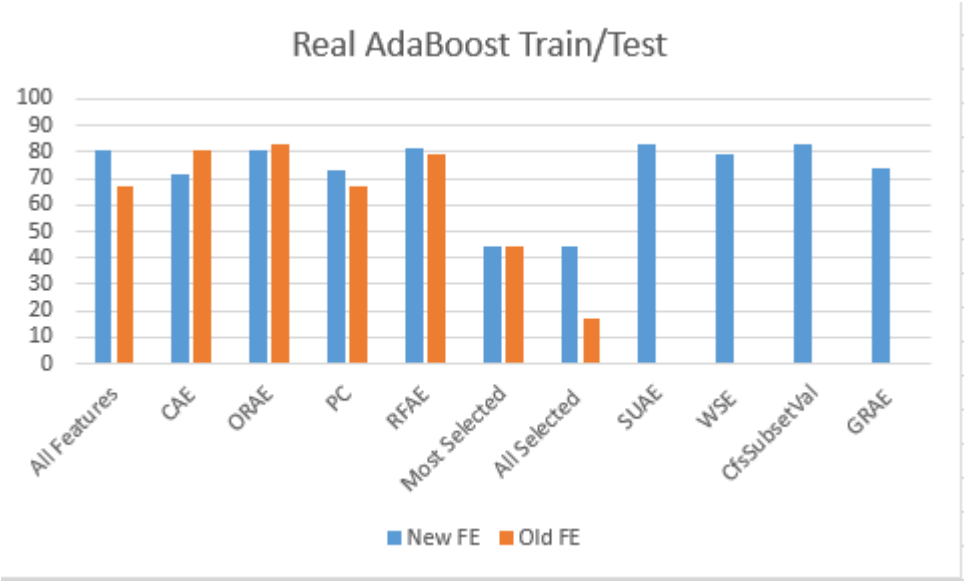


Figure 5.34: Accuracy for all datasets with the Real AdaBoost classifier using Train/Test.

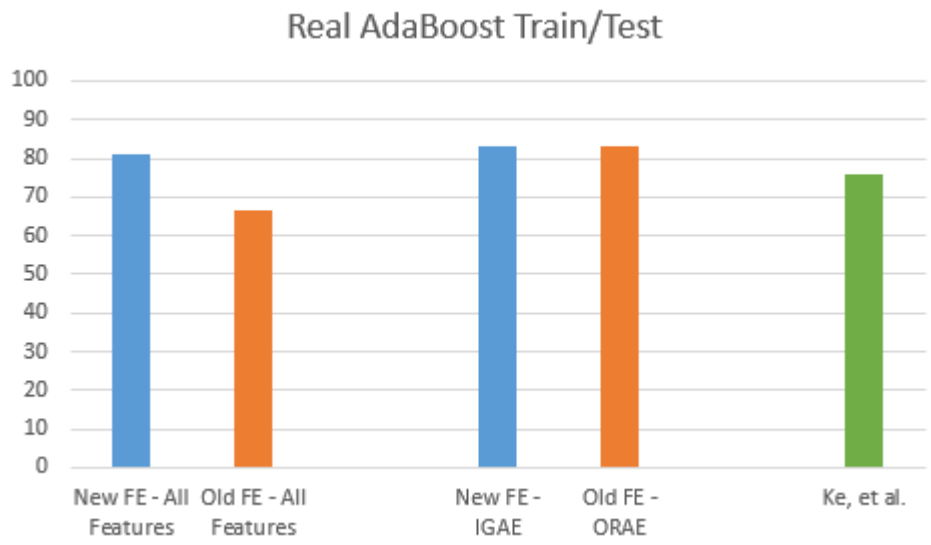


Figure 5.35: Accuracy comparison including all the features, the best feature selection method and similar studies for this classifier using Train/Test.

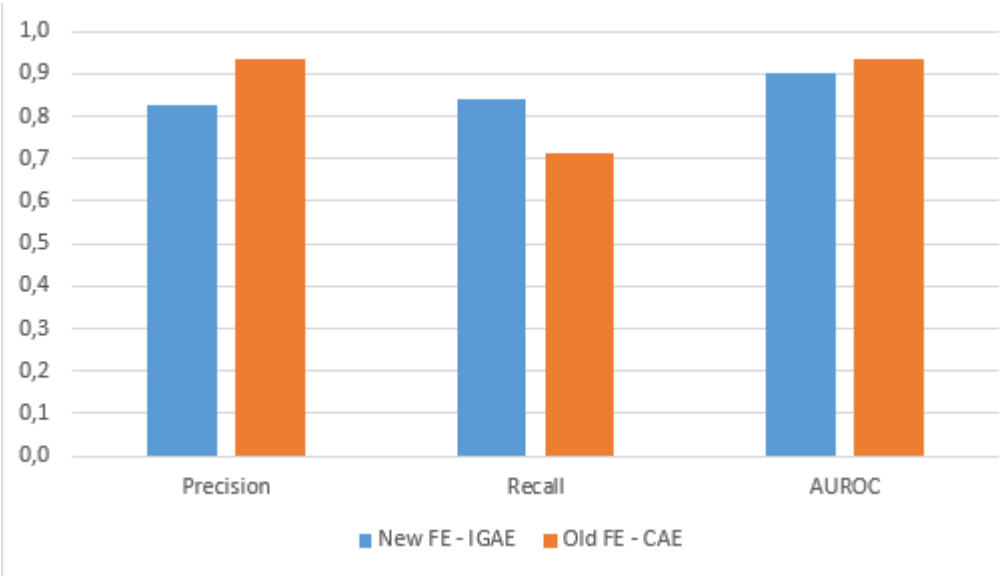


Figure 5.36: Precision, recall and AUROC statistics for the Random Forest classifier using Train/Test between the best feature sets found by the feature selection process.

When comparing the Train/Test data validation method for both new and old feature extractors, we can observe that the accuracy distribution is more uneven (Fig. 5.34). It also maintains a similar advantage, with 3 feature selection sets performing better for both extractors. For further comparison, we have the research provided by Ke et al. (2006), which reports an accuracy value of 76%. Comparing to new and old feature extractors, we see that except for the combined features of

the old FE, all other feature sets have better accuracy results. Precision, recall and AUROC metrics shows us that the old FE is more accurate when predicting positive instances, but does not predict them as much as the old FE overall.

5.4.5 Support Vector Machine

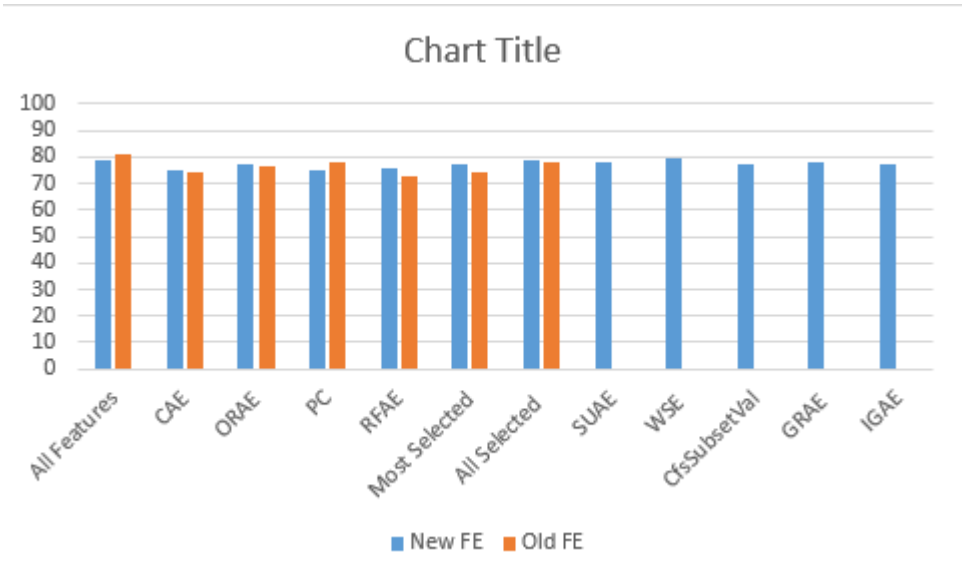


Figure 5.37: Accuracy for all datasets with the SVM classifier using 10-fold CV.

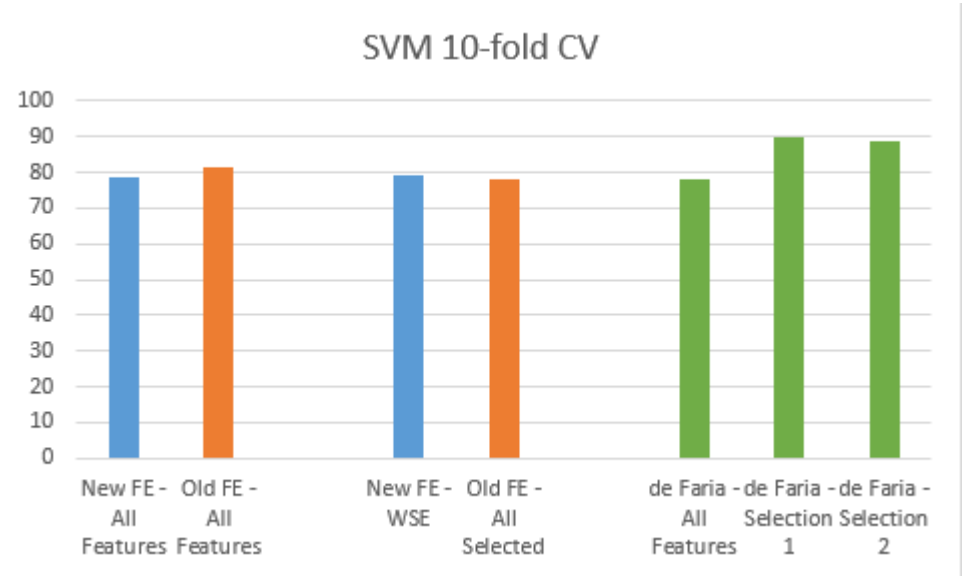


Figure 5.38: Accuracy comparison including all the features and the best feature selection method for SVM using 10-fold CV.

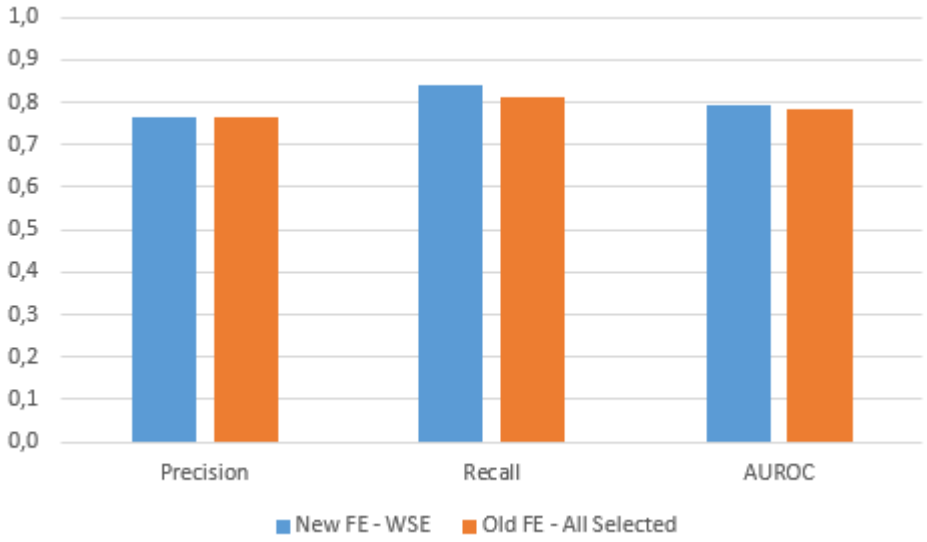


Figure 5.39: Precision, recall and AUROC statistics for the SVM classifier using 10-fold CV between the best feature sets found by the feature selection process.

Looking at the SVM classifier for all the feature selection data sets, once again there is an even distribution for all accuracy values (Fig. 5.37). Comparing between new and old feature extractors, the results show us that the old FE only outperforms the new for the PC feature set and with all features combined (all results can be seen in the appendix E). By switching our attentions to the best feature set for both extractors, we can see that even though the old FE performs better with all the features combined, when we reducing the feature set, the new FE actually performs better. When comparing to other studies, comparing the full feature sets we have the old FE performing better with 81,29%, then we have the new FE reporting 78,74% and finally Faria's work with 78%. When comparing Faria's feature selections both extractors fall behind. This can be attributed to the fact that Faria optimizes all the parameters in the SVM and we didn't. When comparing the precision, recall and AUROC metrics, we can conclude that the new FE performs better overall, making it a clear winner at predicting both classes against the old when using feature selection.

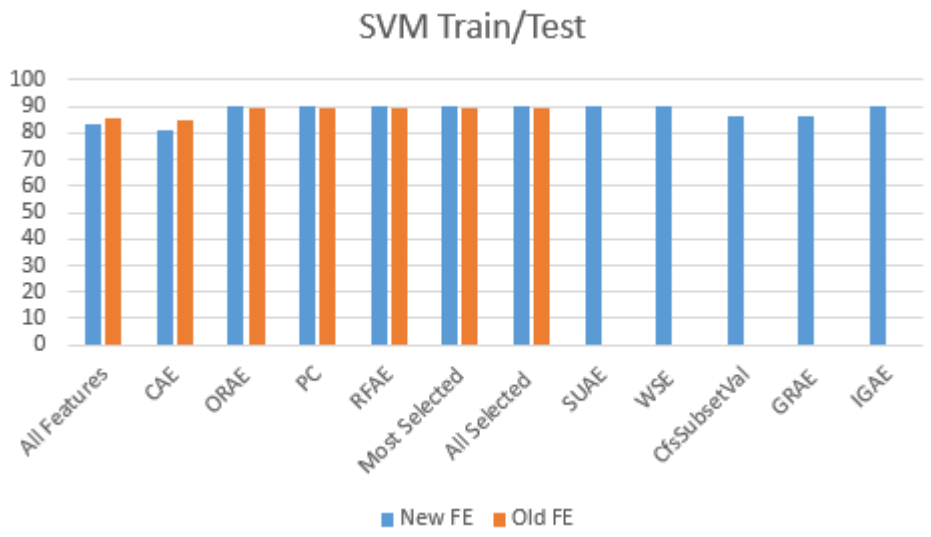


Figure 5.40: Accuracy for all datasets with the SVM classifier using Train/Test.

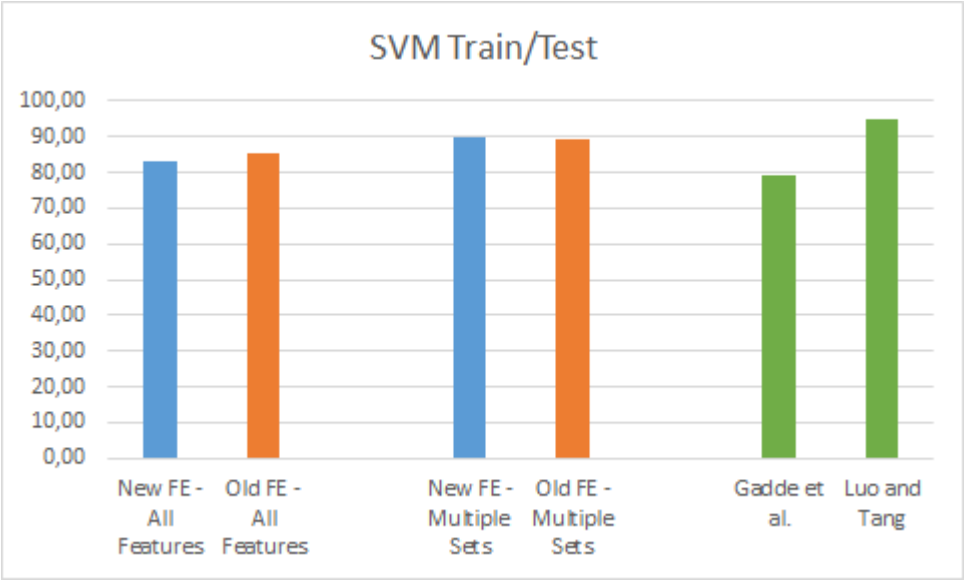


Figure 5.41: Accuracy comparison including all the features and the best feature selection method for SVM using Train/Test.

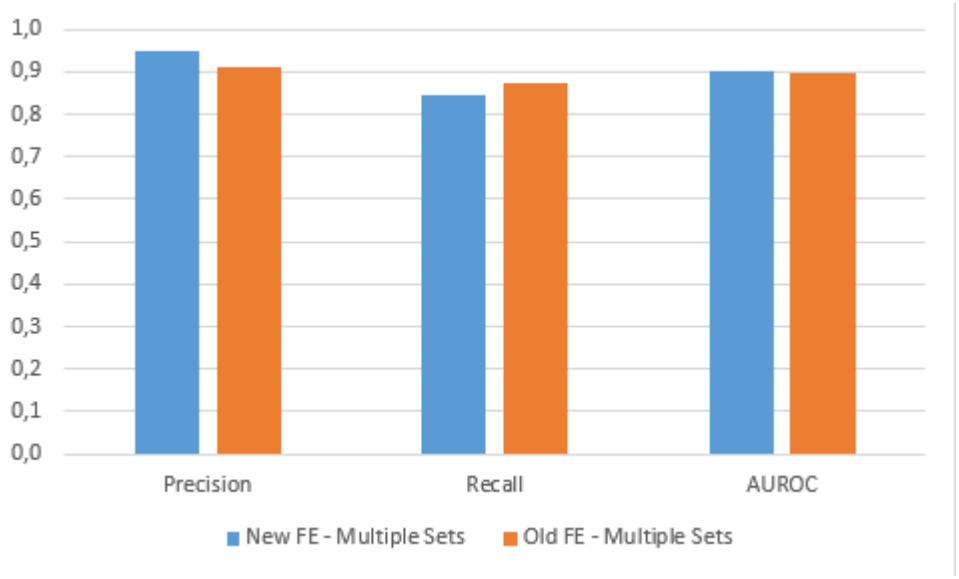


Figure 5.42: Precision, recall and AUROC statistics for the SVM classifier using Train/Test between the best feature sets found by the feature selection process.

By looking at the values for all the feature sets we can immediately realize that these feature sets: IGAE, ORAE, PC, RFAE, SUAE, WSE, Most Selected and All Selected for the new FE have the same accuracy value, which is 89,98%. And ORAE, PC, RFAE, Most Selected and All Selected share the accuracy value of 89,5% using the old FE (Fig. 5.40). The reason for this might be that with default values the classifier is overfitting and therefore achieving a common accuracy result. Even so we can see that the new FE has a better accuracy value overall with the only exceptions being the All Features and CAE data sets. When comparing the best accuracy values with other studies, we can observe that both extractors outperform Gadde and Karlapalem (2011) which reports an accuracy value of 79%, but are unable to match Luo and Tang (2008) study which reports an accuracy value of 95% (Fig. 5.41). The reason for such a high value is linked to the fact that in that study, the focus of the study is to analyze the subject of the image and ignore the background. This indicates that the features in the study are very specific and also require specific images, unlike our features which are very generic in order to analyze a broad variety of images. Finally when comparing the precision, recall and AUROC for the best feature selection set between both extractors, we can observe that the new FE is better at predicting positive instances but the old FE predicts more overall.

5.5 Results Discussions

In this section we will discuss more in-depth the results observed above, which are comprised of the new and old FE comparisons as well as other works.

5.5.1 Full Feature Set

We first start by comparing the full feature set. By looking at the data (Fig. 5.43), as well as the results reported earlier we can easily see that the best performing classifier is SVM and the worst performing classifier is Naive Bayes. The values reported are 78,74% accuracy for the new FE and 81,29% for the old FE. This tells us that the SVM is a strong classifier in helping us separating both classes in an environment where the full data set is used to both train and test.

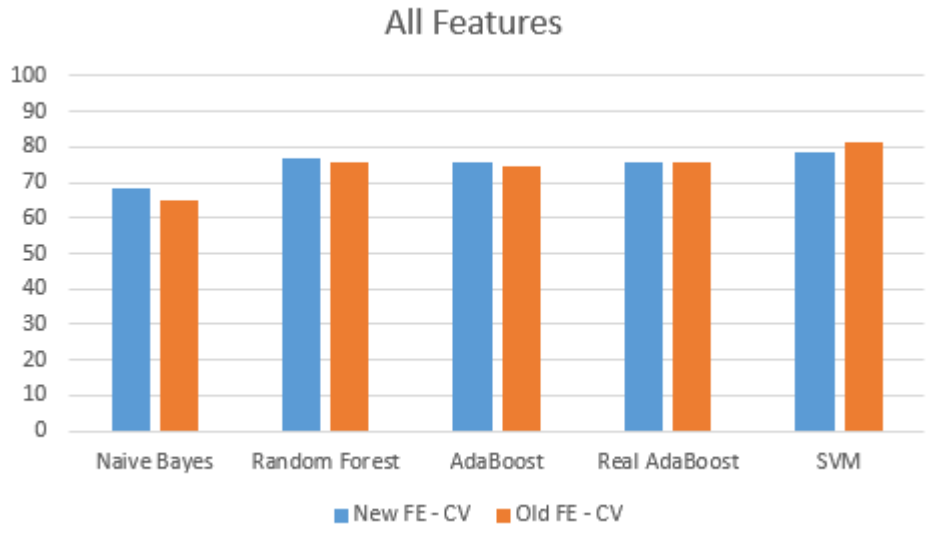


Figure 5.43: Accuracy between new and old FE for all classifiers using the full feature set with Cross Validation, scale was changed to provide more visibility.

Looking at the results for the Train/Test data validation (Fig. 5.43), we see the opposite for the new FE. Here, the Naive Bayes classifier outperforms all the others, while the Random Forest classifier has the lowest accuracy. The SVM classifier also performs well, positioning itself right after Naive Bayes. This indicates that depending on the training set, the Naive Bayes classifier might be the best at learning from a predefined training set, while the SVM classifier is better at predicting classes from alternating training and testing sets. Nonetheless, the new FE has the highest accuracy value, and outperforms the old in accuracy for all examples except for SVM.

Figure 5.44: Accuracy between new and old FE for all classifiers using the full feature set with Train/Test, scale was changed to provide more visibility.

However, when comparing the rest of the metrics we can observe that for recall values, the best classifier is SVM with the highest value of 82%. This indicates that the SVM is a very solid classifier at accurately identifying reported professional photos (positive instances) among all the others photos belonging to this class. For precision, Random Forest performs better again with a value of 78%, followed closely by SVM with 77%. Therefore, the Random Forest classifier could be a valuable tool if our objective is to have the best possible ratio of correctly identified professional photos. The AUROC value is also bigger for the Random Forest classifier, with a value of 84%, very close to AdaBoost and Real AdaBoost, each reporting a value of 83%. This validates the usefulness of these classifiers and reports them as being the best at identifying professional photographs.

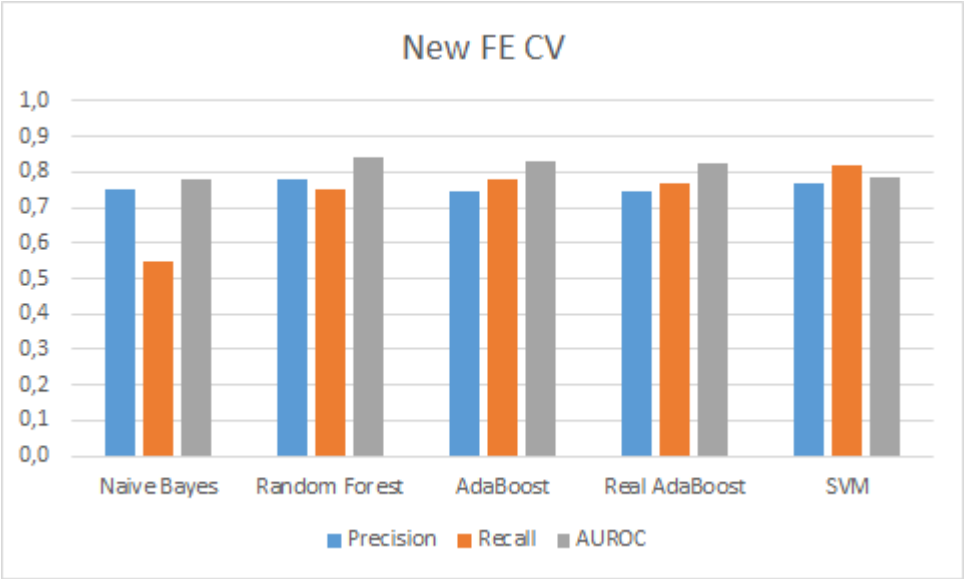


Figure 5.45: Precision, recall and AUROC values obtained for the new FE with Cross Validation.

When looking at Train/Test we can see that the Naive Bayes outperforms all the other classifiers across the board, only tying for precision with Real AdaBoost at 86% and tying for recall with SVM at 84%(Fig. 5.46). Making Naive Bayes one of the best classifier at identifying any of the classes, in any situation.

Figure 5.46: Precision, recall and AUROC values obtained for the new FE with Train/Test.

5.5.2 Feature Selection

By comparing all the feature sets obtained by the feature selection processess, we can see that the feature set obtained by gathering all the selected features for all the methods is the one that has the most accuracy, with the exceptions being for Naive Bayes, where CfsSubsetVal performs better and for SVM where WSE performs better. When comparing with the full feature set, for the Naive Bayes and Random Forest the best accuracy results are 73,67% from CfsSubsetVal and 77,97% from all selected features respectively. While for AdaBoost and Real AdaBoost the full feature set performs better at 75,69% and 75,55% respectively. Finally for SVM the best results are 79,18% for the new FE using WSE and 81,29% for the old FE using the full feature set. Making the feature selection process a worthwhile endeavor, by surpassing the full feature set in multiple occasions while only consisting in 15 features maximum, contrasting with the 92 features included in the full feature set for the new FE and 804 features for the old.

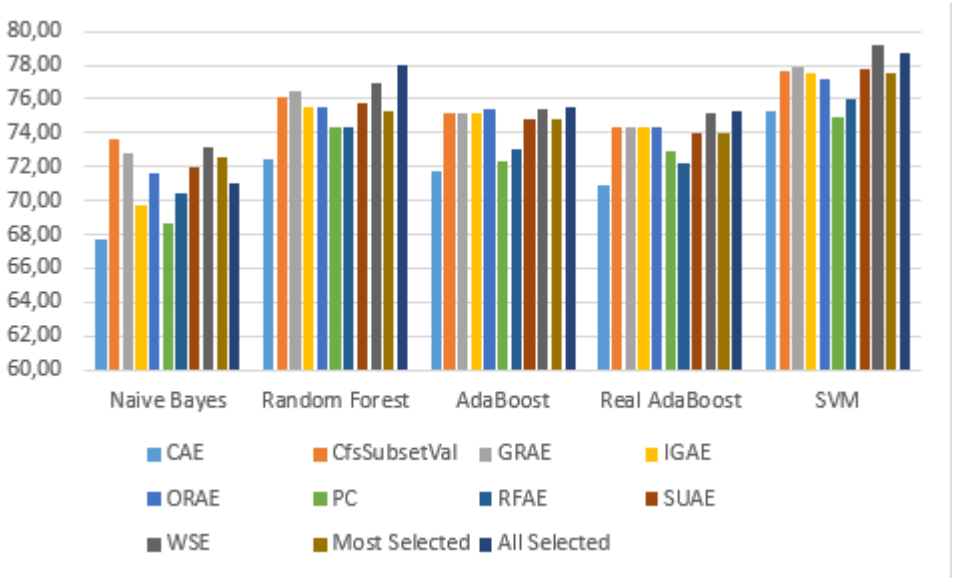


Figure 5.47: Accuracy results for all the feature selection data sets using Cross Validation.

If we analyze the features that comprise the totality of features selected in the feature selection process (Appendix F.11). We can see a solid distribution of all the categories where the features are included, with overwhelming representation from the color and compression categories (which have more features overall), both features from the texture category and five out of 9 features from the composition category. This leads us to believe that spreading our focus for all categories instead of one is a very viable solution if our goal is to classify a wide array of image types. For the CfsSubsetVal we also see some distribution between feature categories, but this time it heavily favors color with 11 out of 15 features being color, one texture, one composition and one compression. With three features being enhanced with saliency methods.

The comparison between classifiers shows us that once again the Random Forest outperforms all the other classifiers, both for the new and the old feature extractors using Cross Validation (Fig. 5.48). Which makes the Random Forest classifier, the best at accurately separating both

classes, contrasting with the Naive Bayes performing the worst again when using Cross Validation. Also, the new FE outperforms the old in every classifier, only showing close results with the Real AdaBoost classifier, reporting 75,55% and 75,54% for the new and old FE respectively.

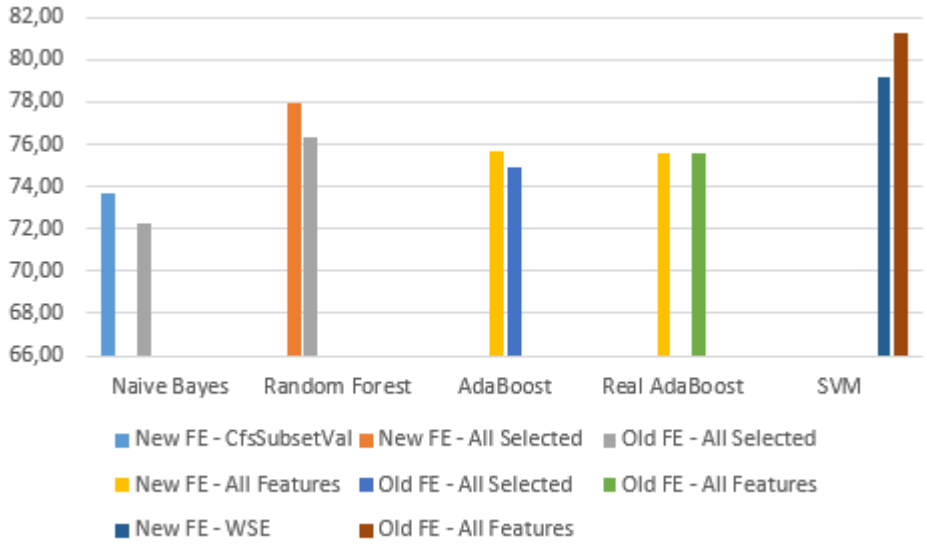


Figure 5.48: Accuracy between new and old FE, showing the best performing feature sets for each FE using Cross Validation.

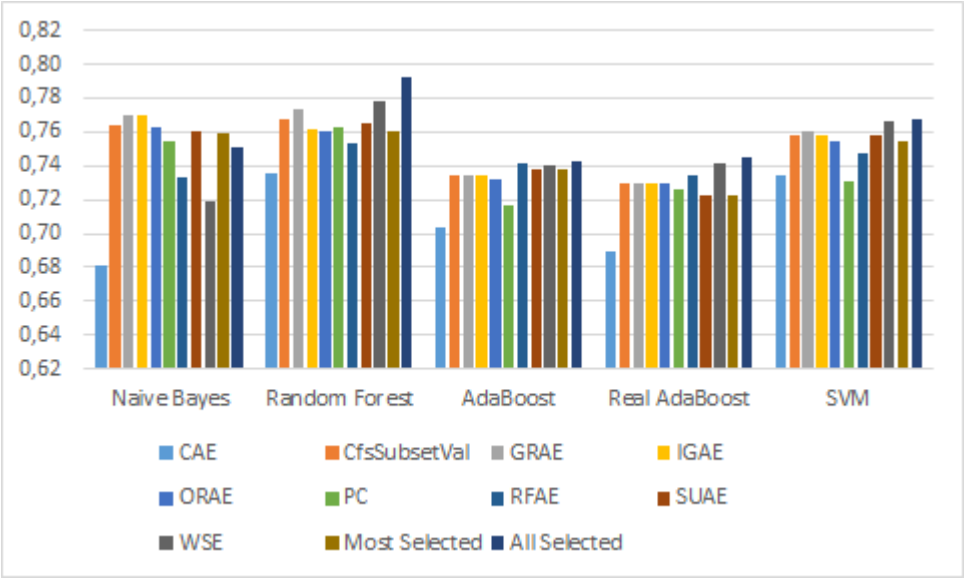


Figure 5.49: Precision values for all selection methods for the new FE using Cross Validation.

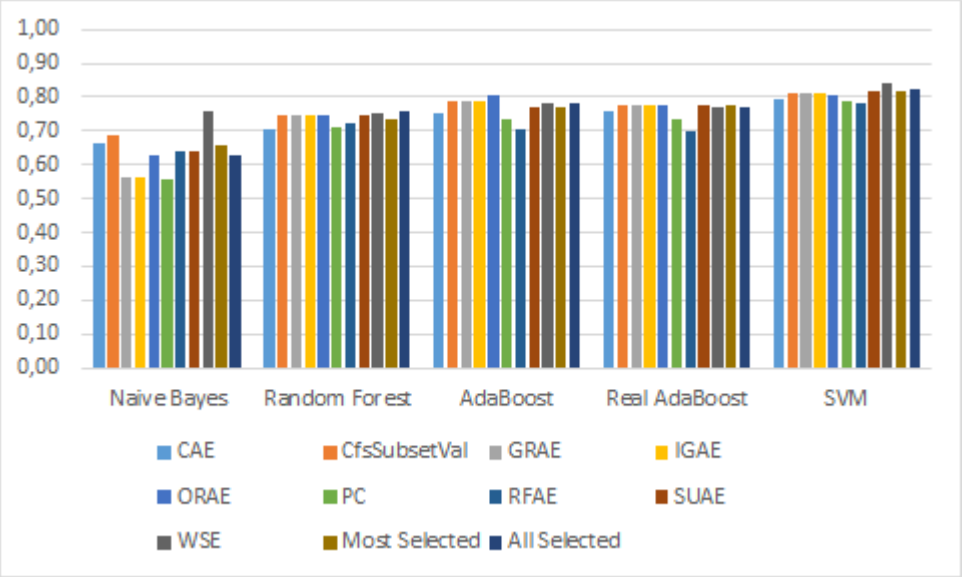


Figure 5.50: Recall values for all selection methods for the new FE using Cross Validation.

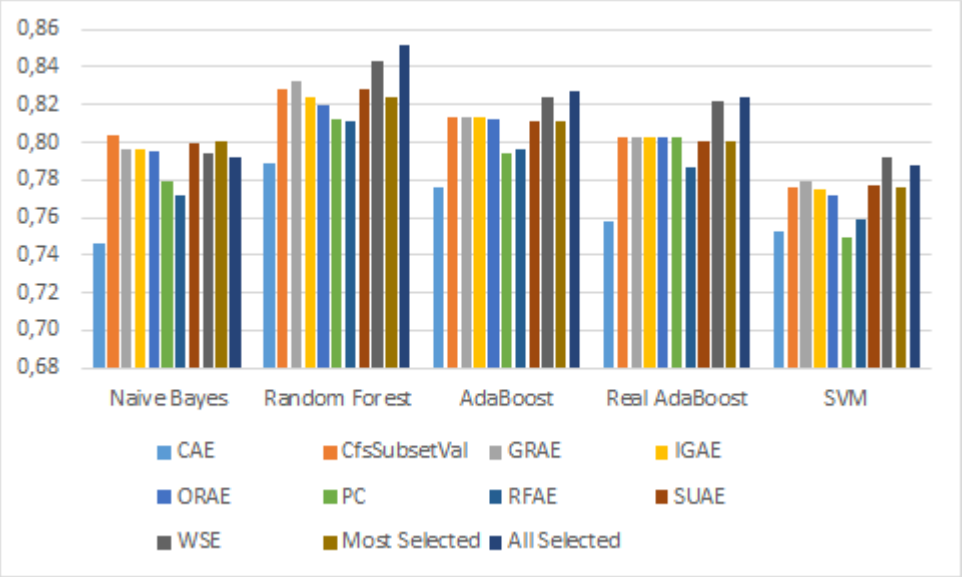


Figure 5.51: AUROC values for all selection methods for the new FE using Cross Validation.

Looking at the global values between precision, recall and AUROC metrics, we can observe that the Random Forest has the overall higher precision values, making it the best classifier at efficiently identifying professional photographs (Fig. 5.49). The SVM has the overall higher Recall values, making it the better classifier at correctly identifying the majority of the images that were considered high quality (Fig. 5.50). And lastly looking at the AUROC values we can see that the Random Forest outperforms all the other classifiers, which indicates that it is the best classifier at correctly predicting the correct class (Fig. 5.51).

For the Train/Test validation method, by looking at all the feature selection sets for every classifier, we observe that the best performing selection method was IGAE which had the best accuracy values except for Random Forest, where WSE had the highest (Fig. 5.52). When comparing with the full feature set we can observe that all the feature selection methods outperform it, which indicates that feature sets with less features are the best course of action using this validation method.

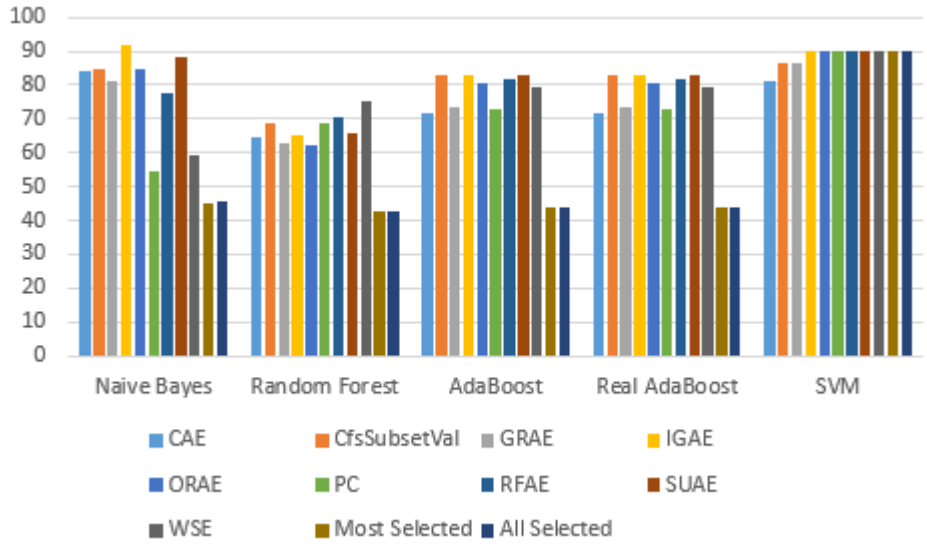


Figure 5.52: Accuracy results for all the feature selection data sets using Train/Test.

Interesting enough, when taking a closer look at the feature set composition, we can observe that 13 out of the 15 total features which compose the IGAE feature set belong to the color category, and the last 2 belong to the composition category. The WSE feature set however, has 4 color features, 3 composition features and 6 compression features. With all of the 4 color features being common in both feature sets and one of the composition features also being common. Unlike the Cross Validation method which had features from all categories in the best selection feature sets, the Train/Test method favors color features heavily, showing us that color is very important for image classification when using this data validation method. The comparison between classifiers, reveal the Naive Bayes classifier having the highest accuracy with 92,1%, and SVM ranking second with 89,98%, while Random Forest performs the worst with 75,12% (Fig. 5.53). These results fall in line with what we observed for the full feature set, once again proving that for this data validation method the Naive Bayes is the best classifier at separating classes and SVM is a very solid classifier in both data validation methods. Also when comparing both new and old feature extractors, we can see that the new FE performs better for most classifiers, with the only exception being AdaBoost.

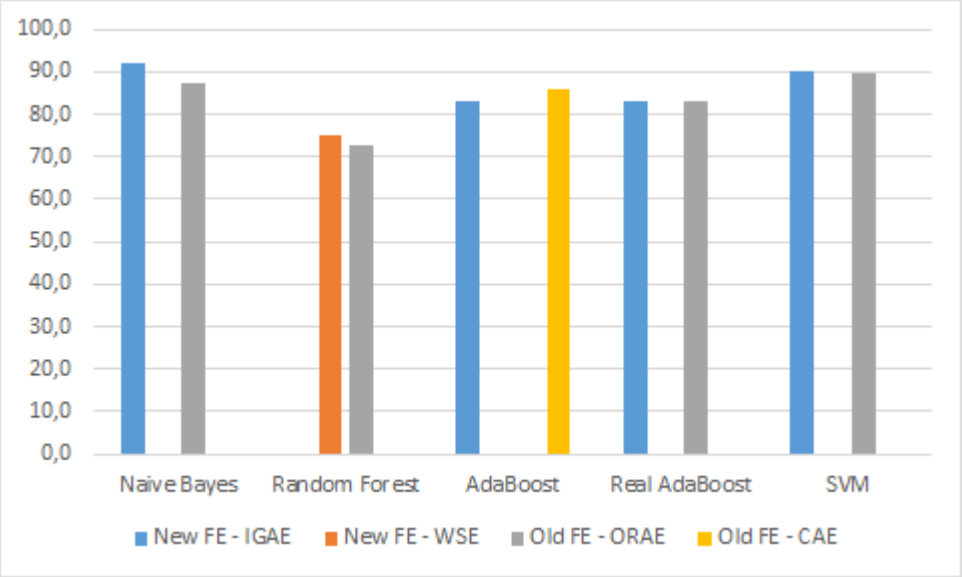


Figure 5.53: Accuracy between new and old FE, showing the best performing feature sets for each FE using Train/Test.

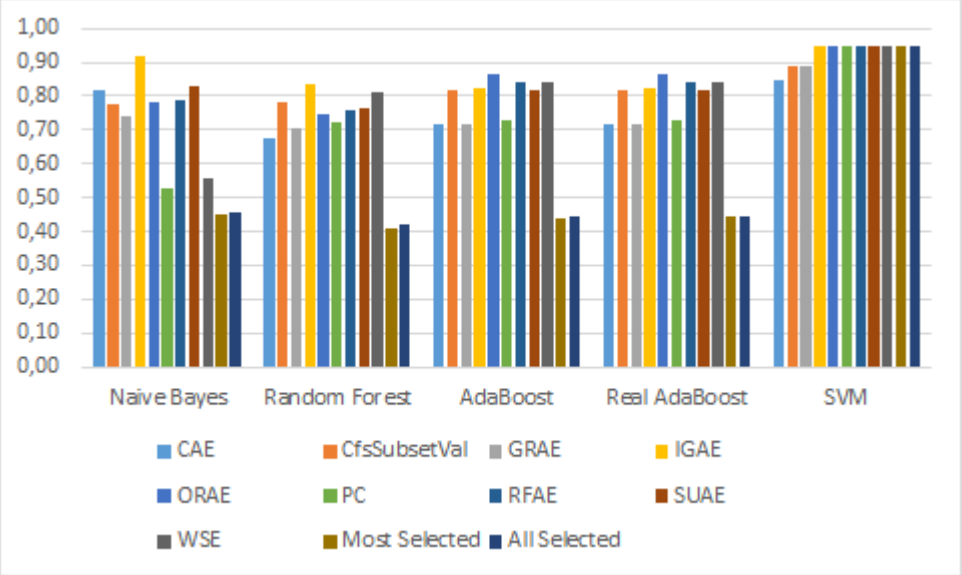


Figure 5.54: Precision values for all selection methods for the new FE using Train/Test.

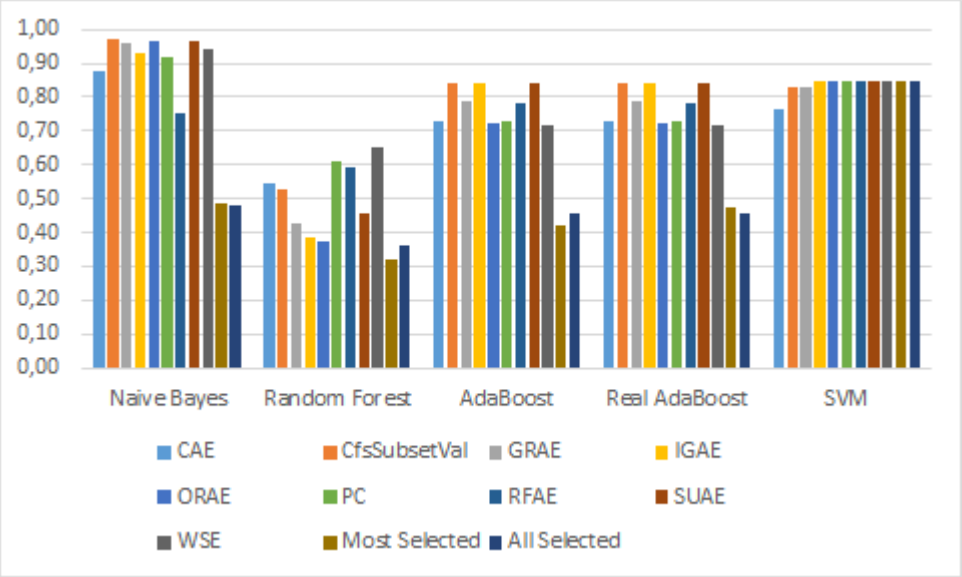


Figure 5.55: Recall values for all selection methods for the new FE using Train/Test.

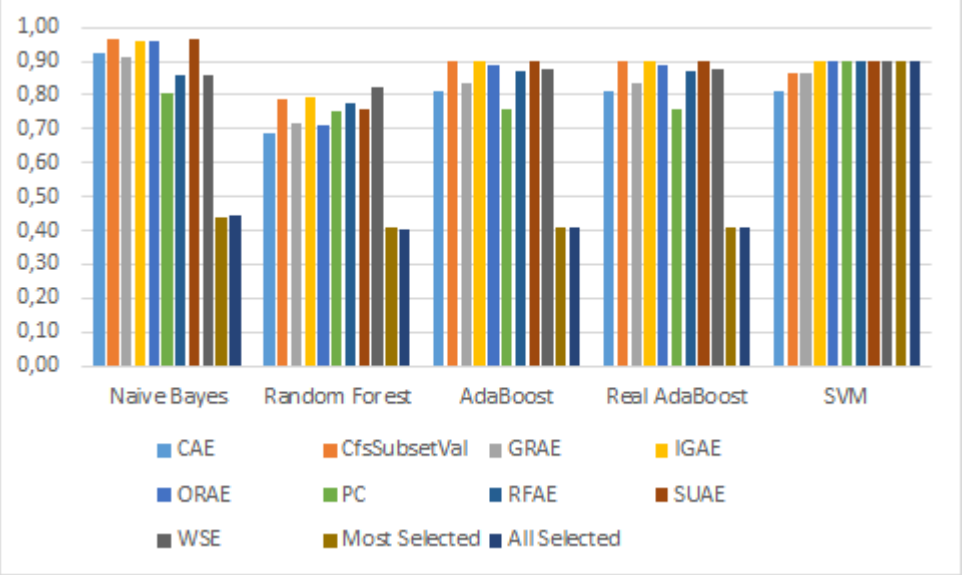


Figure 5.56: AUROC values for all selection methods for the new FE using Train/Test.

Similar to the analysis we did earlier for the Cross Validation method, we will now look at the precision, recall and AUROC metrics for Train/Test to try and see beyond the accuracy values reported above. Starting with the precision values, we can observe that the SVM classifier reports the highest values, which indicates it is the best classifier at predicting the highest quality photographs efficiently (Fig. 5.54). For the recall values, we see that Naive Bayes has the highest values, but it is not as consistent as SVM. Meaning that depending on the feature set, either Naive Bayes or SVM are the best at identifying the majority of the images that were labeled as high quality. And finally

for the AUROC metric we observe the same pattern as the recall metric. With Naive Bayes having the highest values, but it is not as consistent as the SVM. Thus depending on the feature set, either Naive Bayes or SVM can be the best at correctly separating the corresponding class.

Chapter 6

Conclusions

This document reflects the work done during the year concerning the study of image analysis based on aesthetic principles. We started by doing a review for the current state-of-the-art features and methods on image analysis. Our goal is to succeed as much as possible in creating a set of competitive features capable of efficiently analyzing the quality of an image. After deciding on which features could prove to be the most useful, we implemented a feature extraction program inspired by the work done by Correia (2011). Our finished FE contains 92 features, unlike the old version who has a total of 804. From the features implemented, only the rule-of-thirds was developed by ourselves, although it is based upon a well known theory used by professional photographers. To test our final set of features we performed classification in a data set which is very popular among works similar to ours. We utilized five different classifiers: Naive Bayes, Random Forest, AdaBoost, Real AdaBoost and SVM as well as two different data validation methods: cross validation and using a predefined training and testing set. We expanded our tests, not only to the new FE but also to the old, in order to be able to learn if our new set of features is more efficient at separating images from high and low quality compared to the previous FE. Whenever possible we tried to vary the parameters of each classifier, in order to obtain the best results possible. We tested not only the full feature set, but also ran feature selection tests in order to obtain multiple small feature sets to help us understand which feature categories are the best and if we can obtain better results with less features. We learned that by using the full feature set, the new FE outperforms the old in accuracy for all the classifiers except for SVM, using both data validation methods. The old FE has the highest accuracy value when using cross validation, while the new FE has the highest accuracy value when using train/test. When performing the same tests for the feature selection methods we can see that the new FE outperforms the old in every classifier when using cross validation. And for train/test the new FE outperforms the old FE in almost every classifier, except for AdaBoost, but it still maintains the highest accuracy value. We also observed that when using cross validation, feature selection has highest accuracy than the full feature set for 3 out of 5 classifiers, and when using train/test, feature selection always presents higher accuracy then the full feature set. This success can be attributed to a well rounded selection of feature categories for cross validation, and color features for train/test. Finally when compared with other studies, we present higher accuracy results in 5 out of 15 examples, which makes us competitive but leaves room for improvement.

I believe this work to have accomplished its initial goal. By reducing the feature set we still

managed to obtain overall better results in image classification. Unfortunately we did not manage to propose any new feature, since we spent so much time understanding and implementing state-of-the-art features, we ended up not having time to add anything else. We believe our accuracy results could have been better if we could experiment with an even bigger range of classifier parameters. However, this was difficult due to the old FE full feature size which increased the computing time tremendously. Another possibility of improvement, is of course, adding different features.

Bibliography

Achanta, R., Hemami, S., Estrada, F., and Süsstrunk, S. (2009). Frequency-tuned Salient Region Detection. In *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR 2009)*, pages 1597 – 1604. For code and supplementary material, click on the url below.

Altman, N. S. (1992). An Introduction to Kernel and Nearest-Neighbor Nonparametric Regression. *The American Statistician*, 46(3):175–185.

Aslam, M. M. (2006). Are you selling the right colour? a cross-cultural review of colour as a marketing cue. *Journal of Marketing Communications*, 12(1):15–30.

Benbouzid, D., Busa-Fekete, R., Casagrande, N., Collin, F.-D., and Kégl, B. (2012). Multiboost: A multi-purpose boosting package. *Journal of Machine Learning Research*, 13:549–553.

Bense, M. (1969). *Einführung in die informationstheoretische Asthetik. Grundlegung und Anwendung in der Texttheorie*. Rowohlt Taschenbuch Verlag.

Berys Gout, D. M. L. (2005). *The Routledge Companion to Aesthetics*. Routledge.

Bezdek, J. C. (1981). *Pattern Recognition with Fuzzy Objective Function Algorithms*. Kluwer Academic Publishers, Norwell, MA, USA.

Birkhoff, G. (1933). *Aesthetic Measure*. Harvard University.

Canny, J. (1986). A computational approach to edge detection. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, PAMI-8(6):679–698.

Celia, B. and Felci Rajam, I. (2012). An efficient content based image retrieval framework using machine learning techniques. In Kannan, R. and Andres, F., editors, *Data Engineering and Management*, volume 6411 of *Lecture Notes in Computer Science*, pages 162–169. Springer Berlin Heidelberg.

Cheng, M.-M., Zhang, G.-X., Mitra, N. J., Huang, X., and Hu, S.-M. (2011). Global contrast based salient region detection. *2013 IEEE Conference on Computer Vision and Pattern Recognition*, 0:409–416.

Ciesielski, V., Barile, P., and Trist, K. (2013). Finding image features associated with high aesthetic value by machine learning. In *Evolutionary and Biologically Inspired Music, Sound, Art and Design - Second International Conference, EvoMUSART*, pages 47–58, Vienna, Austria.

Coe, K. (1992). Art: The replicable unit - An inquiry into the possible origin of art as a social behavior. *Journal of Social and Evolutionary Systems*, pages 217–234.

Correia, J. a. (2011). Evolutionary Computation for Assessing and Improving Classifier Performance. Technical report, Faculdade de Ciências e Tecnologia da Universidade de Coimbra.

Cortes, C. and Vapnik, V. (1995). Support-vector networks. *Machine Learning*, 20(3):273–297.

Datta, R., Joshi, D., Li, J., and Wang, J. Z. (2006). Studying aesthetics in photographic images using a computational approach. In *Proceedings of the 9th European Conference on Computer Vision - Volume Part III*, ECCV'06, pages 288–301, Berlin, Heidelberg. Springer-Verlag.

de Faria, J. a. M. P. (2012). *What Makes a Good Picture?* PhD thesis, Cranfield University.

den Heijer, E. (2012). Evolving art using measures for symmetry, compositional balance and liveliness. In *IJCCI'12*, pages 52–61.

Desnoyer, M. and Wettergreen, D. (2010). Aesthetic Image Classification for Autonomous Agents. *2010 20th International Conference on Pattern Recognition*, pages 3452–3455.

Dhar, S., Ordonez, V., and Berg, T. (2011). High level describable attributes for predicting aesthetics and interestingness. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 1657–1664.

Dutton, D. (2009). *The Art Instinct: Beauty, Pleasure, & Human Evolution*. Oxford University Press.

Fawcett, T. (2006). An introduction to roc analysis. *Pattern Recogn. Lett.*, 27(8):861–874.

Felci Rajam, I. and Valli, S. (2011). Content-based image retrieval using a quick svm-binary decision tree - qsvmbdt. In Nagamalai, D., Renault, E., and Dhanuskodi, M., editors, *Advances in Digital Image Processing and Information Technology*, volume 205 of *Communications in Computer and Information Science*, pages 11–22. Springer Berlin Heidelberg.

Fukunaga, K. and Hostetler, L. (1975). The estimation of the gradient of a density function, with applications in pattern recognition. *Information Theory, IEEE Transactions on*, 21(1):32–40.

Gadde, R. and Karlapalem, K. (2011). Aesthetic guideline driven photography by robots.

Goferman, S., Zelnik-Manor, L., and Tal, A. (2012). Context-aware saliency detection. *IEEE transactions on pattern analysis and machine intelligence*, 34(10):1915–26.

Graf, A. B., Smola, A. J., and Borer, S. (2003). Classification in a normalized feature space using support vector machines. *Neural Networks, IEEE Transactions on*, 14(3):597–605.

Graves, M. (1946). *Design Judgement Test*. Psychological Corporation.

Hall, M. A. (1998). *Correlation-based Feature Subset Selection for Machine Learning*. PhD thesis, University of Waikato, Hamilton, New Zealand.

Harel, J., Koch, C., and Perona, P. (2006). Graph-Based Visual Saliency. *Proceedings of the 20th Annual Conference on Neural Information Processing Systems*.

Holte, R. (1993). Very simple classification rules perform well on most commonly used datasets. *Machine Learning*, 11:63–91.

Hou, X. and Zhang, L. (2007). Saliency detection: A spectral residual approach. In *Computer Vision and Pattern Recognition, 2007. CVPR '07. IEEE Conference on*, pages 1–8.

Jacobs, R. (2011). *Aesthetics by numbers.* PhD thesis. Relation: http://www.rug.nl/ Rights: University of Groningen.

João Faria, Stanislav Bagley, Stefan Rüger, T. B. (2013). Challenges of Finding Aesthetically Pleasing Images. *Image Analysis for Multimedia Interactive Services (WIAMIS), 2013 14th International Workshop on*, 2:1–4.

Karimi, K. and Hamilton, H. (2002). Timesleuth: a tool for discovering causal and temporal rules. In *Tools with Artificial Intelligence, 2002. (ICTAI 2002). Proceedings. 14th IEEE International Conference on*, pages 375–380.

Ke, Y., Tang, X., and Jing, F. (2006). The design of high-level features for photo quality assessment. In *in IEEE Computer Society Conference on Computer Vision and Pattern Recognition,2006,vol.1,pp.419-426*, pages 419–426.

Khan, S. S. and Vogel, D. (2012). Evaluating visual aesthetics in photographic portraiture. In *Proceedings of the Eighth Annual Symposium on Computational Aesthetics in Graphics, Visualization, and Imaging*, CAe '12, pages 55–62, Aire-la-Ville, Switzerland, Switzerland. Eurographics Association.

Kira, K. and Rendell, L. A. (1992). A practical approach to feature selection. In Sleeman, D. H. and Edwards, P., editors, *Ninth International Workshop on Machine Learning*, pages 249–256. Morgan Kaufmann.

Koffka, K. (1955). *Principles of Gestalt Psychology.* International library of psychology, philosophy, and scientific method. Routledge & K. Paul.

Kohavi, R. and John, G. H. (1997). Wrappers for feature subset selection. *Artificial Intelligence*, 97(1-2):273–324. Special issue on relevance.

Kolmogorov, A. N. (1963). On tables of random numbers. *Sankhyā: The Indian Journal of Statistics, Series A*, 25(4):369–376.

Kononenko, I. (1994). Estimating attributes: Analysis and extensions of relief. In Bergadano, F. and Raedt, L. D., editors, *European Conference on Machine Learning*, pages 171–182. Springer.

Koshelev, M., Kreinovich, V., and Yam, Y. (1998). Towards the use of aesthetics in decision making: Kolmogorov complexity formalizes birkhoff's idea. In *Bulletin of the European Association for Theoretical Computer Science (EATCS*, pages 166–170.

Li, C. and Chen, T. (2009). Aesthetic visual quality assessment of paintings. *Selected Topics in Signal Processing, IEEE Journal of*, 3(2):236–252.

Lind, R. W. (1980). Attention and the Aesthetic Object. *The Journal of Aesthetics and Art Criticism*, Vol. 39(No. 2):131–142.

Lo, K.-Y., Liu, K.-H., and Chen, C.-S. (2012). Assessment of photo aesthetics with efficiency. In *ICPR*, pages 2186–2189. IEEE.

Locher, P. and Nodine, C. (1989). The perceptual value of symmetry. *Comput. Math. Appl.*, 17(4-6):475–484.

Luo, Y. and Tang, X. (2008). Photo and video quality evaluation: Focusing on the subject. In *Proceedings of the 10th European Conference on Computer Vision: Part III*, ECCV '08, pages 386–399, Berlin, Heidelberg. Springer-Verlag.

Machado, P. and Cardoso, A. (1998). Computing aesthetics. In *Advances in Artificial Intelligence*, pages 219–228. Springer Berlin Heidelberg.

Machado, P., Romero, J., Santos, A., Cardoso, A., and Pazos, A. (2007). On the development of evolutionary artificial artists. *Computers & Graphics*, 31(6):818–826.

Moles, A. and Cohen, J. (1968). *Information theory and esthetic perception*. Illini Books. University of Illinois Press.

Nishiyama, M., Okabe, T., Sato, I., and Sato, Y. (2011). Aesthetic quality classification of photographs based on color harmony. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 33–40.

Rigau, J., Feixas, M., and Sbert, M. (2008). Informational dialogue with van gogh's paintings. In *Proceedings of the Fourth Eurographics Conference on Computational Aesthetics in Graphics, Visualization and Imaging*, Computational Aesthetics'08, pages 115–122, Aire-la-Ville, Switzerland, Switzerland. Eurographics Association.

Robnik-Sikonja, M. and Kononenko, I. (1997). An adaptation of relief for attribute estimation in regression. In Fisher, D. H., editor, *Fourteenth International Conference on Machine Learning*, pages 296–304. Morgan Kaufmann.

Romero, J., Machado, P., Carballal, A., and Correia, J. (2012a). Computing aesthetics with image judgement systems. In McCormack, J. and d'Inverno, M., editors, *Computers and Creativity*, pages 295–322. Springer Berlin Heidelberg.

Romero, J., Machado, P., Carballal, A., and Osorio, O. (2011). Aesthetic classification and sorting based on image compression. In Di Chio, C., Brabazon, A., Di Caro, G., Drechsler, R., Farooq, M., Grahl, J., Greenfield, G., Prins, C., Romero, J., Squillero, G., Tarantino, E., Tettamanzi, A., Urquhart, N., and Uyar, A., editors, *Applications of Evolutionary Computation*, volume 6625 of *Lecture Notes in Computer Science*, pages 394–403. Springer Berlin Heidelberg.

Romero, J., Machado, P., Carballal, A., and Santos, A. (2012b). Using complexity estimates in aesthetic image classification. *Journal of Mathematics and the Arts*, 6(2-3):125–136.

Rubner, Y., Tomasi, C., and Guibas, L. J. (2000). The Earth Mover's Distance as a Metric for Image Retrieval. *Int'l. J. Computer Vision*, pages 4(2):99–121.

Rumelhart, D., Hintont, G., and Williams, R. (1986). Learning representations by back-propagating errors. *Nature*, 323(6088):533–536.

Shannon, C. E. (1948). A mathematical theory of communication. *Bell System Technical Journal*, 3(27):379–423.

Sheppard, Oleg V.Bychkov, A. (2010). *Greek and Roman Aesthetics*. Cambridge University Press.

Shi, J. and Malik, J. (2000). Normalized cuts and image segmentation. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 22(8):888–905.

Statistics, L. B. and Breiman, L. (2001). Random forests. In *Machine Learning*, pages 5–32.

Stricker, M. and Orengo, M. (1995). Similarity of color images. pages 381–392.

Su, H.-H., Chen, T.-W., Kao, C.-C., Hsu, W. H., and Chien, S.-Y. (2011). Scenic photo quality assessment with bag of aesthetics-preserving features. In *Proceedings of the 19th ACM International Conference on Multimedia*, MM '11, pages 1213–1216, New York, NY, USA. ACM.

Tamura, H., Mori, S., and Yamawaki, T. (1978). Texture features corresponding to visual perception. *IEEE Transactions on System, Man and Cybernatic*, 6.

Torralba, A., Murphy, K., and Freeman, W. (2004). Sharing features: efficient boosting procedures for multiclass object detection. In *Computer Vision and Pattern Recognition, 2004. CVPR 2004. Proceedings of the 2004 IEEE Computer Society Conference on*, volume 2, pages II–762–II–769 Vol.2.

Wang, X., Jia, J., Yin, J., and Cai, L. (2013). Interpretable aesthetic features for affective image classification. In *IEEE International Conference on Image Processing*, pages 3230–3234, Melbourne, Australia.

Wong, L.-K. and Low, K.-L. (2009). Saliency-enhanced image aesthetics class prediction. In *Image Processing (ICIP), 2009 16th IEEE International Conference on*, pages 997–1000.

Xu, Q., D'Souza, D., and Ciesielski, V. (2007). Evolving images for entertainment. In *Proceedings of the 4th Australasian Conference on Interactive Entertainment*, IE '07, pages 26:1–26:8, Melbourne, Australia, Australia. RMIT University.

Yeh, C.-H., Ho, Y.-C., Barsky, B. a., and Ouhyoung, M. (2010). Personalized photograph ranking and selection system. *Proceedings of the international conference on Multimedia - MM '10*, page 211.

Zakariya, S., Ali, R., and Ahmad, N. (2010). Unsupervised content based image retrieval by combining visual features of an image with a threshold. *International Conference on Computer Technology*, 2.

Zhang, J. and Sclaroff, S. (2013). Saliency detection: A boolean map approach. In *Computer Vision (ICCV), 2013 IEEE International Conference on*, pages 153–160.

Appendix A

Accuracy, precision, recall and AUROC values for all feature sets with the Naive Bayes classifier, highlighting the highest accuracy data set.

Feature set	Accuracy	Precision	Recall	AUROC
All Features	68,22	0,75	0,55	0,78
CAE	67,68	0,68	0,66	0,75
CfsSubsetVal	**73,67**	**0,76**	**0,69**	**0,80**
GRAE	69,71	0,77	0,56	0,80
IGAE	69,71	0,77	0,56	0,80
ORAE	71,58	0,76	0,63	0,80
PC	68,69	0,75	0,55	0,78
RFAE	70,45	0,73	0,64	0,77
SUAE	71,95	0,76	0,64	0,80
WSE	73,13	0,72	0,76	0,79
Most Selected	72,53	0,76	0,66	0,80
All Selected	71,07	0,75	0,63	0,79

Table A.1: New FE using 10-fold CV.

Feature set	Accuracy	Precision	Recall	AUROC
All Features	64,83	0,69	0,53	0,71
CAE	69,88	0,66	0,81	0,76
ORAE	68,88	0,64	0,87	0,78
PC	70,63	0,74	0,64	0,78
RFAE	68,78	0,69	0,67	0,73
Most Selected	71,18	0,67	0,84	0,78
All Selected	**72,23**	**0,69**	**0,80**	**0,78**

Table A.2: Old FE using 10-fold CV.

Feature set	Accuracy	Precision	Recall	AUROC
All Features	84,93	0,86	0,84	
CAE	83,9	0,82	0,88	0,92
CfsSubsetVal	84,48	0,78	0,97	0,97
GRAE	81,37	0,74	0,96	0,91
IGAE	**92,1**	**0,92**	**0,93**	**0,96**
ORAE	84,78	0,78	0,96	0,96
PC	54,35	0,52	0,92	0,81
RFAE	77,32	0,79	0,75	0,86
SUAE	88,35	0,83	0,97	0,96
WSE	59,3	0,55	0,94	0,86
Most Selected	44,94	0,45	0,49	0,44
All Selected	45,66	0,46	0,48	0,45

Table A.3: New FE using Train/Test.

Feature set	Accuracy	Precision	Recall	AUROC
All Features	50,02	1	3,3E-04	0,83
CAE	83,52	0,80	0,90	0,92
ORAE	**87,35**	**0,83**	**0,93**	**0,95**
PC	62,88	0,58	0,89	0,81
RFAE	80,85	0,83	0,78	0,88
Most Selected	43,37	0,40	0,28	0,42
All Selected	13,73	0,18	0,20	0,06

Table A.4: Old FE using Train/Test.

Appendix B

Accuracy, precision, recall and AUROC values for all feature sets with the Random Forest classifier, highlighting the highest accuracy data set.

Feature set	Accuracy	Precision	Recall	AUROC
All Features	76,97	0,78	0,75	0,84
CAE	72,5	0,74	0,70	0,79
CfsSubsetVal	76,06	0,77	0,75	0,83
GRAE	76,48	0,77	0,75	0,83
IGAE	75,51	0,76	0,74	0,82
ORAE	75,48	0,76	0,74	0,82
PC	74,38	0,76	0,71	0,81
RFAE	74,33	0,75	0,72	0,81
SUAE	75,81	0,76	0,75	0,83
WSE	76,93	0,78	0,75	0,84
Most Selected	75,27	0,76	0,74	0,82
All Selected	**77,97**	**0,79**	**0,76**	**0,85**

Table B.1: New FE using 10-fold CV.

Feature set	Accuracy	Precision	Recall	AUROC
All Features	75,73	0,77	0,73	0,83
CAE	70,43	0,72	0,68	0,78
ORAE	75,12	0,76	0,73	0,82
PC	75,1	0,76	0,73	0,82
RFAE	71,3	0,73	0,68	0,77
Most Selected	71,39	0,72	0,69	0,78
All Selected	**76,3**	**0,77**	**0,75**	**0,83**

Table B.2: Old FE using 10-fold CV.

Feature set	Accuracy	Precision	Recall	AUROC
All Features	72,9	0,82	0,59	0,83
CAE	64,33	0,68	0,55	0,69
CfsSubsetVal	68,92	0,78	0,53	0,79
GRAE	62,52	0,71	0,43	0,72
IGAE	65,3	0,83	0,38	0,79
ORAE	62,18	0,74	0,37	0,71
PC	68,92	0,72	0,61	0,76
RFAE	70,2	0,76	0,59	0,78
SUAE	65,87	0,77	0,46	0,76
WSE	**75,12**	**0,81**	**0,65**	**0,82**
Most Selected	42,86	0,41	0,32	0,41
All Selected	42,96	0,42	0,36	0,40

Table B.3: New FE using Train/Test.

Feature set	Accuracy	Precision	Recall	AUROC
All Features	68,4	0,76	0,54	0,76
CAE	64,6	0,71	0,49	0,71
ORAE	**72,85**	**0,87**	**0,54**	**0,85**
PC	69,15	0,69	0,69	0,76
RFAE	66,87	0,73	0,54	0,73
Most Selected	42,32	0,41	0,33	0,41
All Selected	29,41	0,20	0,14	0,18

Table B.4: Old FE using Train/Test.

Appendix C

Accuracy, precision, recall and AUROC values for all feature sets with the AdaBoost classifier, highlighting the highest accuracy data set.

Feature set	Accuracy	Precision	Recall	AUROC
All Features	**75,69**	**0,74**	**0,78**	**0,83**
CAE	71,73	0,70	0,75	0,78
CfsSubsetVal	75,14	0,73	0,79	0,81
GRAE	75,14	0,73	0,79	0,81
IGAE	75,14	0,73	0,79	0,81
ORAE	75,42	0,73	0,80	0,81
PC	72,28	0,72	0,74	0,79
RFAE	73,02	0,74	0,71	0,80
SUAE	74,82	0,74	0,77	0,81
WSE	75,4	0,74	0,78	0,82
Most Selected	74,82	0,74	0,77	0,81
All Selected	75,56	0,74	0,78	0,83

Table C.1: New FE using 10-fold CV.

Feature set	Accuracy	Precision	Recall	AUROC
All Features	74,54	0,74	0,76	0,81
CAE	71,59	0,73	0,69	0,78
ORAE	74,27	0,75	0,73	0,81
PC	72,96	0,73	0,73	0,80
RFAE	69,64	0,73	0,62	0,77
Most Selected	71,41	0,72	0,70	0,79
All Selected	**74,92**	**0,74**	**0,76**	**0,82**

Table C.2: Old FE using 10-fold CV.

Feature set	Accuracy	Precision	Recall	AUROC
All Features	78,22	0,80	0,75	0,87
CAE	71,88	0,71	0,73	0,81
CfsSubsetVal	82,73	0,82	0,84	0,90
GRAE	73,67	0,71	0,79	0,83
IGAE	**83,05**	**0,82**	**0,84**	**0,90**
ORAE	80,55	0,86	0,72	0,89
PC	72,77	0,73	0,73	0,76
RFAE	81,5	0,84	0,78	0,87
SUAE	82,73	0,82	0,84	0,90
WSE	79,13	0,84	0,72	0,88
Most Selected	43,84	0,44	0,42	0,41
All Selected	44,07	0,44	0,45	0,41

Table C.3: New FE using Train/Test.

Feature set	Accuracy	Precision	Recall	AUROC
All Features	74,02	0,72	0,94	0,91
CAE	**85,73**	**0,90**	**0,80**	**0,92**
ORAE	83,87	0,89	0,78	0,92
PC	78,68	0,72	0,94	0,91
RFAE	82,45	0,89	0,74	0,90
Most Selected	44,32	0,44	0,40	0,42
All Selected	12,31	0,13	0,13	0,06

Table C.4: Old FE using Train/Test.

Appendix D

Accuracy, precision, recall and AUROC values for all feature sets with the Real AdaBoost classifier, highlighting the highest accuracy data set.

Feature set	Accuracy	Precision	Recall	AUROC
All Features	**75,55**	**0,75**	**0,77**	**0,83**
CAE	70,88	0,69	0,76	0,76
CfsSubsetVal	74,34	0,73	0,77	0,80
GRAE	74,34	0,73	0,77	0,80
IGAE	74,34	0,73	0,77	0,80
ORAE	74,34	0,73	0,77	0,80
PC	72,91	0,73	0,74	0,80
RFAE	72,26	0,73	0,70	0,79
SUAE	73,93	0,72	0,78	0,80
WSE	75,18	0,74	0,77	0,82
Most Selected	73,93	0,72	0,78	0,80
All Selected	75,33	0,75	0,77	0,82

Table D.1: New FE using 10-fold CV.

Feature set	Accuracy	Precision	Recall	AUROC
All Features	**75,54**	**0,75**	**0,76**	**0,83**
CAE	71,53	0,73	0,69	0,78
ORAE	74,41	0,75	0,72	0,81
PC	73,42	0,74	0,73	0,81
RFAE	70,75	0,71	0,71	0,77
Most Selected	71,31	0,72	0,69	0,78
All Selected	74,42	0,76	0,72	0,81

Table D.2: Old FE using 10-fold CV.

Feature set	Accuracy	Precision	Recall	AUROC
All Features	81,05	0,86	0,75	0,90
CAE	71,88	0,71	0,73	0,81
CfsSubsetVal	82,73	0,82	0,84	0,90
GRAE	73,67	0,71	0,79	0,83
IGAE	**83,05**	**0,82**	**0,84**	**0,90**
ORAE	80,55	0,86	0,72	0,89
PC	72,77	0,73	0,73	0,76
RFAE	81,5	0,84	0,78	0,87
SUAE	82,73	0,82	0,84	0,90
WSE	79,13	0,84	0,72	0,88
Most Selected	44,04	0,44	0,47	0,41
All Selected	44,07	0,44	0,45	0,41

Table D.3: New FE using Train/Test.

Feature set	Accuracy	Precision	Recall	AUROC
All Features	66,7	0,63	0,81	0,72
CAE	81	0,83	0,79	0,89
ORAE	**83,28**	**0,94**	**0,71**	**0,94**
PC	67,17	0,61	0,95	0,87
RFAE	79,45	0,86	0,7	0,88
Most Selected	44,06	0,43	0,34	0,42
All Selected	16,95	0,13	0,11	0,08

Table D.4: Old FE using Train/Test.

Appendix E

Accuracy, precision, recall and AUROC values for all feature sets with the SVM classifier, highlighting the highest accuracy data set.

Feature set	Accuracy	Precision	Recall	AUROC
All Features	78,74	0,77	0,82	0,79
CAE	75,28	0,73	0,79	0,75
CfsSubsetVal	77,63	0,76	0,81	0,78
GRAE	77,89	0,76	0,81	0,78
IGAE	77,53	0,76	0,81	0,78
ORAE	77,18	0,75	0,81	0,77
PC	74,89	0,73	0,79	0,75
RFAE	76	0,75	0,78	0,76
SUAE	77,74	0,76	0,82	0,78
WSE	**79,18**	**0,77**	**0,84**	**0,79**
Most Selected	77,58	0,76	0,82	0,78
All Selected	78,74	0,77	0,82	0,79

Table E.1: New FE using 10-fold CV.

Feature set	Accuracy	Precision	Recall	AUROC
All Features	77,32	0,76	0,8	0,77
CAE	74,12	0,72	0,79	0,74
ORAE	76,88	0,75	0,81	0,77
PC	77,68	0,76	0,81	0,78
RFAE	73,04	0,72	0,75	0,73
Most Selected	74,41	0,72	0,80	0,74
All Selected	**78,27**	**0,77**	**0,81**	**0,78**

Table E.2: Old FE using 10-fold CV.

Feature set	Accuracy	Precision	Recall	AUROC
All Features	83,32	0,83	0,84	0,83
CAE	81,30	0,85	0,77	0,81
CfsSubsetVal	86,35	0,89	0,83	0,86
GRAE	86,35	0,89	0,83	0,86
IGAE	**89,98**	**0,95**	**0,85**	**0,90**
ORAE	**89,98**	**0,95**	**0,85**	**0,90**
PC	**89,98**	**0,95**	**0,85**	**0,90**
RFAE	**89,98**	**0,95**	**0,85**	**0,90**
SUAE	**89,98**	**0,95**	**0,85**	**0,90**
WSE	**89,98**	**0,95**	**0,85**	**0,90**
Most Selected	**89,98**	**0,95**	**0,85**	**0,90**
All Selected	**89,98**	**0,95**	**0,85**	**0,90**

Table E.3: New FE using Train/Test.

Feature set	Accuracy	Precision	Recall	AUROC
All Features	85,33	0,85	0,85	0,85
CAE	84,97	0,82	0,89	0,85
ORAE	**89,5**	**0,91**	**0,87**	**0,90**
PC	**89,5**	**0,91**	**0,87**	**0,90**
RFAE	**89,5**	**0,91**	**0,87**	**0,90**
Most Selected	**89,5**	**0,91**	**0,87**	**0,90**
All Selected	**89,5**	**0,91**	**0,87**	**0,90**

Table E.4: Old FE using Train/Test.

Appendix F

Features Selected for the New FE.

Principal components + Ranker
Canny_S_Fractal_2_3
Canny_V_JPEG_20
Lighting_GaborSaliency
Palette_min_dist(V)
No_filter_H_Fractal_3_6
Palette_min_dist(ALL)
Mean_of_differences_in_occurrences
TamuraContrast
Pallete_Occurrences_1
Palette_max_dist(H)
WarmAndCool
Palette_min_pure(S)
RuleOfThirds_GaborSaliency
Blur
Palette_max_dist(H)

Table F.1: Selected features using principal components for the new FE.

CorrelationAttributeEval + Ranker
BGSimplicity
Pallete_Occurrences_0
Pallete_Occurrences_0_relative
ContrastingColors
SubjectSize_SpectralResidual
Canny_V_JPEG_60
Canny_V_JPEG_20
Canny_S_JPEG_60
Canny_S_JPEG_20
Canny_S_Fractal_2_3
Canny_V_Fractal_3_6
Canny_S_Nonzero
Canny_V_Fractal_2_3
Canny_S_Fractal_3_6
Canny_V_Nonzero

Table F.2: Selected features using CorrelationAttributeEval for the new FE.

GainRatioAttribueEval + Ranker
MichelsonContrast
Palette_mode(S)
No_filter_S_Fractal_3_6
No_filter_S_JPEG_60
avgPI
BGSimplicity
Palette_min_pure(H)
Lighting_GaborSaliency
SubjectSize_SpectralResidual
Palette_mode(H)
Pallete_Occurrences_0
Pallete_Occurrences_0_relative
ContrastingColors
RuleOfThirds_SpectralResidual
Canny_V_JPEG_20

Table F.3: Selected features using GainRatioAttribueEval for the new FE.

InfoGainAttributeEval + Ranker
BGSimplicity
Palette_mode(S)
RuleOfThirds_SpectralResidual
SubjectSize_SpectralResidual
Pallete_Occurrences_0_relative
Pallete_Occurrences_0
Palette_mode(H)
Avg_Distance_Next
MichelsonContrast
avgPI
Palette_avg_dist(V)
Std_of_differences_in_occurrences
ContrastingColors
Mean_of_differences_in_occurrences
Palette_mode(V)

Table F.4: Selected features using InfoGainAttributeEval for the new FE.

OneRAttributeEval + ranker
BGSimplicity
Palette_mode(S)
Palette_mode(H)
MichelsonContrast
Palette_mode(V)
Palette_min_pure(S)
Palette_min_pure(V)
Pallete_Occurrences_0_relative
Pallete_Occurrences_0
SubjectSize_SpectralResidual
Palette_max_dist(S)
RuleOfThirds_SpectralResidual
Avg_Distance_Next
Std_of_differences_in_occurrences
ContrastingColors

Table F.5: Selected features using OneRAttributeEval for the new FE.

ReliefFAttributeEval + Ranker
Palette_max_dist(V)
Palette_mode(S)
Palette_mode(V)
HueCount
Palette_max_pure(H)
Palette_min_dist(H)
Palette_max_dist(H)
Palette_mode(H)
avgPI
SubjectSize_SpectralResidual
Palette_min_pure(H)
No_filter_S_JPEG_60
Canny_V_JPEG_60
Palette_max_pure(S)

Table F.6: Selected features using ReliefFAttributeEval for the new FE.

SymmetricalUncertAttributeEval + ranker
MichelsonContrast
Palette_mode(S)
BGSimplicity
avgPI
SubjectSize_SpectralResidual
Palette_mode(H)
Pallete_Occurrences_0
Pallete_Occurrences_0_relative
RuleOfThirds_SpectralResidual
ContrastingColors
Lighting_GaborSaliency
Avg_Distance_Next
Std_of_differences_in_occurrences
Canny_V_JPEG_60
Canny_V_JPEG_20

Table F.7: Selected features using SymmetricalUncertAttributeEval for the new FE.

CfsSubsetVal +BestFirst
avgPI
TamuraCoarseness
MichelsonContrast
ContrastingColors
BGSimplicity
RuleOfThirds_SpectralResidual
SubjectSize_SpectralResidual
Lighting_GaborSaliency
Pallete_Occurrences_0
Avg_Distance_Next
Palette_mode(H)
Palette_mode(S)
Palette_min_pure(H)
Palette_min_pure(V)
Palette_std_dist(V)
Canny_V_JPEG_60

Table F.8: Selected features using CfsSubsetVal for the new FE.

WrapperSubsetEval + BestFirst
MichelsonContrast
Liveliness
BGSimplicity
SubjectSize_SpectralResidual
Blur
Avg_Distance_Next
Palette_mode(H)
Canny_H_JPEG_60
No_filter_S_JPEG_60
No_filter_V_JPEG_60
Canny_V_JPEG_60
No_filter_S_Fractal_3_6
Canny_S_Fractal_2_3

Table F.9: Selected features using WrapperSubsetEval for the new FE.

Most Selected
SubjectSize_SpectralResidual
Palette_mode(H)
BGSimplicity
MichelsonContrast
ContrastingColors
Palette_mode(S)
Pallete_Occurrences_0
Avg_Distance_Next
avgPI
Pallete_Occurrences_0_relative
RuleOfThirds_SpectralResidual
Canny_V_JPEG_60
Canny_V_JPEG_20
Lighting_GaborSaliency
Palette_max_dist(H)

Table F.10: Most selected features for the new FE.

Top Total
Avg_Distance_Next
avgPI
BGSimplicity
Blur
Canny_S_Fractal_2_3
Canny_S_JPEG_60
Canny_S_JPEG_20
Canny_S_Nonzero
Canny_S_Fractal_3_6
Canny_V_JPEG_20
Canny_V_JPEG_60
Canny_V_Fractal_3_6
Canny_V_Fractal_2_3
Canny_V_Nonzero
Canny_H_JPEG_60
ContrastingColors
HueCount
Lighting_GaborSaliency
Liveliness
Mean_of_differences_in_occurrences
MichelsonContrast
No_filter_H_Fractal_3_6
No_filter_S_Fractal_3_6
No_filter_S_JPEG_60
No_filter_V_JPEG_60
Palette_min_dist(ALL)

Palette_min_dist(V)
Palette_min_dist(H)
Palette_min_pure(S)
Palette_min_pure(H)
Palette_min_pure(V)
Pallete_Occurrences_1
Palette_max_dist(H)
Palette_max_dist(S)
Palette_max_dist(V)
Palette_max_pure(H)
Palette_max_pure(S)
Palette_avg_dist(V)
Pallete_Occurrences_0
Pallete_Occurrences_0_relative
Palette_mode(S)
Palette_mode(H)
Palette_mode(V)
Palette_std_dist(V)
RuleOfThirds_GaborSaliency
RuleOfThirds_SpectralResidual
SubjectSize_SpectralResidual
Std_of_differences_in_occurrences
TamuraContrast
TamuraCoarseness
WarmAndCool

Table F.11: All selected features for the new FE.

Appendix G

Features Selected for the Old FE.

Principal components + Ranker
FULL_IMAGE-CANNY_T(S)-(STD)
FULL_IMAGE-DISTRANS(S)-(JPEG_40)
FULL_IMAGE-NO_FILTER(CS)-(AVG)
FULL_IMAGE-SUB_SALIENCE(H)-(JPEG_20)
FULL_IMAGE-NO_FILTER(V)-(AVG)
FULL_IMAGE-CANNY_V(CS)-(ZIPF_RANK_FREQ_TRENDLINE)
FULL_IMAGE-NO_FILTER(V)-(ZIPF_SIZE_FREQ_TRENDLINE)
FULL_IMAGE-DISTRANS(V)-(AVG)
FULL_IMAGE-CANNY_T(V)-(ZIPF_RANK_FREQ_R2)
FULL_IMAGE-SUB_SALIENCE(H)-(NORM)
FULL_IMAGE-NO_FILTER(H)-(CS_ANGLE)
FULL_IMAGE-BG_SALIENCE(V)-(STD)
FULL_IMAGE-SOBEL_V(V)-(STD)
FULL_IMAGE-NO_FILTER(V)-(STD)
FULL_IMAGE-SOBEL_V(H)-(NORM)

Table G.1: Selected features using principal components for the old FE.

CorrelationAttributeEval + Ranker
FULL_IMAGE-NORM_DISTRANS(V)-(FRACTAL_LOW2-3)
FULL_IMAGE-SOBEL_V(H)-(STD)
FULL_IMAGE-SOBEL_V(H)-(NORM)
FULL_IMAGE-NORM_DISTRANS(V)-(ZIPF_RANK_FREQ_TRENDLINE)
FULL_IMAGE-DISTRANS(V)-(ZIPF_RANK_FREQ_TRENDLINE)
FULL_IMAGE-SUB_SALIENCE(V)-(AVG)
FULL_IMAGE-NORM_DISTRANS(ALL_CHANNELS)-(JPEG_60)
FULL_IMAGE-NORM_DISTRANS(CS)-(FRACTAL_LOW2-3)
FULL_IMAGE-SOBEL_T(H)-(NORM)
FULL_IMAGE-NORM_DISTRANS(ALL_CHANNELS)-(JPEG_40)
FULL_IMAGE-SOBEL_T(H)-(STD)
FULL_IMAGE-NORM_DISTRANS(S)-(FRACTAL_LOW2-3)
FULL_IMAGE-NORM_DISTRANS(V)-(FRACTAL_NORMAL2-5)
FULL_IMAGE-NORM_DISTRANS(S)-(ZIPF_RANK_FREQ_TRENDLINE)
FULL_IMAGE-DISTRANS(S)-(ZIPF_RANK_FREQ_TRENDLINE)

Table G.2: Selected features using CorrelationAttributeEval for the old FE.

OneRAttributeEval + ranker
FULL_IMAGE-SUB_SALIENCE(S)-(ZIPF_SIZE_FREQ_TRENDLINE)
FULL_IMAGE-BG_SALIENCE(S)-(ZIPF_SIZE_FREQ_R2)
FULL_IMAGE-SUB_SALIENCE(V)-(AVG)
FULL_IMAGE-NORM_DISTRANS(V)-(JPEG_20)
FULL_IMAGE-NORM_DISTRANS(V)-(JPEG_40)
FULL_IMAGE-NO_FILTER(S)-(ZIPF_SIZE_FREQ_TRENDLINE)
FULL_IMAGE-NORM_DISTRANS(V)-(FRACTAL_NORMAL2-5)
FULL_IMAGE-NORM_DISTRANS(ALL_CHANNELS)-(JPEG_40)
FULL_IMAGE-SOBEL_V(H)-(AVG_ANGLE)
FULL_IMAGE-NORM_DISTRANS(V)-(ZIPF_RANK_FREQ_TRENDLINE)
FULL_IMAGE-CANNY_T(ALL_CHANNELS)-(JPEG_60)
FULL_IMAGE-NORM_DISTRANS(V)-(JPEG_60)
FULL_IMAGE-NORM_DISTRANS(CS)-(FRACTAL_HIGH3-6)
FULL_IMAGE-BG_SALIENCE(S)-(AVG)
FULL_IMAGE-DISTRANS(V)-(STD)

Table G.3: Selected features using OneRAttributeEval for the old FE.

ReliefFAttributeEval + Ranker
FULL_IMAGE-BG_SALIENCE(H)-(AVG_ANGLE)
FULL_IMAGE-GIFQUANTIZATION(V)-(AVG)
FULL_IMAGE-BG_SALIENCE(H)-(CS_ANGLE)
FULL_IMAGE-SUB_SALIENCE(V)-(AVG)
FULL_IMAGE-GIFQUANTIZATION(H)-(STD)
FULL_IMAGE-SUB_SALIENCE(H)-(STD)
FULL_IMAGE-BG_SALIENCE(V)-(STD)
FULL_IMAGE-NO_FILTER(H)-(STD)
FULL_IMAGE-BG_SALIENCE(H)-(STD)
FULL_IMAGE-NO_FILTER(H)-(AVG_ANGLE)
FULL_IMAGE-GIFQUANTIZATION(H)-(AVG_ANGLE)
FULL_IMAGE-SUB_SALIENCE(H)-(AVG_ANGLE)
FULL_IMAGE-NO_FILTER(H)-(NORM)
FULL_IMAGE-GIFQUANTIZATION(H)-(NORM)

Table G.4: Selected features using ReliefFAttributeEval for the old FE.

Most Selected
FULL_IMAGE-NO_FILTER(V)-(AVG)
FULL_IMAGE-BG_SALIENCE(V)-(STD)
FULL_IMAGE-SOBEL_V(H)-(NORM)
FULL_IMAGE-NORM_DISTRANS(V)-(ZIPF_RANK_FREQ_TRENDLINE)
FULL_IMAGE-NORM_DISTRANS(V)-(FRACTAL_NORMAL2-5)
FULL_IMAGE-NORM_DISTRANS(ALL_CHANNELS)-(JPEG_40)
FULL_IMAGE-SUB_SALIENCE(V)-(AVG)

Table G.5: Most selected features for the old FE.

Top Total
FULL_IMAGE-CANNY_T(S)-(STD)
FULL_IMAGE-DISTRANS(S)-(JPEG_40)
FULL_IMAGE-NO_FILTER(CS)-(AVG)
FULL_IMAGE-SUB_SALIENCE(H)-(JPEG_20)
FULL_IMAGE-CANNY_V(CS)-(ZIPF_RANK_FREQ_TRENDLINE)
FULL_IMAGE-NO_FILTER(V)-(ZIPF_SIZE_FREQ_TRENDLINE)
FULL_IMAGE-DISTRANS(V)-(AVG)
FULL_IMAGE-CANNY_T(V)-(ZIPF_RANK_FREQ_R2)
FULL_IMAGE-SUB_SALIENCE(H)-(NORM)
FULL_IMAGE-NO_FILTER(H)-(CS_ANGLE)
FULL_IMAGE-SOBEL_V(V)-(STD)
FULL_IMAGE-NO_FILTER(V)-(STD)
FULL_IMAGE-NORM_DISTRANS(V)-(FRACTAL_LOW2-3)
FULL_IMAGE-SOBEL_V(H)-(STD)
FULL_IMAGE-DISTRANS(V)-(ZIPF_RANK_FREQ_TRENDLINE)

FULL_IMAGE-NORM_DISTRANS(ALL_CHANNELS)-(JPEG_60)
FULL_IMAGE-NORM_DISTRANS(CS)-(FRACTAL_LOW2-3)
FULL_IMAGE-SOBEL_T(H)-(NORM)
FULL_IMAGE-SOBEL_T(H)-(STD)
FULL_IMAGE-NORM_DISTRANS(S)-(FRACTAL_LOW2-3)
FULL_IMAGE-NORM_DISTRANS(S)-(ZIPF_RANK_FREQ_TRENDLINE)
FULL_IMAGE-DISTRANS(S)-(ZIPF_RANK_FREQ_TRENDLINE)
FULL_IMAGE-SUB_SALIENCE(S)-(ZIPF_SIZE_FREQ_TRENDLINE)
FULL_IMAGE-BG_SALIENCE(S)-(ZIPF_SIZE_FREQ_R2)
FULL_IMAGE-NORM_DISTRANS(V)-(JPEG_20)
FULL_IMAGE-NORM_DISTRANS(V)-(JPEG_40)
FULL_IMAGE-NO_FILTER(S)-(ZIPF_SIZE_FREQ_TRENDLINE)
FULL_IMAGE-SOBEL_V(H)-(AVG_ANGLE)
FULL_IMAGE-CANNY_T(ALL_CHANNELS)-(JPEG_60)
FULL_IMAGE-NORM_DISTRANS(V)-(JPEG_60)
FULL_IMAGE-NORM_DISTRANS(CS)-(FRACTAL_HIGH3-6)
FULL_IMAGE-BG_SALIENCE(S)-(AVG)
FULL_IMAGE-DISTRANS(V)-(STD)
FULL_IMAGE-BG_SALIENCE(H)-(AVG_ANGLE)
FULL_IMAGE-GIFQUANTIZATION(V)-(AVG)
FULL_IMAGE-BG_SALIENCE(H)-(CS_ANGLE)
FULL_IMAGE-GIFQUANTIZATION(H)-(STD)
FULL_IMAGE-SUB_SALIENCE(H)-(STD)
FULL_IMAGE-NO_FILTER(H)-(STD)
FULL_IMAGE-BG_SALIENCE(H)-(STD)
FULL_IMAGE-NO_FILTER(H)-(AVG_ANGLE)
FULL_IMAGE-GIFQUANTIZATION(H)-(AVG_ANGLE)
FULL_IMAGE-SUB_SALIENCE(H)-(AVG_ANGLE)
FULL_IMAGE-NO_FILTER(H)-(NORM)
FULL_IMAGE-GIFQUANTIZATION(H)-(NORM)
FULL_IMAGE-NO_FILTER(V)-(AVG)
FULL_IMAGE-BG_SALIENCE(V)-(STD)
FULL_IMAGE-SOBEL_V(H)-(NORM)
FULL_IMAGE-NORM_DISTRANS(V)-(ZIPF_RANK_FREQ_TRENDLINE)
FULL_IMAGE-NORM_DISTRANS(V)-(FRACTAL_NORMAL2-5)
FULL_IMAGE-NORM_DISTRANS(ALL_CHANNELS)-(JPEG_40)
FULL_IMAGE-SUB_SALIENCE(V)-(AVG)

Table G.6: All selected features for the old FE.