Final Year Project Report

Makeup Recommendation

Rory O'Reilly

A thesis submitted in part fulfilment of the degree of

BSc. (Hons.) in Computer Science

Supervisor: Dr. Chris Bleakley

UCD School of Computer Science
University College Dublin

April 8, 2016

Project Specification

The aim of this project is to develop a mobile application that provides consumers with personalised recommendations on makeup choice that users should select to give the best results for their complexion and facial features.

With so many different products and colours in the market it can be daunting choosing which ones will complement a person's complexion, hair colour or eye colour. Also, there can be many different ways in which make-up can be applied that suit a face shape, or facial features. The application should give the user help with both of these problems. Providing a simple picture of themselves will allow the software to create a list of suggested types of makeup, including lipstick, foundation, blush, mascara, eye shadow etc. that should suit the user. The application should also give information on how to apply the makeup in a style that suits them.

The project will develop a prototype mobile application using open source facial recognition software. The application will extract a number of features from the image. These features will be used to find the best matching make-up recommendations held in a pre-stored database. The database will be populated based on advice from experts and magazines.

The software will be developed on Android using the Face++ facial recognition SDK.

Mandatory:

Create an application that, given a user's head and shoulders photograph, will return a textual list of suggested make-up products suited to their complexion, hair colour, and lip colour (i.e. colour based recommendations). The suggested products will include blush, lip stick/gloss, foundation, eye shadow, concealer, etc. and options for different occasions.

Discretionary:

Add a feature to make make up recommendations based on the shape of facial features (i.e. shape based recommendations). The chosen facial recognition software will give the different points of reference on the face.

Exceptional:

Evaluation and improvement of the accuracy of recommendation compared to expert human recommendation. A makeup expect will be used to judge if more suitable products and styles are recommended by the application.

Abstract

The aim of this project was to design a mobile application, which recommended a set of makeup products for a user based on a photograph. This project completed the mandatory, discretionary, and exceptional as stated above. The main work of this product consisted of extracting and classifying a person's facial features; these include, skin complexion, eye colour, hair colour, lip colour and face shape. These classified features were all used in different ways to recommend the set of makeup products. The app also gave personalised makeup looks for different occasions. The accuracy of the eye, hair and face shape classifications were 87%, 87%, and 57% respectively. The recommendations were considered by a makeup expert to be reliable.

Acknowledgments

Sincere thanks to Dr. Chris Bleakley for his mentorship and supervisor during the project. He provided valuable insight and guidance throughout the project. Another thank to Rachel O'Reilly, whose expertise in makeup was essential to the project.

Table of Contents

Chapter 1: **Introduction**

1.1 Motivation

The global cosmetic market was 460 billion USD in 2014 and is estimated to have a compound annual growth rate of 6.4% over the next five years. [8] Consumers have been spending more and more disposable income in this market and every year there are more young adolescent females joining it.

It can be daunting for a person coming into the makeup market trying to choose the correct products from the wide variety available to them. For make up to enhance one's attributes one must consider a person's complexion, skin texture, hair and eye colour and face shape.

Makeup professionals are trained to understand this balance and know what work well together. However, going into a store looking for advice has its difficulties. The outcome can depend on the personal taste of the makeup professional or the products that they are promoting. It is also time consuming and a full make up consultation is expensive, often ending with the consumer expected to buy a certain amount of products.

For these reasons and in an age where technology is so advanced, there should be a way to automate the make-up selection process and make it simpler for the customer. There is a need to reduce the complicated variables that a person might encounter. Processes such as facial recognition would help automate this experience and technology should be used to make choices more accurate.

1.2 Approach

The approach to this this project was broken down into a number of different areas. Background research, current papers and related systems will be the first step. One of the areas of focus for this background research is to find papers on recommendation systems using facial recognition software. Extraction of skin colour, hair colour, eye colour or similar was investigated. The ability to classify an individuals' face shape or other object shapes was also important. Machine-learning techniques to classify data and create a model for recommendation from data was researched. Finally, knowledge was sought in colour theory and facial recognition. Two areas of colour theory that are important to learn about deal with the ability to blend colours that are appealing to the eye, as well as, how the science of colour interacts with technology.

Designing the project was then followed. A number of features were required for the recommender system - skin complexion, eye colour, hair colour, lip colour, and face shape. The work on designing the project was included extraction of data on each these features. Following the extraction, testing different methods of classification was done for each feature to find the most suitable.

Lighting is a major problem in this project. Lighting changes based on the time of day, location of the user, and how shadows hit their face. Work will be done to resolve this problem.

These features along with an occasion are used to create the makeup recommender. Every product was recommended in their own way to ensure they are suitable for the user and complement the other products.

The app will be tested and evaluated to find the accuracy of the feature classification and makeup recommendation.

1.3 Report Structure

The introduction defines the project, as well as, the motivation for such an application.

The next chapter is the background research that has been conducted. Many research papers have been valuable for this project. This is not only because they are related to the project, i.e. skin detection, eye classification, detection and analysis of hair but also because they deal with important concepts that can be used for the project. Research into similar systems helped in determining some important problems in the technological makeup industry.

The project is based around some core subjects namely, colour theory, machine learning and facial recognition. Research into these built a foundation for the project and gave an understanding on how to approach problems that may be faced.

Following background research, the software functionality for the user is given. This area describes what the user sees and experiences when using the app.

Chapter 4, contains the design algorithms that were created for the project. It also gives a the final algorithms for the app. The chapter begins with a overview of how the app works. It leads onto feature extraction, which is how the skin complexion, eye colour, hair colour, lip colour, and face shape attributes from the user image are taken. Algorithms on blending colours follows. The chapter then deals with feature classification. Two algorithms on light normalisation are given. Finally the chapter concludes with the design approach for the makeup recommendations.

Implementation is found in chapter 5. This discusses the main toolkits and implementation of areas such as the product database.

The timeline in the project are given in chapter 6. This chapter also deals with the main challenges encountered, such as, facial image dataset.

The results of the feature extraction and classification are presented and explained in the penultimate chapter. Here, the method for evaluation of the makeup recommendation is given. An evaluation of the app itself, is also discussed.

The report finishes with a conclusion and future work in the project.

Chapter 2: **Background Research**

2.1 Related Publications

There have been more patents than research papers, relating to makeup product and style recommendation. Na (surname) discusses a recommendation system of hairstyle, makeup style or product information based on features and style features extracted from a user image in his patent [20]. It is a broader idea than this paper because of the possibility of using it for season and weather information, apparel patterns, and the user's age. The inventors of [15] created a system that will perform colour correction techniques on a single image, extract the user's skin colour and recommend at least one product based on this skin colour. Finally the inventors of [25] suggest at least one beauty product from local information (cosmetic usage in the user's location as an example) and personal information (including physical characteristics of the user). This invention shows that there can be many aspects that can alter an individual's choice in using a beauty product.

A closely related research paper is [11]. The main theme of the paper was makeup harmonising with eye colour. The paper discussed a method for extracting a user's eye colour from an image and the classification of the extracted eye colour. The paper gave great insight into the design of the eye colour classifier for this project due to the author's use of a professional makeup artist and well-presented results.

A number of papers have been written in the area of skin colour detection and classification under different lighting conditions. Liu, Sang, Yang, and Huang focused very much on skin colour correction on each frame of a video in their paper [17]. Even though it was performed on video input it gave knowledge of colour correction such as the White Patch algorithms. The enhancing of images to achieve a favourite skin tone is the topic of [19]. The use of converting RGB values to HSB values for the colour space is the interesting aspect of this paper. An interesting technique to calibrate images to achieve very clear skin colour is used in [28]. In this paper a user holds up a sheet of paper with an array of colours. Their system uses this array of colours to correct the image and classify the skin colour. It makes it quite difficult for a user. However, it is an interesting technique and might be useful if other colour correcting methods fail. Finally mobile phone as imaging sensors is used in [3]. It contains a section on finding the skin colour to be used as a classifier for foundation makeup recommendation. It uses the same technique as [28] and therefore, achieved good results.

2.2 Similar Systems

2.2.1 Plum Perfect

Plum Perfect is an application for both iPhone and Android, built with a similar concept. The application uses a profile image of the user to create a colour signature for the skin, lips, eyes and hair. These colour signatures into terms such as soft, neutral, dark, cool, warm.

Makeup products recommended from the colour signature are presented to the user as combinations of face/ lip/ eye or as entire looks or styles. There are six of these looks including 'Everyday Essentials', and 'Night Out'.

A user must be in a bright naturally lit shadow-free environment to get accurate results. This may have been a design approach taken by the developers because of colour normalisation proving too difficult to get correct results. It has come a long way from the beginning of its development. [16]

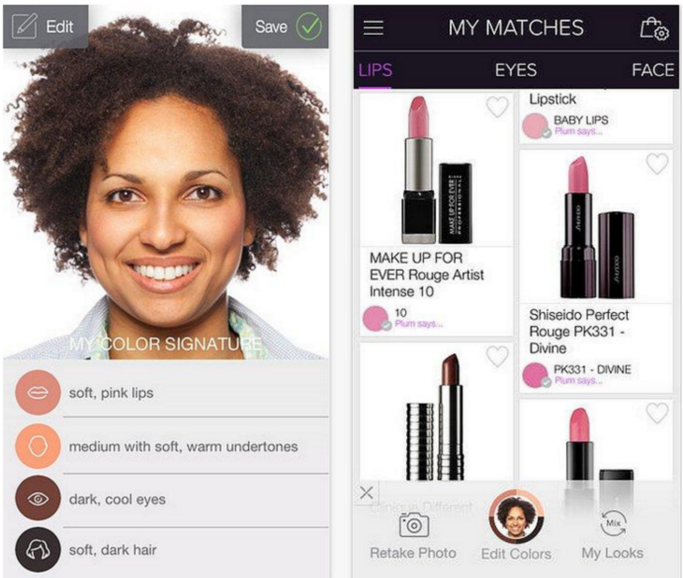


Figure 2.1: Plum Perfect's colour signature screen and lip recommendation screen

2.2.2 Pantone Capsure & Sephora Color IQ

Pantone Capsure & Sephora Color IQ is the combination of a device and a library that can give makeup recommendation to an individual in a store. Research into this area has shown that it can yield quite accurate results. [6]

The Pantone Capsure is a device that can determine the colour of the surface the device is put against. There are many ways an object can change colour due to lighting and the device solves this by only using ambient light. All other light sources and variables are removed by placing the device against the surface of the object. Pantone Capsure can classify a surface to 10,000 possible colours due to the use of an array of filters inside the device. [21]

Sephora Color IQ is a skin colour library of 110 skin tones. The skin tones in the library are mapped to different foundations and concealers. The result from the Pantone Capsure is used to find a customers skin tones in the library and therefore, a matching to one of the foundations. [26]

2.2.3 Virtual Makeup Applications

Virtual Makeup Applications are a similar idea to this research project. The user inputs their face to the application by means of a real-time video or an image and they select makeup project they would like to 'try on'. The application will simulate how the user would look with the makeup and return the result. L'Oreal Makeup Genius is an example of this type of application. [18]

A design approach to this is using two images, the user's face image and a style example image (image with the makeup on). These images are separated into three layers, a face structure layer, skin detail layer and the colour layer. Transferring the skin detail layer and colour layer of the style image to the face structure layer of the user's image creates the final image. [14]

2.3 Technical areas

2.3.1 Colour Theory

In colour theory, two areas of interest related to this project are in the art of colour and the technical aspect of colour. The art of colour focuses on the global harmonising of the colour from facial features and the colour of makeup used.

There is psychology behind colours. A red is passionate and energetic while a blue creates a soothing effect. With colour you can set a mood, attract attention, or make a statement. By selecting the right colour scheme, you can create an ambiance of elegance, serenity, or you can convey an image of youthfulness. The basics of colour combinations can be explained with the colour wheel. The colour wheel is split into three groups, primary colours (blue, red, and yellow), secondary colours (green, orange, violet) which are a combination of primary colours, and tertiary colours which are a combination of a primary and a secondary colour. The colour wheel can be used to select complementary colours, or analogous colours. Complementary colours are colours opposite on the colour wheel and create a strong visual contrast between each other. Analogous colours are colours close to each other on the wheel and can create a better colour blending effect.

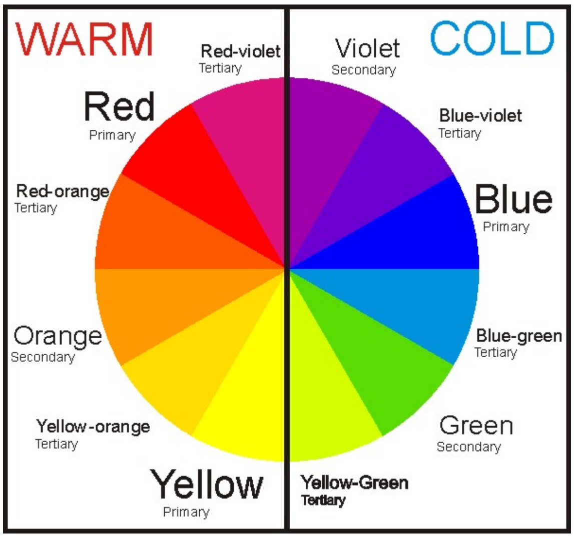


Figure 2.2: Colour Wheel showing primary, secondary and tertiary colours. This chart also shows the difference in warm colours and cool colours

Makeup is not only about special events where strong colour balancing and feature enhancing is done. Makeup is used for everyday wear as well and colour theory helps in selecting neutral colours that complement a person's hair, eye and skin colour. These can be subtle colour differences, for example, it is recommended an individual with blonde hair use ash blond, and soft grey for the eyebrow pencils, shadows, and shaper tones whereas an individual with brunette hair should use a mahogany, or warm brown. [22]

Due to the use of facial recognition software and a camera device in the application, technical knowledge on colour is required. There can be a large separation in colour from one image to another due to the lighting. The amount of light at the images location, whether natural or artificial light is present, presence of shadows, and camera quality can affect the colour. [23] Therefore, extrapolating the colour or the skin, hair and eyes yields inaccurate results in most cases. Colour normalisation is required. Algorithm such as White Patch [24] could be useful but may be too time intensive for this project.

The choice of colour space is another vital technical area. Colour space is the abstract configuration describing how colour impressions can be created. RGB colour space (blending of red, green, and blue values) is the most commonly used in hardware and software. However, luminance and chrominance are difficult to tease apart in RGB unless they are converted to another colour. [7] [27] Therefore, the use of another colour space is required and the two that have been found to be most interesting are HSB and CIE L*a*b*. HSB stands for hue, saturation, and brightness. It is a cylindrical representation of the colour space. The hue is similar to the colour wheel. It is a continuous 360 degree of the chroma value. CIE L*a*b* is a colour space created to be more perceptually linear than other colour spaces. The L component matches human perception of lightness quite well. Each has their own advantages.

2.3.2 Machine Learning

Machine learning plays a vital in this project. To get an advanced and accurate recommendation model it is used not only to create a model for the suggested makeup products based on the skin colour, eye colour, hair colour, and face shape but also in determining the skin colour, eye colour, hair colour, and face shape classifiers. Machine learning algorithms can accept a set of data and discover a pattern between this data. The pattern is represented in a model and this model is used to determine which class a person belongs to from their data. [4]

Weka is a Java open software workbench. It provides many machine learning algorithm that can be used to classify features. It also contains many visualisation tools which can be used easily in the testing phase.

There are two types of machine learning algorithms, supervised and unsupervised. Supervised algorithms use a set of training examples to make its model. Unsupervised algorithms use unlabelled data and finds hidden patterns using exploratory data analysis. Consequently supervised algorithms is employed in this project. There are many supervised algorithms and it is important to select the algorithm that best fits the required model.

Support Vector Machine (SVM) is one algorithm. SVM are based on the concept of decision planes that define decision boundaries. [4] SVM is known as a two-class algorithm. This means that it can separate inputted data into one of two classes. However, multiclass SVM is possible by reducing a multiclass problem into multiple binary class problems. [9]

Naive Bayes algorithm is a probabilistic classifier. It is based on the Bayes' Theorem. It is assumed that an attribute is independent of every other attribute. [1]

Decision trees are another set of classifiers. They create a tree to determine how to classify a new feature. Decision trees are very readable and easily interpreted by a human and can be useful in this project. If it is found too difficult to introduce machine learning models into a mobile application, a decision tree can be used to easily convert it to code. One such decision tree is C4.5, also known in Weka as J48.

2.3.3 Facial Recognition

Facial recognition software is used to help in the collection of data from a user. Old Facial Recognition systems represented the facial images as feature vectors. After obtaining these representations, various learning algorithms were applied to perform classification, verification or searching. The performance of a system depended heavily on the representation. New facial recognition systems use LBP (local binary pattern) descriptors. [10] Advanced systems can even use deep machine learning and multiple layers such as 2d alignment, 3d alignment and frontalization. This has lead to Facial Recognition system's accuracy being matched only by humans. [27]

One of the best open source facial recognition software that can be used with Android is Face++. It has a very high accuracy of 99.5% in finding people's faces in images. It uses the LBP descriptor to describe each pixel in an image [10]. Each descriptor of the image is fed into as input to a nave-deep training toolset. [29] Some of the features of Face++, which put it beyond other facial recognition software, are its ability to give an estimate on the detected face's age, race and level of smiling. It can also gives many position coordinates or landmarks of the face. It finds 83 of these landmarks and they include positions of many points on the jaw line, the outline of the eye, eyebrow and mouth, as well as points for the pupil, nose and centre of the face. These points help immensely in searching for areas of important pixels for the colour extraction. It also helps very much in the face shape classification due to the jaw line coordinates.

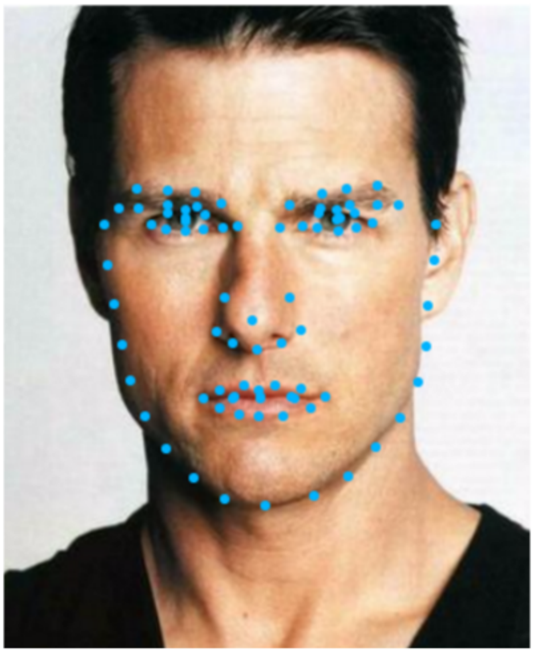


Figure 2.3: 83 landmark points from the Face++ detection

2.3.4 Android

This project is made to work on an android phone. It is important to know and understand how Android is different from normal Java development. Android has their own documentation, explanation, and many examples for developers on their webpage. [13]

Care must be taken when developing on the Android environment. An app can use sensors on the phone by declaring permission. However, some phones may not have a given sensor and there must be matters taken by the developer to ensure the app does not break due to this.

Memory is another issue on Android devices as phones are not as powerful as computers and applications are not allocated as much storage and RAM. Designing the application with these in mind is needed.

Android devices are more prone to not having constant internet connection opposed to a computer. As this project requires the internet, thought should be put into the development to see if there is any way to build the app when the user has no internet.

2.4 Facial Features

2.4.1 Skin Complexion

The skin colour, tone and undertone are critical for makeup and a chosen style. They is the basis of a person's complexion. The complexion is split into two ranges.

The first range is known as the warmth of the skin. It ranges from cool to warm. A cool skin tone is one, which is yellow/gold, while a warm skin tone is one, which is pink/red. Many makeup companies split this range into five classifies - cool, neutral cool, neutral, neutral warm and warm. The warmth classification for an individual can be found a number of ways such as if the individual sunburns or tans easily, the mixture of their hair and eye colour, or what colours which suit them. However, one of the best ways to determine an individual's skin warmth is to compare the colour of their veins on the underside of the wrists. A blue/purple colour represents a cool tone, while a green vein colour represents a warm tone. [5]

The second range is the lightness or darkness of the skin. The darker skin the darker the shadow of the undertone will be selected. This is usually found by finding the closest match along the individual's skin warmth chart.

A person's complexion is used to decide other makeup products. The most important product of these is the foundation. Foundation creates an even, uniform colour to a person's face. It is obvious when a person wears foundation that does not match their complexion. Many companies have many different colour charts for their foundation and this can make it difficult for a person to find a similar competitor's shade i.e. NC15 in Fix Fluid and NC15 in Studio Sculpt are different. [12]

2.4.2 Eye Colour

The eyes are often considered in the cosmetics industry as one of the most important features of the face. [12] The choice of eye shadow, mascara, and eyeliner is important to help in the global harmony of the face. The eye colour is important as the makeup surrounding it should blend well with it. The eye colour can also decide if makeup should enhance the eyes or not. One of the greatest aspect of makeup is its ability to enhance features in a person's face. An example of this can be found with bright blue eyes. Subtle colour choice in makeup around the eyes can help the beautiful blue become an impacting feature. [2]

Generally, three eye colour classifications (blue, brown, green) suffice for amateur makeup. However, for a professional artist the tones (cold, warm) and the brightness (light, dark) of the eyes matter more. The most common eye colours can be split into these two labels Cold, Hybrid, Warm and Light, Medium, Dark [11]. However, this larger breakdown may be too detailed and harder to classify accurately.

2.4.3 Hair Colour

It is important to harmonise makeup with hair colour because the hair provides a frame and an accent for the face. If a woman decided to change her hair colour, new makeup is required to be in harmony with it. The hair colour is important for eyebrow pencils. However, it can also be used to select lipstick, lip-gloss, blush, and matching colours around the eyes. [5]

Hair can be classified into many colours in the modern age due to hair dye and possibly hair extensions. However, covering all those possibilities will lead to less work in other areas. Therefore, four basic colour groups blonde, red, brunette, black suffice for colour blending with the makeup style. A more advanced classifier could incorporate the lightness or darkness of each of these colours.

2.4.4 Lip Colour

It was discovered that lip colour is not very important for the makeup recommendation. On the other hand, it can help with the lipstick recommendation for an everyday occasion. While lip colour can be classified into different shades of warm and cool, similar to skin complexion, it is not essential for this app to classify them.

2.4.5 Face Shape

A makeup recommendation application should give enough knowledge to the user so that they can reproduce the look or style that was promised to them. Some first time users have no knowledge of each product is or where it should be applied. Understanding of contouring, highlighting and emphasising of features can be difficult for many users initially. Face shape is the best way to help complete the users look. Certain features of the face are emphasised because of the face shape. Likewise contouring and highlighting is almost completely depended on the face shape. [12]

There are nine face shapes - diamond, heart, inverted triangle, long, oval, rectangle, round, square and triangle. However, some of these can be grouped together or are very rarely found in individuals who wear makeup. Five face shape classes should be enough to cover it. These classes are Heart, Oblong, Oval, Round, Square]. [5]

Chapter 3: **Software Functionality**

3.1 User Interface

The app consists of four screens. These screens get the user's photograph, show information on the individual's different features, allow the user to select an occasion and present the makeup recommendations.

The app opens on a splash screen. This splash screen gets a photograph of the user. There are two options to get a photo, the user can select to take a photo there and then or select one from their library. Each of these use the Android media tools. These are new screens whereby the camera and image library can be used.

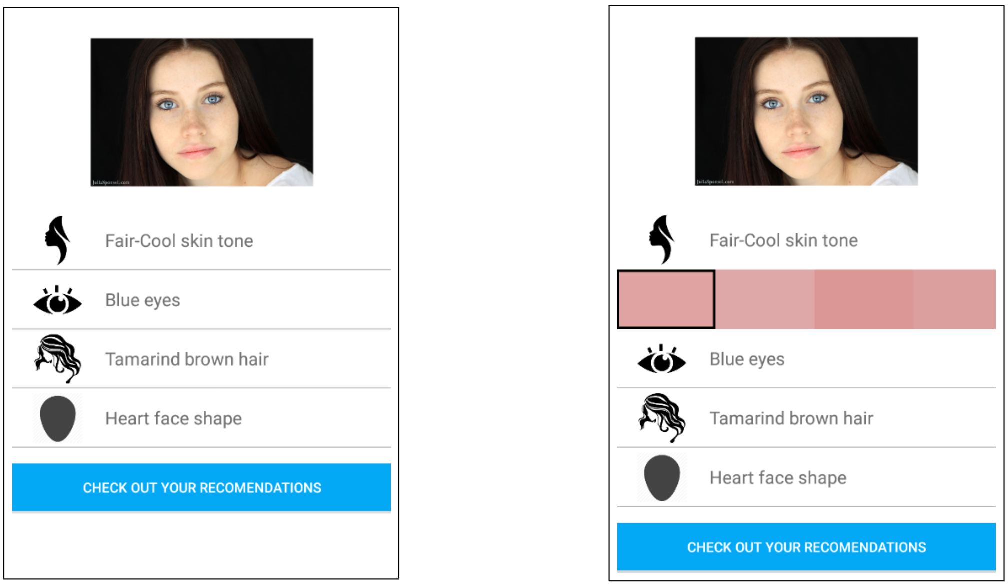
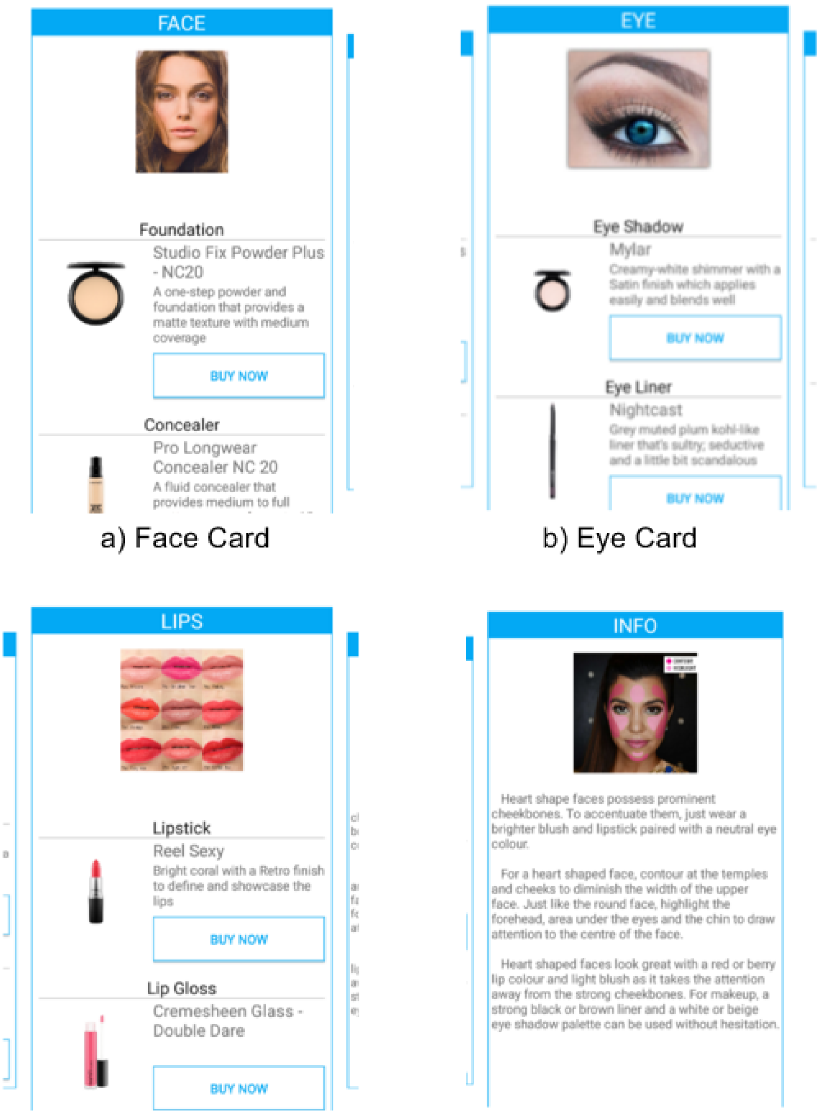


Figure 3.1: Face Profile Screen before and after the feature reselection

The next screen is the face profile screen. It displays the image of the user, the features list and a button to go to the next screen. The features list shows the classification of each - skin complexion, eye colour, hair colour, and face shape. If one of these features is incorrect the user has the ability to change the classifications to the correct one. This is done by clicking on the given feature, and reselecting from the horizontal list presented below it.

After the user has clicked the button at the bottom of the face profile screen, they are brought to the styles / occasion screen. This screen shows the possible styles that the user can choose. There are three images for each of the styles - Night out, Daytime, and Natural. Once one of these images is clicked the user is brought to the recommendation screen.

The Recommendation Activity shows the recommendations for the user. As the activity loads, the recommendations for every product is found for the user. The activity displays an array of cards. This number of cards is dependent on the occasion. Each card will display products in the groups; Face, Eyes, Lips, and Info. These cards cover the entire screen. To move to the next card the user can scroll horizontally. A card is always centered in the screen.



a) Face Card b) Eye Card

c) Lips Card d) Info Card

Figure 3.2: Makeup Recommendation screen for each of the the cards

A card has a title, an image relating to the given card, and either a description (in the case of the info card) or a list for the recommended products. Each product will display the type of product e.g. foundation, concealer, blush, the name of the recommended product, a description, an image of the product, and a button to 'Buy Now'. Every product has their own individual image, taken from the Mac web store. The 'Buy Now' button opens the browser to the exact product webpage. The user is able to buy the product from the Cosmetic merchant's website.

3.2 User Recommendations

There are three groups of makeup products - face makeup, eye makeup, and lip makeup. Each product complements the other to give an overall recommendation. There are 800 possible products that can be suggested to the user.

3.2.1 Face Makeup Recommendations

There are five different products in the face makeup - foundation, concealer, blush, highlighter, and bronzer.

The most important product is foundation. Foundation, as it name suggests is the base makeup product. The foundation shown to the user matches their skin complexion.

The concealer is based on the foundation. It should be the same undertone, either warm or cool (if the foundation was as such). It should also try to be a shade lighter than the foundation.

Bronzer is used to give depth to a person face. Therefore, it should be a darker tone than the foundation.

Highlighter is the opposite of bronzer, in that is should add lightness to areas of the face.

The final product in the face makeup is blusher. The blusher enhances the foundation and should vary with the occasion. It is applied to the cheeks to add emphasise in conjunction with the highlight. It also brings colour back to the skin over the foundation.

3.2.2 Eye Makeup Recommendations

The eye products are split into eye shadow, eye primer, mascara and eye liner. A natural look does not need any eye products, which made the work a little easier.

Eye shadow can be applied to the brow bone, the crease, the inner and outer lid, and the inner and outer lower lash line of the eye. It is commonly used to make the wearer's eyes stand out or look more attractive. Eye shadow can add depth and dimension to eyes, complement the eye colour, make one's eyes appear larger, or simply draw attention to the eyes. During daytime a single eye shadow is applied to slightly enhance the eyes. However, for night out makeup, eye shadow is more dramatic. Numerous eye shadows are required because of this and therefore, the app gives 4 products to the user.

There are special looks that can be created with different eyeliners but these are often for specialised situations. Generally speaking, the eyeliner is selected to match the darkest eye shadow. The recommendation can therefore, be seen as a whole.

Almost all mascaras are the same. Some will increase the thickness of the eye lashes while others increase the length. Extra thick and extra long mascara are more suited for night out styles.

Eye primer gets the eye ready for the makeup. It is needed as a base so that the eye shadow sticks longer. It does not add much to the makeup look but obviously still useful for the user.

3.2.3 Lips Makeup Recommendations

Lip makeup consists of lipstick, lip gloss and lip pencil. The most common suggestion which is to give a lipstick and lipgloss for the 'Night out' look while only giving the user a lipstick for the 'Daytime' look.

The lipstick for a 'Daytime' look should enhance the lips only slightly as the bright lights will make bright and vibrant lipstick look strange. The lipstick for the 'Night out' must match the user but it also takes into account the eye shadow being used.

Lipgloss is only chosen for a 'Night out' look and is recommended to give shine to the lips. It should not try blend or mix too much with the lipstick colour.

Lip Pencil gives defined shape to the lips and is only necessary for 'Night out' looks. It should blend with the lipstick.

3.2.4 Face Shape Info Recommendations

All the products are presented in cards in the given groups. However, a card with information on how to apply the makeup recommendations to best enhance the face shape is shown. An image is chosen based on the face shape. This shows locations where to apply bronzer and highlighter to help define the face. There are also details describing which features to enhance - for example a heart shape is suggested to use the bright lipstick.

Chapter 4: **Software Design**

In order to design and develop the app a set of facial landmarks for the face is acquired. Face++ is used. These landmark points are used to extract attributes for each of the features. From research it is found that skin complexion, eye colour, hair colour, lip colour, and face shape are the most appropriate features for makeup recommendation. The feature attributes are used with machine learning algorithms to classify the features. These features, together with an occasion, e.g. Night out, are used to recommend a set of makeup products.

4.1 Software Overview

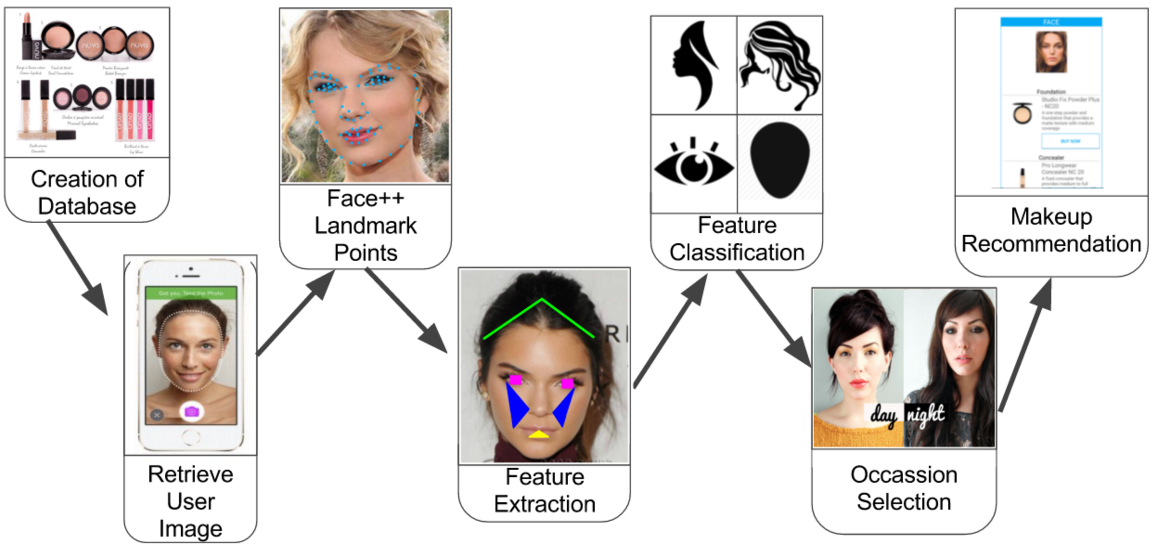


Figure 4.1: Step by Step overview of the application

As the app is turned on a database of the makeup products is created. There are 800 products, in total, in the app. Each of these are taken from the Mac range and contain different information regarding each product.

The app begins by getting an image of the user. This image is acquired as mentioned in the previous section. The image is passed to the next activity in the app.

The image is set to the correct format and sent to the Face++ servers. Face++ analyses the image and extract data on the face. This data includes the 83 landmark points on the face.

The result is returned from Face++ and the data, as well as the user image, is used to extract attributes for each of the features. The colour features, skin complexion, eyes, hair, and lips, will get a HSB value as it's attribute. There will be six different attributes extracted for the face shape.

The features are classified using the attributes for each. They are classified using either Weka, in the case of eye colour and face shape, or a nearest neighbour algorithm in the case of skin complexion and hair colour. The user is able to reselect the correct classification in the cases where the app is wrong.

The occasion is selected by the user after the feature classification. There are three occasions, Night out, daytime, and natural. The occasion selection as well as the feature classes are passed to the next activity in the app, where the makeup is recommended.

Finally, the makeup is recommended and presented to the user. Depending on the occasion there are 5 to 15 products recommended for use. Each of the product recommendations are made individually so that they will be suited for the user and are complementary to each other.

4.2 Feature Extraction

For each of the following features - skin complexion, eyes, hair, and lip the focus is on colour. The colour space chosen is HSB (Hue, Saturation, Brightness). To get this HSB colour, a set of colours for a feature must be found. In order to give an accurate average colour for the feature, this set must be reduced down to a single value. Face shape will have six different attributes which will be dealt with later in the report.

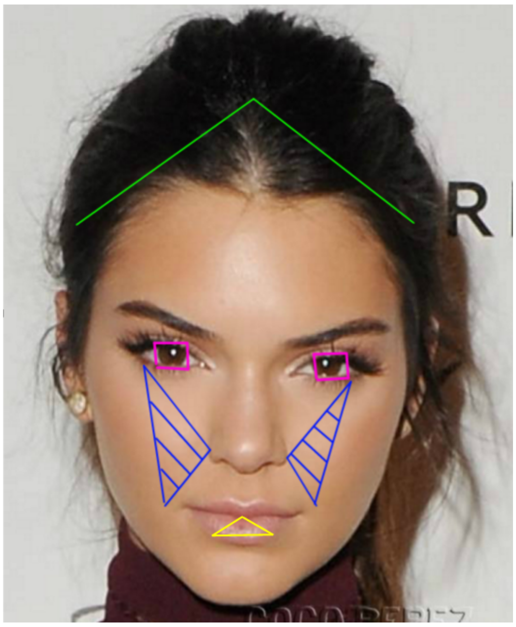


Figure 4.2: Locations for pixel colour extractions

4.2.1 Skin Complexion

It is difficult to get an average colour value of facial complexion, as some areas give different readings to others e.g. cheek and forehead. To counteract this, two different methods were tested.

The first algorithm involves getting the HSB colour value for different pixels around the face. An average is found. A plus-minus range is set from this average. The colour value for every pixel in the image inside this range is gathered. A new average is found. This is repeated a number of times until it converges. This can have favourable results when the skin colour is different from everything else in the image. However, if the background, hair, clothing, etc. have similar colour values the results are distorted.

The second algorithm is more successful and therefore, the final design choice. This method gets all pixels from a given area of the face. From research the best area is between the nose and cheek. Face++ does not provide any landmark points for the cheek. Therefore, a triangle using the corners of the mouth, nose, and eye is created (as seen in the blue area in figure 4.2). Both left and right sides of the face are used to get a better average. These give two triangles in which to get the pixel colour array. These pixel colours are found using a similar algorithm to the triangle rasterisation algorithm.

The second algorithm is chosen for the design of the project. The single attribute colour is found using this colour array and the colour selection technique discussed in section 4.3

4.2.2 Eye Colour

The eye colour is separated into blue, green and brown eyes. Once again, two techniques are attempted.

The first method is inspired by a research paper read [11]. This technique involves getting an image of the eye. Face++ does not give a large enough cover surrounding the eye for the algorithm to work. OpenCV is chosen instead. OpenCV has eye detection and can be used to get a single image of the eye.

Firstly, a gaussian filter is applied to the image to reduce noise. This is to prevent false detection of edges in the next step.

A canny edging technique is applied to the image. Canny edging turns the image into a black and white photograph where the white of the image are the edges. The canny edging algorithm finds a gradient intensity of the image over four different directions. The image is suppressed to thin the lines as there is some noise caused by the gradient intensity. A double threshold is used to chose the accepted intensity of the edges. The image is left with weak and strong edges. Some weak edges are removed if they are not connected to enough strong edges.

The final step uses a Hough Circle Transformation. This is applied to this created canny edge image. Hough Circle Transformation attempts to find a circle in an image. It tests circles throughout the image, over varying radius sizes. If a circle whose edges are found to be within the threshold of the canny edges, it is placed on the image.

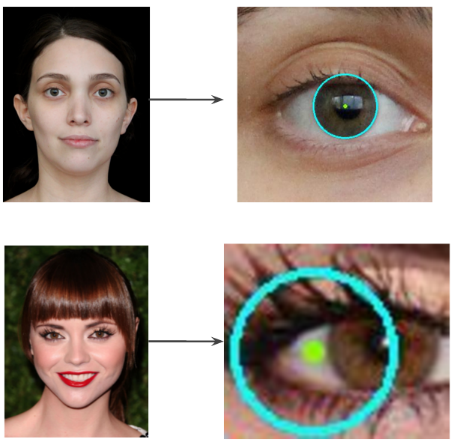


Figure 4.3: Results from the Canny edging and Hough Circle Transformation process

A problem was discovered after testing. Canny edging and Hough Circle Transformation use set threshold values. Changing the threshold can give many different results. Hough Circle Transformation gives more than one circle for a large threshold. In order to rectify this the technique has to be modified. The fix is to start with low threshold values and continually increase it until a single circle is found in the image. The iris is the most defined circle from an image of the eye and should be the first circle found. Therefore, this solution fixes the threshold problem.

This yields perfect results in a number of cases. However, in lesser quality images the found circle is incorrectly placed. If this occurs, it totally distorts the result.

The second method is to use points from Face++. A box is created in and around the iris. The corners are the interquartile points of the top and bottom lid of the eye (as seen in the magenta boxes in figure 4.2). To make this method more accurate, it attempts to only extract the vibrant colours. Therefore, colour pixels of black (from the pupil) and white/grey colours (from the sclera of the eye) are not included. What remains is an array of colours, which includes colours from the iris and possibly a select few from the skin. The colours from the skin should not affect the colour selection algorithm dealt with later in the report in section 4.3.

Using the points from Face++ proved to have better results and therefore, is the chosen design.

4.2.3 Hair Colour

The main focus is to classify hair into four different colours - blonde, brown, black, and red. The averaging and converging technique used with skin complexion is tested in relation to hair colour. However, once again the same errors occur.

A second method is developed and used in the app. This involves finds three points carefully estimated in relation to the upper face. The points are to the left, upper and right of the upper face. They are found using Face++'s 'center of the face' point. In Face++'s documentation, it is stated that rotation of the face in 3d space is given. This rotation (the roll angle), in conjunction with trigonometry is used to find the three points more accurately. The pixel colours are taken from a line connecting the left point to the upper point and the upper point the the right point (as seen in the green lines in figure 3.1). The Bresenham's line algorithm is used to get the pixels on the lines.

It should be noted at this point that Face++ never supplied this rotation in the JSON response. The reason for this is unknown. This algorithm works without the rotation, however, it will be more accurate once Face++ starts sending the rotation again.

4.2.4 Lip Colour

From research it is learnt that lip colour is the least important feature in makeup recommendation. However, it can be used for daytime lipstick choice. Due to this it is still worthwhile to extract the colour of the lips.

The colour pixel are taken from the bottom lip. The bottom lip is chosen as it gives a more accurate representation of the colour. This is because the upper lip will have more shadow. Three points are taken from the bottom lip. Again a similar triangle rasterisation technique is used to get all the pixel colours (as seen in the yellow triangle in figure 4.2).

4.2.5 Face Shape

The face shape is by far the most difficult feature of the app. The app classifies the face into five shapes - oval, heart, long, square and round. Six different attributes associated with the face shape are found and are as follows.

The first attribute is the width of the face over the height of the face. A long face shape should have the smallest value here, while, a round face will have the largest value.

The next value is the angle of the jaw line. This is found by using points from Face++ with the anchor point being at the interquartiles of jaw line. A square face shape should have the most acute angle and the oval and heart having the most obtuse angle. In order to reduce the error, the angle is found on both sides of the face and the values are averaged.

A curve fitting algorithm is used on the jaw line landmark points from Face++. The curve fitting algorithm uses a gradient descent technique to find the curves that is as close as possible to every point. Two curves are found; a curve of degree two (i.e. a quadratic) and a curve of degree four.

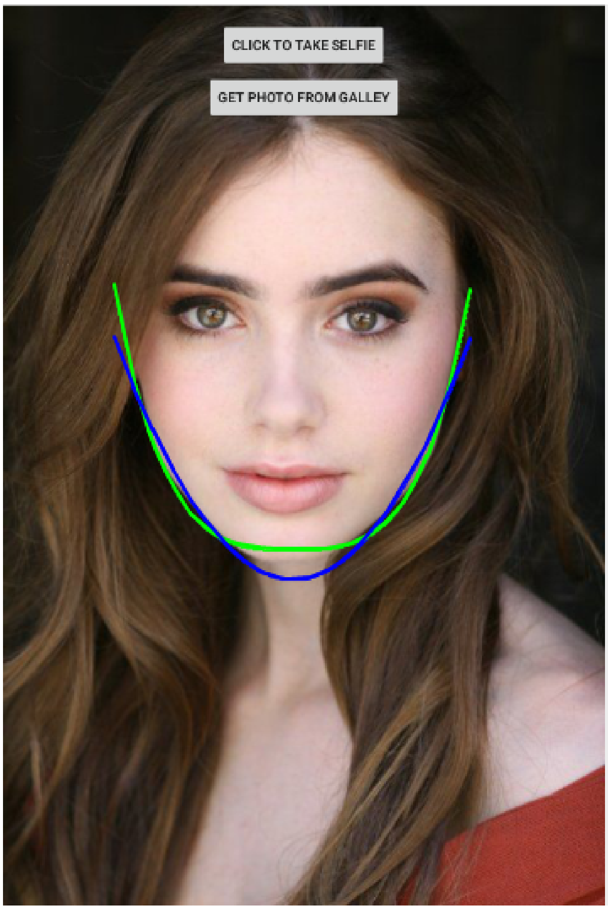


Figure 4.4: 83 landmark points from the Face++ detection

A total error is found using these two curves to the Face++ landmark points. The total error is the sum of the distance from the landmark points to the curves. The curve of degree four is more suited for square and long shapes compared to oval and heart shapes. The curve of degree two is more suited to oval and heart shapes as opposed to the square and long shapes. The curves for each person is different from person to person and depending on how far away they are from the camera. Therefore, a ratio of the two errors is found. This allows the value to be normalised while still giving information.

The next attribute finds the integral of the quadratic and compares it to the area inside the face for the given bounds of the curve (the bounds are the end Face++ jaw line points). Heart face shapes should have a higher value than the other shapes.

A line is created from the center of the face to the corner of the facial recognition box. This line is split by the quadratic. The distance of these two lines are then compared and create a value of the outside the face line over the inside line. Square shapes have a low value here while oval and hearts have a higher value.

The quadratic is lower than the chin in all cases. However, the curve will be a further distance from the chin for square and long face shapes than oval and heart shapes. The final attribute is based off of this. The distance from the chin to the quadratic is compared to the distance from the chin to the center of the face.

4.3 Colour Selection Algorithm

Each of the colour features extracts an array of pixel colours. However, the machine learning classifiers requires a single HSB colour value. The most difficult aspect of this area lies in the fact that mixing colours does not work well. Getting the mean of a set of RGB colours gives a dull single colour. Many cases this can be improved by using another colour space, such as HSB. However, this still produce a dull brown/ grey colour in many cases. The mean is not a good average in this case. Mode or median are normally better. However, getting a mode or median of a 3d space presents a number of problems.

4.3.1 Colour Palette Technique

The first technique is to make a colour palette from the set of colours. This algorithm continuously splits the group of colours into smaller groups by dividing one of the attributes, hue, saturation, or brightness values. Which of the latter is used is decided by whichever has the largest range from smallest to largest values in the colour set. The group of colours is split into two sets depending on whether the colours attribute value is greater or less than the mean of the largest and smallest attribute over the group of colours. This is continuously done until a colour palette of a certain size is found. The set which contains the most colours is chosen as the final colour. The extreme/unwanted values only have small colours in their set and therefore, be eliminated. This can be similar to finding a set of medians for an entire colour set. However, results found that it does not work very well. The colours are not split into the expected final colour. To get the desired colour, it requires splitting the set many, many times and therefore, leading to more and more errors.

4.3.2 K-Means Technique

The next technique uses the K-means clustering algorithm. This algorithm selects K colours from the set as the base means. Colours are separated into a cluster by their closest K mean. A centroid is found for each of these clusters and the process is repeated until a convergence is reached. This algorithm needs to be implemented in such a way that the HSB values are in the same range i.e. 0-1. This is done because the euclidean distance needs to be uniform over the HSB values. The mean value with the most colours in its cluster should be the final colour value. This algorithm again did not work as hoped. The final colours is a mix of light brown to a dark brown.

4.3.3 Weiszfeld Geometric Median with Colour Removal

This algorithm was created just for this project. It takes into account that some colours in the set will be an incorrect value to what is wanted. Therefore, there is need to get rid of these colours. However, it is difficult to do that. It is done in numerous steps.

The first step is to use the Weiszfeld algorithm. This algorithm gets a geometric median of a set of n-d values (in this case 3-d). A geometric median is the point minimising the sum of distances to the sample points. The geometric median is found as it gets a more accurate results than using a centroid. The Weiszfeld algorithm is a form of iteratively re-weighted least squares. The point is continually converged to a single location. Care is given in making sure the ranges for the hue, saturation and brightness value are the same. All the points are set to a 0-1 range initially and reset to their original range. This will prevent any bias in the algorithm.

The set of colours are sorted by the euclidean distance to this geometric median. The last 25% of the colours in the set are removed. The geometric median using the Weiszfeld algorithm is found again and the process is repeated 10 times in total. There is only a little more than 5% of the original colour set remaining. A final geometric median is found and this is our final chosen colour.

```
FUNC selectColour(List<double[]> colours)
    FOR x = 1 to 10
        double[] geometricMed = weiszfeld(colours)
        sort  colours by euclidean distance to geometricMed
        remove last 25% from colours
    END FOR
    double[] finalColour = weiszfeld(colours)
    return finalColour
END FUNC
\caption{The removal of colours for the Weiszfeld Geometric Median with Colour Removal}
```

This method was found to be very successful and the final colour is very much related to the colour that was expected. Therefore, it is used to get the single HSB value for the colour attributes.

For a blue eye the colours found from the colour palette and k-means algorithm are both brown. The Weiszfeld algorithm with colour removal gets a light blue colour. It was slightly affected by the sclera of the eye, However, this would not affect the classification.

4.4 Feature Classification

These features must be classified once the attributes have been extracted. Two techniques are used for the classifier. The first uses Weka, the java open software toolkit. The second uses a Nearest Neighbour over a set of values.

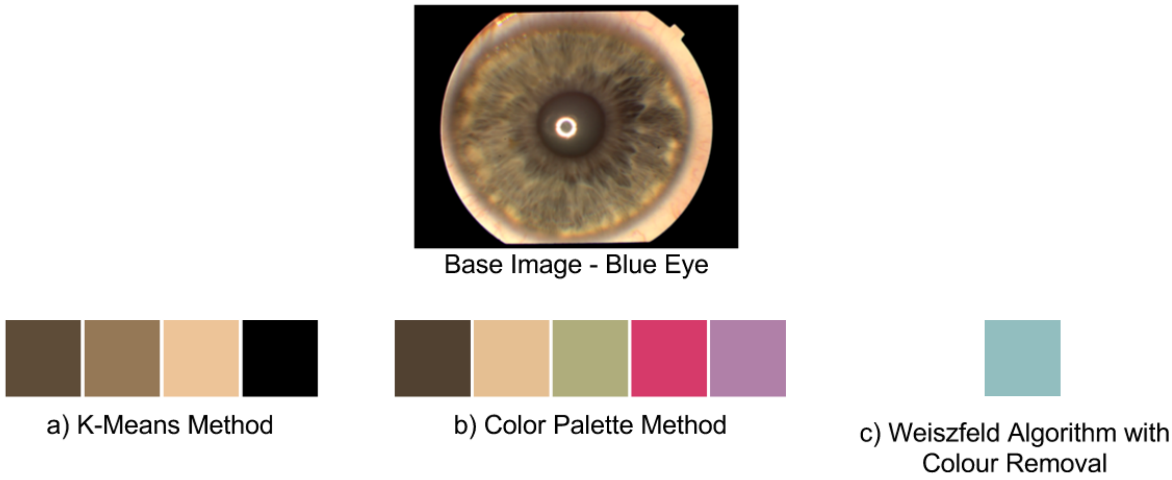


Figure 4.5: Results from the different colour separation techniques. Colour Palette and K-Means have been sorted by the most common colour

4.4.1 Skin Complexion

As it is difficult to accurately classify a large set faces, a machine learning model could not be created. The nearest neighbour technique is used. This method gives the app a set of 19 different HSB colour values found from the base colours of makeup foundation colours. The Skin complexion is classified into one of these foundation codes. The foundation codes and colours are taken from Mac, the cosmetic company. Each of the foundation codes are split into NC or NW for warm and cool faces respectively followed by a shade number. The shade number ranges from 10 to 58. The foundation code is chosen by doing a nearest neighbour from the extracted colour to the set of foundation colours.

A mapping is created so that these foundation codes converted to a name - 'pale', 'fair', 'light', 'medium', 'olive', 'brown', and 'black'. Each of these can be either 'cool' or 'warm'. This is done as it is easier for the user to understand what a 'light warm skin tone' is compared to 'NC25'. This is important as the app is created with new makeup users in mind.

4.4.2 Eye Colour

As it is easy to see if a person has either blue, green or brown eyes from a photograph; a classified dataset could be created. Therefore, the weka machine learning model is used. One hundred faces are classified by a human and their HSB colour value extracted by the program. A datafile is created with all these attribute values and classes. Weka uses this datafile to create a model so that a new user's eye colour can be classified in the app.

The machine learning algorithm for the eye classifier is J48/C4.5. This is found to be the best performing algorithm over different tests. This algorithm works well with the eye colour classification as it is obvious where to split the attributes for blue, green, and brown in the HSB colour.

4.4.3 Hair Colour

Initially, hair colour is found by creating a machine learning model with preclassified user images. The classes are blonde, brown, black, and red. 120 users are classified and values extracted. While results are fairly accurate, it was decided to test another method to see what improvements could be made.

Using the knowledge from how the skin complexion is classified, the same method is attempted for the hair colour to improve the results. The extracted hair colour is compared and classified by the nearest neighbour technique and a set of different HSB colour values. The extracted HSB hair colour is compared to 21 different colours. Hair dye colour values are used to get these 21 shades. These hair dye colours are taken from Garnier, a hair and skin products company.

21 different colours are used in the nearest neighbour algorithm to give more accurate results. However, colours are divided into four groups - blonde, brown, black, and red. The four groups are effective for the makeup recommendation. 21 colours are used in the nearest neighbour algorithm rather than 4 because of the interaction between hair colour and its HSB value. As a user's hair changes from a light blonde to a medium blonde all of the HSB values change. The nearest neighbour incorrectly classifies many hair colours when only 4 main colours are used. Having the 21 colours also allows the app the ability to classify hair into more stylish names i.e. 'Almond creme blonde hair' as opposed to 'blonde hair'.

This new technique is found to be more accurate than the machine learning technique.

4.4.4 Lip Colour

Lips are not classified as they are not essential for the makeup recommendations.

4.4.5 Face Shape

The face shape has to use a machine learning algorithm. 75 faces are classified by a human and attributes extracted for these faces. It is difficult to find faces for this dataset, as many faces have a rotation to them. Any rotation causes the attribute extraction to be incorrect. The faces are found for google images and only good quality 'face on' images are used.

A few classifier algorithms were tested to find the most suitable. It was found that the Naive Bayes algorithm is the best.

4.5 Light Normalisation

Lighting is a major problem in this project as a person's face can change colour by changing environments, turning their heads, or taking the photograph at a different time of day. It was attempted to find a solution to light normalisation which would help the feature colour extraction and classification, and resultantly help the makeup recommendations.

As discussed in background research, there are research papers on techniques to normalise lighting. One paper spoke about normalising skin tone by having the user hold up a sheet of paper with an array of different colours. While this works well; this project was developed to be as simple as possible for the user. Therefore, that technique does not seem appropriate.

This project attempts to test two techniques to normalise lighting - histogram equalisation and white patch normalisation.

4.5.1 Histogram Equalisation

Histogram equalisation increases the global contrast of an image. A histogram equalisation distributes every pixel separated by each component of the colour space. A histogram is created from these pixels and 'flattened' out. The resulting image has all the pixels normalised over the whole image.

As histogram equalisation works separately over each component in a colour space, different colour spaces are tested as each would yield different results. Luminance/brightness in HSB and CIE L*a*b* colour spaces are also tested.

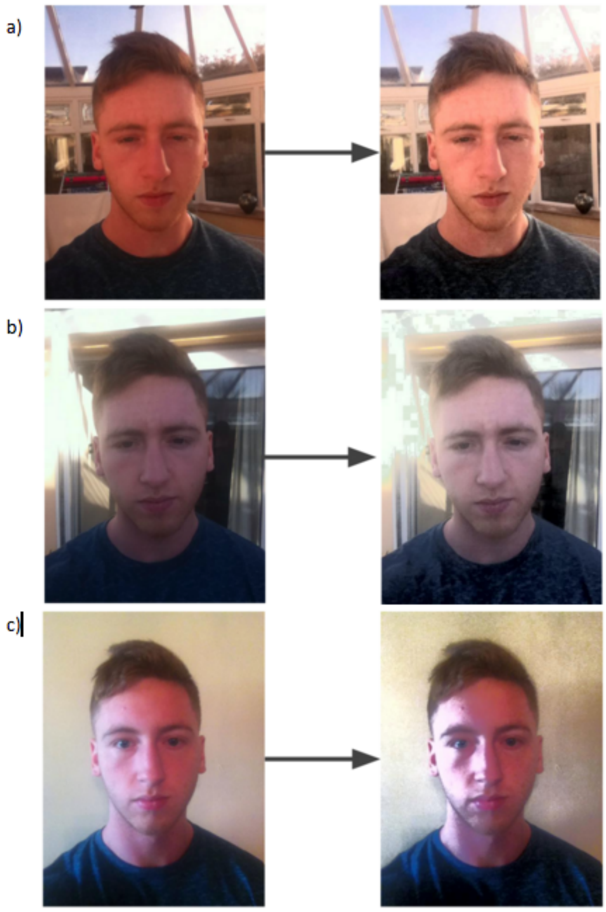


Figure 4.6: Examples of histogram normalisations, where (a) is a reddened face, (b) is a dark face, and (c) is a face with a plain background

It was discovered that the equalisation works well when there is a wide range of colours in the background. For example, a dark face is lightened and the red component in a reddened face is softened. The results were similar over the different colour spaces. CIE L*a*b* tends to work the best.

However, when the background is plain or of a similar brightness the algorithm does not work well. It oversaturated the face. As a user is more than likely to take their photograph indoors, this solution is not suited for the app.

4.5.2 White Patch Normalisation

The White Patch Normalisation takes a point in the photograph which is be of a known colour. The only point where this is the case is on the sclera i.e. the white of the eye. The base sclera colour is found to be slightly off-white.

The sclera is found by getting every pixel inside the eye. The set then removes the pixels that have pure white, as these can be caused by flash or reflections. After these pixels have been removed the brightest pixel was taken as the base colour. A distance is taken from this pixel to the base sclera colour. This distance is added/subtracted to every pixel in the image.

Images of faces under extreme, slightly off and normal lighting were tested. This algorithm appears to work well in the extreme lighting. However, points are over-done. The same occurrs for normal and slightly off lighting. The area around the cheeks was found to be lighter than the sclera and was very much affected. It becomes brighter than sought. Therefore, it was not used for the app.

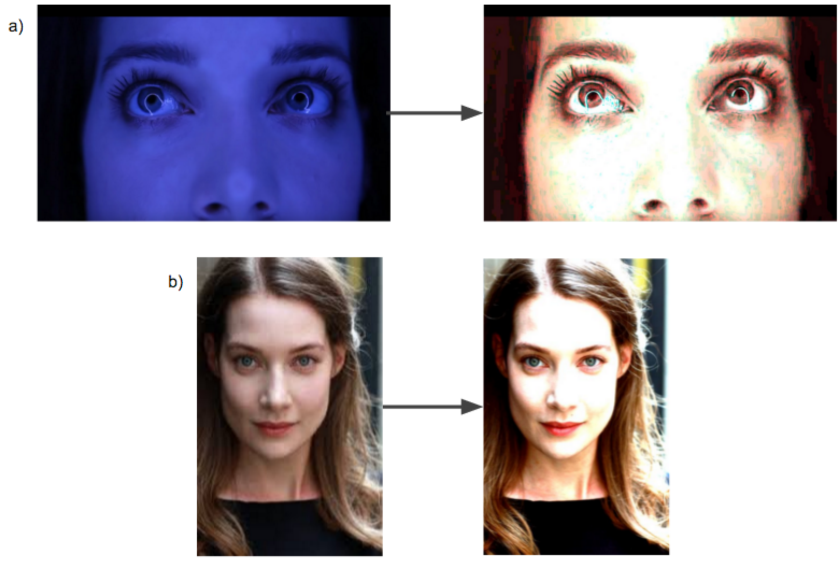


Figure 4.7: Examples of the white patch on sclera results, where (a) was a face in extreme lighting, and (b) is a face is normal lighting

4.6 Makeup Recommender

As stated, there are three of products - face makeup, eye makeup, and lip makeup. Every product is unique in how it is selected. They are recommended to suit the user's features. However, they also selected to combine with other products to give an overall makeup recommendation. There is a makeup database in the app which has over 800 products. This large range of products is necessary and each should have enough metadata so that they can be recommended and all information on the product can be given to the user. The most important metadata for the products in the database is a colour hexcode.

4.6.1 Face Makeup Recommendations

The main feature of the face makeup is the skin complexion. As said in section 3.2.1 there are 5 products - foundation, concealer, bronzer, highlighter, and blusher.

Foundation

As said in section 3.2.1 foundation is the base makeup product. The database for the app has 218 foundations which are subdivided into 9 different types i.e. liquid, powder etc. As previously mentioned each of these foundations has a colour hex code. The recommendation is based on the skin complexion feature of the user. As foundation is supposed to be as close as possible to the user's skin complexion, the app finds the closest foundation using a euclidean distance.

Concealer

The concealer uses the foundation for the recommendation. It should be the same undertone, either warm or cool (if the foundation is as such). It should also try to be a shade lighter than the foundation. The colour hex codes have to be manually adjusted from the ones used in the Mac store. This is because they are not the same as their foundation counterparts. The concealer is found by first checking if the foundation chosen for the user is of a warm or cool undertone. Only the same undertone product remains; the others are removed from the list. The concealers are sorted to the foundation colour making sure to only select concealers that are a shade lower than the foundation.

Bronzer

The Mac store does not have a large selection of bronzers. Due to this, the bronzer product selection could be done using if-else statements. This insured that the system would give great matches.

Highlighter

Mac has a wider selection of highlighters. As there are more products, the highlighter can not be selected the same way as the bronzer. The highlighter attempts to find the best product again by how similar it is to the user's skin and lighter than the foundation.

Blush

This was the most difficult face makeup for the app to choose as only certain blushers are suited for some users skin tones, hair colour and occasions. Skin tone and undertone are the most important features for the blush recommendation. Blushers can be split into 6 groups for the skin tone and undertone - light warm, light cool, tan warm, tan cool, dark warm, and dark cool. While the hair colour is important in blusher selection, the ethnicity of people comes into play. Dark colour skin tone people are more than likely to have dark hair and only lighter skin tones have red hair. Therefore, hair colour does not need to be considered.

The app gets the 52 blushers from the database. Every product is examined to see if it is suited for the user, if it is not then it is removed. Finally, the blusher is selected from the remaining list, where the brighter colours are suited for 'Night out' occasions and 'Daytime' or 'Natural' have the softer products.

To find if a product is suited for a user, machine learning is used. A J48/C4.5 tree was created for each of the skin tones and undertones. The suited blusher colours were classified on whether they are or aren't suited to the skin tone. The J48 tree model could be written as if-statements based on the warmth, hue saturation, and brightness of the skin.

4.6.2 Eye Makeup Recommendation

Eye Makeup consists of eye shadow, eye liner, mascara, and eye primer.

Eye Shadow

Eye shadow is one of the most difficult products to recommend as it can depend on the occasion, face shape, hair colour, skin colour and most importantly eye colour. There are 120 eye shadows in the database. Also numerous eye shadows are required for different styles in a 'Night out', look. Due to this the selection of eye shadow is split between the two occasions.

The 'Daytime' eye shadow is used to slightly enhance the eyes. This means that a shadow close to the skin colour is chosen. It is selected in the same way as the foundation - using the euclidean distance of the different eye shadows to the user's skin colour. However, there is a slight bias for the saturation of the product colour as it gives better results.

The 'Nightout' products selections are more difficult. 4 products are given to the user. The same technique is used as the blusher - take away the incorrectly matching products then select the eye shadows. The app goes through each product and sees if it matches the user, if it is not it is removed. For example if the user has blue eyes, selected eye shadows are removed so that only brown, orange, pinks and reds eye shadows remain. Products are then checked with the hair colours - blondes only having bright saturation of eye shadows and red hair should not have too dark or too bright eye shadows. Finally, face shape is checked with the products so that there is proper blending with the lipstick at a later stage.

What remains in the set are eye shadows that match with the user, but picking 4 random eye shadows does not mean they blend together. A random eye shadow from the list is chosen and all eye shadows that do not have the similar hue are removed. Finally, the remaining eye shadows are sorted by brightness and a random eye shadow is chosen from each of the quarters to get a light, medium-light, medium-dark, and dark eye shadow for the user.

Eye Liner

The eyeliner is selected to match the darkest eye shadow chosen. Again the euclidean distance method is chosen to pick the eyeliner.

Mascara

As already said there are certain mascara's that are suited for the night out look. These products are removed from the selection set for the 'Daytime' looks. All that was needed after this removal was to select a random mascara.

Eye Primer

The primer is chosen as the closest matching to the skin colour. This is to make sure that is does not interfere with the blending of the eye shadow.

4.6.3 Lip Makeup Recommendation

As mentioned in section 3.2.3 the lip makeup consists of lipstick, lip gloss, and lip pencil.

Lipstick

The lipstick 'Daytime' is chosen to be similar to the exact lip colour extracted from the user. It was found that giving slight preference to the saturation in the selection gives better results.

Similar to the blusher, a J48 tree is created to help with the building of this recommendation as well as to improve the accuracy for the 'Night out' lipstick. A number of the 200 lipsticks are classified to match hair colour. Lipsticks that do not suit the eye shadow are dealt with, subsequently. If there is a high saturation for the eye shadow the lipstick should have a low saturation and vice versa. The lipstick is chosen based on the face shape. For example, a heart face shape wants to emphasise a feature away from the cheeks and the lips is best for this - a bright lipstick is chosen in that case.

Lipgloss

Lipgloss is selected to be as similar as possible to the lipstick, using euclidean distance.

Lip Pencil

The same is the lipgloss, lip pencil is as similar as possible to the lipstick and the euclidean distance is used.

Chapter 5: **Implementation Technology**

5.1 Android

Much of the android development is standard. XML files are created for each of the screens and for every adapter that was needed. The sqllite, Android's built in database handler, is used to store all of the products. The camera, internet, and storage functions are needed on the phone and permissions are required in the Manifest.

The main process of the app is on the feature extraction and classification. This is a time consuming task. AsyncTask is created so that the app would not break and a loading screen could be shown. Before the AsyncTask, a fragment is placed on the screen. The fragment has a loading GIF. Glide, an image loading library, is used to display the GIF. During the background process of the AsyncTask the image of the user is sent to the Face++ servers to get the different data on the user. This data is used to extract the attributes on the facial features and classifiy them using Weka. Once this process is completed the fragment is removed and the features are displayed to the user.

RecyclerView is the best way to handle lists on Android. Using these in the Face Profile activity and Recommendation activity becomes a complicated process. A double RecyclerView is used for each. The Face Profile uses a RecyclerView for each of the features. As there can be some errors in classifying features, the user is allowed to reselect the correct feature. An adapter is used to create the RecyclerView. The adapter changes the image and name for each feature in the list. It also adds an OnClickListener to each feature item. Once clicked, another RecyclerView is displayed under the given feature. This RecyclerView is a horizontal list of all the other classes of the feature. If the user clicks on one of these 'reselect' items, the feature and the UI are updated.

The Recommendation Activity uses a RecyclerView for each of the cards on the screen. This is a horizontal scrolling list. These cards cover almost the entire screen. The scroll function is made to ensure that a card is always centered in the screen. There is an inner RecyclerView for the list of products on each of the cards. Every name, title, description etc. is changed, based on the contents of the products. An AsyncTask is also used to get the images from the URL of the product.

5.2 Product Database

The database is taken from the Mac online website. Mac has approximately 1000 products. However, some of these are inapplicable i.e. brushes. Therefore, about 800 products are taken from their store for this app. Every single product has metadata on:

- The type of product
- The name of the product
- The colour hex code of the product
- The description of the product

- The url to the exact product
- The url to an image of the product

There is no online resource for all the products with the metadata above and therefore, they have to be taken from their website. Going through 800 products and copying and pasting in 6 different data values is a gruelling task. As a result the html of the web pages containing each product was downloaded using 'wget'. The html is parsed so that all the information is found. Finally, the database was checked to make sure all the metadata is correct and a .txt file is created from it.

The app uses a Sqllite database to store all of the products. The database is created once the app is opened. It is filled by parsing the .txt file and adding each line as a new product to the Sqllite database. There are columns for each of the metadata info, as well as an id. A method is added, using a SQL query, so that a list of products could be returned from the makeup type.

5.3 Face++

As said in section 5.1, the detection from Face++ is done during the AsyncTask. The image is required to be in a byte array for the http request.

Face++ has open software to allow for the server request and JSON handling. However, a few aspects of their source code needed to be changed. It uses Apache's httpmime and httpcore, for the server requests. However, this was done using deprecated methods and this cause some issues. A newer technique using MultipartEntityBuilder, is created in their place. Methods had to also be added so that the server and timeout duration could be changed.

5.4 Weka

Using Weka was quite easy in this project. Their toolkit allow for the machine learning classification algorithms to be used with little effort. A text file containing all of the preclassified data for a feature, as well as, the chosen classification algorithm is given to weka. A model is then created for the feature. Attributes could be given to the model and the feature class found. There is worry if creating the model inside the app would lead to too much of the phone processing power to be used. This turned out to not be the case.

5.5 Apache

Apache is used to for the server requests to the Face++ server. However, Apache's Commons Math library is also used for the face shape extraction. A curve fitting algorithm is needed to make the 2 degree and 4 degree polynomial functions. The PolynomialCurveFitter is used to create these curves. The library also contains classes to find the integral of the a curve. The TrapezoidIntegrator class, which is based on the Trapezoidal Rule, is chosen to find the integral.

Chapter 6: **Development Process**

The timeline and challenges for the development of the project are given below.

6.1 Timeline

The first step was to learn the basics of the Android environment. A Java developer can find it difficult developing on Android. Even the process of passing data and objects between activities is difficult. Many aspects in this area had to be learned on an ongoing basis.

A simple face detection app is created in order to learn how to use Face++. There is very little information on how to use Face++ which made it even more challenging. The problems discussed in section 5.3 were found and fixed at this stage. Results from the Face++ detection were examined here. It was originally thought that Face++ could detect the skin colour, eye colour and hair colour. However, these results were further research proved this to be false. This resulted in a great deal more of extra work and research having to be done. The full effects of lighting on an image was also understood at this point.

A solution to this lighting had to be found. The solution was to allow the user to change to change the feature classes if they were incorrect.

The lighting problem proved to be very challenging and frustrating. To ensure there was continued progression in the project, the face shape extraction and classification was done in tandem.

The next area was to extract the values for the other features. The classification was made at the same time as it could be used for testing the feature extraction. The first feature attempted was the eye colour. The first algorithm attempt was the extraction of all the colours inside the interquartile Face++ eye points. The discovery that blending colours results in a brown or grey colour was made at this point.

The first plan was to try and get rid of all the unnecessary colours. It was thought that a better extraction method would do this. The iris detection method from section 4.2.2 was attempted. As discussed the results were fair. The hair colour was attempted next.

The two methods from section 4.2.3 were created but again the colour blending problem occurred.A colour selection algorithm had to be made. The colour selection algorithm was created and tested until a solution was found.

With the colour section algorithm, the eye colour was found using the old method with the face++ points. The hair colour could also be found. The next was the skin colour. The skin colour extraction was made as stated in section 4.2.1

The app was built as explained, again learning as the build when on. Once the development of the app got to the recommendation screen, the recommendation algorithms were made.

The app was built and tested on different users to understand the user functionality. It was also tested on different faces to see if the recommendations could be improved.A few things were altered such as making sure that the lipstick and eye shadow were dependent on each other.

6.2 Challenges

There were many areas that caused problems but the image dataset, face shape, colour separation, and lighting proved to be the most challenging.

6.2.1 Image dataset

One difficulty with this project lay in the fact that it required a large dataset of face images. There are some face datasets available. However, they are mainly created for facial recognition. This means that they were not suited to this project because of poor quality and/or orientation and location of the face. Some available datasets need to be purchased at a large cost and therefore, out of reach for small research projects. The solution to this was trawling through Google Image Search for suitable faces.

6.2.2 Face Shape

Through research, it was found that a face shape can be classified by a width over height ratio and whether the jaw line is pointy, round, or square. It was hoped that the coefficients of the curve from the curve fitting algorithm would each have a clear distinction for the three jaw lines. However, as the user moves in any direction the coefficients drastically changed. The best solution to this was getting all the attributes that were mentioned in section 4.2.5. If there was any attribute that could help in the detection of the face shape it was added.

6.2.3 Colour Separation

Colour separation was vital for this project. There didn't seem to be a clear algorithm from research that would work. The only way to test the different methods was to create them from scratch. It was time consuming and lead to failure on two occasions.

6.2.4 Lighting

The problem with lighting proved to far more challenging than originally thought. The first idea was to allow the user to change the lighting or colour feature classifications themselves. Another idea was the sheet of paper method discussed in the paper [19]. Both of these required more work from the user than wanted. It was hoped that the user would only have to take the photo. The two other methods, discussed in section 4.5 were tried and tested. It wasn't possible to get them to work in all environments. The design of the app had to be changed and get the user to have more input than originally desired.

Chapter 7: **Results / Evaluation**

7.1 Evaluation of the App

The app is fully functional, as described in all the previous chapters. The mandatory of this project stated that the colour based recommendation were complexion, hair colour, and lip colour. The makeup recommendations use these, but eye colour class was also added.

Again the mandatory stated only a textual list of the recommendations was needed. However, in the final version the app presented withthe user the makeup products in an informative, clear and more functional manner (see figure 3.2). Testing the app with different people proved this.

It was attempted to fix the lighting problem for the app. However, this proved unsuccessful and the final solution, which was to allow the users to change their feature classification, can cause confusion.

7.2 Feature Accuracy

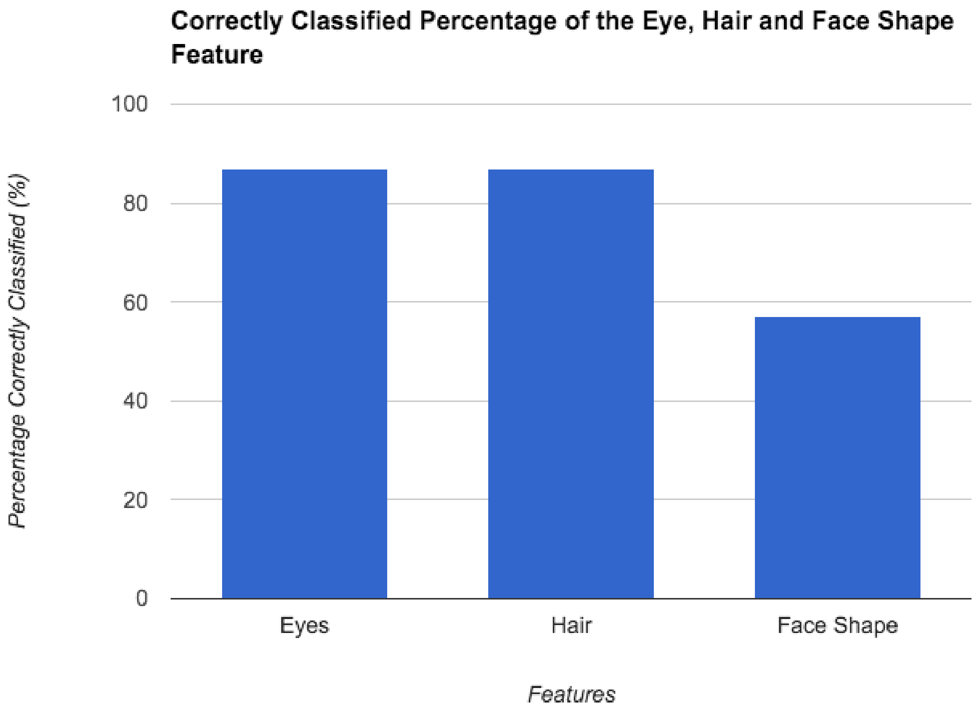


Figure 7.1: Results showing the correctly classified percentage of the eyes, hair, and face shape

The graph shows the percentage of correctly classified features compared to a human classification. 100 faces were classified by a human for each of the features. The same faces were classified by the app. The app was deemed correct if it gave the same result for a face as the human.

The eyes were found to give the best results with the J48 tree algorithm in Weka. Over the 100 faces it was 87% correctly classified.

The hair was classified using the nearest neighbour algorithm as discussed. The results shown the correctly classified percentage into the 4 classes, blonde, brown, black and red. Using Weka for the classification lead to an accuracy of 70%. This increase to 87% is obviously a big jump. It showed that it was very much worth trying the new method.

The face shape was found to be 57% accurate with NaiveBayes. NaiveBayes was the most accurate classification model. As an example, J48 tree had a correctly classified percentage of 37%. While this is a significant improvement on a random guess for the 5 classes, it was hoped to get better results.

Skin complexion was unable to be compared in the same way, as there are many classes for it. As skin complexion is converted to a foundation code, the results in figure **??** for foundation can show it's accuracy.

7.3 Makeup Recommendation Accuracy

The app gives different recommendations each time. There is also a large range of products that are suited to a user. This made it very difficult to represent the accuracy. 20 faces were used for the evaluation. There was a broad range of faces, to give a more accurate representation of the results. The faces used were also in optimal lighting.

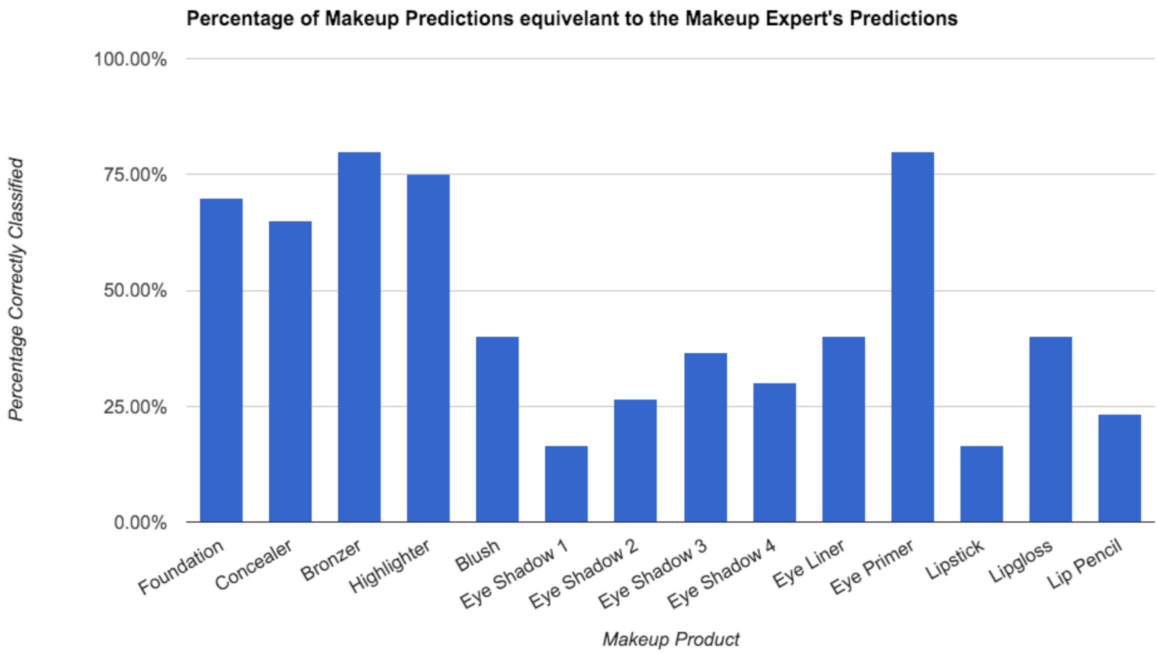


Figure 7.2: Percentage of Makeup Recommendations equivalent to the Makeup Expert's Recommendations

This is the first graph which shows the accuracy of the makeup recommendations. A makeup expert gave their recommendations for 20 different faces. The app gave 30 recommendations for each of the 20 faces. A comparison was made between both. The result was calculated for a product by finding the percentage of time the product was equivalent to the expert's over the 30 recommendations. This value was averaged over the 20 faces.

An equivalent recommendation for foundation meant that the same foundation code were from both, the app and the expert. This was also the case for concealer. An equivalent for the rest of the products meant the exact product was chosen by both.

The foundation, concealer, bronzer, highlighter, and eye primer were similar in results and higher than the other products. The reason for this is that they are all based on the skin complexion and if that was correct they would all be correct. They were more accurate in this graph as there is an exact product suited for a user here.

There is a large set of products that can be suited for the user in the other results. Because the percentage is not high does not mean that the recommendations were incorrect.

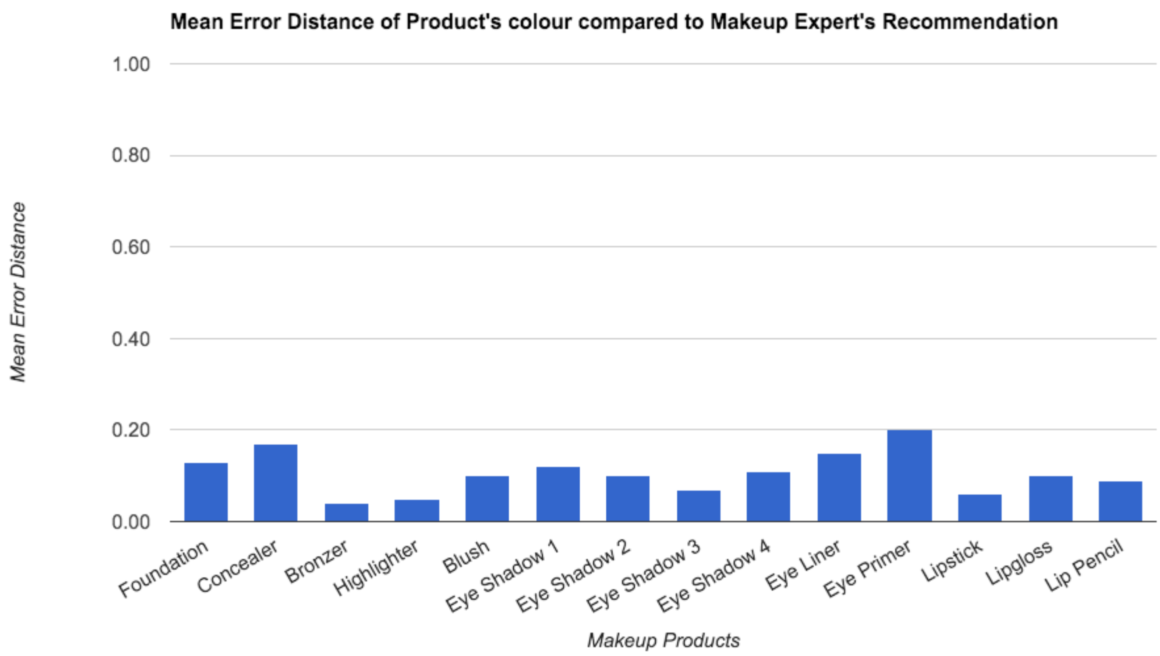


Figure 7.3: Mean Error Distance of Product's colour compared to Makeup Expert's Recommedations

This is another graph to show the accuracy of the app's recommendations. The colour value of every recommendation was found. A euclidean distance was found between each of app's recommendations and the expert's recommendations. The resultant graph value was this distance over the largest possible distance between two products. As an example the largest distance for foundation is from NC10 to NW58.

These fractions were averaged over the 30 of the app's recommendations and then averaged again over the 20 faces. The lower the value the closer the app's products were to the expert's. This shows a clearer representation of the products such as eye shadow.

Foundation, concealer and eye primer are noticeably higher than the other results. This was normally due to the warmth of the face being incorrect for a few users. There is a large euclidean distance between warm and cool skin and incorrectly classifying them, affected the means.

Chapter 8: **Conclusion**

The cosmetic industry could be more advanced in the area of advanced technology. It is a multi-billion dollar business, yet there are few technological systems available to the consumer which would help in their makeup choice. This project was designed to advance this process.

This project started mainly as a design and implementation project. However, as it developed it also had major research components. Research had to be put into the areas of face shape, makeup recommendation, light normalisation, colour extraction and separation.

The eye colour and hair colour results were excellent for the dataset. The main reason for this was the work put into the colour separation techniques. A more developed algorithm had to be created as previous ideas yielded poor results. It was difficult to evaluate the makeup recommendation. However, results shown and discussion with a makeup expert show the accuracy of the recommendations.

In the current form this app does not have a solution to the light normalisation problem. This greatly affects the feature extraction. General normalisation techniques were found not to be suited. More work is either required to adapt them or a whole new method to solve this problem needs to be developed. The fix for this problem in the app is to allow the user to reselect the features' classification.

Face shape is a very difficult area to classify. Many attributes were created for the face shape extraction and it had an accuracy of 57%. A whole new method of classification might have to be attempted for this problem to be solved.

The research in this area showed that this project could be further developed for commercial use or aspects of it adapted to further research in other areas.

The app could be expanded to have a larger database of products. Currently there are 800 products alone from Mac. Some people may prefer another makeup company or find that the cost of the products is too high. With a larger database of products, a filter could be added for price or the company. Other filters could be added for type of skin e.g. oily or dry skin, and type of finish e.g. matte or satin finish.

More information could be given to the user such as a step by step guide on how to apply the makeup to get the suggested style of the makeup recommendation.

Some other final features to the app could be further developed. Recommendation based on eye brows, such as eyebrow pencils or how best to shape the eyebrows could be added. Some hair styles are more suited to certain face shapes and makeup products, and these could be suggested to the user. Finally, more work could be done to involve the colour wheel in the makeup recommendations e.g. antagonistic colours for eyeshadow.

The area of face shape has further possibilities for development in the fashion world. The shape of face for example could dictate the choice of glasses, hair styles, and possibly even clothes.

Finally this app could also be developed into a major marketing idea for makeup companies. It is important for them to show that they are innovative and looking for new ideas to advance their products. A recommendation system containing only their products could attract new users.

Bibliography

[1] Nahla Ben Amor, Salem Benferhat, and Zied Elouedi. Naive bayes vs decision trees in intrusion detection systems. In *Proceedings of the 2004 ACM symposium on Applied computing*, pages 420–424. ACM, 2004.

[2] Kevyn Aucoin. *Face Forward*. Little, Brown and Company, 2000.

[3] Nina Bhatti, Harlyn Baker, Joanna Marguier, Jérôme Berclaz, and Sabine Süsstrunk. Cell phones as imaging sensors. In *SPIE Defense, Security, and Sensing*, pages 770802–770802. International Society for Optics and Photonics, 2010.

[4] Christopher M Bishop. Pattern recognition. *Machine Learning*, 2006.

[5] Bobbi Brown. *Bobbi Brown Makeup Manual: For Everyone from Beginner to Pro*. Grand Central Life & Style, 2008.

[6] Cheng-Chun Chang, Yung-Chi Chuang, Chien-Ta Wu, Byung Il Choi, Kwansik Lee, Seongsu Woo, Saifullah Rao, and Jihoon Kim. A low-cost mobile device for skin tone measurement using filter array spectrum sensor. In *SENSORS, 2014 IEEE*, pages 499–502. IEEE, 2014.

[7] Abbas Cheddad, Joan Condell, Kevin Curran, and Paul Mc Kevitt. A new colour space for skin tone detection. In *Image Processing (ICIP), 2009 16th IEEE International Conference on*, pages 497–500. IEEE, 2009.

[8] Oristep Consulting. Global cosmetics market - by product type, ingredient, geography, and vendors - market size, demand forecasts, industry trends and updates, supplier market shares 2014- 2020. *Research and Markets*, June 2015.

[9] Kai-Bo Duan and S Sathiya Keerthi. Which is the best multiclass svm method? an empirical study. In *Multiple classifier systems*, pages 278–285. Springer, 2005.

[10] Haoqiang Fan, Zhimin Cao, Yuning Jiang, Qi Yin, and Chinchilla Doudou. Learning deep face representation. *arXiv preprint arXiv:1403.2802*, 2014.

[11] Camelia Florea, Mihaela Moldovan, Mihaela Gordan, Aurel Vlaicu, and Radu Orghidan. Eye color classification for makeup improvement. In *Computer Science and Information Systems (FedCSIS), 2012 Federated Conference on*, pages 55–62. IEEE, 2012.

[12] Ramy Gafni. *How to Fake Real Beauty*. Sydney: Running Press, 2015.

[13] Google. Android developers. `http://developer.android.com`. Accessed: 2015-4-02.

[14] Dong Guo and Terence Sim. Digital face makeup by example. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 73–79. IEEE, 2009.

[15] M. Harville, H. Baker, N. Bhatti, and S. Susstrunk. Method and system for recommending a product based upon skin color estimated from an image, March 15 2007. US Patent App. 11/233,599.

[16] ColorModules INC. Plum perfect. `https://www.plumperfect.com`. Accessed: 2015-11-24.

[17] Leyuan Liu, Nong Sang, Saiyong Yang, and Rui Huang. Real-time skin color detection under rapidly changing illumination conditions. *Consumer Electronics, IEEE Transactions on*, 57(3):1295–1302, 2011.

[18] L'Oreal. Make up genius. `http://www.makeupgenius.com`. Accessed: 2015-12-01.

[19] Joanna Marguier, Nina Bhatti, Harlyn Baker, Michael Harville, and Sabine Süsstrunk. Assessing human skin color from uncalibrated images. *International Journal of Imaging Systems and Technology*, 17(3):143–151, 2007.

[20] S.W. Na. Recommendation system based on the recognition of a face and style, and method thereof, May 23 2013. US Patent App. 13/813,003.

[21] Pantone. Capsure. `https://www.pantone.com/capsure`. Accessed: 2015-11-30.

[22] Tommy Beauty Pro. Makeup 101: Color theory and artistry. `https://tommybeautypro.wordpress.com/2013/01/05/makeup-101-colour-theory-make-up-artistry`. Accessed: 2015-11-27.

[23] Rajeev Ramanath, Wesley E Snyder, Youngjun Yoo, and Mark S Drew. Color image processing pipeline. *Signal Processing Magazine, IEEE*, 22(1):34–43, 2005.

[24] Alessandro Rizzi, Carlo Gatta, and Daniele Marini. Color correction between gray world and white patch. In *Electronic Imaging 2002*, pages 367–375. International Society for Optics and Photonics, 2002.

[25] G. Rubinstenn, D. Giacchetti, and F. Pruche. Beauty advisory system and method, April 3 2003. US Patent App. 10/024,496.

[26] Sephora. Color iq. `http://www.sephora.com/color-iq`. Accessed: 2015-11-29.

[27] Yaniv Taigman, Ming Yang, Marc'Aurelio Ranzato, and Lior Wolf. Deepface: Closing the gap to human-level performance in face verification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1701–1708, 2014.

[28] Xiao-Ning Zhang, Jue Jiang, Zhi-Hu Liang, and Chun-Liang Liu. Skin color enhancement based on favorite skin color in hsv color space. *Consumer Electronics, IEEE Transactions on*, 56(3):1789–1793, 2010.

[29] Erjin Zhou, Zhimin Cao, and Qi Yin. Naive-deep face recognition: Touching the limit of lfw benchmark or not? *arXiv preprint arXiv:1501.04690*, 2015.