AUTOMATIC AGE PROGRESSION AND ESTIMATION FROM FACES

Ali Maina BUKAR

A Thesis Submitted for the Degree of

Doctor of Philosophy

School of Media Design and Technology

Faculty of Engineering and Informatics

University of Bradford

October 2017

Abstract

Recently, automatic age progression has gained popularity due to its numerous applications. Among these is the frequent search for missing people, in the UK alone up to 300,000 people are reported missing every year. Although many algorithms have been proposed, most of the methods are affected by image noise, illumination variations, and facial expressions. Furthermore, most of the algorithms use a pattern caricaturing approach which infers ages by manipulating the target image and a template face formed by averaging faces at the intended age. To this end, this thesis investigates the problem with a view to tackling the most prominent issues associated with the existing algorithms. Initially using active appearance models (AAM), facial features are extracted and mapped to people's ages, afterward a formula is derived which allows the convenient generation of age progressed images irrespective of whether the intended age exists in the training database or not. In order to handle image noise as well as varying facial expressions, a nonlinear appearance model called kernel appearance model (KAM) is derived. To illustrate the real application of automatic age progression, both AAM and KAM based algorithms are then used to synthesise faces of two popular long missing British and Irish kids; Ben Needham and Mary Boyle. However, both statistical techniques exhibit image rendering artefacts such as low-resolution output and the generation of inconsistent skin tone. To circumvent this problem, a hybrid texture enhancement pipeline is developed. To further ensure that the progressed images preserve people's identities while at the same time attaining the intended age, rigorous human and machine based tests are conducted; part of this tests resulted to the development of a robust age estimation algorithm. Eventually, the results of the rigorous assessment reveal that the hybrid technique is able to handle all existing problems of age progression with minimal error.

Declaration

I hereby declare that except where specific reference is made to the work of others, the contents of this dissertation are original and have not been submitted in whole or in part for consideration for any other degree or qualification in this, or any other university. This dissertation is my own work and contains nothing which is the outcome of work done in collaboration with others, except as specified in the text and Acknowledgments.

The main part of this is based on seven peer reviewed papers published or accepted for publication in different academic journals, book chapters and conferences.

Journal articles

1. A. M. Bukar and H. Ugail, “Facial Age Synthesis using Sparse Partial Least Squares (The Case of Ben Needham),” J. Forensic Sci., 2017.

2. A. M. Bukar, H. Ugail, and D. Connah, “Automatic age and gender classification using supervised appearance model,” J. Electron. Imaging, vol. 25, no. 6, pp. 1–11, 2016.

3. A. M. Bukar and H. Ugail, “Automatic Age Estimation from Facial Profile View,” IET Comput. Vis., 2017.

Book chapters

1. A. M. Bukar, H. Ugail, and N. Hussain, “On facial age progression based on modified active appearance models with face texture,” in Advances in Computational Intelligence Systems, Springer, 2017, pp. 465–479.

2. A. M. Bukar and H. Ugail, “A Nonlinear Appearance Model for Age Progression,” in Soft Computing and Machine Learning in Image Processing, Springer Berlin Heidelberg, 2017.

Conference papers

1. A. M. Bukar, H. Ugail, and D. Connah, “Individualised model of facial age synthesis based on constrained regression,” in Image Processing Theory, Tools and Applications (IPTA), 2015 International Conference on, 2015, pp. 285–290.
2. A. M. Bukar and H. Ugail, “ConvNet Features for Age Estimation,” in 11th International Conference on Computer Graphics, Visualization, Computer Vision and Image Processing, 2017.

Ali Maina Bukar
October 2017

Acknowledgments

All praise be to Lord of the Worlds the exalted and most high.

I will like to thank my supervisor Prof. Hassan Ugail for his dedicated supervision and insightful input to the research over the past four years. Thank you also for agreeing to fund several International conferences. Thank you once again for paying my fourth year (writing-up) tuition fees.

A special thanks to Dr David Connah from whom I learnt a lot during the first two years of my PhD. I also appreciate the advices of Dr Tao Wan and Professor Rami Qahwaji.

My fellow PhD colleagues especially Nosheen Hussain, Zahra Sayed, Shelina Jilani, Bashir Mohammed Halima Abdullahi, and many more who I can't mention in this little space that directly or indirectly helped with ideas, proofread my works and/or supported me financially.

A special thank you to National Information Technology Development Agency of Nigeria (NITDA), for sponsoring the first 3 sessions of my PhD.

Above all, I would like to thank all the members of my family for their continuous support and encouragement.

Contents

List of Figures

List of Tables

List of Acronyms

AAM	Active Appearance Model
BIF	Biological-Inspired Features
CNN	Convolutional Neural Network
ConvNet	Convolutional Neural Network
CS	Cumulative Score
ED	Euclidean Distance
FGNET-AD	Face and Gesture Recognition Network Ageing Database
GAP	Generalised Procrustes Analysis
IMED	Image Euclidean Distance
KAM	Kernel Appearance Model
KPCA	Kernel Principal Component Analysis
LBP	Local Binary Patterns
MAE	Mean Absolute Error
NN	Neural Network
OLS	Ordinary Least Squares
PCA	Principal Component Analysis
PLS	Partial Least Squares
QF	Quadratic Function
RMS	Root Mean Square
sAM	Supervised Appearance Model
SGD	Stochastic Gradient Descent
sPLS	Sparse Partial Least Squares

1 Introduction

1.1 Background

The human face is like a window to the soul, carrying a vast amount of information which we humans have a remarkable ability to extract, identify and interpret. It is no surprise that faces are used as cues for recognising identities [1], emotions [2], gender [3], kinship [4], ethnicity [5] and developmental disorders [6] to mention but a few. It has been well documented that, facial structures and appearances change considerably as people age, which can make recognising individuals difficult over long periods. In particular, the shape of the face changes substantially from birth to adulthood. During adulthood, while the shape remains relatively constant, there are changes in musculature and skin tautness which affect the facial 'texture' [7]. As a result of these observed changes, automatic facial ageing has been studied for over a decade.

Research on facial ageing focuses on two main subjects; age progression and estimation. Age progression entails the re-rendering of the face image with natural ageing effect. Its most significant applications include the search for missing people and the identification of fugitives. Furthermore, facial image correction via age progression can be used to enhance face recognition algorithms. Age estimation on the other hand automatically labels specific age or age group of individuals from their facial image [7]. Age estimation can be used in the automatic retrieval of images. It can also be deployed as a filter for searching through a face recognition engine. Its other areas of application include access control and human-computer interaction.

Similar to other branches of automatic facial analysis, age progression and estimation are obstructed by a number of factors such as facial expressions,

illumination variation and pose variation to mention but a few. Thus, several techniques have been documented in the literature to circumvent these problems, however, the problem is still not considered solved [8]. This is due to lack of standardized age progression performance evaluation metric. Furthermore, researchers use different datasets thus obstructing replication of experiments and comprehensive comparison.

In light of these adverse factors, this work approaches the problems of age estimation and synthesis by building upon existing methods, proposing improvements and developing novel algorithms. Specifically, in this thesis, models of age progression that are robust to insufficient training data, image noise, illumination variation, the negative effects of facial expressions and poor image resolution are built. To evaluate the performance of the proposed models, advanced face representation techniques for age estimation are investigated and deployed. After rigorous experimental evaluation of the age progression models, best performing techniques are deployed in an attempt to solve real life problem of identifying missing persons.

In summary, work done in this thesis starts off with facial age synthesis using Active Appearance Models (AAMs), then due to obvious problems associated with the linear model, a nonlinear appearance model termed Kernel Appearance Model (KAM) is proposed and used for synthesis. To achieve further improvements, a nonparametric procedure is introduced and incorporated to the KAM. The latter part of this thesis explores the development of an automatic age estimation algorithm that can be used to evaluate the performance of the proposed synthesis algorithms. Precisely, two age estimation approaches are considered in succession; age estimation via supervised appearance models (sAM), and then using ConvNets.

It is worth mentioning that, this report interpolates ideas from six articles published by the author. Chapter 3 uses material from [9] and [10]. Chapter 4 is based on [11], chapter 5 emanates from [12], chapter 6 is based on [13] and finally, chapter 7 is built on [14].

1.2 Thesis Statement

This thesis addresses the research questions outlined in Table 1.1.

Table 1.1: Research Questions.

Research Question	Motivation
1. Is there an effective way of achieving automatic age progression using statistical models?	Lanitis et al. [15] demonstrated that an ageing function can be defined that relates ages to face parameters retrieved using AAMs. However, this was not implemented explicitly.
2. Can nonlinearity be used to tackle noise that hinders the performance of AAMs? 3. Can age progression that is robust to facial expression be achieved?	In a comprehensive review, Gao et al. [16] discuss the factors that affect the performance of AAMs amongst which is noise in the data. Furthermore, preliminary experiments conducted in this thesis also show that noise and facial expressions affect the reconstruction ability of AAMs. When searching for missing people, the image at hand can be noisy and be displaying facial expression. An ideal

Research Question	Motivation
	algorithm should render images that are robust to these factors.
4. Is there a way of enhancing the texture quality of age progressed images?	Statistical models produce faded unrealistic images [17]. Low resolution faded images lack fine grained texture details such as wrinkles that act as age indicators. Is there a way of augmenting facial texture?
5. Is it possible to develop an automatic age estimator that is able to predict ages with minimal errors, despite relatively small training data size?	Ideally, age progression algorithm should exhibit two capabilities: preserve the identity of the subject and render the expected age. The former can be evaluated by measuring image similarities (i.e. between real and synthesised images). On the other hand, a non-subjective way of assessing an algorithm's ability to render faces that meet the target age is through the use of automatic age estimation. Can an effective age estimator for evaluating age progression performance be built?

1.3 Thesis Contribution

The contributions made to the field of knowledge are two fold; those related to age progression and others associated with automatic of age estimation.

- Age Progression
    - The classical work of Lanitis et al. [15] is improved by deriving a linear formula for computing facial attributes, thereby giving the ability to synthesise faces even when the training data is insufficient.
    - A novel nonlinear model for face abstraction is developed, this is termed Kernel Appearance Model (KAM). It is further shown that the KAM is robust to noise, illumination and facial expressions.
    - Finally, a non-parametric framework for image texture enhancement is proposed.

- Age Estimation
    - An appearance model that preserves facial features that retain the most significant ageing information is proposed, the model is termed a Supervised Appearance Model (sAM).
    - Using a pre-trained convolutional neural network (ConvNet/CNN), for face representation is also investigated. Hence, based on these features a robust age estimation algorithm that outperform state-of-the-art algorithms is developed.

1.4 Outline of Thesis

Chapter 2 reviews existing works on automatic age progression and estimation, concentrating on those closely related to this work.

Chapter 3 describes an initial work on age progression. Firstly, improvement to the work of Lanitis et al. [15] is proposed. Thus AAM features coupled with ordinary least squares regression algorithm are utilised for rendering images at different ages. Thereafter, the method is extended by introducing more sophisticated regression models. It is observed that the better the regression technique, the lesser is the face reconstruction error. Using the algorithm the best performing algorithm, the progression model is then used in a real life problem, i.e. to synthesise the face of Ben Needham [18].

Chapter 4 introduces a kernel appearance model (KAM) which captures nonlinear shape and texture variations. Facial features extracted using the KAM are then used to synthesise faces. It is observed that KAM's nonlinear transformation effectively handles noise and facial expression variations. The KAM age progressor is then used to synthesise the face of Mary Boyle [19].

Chapter 5 entails the development of texture enhancement pipeline. In order to tackle the problem of low resolution, the chapter details a procedure for augmenting age progressed image output with a fine-grained skin-texture detail.

Chapter 6 reports the work done on age estimation. A supervised appearance model (sAM) is derived and used to capture facial ageing features. Next, age estimation is performed via regression.

Chapter 7 introduces an alternative technique for age estimation. To enhance the performance of the age estimator discussed in chapter 6, CNNs are investigated. Precisely, a pre-trained ConvNet is used to extract features, thereafter, age estimation is conducted via a regression model. Next, thorough

performance evaluation of the age progression models is conducted using the age estimator.

Chapter 8 presents conclusion and future direction of this work.

2 Literature Review

In this chapter existing approaches to age progression is reviewed. Our primary focus is on statistical-model-based approaches, next to the reviews on existing works on age estimation.

2.1 Age Progression

Age progression also called age synthesis, involves the automatic reconstruction of a human face with natural ageing effects [7]. This area of automatic facial analysis has been active due to its real-life applications, which include identification of fugitives and the search for missing people.

The earliest method used for age progression is the forensic artist's approach. Here the subject's image, in combination with images of his/her relatives, as well as additional information such as life style, is used to render the picture as an artistic hand sketch. Alternatively, a computer based graphic drawing approach guided by the knowledge of the forensic artist can be applied [20]. Police departments around the world, still predominantly use the former. While the method has been successful in the past, it requires remarkable talent and years of experience. Normally the forensic artist undergoes thorough training and requires a good knowledge of interviewing procedures, behavioural science, cognitive psychology and craniofacial anthropometry [20].

In Computer Vision, automatic age progression has been approached through the use of geometric, texture-specific and appearance based methods [7], [8].

2.1.1 Geometric Models

Studies on human perception have shown that geometric transformations of the human head, in other words, changes in the shape of the human skull,

significantly affect how facial age is perceived [21]. Pittenger and Shaw [22] studied the perception of facial ageing by using spatial and coordinate transformations to build a facial profile growth model. Particularly, they compared the effects of affine shear and cardioidal strain transformations on the perception of ageing on profile face images. It was discovered that shear had less effect on the overall shape as well as the perceived age and on the other hand, the cardioidal strain had more significant effect on the perceived age. This concept was then modified by Todd et al. [23], by assuming the structure of the head conforms to hydrostatic pressure gradient, thereby giving rise to a revised cardioidal growth model that affects the size of the human head in a manner that is more in line with the effects of actual growth. Subsequently, 3D facial growth models were developed by extending the revised cardioidal strain model [24].

In Computer Vision, geometric models represent the face shape using ratios and geometric units. The face is animated using interpolation and displacement of vertices. D'arcy Thompson is considered one of the pioneers in the area of geometric based face modelling. The theory of transformation described in his book "on growth and form" [25] states that differences in related species can be represented geometrically [26]. Geometric face models have been used in caricaturing [27] and cartoon faces [28]. Furthermore, they have been used by a number of researchers to model variations in young faces, for example, the works of [29] and [30]. An obvious setback of this approach is the fact that it does not take into account the facial texture such as skin tautness as well as wrinkles [7].

2.1.2 Texture Models

Texture related models focus on extracting and manipulating fine grained face details such as the facial skin, creases, and wrinkles to render photorealistic aged-faces [7], [31]. This approach has been explored using various techniques such as the transfer of wrinkles from old faces onto young faces [32]; by substituting high frequency components of the first image with those of a second image. Mukaida & Ando [33] proposed a method that utilised adaptive thresholding to extract wrinkles from facial images, the resulting binary features were then used to independently progress ages. The construction of 3D wrinkles has also been reported in the literature [34]–[36]. However, wrinkles, creases, and other skin deformations are not found in young people. Hence, this approach is not suitable for age reversing.

2.1.3 Appearance Based Approach

Appearance based techniques use both shape and texture information to model the face. One of the earliest attempts is that of Burt and Perette [37] who used facial composites to simulate ageing. Precisely, their approach entails computing averages of face shapes and colour information for different age groups. Subsequently, the difference between the target age group and the current age group is computed, scaled and added to the subject's image. Over the years Burt and Perette's technique has undergone a number of improvements. Since the prototyping technique results in a low-resolution images, Tiddeman et al. [38], proposed using wavelets to enhance facial texture. Fu and Zheng [39] proposed a prototyping framework to transfer different views in the 2D domain, thus they were able to render both frontal and semi-frontal images. Kemelmacher-Shlizerman et al. [40] extended Burt &

Perett's work, for unconstrained images. Their improvements include the formation of a large database, implementation of a robust face alignment procedure, as well as a technique for compensating illumination variations; a relighted average face is used in place of the normal average face. One challenging problem of this prototyping approach is the fact that the ageing effect applied to different people is the same provided they belong to same age group. Additionally, the averaging procedure results in low-resolution output. Furthermore, face expression is not completely normalised hence ageing a photo that exhibits facial expression results in an output with magnified expression; this usually distorts the output image. Recently, Wang et al. [41] proposed the use of recurrent neural networks (RNN) to enhance the prototyping method by preserving a person's identity. Using intermediate transition states, the RNN was used to smoothly transform the face across different ages. Unfortunately, RNNs require hundreds of thousands of images for training. Additionally, the method still suffers from averaging effects, thus, its output is still having low resolution. Besides, it is also not robust to facial expressions.

A more mathematical method has been taken by Scandrett et al. [42]. This work proposed, two principal component analysis (PCA) based linear equations to describe the face shape and texture, this they termed *aging axis*. Utilizing the two equations, age progression of the shape and texture were then conducted independently. Geng et al. [43], proposed a technique that finds missing faces in an ageing pattern via solving an expectation minimization algorithm, this they termed AGES. Both methods discussed above, are not robust to facial expression, they also produce low resolution images. Suo et al. represented

faces in an age group using hierarchical And-Or graphs. Unlike other researchers, their graph based Markov model also captured details of the forehead as well as the hair for age estimation and progression. However, this technique is only effective on adult faces. Markov model age progressor was again proposed in [44] to generate adult and young faces. It's obvious setback is that the technique suffers from ghosting; an image rendering artefact where warping failure results in distorted, blurry output that is unnatural [45]. Usually, ghosting produces disfigured faces that are hard to recognise.

Lanitis et al. [15] achieved age progression using AAM [46]. Facial features were extracted using AAMs, afterward, an ageing function that relates ages to the raw AAM features was defined. Using regression, ages were estimated and additionally, age progression was realised by computing a new set of AAM features. To be precise, a number of AAM vectors were generated for each age in the training data. Next, the features were stored in a lookup table. In cases where there were several subjects having the same age, the average vector corresponding to that age was computed and stored. To synthesise a new face, features of the current and projected ages were retrieved from the lookup table and then their difference was added to the individual's original AAM parameters. The technique suffers from a number of limitations. It works by adding or subtracting average AAM features, which is actually not far from the prototyping methods discussed earlier. Adding or subtracting "averages" partially masks the identity of the subject, it also results in low resolution images and ghosting. Secondly, being dependent on the lookup table, one cannot synthesise an age that is not in the training set. Lanitis' approach has been utilised by several researchers including [10], [36] and [37]. The idea of using AAMs has also been

extended to 3D by a number researchers [49]–[51]; this involves the use of the Morphable model [52].

Other researchers treated the problem as that of occlusion removal [53] or missing data recovery [48]. In general, all the methods discussed above are affected by peculiar problems which include, varying facial expressions, image noise, and low resolution.

To this end, this study aims to solve this problem by first revisiting the classical approach of Lanitis et al. [6] and improving it by explicitly solving for an ageing function that extrapolates faces even if the projected age is not available in the training set, thereby tackling the problems associated with lookup table. Other improvements include the development of an appearance model that is robust to image noise and facial expression. Lastly, a texture enhancement framework is proposed, in order to compensate for low image resolution.

2.2 Age Estimation

Over the last two decades, age estimation has been studied extensively, due to its numerous real-world applications, which include:

- *Age-Based Access Control*: Cigarette vending machines are a convenient means of purchasing tobacco. However, the benefit comes with a great setback; the underage can also buy tobacco without restriction. To this end, age estimation systems [54] play significant role. This same principle of access control has a wide range of applications, including but not limited to stopping adults from getting onto roller coasters and denying children access to adult movies or websites.

- *Information Retrieval*: The internet, having billions of images, is considered as the world's largest image database [55]. Despite the abundance of resources, searching the internet for images is not an easy task. Image retrieval plays significant role on social media websites for a number of applications such as, tagging people, album formation and friend suggestions. With the aid of an efficient age estimation algorithm, image retrieval can be made even more intelligent by narrowing picture selection to specific age groups [8], [56].
- *Demographic studies*: Age estimation applications can be utilised as vital tools for demographic studies [57]. The accuracy of demographic study relies on information such as gender, ethnicity and most importantly age. Age estimators can provide a means of understanding the dynamics of the population.
- *Evaluation of Age Progression Systems*: The precision of age synthesis algorithm is usually evaluated based on two factors; the degree to which it retains the identity of the subject and its ability to render a face that fits the projected age [7], [8]. While, the former, can be evaluated using a suitable image similarity measure, the most appropriate way of evaluating the latter is via age estimation. Furthermore, since most age progression algorithms are data driven, an accurate age predicting algorithm can be used to crawl for images with a view to enhancing the performance of the progression algorithm. Thus, age estimation directly affects age progression.

Similar to face detection and recognition, facial age estimation is obstructed by several factors such as head pose variation, occlusion, facial expressions, illumination variation and clutter background, to mention but a few. Yet, it is also challenged by other internal and external factors including gender, genes, health and lifestyle [58]. Hence, several approaches have been documented in the literature. Traditionally, age estimation has been achieved via a vital two-step procedure, consisting of feature extraction and pattern learning [6].

2.2.1 Feature Extraction

As an initial mechanism, feature extraction is the process of parameterizing the face with a view to defining an efficient descriptor. Several researchers focused on this concept, thereby devising numerous feature extraction methods.

Two broad categories of feature extraction explored by researchers in the literature are local and holistic techniques [59]. Local also known as part-based, or the analytic approach concentrates on salient parts of the face such as the facial anthropometry and wrinkles. Using local features, the earliest work on age estimation can be traced back to Kwon & Lobo [60]. By representing the face as ratios of distances, they classified the 2D images into three age groups; babies, young adults, and senior adults. To be precise, the computed ratios were used to discriminate infants from adults. Thereafter, they utilized facial wrinkles represented as snakelets to further separate the adults into young and seniors. Several other approaches have extended this basic idea, using Sobel edge detection with region tagging [61], Gabor filters and local binary patterns (LBP) [62] and Robinson Compass Masks [63] to define wrinkle and texture features. More detailed craniofacial growth models have also been developed to define

the ratios between facial features [29]. A drawback of local features is that they are not suited for specific age estimation because geometric features only describe shape changes, which are predominant in childhood, and local textures are limited to wrinkles, which manifest in adulthood.

Holistic, also known as global methods, consider the entire face when extracting features. Subspace learning techniques have been used extensively in the literature, these include PCA, neighborhood preserving projections (NPP), locality preserving projection (LPP), orthogonal LPP [64], [65], locality sensitive discriminant analysis (LSDA) and marginal fisher analysis (MFA) [66]. The active appearance model (AAM) [46], a statistical feature extraction method that captures both shape and texture variation, has been the most widely used technique [8]. Lanitis et al. [15] were the first to perform specific age estimation using the AAMs. Recently Biologically inspired features (BIF) [67] have been used by several researchers [68], [69] with promising results [70]. For comprehensive reviews, the reader should refer to [7], [8], [71].

Recent advances in Convolutional Neural Networks (CNNs), has resulted in a major paradigm shift. Using CNNs, features are automatically learned, facilitating the building of systems that learn from end to end. Hence, researchers have attempted to solve the problem of age estimation using CNNs. One of the earliest works is that of Wang et al. [72], where they used a 5 layered CNN to extract facial features. Their experiment on the two FGNET-AD [15] and Morph [73] databases yielded good results. However, they were unable to outperform state of the art algorithms. This could be due to the shallow nature of the architecture. Levi and Hassner [74] proposed a six layered CNN for age group classification. Niu et al. [75], used a four layered CNN to treat the

problem as an ordinal regression problem. Yi et al. [76], segmented the face into patches that were fed into a multi scale 3 layered sub-networks, afterward the outputs of the sub-networks were aggregated using a final layer. A similar approach was used by [77], however, instead of using 23 patches, they down sampled it to 8 patches per face. Liu et al. [78] fused regression and classification via a 22 layer deep CNN in order to perform apparent age estimation. All these have had little improvements on previous algorithms.

Unfortunately, training CNNs require an enormous amount of training data, often in millions. Additionally, stochastic gradient descent methods (SGD) used for training are difficult to tune and parallelize [79]. It also requires huge computational resources. With the exception of [72], all other researchers mentioned above that used CNNs, failed to compare their results to the FGNET-AD database; probably due its relatively small size.

2.2.2 Pattern Learning

The second step to achieving age estimation is pattern learning, which is the automatic mapping of facial features to target ages. Generally, researchers approach age-learning either as a regression task, or multi-class classification problem [7], [8], [80]. Following the latter approach, conventional classification algorithms such as support vector machines (SVM) [68] and relevance vector machine (RVM) [81] have been employed.

Estimation via the use of regression was first presented in [15] using a quadratic function (QF). Lanitis et al. [47] compared the QF to three traditional classifiers, shortest distance classifiers, Multi-layer perceptron (MLP) and the Kohonen Self

Organizing Maps. They reported that MLP and QF had the best performance. Geng et al. [43] described AGES a method that learns the ageing pattern of individuals and uses AAM for feature extraction. Multiple linear regression was proposed by Fu *et al*. [64]. Using Gaussian mixture models (GMM), Yan *et al*. proposed patch kernel regression [82]. For a comparison of some recent regression algorithms, the reader is referred to the work of Fern´andez *et al*. [71].

Towards this end, this work is aimed at investigating robust age estimation algorithm that seamlessly fits into the age progression framework and at the same time thrive on both small and huge datasets.

3 Face Synthesis using Active Appearance Models

Here age progression is achieved by using AAMs to describe the human face, thereafter, the extracted features are fed into various linear regression models. A thorough evaluation of these different approaches is then conducted.

3.1 Introduction

This chapter describes the first contribution of this dissertation, which is to improve the classical method of [15]. As discussed earlier, Lanitis et al. [15] used grayscale AAM to extract facial features, then a lookup table was formed where average features for each age were saved. To progress an image, the individual appearance features were manipulated by leveraging the parameters of the current and progressed age from the lookup table. Detail of their procedure is shown in Figure 3.1. This technique relies on the lookup table, thus it only render's ages that are contained therein.

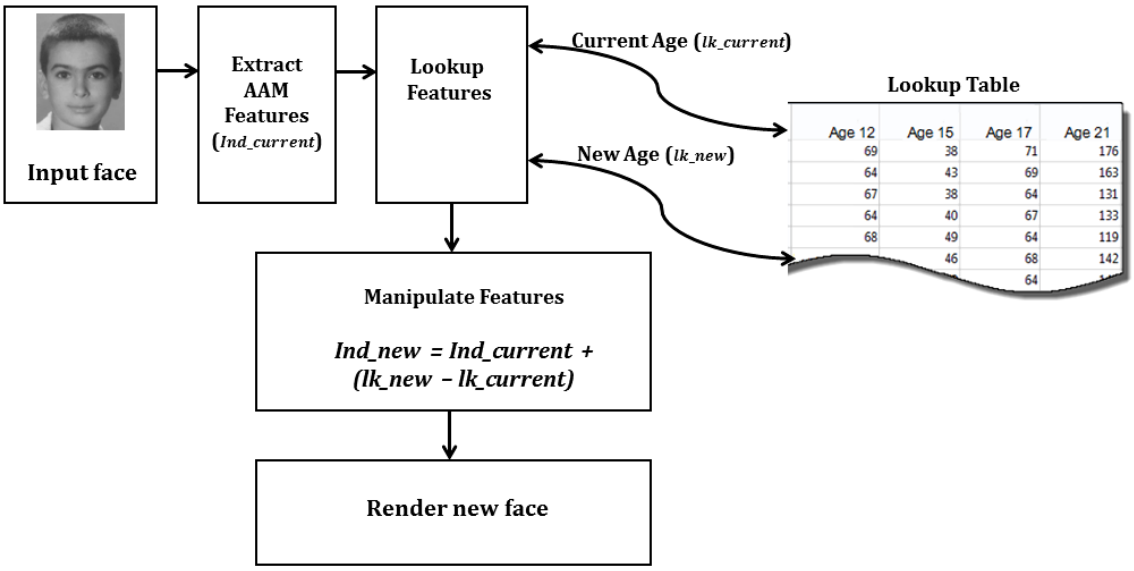


Figure 3.1: Lanitis et al.'s Face Synthesis Procedure.

Besides reliance on ages contained in the lookup database, the method also suffers from a number of setbacks including but not limited to ghosting, low resolution, and intolerance to noise. Hence, an improvement is proposed in this chapter, using AAM features, a technique that does not rely on a lookup table is devised.

The rest of the chapter describes the development of a colour-based AAM for face feature extraction, thereafter, a simple algebraic procedure for progressing ages is derived; a linear function is used to map ages to corresponding face features. Subsequently, the inverse of the function is used to achieve age progression. It may perhaps be observed that the concept of using an ageing function was mentioned by [15] as well as other researchers that used their approach, however it was only implied as they did not explicitly derive the inverse of the function. In a later part of the chapter, two implementations of the ageing function are presented to illustrate its ability to generalise to different linear mappings. It is further shown that the proposed technique can be applied in real life cases to aid the search for missing people. It is worth mentioning that a colour based variant of the conventional AAM is considered only when the photograph to progress is a colour image. Otherwise, a grayscale AAM is utilised.

3.2 Data

Since AAM is a data driven model, an initial step to building the model is to collect images. Hence, two categories of data were formed:

Colour Image dataset: a database of 1002 high quality colour photographs was made. These were acquired from four sources; 149 images were extracted from Politecnico di Torino's "HQFaces" siblings facial images database [83], where the subjects' ages varied between 13 and 50 years, these images have been photographed under controlled lighting condition. Next, all eighty images contained in the Dartmouth Children's Faces Database [84] were obtained, here frontal images that were photographed under one lighting condition and displayed a neutral facial expression were used. The age range for Dartmouth's collection is from 6 to 16 years and a 1:1 gender ratio.

Ninety-six images were taken from FGNET aging database (AD) [85]. This is made of 1002 face-pictures of 82 people, with each subject having multiple images. Their ages are distributed in the range of 0 and 69. This dataset has varying picture qualities; from grey scale to colour images, having diverse illumination, sharpness, and resolution. Furthermore, the subjects display varying facial expressions and head pose. The remaining 677 images were carefully selected from the Internet; these subjects are mainly well-known people, with ages ranging from 1 to 70 years. In total, the database has a male to female ratio of 4:3.

Grayscale Images dataset: For instances where the image to be progressed is in grayscale format, and for testing, comparison and validation purposes, all 1002 images contained in FGNET-AD database are used to build the AAM.
With a view to reducing computational cost, all images have been cropped to a size of 340×340 pixels.

3.3 Colour-based AAM

AAM is a statistical model that captures shape and texture variability from a training dataset. The parameterised model is formed by using PCA to combine shape and texture variations, which can then be used to describe images. The model which was first proposed by Tim Cootes and his team [86] was based on grayscale images. Since both grayscale and colour images are considered in this thesis, the development of a colour-based AAM is hereby presented; this entails the development of shape and texture models and combining them in a single framework.

3.3.1 Shape Model

This models the variability of face shapes using PCA. Shapes, which are represented by a set of n landmarks defined in two-dimensional space $\mathbb{R}^2$, are first aligned to remove translation, scale, and rotational variations. Then their variation is captured and abstracted into a single parameter, this process is explained in detail below.

3.3.1.1 Image Data Annotation

Facial image annotation with landmarks is the first step in the development of a statistical shape model. Annotation involves labelling structures of interest within images. This is achieved by the use of n fiducial points, usually placed at object boundaries and key locations in order to mark particular features of the face which are not affected by rotational, scaling and translational changes; for example, the tip of the nose and the corners of the mouth.

Thus, each shape in the training set can be represented by a 2 dimensional vector $\mathbf{x}$, representing the *x* and *y* coordinates of each landmark (x_i, y_i),

$$\mathbf{x} = (x_1, x_2, \ldots, x_n, y_1, y_2, \ldots, y_n)^T. \quad (3.1)$$

With the aim of describing the face shape accurately, 79 landmarks (shown in Figure 3.2) were manually placed consistently throughout the training data i.e. *n* =79. These 79 landmarks represent the most optimum points used in [88]. While many algorithms for automatic annotation have been proposed in the literature, they are not free from defects, thus hardly giving 100% correct annotation across several images. To reduce variations and to ensure data purity, manual annotation was utilised in this work.

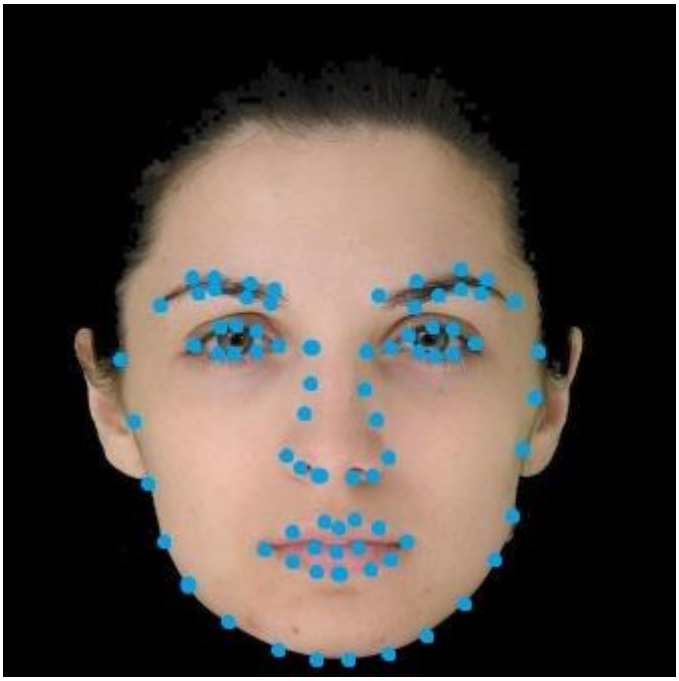


Figure 3.2: Annotation of 79 landmarks to define face shape.

3.3.1.2 Shapes Alignment

In order to capture only the shape variations from the training images, there is a need to remove translation, scale, and rotational variations. This can be achieved by using the Generalised Procrustes Analysis [89]. This refers to a set of least-squares tools used to estimate and conduct similarity transformations of point coordinates matrices between two shapes until an optimum agreement is achieved [90].

In order to perform the transformation between two shapes x_1 and x_2, scale, rotation, and translation are applied to $\mathbf{x}_1$ so that it aligns with $\mathbf{x}_2$ while minimising the Procrustes distance (PD) given by,

$$PD = \sqrt{\sum_{j=1}^{n} \left[\left(x_{j1} - x_{j2} \right)^2 + \left(y_{j1} - y_{j2} \right)^2 \right]} \tag{3.2}$$

where x_{ij} represents the x-coordinates of the j^{th} landmark on the i^{th} face and y_{ij} refers to the corresponding y-coordinates. The procedure can be summarised using Algorithm 3.1.

Algorithm 3.1 Procrustes Analysis

[1] Compute the centroid of each shape
[2] Align both shapes to the origin
[3] Re-scale each shape to have equal size
[4] Arrange the two shapes at their centroids w.r.t. position by translation
[5] Align the two shapes w.r.t. orientation by rotation

Generalised Procrustes Analysis (GPA) is an extension of Procrustes analysis, to *k* number of shapes, hence it is used to align *k* sets of shapes to a target shape, this can be summarised using Algorithm 3.2.

Algorithm 3.2 Generalised Procrustes Analysis

[1] Assume one of the shapes to be the mean shape $\bar{\mathbf{x}}$

[2] Align all the shapes to the approximate mean shape using Procrustes analysis

[3] Compute a new approximate mean $\bar{\mathbf{x}}_{new}$

[4] Repeat (2) and (3) until convergence i.e. $\bar{\mathbf{x}} \approx \bar{\mathbf{x}}_{new}$

The significance of this alignment procedure is realised by comparing the unaligned and aligned face shapes shown in Figures 3,3 and 3.4.

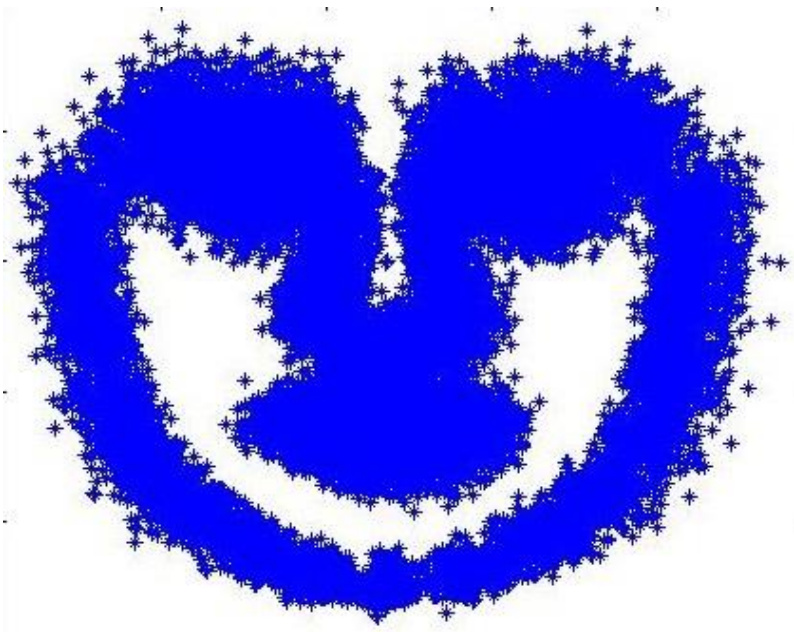


Figure 3.3: Unaligned shapes; each point represents a landmark for a given personal feature.

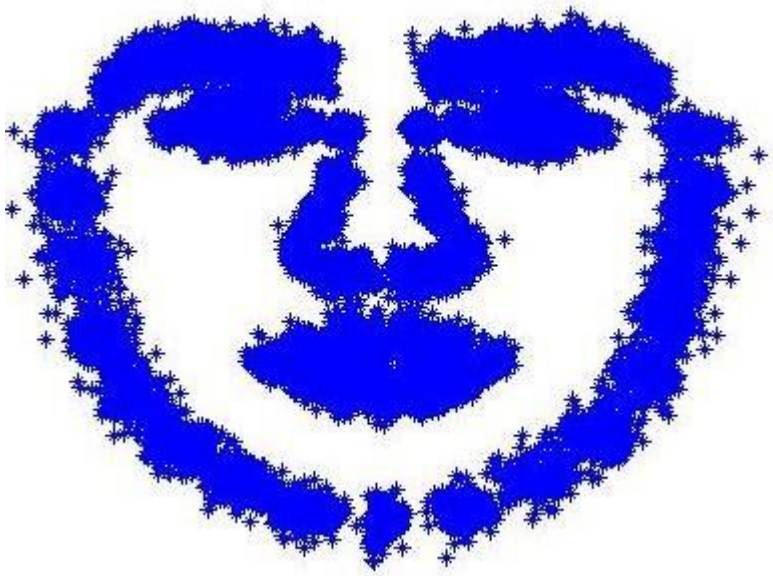


Figure 3.4: Training data shapes aligned using GPA.

3.3.1.3 PCA Modelling of Shapes

Having aligned the 2-dimensional shape vectors $\mathbf{x}_i$ using GPA, now the statistical shape model can be built using PCA as described by Algorithm 3.3.

Algorithm 3.3 Principal Component Analysis of Shapes

[1] Find the mean shape $\bar{\mathbf{x}}$ of the training data
[2] Align the shapes using GPA
[3] Compute the mean shape $\bar{\mathbf{x}}$ of the training data and subtract it from each shape
[4] Compute covariance matrix C_x of the centralised data
[5] Use eigen decomposition to project the shapes to a new basis.

Shape vectors of the training data can now be represented using a set of mutually orthogonal axes obtained using the Algorithm 3.3 above. Thus, the

statistical shape model represents each face shape using a linear equation given by,

$$\mathbf{x} = \bar{\mathbf{x}} + \boldsymbol{P}_k \boldsymbol{b}_k \tag{3.3}$$

where $\boldsymbol{P}_k$ is a matrix of eigenvectors, and $\boldsymbol{b}_k$ the shape parameters.

3.3.2 Texture Model

Here, texture is defined as the image pixel intensities. Hence, the model captures the variability of image pixels; which can be achieved using the EigenFaces approach [91]. However, the drawback of Turk & Pentland's method is the lack of pixel to pixel correspondence across the training set, which is due to person to person shape variations. With a view of tackling the stated problem, all face images can first be warped to a mean shape, thus "shape-free patches" are created [46]. To reduce global illumination variations, image pixel intensities are normalised by aligning them as closely as possible to the mean texture of the training data. In order to model colour texture, RGB channels are extracted and converted to an uncorrelated colour space so that they can be modelled independently using PCA. These steps are explained in detail below.

3.3.2.1 Image Warping

Image warping is a geometric transformation which maps positions in one image plane to positions in another image plane [92]. This is used to deform the training images to a standard shape (i.e. the mean shape), that way, pixel to pixel correspondence is achieved between faces.

There are many approaches to warping, the choice of a particular warping technique is a compromise between achieving a good match and one that distorts the image smoothly [93]. However, the most commonly used warping technique in the literature is the piecewise affine warping [94]. Thus, using the piece-wise affine warping technique, all the training images can be warped to the mean shape, thereby resulting in the shape-free patches as shown in Figure 3.5.

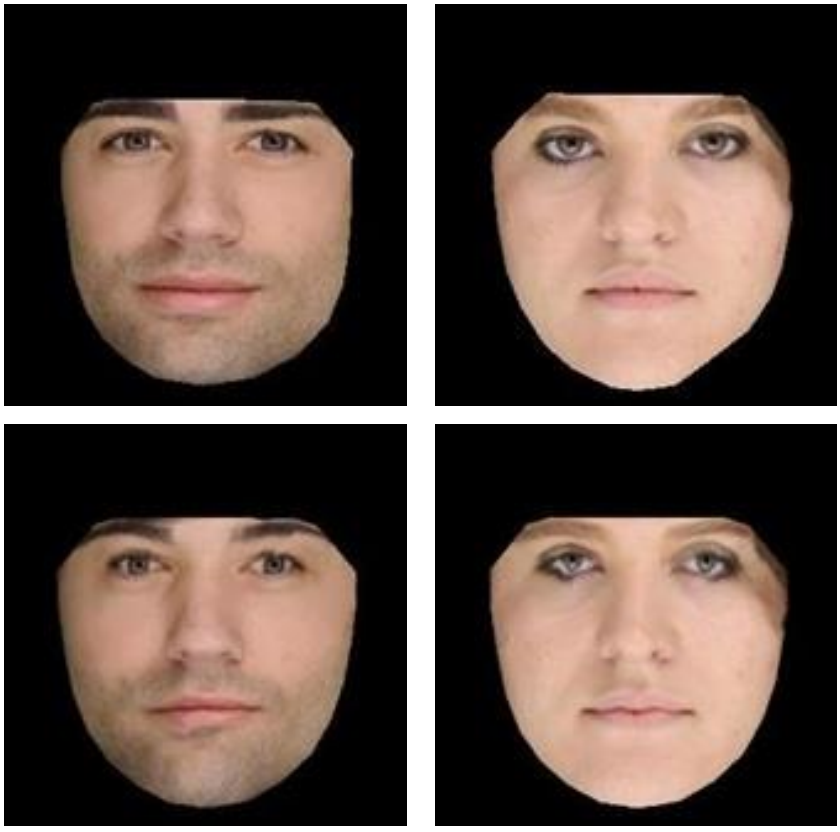


Figure 3.5: Examples of warped images, original images shown (on top) and the corresponding warped images shown below

3.3.2.2 RGB Colour Transformation

Due to the strong cross correlation that exists between RGB colour channels, modelling colour-based AAM with RGB images introduces redundancies

thereby affecting performance. To be precise, the correlation between R and G channels is ~ 0.98, between G and B channels ~ 0.94 and between B and R channels is ~ 0.78 for natural images [94]. While other colour spaces such as CIELAB have been considered in the past, the I1I2I3 colour space [95] has been most successful. This is because it uses Karhunen-Loeve Transform (KLT) to decorrelate the RGB channels. The transformation is given by,

$$I1 = (R + G + B)/3 \tag{3.4}$$

$$I2 = (R - B)/2 \tag{3.5}$$

$$I3 = (2G - R - B)/4 \tag{3.6}$$

The above-stated colour transformation was applied to the training dataset, sample image produced by each of the new channels is shown in Figure 3.6.

Figure 3.6: RGB colour decomposition into I1I2I3 removes inter-channel correlation, it also separates chromaticity and intensity

3.3.2.3 Illumination Normalization

Illumination normalization is of utmost importance in computer vision. Research has shown that differences caused by lighting variation can be more significant than the inherent person to person difference contained within images [96]. In

the literature, global lighting effects are normalised by applying a scaling α and an offset β to a texture vector $\mathbf{g}$ [46]. The normalised texture is given by,

$$\mathbf{g}_{norm} = \frac{(\mathbf{g} - \beta)}{\alpha} \tag{3.7}$$

It is worth mentioning that these scaling and offset parameters are chosen in order to match the texture $\mathbf{g}$ to the normalised mean texture $\bar{\mathbf{g}}$. Suppose $\bar{\mathbf{g}}$ is the mean of the normalised data, offset to zero mean and scaled to unit variance, the values of α and β for n training data are given by,

$$\alpha = \mathbf{g}_{norm} \cdot \bar{\mathbf{g}}, \qquad \beta = \mathbf{g}_{norm}/n \tag{3.8}$$

In this work, the global illumination normalization defined above (3.8) is applied to each $I1$, $I2$, $I3$ sub vector independently.

3.3.2.4 PCA Modelling of Texture

The statistical texture model is constructed by applying PCA to the data retrieved from each of the normalised $I1$, $I2$, and $I3$ channels. As usual, this involves, the Eigen decomposition of the covariance matrix. The texture of each image can then be approximated using three linear equations expressed as,

$$\mathbf{g}_{i1} = \bar{\mathbf{g}}_{i1} + \boldsymbol{P}_{i1}\boldsymbol{b}_{i1} \tag{3.9}$$

$$\mathbf{g}_{i2} = \bar{\mathbf{g}}_{i2} + \boldsymbol{P}_{i2}\boldsymbol{b}_{i2} \tag{3.10}$$

$$\mathbf{g}_{i3} = \bar{\mathbf{g}}_{i3} + \boldsymbol{P}_{i3}\boldsymbol{b}_{i3} \tag{3.11}$$

where $\boldsymbol{P}_{i1}$, $\boldsymbol{P}_{i2}$ and $\boldsymbol{P}_{i3}$ are the orthogonal modes of variations, and $\boldsymbol{b}_{i1}$, $\boldsymbol{b}_{i2}$ and $\boldsymbol{b}_{i3}$ the texture parameters for $I1$, $I2$ and $I3$ colour channels respectively.

3.3.3 Appearance Model

The appearance model combines the shape and three texture models. However, since the shape and three colour models can be described using their respective model parameters $\boldsymbol{b}_k$, $\boldsymbol{b}_{i1}$, $\boldsymbol{b}_{i2}$ and $\boldsymbol{b}_{i3}$, then, the appearance model can simply be built by concatenating the four variation descriptors into a single matrix given by,

$$\boldsymbol{b}_{com} = \begin{bmatrix} \boldsymbol{W}_k \boldsymbol{b}_k \\ \boldsymbol{b}_{i1} \\ \boldsymbol{b}_{i2} \\ \boldsymbol{b}_{i3} \end{bmatrix} = \begin{bmatrix} \boldsymbol{W}_k \boldsymbol{P}_k^T (\mathbf{x} - \bar{\mathbf{x}}) \\ \boldsymbol{P}_{i1}^T (\mathbf{g}_{i1} - \bar{\mathbf{g}}_{i1}) \\ \boldsymbol{P}_{i2}^T (\mathbf{g}_{i2} - \bar{\mathbf{g}}_{i2}) \\ \boldsymbol{P}_{i3}^T (\mathbf{g}_{i3} - \bar{\mathbf{g}}_{i3}) \end{bmatrix}, \tag{3.12}$$

where $\boldsymbol{W}_k$ is a diagonal matrix of weights used to compensate for the difference in the magnitude of the units of shape and texture models. PCA is then applied to the new vector $\boldsymbol{b}_{com}$ to remove any correlation that may exist between the shape and textures. This results in an appearance model given by,

$$\boldsymbol{b}_{com} = \boldsymbol{P}_{com} \boldsymbol{c} \tag{3.13}$$

$\boldsymbol{P}_{com}$ is a matrix of eigenvectors and $\boldsymbol{c}$ the appearance parameter that controls all 4 models (shape and colour channels). $\boldsymbol{P}_{com}$ can be further expressed as composed of 4 modes of direction that are associated with the two models, hence can be expressed as,

$$\boldsymbol{P}_{com} = \begin{bmatrix} \boldsymbol{P}_{com\_x} \\ \boldsymbol{P}_{com\_i1} \\ \boldsymbol{P}_{com\_i2} \\ \boldsymbol{P}_{com\_i3} \end{bmatrix} \tag{3.14}$$

Just as demonstrated in [46], the linear nature of the appearance model makes it possible to express the shape and textures in terms of $\boldsymbol{c}$. Hence, equations (3.12) and (3.13) can be used to rewrite (3.3), (3.9), (3.10) and (3.11) as,

$$\mathbf{x} = \bar{\mathbf{x}} + \boldsymbol{P}_k \boldsymbol{W}_k^{-1} \boldsymbol{P}_{com\_x} \boldsymbol{c} \tag{3.15}$$

$$\mathbf{g}_{i1} = \bar{\mathbf{g}}_{i1} + \boldsymbol{P}_{i1} \boldsymbol{P}_{com\_i1} \boldsymbol{c} \tag{3.16}$$

$$\mathbf{g}_{i2} = \bar{\mathbf{g}}_{i2} + \boldsymbol{P}_{i2} \boldsymbol{P}_{com\_i2} \boldsymbol{c} \tag{3.17}$$

$$\mathbf{g}_{i3} = \bar{\mathbf{g}}_{i3} + \boldsymbol{P}_{i3} \boldsymbol{P}_{com\_i3} \boldsymbol{c} \tag{3.18}$$

3.3.3.1 Selection of Weights

When combining the shape parameter b_k and the 3 texture parameters b_{i1}, b_{i2} and b_{i3} in (3.12) discussed in the section above, there is the need to compensate for the difference in units. Since the shape parameters were obtained from 2-dimensional distance coordinates, and the texture parameters computed from image pixels, there is, therefore, a need to make the models compatible. Here, the approach of [46] is adopted by applying a diagonal matrix W, defined as the root mean square (RMS) change in texture per unit change in shape. Given by,

$$\boldsymbol{W} = r\boldsymbol{I} \tag{3.19}$$

where r^2 is a ratio of sum of image intensity variations to the total shape variations and $\boldsymbol{I}$ is the identity matrix. Thus $\boldsymbol{W}$ can be computed using,

$$\boldsymbol{W} = \left\{ \sum_{j=1}^{n} \lambda_{gj} \Big/ \sum_{j=1}^{n} \lambda_{sj} \right\}^{1/2} \boldsymbol{I} \tag{3.20}$$

where λ_g and λ_s represent the eigenvalues of the texture and shape models respectively.

Since in this work three texture parameters are modelled independently, the mean of the sum of the pixel variations is used,

$$\boldsymbol{W}_k = \left\{ \left(\frac{1}{3} \sum_{i=1}^{3} \sum_{j=1}^{n} \lambda_{g_i j} \right) \Big/ \left(\sum_{j=1}^{n} \lambda_{sj} \right) \right\}^{1/2} \boldsymbol{I} \tag{3.21}$$

3.4 Age Progression Model

Several pieces of research [97], [98] have shown that the appearance of the face consistently changes with age and that is one of the obvious reasons why humans are able to estimate people's age by merely looking at their face, this is of course due to changes in the face shape as well as texture (skin, wrinkles, and colouration) [7]. Since AAM models both shape and texture variations, it is presumed that AAM face features capture ageing variations [15]. Consequently, it would not be a surprise to find a correlation between AAM parameters (obtained from (3.13)) and individual ages. In that case, an ageing function can be defined relating ages to vectors of AAM parameters.

$$age = f(\boldsymbol{c}) \tag{3.22}$$

In data analysis, one important question is the determination of a model to define the relationship that exists between variables [99]. Interestingly, the statistical relationship between a scalar dependent and set of independent continuous variables can be defined using a regression model [100]. While there are many types of regression models, ranging from linear to nonlinear variants, there is no overall best model [101]. However, as a rule of thumb, it is usually best to start off with a simple and interpretable model [101].

Thus, the ageing function that defines a relationship between age and face features is represented via a linear model. Due to its simplicity and interpretability, as will be demonstrated shortly, a linear model gives the ease of inversion and eventual attainment of an age progression framework. Hence the relationship between face features and individual ages can be expressed as,

$$age = \alpha + \boldsymbol{\beta}^T \boldsymbol{c} \tag{3.23}$$

subject to $age_i = f(c_i)$

where α is an offset and $\boldsymbol{\beta}$ is a vector of regression coefficients and $\boldsymbol{c}$ are the face features, and i is the index of individual whose face is to be progressed. Consequently, new face (AAM) features can be generated by inverting the ageing function by,

$$\boldsymbol{c} = f^{-1}(age) \tag{3.24}$$

The equation (3.24) above gives us the ability to construct a new face, by inputting a target age. To achieve a specific solution, the inverse of equation

(3.23) can be computed. Assuming a zero offset, the equation can be written as,

$$age = \boldsymbol{\beta}^T \boldsymbol{c} \tag{3.25}$$

The computation of the inverse, then implies finding the inverse of the vector of coefficients $\boldsymbol{\beta}$. Since $\boldsymbol{\beta}$ is not a square matrix, a possible solution to the inverse problem can be achieved using the Moore-Penrose pseudoinverse [102]. Thus, new face features can be approximated using,

$$\hat{\boldsymbol{c}} = \boldsymbol{\beta}^{\dagger} age \tag{3.26}$$

Since equation (3.26) is a projection, it then follows that the appearance parameter $\hat{\boldsymbol{c}}$ for a certain age will be the same for all individuals at the same age. However, in linear algebra, given a transformation *T* (e.g. projection) of two vectors (c_1 and c_2) that result in same output vector, there actually exists a difference between the two, at a direction which is perpendicular to the two vectors. Hence given,

$$\begin{aligned} Tc_1 &= c_p \\ Tc_2 &= c_p \end{aligned} \tag{3.27}$$

Subtracting the two equations above gives us,

$$T(c_1 - c_2) = 0 \tag{3.28}$$

Thus there is a nonzero vector $T(c_1 - c_2)$ whose image is zero. This implies that the two projections differ by an orthogonal element in the null space. It can then be presumed that each person's AAM parameter contains two orthogonal components; the age-component (c_{age}) that is computed as a projection using (3.26) and an identity-component (c_{id}) differs from individual to individual. It is also obvious that ageing component is orthogonal to the identity component. Interestingly, AAM parameters are inherently orthogonal as a result of the PCA conducted in equation (3.13). Intuitively, the parsimonious elements in $\boldsymbol{c}$ used to fit the regressor in (3.25) form the ageing-component and the remaining orthogonal elements form the identity part. Hence the appearance parameter for each individual can be expressed as,

$$\boldsymbol{c}_{individual} = \boldsymbol{c}_{age} + \boldsymbol{c}_{id} \tag{3.29}$$

In order to compute the AAM parameters for a new age $\boldsymbol{c}_{new}$, equation (3.29) is first used to retrieve the identity component $\boldsymbol{c}_{id}$ for the person, then using equation (3.26), the age component for the new age can be computed. The sum of these two components gives us $\boldsymbol{c}_{individual\_new}$ i.e. the AAM parameter for that individual at a new age. The procedure for age progression has been summarised in Algorithm 3.4 and Figure 3.7 below.

Algorithm 3.4 Age Progression

[1] Given the raw AAM parameters $\boldsymbol{c}_{individual\_now}$ at a current age

[2] Compute the age component $\boldsymbol{c}_{age\_now}$ at current age, using equation (3.29)

[3] Calculate the person's identity features $\boldsymbol{c}_{id} = \boldsymbol{c}_{individual\_now} - \boldsymbol{c}_{age\_now}$

[4] Compute the age component $\boldsymbol{c}_{age\_new}$ for the new age using procedure [2]

[5] Sum the results in [3] and [4] to get the raw AAM parameters at new age

$\boldsymbol{c}_{individual\_new} = \boldsymbol{c}_{age\_new} + \boldsymbol{c}_{id}$

[6] Reconstruct the face using equations (3.15) to (3.18)

In practice $\boldsymbol{c}_{individual\_new}$ minimises $\| \boldsymbol{c}_{individual_{now}} - \boldsymbol{c}_{individual_{new}} \|^2$

An obvious advantage of the new method is the ability to compute AAM parameters for any age, including those that are not in the training set. Its ability to interpolate ages eliminates the dependency on the lookup table. Furthermore, the AAM parameters can now be computed even for non-integer ages.

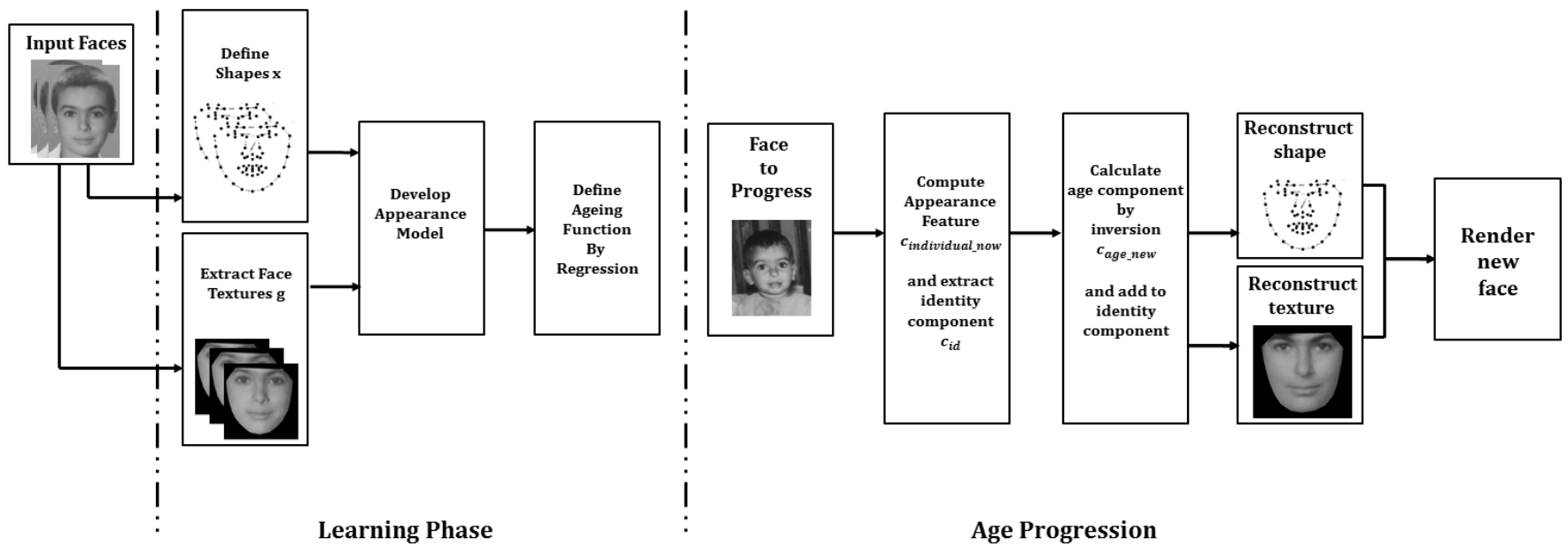


Figure 3.7: Age Progression Framework.

3.5 Generalisation of Ageing Model

Considering the framework described in the previous section, it is quite obvious that the regression model is at the heart of its success. Having initially, chosen a simple ordinary least squares (OLS) model, research has shown that despite its simplicity, interpretability, and popularity the model has a number of pitfalls [103]. These include its sensitivity to outliers and the effect of too many features, especially in cases where the number of observations is n equals to the number of features p. Worst is even the case when $n < p$ because the algorithm completely fails [103]. As a matter of fact, OLS work well only when three conditions are met; the number of features are very few, collinearity is minimum and the relationship between predictor and dependent variables is fully understood [104]. Since the number of features at hand are relatively high and the relationship that exists between the age and facial features is not fully understood, it's presumed that OLS is not the ideal regressor to deploy. Thus, the framework can be extended by using a linear model that is robust to the dimensionality of the features, and that is not sensitive to the underlying relationship between the variables. An ideal linear model that meets such criteria is partial least squares regression (PLS) model.

PLS a simultaneous dimensionality reduction and regression technique which is well suited for regression when $n < p$ (i.e. ill-posed) can be embedded into the framework with a view to enhancing the rendering ability of the age progressor.

PLS regression was introduced by Herman Wold [105], [106], and has been an alternative to OLS regression [107], it improves OLS in two major ways; increased prediction accuracy and enhanced data representation. The statistical

method creates latent features via a linear combination of the predictor ($\boldsymbol{X}$) and response ($\boldsymbol{Y}$) variables.

Let $\boldsymbol{x} \in \mathbb{R}^m$ i.e. $\boldsymbol{X} = \{\boldsymbol{x}_i\}$ be an $n \times p$ matrix of predictor variables (i.e. having n observations and p features) and $\boldsymbol{Y}$ be an $n \times m$ matrix of response variables. PLS decomposes the two matrices into,

$$\begin{aligned} \boldsymbol{X} &= \boldsymbol{Z}\boldsymbol{P}^T + \boldsymbol{E} \\ \boldsymbol{Y} &= \boldsymbol{U}\boldsymbol{Q}^T + \boldsymbol{F} \end{aligned} \tag{3.30}$$

$\boldsymbol{Z}$ and $\boldsymbol{U}$ are $n \times k$ matrix of linear latent (scores) having a reduced dimension, thus $k \ll p$. $\boldsymbol{P}$ and $\boldsymbol{Q}$ are loadings, $\boldsymbol{E}$ and $\boldsymbol{F}$ are matrices of residuals. The scores $\boldsymbol{Z}$ can be computed directly from the feature set $\boldsymbol{X}$ via,

$$\boldsymbol{Z} = \boldsymbol{X}\boldsymbol{R} \tag{3.31}$$

where the matrix of weights $\boldsymbol{R} = \{\boldsymbol{r}_1, \boldsymbol{r}_2, \dots \boldsymbol{r}_k\}$ is computed by solving an optimization problem. The estimate of $\boldsymbol{k}$th direction vector is formulated as,

$$\begin{gathered} \hat{\boldsymbol{r}}_k = argmax_{\boldsymbol{r}}\, \boldsymbol{r}^T\boldsymbol{X}^T\boldsymbol{Y}\boldsymbol{Y}^T\boldsymbol{X}\boldsymbol{r} \\ \text{such that } \boldsymbol{r}^T\boldsymbol{r} = 1 \text{ and } \boldsymbol{r}^T\boldsymbol{X}^T\boldsymbol{X}\boldsymbol{r}_i = 0 \end{gathered} \tag{3.32}$$

for $i = 1 \dots k - 1$

Thus PLS captures the directions of highest variance in $\boldsymbol{X}$ as well as the direction that relates $\boldsymbol{X}$ and $\boldsymbol{Y}$ [108]. While many methods for computing PLS

have been proposed in the literature, in this work, the SIMPLS algorithm proposed by Sijmen De Jong [109] is utilised; thus taking advantage of the method's speed. After computing the latent scores $\boldsymbol{Z}$, PLS regression coefficient is defined as,

$$\boldsymbol{Y} = \boldsymbol{X}\beta^{PLS} + \boldsymbol{F} \tag{3.33}$$

In this work, $\boldsymbol{Y}$ corresponds to a vector of ages while $\boldsymbol{X}$ represents the facial features. Since PLS takes the relationship that exists between the predictor and response variables it is believed that the age-component defined using the model will be more parsimonious and will be the best representative of the age, thus giving an efficient means of separating the age-components from the rest of the facial features that represent identity.

Despite the shrinkage ability and efficiency of PLS regression in problems with a large number of variables, the fact that it is a linear combination of all variables, makes it include information of both relevant and irrelevant (noisy) data. Recently, Chun and Keles [110] proposed sparse PLS (sPLS) regression, in order to integrate sparsity into the conventional PLS dimension reduction procedure there by ensuring the selection of only relevant variables. Their findings showed sPLS to be more efficient than PLS especially when the number of variables is very large as compared to a small sample size [111]. Adopting the idea of [111] thus provides an even further extension to the age progression model.

In order to realize sPLS regression, LASSO [112] L_1 regularization is imposed into the linear formulation of the PLS. This is done by using a tuning parameter η chosen within the range $0 \leq \eta \leq 1$. In this thesis, CRAN 'spls' Package Version 2.2-1 [113] was used for the implementation of sparse partial least squares regression.

3.6 Experiments

3.6.1 Data Usage Protocol

In all the experiments conducted in this chapter and the chapters that follow, data used for training the model is same as those described in section 3.2. For testing the progression techniques, images of all 82 subjects of the FGNET-AD are utilised. The database contains images of subjects in chronological order, hence the image to be progressed and the image of the subject at the progressed age are both available. To evaluate progression to adult age, as well age reversals, 50% of images chosen are from young to adult, and the other half of the test images are for reversing adult faces to a younger age. As shown in Figure 3.8, the selected test images exhibit varying head pose, facial expression, as well as photo quality. This choice was made to fully evaluate the robustness of the algorithms to varying obstructing factors. To furthermore ensure unbiased judgement of the accuracy of algorithms, images used for testing are excluded at the time of training.

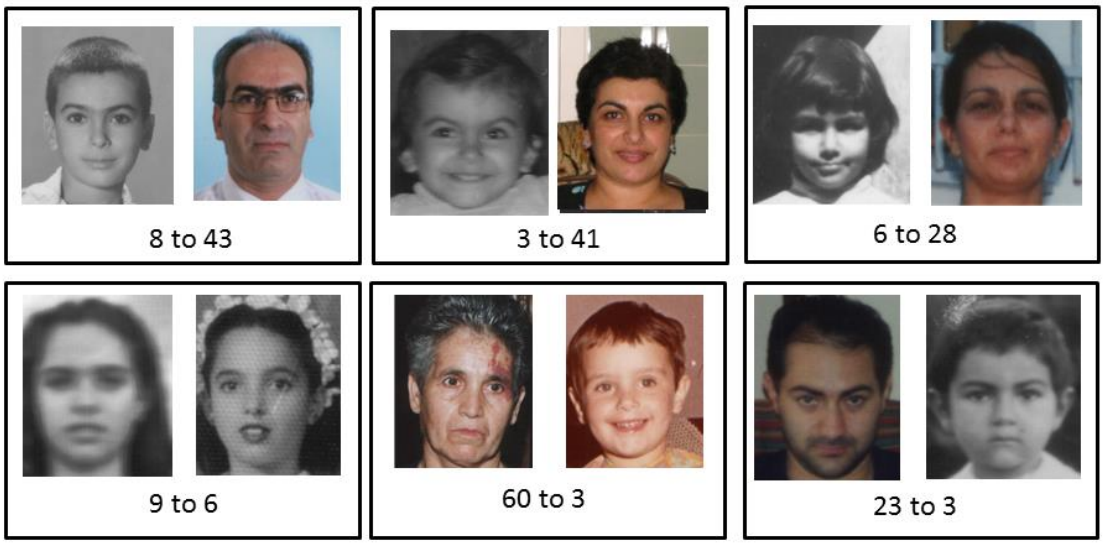


Figure 3.8: Sample images of 82 subjects used for performance evaluation. Images show varying pose, facial expression and photo-quality.

3.6.2 Performance Evaluation

Performance assessment is of the utmost importance in facial analysis as it helps to identify gaps as well as to benchmark progress in the field. An effective means of assessing the facial age progression should have two main priorities: evaluating the ability to synthesise images that fit the intended age, and checking the ability to retain the identity of the subject in age altered images. There are two categories of performance evaluation techniques; machine based and human based methods [7], [114]; both methods are used in this work.

Here, the machine based test, also known as an objective test is conducted using Euclidean Distance (ED). To evaluate the ability of the algorithm to retain identity, distance E_{syn} between the generated image at age t_{now} and the original image at age t_{old} is computed as,

$$E_{syn} = \sqrt{\sum_{i=1}^{N}(c_{Syn} - c_{old})^2}. \quad (3.34)$$

where c_{Syn} is appearance feature of the synthesised image, c_{old} and c_{now} are the vectors of facial features for the real image at ages t_{old} and t_{now}. Next, the distance E_{real} between the real images at ages t_{old} and age t_{now} is computed.

$$E_{real} = \sqrt{\sum_{i=1}^{N}(c_{now} - c_{old})^2} \quad (3.35)$$

Finally these two distances are compared by computing the absolute value of their difference (E_{dif}) given by,

$$E_{dif} = abs(E_{real} - E_{syn}). \quad (3.40)$$

The above absolute distance is scaled to have values between 0 to 1. With 0 indicated exact match (i.e. 100% retained identity) and 1 indicating complete nonsimilarity. In other words, the closer a synthesised image resembles the subject's identity, the closer is distance E_{dif} to 0. Subsequently, E_{dif} is expressed as a similarity score in percentage, such that a score of 100% denotes perfect match and 0%, on the other hand, refers to no match. The percentage score is given by,

$$Sc = \frac{1}{1 + E_{dif}} \times 100\%. \quad (3.41)$$

After computation of the scores for a single test image, the overall performance of an algorithm is evaluated by computing the average scores over all 82 test images.

Human based tests, also known as subjective tests, are conducted by asking 29 human observers to perform two tasks listed below. The human observers were all students of University of Bradford of caucasian origin, having male to female ratio of 19:10 with ages ranging from 19 to 35.

i. Look at the synthesised image and give one of four verdicts; “Yes : it meets the intended age”, “Below: it looks younger than the intended age”, “Above: it appears older than the progressed age” or “Undecided”.
ii. Look at the synthesised image and give a score between 0 and 10 to indicate how closely it resembles the test image. A score of zero signifies no resemblance and 10 refers to a perfect match. Finally, all the scores are averaged to give a single value between 0 and 10 that describes how best an algorithm preserve’s people’s identity.

Summarily the human based test answers two questions, the ability of an algorithm to render well-aged images and its ability to retain identity.

Finally, the results of both machine and human tests are average over the total number of 82 subjects.

3.6.3 Results

Three sets of experiment were conducted using the proposed age progression framework. First using OLS regression and subsequently utilising PLS and

sPLS algorithms respectively. In the course of AAM training, the PCA that binds shape and texture models gives rise to $n-1$ non zero orthogonal projections, hence resulting in $n-1$ features.

For all three age progression variants, the number of features utilised as ageing components were chosen via cross-validation, i.e. by considering the root mean square errors (RMSE) of regression as the number of features varied. Figure 3.9 show's that the OLS regressor requires 152 components to achieve minimum RMSE. In Figure 3.10, it is quite obvious that for PLS, the optimum RMSE is achieved with just 40 components. Meanwhile, tunning the sparsity parameter of the sPLS algorithm gives the best performance when $\eta = 0.7$ (see Figure 3.11).

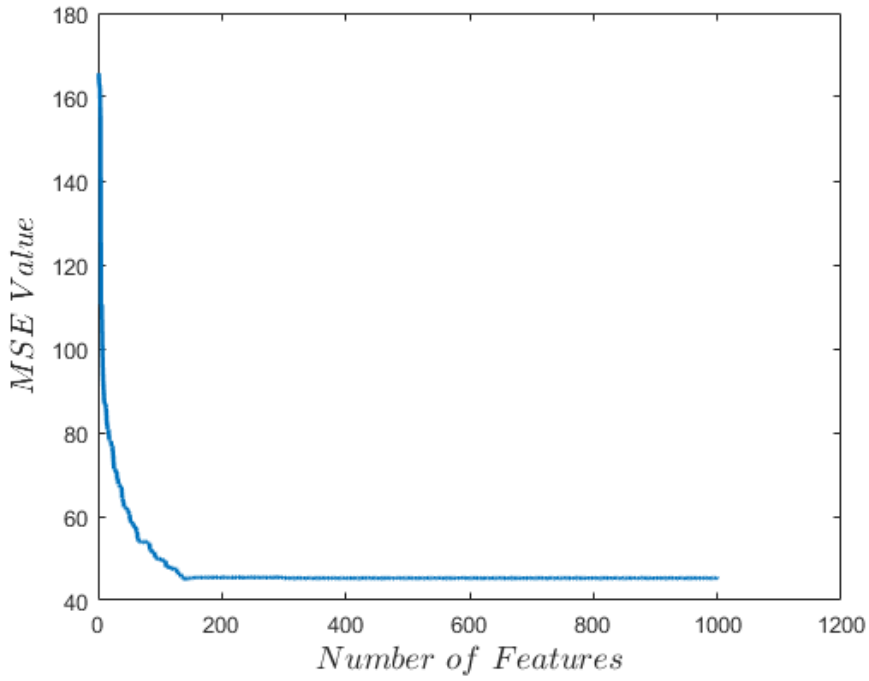


Figure 3.9: Mean square error of OLS regression per number of features. Smallest error is achieved when the number of features is 152.

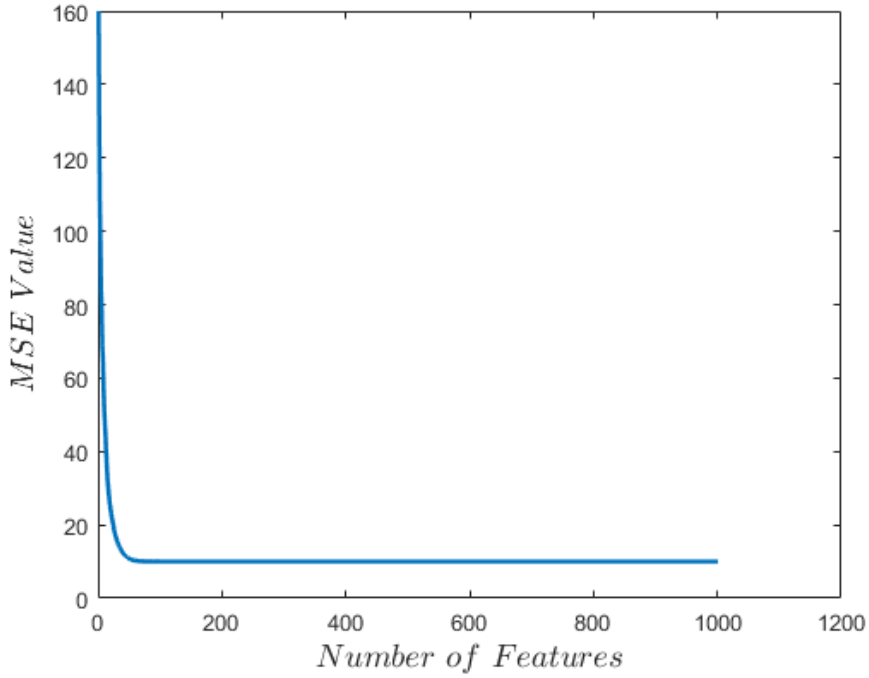


Figure 3.10: Mean square error of PLS regression per number of features. The optimum number of features is 40.

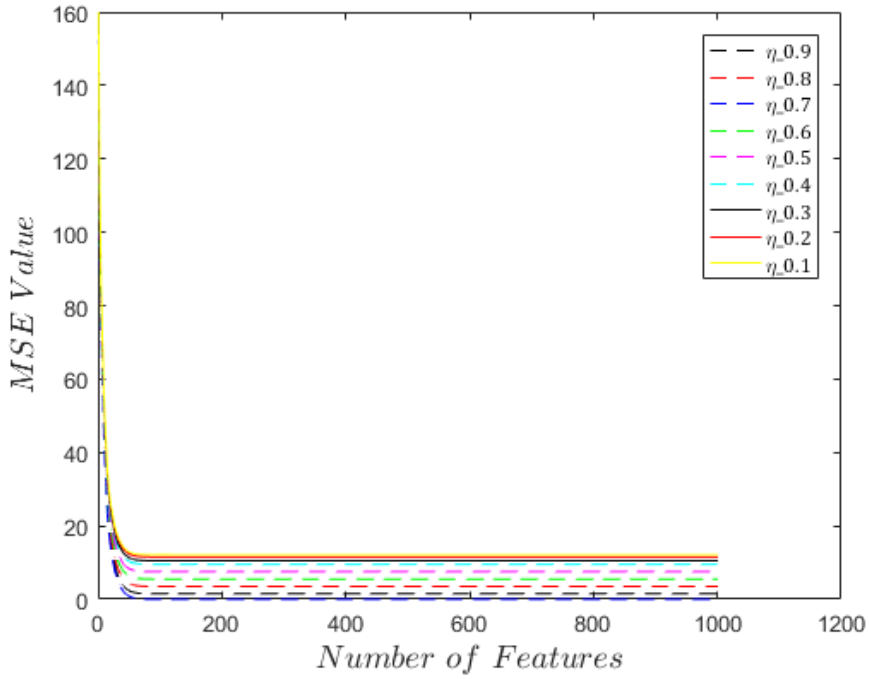
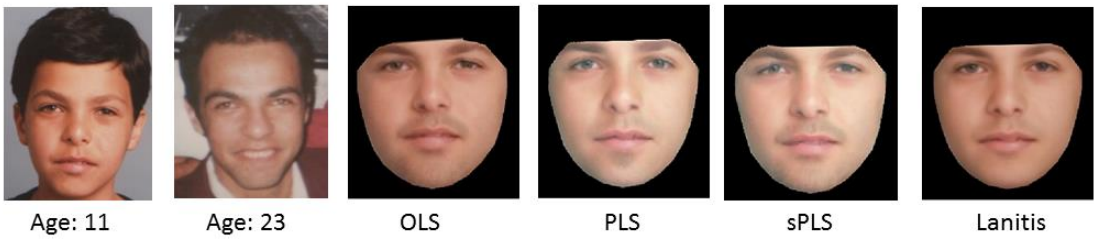
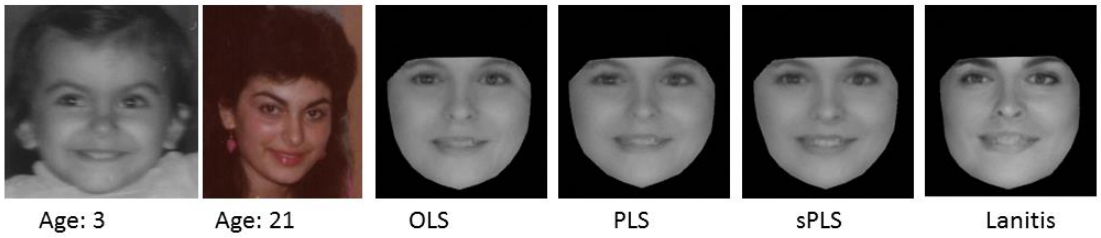


Figure 3.11: Mean square error of sPLS regression at various regularisation values. Optimum performance is achieved when $\eta = 0.7$.

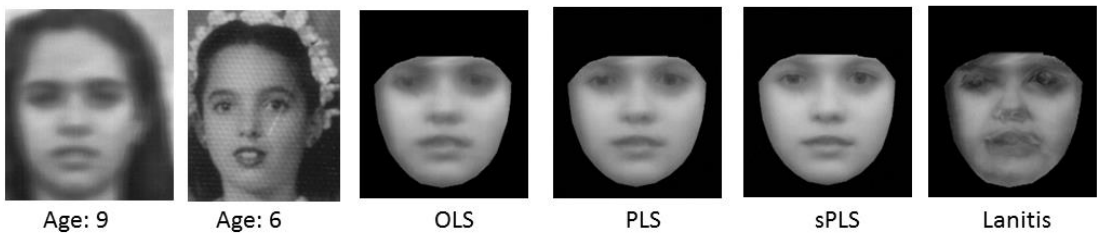
Next, age progression tests were conducted using the 82-subjects' evaluation data. Samples of the age progression results are shown in figure 3.12. Farthest to the left are the test images, adjacent to the test images are the subjects' ground truth images at projected age, next synthesis results generated using OLS, PLS, sPLS and Lanitis method are placed in arranged in that order.



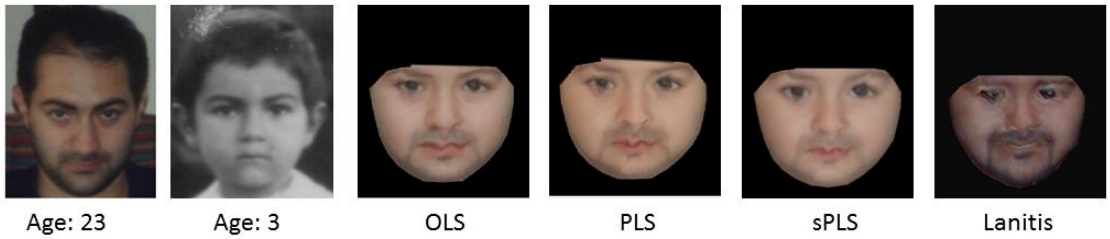
(a)



(b)



(c)



(d)

Figure 3.12: Sample of age synthesis results. Images on the farthest left are the test images.

Observing sample of the generated images presented in Figure 3.12 above, it can be seen that the algorithms perform best when deployed on frontal, neutral face images. In 3.12 (a) facial expression hinders the reconstruction accuracy,

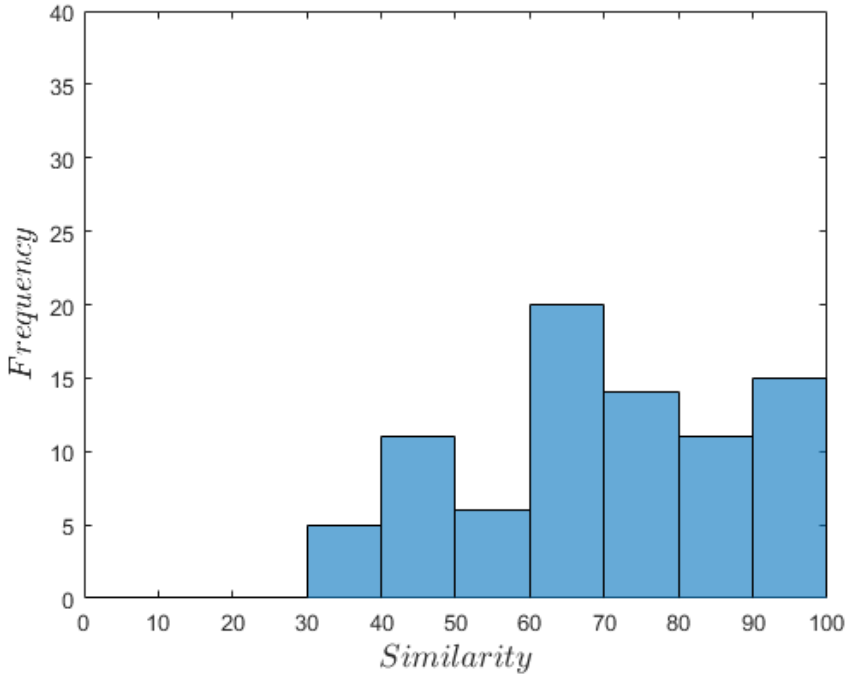
especially around the mouth region. While In Figure 3.12(c), the probe image is noisy and distorted, the proposed techniques clearly, outperform Lanitis's method which suffers from ghosting effect. Furthermore, results in Figure 3.12(d) reveal the problem of facial hair, especially when rendering young faces.

To fully explore the performance of the algorithms, extensive machine (objective) and human (subjective) evaluations were conducted. The mean scores for the objective tests presented in Table 3.1 show that all three proposed techniques have better identity retention ability as compared to the method in [15].

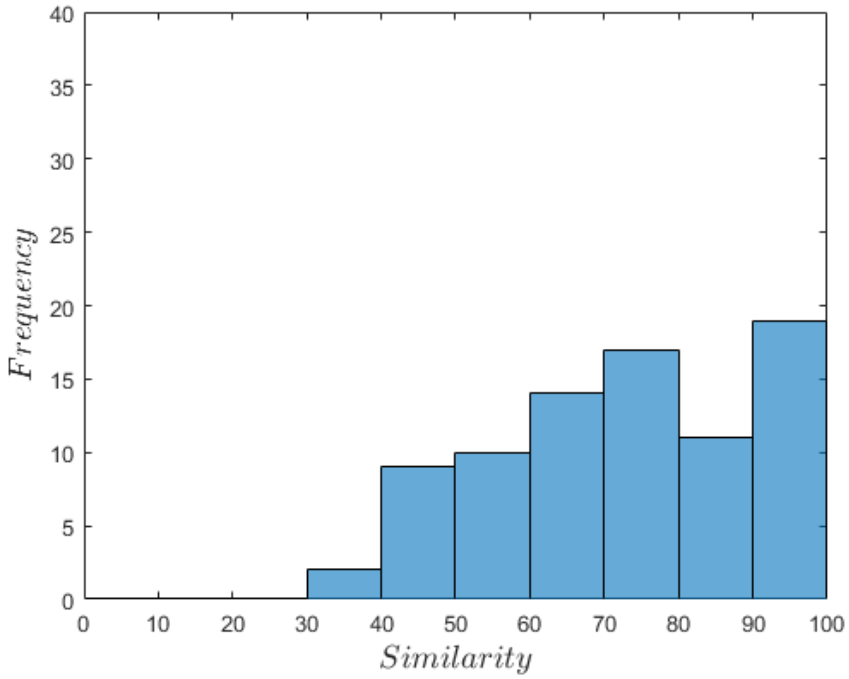
Table 3.1: Mean scores of objective test (AAM based models).

Technique	**Mean Scores (%)**
Lanitis [15] method	69.82
OLS approach	71.86
PLS approach	73.16
sPLS approach	74.36

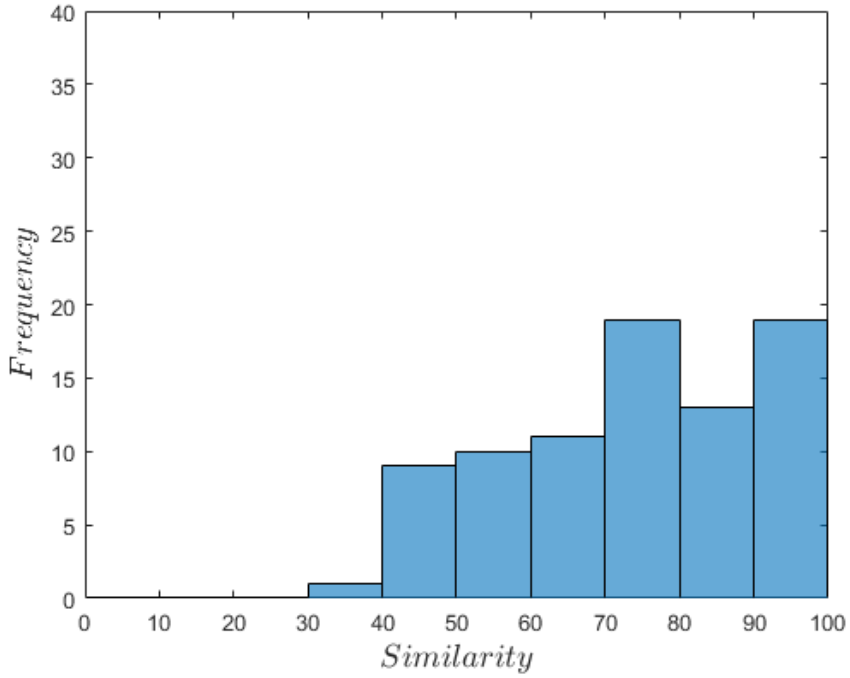
To gain more insight into the algorithms' performance, frequency of the scores were plotted using histograms.


Frequency
Similarity
0
5
10
15
20
25
30
35
40
0
10
20
30
40
50
60
70
80
90
100

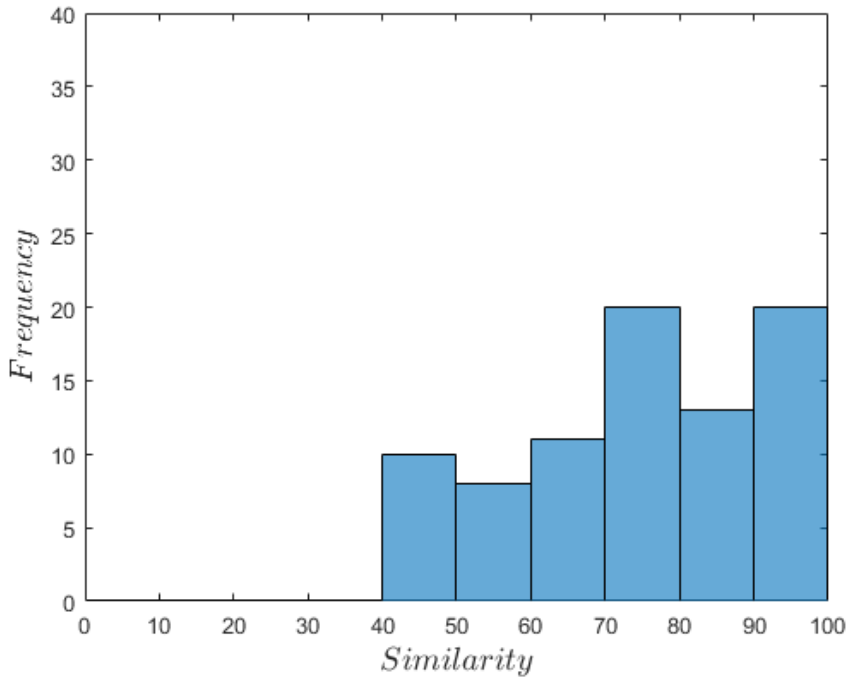
(a)


Frequency
Similarity
0
5
10
15
20
25
30
35
40
0
10
20
30
40
50
60
70
80
90
100

(b)



(c)



(d)

Figure 3.13: Histogram of objective test scores (AAM based models) (a) Lanitis' method (b OLS approach (c) PLS method (d) sPLS technique.

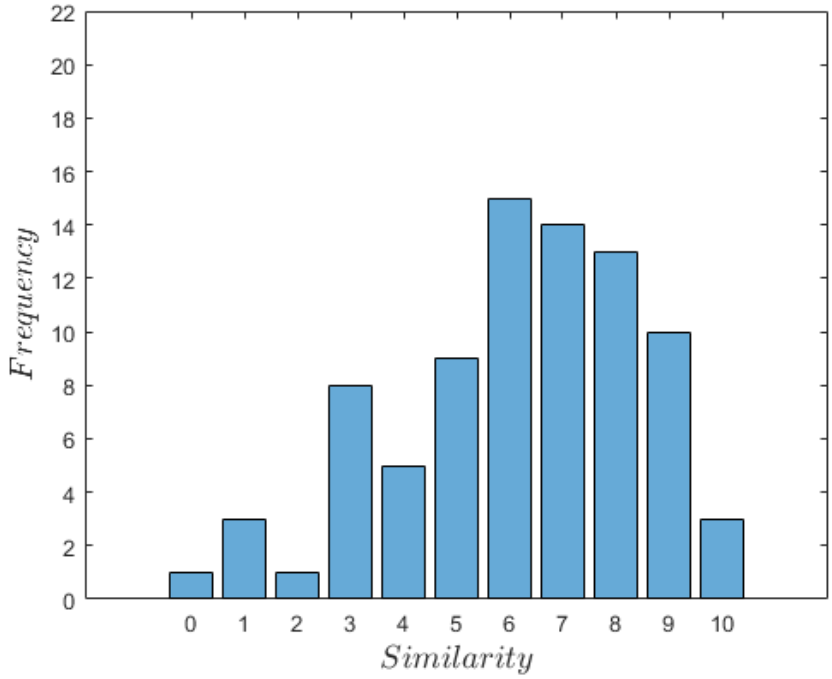
Figure 3.13 (a), (b), (c) and (d) corroborate the results presented Table 3.1. To further check if the improvements are statistically different, two-sample Kolmogorov-Smirnov (KS) test was used to compare the mean scores. It was observed that at 5% significance level, the sPLS based algorithm showed significant improvement over the method of Lanitis (D=0.2105, p=0.0450) as well as the OLS (D=0.2083, p=0.0491) technique. However no significant difference was observed when OLS (D=0.0950, p=0.914) and PLS (D=0.1345, p=0.424) methods were compared to each other and to the method of Lanitis. The sPLS algorithm obviously performs best hence having the ability to retain the subject's identity. This also indicates sPLS exhibits lesser reconstruction error as compared to the rest.

Table 3.2: Mean scores of subjective test (AAM based model).

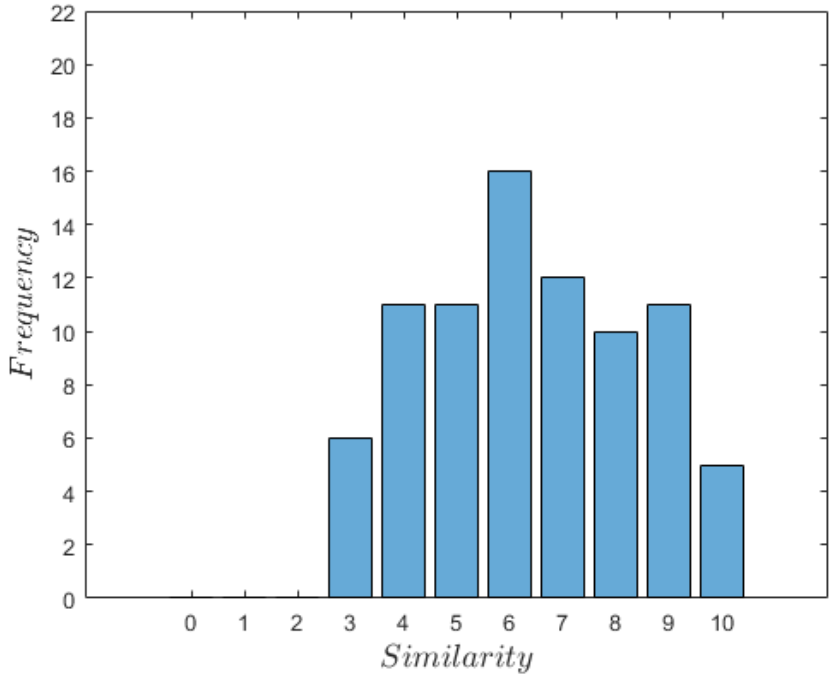
Technique	**Mean Scores**
Lanitis [15] method	6.1707
OLS approach	6.4146
PLS approach	6.6585
sPLS approach	6.7805

In Table 3.2, the mean scores of human evaluations show sPLS-based rendering to have better identity preservation capability, it also shows that the proposed methods surpass the classical technique used of [15]. Here also two-sample KS test revealed significant difference between the proposed sPLS algorithm (D=0.2095, p=0.0483) and that of Lanitis' method. Although the mean

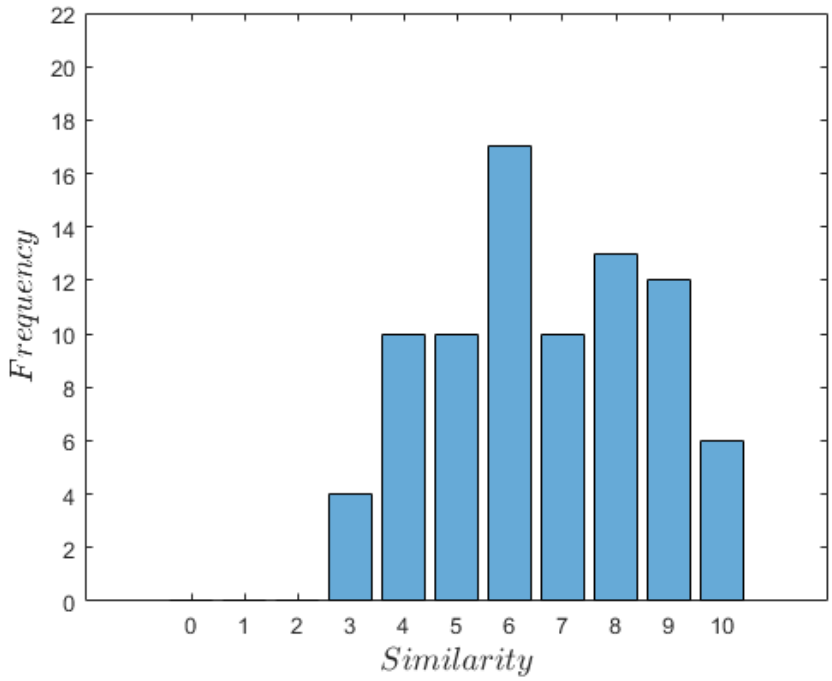
socres of the OLS (D=0.1090, p=0.6813) and PLS (D=0.1829, p=0.1142) methods apparently showed improvement, the non parametric KS test did not reveal difference at 5% significance level. Bargraphs shown in Figure 3.14 further give an in-depth view of subjective assessment. Due construction artefacts, some images generated using Lanitis method appear completely distorted hence it is no surprise that they were scored very low by the human observers, a particular case is that of the image shown in Figure 3.12(c). As can be expected, the bar graphs also show more high score bins as the regression algorithms get more sophisticated.



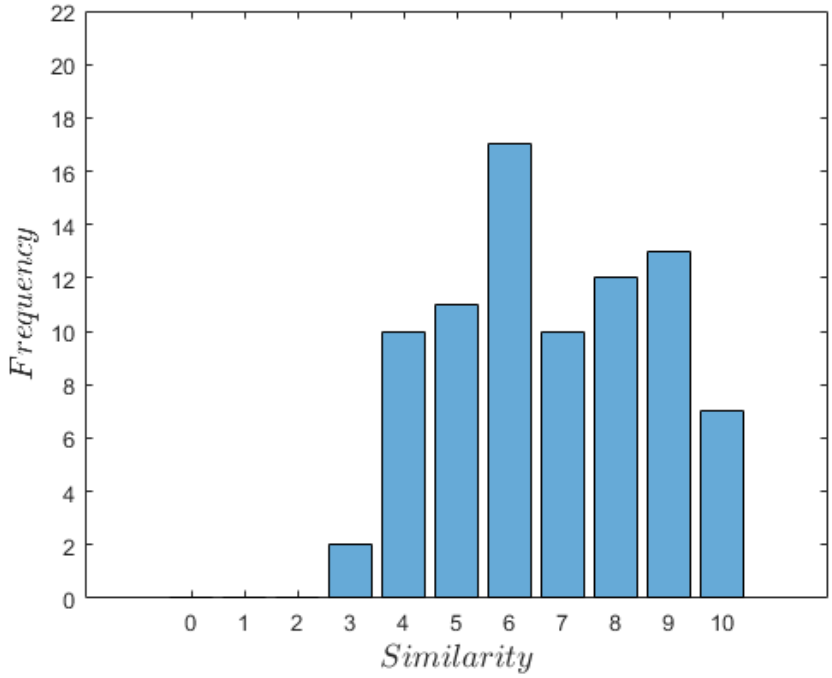
(a)


Frequency
Similarity
0
2
4
6
8
10
12
14
16
18
20
22
0 1 2 3 4 5 6 7 8 9 10

(b)


Frequency
Similarity
0
2
4
6
8
10
12
14
16
18
20
22
0 1 2 3 4 5 6 7 8 9 10

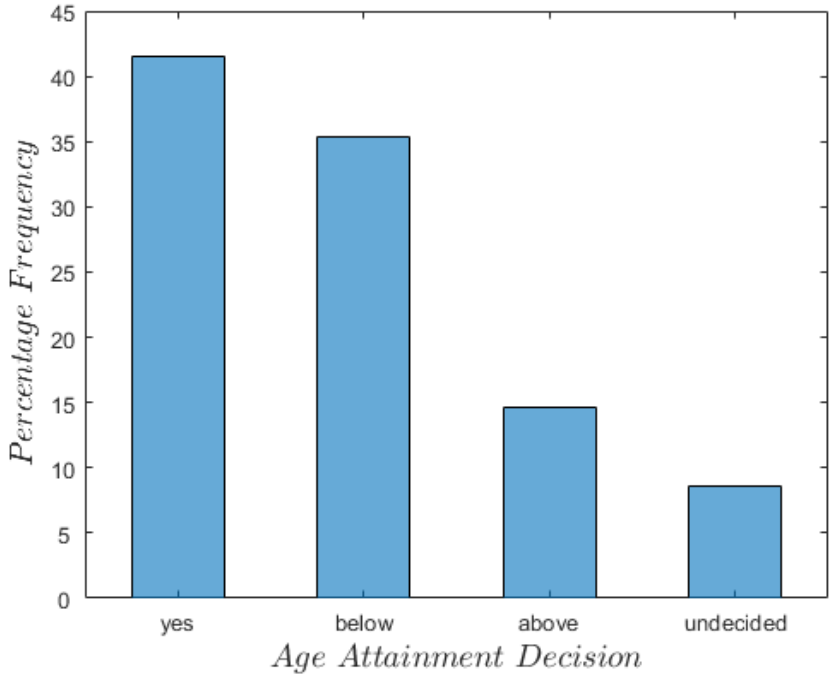
(c)



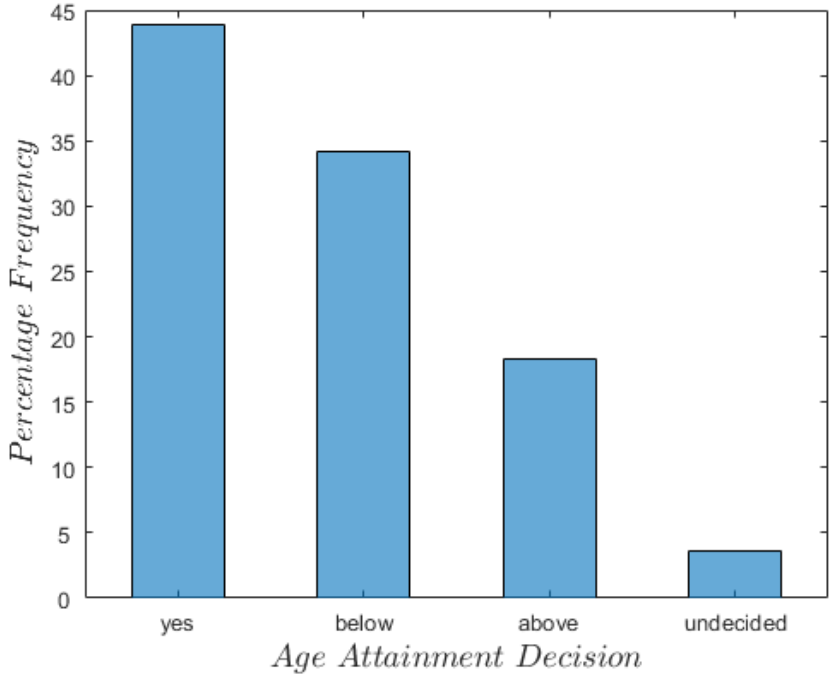
(d)

Figure 3.14: Bar graphs of subject identity scores (AAM based models) (a) Lanitis (b) OLS (c) PLS (d) sPLS.

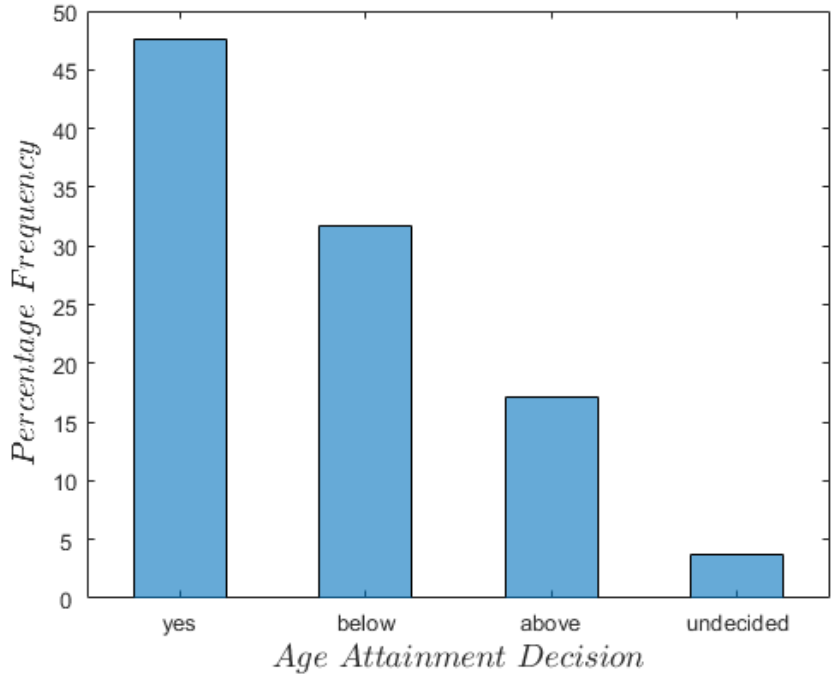
Human-based age attainment test (see Figure 3.15) further confirm's the first findings, 48.8% of sPLS generated faces were perceived to have the expected age, followed closely by 47.6% for PLS, next 43.9% for OLS-faces and finally 41.5% of those generated using Lanitis method.


45
40
35
30
25
20
15
10
5
0
Percentage Frequency
yes
below
above
undecided
Age Attainment Decision

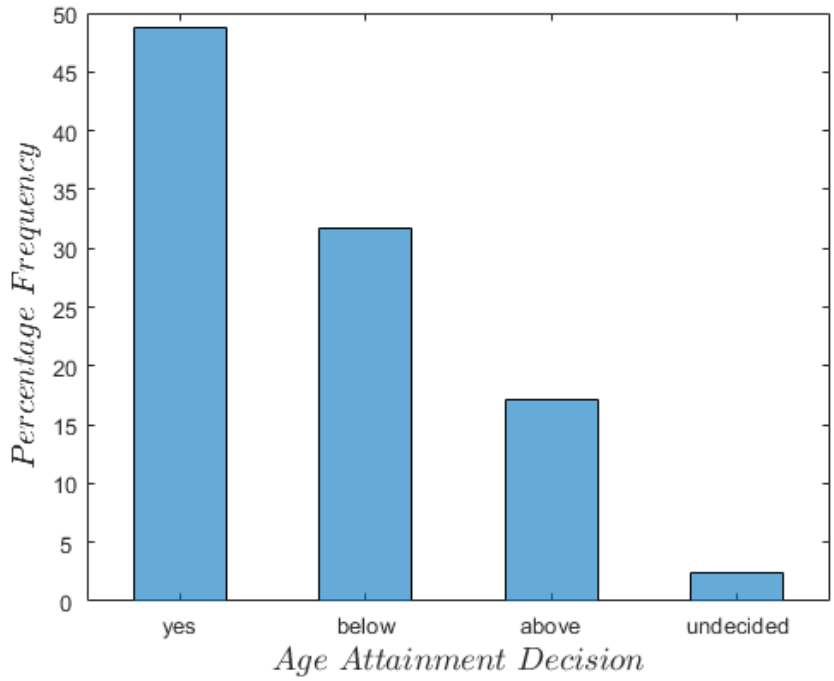
(a)


45
40
35
30
25
20
15
10
5
0
Percentage Frequency
yes
below
above
undecided
Age Attainment Decision

(b)



(c)



(d)

Figure 3.15: Bar graph representation of subjective age attainment (perception) test for AAM based model (a) Lanitis's method (b) OLS approach (c) PLS method (d) sPLS technique.

To sum it up, Lanitis method performed below the proposed techniques mainly due to reconstruction errors that occur as a result of direct subtraction and addition of values from the lookup table. The proposed techniques which use predictive models to render faces have better results. The results further show that predictability of the regression model has significant effect on the synthesis output, hence sPLS regression which has the least prediction error stands out amongst the rest.

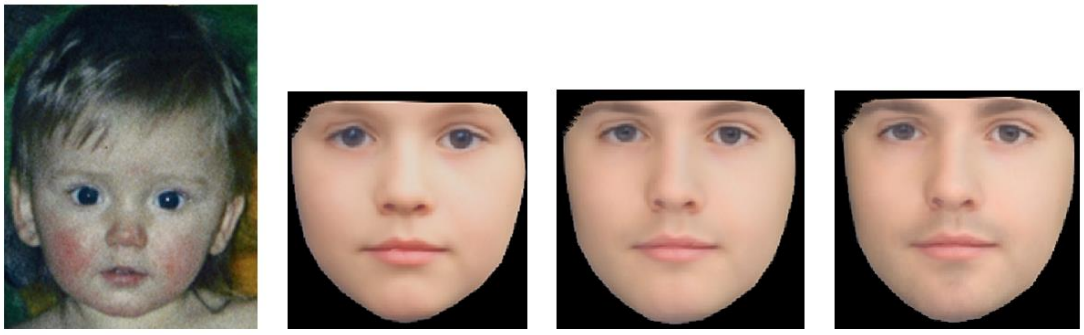
In summary, both human and machine based tests presented above, indicate the ability of the proposed methods to preserve the subject's identity. However, the rendering ability of the algorithms depends on the image quality. It has been observed that facial expressions, poor picture quality, image noise, as well as facial hair hinder the performance of the techniques. Furthermore, despite achieving relatively good construction output on most of the 82 test cases, the ages of a substantial number of the test images were perceived to be below the intended age; this can be attributed to the averaging effect of PCA which results in faded facial texture. In some other cases, the progressed faces looked older than expected due facial hair artifacts that appear on children's faces.

Having achieved best performance using the sPLS-based progression algorithm, it is then ideal to use the proposed ageing framework in a real life application. With the aim of aiding in the search for missing people, the algorithm was then used to synthesise images of Ben Needham [18].

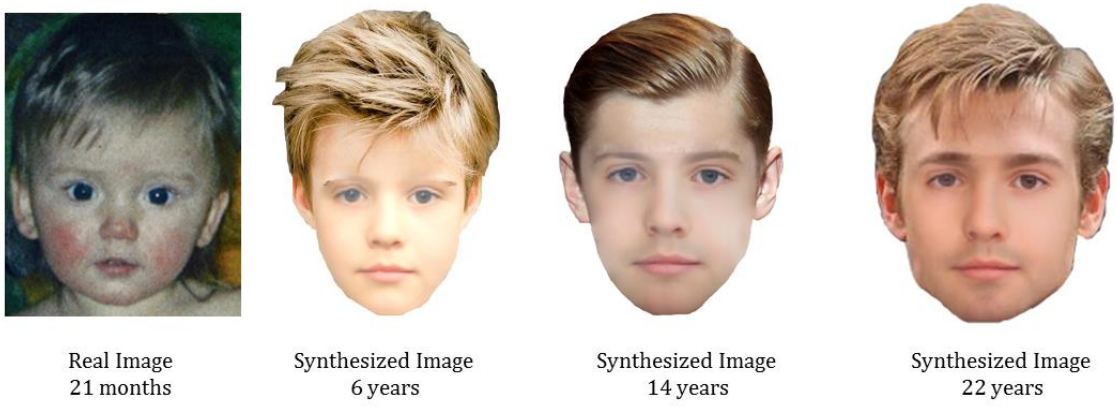
3.6.4 Application

The case of Ben Needham is claimed to be one of the longest missing person's case in British history. Born in Sheffield, on 29th of October 1989, Ben Needham disappeared while on holiday with his parents in the Greek island of Kos on 24th July 1991 [18]. To date, despite numerous false sightings over the years, no trace of the British toddler has ever been found.

In this study, sPLS ageing framework was used to progress the image of Ben Needham to the ages of 6, 14 and 22 years as shown in Figure 3.16(a). By compositing, external features such as hair and ears were then incorporated into the generated images as shown in Figure 3.16(b). Subsequently, the synthesised images were compared to the existing Police generated images as shown in Figure 3.16(c).



(a)



(b)

Figure 3.16: Age progressed images of Ben Needham (a) sPLS-based rendering (b) External features incorporated to improve visualisation (c) Current Police generated images.

Due to variations in hair style and occluded ears, the AAM based model does not consider hair and ears. However, these external facial features have been incorporated into Ben Needham's images as a form of cosmetic to enhance visualisation. However, this compositing approach is highly subjective and might alter the visual look of the real results. In order to fully incorporate these external features into the proposed ageing framework, it will be necessary to thoroughly study and develop further automated methods that can incorporate such external facial features in the future. Visual comparisons were made to the Police generated images to highlight their difference to those generated using the proposed method, hopefully, these new sets of images may help in the search and identification of the missing toddler. Furthermore, this illustrates the applicability of the age progression work to real life cases.

3.7 Summary

In this chapter, a mathematical procedure for progressing facial images was presented. Development of the age progression framework entailed the building of an AAM to extract facial features, after which various linear regression algorithms were used to achieve synthesis. Careful experiments conducted in this chapter have shown that the proposed method performs excellently when used to age neutral, frontal, good quality face images. However, in real world scenarios, one does not always get ideal images that exhibit neutral facial expression, sometimes the pictures happen to be noisy with poor photo quality. Unfortunately, experiments showed that the proposed method is not robust to noisy images, nor is it resilient to varying facial expression. Furthermore, synthesised images have hair artefacts and often produce aged faces that lack facal skin texture detail. Thus, in this work, the aforementioned pitfalls shall be

tackled by first exploring a facial representation technique that handles noise and varying facial expressions. Thereafter, a way of handling texture artefacts will be investigated and addressed.

4 Face Synthesis using Nonlinear Appearance Model

In this chapter a kernel appearance model (KAM) which captures nonlinear shape and texture variations is derived. Facial features are then extracted using the KAM, thereafter, face synthesis is achieved via regression. A thorough evaluation of the proposed algorithm is also conducted.

4.1 Introduction

Although many algorithms have been proposed for age progression [115], one of the most widely used techniques for features extraction is the AAM that was presented in the previous chapter. Whilst AAM's proven ability to model deformable objects, they suffer from some disadvantages [116]. Among these is the fact that principal component analysis, which is at the core of the model, assumes both shape and texture exhibit linear variation. This is not always the case as this approach is sub-optimal when objects lie on a nonlinear manifold in parameter space. Obviously, one cannot proclaim that the conventional PCA detects all structure in a given data set. Most importantly, it cannot tackle higher order noise in data [117], just as was observed in the previous chapter. Hence, PCA is not always robust to noise. Furthermore, the AAM based age progressor as observed is affected by facial expressions; as the face is aged, the expression gets exaggerated thereby distorting the perceived age. To this end, age progression framework that utilizes image de-noising and expression normalizing capabilities of kernel principal component analysis (kernel PCA also KPCA) is proposed.

In this chapter, kernel PCA a nonlinear form of PCA that explores higher order correlations between input variables is used to build a model that captures the

shape and texture variations of the human face. The extracted facial features are then used to perform age progression via a regression procedure, even though computation done in kernel space requires a pre-image calculation to generate a facial image. In a similar manner to the previous chapter, performance of the framework is evaluated through rigorous tests.

In a nutshell, this chapter presents an approach to age progression which improves the framework of chapter 3 by embedding kernel PCA into the model, in order to tackle the problems of image noise, lightening variations, as well as the effect of facial expressions.

4.2 Kernel Machines

Due to their computational efficiency, kernel machines have gained popularity in the last two decades [118], as such, they have been applied to statistical learning theory, signal processing and machine learning in general. The kernel trick, first proposed by Aizeman et al. [119], is a key concept in the development of kernel machines, which has led to the development of nonlinear variants of linear statistical algorithms. This approach is used to map data from the original input space X to a higher dimensional (nonlinear) Hilbert space $\mathcal{H}$ given by $\phi: X \longrightarrow \mathcal{H}$. Interestingly, this mapping to a higher dimension is computed from the dot product of the data [118]. This technique known as the kernel trick does not require explicit calculation. Rather it is achieved by replacing the inner product operator with a symmetric Hermitian “Kernel” function.

One common application of kernel machines is the kernel PCA that was proposed by Schölkopf et al. [120]. Using kernel methods, [120] generalised

PCA into a higher order correlation between input variables. KPCA entails a nonlinear mapping ϕ of data from an input space $\mathbf{x}$ into a feature space F and then the computation of conventional PCA in the F space. As stated earlier, the transformation from $\mathbf{x}$ to a higher dimension is realized using the kernel trick [119].

Consider an intricate data which is intended to be project into a higher dimensional space, with a view to exploring its higher order structure. The mapping to higher dimension can be expressed as,

$$\mathbf{x} \rightarrow \phi(\mathbf{x}) \tag{4.1}$$

where ϕ is the nonlinear mapping function. A kernel $\boldsymbol{K}$ gives us the ability to map $\mathbf{x}$ to ϕ by computation of the dot products expressed as,

$$\kappa(\mathbf{x}_i, \mathbf{x}_j) = \phi(\mathbf{x}_i)\phi(\mathbf{x}_j)^T, \tag{4.2}$$

where $\mathbf{x}_i$ and $\mathbf{x}_j$ refer to the ith and jth elements of $\mathbf{x}$ and T is the transpose operator.

In essence, the mapping creates nonlinear combinations of the input feature. The generalisation of PCA to higher dimensional space [121] involves the computation of the covariance matrix via the kernel trick,

$$\frac{1}{N}\sum_{j=1}^{N} \phi(\mathbf{x}_j)\phi(\mathbf{x}_j)^T \tag{4.3}$$

and subsequently solving the eigenvalue problem,

$$\boldsymbol{K\alpha} = N\lambda\boldsymbol{\alpha}, \tag{4.4}$$

where $\boldsymbol{\alpha} = [\boldsymbol{\alpha}_1, \ldots \boldsymbol{\alpha}_N]$ are set of eigenvectors of $\boldsymbol{K}$, for $\lambda \geq 0$ the set of first k eigenvectors of $\boldsymbol{K}$ can be normalised such that $\boldsymbol{V}^k.\boldsymbol{V}^k = 1$

For the purpose of principal components β_k extraction, the eigenvectors $\boldsymbol{V}^k$ are projected onto the data in F. Assuming $\mathbf{x}$ is a test data with image $\phi(\mathbf{x})$, its projection is given by,

$$\begin{aligned} \beta_k = \phi(\mathbf{x})^T.\boldsymbol{V}^k &= \sum_{i=1}^{N} \alpha_i^k.(\phi(\mathbf{x}).\phi(\mathbf{x}_i)^T) \\ &= \sum_{i=1}^{N} \alpha_i^k.\kappa(\mathbf{x},\mathbf{x}_i). \end{aligned} \tag{4.5}$$

To ensure the mapped data $\phi(\mathbf{x})$ has a zero mean, centring can be achieved by replacing $\boldsymbol{K}$ with a gram matrix $\widetilde{\boldsymbol{K}}$,

$$\widetilde{\boldsymbol{K}} = \boldsymbol{K} - \mathbf{1}_N\boldsymbol{K} - \boldsymbol{K}\mathbf{1}_N + \mathbf{1}_N\boldsymbol{K}\mathbf{1}_N \tag{4.6}$$

where $\mathbf{1}_N = \boldsymbol{I}_N - \boldsymbol{J}_N$, $\boldsymbol{I}_N$ is the identity matrix and $\boldsymbol{J}_N$ is an $N \times N$ matrix whose elements are all 1s.

KPCA has been used in a variety of computer vision applications and in most cases has proven to outperform the conventional PCA [122]. Furthermore, several research works have shown the power of Kernel PCA in conducting image denoising [117], [118], [123], illumination normalization, occlusion recovery, as well as facial expression normalization [124]; this is usually

achieved by computing an inverse map from the higher dimensional space back to the input space.

In this chapter, the problem of age progression is addressed by developing an appearance model based on KPCA, thereby, taking benefit from higher dimensional projections. Consequently, this approach will give us the ability to achieve image denoising and expression correction, i.e. by solving the *KPCA-preImage* problem [125]. It shall be shown that this new technique termed Kernel Appearance Model (KAM) is better suited for the problem of age progression than the conventional AAM.

4.3 Kernel Appearance Model (KAM)

The proposed KAM captures nonlinear shape and texture variability from the training dataset. In a similar manner to the linear model presented in chapter 3, face shapes are represented by a set of annotated landmarks given by a 2-dimensional vector expressed using (3.1). As usual, all 2-dimensional vectors representing the face shape are aligned using GPA. Next, a nonlinear shape model is built by performing KPCA; $\mathbf{x}$ is mapped to the higher dimensional feature space using a kernel method $\boldsymbol{K}_x$.

The mapped data is then centralised using (4.6), subsequently, eigen-decomposition is performed using,

$$\widetilde{\boldsymbol{K}}_x \boldsymbol{\alpha}_x = N\lambda_x \boldsymbol{\alpha}_x, \tag{4.7}$$

where $\boldsymbol{\alpha}_x$ is the set of eigenvectors and λ_x their corresponding eigenvalues. Consequently, the shape of each face in the training set can be defined by the

projection operation defined in equation (4.5), thus shape parameters obtained through the nonlinear model can be expressed as,

$$\boldsymbol{\rho}_x = \sum_{i=1}^{N} \alpha_{xi}^{k}.\ \kappa(\mathbf{x}, \mathbf{x}_i)\ . \tag{4.8}$$

To build a nonlinear texture model, all face images are affine warped to a template shape, this is done to remove unnecessary face size variations just as described in chapter 3. Then, illumination effects are normalised by applying a scaling and an offset to the image pixels $\mathbf{g}$ using equations (3.7) and (3.8). Subsequently, KPCA is used to model the nonlinear variations of $\mathbf{g}$, this involves repeating the procedure that was used to model the nonlinear shape variations. Hence, the texture of each face in the training set can be defined by $\boldsymbol{\rho}_g$ obtained via,

$$\widetilde{\boldsymbol{K}}_g \boldsymbol{\alpha}_g = N\lambda_g \boldsymbol{\alpha}_g, \tag{4.9}$$

$$\boldsymbol{\rho}_g = \sum_{i=1}^{N} \alpha_{gi}^{k}.\ \kappa(\boldsymbol{g}, \boldsymbol{g}_i), \tag{4.10}$$

Next, a combined appearance model is built by capturing both shape and texture information,

$$\boldsymbol{\rho}_a = \left(\boldsymbol{\vartheta}\boldsymbol{\rho}_x\ ,\ \boldsymbol{\rho}_g\right)^T. \tag{4.11}$$

where $\boldsymbol{\vartheta} = \sum \lambda_g / \sum \lambda_x$ is a diagonal matrix of weights used to compensate for the difference in magnitude between the units of shape and texture models. Thereafter, conventional PCA is used to reduce the dimension of the combined

model. Hence, a single model which defines the nonlinear shape and texture variations is formed via,

$$\boldsymbol{\rho}_a = \boldsymbol{P}\boldsymbol{f}\,. \tag{4.12}$$

The parameter $\boldsymbol{f}$ now captures both variations, thus it can be used to manipulate the appearance of an individual face in the higher dimensional space. $\boldsymbol{P}$ is a matrix of eigenvectors associated with both shape and texture. Interestingly, the linear nature of equation (4.12) makes it possible to use $\boldsymbol{f}$ for reconstructing the shape and texture variations,

$$\boldsymbol{\rho}_x = \boldsymbol{\vartheta}^{-1}\boldsymbol{P}_{kx}\boldsymbol{f}, \qquad \boldsymbol{\rho}_g = \boldsymbol{P}_{kg}\boldsymbol{f}\,, \tag{4.13}$$

where $\boldsymbol{P}_{kx}$ and $\boldsymbol{P}_{kg}$ are orthogonal modes of variation associated with the shape and texture, $\boldsymbol{P} = \left(\boldsymbol{P}_{kx}\,, \boldsymbol{P}_{kg}\right)^T$.

Equation (4.12) is the core of the KAM model and it represents the nonlinear variant of the AAM defined by equation (3.13). The parameter $\boldsymbol{f}$ encodes the shape and texture of the face, thereby giving us an avenue to defining non-linear shape and texture variation. It can, therefore be used to extract facial features which are then used for age progression. The age progression framework described in the next section leverages the nonlinear characteristics of the face feature $\boldsymbol{f}$.

4.4 Nonlinear Framework for Age Progression

Having built the KAM, it now represents a nonlinear variant of AAM that can be integrated into the age progression framework in a similar passion to that described in 3.4. This involves defining an ageing function $age = g(\boldsymbol{f})$ that maps the nonlinear face features to the individual age. Using same rationale described in previous 3.4, a linear equation can be used to describe this mapping. Since this linear mapping is implemented in a higher (nonlinear) dimensional space, it is in fact not linear, rather it efficiently captures nonlinear variations, that will have been missed by the simplistic AAM approach. The expression is given by,

$$age = \boldsymbol{\delta}^T \boldsymbol{f} \tag{4.14}$$

$$\text{subject to } age_i = g(\boldsymbol{f}_i)$$

where $\boldsymbol{\delta}$ is a vector of regression coefficients in the F feature space and i is the index of individual whose face is to be progressed.

Inverting equation (4.14) is a key part of the age progression framework, hence the procedure described in 3.4 is repeated. At current age t_{now}, an individual's face features are decomposed into age and identity components,

$$\boldsymbol{f}_{individual\_now} = \hat{\boldsymbol{f}}_{age\_now} + \boldsymbol{f}_{id}, \tag{4.15}$$

Moore-Penrose pseudoinverse † is then used to compute the age component $\hat{\boldsymbol{f}}_{age\_new}$ for the new age,

$$\hat{\boldsymbol{f}}_{age\_new} = \boldsymbol{\delta}^{\dagger} age_{tnew}. \quad (4.16)$$

Subsequently, new age progressed face features are computed by aggregating the new age and identity components,

$$\boldsymbol{f}_{individual\_new} = \hat{\boldsymbol{f}}_{age\_new} + \boldsymbol{f}_{id}. \quad (4.17)$$

After successful computation of a new appearance parameter $\boldsymbol{f}_{individual\_new}$, the corresponding face shape and texture can be extracted using equation (4.13); the nonlinear shape and texture parameters at the new age $\rho_{\mathrm{x\_new}}$ and $\boldsymbol{\rho}_{g\_new}$.

Since the extracted facial features are in higher dimensional space, a new face cannot be reconstructed until after finding a reverse map. That is the computation of aged shape and texture parameters $\mathbf{x}_{new}$ and $\boldsymbol{g}_{new}$ in the input space.

The problem of mapping back to initial input space is indeed an ill posed one known as the *KPCA preImage problem* [118], [126]. It is ill posed due to the size and dimension of the F space. Also, a reverse map may not exist, and even when it does exist, it may not be unique. To be precise, only a few features in *F* have a preimage in $\mathbf{x}$ [118]. The *preImage problem* hence involves procedures for finding an approximate inverse map of the feature space in the $\mathbf{x}$ space as shown in Figure 4.1.

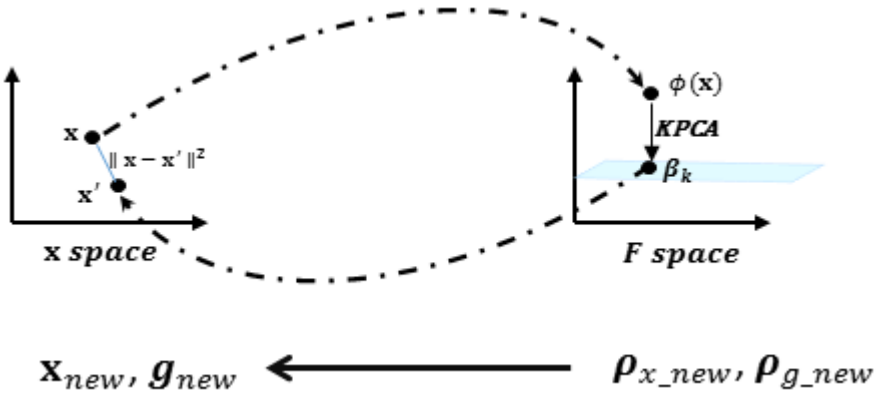


Figure 4.1 The pre-image (inverse) mapping

Given a projection β_k, in F space, its reconstruction in the feature space can be achieved via a projection operator P_n given by,

$$P_n\phi(\mathbf{x}) = \sum_{k=1}^{n} \beta_k . \boldsymbol{V}^k \,, \tag{4.18}$$

where $\boldsymbol{V}^k$ is a matrix of normalised eigenvectors of kernel $\boldsymbol{K}$. When the vector $P_n\phi(\mathbf{x})$ has no pre-image $\mathbf{x}'$ in the input space, solving the *preImage problem* will be that of approximating $\mathbf{x}'$ via minimising,

$$\rho(\mathbf{x}') = \| \phi(\mathbf{x}') - P_n\phi(\mathbf{x}) \|^2 \tag{4.19}$$

where $\| \|^2$ is the L2-norm.

By discarding terms that are independent of $\mathbf{x}'$, this results to,

$$\rho(\mathbf{x}') = \| \phi(\mathbf{x}') \|^2 - 2(\phi(\mathbf{x}') . P_n\phi(\mathbf{x})). \tag{4.20}$$

Using the kernel trick and substituting (4.18) into (4.20), one gets,

$$\rho(\mathbf{x}') = \kappa(\mathbf{x}', \mathbf{x}') - 2\sum_{k=1}^{n} \beta_k \sum_{i=1}^{l} \alpha_i^k . \kappa(\mathbf{x}', \mathbf{x}_i) \tag{4.21}$$

This can be simplified as,

$$\rho(\mathbf{x}') = \kappa(\mathbf{x}', \mathbf{x}') - 2\sum_{i=1}^{l} \gamma_i . \kappa(\mathbf{x}', \mathbf{x}_i), \tag{4.22}$$

where $\gamma_i = \sum_{k=1}^{n} \beta_k \alpha_i^k$.

Several methods have been proposed for solving the optimization problem in [123] Honeine and Richard used a linear algorithm that learns the inverse map from the training set. Kwok and Tsang [125] proposed using multidimensional scaling (MDS) to embed $P_n\phi(\mathbf{x})$ in the lower dimension $\mathbf{x}$; the algorithm seeks to minimise the pairwise distances in input and feature space. However, the most popular approach is the fixed point iteration method proposed by Mika et al. [117], which is guaranteed to produce a pre-image that lies within the span of the training data. However, the method in [117] suffers from a number of drawbacks which include; sensitivity to initialization, local minima, and numerical instability. To overcome the first setback, re-initialization with different values has been proposed in [127], furthermore, these same authors tackled the problem of instability via the use of input space regularization to stop the denominator of the iteration rule from going to zero. The reformulation proposed in [127] is given by,

$$\rho(\mathbf{x}') = \| \phi(\mathbf{x}') - P_n\phi(\mathbf{x}) \|^2 + \zeta \| \mathbf{x}' - \mathbf{x}_o \|, \tag{4.23}$$

where ζ is a non-negative regularization parameter, $\mathbf{x}_o$ is an input space value used for the regularization.

With a view to enhancing image reconstruction, centring of the weighting coefficient γ_i has also been suggested in the literature [125], [127], [128]. So that,

$$\tilde{\gamma}_i = \gamma_i + \left(\frac{1}{N}\right)\left(1 - \sum_{j=1}^{N} \gamma_j\right). \tag{4.24}$$

where N is the total number of observations.

In this work, the pre-images of $\boldsymbol{\rho}_{x\_new}$ and $\boldsymbol{\rho}_{g\_new}$ are computed by incorporating (4.24) into the optimization problem of (4.23). Finally, after computing the pre-images of $\rho_{\text{x\_new}}$ and $\boldsymbol{\rho}_{g\_new}$, the resulting progressed face is rendered by warping the new texture $\boldsymbol{g}_{new}$ to the new shape $\mathbf{x}_{new}$. In essence, the computation of the inverse map using a technique that minimises the distance from training space results in facial expression normalisation. More so, the higher order projection effectively separates noise from the signal. The nonlinear age progression framework can be summarised pictorially using Figure 4.2.

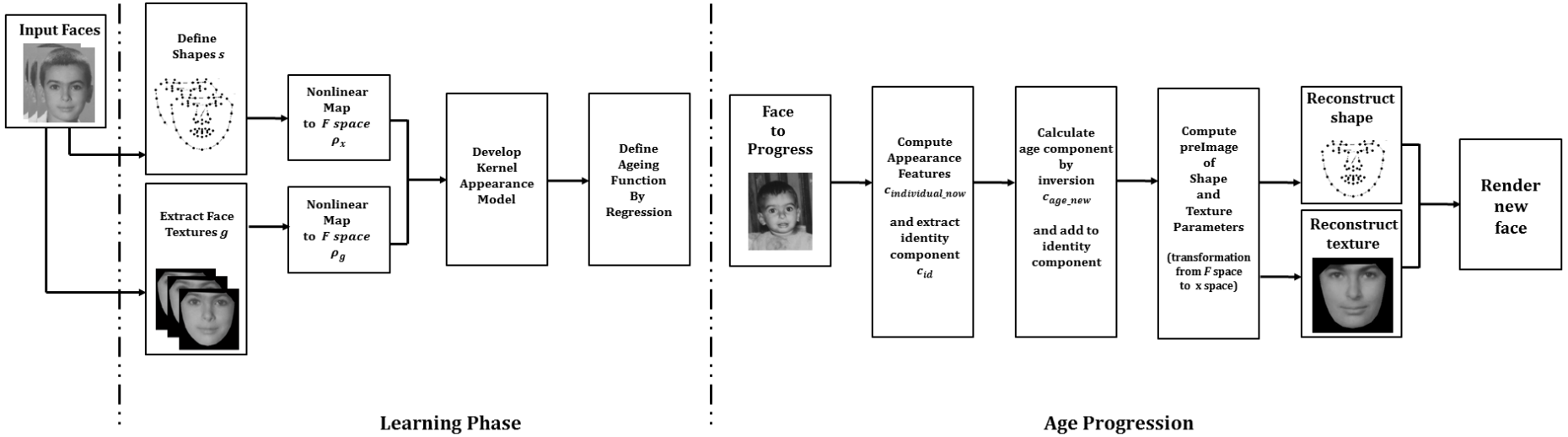


Figure 4.2: Nonlinear Age Progression Framework.

4.5 Experiments

All experiments conducted with a view to evaluating the KAM age progressor follow same dataset usage protocol outlined in 3.6.1. Furthermore, the performance evaluation technique is also as outlined in 3.6.2; both human and machine based evaluation procedures are deployed.

4.5.1 Choice of Kernels

Kernels used in machine learning literature have been categorised into two broad classes; isotropic and projective kernels [118]. Isotropic also known as radial kernels are functions of distances; the most popular of this kind is the Gaussian kernel. On the other hand, projective kernels are functions of inner dot products. Here Gaussian, Sigmoid and Log kernels are considered. Gaussian kernels have been the de facto in computer vision; the other two functions are examined as a form of comparison. Specifically, Sigmoid is included due to its origin from neural networks [129]. Log kernels, which were derived from power kernels, are included as they have been shown to outperform Gaussian kernels when applied to the problem of image recognition [130]. The equations for the kernels and their pre-image rule computed using equations (4.23) and (4.24) are presented in Tables Table 4.1 and Table 4.2 respectively.

Table 4.1: Reproducing kernels used for KAM age progression experiments.

Kernels	**Type**	**Expression**	**Parameter Condition**
Gaussian	Radial	$\exp(\frac{-\parallel x_i - x_j \parallel^2}{2\sigma^2})$	$\sigma > 0$
Sigmoid	Projective	$tanh(a(x_i^T \cdot x_j) + r)$	$a > 0, \quad r < 0$
Log	Radial	$-\log(\parallel x_i - x_j \parallel^\beta + 1)$	$0 < \beta \leq 2$

Table 4.2: Kernel pre-image iteration rules.

Kernels	**Gradient of the Optimization Equation**
Gaussian	$\frac{1}{\sigma^2}\sum_{i=1}^{l}\gamma_i \exp(\frac{-\parallel x_* - x_i \parallel^2}{2\sigma^2})(x_* - x_i) + \zeta(x_* - x_o)$
Sigmoid	$ax_*(1 - \tan^2 h(ax_*^T x_* + r)) - \sum_{i=1}^{l}\gamma_i a(1 - \tan^2 h(ax_*^T x_i + r))x_i + \zeta(x_* - x_o)$
Log	$\sum_{i=1}^{l}\gamma_i \frac{\beta}{(\parallel x_* - x_i \parallel^\beta + 1)}(x_* - x_i)^{\beta-1} + \zeta(x_* - x_o)$

4.5.2 Parameter Selection

While, there exist several methods for selecting an optimal kernel for supervised learning problems [131], [132], the lack of proper evaluation criteria makes parameters selection in unsupervised learning problematic. As a consequence, the choice of kernel parameters will be in accordance with techniques that have been used in the literature.

To ensure a gaussian kernel width σ remains small enough while capturing optimum neighbourhood information of each image pixel, the width σ is selected using the formula defined in [133], expressed as,

$$\frac{5}{l}\sum_{i=1}^{l} d_i^{NN} \tag{4.25}$$

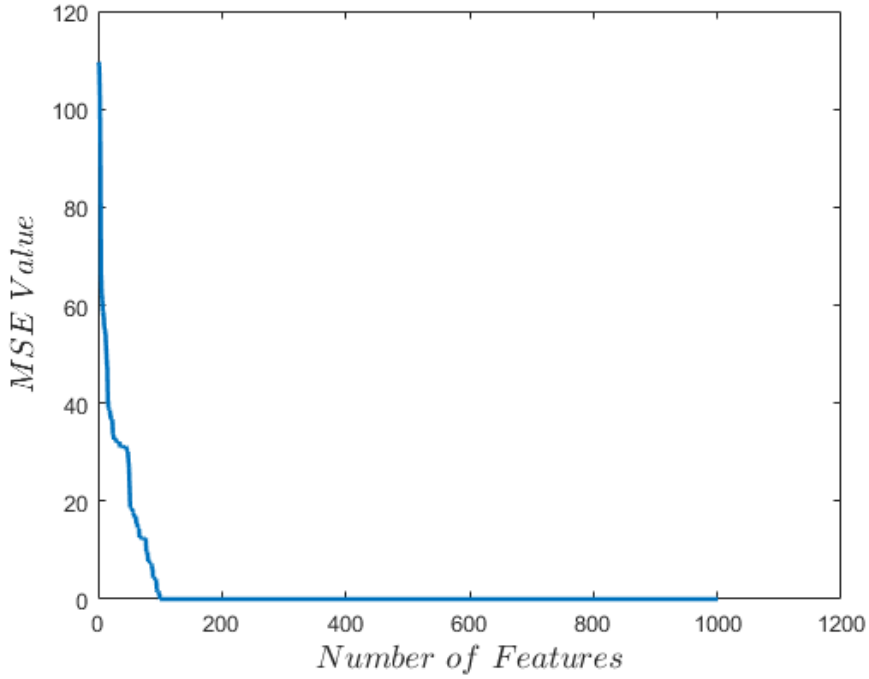
where d_i^{NN} is the distance between image pixel x_i to its nearest neighbour, and l is a total number of pixels. For the Sigmoid kernel, the optimum value of the gradient α has been shown in [129] to be $1/N$ where N is the data dimension. To meet the condition of positive-definiteness [129], throughout the experiments the value of the negative intercept is fixed to be $r = -1$. In the literature, it has been shown that the degree of a log kernel must lie within the range of $0 < \beta \leq 2$, in this work a value of $\beta = 2$ is used as it guarantees a differentiable cost function during the pre-image computation.

In order to ensure convergence of pre-image objective functions, the initial iteration parameter $\mathbf{x}_o$ is always set to the mean $\left(\frac{1}{N}\sum \mathbf{x}\right)$ of training data [133], on the other hand, the non-negative regularization parameter is set to a minimum value of $\zeta = 0.0001$ [127].

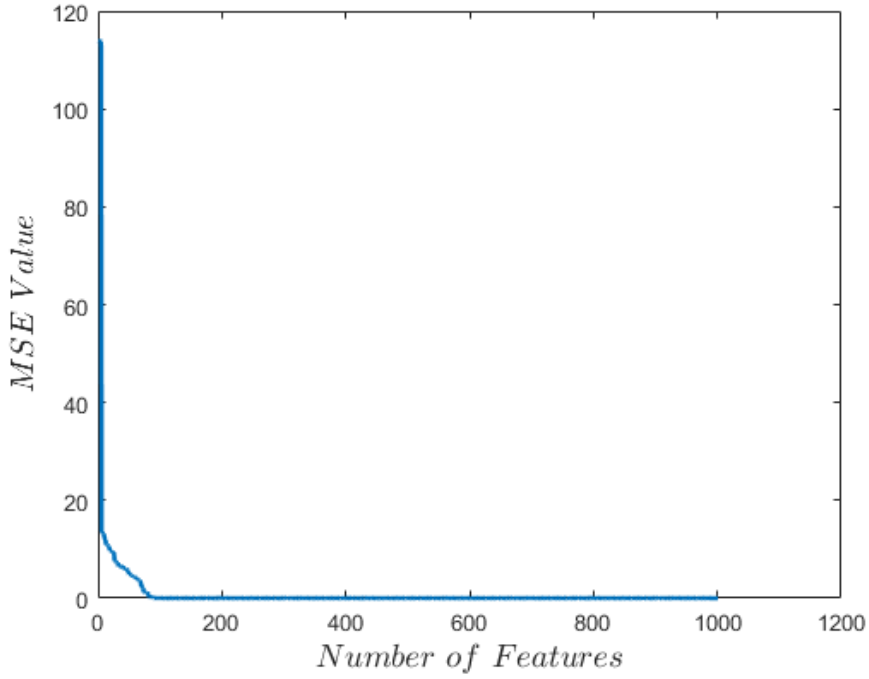
4.5.3 Results

Various experiments were conducted just as outlined in 3.6.3. Here, KAM was built using all three kernels mentioned above. To achieve age progression, the linear regressor defined in (4.14) was utilised. For all three KAMs, the number

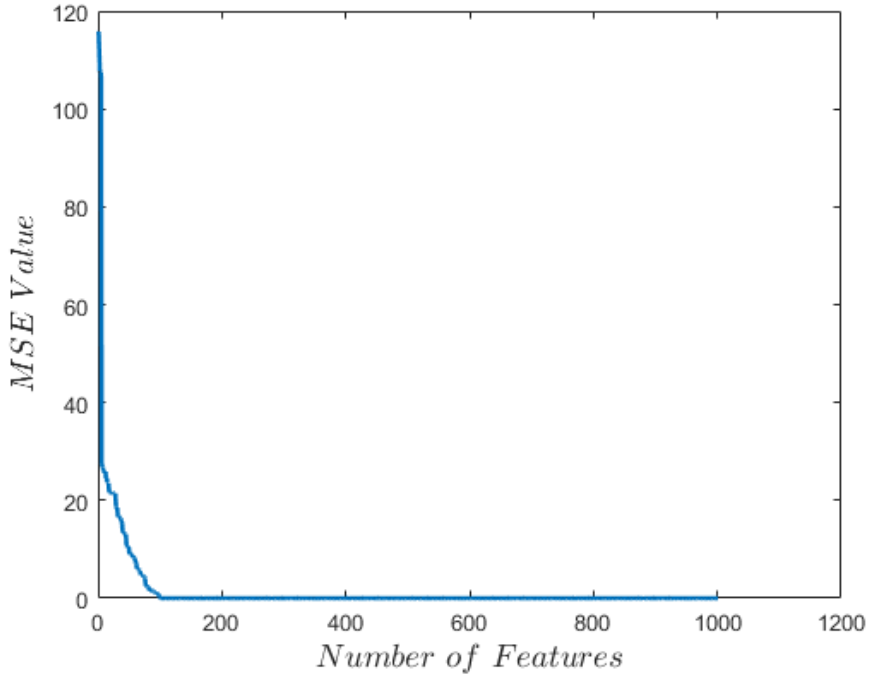
of features plugged into the regression model was chosen by considering the dimension that resulted to the least MSE as shown in Figure 4.3. Interestingly, it was observed that all three KAMs required 100 features to achieve optimum MSE; this was attained via cross validation.



(a)



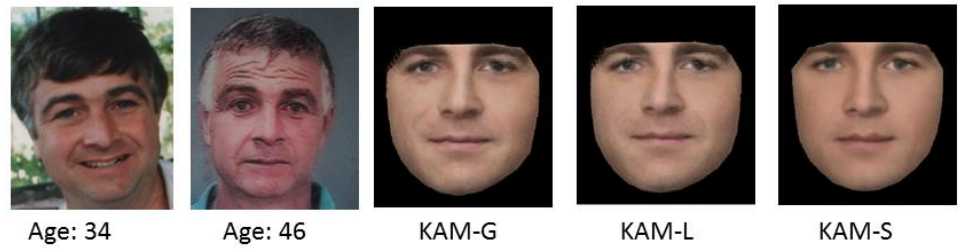
(b)



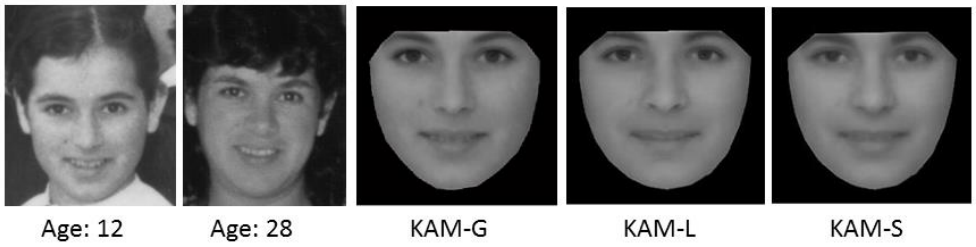
(c)

Figure 4.3: Mean square error per number of features (a) Gaussian kernel KAM (b) Log kernel KAM(c) Sigmoid kernel KAM.

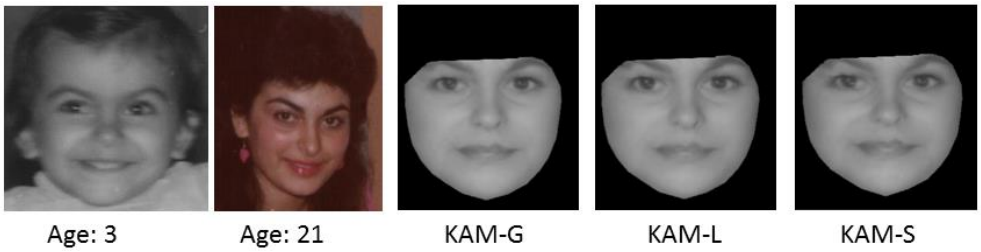
The 82-subjects' test data was then used to perform age progression. In Figure 4.4, a sample of the age synthesis results show the test image at the farthest left, next to it is the ground truth picture of the subject at intended age, and then follows the synthesis results generated using Gaussian (KAM-G), Log (KAM-L) and Sigmoid (KAM-S) kernels.



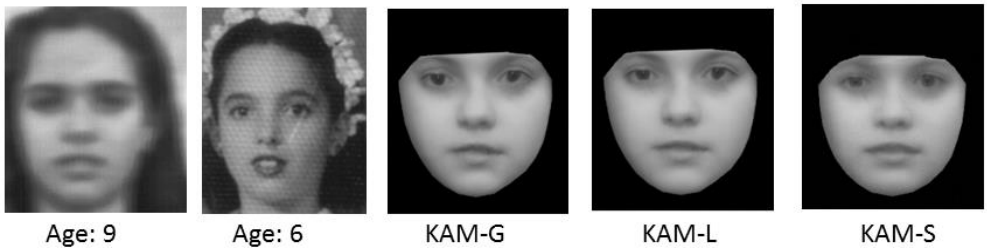
(a)



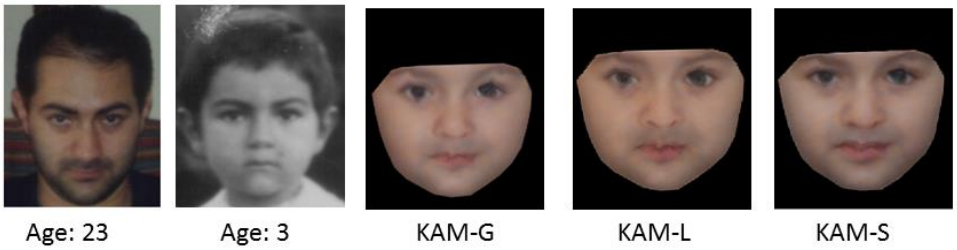
(b)



(c)



(d)



(e)

Figure 4.4: Sample of KAM age synthesis results. Images on the farthest left are the test images.

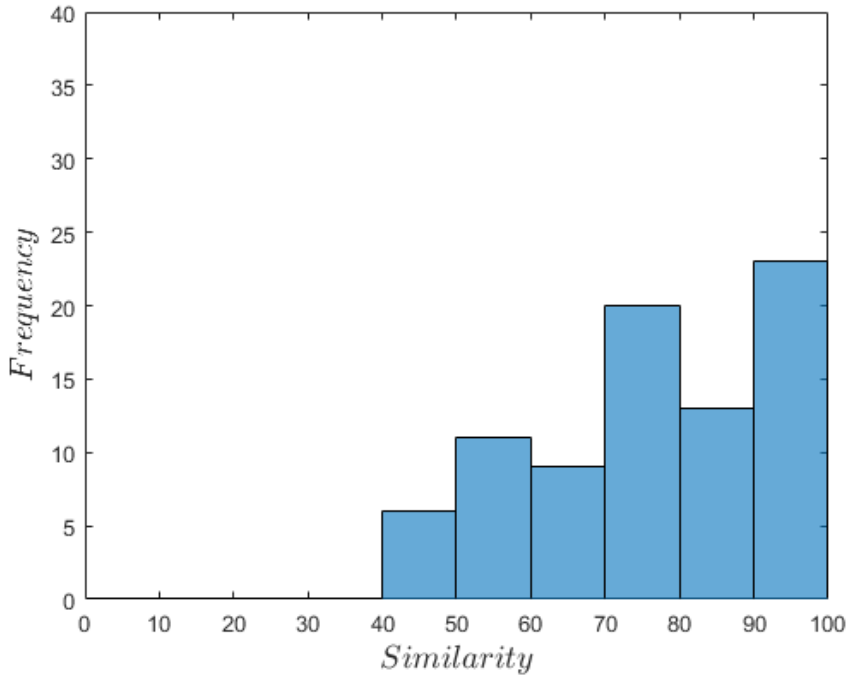
From the results above, it is quite obvious that the KAM framework handles noise and facial expression with a certain degree of visual accuracy. Furthermore, it can be observed that the distance based kernels produce better images than the projective (Sigmoid) kernel; this can be attributed to the ability of distance kernels to encode spatial relation of image pixels. Observing Figures 4.4 (a) and (e), it is also obvious that the KAM technique also suffers from facial hair artefacts, as well as the low texture resolution phenomenon.

In order to evaluate performance of the proposed algorithm, and to compare the efficacy of the three kernels, both objective and subjective evaluations were conducted. Mean scores for the objective tests (see

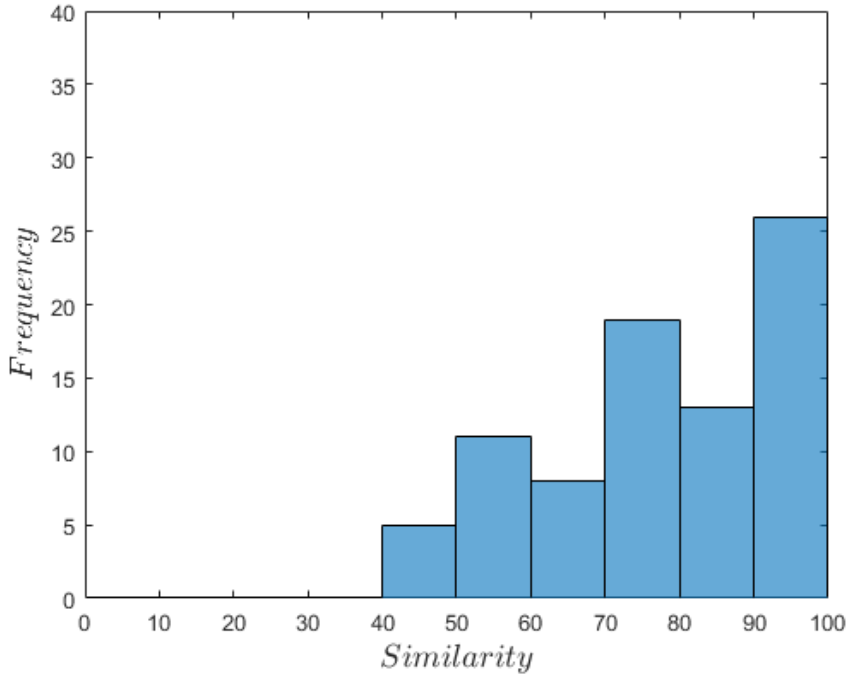
Table 4.3) show the superiority of the KAM age progression framework; having average identity scores of between 77% and 79%, all three kernels outperform the AAM based model presented in chapter 3. To further check if the improvements are statistically different, again two-sample KS test was used to compare the mean scores. It was observed that at 5% significance level, the KAM-G (D=0.225, p=0.0305) based algorithm showed significant improvement over the sPLS algorithm that proved to be the best of the AAM methods. However no significant difference was observed when compared to the other two kernel techniques, also KAM-S (D=0.1585, p=0.2325) and KAM-L (D=0.1951, p=0.065) did not show significant statistical improvement over the sPLS method.The results clearly show the superiority of the Gaussian kernel function. This can be better visualised by observing the histograms shown in Figure 4.5.

Table 4.3: Mean scores of objective test (KAM based models).

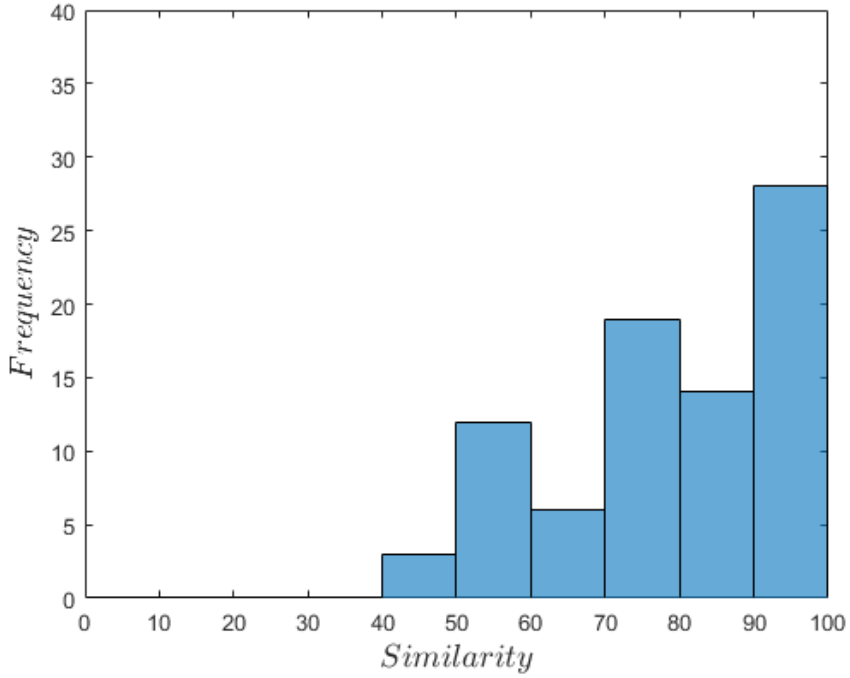
Technique	**Mean Scores (%)**
KAM-S	77.03%
KAM-L	78.19%
KAM-G	79.34%



(a)



(b)



(c)

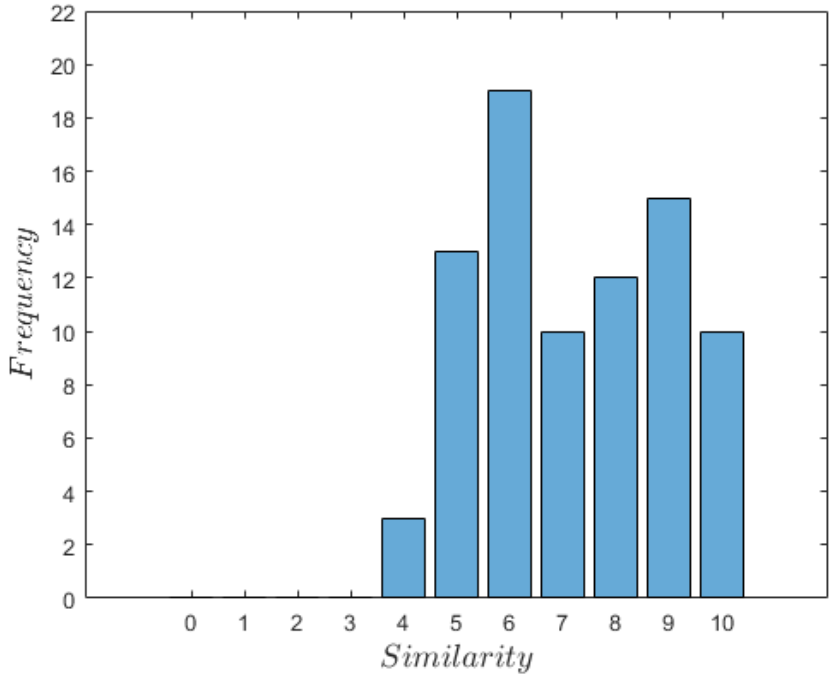
Figure 4.5: Histogram of objective test scores (KAM based models) (a) KAM-S (b) KAM-L (c) KAM-G.

Subjective test conducted to identify human perception of how well the algorithm retains identity further shows the excellence of the proposed algorithm. As a matter of fact, the mean scores of the subjective test presented in Table 4.4 results corroborate the findings of the objective test.

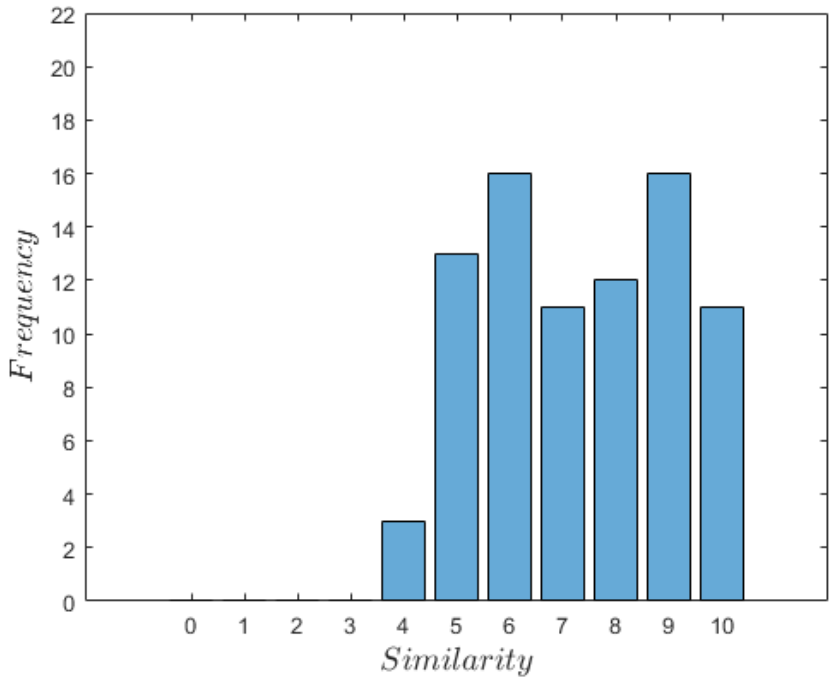
Table 4.4: Mean scores of subjective test (KAM based model).

Technique	**Mean Scores**
KAM-S	7.2195
KAM-L	7.3171
KAM-G	7.4512

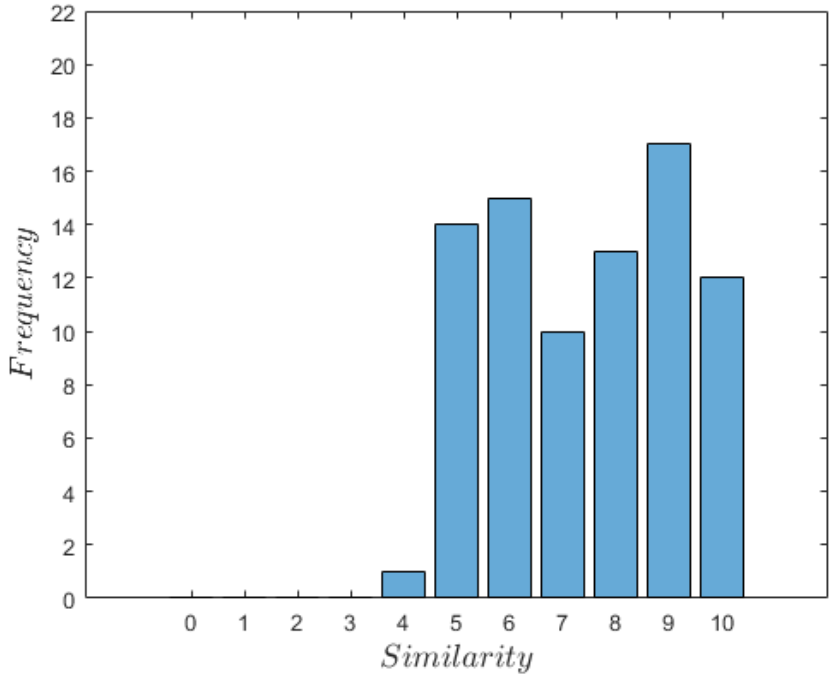
Among the three kernel methods, images generated using KAM-G had the highest mean score, followed by KAM-L, and then KAM-S. When compared to the result of the model presented in the previous chapter, the mean score of KAM-S, despite being considerably lower than the other two kernel methods, outperforms the best AAM-based algorithm. However two-sample KS test did not reveal statistical difference between KAM-S ($D=0.1741$, $p=0.124$), KAM-L ($D=0.1907$, $p=0.076$) and the sPLS technique. This indicates that although the mean scores shows improvement, statistically study of the distribution does not reveal massive improvement. As for KAM-G ($D=0.2305$, $p=0.027$) scores, when compared to sPLS scores via the KS test a significant improvement is observed. To gain more insight into the overall performance of the models, bar graphs have been used to plot the response of human observers. One can clearly see that the KAM-G has more bins closer towards the best score (i.e. 10), while the other two implementations have more bins down the scale.


Frequency
Similarity
0
2
4
6
8
10
12
14
16
18
20
22
0 1 2 3 4 5 6 7 8 9 10

(a)


Frequency
Similarity
0
2
4
6
8
10
12
14
16
18
20
22
0 1 2 3 4 5 6 7 8 9 10

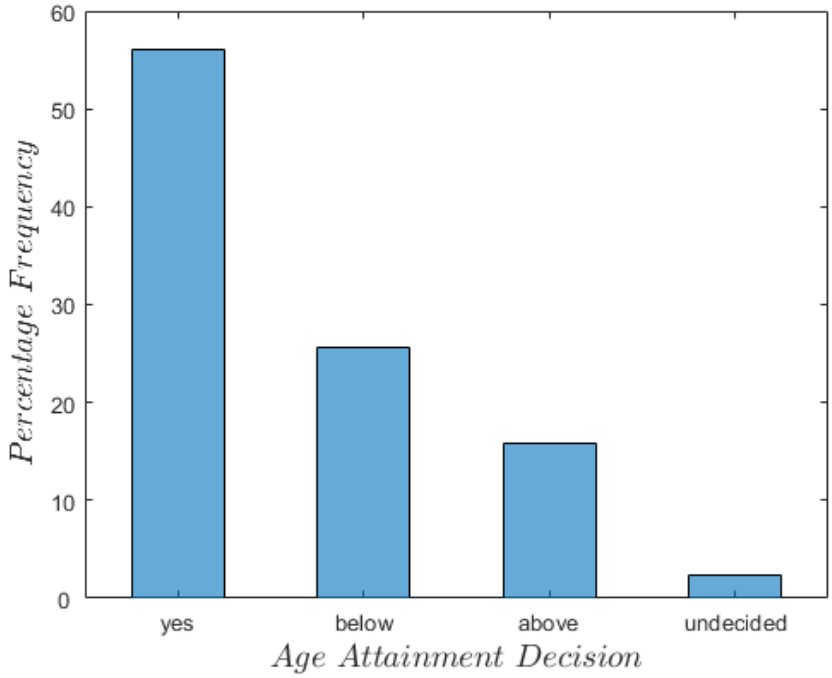
(b)



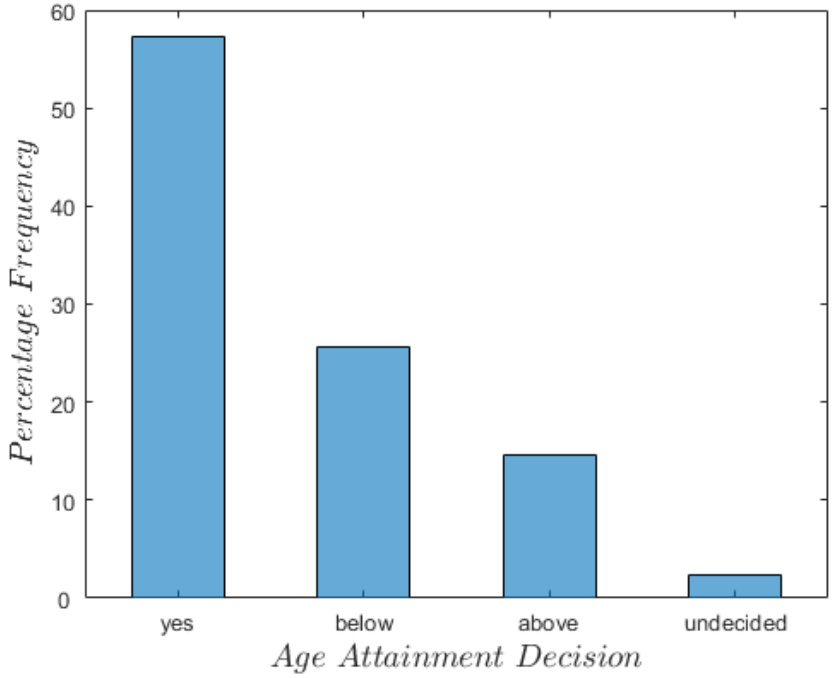
(c)

Figure 4.6: Bar graphs of subject identity scores (KAM based models).

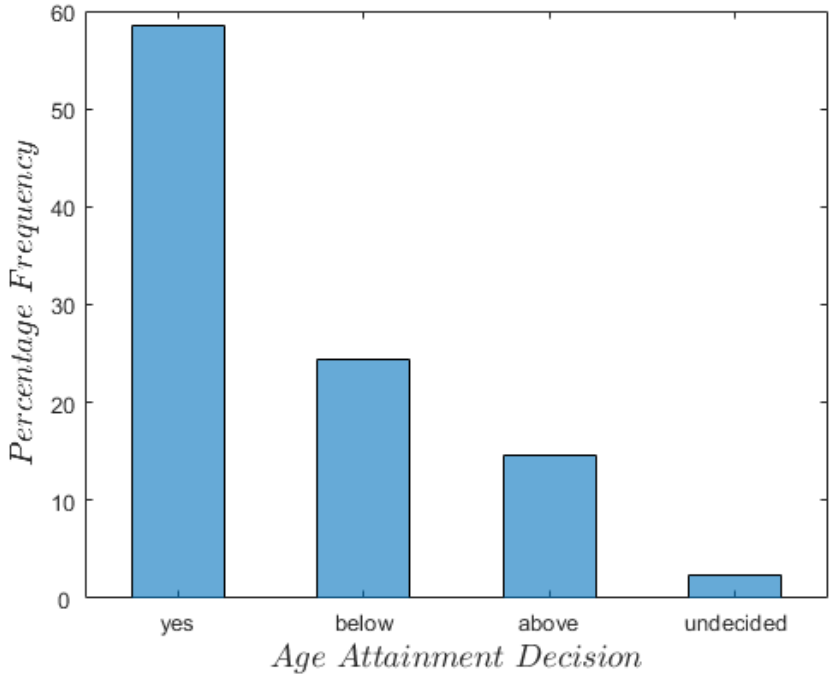
However, the subjective age attainment test (see Figure 4.7), despite being better than what was reported for the AAM framework, shows that a fair bit of the KAM synthesised images are perceived to look either younger or older than the age. Specifically between 56% and ≈59% of the images were perceived to attain the intended age. Others were either thought to look younger, older or in rare cases no judgement was made. It is believed that most images were perceived to look younger than expected due to the low-resolution phenomenon of PCA, on the other hand, older looking faces were obtained due to the facial hair artefacts that appear on young faces.


60
50
40
30
20
10
0
Percentage Frequency
yes
below
above
undecided
Age Attainment Decision

(a)


60
50
40
30
20
10
0
Percentage Frequency
yes
below
above
undecided
Age Attainment Decision

(b)



(c)

Figure 4.7: Bar graph representation of subjective age attainment (perception) test for nonlinear models (a) KAM-S (b) KAM-L (C) KAM-G.

The proposed KAM based framework shows promising results and clearly, outperforms the AAM based approach. However, the method still has a few setbacks. Faces that are progressed backward at times show signs of stubble (Figure 4.3b) due to facial hair artefact, and most obviously adult generated faces lack sufficient amount of texture detail such as wrinkles, thus the perceived age at times seems below the intended age.

4.5.4 Application

Having investigated, and observed the performance of the proposed approach, the best performing implementation of the nonlinear ageing framework (i.e. KAM-G) was used in progressing image in a real world missing person scenario. Here experiments were conducted using the images of Mary Boyle

[19]. In Ireland, the case of Mary Boyle is considered one of the longest missing persons' case. The six-year-old Irish girl went missing from her grandparent's farm near Ballyshannon, County Donegal, Ireland in March of 1977. The Police have since closed the case, however, the fact that no trace has been found has left her family most especially her twin sister (Ann Doherty) asking questions, and hoping she will be found someday.

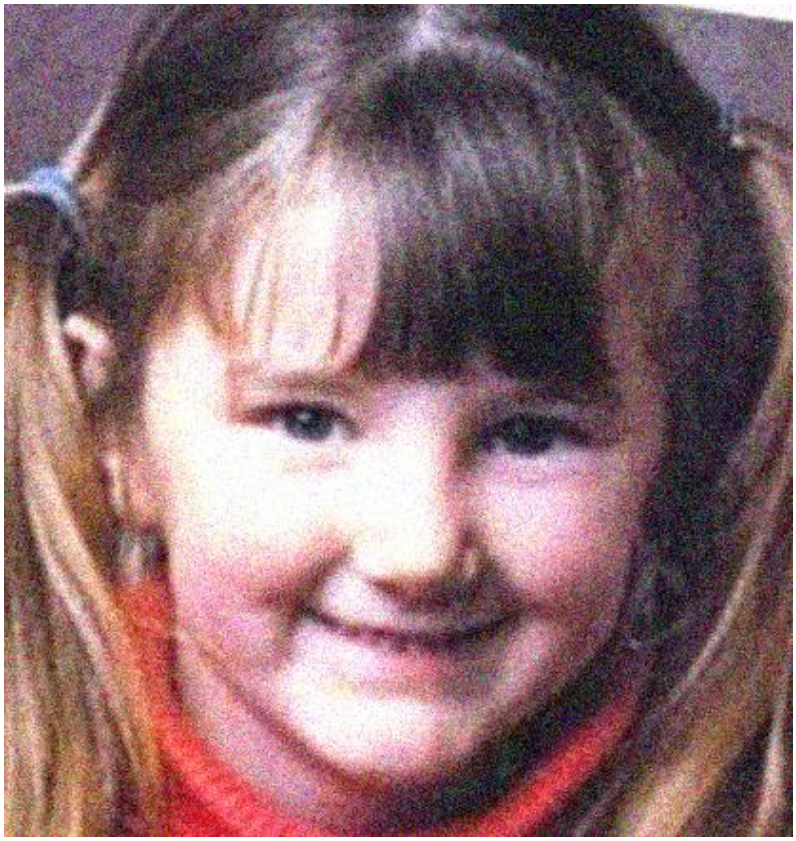


Figure 4.8: Picture of Mary Boyle at 6years. Image downloaded from https://www.irishtimes.com in January 2016.

Motivated by the fact that Mary's picture that is readily available (see Figure 4.8) is of poor quality and having facial expression, the proposed nonlinear framework with the Gaussian kernel is used to progress her face from 6 to 45 years. Thereafter, the progressed image is then compared to that of Ann at the

same age. It is hoped that this real application helps the police and the general public in the search for missing people.

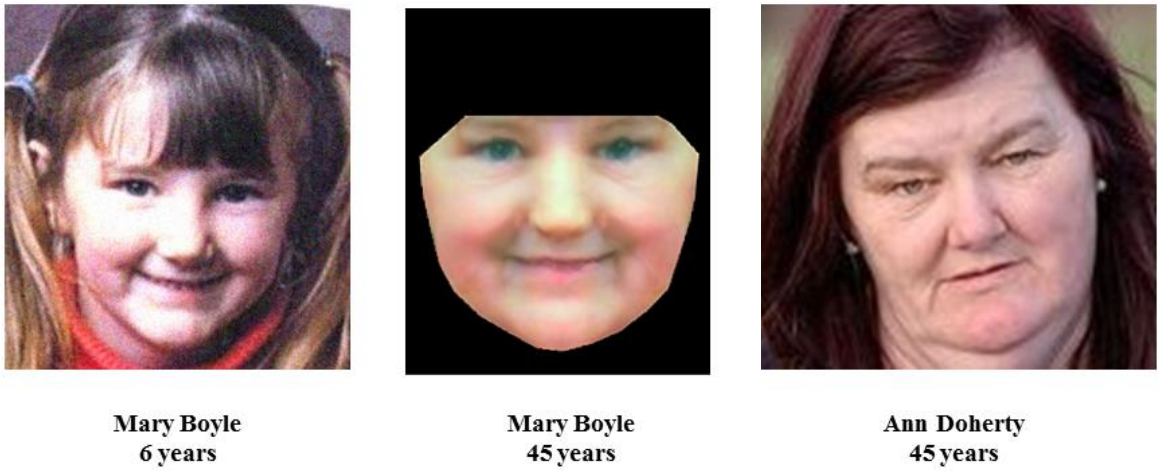


Figure 4.9 Age progressed image of Mary Boyle.

4.6 Summary

In this chapter, a nonlinear variant of the active appearance model (AAM) was proposed. Dubbed “KAM”, the model takes advantage of kernel machines and their ability to explore higher order correlations between input variables. Due to the image pre-processing ability of the *KPCA preImage* computation, age progression achieved via the nonlinear framework generates realistic images despite the effects of image noise, lightening variations, and facial expressions.

However, despite handling noise and varying facial expressions, the model suffers from hair artefacts and low resolution of the output image. These problems clearly affect both AAM and KAM frameworks, and it is not a surprise, as they rely on PCA and KPCA which are subspace learning techniques. Our findings show that subspace learning techniques have an averaging effect, thus

when used for image reconstruction, the generated images seem faded i.e. having low resolution, and some other times carry irrelevant texture details such as hair across ages. Thus, adult faces rendered using these age progression techniques lack sufficient texture information such as wrinkles. Furthermore, the averaging footprint at times transfers irrelevant texture details such as hair across ages, as a result some young faces that were rendered using the framework appear to have stubbles.

Towards, this end, there is the need to investigate a suitable technique of augmenting this texture deficit and defects, a method that will handle the problem of low resolutions as well as hair artefacts.

5 Texture Enhancement via an Example Based Approach

This chapter describes the development of a nonparametric pipeline used to enhance the texture quality of face images. Hence the pipeline improves low-resolution output of the KAM-based age progression framework.

5.1 Introduction

Both age progression techniques presented in earlier chapters of this thesis have been data driven, hence, learning patterns from the training data. Consequently, the algorithms represent information statistically via linear and nonlinear variants of principal component analysis (PCA). However, PCA, which is a vital part of these models, has an unfortunate drawback of averaging out texture details, therefore working as a low pass filter, and as such many of the face skin deformations and minor details become faded, resulting in a younger looking and faded out facial image. Furthermore, this averaging phenomenon also results to artefacts and ghosting at the time of image reconstruction. Interestingly, recent work in 2D [134] and 3D [135] animation has shown that patches of the human face are somewhat similar when compared remotely. Thus, researchers have proposed generating novel faces by compositing small face patches, usually from large image databases.

With the abundance of images on the Internet, it is then possible for us to use this patch-based synthesis approach to replace regions of blurred images with similar patches that have finer and detailed quality. Additionally, this same

approach can be used to augment other artefacts such as stubble and ghosting of eye colours.

Following these ideas, in this chapter, a method of hybridising parametric (statistical) and nonparametric procedures for age progression is proposed. Precisely, the method leverage's the robust facial reconstruction ability of KAM, while at the same time compensating texture information by forming composites with the aid of an Age-wise Example-based Texture Synthesis technique (AETS) to enhance the texture details.

5.2 Texture enhancement pipeline

To address the issue of low texture resolution and at the same correct facial artefacts, a number of procedures are therefore coupled into a pipeline (see Figure 5.1) to achieve AETS. The pipeline entails, the construction of a low-quality age progressed image using a statistical model, here, the statistical method of choice is the KAM-G age progression framework that was presented in the previous chapter. Secondly, both age-progressed and the original individual's image are segmented into uniform overlapping patches. Thereafter, based on the segmentation model, age-wise patch library (database) of finely grained textures is formed from hundreds of images collected over the Internet. Finally, an appropriate template matching algorithm is deployed to select and replace patches of the KAM-G synthesised facial image with enhanced textures retrieved from the database whilst ensuring patch consistency, as well as resemblance to the original image.

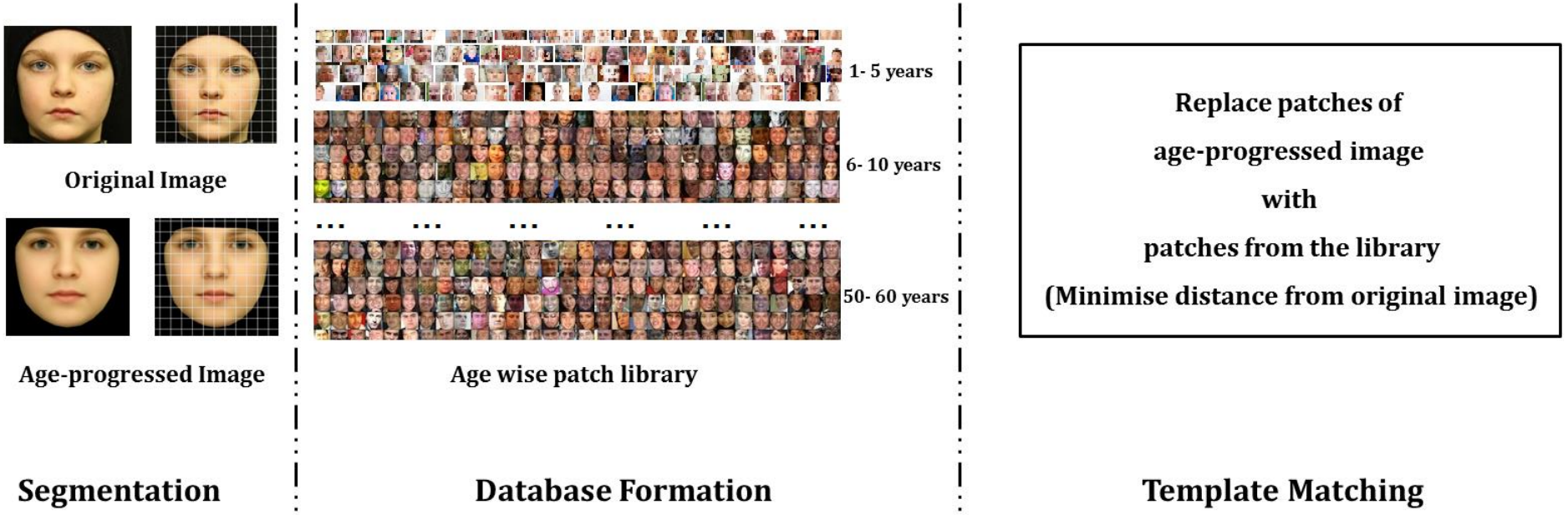


Figure 5.1: Texture enhancement pipeline.

5.2.1 Segmentation

As an initial step the original (F_{init}) and KAM-G progressed (F_{aged}) images are affine warped to a template shape, this is done as a form of pre-processing to ensure patch to patch correspondence. Since enhancing the age-progressed image is the ultimate goal of the procedure, then, its shape is an ideal template. Hence, one only has to warp the original image F_{init} to the template shape i.e. the shape of F_{aged}. Next, each image is segmented into an array of 72 overlapping patches arranged as a regular grid having 9×8 dimensions (see Figure 5.2). To be precise, the width of the overlap region is one-fifth that of the patch. The rationale, behind this segmentation procedure, is to reduce the face into small sections for the purpose of comparison, and the overlap ensures smooth transition and stitching of textures when rendering the composite face.

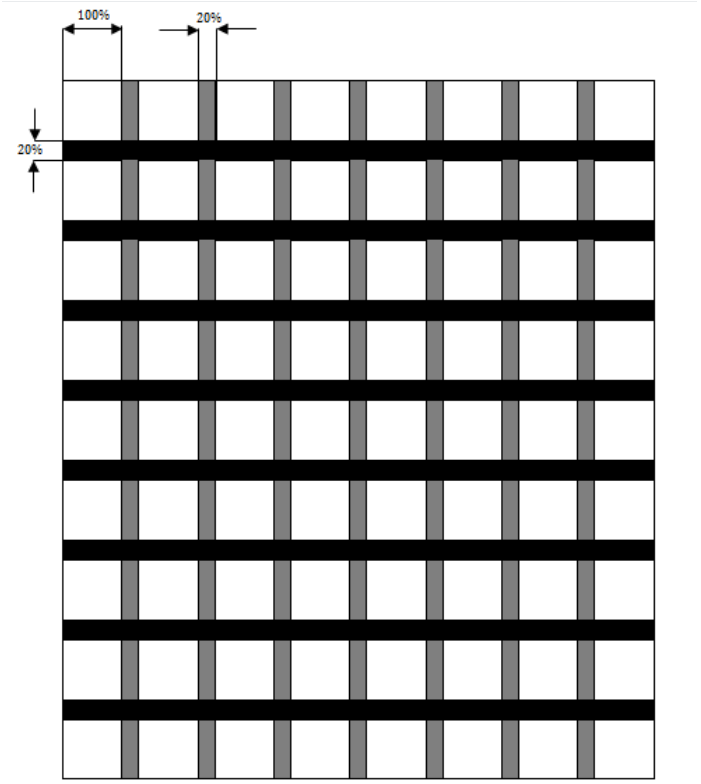


Figure 5.2: Image segmentation pattern.

5.2.2 Database Formation

Using images collected over the Internet, age-wise patch databases for 9 age groups were formed. The age groups are, 0-2, 3-7, 8-13, 14-21, 22-28, 29-36, 37-45, 46-58, and 59 -70 years. Next, all the images are converted to patches following the segmentation procedure that was utilised for the original and synthesised faces. In a nutshell, they are affine warped to the shape of image F_{aged} and cut into 72 overlapping patches.

5.2.3 Template Matching

After the successful creation of 9 patch libraries i.e. one for each age group, there follows a systematic patch selection procedure. Given a synthesised image F_{aged} at age a_{new}, the patch library to use for swapping textures is the one whose age group corresponds to that of F_{aged}. The patch swapping procedure is achieved by performing three region matching comparisons:

— Comparison between a patch in the library p_k and corresponding patch $p_k{}'$ on the KAM-G synthesised face F_{aged}

— Comparison between overlap region of the patch p_k to the overlap region of the patch above it $p_i{}'$ and to its left $p_j{}'$ both on the face F_{aged}

— Comparison between a patch from the library p_k and corresponding patch p_k'' on the original face F_{init}

The rationale behind the above three comparisons is to ensure that the patch to be copied best matches the patch to be replaced on F_{aged}. It also guarantees that the patch overlays smoothly, matching the patch above and below it on the synthesised face F_{aged}. Furthermore, the last comparison helps to recover

features of the original face that the statistical model missed at the time of initial face synthesis, thus it acts as a constraint that minimises the distance between the synthesised composite and the original face.

Image texture comparison requires some form of measurement and has been a well-studied problem in computer vision. As a matter of fact, there is a huge literature devoted to this problem, some past [136] and recent researchers [137] have even made attempts to compare some of the most popular techniques. In general, there is a need to have a trade-off between the error rate of the algorithm and its computational speed. As such one of the most popular similarity measures is the Euclidean Distance (ED) due to its simplicity and computational speed. Unfortunately, ED which is simply the straight line distance between two points does not take into account the spatial relationship of image pixels, hence it is quite sensitive to small deformations (rotation, translation, and scaling), noise, and change in illumination [138]. In this regard, an improved variant of ED is used here, that is the Image Euclidean Distance (IMED) [139], a distance measure which takes into account the pixel's spatial relationship, hence relatively insensitive to spatial deformations.

IMED is computed by embedding a positive definite matrix G into the traditional Euclidean Distance (ED). The matrix G which is usually a Gaussian function defines the distance between the ith and jth pixels. Since the function is continuous and monotonically decreasing as the distance between pixels increases, it then follows that smaller deformation causes smaller changes in the distance. The distance can be expressed as,

$$IMED = (\boldsymbol{x}_1 - \boldsymbol{x}_2)^T G(\boldsymbol{x}_1 - \boldsymbol{x}_2)$$
$$G = \sum_{i=1}^{MN}\sum_{j=1}^{MN} g_{ij},\ g_{ij} = \frac{1}{2\pi\sigma^2}\exp\left(-\frac{(d_{ij}^S)}{2\sigma^2}\right) \quad (5.1)$$

where $\boldsymbol{x}_1$ and $\boldsymbol{x}_2$ are vectorised forms of the images to be compared, σ is the width of the Gaussian function, and d_{ij}^S is the spatial pixel distance between ith and jth pixels, so if P_i is at location (m, n) and P_j is at location (m', n') their distance is given by,

$$d_{ij}^S = ((m - m')^2 + (n - n')^2))^{1/2} \quad (5.2)$$

After successful selection of image patches, corresponding regions of the age-progressed face are replaced. Finally, texture normalisation of the resulting composite face is performed to discard illumination variations.

5.3 Experiments

5.3.1 Database

In order to implement the proposed AETS pipeline, the nine agewise patch libraries were populated using a total of 3000 images. All the photographs are high-quality colour images that were collected over the Internet, with a gender to male ratio of 50:50. To be precise, each age-group patch database has up to 300 images. All subjects are Caucasian, displaying varying facial expressions and head poses. The picture qualities also show illumination, sharpness, and resolution variations.

5.3.2 Implementation of the proposed approach

As stated earlier, the KAM-G proposed in chapter 4 was used to generate the statistical age-progressed image. In order to form the array of corresponding patches, the original image, KAM-G age-progressed image, as well as all photographs that were added to the patch libraries were first cropped to a size of 340×340 pixels, then following the proposed segmentation procedure, they were all sliced into the 9×8 grid array with each patch having a size of 25×20 pixels as shown in Figure 5.3. Thus the vertical and horizontal overlays have dimensions of 5×20 and 25×4 pixels respectively.

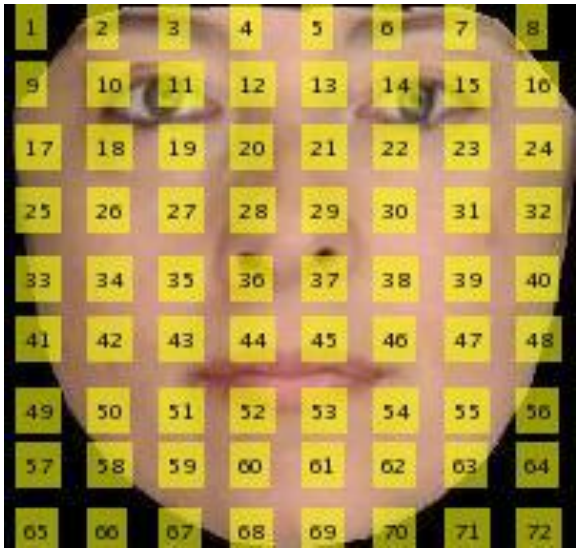


Figure 5.3: Grid of 9 x 8 patches.

To conduct the patch selection, the 3 region comparison steps were performed on grayscale versions of the images. Using a heuristic, weighted distances were computed. As shown in Table 5.1 the weights indicate how each of the distances contributes to the final patch selection metric which given by,

$$\partial_{all} = \alpha\partial_{patch} + \beta\partial_{overlay} + \gamma\partial_{original} \quad (5.3)$$

where α, β, and γ are the weights defined in Table 5.1, ∂_{patch}, $\partial_{overlay}$, and $\partial_{original}$ are the results of the different patch distances also defined in Table 5.1.

Weights were chosen such that, much relevance is given to how closely the faded patch resembles the new patch to be copied, next relevance is given to the similarity of overlay regions since they ensure the smooth transition of image regions. Finally, to ensure identity retention, consideration is given to the similarity of the patch to be copied to the original image; this is given a small weight to avoid complete distortion of the KAM-G output. As a matter of fact, the computed weighted distance ∂_{all} is scaled to a value within the range of 0 and 1, with zero indicating the best match and one referring to total mismatch.

Table 5.1: Weights assigned to region similarities using a heuristic.

Weight	**Notation**	**Comparison**
0.5	∂_{patch}	Between region p_k in the library and corresponding $p_k{}'$ on the aged face F_{aged}
0.3	$\partial_{overlay}$	Between overlay regions of patch p_k and the overlay regions above and to left of $p_k{}'$ on the aged face F_{aged}
0.2	$\partial_{original}$	Between patch p_k and corresponding patch p_k'' on the original face F_{init}

In other to compute the distance metric, IMED's the width of the Gaussian function was computed using equation (4.25), so that σ remains small enough

while capturing optimum neighbourhood information of each image pixels. It was observed that, as σ tends towards zero, IMED turns to the traditional Euclidean Distance. On the other hand, as the value of σ becomes substantially large, the images become completely blurred.

To cut down computational cost by reducing the number of patch to patch comparisons, a symmetry constraint was enforced such that only patches for one-half of the face are searched, afterwards the same patch for the other half were automatically copied; this way one is sure that generated pairs of eyes, nostrils, and lips are consistent. Symmetry constraint is depicted in Figure 5.4 using different colours to represent patches that were copied in a particular instance of the conducted experiments. In the example shown in Figure 5.4, the face was formed from 23 unique patches learned from the database. As stated earlier, the symmetry constraint ensures one half of the generated face is identical to the second half of the face.

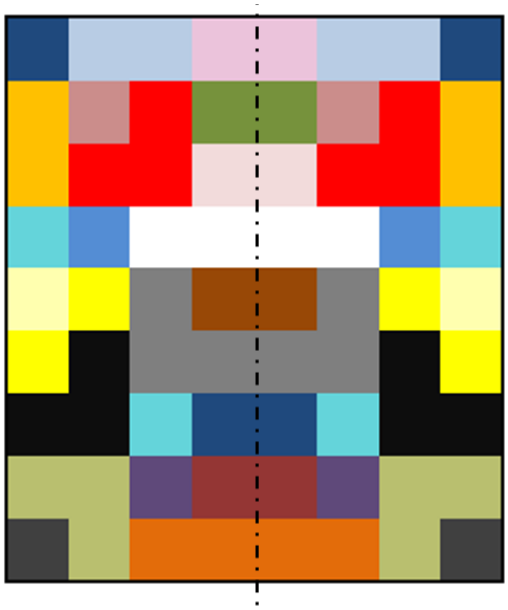


Figure 5.4: Colours to indicate origins of patches with symmetry constraint.

After successful patch selection, next, the chosen patches are stitched as shown in Figure 5.5, to form a composite face with enhanced textures. Finally, gradient and other illumination variations are eliminated using Poisson Image Editing [140] as shown in Figure 5.6.

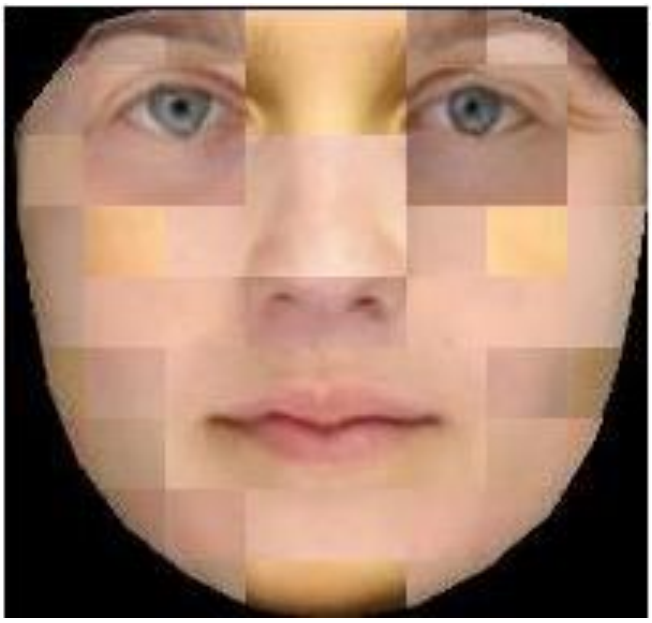


Figure 5.5: Sample of composite face formed from patches.

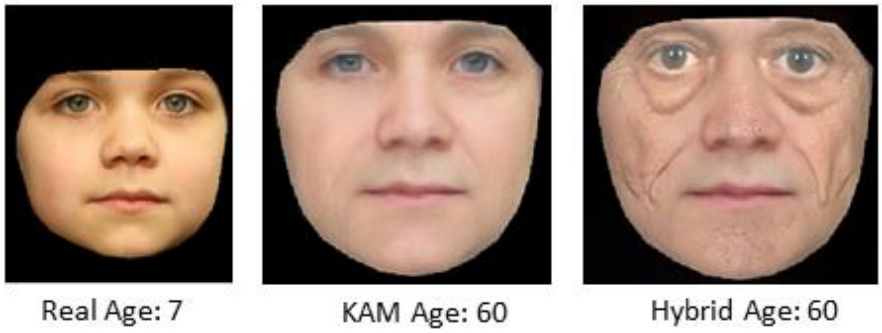
Figure 5.6: Illumination-normalised texture enhanced output.

5.3.3 Results

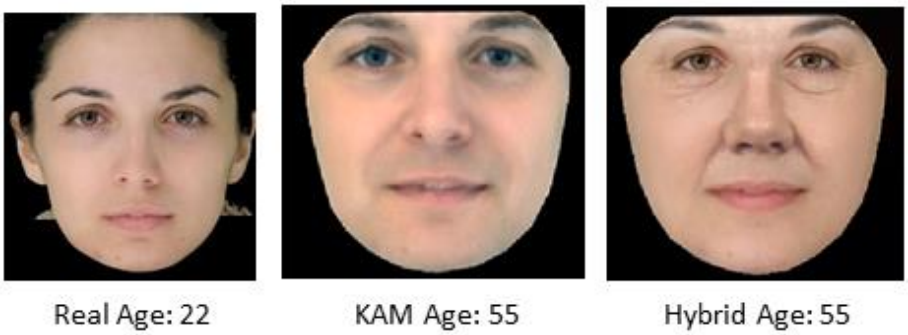
Samples of generated results are shown in Figure 5.7. Original faces are on the farthest left, next to which are those generated using the statistical model (KAM-G) and on the farthest right are the texture enhanced outputs. As can be observed the outputs of the hybrid procedure have more detailed texture information, furthermore, they are free from artefacts such as ghosting and stubbles that appear on young and feminine faces.

(a)

(b)



(c)



(d)

Figure 5.7: Sample of age progression using hybrid technique.

Next, both machine and human based tests were conducted to evaluate the ability of the rendered output to retain identity and to ascertain the intended age. Having achieved a mean objective score of 84.36%, this method obviously outperforms both KAM and AAM approaches to age progression; for a comparison, the mean scores of the objective tests have been presented in Table 5.2. Histogram shown in Figure 5.8 has been used to further explore the overall performance of the test, as can be seen over 60% of the test images had scores between 80% and 100% this indicates that most of the images retained identity of the subject with a high degree of accuracy. Additionally, KS test showed that the test scores were significantly better than the results of KAM (D=0.2327, p=0.0202) and AAM (D=0.3501, p=0.0001) based techniques.

Table 5.2: Comparison of mean scores (objective test).

Technique	**Mean Scores (%)**
AAM based implementations	
Lanitis [15] method	69.82
OLS approach	71.86
PLS approach	73.16
sPLS approach	**74.36**
KAM based implementations	
KAM-S	77.03%
KAM-L	78.19%
KAM-G	**79.34%**
Proposed model	
Hybrid	**84.36%**

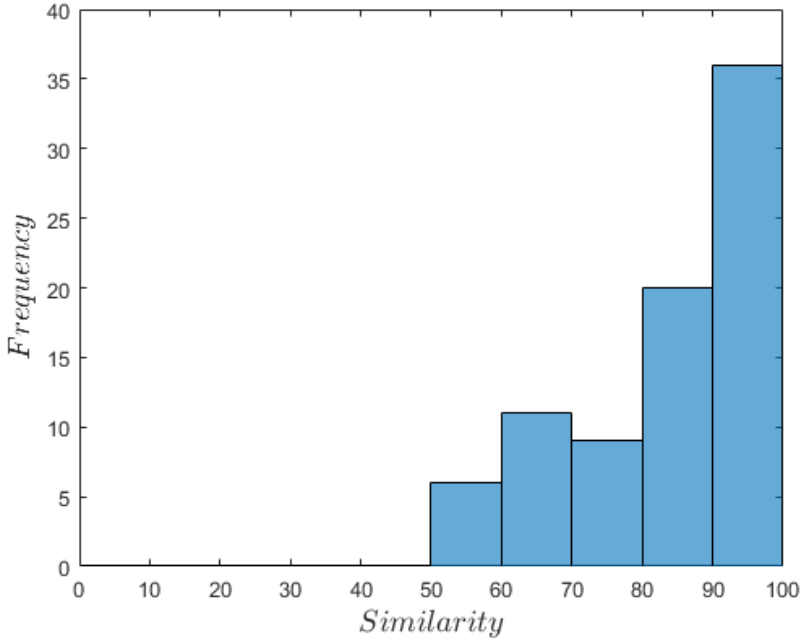


Figure 5.8: Histogram of objective test scores (Hybrid technique).

The bar graph in Figure 5.9 indicates that most of the progressed images were perceived by human observers to greatly resemble the test subjects.

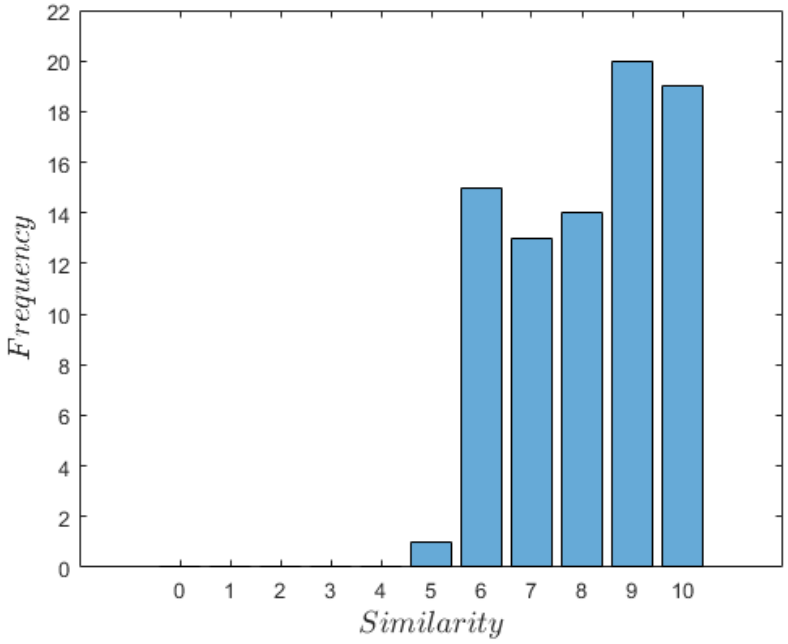


Figure 5.9: Bar graphs of subject identity scores (Hybrid technique).

Table 5.3: Mean scores of subjective test (Hybrid technique).

Technique	**Mean Scores**
AAM based implementations	
Lanitis [15] method	6.1707
OLS approach	6.4146
PLS approach	6.6585
sPLS approach	**6.7805**
KAM based implementations	
KAM-S	7.2195
KAM-L	7.3171
KAM-G	**7.4512**
Texture Enhanced model	
Hybrid	**8.1463**

The mean score of human identity ratings presented in Table 5.3 further shows the hybrid technique's superiority. Evidently, the texture enhanced technique surpasses the KAM method, which in turn surpasses the AAM approach.

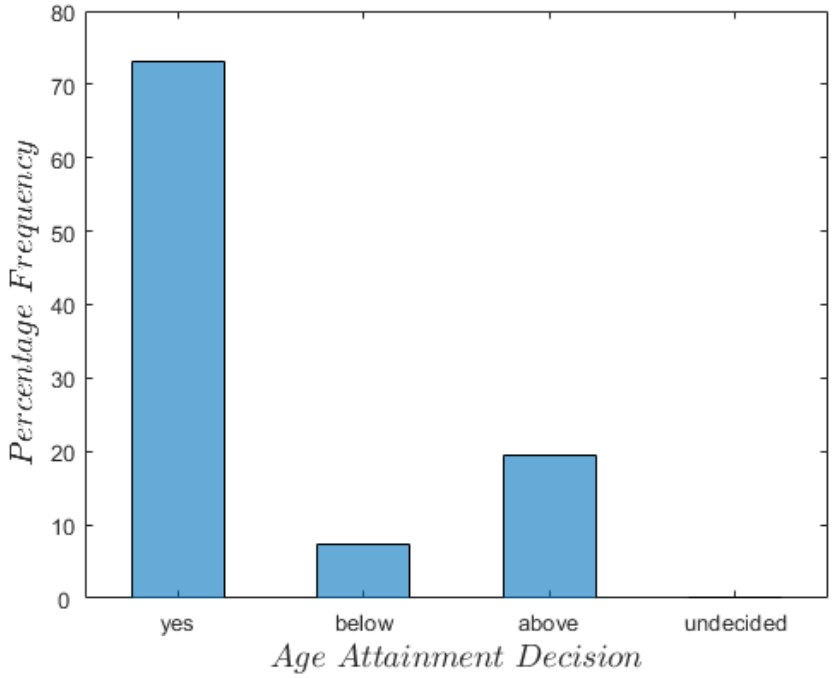


Figure 5.10: Bar graph representation of subjective age attainment test for hybrid technique.

Thus, this hybrid approach presents a way of getting the advantages of both parametric and nonparametric synthesis techniques. An obvious advantage of the hybrid technique is its ability to provide better texture. The fact that the pipeline considers both synthesised and original faces during patch selection makes it possible to better preserve unique features such as the eye and lip colour which easily get tempered by the averaging effects of PCA and KPCA.

5.4 Summary

A hybrid technique for augmenting problems associated with statistical models was presented in this chapter. When utilised for automatic facial age progression, the method provides promising results. Coupled with the findings of previous chapters, this hybridisation has provided a means of solving some

of the long-standing problems associated with age progression. It is now possible to project faces to varying ages without having to solely rely on a lookup table, facial expressions, and noise effects have also been tackled. Finally, faces with enhanced texture detail that are free from defective artefacts can be rendered, thus achieving realistic automatic facial age synthesis.

The remaining part of this thesis will be focussed on developing a suitable age estimation algorithm which will then be used to objectively evaluate the ability of the proposed age progression frameworks in rendering the intended age. It is worth mentioning that until this point, only the perception of human observers has been used to evaluate the ability of the techniques to progress the face to the intended age.

6 Age Estimation using Supervised Appearnace Models

Here, a Supervised Appearance Model (sAM) is derived and used to capture facial ageing features. Next, automatic age estimation is performed via regression.

6.1 Introduction

In previous chapters of this thesis, techniques for age progression were proposed with the sole goal of addressing three major problems; challenge of working with noisy images, the effect of facial expression on rendering, and the problem of low resolution output. As this research traversed from AAM methods, to KAM as well as hybrid techniques, results of several experiments have revealed consistent increment in the rendering capability of the algorithms. However, just as stated in [8], the key issue in automatic age synthesis is the generation of accurate predictions. Although the proposed techniques have been evaluated using rigorous procedures, one assessment is obviously missing; no objective test has been conducted to evaluate the ability of the algorithms to attain the intended age. Precisely, the machine based ED score test only evaluates the ability of the age synthesis methods to preserve identity. Although human based age attainment test has been conducted, the fact that this test is subjective makes it necessary to conduct an equivalent machine based assessment. A simple yet veracious way of conducting this assessment is to use automatic age estimation; given an age progressed image, an automatic age estimator can be used to find the precision of the progression technique. Ideally, it is expected that the estimated age will be

equivalent to the intended age. Hence, the remaining part of this thesis is focused on the exploring an optimal automatic age estimation algorithm, so that it can be used as an evaluation tool.

6.2 Age estimation problem

Just as previously mentioned, work done on age estimation tends to follow a two-stage process; feature extraction and pattern learning. To start with, in this chapter, feature extraction is conducted by employing a statistical model due to its ability to capture both shape and texture details of the face. It may be noted that both statistical models considered in this thesis have been driven by PCA (linear and nonlinear), however, due to the unsupervised nature of PCA, it only captures characteristics of the predictor variables(face data). Hence, both PCA and KPCA do not give importance to how each face feature may be related to the class label (age). Nonetheless, in a typical problem of estimation/prediction, there is a need to explore attribute of the predictor variable that is best related to the response variable. Thus, to perform feature extraction for age estimation, a variant of the conventional statistical model is proposed; a model that improves on both AAM and KAM by embedding PLS in place of PCA/KPCA.

As discussed in chapter 3, PLS is a dimensionality reduction technique which maximizes the covariance between the predictor and the response variables, thereby generating scores that have both reduced dimension and superior predictive power. Here, the proposed model termed supervised appearance model (sAM) will then be used for feature extraction with a view to achieving age estimation. In the latter part of this chapter, the proposed age estimation

technique is evaluated by comparing to state-of-the-art algorithms, for this purpose the FGNET-AD benchmark database is utilised.

6.3 Partial Least Squares Regression (PLS)

As a recap to our previous introduction of PLS, the technique generalizes and combines features from multilinear regression and PCA [141], thus it has been used for both regression and dimensionality reduction in the literature [142]. The technique is very useful when there is a need to predict a dependent variable from a large set of predictors. Although similar to PCA, it is much more powerful in regression and classification applications, because it searches for components (latent vectors) that capture directions of highest variance in X as well as the direction that best relates $\boldsymbol{X}$ and $\boldsymbol{Y}$ (i.e. covariance between $\boldsymbol{X}$ and $\boldsymbol{Y}$). Hence it performs simultaneous decomposition of $\boldsymbol{X}$ and $\boldsymbol{Y}$ while PCA only finds the direction of highest variance in $\boldsymbol{X}$, so the principal components (PCs) only describe $\boldsymbol{X}$; however, nothing guarantees that these PCs which explain $\boldsymbol{X}$ optimally, will be appropriate predictors of $\boldsymbol{Y}$.

To sum up it can be said, PCA performs dimensionality reduction in an unsupervised manner, while PLS does so in a supervised manner. Hence, it intuitively performs better in prediction applications such as that of age estimation.

The formulation of PLS involves the decomposition of $\boldsymbol{X}$ and $\boldsymbol{Y}$ variables using (3.30). Mathematically, it is possible to reconstruct the original data $\boldsymbol{X}$ from the latent score $\boldsymbol{Z}$ defined in equation (3.31), by inverting $\boldsymbol{R}$ the matrix of weights $\{\boldsymbol{r}_1, \boldsymbol{r}_2, \dots \boldsymbol{r}_k\}$,

$$\boldsymbol{X} = \boldsymbol{Z}\boldsymbol{R}^{-1}, \text{ or } \boldsymbol{X} = \boldsymbol{Z}\boldsymbol{T} \text{ where } \boldsymbol{T} = \boldsymbol{R}^{-1} \tag{6.1}$$

Here the inverse of weight $\boldsymbol{T}$ is referred to as the projection matrix.

Suppose there is a mean centred training set $\boldsymbol{X}_{tr}$ consisting of observations whose class labels are known and denoted by $\boldsymbol{Y}_{tr}$. Given a test set $\boldsymbol{X}_{ts}$ whose class label has to be predicted, PLS can be used for dimensionality reduction by projecting the test data onto the weight matrix $\boldsymbol{R}$. Hence the latent scores matrix $\boldsymbol{Z}_{ts}$ for the test data is computed as shown below,

$$\begin{aligned} \boldsymbol{Z}_{tr} &= \boldsymbol{X}_{tr}\boldsymbol{R} \\ \boldsymbol{Z}_{ts} &= \boldsymbol{X}_{ts}\boldsymbol{R} \end{aligned} \tag{6.2}$$

The equation (6.2) above provides a convenient means of using PLS for supervised dimensionality reduction. Thus, shall be utilised in building the sAM.

6.4 Supervised Appearance Model (sAM)

Just like the KAM and conventional AAM, the proposed sAM is built to capture both shape and texture variability from the training dataset. This can be realised by forming a parameterised model using PLS dimensionality reduction to capture the variations as well as combine them in a single model.

As an initial procedure, the shape of each face in the training database is represented by a set of two-dimensional landmark's vector $\mathbf{x}$, representing the *x* and *y* coordinates of the fiducial points defined in equation (3.1).

Following the procedure mentioned previously, rotational, translational and scaling variations are eliminated from the landmarks data by aligning all the shapes using GAP.

Thereafter, a supervised shape model is built by performing PLS dimensionality reduction described in equation (6.1). Here, the predictor variable is the matrix of face shapes $\boldsymbol{X} = \{\mathbf{x}_i\}$ for each individual face used to train the model. The response variable $\boldsymbol{Y}$ is a $1 \times n$ vector containing ages of n persons used to train the model. Using the latent scores $\boldsymbol{Z} = \{\boldsymbol{z}_1, \boldsymbol{z}_2, \ldots \boldsymbol{z}_n\}^T$ computed via PLS, each face shape can then be represented by a linear equation given by,

$$\mathbf{x} - \bar{\mathbf{x}} = \boldsymbol{z}_{\mathrm{x}} \boldsymbol{T}_{\mathrm{x}} \tag{6.3}$$

For convinience, the equation (6.3) above, can be written as,

$$\mathbf{x} = \bar{\mathbf{x}} + \boldsymbol{z}_{\mathrm{x}} \boldsymbol{T}_{\mathrm{x}} \tag{6.4}$$

where $\bar{\mathbf{x}}$ is the mean shape, $\boldsymbol{z}_{\mathrm{x}}$ is a vector of latent scores representing that particular shape, and $\boldsymbol{T}_{\mathrm{x}}$ the projection matrix computed from all the training shapes.

Next, to build the supervised texture model, all face images are affine warped to the mean shape $\bar{\mathbf{x}}$. Illumination discrepancies are then normalised by applying a scaling and an offset to the warped images in the same way as that of chapter 3. Finally, each matrix of image pixel intensities (textures) is converted to vector

$\mathbf{g}$. By applying PLS to the matrix $\boldsymbol{G} = \{\mathbf{g}_i\}$, the texture of each image can be represented in a supervised manner via,

$$\mathbf{g} = \bar{\mathbf{g}} + \boldsymbol{z}_g \boldsymbol{T}_g \tag{6.5}$$

where $\bar{\mathbf{g}}$ is the mean texture, $\boldsymbol{z}_g$ is a vector of latent scores representing the texture of a particular face, and $\boldsymbol{T}_g$ the projection matrix of textures.

From equations (6.4) and (6.5) above, it is then possible to summarise both face shape and texture using the latent vectors $\boldsymbol{z}_{\mathrm{x}}$ and $\boldsymbol{z}_g$. Consequently, a combined appearance model of shape and texture can be derived by concatenating the two vectors.

$$\boldsymbol{z}_c = (\boldsymbol{z}_{\mathrm{x}} \quad \boldsymbol{z}_g)^T \tag{6.6}$$

To further eliminate correlation that may exist between shape and texture, PLS is used to reduce the dimension of $\boldsymbol{z}_c$. Thus, the sAM can be represented by a linear equation,

$$\boldsymbol{z}_c = \boldsymbol{l}\boldsymbol{T}_c\,, \qquad \boldsymbol{T}_c = (\boldsymbol{T}_{cx} \quad \boldsymbol{T}_{cg})^T \tag{6.7}$$

here, $\boldsymbol{l}$ is a vector of latent scores representing both shape and texture, and $\boldsymbol{T}_c$ is the projection matrix of the combined model. It is worth noting that as expressed in (6.7), $\boldsymbol{T}_c$ has two components related to the shape and textures respectively. Since both $\boldsymbol{z}_{\mathrm{x}}$ and $\boldsymbol{z}_g$ have zero mean, then $\boldsymbol{z}_c$ also has zero mean.

Similar to the conventional AAM, the linear nature of the supervised model makes it possible to express both shape and texture in terms of the parameter $\boldsymbol{l}$.

$$\mathbf{x} = \bar{\mathbf{x}} + \boldsymbol{l}\boldsymbol{T}_{cx}\boldsymbol{T}_{\mathrm{x}}, \qquad \boldsymbol{g} = \bar{\boldsymbol{g}} + \boldsymbol{l}\boldsymbol{T}_{cg}\boldsymbol{T}_{g} \tag{6.8}$$

Equation (6.8) above, describes the supervised appearance model (sAM), a variant of the statistical models discussed in chapters 3 and 4. Since the parameter $\boldsymbol{l}$ summarises both shape and texture information, it gives us a convenient supervised way of representing faces with a view to solving the problem of age estimation.

6.5 Pattern Learning

The sAM contains both shape and texture components and has been derived to encode age related information via the PLS which performs simultaneous dimensionality reduction of both face and age details. Thus, it is used to extract face features $\boldsymbol{l}$, thereafter, an ageing pattern is learnt using a regression approach. Here regression is achieved using simple models with a view to exploring the power of the feature extraction technique (i.e. sAM). Hence, ordinary linear (OLS) and quadratic function (QF) are used,

$$age = \alpha + \boldsymbol{\beta}^{T}\boldsymbol{l} \tag{6.9}$$

$$age = \alpha + \boldsymbol{\beta}_1^{T}\boldsymbol{l} + \boldsymbol{\beta}_2^{T}\boldsymbol{l}^2 \tag{6.10}$$

where α is the intercept, $\boldsymbol{\beta}$, $\boldsymbol{\beta}_1$ and $\boldsymbol{\beta}_2$ are vectors of regression coefficients.

6.6 Experiments

The effectiveness of the proposed feature extraction technique is evaluated by comparing the results of age estimation conducted using sAM features, to those performed using AAM and the KAM. As stated in the previous section, estimation is evaluated by incorporating the sAM features to two simple traditional regression algorithms; linear and quadratic functions. Furthermore, results of the sAM-based age estimator are compared to other state-of-the-art algorithms. To ensure consistency and fairness of comparison all experiments are conducted using the FGNET-AD [85].

6.6.1 Performance Evaluation Metric

To evaluate the accuracy of age estimations, leave one person out (LOPO) cross validation method [43] is utilised for all our experiments. LOPO entails using the image of 1 person as test set while an estimation model is built using images of all the other subjects contained in the database. So, by the end of 82 folds, each subject in the FGNET-AD will have been used for testing. This approach mimics a real life scenario where the classifier is tested on an image that has not been seen before. In addition, the LOPO approach unlike other cross validation techniques ensures consistency of results and ease of comparative evaluation of different algorithms.

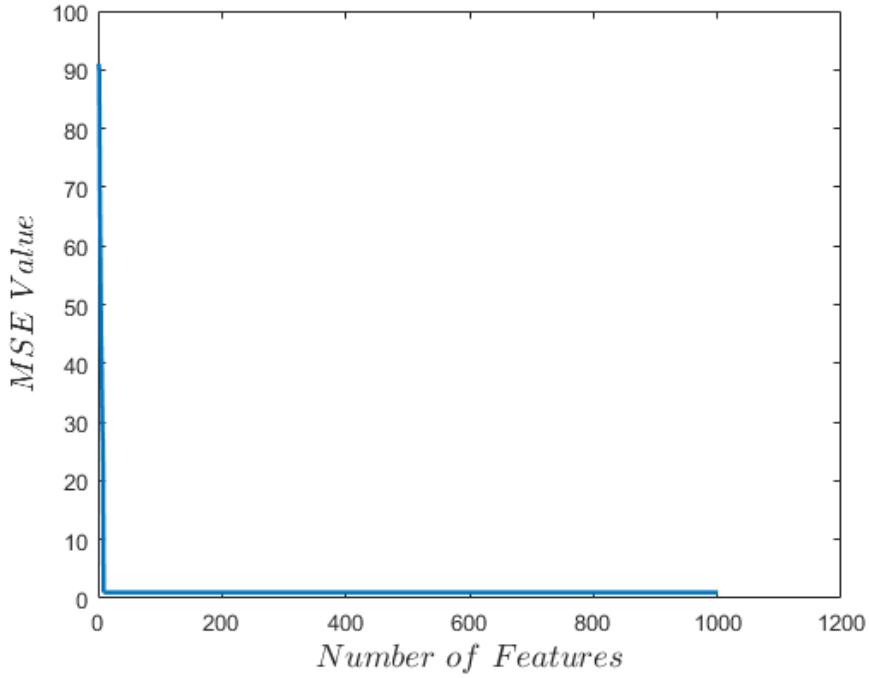
The performance measures used for age estimation are Mean Absolute Error (MAE) and Cumulative Score (CS), given by,

$$MAE = \sum_{i=1}^{N} | y - y' | / N, \qquad CS(m) = N_{error \leq k} / N \times 100\% \tag{6.11}$$

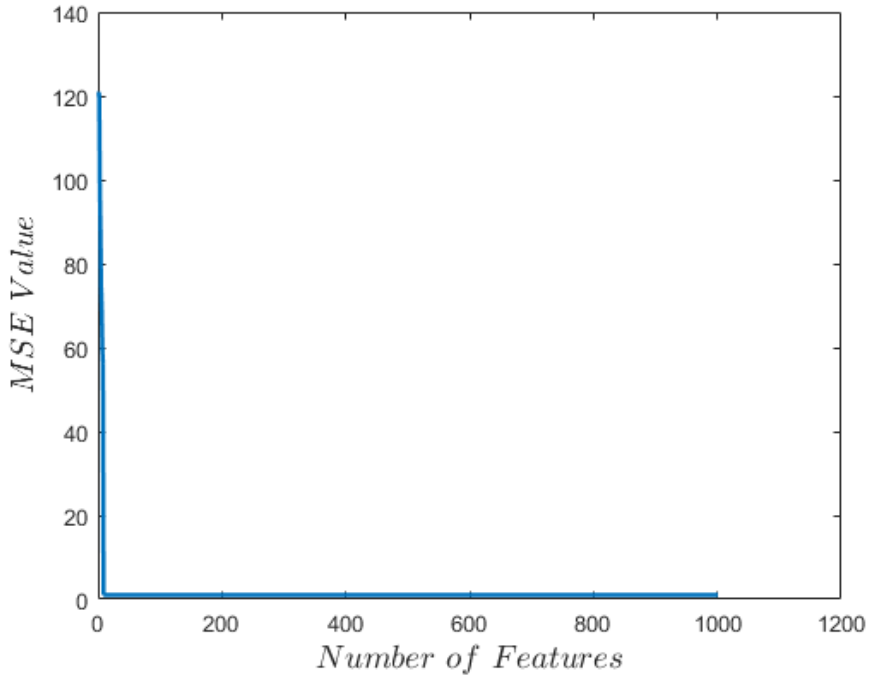
where y is the ground truth age, and y' is the estimated age, N the number of test images, and $N_{error \leq k}$ denotes the number of images on which the system makes the absolute error not higher than k years.

6.6.2 Implementation

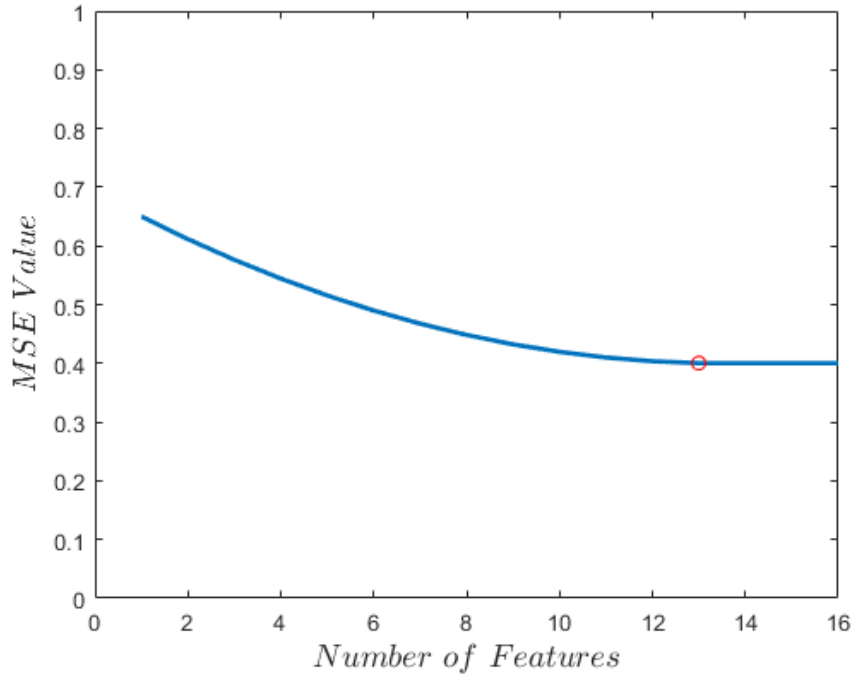
In order to extract features via the sAM, equations (6.4) and (6.5) were used to compute the latent parameters of shape $\boldsymbol{z}_{\mathrm{x}}$ and texture $\boldsymbol{z}_{g}$, each of these two was represented using just 8 components chosen via cross-validation (see Figure 6.1). Next, the supervised shape and texture models are combined using equation (6.6) to form an $n \times 16$ matrix, where n refers to the number of observations, for the FGNET-AD $n = 1002$. Finally, the second PLS is performed using (6.7). Eventually, the face $\boldsymbol{l}$ is represented using 13 components; again the optimum number of components is decided via cross-validation as shown in Figure 6.1 (c). The shapes of the first two graphs shown in Figure 6.1 (a and b) reveal that using the supervised model, very few PLS components were required to represent the shape and texture of the human face, specifically all the variations are accounted for by as small as 8 shape and texture components. The curve in Figure 6.1c reveals that when combined into a single model, the 16 features representing shape and texture of the face can further be reduced to just 13 optimal features.


MSE Value
Number of Features
0
10
20
30
40
50
60
70
80
90
100
0
200
400
600
800
1000
1200

(a)


MSE Value
Number of Features
0
20
40
60
80
100
120
140
0
200
400
600
800
1000
1200

(b)



(c)

Figure 6.1: Mean square error per number of features (a) supervised shape model (b) supervised texture model (c) supervised appearance model.

To achieve age estimation, two regression algorithms described in equations (6.9) and (6.10) were utilised. For the quadratic function, the number of squared terms were computed in a sparse manner; as a form of regularization, only few predictor variables were squared. Hence, instead of computing the second order terms of all 13 components, only the 2nd order terms of the first 7 independent variables $(l_1^2, l_2^2, \ldots\ l_7^2)$ were used, once again this choice is made by cross-validation i.e. comparison results of different combinations as shown in Figure 6.2.

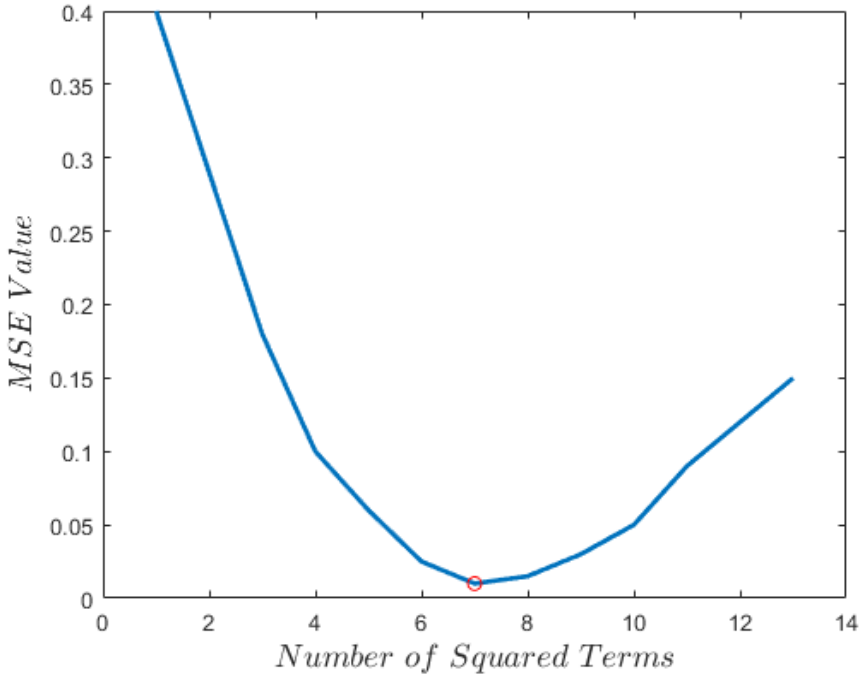


Figure 6.2: Choice of squared terms for QF.

6.6.3 Results

To fully evaluate the sAM based age estimator, three sets of experiments were conducted. First, the two regression techniques sAM fed to OLS regressor (sAM-1), and sAM feature with sparse quadratic regressor (sAM-2) were compared to estimations conducted using AAM and KAM algorithms that were presented in earlier parts of this thesis. MAE and CS results presented in Table 6.1 obviously shows the superiority of sAM as compared to features extracted using the other statistical methods. This proves the ability of PLS encoding to preserve face features that are best related to the age information.

Table 6.1: Comparison of sAM to AAM and KAM estimations.

Feature	**MAE**	**CS < 10**
AAM-OLS	10.01	55.88%
AAM-PLS	7.14	77.82%
AAM-sPLS	6.97	78.94%
KAM_S	6.93	79.67%
KAM_L	6.77	80.67%
KAM_G	6.75	80.74%
sAM-1	5.92	83.03%
sAM-2	**5.49**	**85.34%**

Next, a second experiment was conducted to compare, sAM-2 results to those of published works, where researchers used the conventional statistical model (AAM) for their feature extraction, coupled with variety of regression models for estimation. Despite, the sophistication of their regression techniques, the simple yet powerful sAM-2 method clearly gives promising results (see Table 6.2). This further highlights the superiority of PLS over PCA dimensionality reduction; it also shows the significance of feature extraction in age estimation tasks.

Table 6.2: Comparison of sAM to research works that used statistical models.

Feature	**Algorithm**	**MAE**	**CS < 10**
AAM	WAS[47]	8.06	≈77%
AAM	QF [15]	7.57	≈78%
AAM	SVM [43]	7.25	≈76%
AAM	AGES [43]	6.77	≈81%
AAM	AGES LDA [43]	6.22	≈82%
AAM	RUN1[143]	5.78	≈84%
AAM	MLP [47]	10.39	≈60%
AAM	IIS-LLD [144]	5.77	NA
AAM	OLS [9]	10. 01	55.88%
Proposed	**sAM-2**	**5.49**	**85.34%**

Finally, a third comparison was made between sAM-2 and other techniques that did not use statistical models. Purposely BIF and CNN results are compared since they have been reported to be the state-of-the-art in recent times.

Table 6.3: Comparison of sAM to other state-of-the-art techniques.

Method	**MAE**
sAM-2	5.49
BIF[68]	4.77
C & H BIF [145]	4.60
OHR [146]	4.48
LSR [147]	4.38
CNN [72]	4.22
BI. AAM [148]	4.18
EBIF [69]	3.17

Obviously, results of Table 6.3 show that the statistical model performs below other more recent algorithms. Thus, notwithstanding, its superior predict performance as compared to AAM and KAM, the sAM age estimation cannot be regarded as the gold standard for evaluating our age progressor. Hence, there is a need to investigate further, with a view to achieving an algorithm that minimises the estimation error even further.

6.7 Summary

In this chapter, a supervised appearance model (sAM) which improves on the traditional AAM was proposed. When used for facial feature extraction, the model describes the face with very few components. For instance, it required only 13 components to effectively represent FGNET-AD faces as opposed to AAM and KAM which require a large number of parameters. When used for age

estimation, the sAM based estimator achieved 5.49 mean absolute error which is better than most algorithms that used AAM for feature extraction. This proves the predictive power and superior dimensionality reduction ability of the sAM. Additionally, sAM provides an avenue for face reconstruction, thus, in the future, researchers can investigate using sAM for automatic facial age progression.

However, the supervised statistical model performs below more sophisticated state-of-the-art algorithms. Hence, if used for evaluating the performance of age progression methods, it might introduce much bias. Hence, there is a need to explore further with a view to getting an age estimation algorithm that has lower estimation error. Another obvious finding is that the deep neural networks despite their staggering performance in other computer vision applications fall below the enhanced BIF [69] when used for age estimation. It is presumed the problem encountered by the deep network (CNN) is the small size of the training dataset. To this end, ways of optimising CNN is investigated, especially for small datasets such as the FGNET.

7 Age Estimation using Deep Learning

In this chapter, more accurate age estimation is achieved by facial feature extraction using the learned weights of a pre-trained convolutional neural network. Thereafter, the proposed algorithm is used to evaluate the performance of the age progression frameworks that were proposed in earlier parts of this thesis.

7.1 Introduction

In recent years, convolutional neural networks (ConvNets or CNNs) have had a great impact on computer vision and machine learning fields due to their ability to learn complex features using nonlinear multi-layered architectures [79]. Although originating in the early 1990s, ConvNets were forsaken by the research community due to the assumption that feature extraction using gradient descent will always over fit as a result of local minima [79]. However, its remarkable success in the ImageNet competition of 2012 [149] altered the negativity associated with them. Today, state-of-the-art deep models are used in almost all computer vision applications including, but not limited to, detection [150], recognition [151], classification [152], and information retrieval [153], researchers have also attempted to solve the problem of age estimation using CNNs [74]. However, CNNs used in [74], were unable to outperform state-of-the-art algorithms when evaluated on the FGNET-AD.

Actually, age estimation using biologically inspired features (BIF) [69] have maintained state of the art results since 2010. The method involves convolving an input image with a bank of multi-orientation and multi-scale Gabor filters. After which a pooling operation is used to downscale the huge feature

dimension [154]. Although deep neural networks have become the de-facto standard models for image understanding, the failure of [74] to outperform [69] on FGNET-AD can be attributed to the small size of the dataset, as well as shallow nature of the architecture proposed in [74]. Thus, it is no surprise that most recent researchers [75], [77], [78] that followed on, failed to experiment and evaluate their CNN models using the FGNET-AD database. Moreover, all these researchers built their CNN models from the scratch, thus utilising huge datasets and deploying huge computational power.

Interestingly, research has shown that ConvNets efficiently learn generic image features [79], [155]. Thus, these features can be used directly with simple classifiers to solve computer vision problems. This approach known as off-the-shelf feature extraction has been used by several researchers [155]–[157] to achieve promising results on computer vision tasks. As a matter of fact, researchers advise that, rather than training CNNs from scratch, transfer learning should be the first approach to solving a computer vision task [157]. Likewise, some studies suggest that, for a dataset with a small number of images, the off-the-shelf feature extraction technique outperforms training a network from scratch [158].

However, despite numerous works conducted using off-the-shelf features, focus has been concentrated only on object classification, detection, segmentation, and instance retrieval. Thus, the technique has not been exhaustively applied to the problem of age estimation. As a result of this fact, this chapter attempts to bridge the gap by using very deep off-the-shelf CNN features to build an automatic age estimator that transcends on both small and huge datasets,

thereby presenting an avenue for comparing to previous algorithms that were tested on FGNET-AD. By avoiding to build a new model from scratch, transfer learning (off-the-shelf features) will be used to extract ageing features. Then, a suitable dimensionality reduction algorithm will be used to reduce the size of the extracted features, afterwards age-pattern learning will be conducted using a suitable regression algorithm. Using FGNET-AD, a thorough evaluation of the proposed age estimation technique shall be conducted. Furthermore, the algorithm is compared to the works of other researchers by testing it using the Morph album II dataset; this is because recent researchers that use CNNs, mostly utilised Morph album II for performance evaluation. Finally, the age estimation model is deployed as an evaluation tool to assess how best the age progression algorithms proposed in this thesis attain the intended age.

It is worth noting that, in addition to the initial goal of developing a suitable age estimator that can be used to evaluate a progression performance, this chapter also answers the question of whether there is a need to build new CNN models for every task at hand, or to transfer learned features especially when there is limited labelled data. Furthermore, this chapter also explores, analyses and evaluates which layer of the existing pre-trained model is most suitable to use for feature extraction.

7.2 Convolutional neural network (ConvNet/CNN)

7.2.1 Background

Supervised learning is the most common form of machine learning. In order to appreciate the concept, consider the problem of building a system that can classify banknotes as real or counterfeit. One will collect a large dataset of

images (real and fake notes), then at training time, the machine is shown a photograph and its category, the machine then produces an output in the form of two scores one for each category. Ideally, the output from the machine is assumed to be good if it gives the best score to the target category, however, one can be almost sure the machine can't produce such output prior to training. Fortunately, one can track the performance of the machine at the time of training, by computing a cost function that measures the difference (error) between the machine's output and that of the target scores. In order to reduce the error, the machine then perturbs its internal adjustable parameters, normally these parameters called weights define a mapping of the input to output. This training procedure of showing an image, estimating its scores, comparing to the target, and adjusting the weights goes on iteratively until a zero error, or very minimal acceptable error is achieved. Once training has been accomplished, the weights of the machine now have intelligent values which can be used to map a new (unseen) image of the banknote to a category with minimal error. In fact, the above scenario explains the basic working of a feed forward neural network.

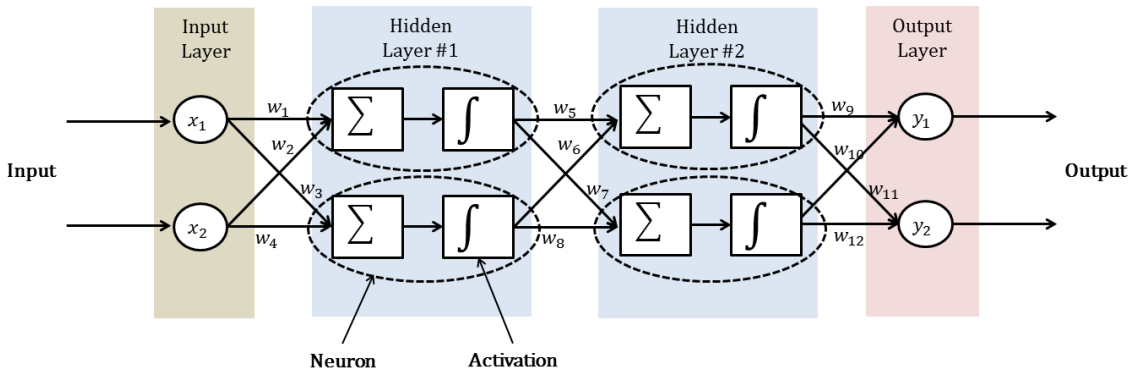


Figure 7.1: Feed forward artificial neural network.

A neural network like the one shown in Figure 7.1 can perform the stated cognitive task of classifying banknotes, only if trained to do so. In the example of Figure 7.1, the network has 2 inputs, 2 hidden layers with 4 neurons and 2 outputs, so it has $(2^2 + 2^2 + 2^2) = 12$ connections; consequently, each connection has a weight. As described in the example above, training involves iterative tweaking of the weights based on the output error. It is also obvious that every neuron from layer l is connected to the output of every neuron from layer $l - 1$, this is the key attribute of a fully connected neural network (NN), and it is called a feed-forward NN since the output of a layer becomes the input of the next layer. Passing an input and getting the predicted output is called the forward pass. This involves computing the total net input to each hidden layer neuron, pass the result through a non-linear (activation) function, then repeat the process with the next layer neurons. Various activation functions have been used in the literature [79], today, the rectified linear unit (ReLU) is the most popular activation function, which is simply ramp function $f(z) = \max(z, 0)$. During the backward pass, weights are tuned to minimise the error; this is achieved by a technique known as back propagation [159]. The procedure computes the partial derivative of the error with respect to weights, achieved by working backward. This computation then indicates by what amount the error decreases or reduces as a result of small change in the weights. Subsequently, the weights are adjusted in opposite direction of the computed gradient. After adjustment of the weights, the output error changes, thus before the next iteration, the partial derivatives have to be recomputed once again. Due to a huge number of parameters involved in a neural network, the algorithm often over-fit [160]; a phenomenon in which the model performs excellently on training dataset (classification with minimum error), but fails to generalise on a

new unseen dataset. Methods for avoiding over-fitting include large training dataset, stopping the training as soon as performance on a validation set starts to get worse, regularization, and dropout [160].

A Convolutional Neural Network (CNN) is a type of artificial neural network that takes into consideration the spatial structure of the input data. To ensure shift and distortion invariance, CNNs combine three architectural ideas: shared weights, local receptive fields, and spatial or temporal subsampling [161]. Weight sharing refers to the procedure of applying repetitive (shared) tiles of neurons across space. This results in lesser parameters to optimize, and consequently, increase in learning efficiency. As it is impractical to connect neurons to all neurons in the previous volume, especially in large dimensional data (for instance images), local receptive field ensures the connection of neurons to only local regions in the input volume. Subsampling and local averaging enhance the efficiency of the algorithm by decreasing the resolution of the feature map, and therefore, decreasing the sensitivity of the output to shifts and distortions. In general, CNNs take the shape of a 3D structure tensor, having $W \times H \times D$ dimensions where W (width) and H (height) are spatial dimensions whereas D is the feature dimension. Specifically, the structure of CNNs makes them most appropriate for image, speech and time series tasks [161]. More specifically, the algorithm earned its name due to the convolution operation that is used to apply a set of weights to the input.

Just like the feed forward neural network, CNN can be defined as a function q composed of a sequence of simpler functions p [162]. Given by,

$$q = p_l \circ p_{l-1} \circ \cdots \circ p_1 \tag{7.1}$$

where each function p defines a mapping of input x_{l-1} of the previous layer to its output x_l expressed as,

$$x_l = f(\sum w_l x_{l-1} + b) \tag{7.2}$$

where f is an activation function, w_l are weights and b is the bias.

CNNs were first discovered in the 1990s [161], unfortunately, despite their breakthrough in document recognition [163], [164], they were forsaken by researchers due to the assumption that neural networks will always overfit or get trapped in local minima [165]. However, research [165] has shown that local minima is not necessarily that much of a problem; the difference in performance is only slightly affected when the local minima is minimally non-optimal. Moreover, a conventional method of avoiding local minima is by perturbing the stability of the algorithm, so that it suddenly hops out of the local optimum. Varying the learning rate by gradually and repeatedly increasing and decreasing it reduces the stability of the algorithm [166], thus giving the algorithm the ability to jump out of a local optimum. Additionally, the stochastic gradient descent (SGD) [167] optimization algorithm also introduces some sort of noise or randomness which helps the algorithm to escape from local minima. In essence obvious difference between ConvNets of today and those of the 90s include; availability of extremely large datasets such as the ImageNet [168], faster computation realised by parallel processing ability of GPUs, advanced techniques for initialising weights at the start of training [169], and the use of simple and easy to differentiate activation functions [170].

7.2.2 Architecture of a ConvNet

The structure of a typical ConvNet is comprised of multiple layers (see Figure 7.2), which fall into three broad categories, convolution layer (CONV), subsampling layer (POOL), and a fully connected layer (FC). Furthermore, the activation function can also be considered as a layer in the architecture. Usually, a combination of the aforementioned, layers are arranged in a specific manner with the sole goal of transforming the input of the networks into a useful representation that gives an output.

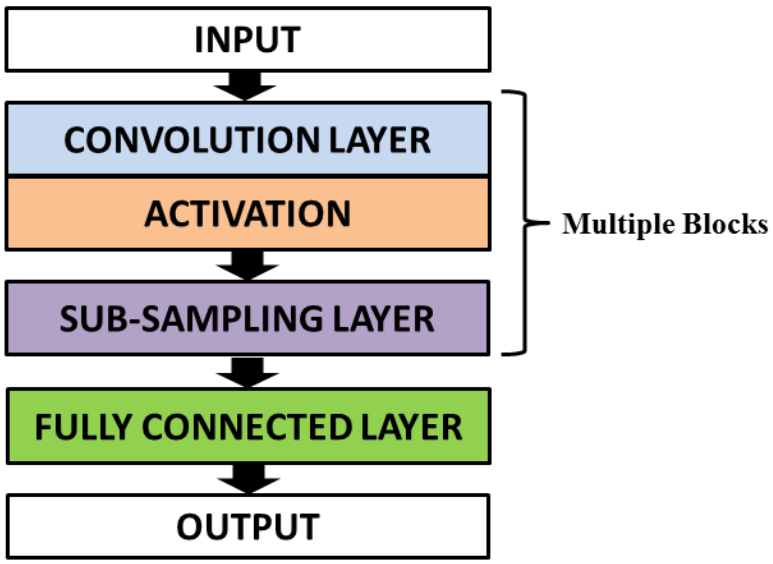


Figure 7.2: Structure of a typical ConvNet.

CONV Layer: This is the fundamental building block of ConvNet, as stated earlier it derives its name from the mathematical (convolution) operation performed. This layer computes a dot product between the weights of neurons and a small region of the input volume. The neurons are arranged as a stack of 2-dimensional filters/kernels that extend the depth of the input volume, hence

they are 3D structured. During the forward pass, each kernel is convolved across the width and height of the input volume to produce a 2D feature map (as shown in Figure 7.3). Thus, these feature maps are the outputs of the convolution operation at each spatial operation. In comparison to the feed forward NN, here filters represent neurons which activate when they come across visual features such as edges. As discussed earlier, ConvNets use local connectivity to reduce complexity, hence each neuron is connected to a local region whose spatial dimension is defined by the filter size known as the receptive field of the neuron, and its depth is always equal to the depth of the input volume.

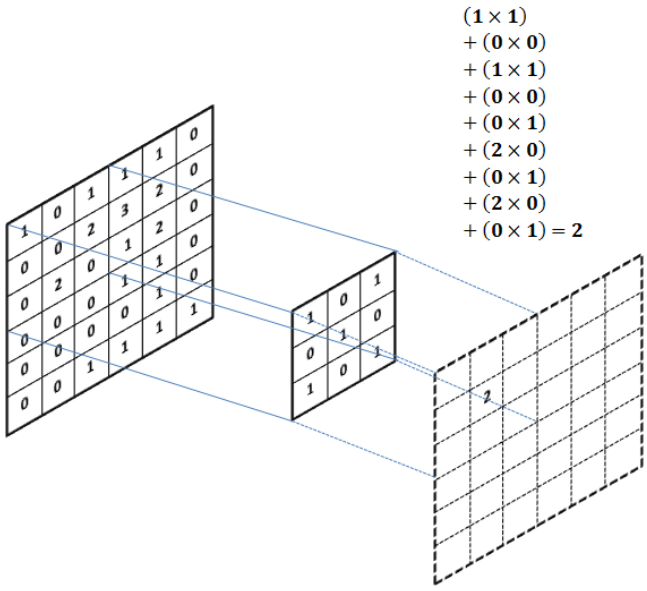


Figure 7.3: Convolution operation.

Hence, for a $256 \times 256 \times 3$ input image, if the receptive field is 3×3, then each neuron in the CONV layer will have a total of $3 \times 3 \times 3 = 27$ connections, and 1 bias parameter. Obviously, the connectivity is spatially local but full along the input depth. Subsequently, the size of the feature map (i.e. output) is computed using three hyper-parameters, depth, zero padding, and stride. Depth refers to

the number of filters deployed. The more the number of filters, the more the information retrieved since each filter learns to look for a specific feature. Stride defines a pattern used to slide the filter across the input, $S = 1$ means the filter will be moved one pixel at a time across the input. Zero padding defines the number of zero pixels placed around the input volume in order to preserve the spatial size of the output volumes. One can compute the spatial size of the output using,

$$O = \{\frac{I - F + 2P}{S}\} + 1 \quad (7.3)$$

where I is the spatial size of the input, F the filter size, P number of zero paddings and S the stride. Hence, if applied to input images of size $[224 \times 224 \times 3]$ and assuming; neurons having receptive field of 3×3 size, depth $K = 64$, a single stride $S = 1$, and zero padding $P = 1$, then one gets $\frac{224 - 3 + 2}{1} + 1 = 224$. This means that the output volume of this particular CONV layer will have a size $[224 \times 224 \times 64]$. Consequently, there will be $224 \times 224 \times 64 = 3211264$ neurons each having $3 \times 3 \times 3 = 27$ weights and 1 bias. Interestingly, rather than having 3211264×27 weights and 3211264 biases, the concept of weight sharing makes all the neurons on one slice to share the same weight and bias. Hence, the number of weights and biases drastically reduces to 1728 and 64 respectively.

Activation Layer: In neural networks, activation function plays a significant role of introducing nonlinearity to the output of a neuron. Introducing this nonlinearity makes the neural network a universal function approximator, thereby, giving it

the ability to understand various types of relationships. The most effective and commonly used activation function for ConvNets is the rectified linear unit (RELU) [171]; this involves element-wise application of a zero thresholding function $f(x) = \max(0, x)$ where x is the input of the neuron. Compared to other activation functions, ConvNets with ReLUs train several times faster [149]. The activation layer does not introduce additional parameters to its input, it also does not change the dimension of the input. In the architecture, activation layers are placed after every CONV layer. Additionally, networks with more than one FC layer also deploy it with the exception of the last fully connected layer.

POOL Layer: Pooling layers are usually inserted between successive CONV layers. Their main function is to consistently reduce the number of parameters and consequently decrease computation complexity of the network by reducing the spatial size of the feature maps. Hence they summarise the output of neighbouring neurons [149]. For every 2D slice of the feature map, the most common type of pooling operation called MAX-POOLING usually takes the maximum of each 2 x 2 region, thus discarding 75% of the activations as shown in Figure 7.4.

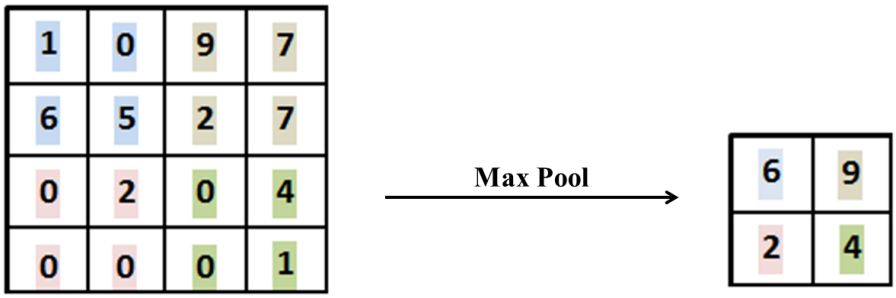


Figure 7.4: Max-pool with 2×2 filter having stride of 2.

In a nutshell, pooling operation does not introduce new parameters; rather it leads to shrinkage of the first and second dimensions of the feature map. The operation takes two parameters, stride S and spatial dimension F. Hence the pooling operation reduces a feature map from $W_1 \times H_1 \times D$ to $W_2 \times H_2 \times D$ dimension. Here W_2 and H_2 are computed via,

$$W_2 = \frac{W_1 - F}{S} + 1,\ H_2 = \frac{H_1 - F}{S} + 1, \tag{7.4}$$

Interestingly, this operation introduces translational invariance with respect to elastic distortions [172].

Fully Connected (FC) Layer: Fully Connected layer has neurons that have full connection to the previous layer's activation, unlike the CONV and POOL layers, FC have a 2D dimension. They are typically configured to output the networks predicted label/classes hence FC is usually the last layer of the network. In the work that won the 2012 ImageNet Large Scale Visual Recognition Competition (ILSVRC) [150], 3 FC layers were used, and since then this has been the rule of thumb among most researchers. Intuitively, flattening the 3D feature maps at the end of the computation gives us an avenue for interpreting the learned spatial invariant features.

7.2.3 ConvNet Layer Pattern

The most popular arrangement used by researchers [149], [173], [174] starts with the image-input layer, and ends with an FC (decision) layer, in between these two are repeated stacks of CONV-RELU layers followed by POOL layers, then a few FC-RELU layers. This layer pattern can be described mathematically as,

$$INPUT \implies N\{M(CONV \implies RELU) \implies POOL\} \implies K(FC \implies RELU) \implies FC$$

Usually, the number of CONV-RELU layers that appear before POOL are within the range $0 < N < 4$, and the combinations variables M and K are usually greater than 1. A typical example is the popular VGG16 model [173] shown in Figure 7.5 below.

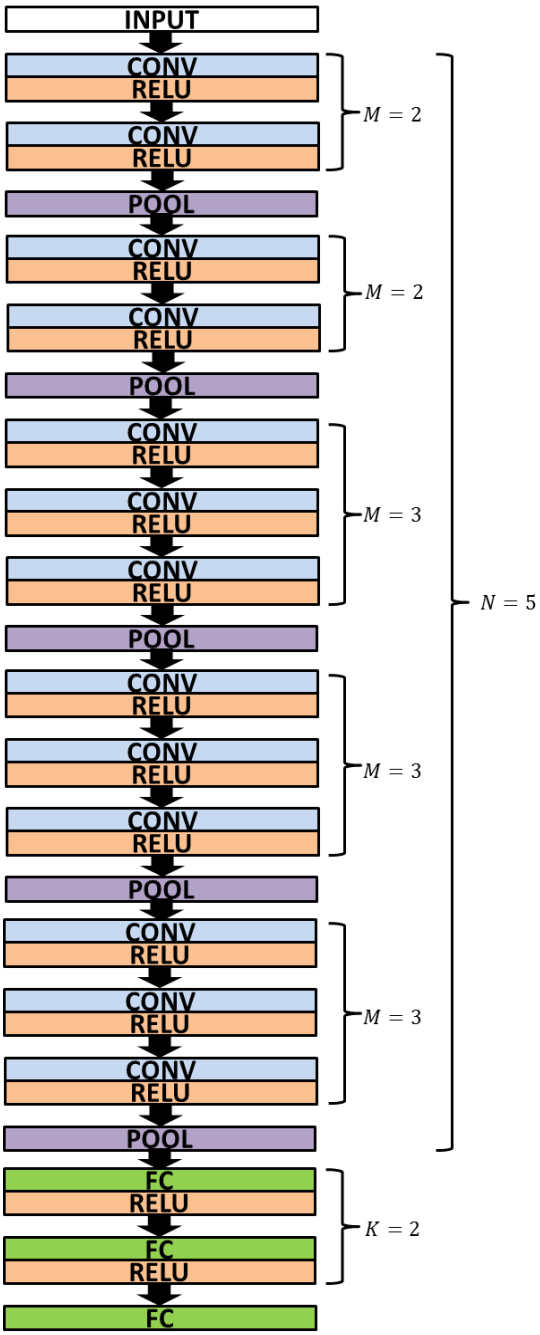


Figure 7.5: Typical ConvNet Layer Pattern.

7.2.4 Methods of Training ConvNets

Generally, there are three ways of deploying ConvNets; training a network from scratch, fine tuning an existing model, or using off-the-shelf CNN features [157]. The latter two approaches are referred to as transfer learning [175]. Since

training ConvNets from scratch, using the back-propagation algorithm involves the automatic learning of millions of parameters, this approach requires an enormous amount of data, often in millions [162]. More so, this data-hungry nature of ConvNets consequently demands large computational power. Furthermore, the procedure involves the adjustment of several hyper parameters. Thus, people rarely train an entire network from scratch.

Fine tuning involves transferring the weights of the first n layers learned from a base network to a target network [176], and then continuing the backpropagation using the new dataset. Hence, the target network is trained using the new dataset for a specific task, usually different from that of the base network. Fine tuning is recommended when the new dataset is moderately large (tens to hundreds of thousands) and very different from the base network's dataset. Using the weights of the old network to initialise helps the back-propagation algorithm, and so leading to relatively fast automatic learning of more specific features.

In situations where the dataset is quite small (few hundreds), even fine tuning the weights results to over-fitting. However, since ConvNets efficiently learn generic image features [79], [157], it is then possible to directly use a trained network as a fixed feature extractor. Hence, features from new data are extracted by projecting them on to activations of a specific layer of the pre-trained network. Thereafter, the learned representations are fed into simple classifiers to solve the task at hand. This approach known as off-the-shelf

feature extraction has been used by several researchers [155]–[157] to achieve promising results.

7.3 Our Approach

In this chapter, off-the-shelf ConvNet features are utilised. This is due to the relatively small size of the FGNET-AD. More precisely, the VGG-Face model [1] is utilised, due to its depth, reported excellence, and the similarity of the data it was trained on, to the data used in this research (i.e. images of the human face).

7.3.1 VGG-Face Model

VGG-Face [1] developed at Oxford University's Visual Geometry Group (VGG), is the application of the very deep ConvNet architecture VGG-16 [173]. It is a publicly available model that was trained using 2.6 million face images of 2622 unique subjects. The model is configured to take a fixed sized $[224 \times 224 \times 3]$ RGB image as an input; as a form of pre-processing, all the images used are center-normalised. The network is made of a stack of 13 convolutional layers with filters having a uniform receptive field of size 3×3 and a fixed convolution stride of 1 pixel. As shown in Figure 7.6 groups of these convolution layers are followed by five max-pooling layers. Finally, the CONV layers are then followed by three fully connected layers; FC6, FC7, and FC8. The first two have 4096 channels, while FC8 has 2622 channels which are used to classify the 2622 identities. In addition to center normalisation, the model's implementation also incorporates 2D alignment. Parkhi et. al [1] have shown that the model outperforms Google's FaceNet [177] and Sun et. al's DeepID [178] when tested on YouTubeFaces[179].

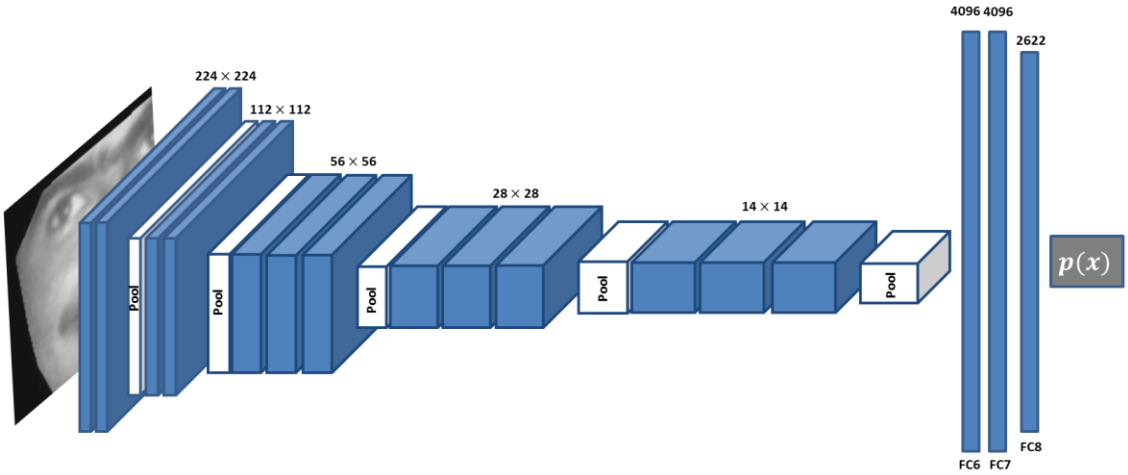


Figure 7.6: Architecture of the VGG-Face model.

7.4 Feature Extraction and Pattern Learning

For face representation weights from different layers of the VGG-Face model are used to extract deep features. Dimensions of the resulting features are then reduced before using regression for age estimation.

7.4.1 Feature Extraction

Given an input image $\boldsymbol{X}_0$ represented as a tensor $\boldsymbol{X}_0 \in \mathbb{R}^{H \times W \times D}$ where H is the image height, W is the width and D the colour channels, and a pre-trained L layered ConvNet expressed as a series of functions $q_L = p_1 \rightarrow p_2 \rightarrow \cdots p_L$. In order to fully investigate and evaluate which layer yields the best age descriptor, the activation of five layers; the last two convolution layers (conv5_2, conv5_3), the last max-pool layer (pool5) and first two fully connected layers FC6 and FC7 of the VGG-Face model are used as separate feature channels. The choice of layers has been restricted to the top 5 layers, because going further down yields extremely huge dimensions that will result in no significant gain even after reducing the dimension.

7.4.2 Dimensionality Reduction and Regression

Due to large dimensions of the extracted features, ranging from 4096 in FC7 to 100352 in conv5_2, there is a need to reduce the feature size thus removing redundant information. Moreover, it is a well-known fact that, for n observations and p features, the regression estimate is actually not well-defined in a situation where $p > n$.

In the past, researchers used PCA for dimensionality reduction. However, due to obvious problems of PCA that were mentioned in earlier chapters, partial least squares regression (PLS) is used to simultaneously reduce the dimension and regress. Hence, the relationship between the extracted features $\boldsymbol{X}$ and the vector of ages $\boldsymbol{Y}$ is formulated as,

$$\boldsymbol{Y} = \boldsymbol{X}\beta^{PLS} + \boldsymbol{b}. \tag{7.5}$$

where $\boldsymbol{b}$ is the intercept.

7.5 Experiments I: Age Estimation Evaluation (a)

Here, the performance of the age estimation procedure is evaluated using the same metrics outlined in 6.6.1. As an initial evaluation, features extracted using different layers of the VVG-Face model are compared in order to identify which weights of the deep network carry the most optimal ageing information. Next, the performance of the proposed algorithm is compared to state-of-the-art methods. To enhance the specificity of the technique, all images are first cropped to a size of 224×224, then a data pre-processing step is deployed;

this will be discussed shortly. Finally, the extracted features are fed into a PLS-age-learner to achieve age prediction.

7.5.1 Image Pre-processing

For this experiment, all the test images were aligned using landmark annotations provided with the FGNET-AD dataset. Furthermore, their backgrounds were removed to increase image purity (refer to Figure 7.7). Thereafter, data augmentation was conducted; this is a popular technique used to increase data size during the training phase. As a result of augmentation, each image was responsible for the generation of 7 additional images achieved via random cropping and warping to the mean shape, as shown in Figure 7.7.

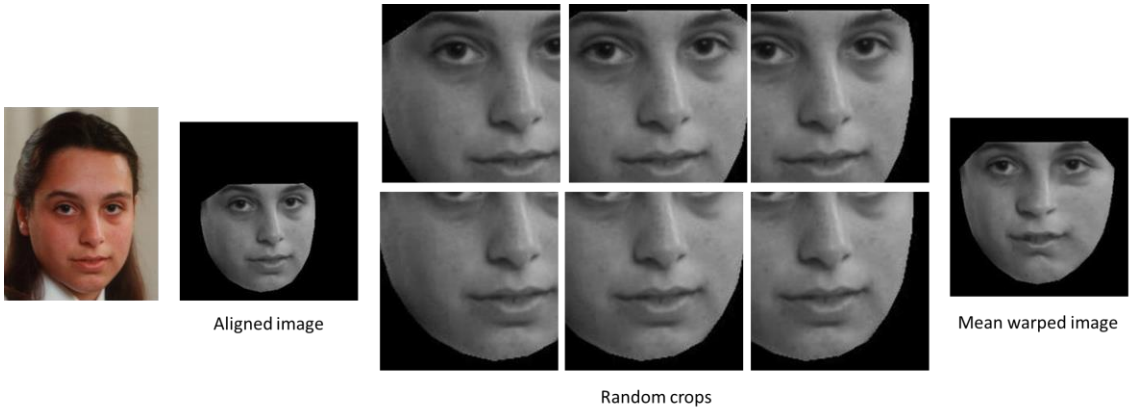


Figure 7.7: Image Pre-processing pipeline.

7.5.2 Performance Evaluation

Utilizing the procedure described in the section above, five sets of estimations were conducted. Each estimation was performed by extracting features using one of the five layers of the VGG-Face model; conv5_2, conv5_3, pool5, FC6, or FC7 layers, after which they were fed into equation (7.5). In all the

experiments, the numbers of PLS latent variables were chosen via cross validation.

To evaluate the performance of the estimation procedure, two metrics, the Mean Absolute Error (MAE) and Cumulative Score (CS), were used. Comparison of the performance of the five ConvNet features represented in

Table 7.1 shows that conv5_2 activations give the most minimal estimation error. It is also obvious that the performance degrades as one moves higher along the hierarchy. This suggests that generic features learnt from intermediate layer activations carry more ageing information than the latter layers that are more specific to the problem of face identification. The dimensionality reduction capability of PLS is also remarkable, as it reduced thousands of features to just a few (18) latent-variables.

Table 7.1: Evaluation of features extracted from different ConvNet layers.

Layer	**Latent Variables**	**MAE**	**CS <10 years**
conv5_2	**18**	**2.70**	**100%**
conv5_3	18	2.83	99.01%
pool5	18	2.97	98.05%
FC6	18	3.89	96.31%
FC7	18	5.51	84.21%

Next, the performance of the best performing ConvNet feature was compared to state-of-the-art algorithms. MAEs and CS presented in Table 7.2, further shows the excellence of the proposed method. This proves that carefully choosing the activations of pre-trained ConvNets, coupled with an effective regression algorithm, one achieves superior results despite the size of the dataset. To that effect, the proposed method has surpassed all state-of-the-art results with a clear gap.

Table 7.2: Comparison of our best result to state-of-the-art algorithms on FGNET-AD.

Method	**MAE**
BIF[68]	4.77
C & H BIF [180]	4.60
OHR [146]	4.48
LSR [147]	4.38
CNN [72]	4.22
BI. AAM [148]	4.18
EBIF [69]	3.17
sAM	5.49
Proposed	**2.70**

7.6 Experiments I: Age Estimation Evaluation (b)

Since some of the recent works on estimation use Morph album II [73] rather than FGNET-AD, it is then ideal to evaluate the proposed algorithm using the Morph II dataset.

7.6.1 About the Dataset

Morph Album II is the largest publicly available longitudinal face database, consisting of 55,134 images of 13,000 individuals. The age distribution lies between 16 - 77 years, with a median age of 33 years. Each subject has up to 4 images which were collected within a period of 4 years. The database contains people from different ethnicities, with various head poses and facial expressions. Furthermore, the image quality has varying scale, rotation, and translation as well as illumination.

Due to the size of the Morph dataset, LOPO evaluation approach is not ideal. Hence data splitting protocol proposed and used by [5] is adopted. The protocol entails splitting the dataset into three (3) non-overlapping partitions; S1, S2, and S3 (Others). Then, the algorithm is trained and tested twice. Firstly, S1 is used for training after which test is conducted on a combination of S2 and S3 partitions. In a second run, S2 is used for training, while reserving S1 and S3 for testing. Finally, results of the two tests are averaged. The rationale for splitting is to reduce the effect of different ethnicities on the algorithm; partition S1 has Caucasian subjects, while S2 contains African American subjects, since it's presumed these two races age in a different manner, and the fact that there are sufficient images for both races in the dataset makes it possible to train two models. After Guo & Mu [5] proposed the S1, S2, and S3 protocol, researchers have regarded it as a benchmark for evaluation on the Morph II dataset.

7.6.2 Image Pre-processing

Automatic image alignment was deployed via Zhu and Ramanan's [181] algorithm, hence faces were detected, annotated and aligned on the fly. Due to the huge size of the dataset, data augmentation was not applied, additionally, the background removal step was omitted. Here, the intent is to fully investigate how well the ConvNet feature coupled with PLS estimation procedures fare in the absence of extensive preprocessing.

7.6.3 Performance Evaluation

In this second experiment, two sets of tests were conducted, by utilising aligned and unaligned images, denoted as *wAlg* and *woAlg* respectively. Besides the above mentioned preprocessing omissions, all other steps used in the first experiment for feature extraction and estimation were repeated. Eventually, various estimations were conducted using the same five ConvNet activations. Comparison of the estimation results presented in Table 7.3 and Table 7.4

corroborate findings of the first experiment; once again conv5_2 activations give superior performance. The results also show that image alignment increases the performance of the technique.

Table 7.3: Evaluation on Morph II $woAlg$.

Layer	Tr. Set	Latent Vars.	MAE	Avg. MAE	CS < 10 years
conv5_2	**S1** **S2**	**17** **17**	**3.93** **3.91**	**3.92**	**96.71%**
conv5_3	S1 S2	17 17	3.95 3.93	3.94	96.61%
pool5	S1 S2	17 17	4.06 4.03	4.05	96.06%
FC6	S1 S2	24 24	4.33 4.29	4.31	94.32%
FC7	S1 S2	24 24	4.50 4.51	4.51	93.26%

Table 7.4: Evaluation on Morph II $wAlg$.

Layer	**Tr. Set**	**Latent Vars.**	**MAE**	**Avg. MAE**	**CS < 10 years**
conv5_2	**S1** **S2**	**17** **17**	**3.84** **3.82**	**3.83**	**96.82%**
conv5_3	S1 S2	17 17	3.87 3.86	3.87	96.75%
pool5	S1 S2	17 17	4.01 3.97	3.99	96.18%
FC6	S1 S2	24 24	4.27 4.25	4.26	94.43%
FC7	S1 S2	24 24	4.45 4.45	4.45	93.40%

Finally, comparison to state-of-the-art algorithms (presented inTable 7.5) further shows the excellence of the proposed method. As can be seen, the best performing activation considering image alignment, outperforms most of the state-of-the-art algorithms.

Table 7.5: Comparison to state-of-the-art algorithms on Morph II database.

Layer	**Tr. Set**	**MAE**	**Avg. MAE**
FMBS [182]	S1	3.96	3.99
	S2	4.01	
KCCA [5]	S1	4.00	3.98
	S2	3.95	
KPLS [142]	S1	4.21	4.18
	S2	4.15	
3-step [183]	S1	4.44	4.45
	S2	4.46	
BIF [68]	S1	5.06	5.09
	S2	5.12	
Proposed	**S1**	**3.80**	**3.83**
	S2	**3.76**	

In general, both experiments conducted on FGNET-AD and Morph II datasets prove the power of ConvNet features and their efficiency especially after conducting meticulous pre-processing steps such as alignment, background removal, and augmentation. Our findings further show that using an appropriate regression algorithm, the extracted features have the potential of out performing even end-to-end learned networks. Having achieved minimum estimation errors, the proposed age estimation procedure can then be used as an automatic tool for assessing the performance of age progression techniques, i.e. their ability to generate faces that attain the intended age.

7.7 Experiment II : Age Progression Evaluation

In this section, a ConvNet-based age estimator is used to develop a metric for evaluating the performance of age synthesis algorithms. It is worth mentioning that the metric presented in this section complements other performance measures that were used in chapters 3, 4 and 5. It acts as a machine based evaluation of how well the generated faces meet the intended age.

7.7.1 Evaluation Procedure

Given an image I, progressing it to a new age yields a synthetic image I', the ability of the algorithm to render well-aged face I' that attains the expected age can be measured by comparing it to the ground truth image of the same subject I'' at that same age. Precisely, the age attainment test can be done by measuring how much the estimated ages of I' and I'' differ. Hence, assuming the estimated age of ground truth image I'' to be y and the estimated age of the synthesised face I' to be y' MAE and CS can then be used to measure the performance progression algorithms. The proposed procedure is illustrated pictorially in Figure 7.8.

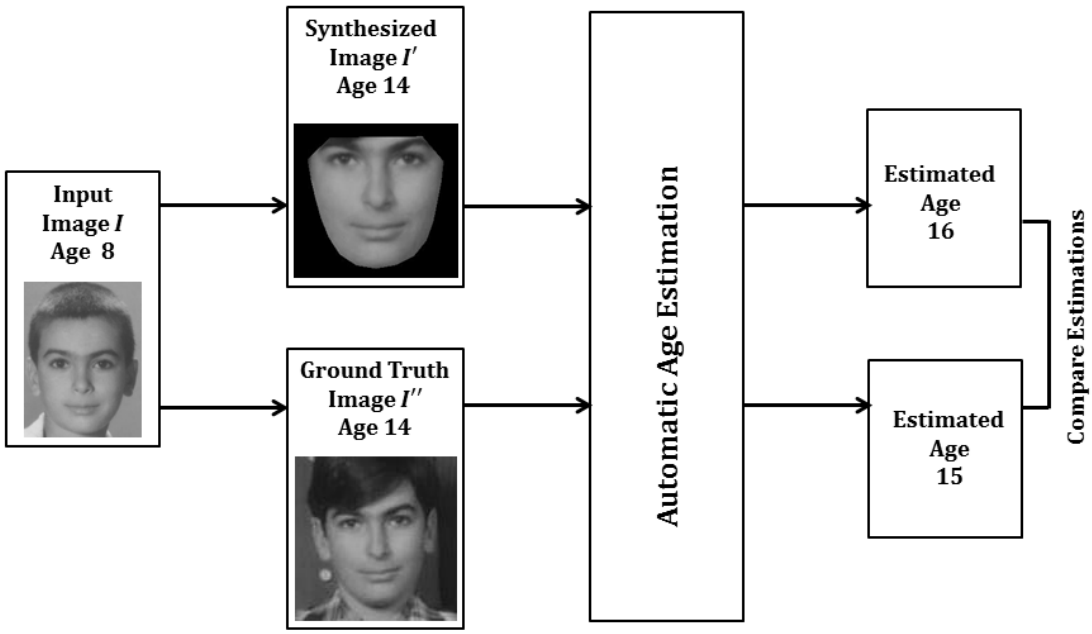


Figure 7.8: Machine-Based Age Attainment Test.

7.7.2 Results

In line with previous age progression assessments, FGNET-AD was used for the age attainment test. Here, the test was conducted on all 8 age progression algorithms; Lanitis, AAM-OLS, AAM-PLS, AAM-sPLS, KAM-G, KAM-L, KAM-S and Hybrid techniques. For each of the 82 unique subjects contained in the database, two images were selected, an image to progress and a ground truth image at the progressed age to compare to. Just as stated earlier, out of 82 images rendered, 50% of the progressions were from adult to young faces and the remaining 50% for the vice versa. Table 7.6 presents the computed MAEs and CSs for experiments conducted on the 8 algorithms.

Table 7.6: Comparison of Age Attainment Test (MAEs & CS).

Algorithm	**MAE**	**CS < 10 years**
Hybrid	4.91	88.90
KAM-G	6.07	80.02
KAM-L	6.13	79.50
KAM-S	6.22	79.01
AAM-sPLS	6.54	76.03
AAM-PLS	6.61	75.25
AAM-OLS	7.54	74.31
Lanitis	7.23	73.40

From the results shown in above, the hybrid algorithm having an MAE of 4.91 years clearly has the least estimation error which indicates its ability to achieve the intended age. It can further be observed that the hybrid technique is seconded by the KAM-G method with MAE of 6.07 years, and the least performance was recorded by AAM and Lanitis algorithms.

7.8 Summary

In this chapter, facial feature extraction using weights of pre-trained ConvNet was extensively explored. Using activations from different layers of the VGG-Face model, experiments were conducted on both FGNET-AD and Morph Album II databases. With the simultaneous dimensionality reduction capability of PLS, it has been demonstrated that promising results can be achieved without having to train a ConvNet from scratch, specifically for age estimation. Having achieved excellent age estimation with minimal prediction errors, the

estimator was then used to conduct machined based evaluation of age progression algorithms that were proposed in earlier chapters of this thesis. The results obtained clearly show the superiority of the hybrid age synthesis technique, this was then followed by the KAM based methods. These results corroborate the findings of the earlier chapters of this thesis.

8 Conclusion and Future Work

This thesis has described the development of automatic age progression framework that is robust to noise, illumination variation as well as varying facial expressions. Algorithms introduced between chapters 3 to 7 of this work have successfully answered all the research questions outlined at the beginning of the thesis. This chapter summarises the main achievements of the research. Furthermore, it also highlights directions for future research.

8.1 Conclusion

In this work, problems of automatic facial age synthesis and estimation were addressed. Specifically, ways of tackling problems associated with existing age progression techniques were investigated and implemented. Evaluation of the algorithms proposed across chapters 3 to 7 show progressive improvement on existing techniques. Particularly, realistic 2D face images were aesthetically synthesised at different ages. This incorporated methods of handling image noise, varying facial expressions, poor texture quality as well as reconstruction artefacts. With a view to fully evaluating the proposed synthesis algorithms, robust age estimation procedure that outperformed state-of-the-art algorithms was also implemented and utilised as an assessment tool.

In chapter 3, a mathematical procedure for progressing facial images was presented. As an initial part of the procedure, conventional AAM was used for facial feature extraction. Then, using various linear regression models, the formula which computes inverse mapping of the relationship between ages and face images was used to render realistic face images. Extensive evaluation of the method's ability to preserve the identity of the subject while attaining the

intended age showed promising results when used to age neutral, frontal, good quality face images. To that effect, the algorithm was used to progress the image of Ben Needham, a British toddler that mysteriously went missing over two decades ago. However, our findings also showed that like other existing face synthesis algorithms, the proposed method was not robust to image noise and the effect of varying facial expression.

In chapter 4, a nonlinear variant of the AAM was proposed. Termed kernel appearance model (KAM), the algorithm which takes advantage of nonlinear principal component analysis was used to implement a face synthesis framework that performed image denoising as well as facial expression normalisation. A thorough evaluation of the technique using various kernel functions showed that the Gaussian radial basis function is best suited for the task. Furthermore, results of performance evaluations showed significant improvement over the AAM-based algorithm that was proposed in chapter 3. Also to illustrate the real application of facial age synthesis, the nonlinear age synthesiser was used to progress images of Mary Boyle; an Irish toddler that went missing over 3 decades ago. However, it was observed that despite handling noise and varying facial expressions, the KAM-based framework like its linear counterpart, generated images which had low texture quality, thus lacking valuable age related traits such as wrinkles and muscular tautness. It was further observed that both statistical ageing models at times suffered from reconstruction artefacts, such as inconsistent eye colouration and facial hair that appeared on toddler and feminine faces.

In chapter 5, a hybrid technique for augmenting problems associated with statistical models was implemented. The approach which builds upon images that were generated using the KAM framework deployed tiny skin patches retrieved from a large pool of images to boost the low resolution of age-progressed pictures. Furthermore, it was observed that the technique effectively corrected unrealistic facial artefacts. Extensive machine and human based evaluations of the ability of the algorithm to retain the people's identity showed that the method surpassed the previous techniques. Furthermore, our findings showed that human observers perceived most of the images rendered to have aged well as intended.

In general it will have been interesting to observe statistically, if rendering images at specific ages actually affected the performance of the algorithms. Unfortunately, due to relative small size of the FGNET database, this was not investigated.

In chapters 6 and 7 automatic age estimation was explored, so it can be employed as a tool to objectively quantifying the ability of the age progression algorithms to attain the intended age. Specifically, chapter 6 entailed the development of a supervised appearance model (sAM) which improved on both AAM and KAM especially when used in the context of prediction and classification. Utilising the excellent characteristics of PLS, the sAM captures shape and texture variations in a supervised manner. Hence age estimation conducted using sAM features had better estimation accuracy as compared to the two unsupervised models. However, when compared to other non-statistical algorithms the performance of the sAM-based age estimator lagged behind, hence, making it unsuitable for use as an age progression evaluation tool.

In chapter 7, age estimation via transfer learning was explored. Utilising activations of various layers of a pre-trained deep neural network, five sets of facial features were extracted and fed into a partial least squares regressor. After rigorous evaluation, it was observed that activations of the second to the last convolution layer of the VGG-Face neural network model carried the most ageing information. Thus, it achieved the least estimation error. Astonishingly, the proposed technique consistently outperformed state-of-the-art algorithms. Our findings showed that using off-the-shelf ConvNet features, age estimation results even outperforms CNNs that were trained from scratch for that particular task of age estimation. It can be deduced that the reason for the superiority of our approach is that learning is conducted twice; first unsupervised via the neural network and second in a supervised manner by the regression algorithm. This suggests that rather than conducting complex hand-engineered image representation or taking the route to development of a new neural network, deep features obtained from already pre-trained ConvNets should be the primary candidate for face representation in age estimation tasks. After implementing an efficient age estimator, the tool was then used to evaluate the age-attainment accuracy of the age progression algorithms that were proposed in chapters 3, 4 and 5. Our findings showed that the hybrid technique of chapter 5 achieved best result. This clearly corroborates other machine (identity) and human (identity and age attainment) based tests that were conducted in earlier parts of the thesis.

To sum up, this thesis addressed four main issues that affect existing age progression algorithms; complete reliance on training dataset, the effect of noise, distortion due to facial expression amplification, and face rendering

artefacts. Furthermore, rigorous evaluation techniques have been proposed, and it will be ideal for researchers in the future to use them. Additionally, the age estimation algorithm proposed in this thesis can be used in other areas of application such as security access control.

8.2 Future Work

Despite achievements of this work, it does leave room for future improvements. Hence, several future directions are available:

Face Synthesis via sAM: The sAM proposed in chapter 6 has been observed to outperform the AAM and KAM when used for estimation. It will be interesting to investigate ways of utilising the model for face synthesis. Since the PLS algorithm which is at the core of the model retains much age related information, it is hoped that age progression conducted using the model will give promising results.

Age Synthesis from Facial Profile View: To date, all face synthesis researches have been focussed on frontal or semi frontal images, however, in reality, images obtained in unconstrained environments are not always frontal. For instance, the only available image of a missing person can be a picture of his facial profile (side-view). Hence, a challenging problem worth exploring in the future is face synthesis from extreme viewing angle. Normally this can take one of two routes; automatic conversion of facial profiles to frontal images, or the generation novel images that are themselves side-views. Furthermore, this can lead to the development of age estimation algorithms for profile faces; to

the best of my knowledge, the first to attempt towards age estimation from the side-view of face images is that published as part of this research [184].

Face synthesis robust to all external factors: Although this thesis has gone a long way to tackle most of the factors challenging facial reconstruction, there still exist number factors that are not considered by the age progression frameworks presented. These include gender, ethnicity, diet, and life style. For instance, it has been well documented that gender and ethnicity affect ageing. To the best of our knowledge, no age progression research has been reported that utilised African faces. In the future, the formation of a database of African faces can be pursued. Thereafter algorithms proposed in this thesis can be evaluated on those African faces. More generally, a future problem worth investigating is how to incorporate all the aforementioned factors into the automatic age progression framework.

9 Reference

[1] O. M. Parkhi, A. Vedaldi, and A. Zisserman, “Deep face recognition,” in *British Machine Vision Conference*, 2015, vol. 1, no. 3, p. 6.

[2] Z. Zeng, M. Pantic, G. Roisman, and T. S. Huang, “A survey of affect recognition methods: Audio, visual, and spontaneous expressions,” *Pattern Anal. Mach. Intell. IEEE Trans.*, vol. 31, no. 1, pp. 39–58, 2009.

[3] E. Eidinger, R. Enbar, and T. Hassner, “Age and gender estimation of unfiltered faces,” *Inf. Forensics Secur. IEEE Trans.*, vol. 9, no. 12, pp. 2170–2179, 2014.

[4] H. Yan, J. Lu, W. Deng, and X. Zhou, “Discriminative Multimetric Learning for Kinship Verification,” *IEEE Trans. Inf. Forensics Secur.*, vol. 9, no. 7, pp. 1169–1178, 2014.

[5] G. Guo and G. Mu, “Joint estimation of age, gender and ethnicity: CCA vs. PLS,” in *Automatic Face and Gesture Recognition (FG), 2013 10th IEEE International Conference and Workshops on*, 2013, pp. 1–6.

[6] Q. Zhao, K. Rosenbaum, K. Okada, D. J. Zand, R. Sze, M. Summar, and M. G. Linguraru, “Automated down syndrome detection using facial photographs,” in *Engineering in Medicine and Biology Society (EMBC), 2013 35th Annual International Conference of the IEEE*, 2013, pp. 3670–3673.

[7] Y. Fu, G. Guo, and T. Huang, “Age Synthesis and Estimation via Faces : A Survey,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 11, pp. 1955–1976, 2010.

[8] G. Panis, A. Lanitis, N. Tsapatsoulis, and T. F. Cootes, “Overview of research on facial ageing using the FG-NET ageing database,” *IET Biometrics*, vol. 5, no. 2, pp. 37–46, 2016.

[9] A. M. Bukar, H. Ugail, and D. Connah, “Individualised model of facial age synthesis based on constrained regression,” in *Image Processing Theory, Tools and Applications (IPTA), 2015 International Conference on*, 2015, pp. 285–290.

[10] A. M. Bukar and H. Ugail, “Facial Age Synthesis using Sparse Partial Least Squares (The Case of Ben Needham),” *J. Forensic Sci.*, 2017.

[11] A. M. Bukar and H. Ugail, “A Nonlinear Appearance Model for Age Progression,” in *Soft Computing and Machine Learning in Image Processing (In Press)*, Springer Berlin Heidelberg, 2017.

[12] A. M. Bukar, H. Ugail, and N. Hussain, “On facial age progression based on modified active appearance models with face texture,” in *Advances in Computational Intelligence Systems*, Springer, 2017, pp. 465–479.

[13] A. M. Bukar, H. Ugail, and D. Connah, “Automatic age and gender classification using supervised appearance model,” *J. Electron. Imaging*, vol. 25, no. 6, pp. 1–11, 2016.

[14] A. M. Bukar and H. Ugail, “ConvNet Features for Age Estimation,” in *11th International Conference on Computer Graphics, Visualization, Computer Vision and Image Processing*, 2017.

[15] A. Lanitis, C. Taylor, and T. Cootes, “Toward Automatic Simulation of Aging Effects on Face Images,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 24, no. 4, pp. 442–455, 2002.

[16] X. Gao, S. Member, Y. Su, X. Li, and D. Tao, “A Review of Active Appearance Models,” *IEEE Trans. Syst. Man, Cybern. Part C Appl. Rev.*, vol. 40, no. 2, pp. 145–158, 2010.

[17] A. Dessein, A. Makejevs, and W. A. P. Smith, “Towards Synthesis of Novel, Photorealistic 3D Faces,” in *Proceedings of the Facial Analysis*

and Animation, 2015, p. 1.

[18] K. Needham, *Ben*. London: Ebury Publishing, 2013.

[19] K. Harrington, “Twelve Facts about Mary Boyle – the Little Irish Girl Who Vanished and the Allegations of 40 Year Cover up,” *The Irish Post (London England)*, 18-Jul-2016.

[20] E. Patterson, A. Sethuram, M. Albert, and K. Ricanek, “Comparison of synthetic face aging to age progression by forensic sketch artist,” in *International Conference on Visualization, Imaging, and Image Processing*, 2007, pp. 247–252.

[21] L. A. Zebrowitz, P. M. Bronstad, and J. M. Montepare, “An ecological theory of face perception,” *Sci. Soc. Vis.*, pp. 1–30, 2011.

[22] J. B. Pittenger and R. E. Shaw, “Aging faces as viscal-elastic events: implications for a theory of nonrigid shape perception.,” *J. Exp. Psychol. Hum. Percept. Perform.*, vol. 1, no. 4, p. 374, 1975.

[23] J. T. Todd, L. S. Mark, R. E. Shaw, and J. B. Pittenger, “The perception of human growth.,” *Sci. Am.*, vol. 242, no. 2, p. 132, 1980.

[24] L. S. Mark and J. T. Todd, “The perception of growth in three dimensions,” *Attention, Perception, Psychophys.*, vol. 33, no. 2, pp. 193–196, 1983.

[25] D. W. Thompson, “On growth and form.” 1942.

[26] W. Arthur, “D’Arcy Thompson and the theory of transformations,” *Nat. Rev. Genet.*, vol. 7, no. 5, pp. 401–406, 2006.

[27] P. J. Benson and D. I. Perrett, “Synthesising continuous-tone caricatures,” *Image Vis. Comput.*, vol. 9, no. 2, pp. 123–129, 1991.

[28] R.-L. Hsu and A. K. Jain, “Generating discriminating cartoon faces using interacting snakes,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 25, no.

11, pp. 1388–1398, 2003.

[29] N. Ramanathan and R. Chellappa, “Modeling Age Progression in Young Faces,” *2006 IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit. - Vol. 1*, vol. 1, pp. 387–394, 2006.

[30] C.-T. Shen, F. Huang, W.-H. Lu, S.-W. Shih, and H.-Y. M. Liao, “3D Age Progression Prediction in Children’s Faces with a Small Exemplar-Image Set.,” *J. Inf. Sci. Eng.*, vol. 30, no. 4, pp. 1131–1148, 2014.

[31] N. Ramanathan, R. Chellappa, and S. Biswas, “Age progression in Human Faces : A Survey,” *J. Vis. Lang. Comput.*, vol. 15, no. 1, pp. 3349–3361, 2009.

[32] Y. Shan, Z. Liu, and Z. Zhang, “Image-Based Surface Detail Transfer,” *IEEE Comput. Graph. Appl.*, vol. 24, no. 3, pp. 30–35, 2004.

[33] S. Mukaida and H. Ando, “Extraction and manipulation of wrinkles and spots for facial image synthesis,” *Sixth IEEE Int. Conf. Autom. Face Gesture Recognition, 2004. Proceedings.*, pp. 749–754, 2004.

[34] Y. I. N. Wu and N. M. Thalmann, “A plastic-visco-elastic model for wrinkles in facial animation and skin aging,” in *Second Pacific Conference on Computer Graphics and Applications, Pacific Graphics*, 1998, pp. 201–214.

[35] A. Mehdi, R. Qahwaji, H. Ugail, and M. Abdullah, “Construction of 3D Facial Wrinkles using Splines,” in *Fourth International Conference on Information Technology*, 2009.

[36] S. Al-Qatawneh, A. Mehdi, and T. Al Rawashdeh, “3D Modelling, Simulation and Prediction of Facial Wrinkles,” *J. Commun. Comput.*, vol. 11, pp. 365–370, 2014.

[37] D. M. Burt and D. I. Perrett, “Perception of age in adult Caucasian male

faces: computer graphic manipulation of shape and colour information.," *Proc. Biol. Sci.*, vol. 259, no. 1355, pp. 137–43, Feb. 1995.

[38] B. Tiddeman, M. Burt, and D. Perrett, "Prototyping and transforming facial textures for perception research," *IEEE Comput. Graph. Appl.*, vol. 21, no. 5, pp. 42–50, 2001.

[39] Y. Fu and N. Zheng, "M-Face: An appearance-based photorealistic model for multiple facial attributes rendering," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 16, no. 7, pp. 830–842, 2006.

[40] I. Kemelmacher-Shlizerman, S. Suwajanakorn, and S. M. Seitz, "Illumination-Aware Age Progression," *2014 IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 3334–3341, Jun. 2014.

[41] W. Wang, Z. Cui, Y. Yan, J. Feng, S. Yan, X. Shu, and N. Sebe, "Recurrent Face Aging," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016.

[42] C. M. Scandrett, C. J. Solomon, and S. J. Gibson, "A person-specific, rigorous aging model of the human face," *Pattern Recognit. Lett.*, vol. 27, no. 15, pp. 1776–1787, 2006.

[43] X. Geng, Z.-H. Zhou, and K. Smith-Miles, "Automatic age estimation based on facial aging patterns," *Pattern Anal. Mach. Intell. IEEE Trans.*, vol. 29, no. 12, pp. 2234–2240, 2007.

[44] J. Suo, X. Chen, S. Member, and S. Shan, "A Concatenational Graph Evolution Aging Model," vol. 34, no. 11, 2012.

[45] E. Shechtman, A. Rav-Acha, M. Irani, and S. Seitz, "Regenerative morphing," in *Computer Vision and Pattern Recognition (CVPR), 2010 IEEE Conference on*, 2010, pp. 615–622.

[46] T. F. Cootes, G. J. Edwards, and C. J. Taylor, "Active Appearance

Models," vol. 23, no. 6, pp. 681–685, 2001.

[47] A. Lanitis, C. Draganova, and C. Christodoulou, "Comparing Different Classifiers for Automatic Age Estimation," *IEEE Trans. Syst. Man, Cybern. Part B Cybern.*, vol. 34, no. 1, pp. 621–628, 2004.

[48] Y. Wang, Z. Zhang, W. Li, and F. Jiang, "Combining Tensor Space Analysis and Active Appearance Models for Aging Effect Simulation on Face Images.," *IEEE Trans. Syst. Man. Cybern. B. Cybern.*, vol. 42, no. 4, pp. 1107–1118, Mar. 2012.

[49] U. Park, Y. Tong, and A. K. Jain, "Face recognition with temporal invariance: A 3D aging model," *2008 8th IEEE Int. Conf. Autom. Face Gesture Recognition, FG 2008*, pp. 0–6, 2008.

[50] U. Park, Y. Tong, and A. K. Jain, "Age-invariant face recognition.," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 32, no. 5, pp. 947–54, May 2010.

[51] D. W. Hunter and B. P. Tiddeman, "Visual ageing of human faces in three dimensions using morphable models and projection to latent structures," *VISAPP 2009*, 2009.

[52] V. Blanz and T. Vetter, "A morphable model for the synthesis of 3D faces," *Proc. 26th Annu. Conf. Comput. Graph. Interact. Tech. - SIGGRAPH '99*, pp. 187–194, 1999.

[53] A. Maronidis and A. Lanitis, "Facial Age Simulation using Age-specific 3D Models and Recursive PCA.," in *VISAPP (1)*, 2013, pp. 663–668.

[54] R. Tomassi, "Vending machine having a biometric verification system for authorizing the sales of regulated products." Google Patents, 23-Mar-2004.

[55] P. Angelov and P. Sadeghi-Tehran, "Look-a-Like: A Fast Content-Based Image Retrieval Approach Using a Hierarchically Nested Dynamically

Evolving Image Clouds and Recursive Local Data Density," *Int. J. Intell. Syst.*, vol. 32, no. 1, pp. 82–103, 2017.

[56] C. Xu, Q. Liu, and M. Ye, "Age invariant face recognition and retrieval by coupled auto-encoder networks," *Neurocomputing*, vol. 222, pp. 62–71, 2017.

[57] Y. Sun, M. Zhang, Z. Sun, and T. Tan, "Demographic Analysis from Biometric Data: Achievements, Challenges, and New Frontiers," *IEEE Trans. Pattern Anal. Mach. Intell.*, 2017.

[58] J. K. Pontes, A. S. Britto, C. Fookes, and A. L. Koerich, "A flexible hierarchical approach for facial age estimation based on multiple features," *Pattern Recognit.*, vol. 54, pp. 34–51, 2016.

[59] S. E. Choi, Y. J. Lee, S. J. Lee, K. R. Park, and J. Kim, "Age estimation using a hierarchical classifier based on global and local facial features," *Pattern Recognit.*, vol. 44, no. 6, pp. 1262–1281, Jun. 2011.

[60] Y. H. Kwon and V. Lobo, "Age classification from facial images," *Proc. IEEE Conf. Comput. Vis. Pattern Recognit. CVPR-94*, pp. 762–767, 1994.

[61] W. B. Horng, C. P. Lee, and C. W. Chen, "Classification of age groups based on facial features," *Tamkang J. Sci. Eng.*, vol. 4, no. 3, pp. 183–192, 2001.

[62] S. E. Choi, Y. J. Lee, S. J. Lee, K. R. Park, and J. Kim, "Age estimation using a hierarchical classifier based on global and local facial features," *Pattern Recognit.*, vol. 44, no. 6, pp. 1262–1281, Jun. 2011.

[63] C. R. Babu, E. S. Reddy, and B. P. Rao, "Age group classification of facial images using Rank based Edge Texture Unit (RETU)," *Procedia Comput. Sci.*, vol. 45, no. C, pp. 215–225, 2015.

[64] Y. Fu, Y. Xu, and T. S. Huang, "Estimating human age by manifold

analysis of face pictures and regression on aging features," in *2007 IEEE International Conference on Multimedia and Expo*, 2007, pp. 1383–1386.

[65] G. Guo, Y. Fu, C. R. Dyer, and T. S. Huang, "Image-Based Human Age Estimation by Manifold Learning and Locally Adjusted Robust Regression," vol. 17, no. 7, pp. 1178–1188, 2008.

[66] G. Guo, G. Mu, Y. Fu, C. Dyer, and T. Huang, "A study on automatic age estimation using a large database," *Comput. Vision, 2009 IEEE 12th Int. Conf.*, vol. 12, no. C, pp. 1986–1991, 2009.

[67] M. Riesenhuber and T. Poggio, "Hierarchical models of object recognition in cortex.," *Nat. Neurosci.*, vol. 2, no. 11, pp. 1019–25, 1999.

[68] G. Guo, G. Mu, Y. Fu, and T. S. Huang, "Human age estimation using bio-inspired features," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, 2009, pp. 112–119.

[69] M. Y. El Dib and M. El-Saban, "Human age estimation using enhanced bio-inspired features (EBIF)," *2010 IEEE Int. Conf. Image Process.*, pp. 1589–1592, Sep. 2010.

[70] G. Panis, A. Lanitis, N. Tsapatsoulis, and T. F. Cootes, "Overview of research on facial ageing using the FG-NET ageing database," *IET Biometrics*, 2015.

[71] C. Fernandez, I. Huerta, and A. Prati, "A Comparative Evaluation of Regression Learning Algorithms for Facial Age Estimation," *FFER conjunction with ICPR, Press. IEEE*, 2014.

[72] X. Wang, R. Guo, and C. Kambhamettu, "Deeply-learned feature for age estimation," in *2015 IEEE Winter Conference on Applications of Computer Vision*, 2015, pp. 534–541.

[73] K. Ricanek and T. Tesafaye, "Morph: A longitudinal image database of

normal adult age-progression," in *7th International Conference on Automatic Face and Gesture Recognition (FGR06)*, 2006, pp. 341–345.

[74] G. Levi and T. Hassner, "Age and Gender Classification using Convolutional Neural Networks," pp. 34–42, 2015.

[75] Z. Niu, M. Zhou, L. Wang, X. Gao, and G. Hua, "Ordinal regression with multiple output cnn for age estimation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2016, pp. 4920–4928.

[76] D. Yi, Z. Lei, and S. Z. Li, "Age estimation by multi-scale convolutional network," in *Asian Conference on Computer Vision*, 2014, pp. 144–158.

[77] T. Liu, J. Wan, T. Yu, Z. Lei, and S. Z. Li, "Age Estimation Based on Multi-Region Convolutional Neural Network," in *Chinese Conference on Biometric Recognition*, 2016, pp. 186–194.

[78] X. Liu, S. Li, M. Kan, J. Zhang, S. Wu, W. Liu, H. Han, S. Shan, and X. Chen, "Agenet: Deeply learned regressor and classifier for robust apparent age estimation," in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2015, pp. 16–24.

[79] Y. LeCun, Y. Bengio, and G. Hinton, "Deep learning," *Nature*, vol. 521, no. 7553, pp. 436–444, 2015.

[80] C. Fernández, I. Huerta, and A. Prati, "A comparative evaluation of regression learning algorithms for facial age estimation," in *Face and Facial Expression Recognition from Real World Videos*, Springer, 2015, pp. 133–144.

[81] T. Wu, P. Turaga, and R. Chellappa, "Age Estimation and Face Verification Across Aging," vol. 7, no. 6, pp. 1780–1788, 2012.

[82] S. Yan, X. Zhou, M. Liu, M. Hasegawa-Johnson, and T. S. Huang,

"Regression from patch-kernel," in *Computer Vision and Pattern Recognition, 2008. CVPR 2008. IEEE Conference on*, 2008, pp. 1–8.

[83] T. F. Vieira, A. Bottino, A. Laurentini, and M. De Simone, "Detecting siblings in image pairs," *Vis. Comput.*, vol. 30, no. 12, pp. 1333–1345, 2014.

[84] K. A. Dalrymple, J. Gomez, and B. Duchaine, "The dartmouth database of children's faces: Acquisition and validation of a new face stimulus set," *PLoS One*, vol. 8, no. 11, pp. 1–7, 2013.

[85] FG-NET, "The Fg-Net Aging Database," 2014.

[86] T. F. Cootes, G. J. Edwards, and C. J. Taylor, "Active Appearance Models," in *In Computer Vision—ECCV'98*, 1998, pp. 484–498.

[87] S. J. Gibson, C. J. Solomon, and a. P. Bejarano, "Synthesis of Photographic Quality Facial Composites using Evolutionary Algorithms," *Procedings Br. Mach. Vis. Conf. 2003*, p. 23.1-23.10, 2003.

[88] G. J. Edwards, T. F. Cootes, and C. J. Taylor, "Face Recognition Using Active Appearance Models," in *Computer Vision—ECCV'98*, 1998, pp. 581–595.

[89] J. C. Gower, "Generalized procrustes analysis," *Psychometrika*, vol. 40, no. 1, pp. 33–51, 1975.

[90] M. B. Stegmann, D. D. Gomez, R. P. Plads, and D.-K. Lyngby, "A Brief Introduction to Statistical Shape Analysis," no. March, pp. 1–15, 2002.

[91] M. A. Turk and A. P. Pentland, "Eigenfaces for recognition," *J. Cogn. Neurosci.*, vol. 3, no. 1, pp. 71–86, 1991.

[92] C. a. Glasbey and K. V. Mardia, "A review of image-warping methods," *J. Appl. Stat.*, vol. 25, no. 2, pp. 155–171, Apr. 1998.

[93] C. a. Glasbey and K. V. Mardia, "A review of image-warping methods," *J.*

Appl. Stat., vol. 25, no. 2, pp. 155–171, Apr. 1998.

[94] M. C. Ionita, P. Corcoran, and V. Buzuloiu, “On Colour Texture Normalization for Active Appearance Models,” *IEEE Trans. Image Process.*, vol. 18, no. 6, pp. 1372–1378, 2009.

[95] Y. I. Ohta, T. Kanade, and T. Sakai, “Colour Information for Region Segmentation,” *Comput. Graph. Image Process.*, vol. 13, no. 1, pp. 222–241, 1980.

[96] T. Chen, W. Yin, X. Sean, Z. Dorin, and C. Thomas, “Total Variation Models for Variable Lighting Face Recognition,” *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 9, pp. 1519–1524, 2006.

[97] D. Yadav, R. Singh, M. Vatsa, and A. Noore, “Recognizing age-separated face images: Humans and machines,” *PLoS One*, vol. 9, no. 12, p. e112234, 2014.

[98] E. Moyse, “Age estimation from faces and voices: a review,” *Psychol. Belg.*, vol. 54, no. 3, 2014.

[99] K. P. Burnham and D. R. Anderson, *Model selection and multimodel inference: a practical information-theoretic approach*. Springer Science & Business Media, 2003.

[100] C. A. Mertler and R. V. Reinhart, *Advanced and multivariate statistical methods: Practical application and interpretation*. Routledge, 2016.

[101] R. M. Forte, *Mastering predictive analytics with R*. Packt Publishing Ltd, 2015.

[102] F. Soleymani, M. Sharifi, and S. Shateyi, “Approximating the inverse of a square matrix with application in computation of the Moore-Penrose inverse,” *J. Appl. Math.*, vol. 2014, 2014.

[103] R. R. Wilcox, *Introduction to robust estimation and hypothesis testing*.

Academic Press, 2011.

[104] R. D. Tobias, “An introduction to partial least squares regression,” *SAS Conf. Proc. SAS Users Gr. Int. 20 (SUGI 20)*, pp. 2–5, 1995.

[105] H. Wold, *Quantitative sociology: international perspectives on mathematical and statistical model building, chapter path models with latent variables: the NiPALS Approach.* Academic, London, 1975.

[106] H. Wold, “Partial least squares,” *Encyclopedia of the Statistical Sciences*. John Wiley & Sons, pp. 581–591, 1985.

[107] O. Yeniay and A. Goktas, “A comparison of partial least squares regression with other prediction methods,” *Hacettepe J. Math. Stat.*, vol. 31, no. 99, pp. 99–101, 2002.

[108] Ll. E. Frank and J. H. Friedman, “A statistical view of some chemometrics regression tools,” *Technometrics*, vol. 35, no. 2, pp. 109–135, 1993.

[109] S. De Jong, “SIMPLS: An alternative approach to partial least squares regression,” *Chemom. Intell. Lab. Syst.*, vol. 18, no. 3, pp. 251–263, 1993.

[110] H. Chun, *Sparse Partial Least Squares Regression for Simultaneous Dimension Reduction and Variable Selection with Applications to High Dimensional Genomic Data*. Michigan: ProQuest, 2008.

[111] H. Chun and S. Keleş, “Sparse partial least squares regression for simultaneous dimension reduction and variable selection,” *J. R. Stat. Soc. Ser. B (Statistical Methodol.*, vol. 72, no. 1, pp. 3–25, 2010.

[112] R. Tibshirani, “Regression shrinkage and selection via the lasso,” *J. R. Stat. Soc. Ser. B*, pp. 267–288, 1996.

[113] D. Chung, H. Chun, and S. Keles, “Package ‘spls’ , Version 2.2-1.” June, 2015.

[114] A. Lanitis, “Evaluating the Performance of Face-Aging Algorithms,” in

IEEE International Conference on Automatic Face & Gesture Recognition, 2008, 2008, pp. 1–6.

[115] Y.-L. Chen and C.-T. Hsu, “Subspace learning for facial age estimation via pairwise age ranking,” *IEEE Trans. Inf. Forensics Secur.*, vol. 8, no. 12, pp. 2164–2176, 2013.

[116] X. Gao, Y. Su, X. Li, and D. Tao, “A Review of Active Appearance Models,” *IEEE Trans. Syst. Man, Cybern. Part C Appl. Rev.*, vol. 40, no. 2, pp. 145–158, 2010.

[117] S. Mika, B. Schölkopf, A. Smola, K. Müller, M. Scholz, and G. Rätsch, “Kernel PCA and De-Noising in Feature Spaces,” in *Advances in Neural Information Processing Systems 11*, 1999, pp. 536–542.

[118] P. Honeine and C. Richard, “Preimage problem in kernel-based machine learning,” *IEEE Signal Process. Mag.*, vol. 28, no. 2, pp. 77–88, 2011.

[119] M. A. Aizerman, E. M. Braverman, and L. I. Rozonoer, “Theoretical foundations of the potential function method in pattern recognition,” *Autom. Remote Control*, vol. 25, pp. 917–936, 1964.

[120] B. Scholkopf, A. Smola, and K. R. Muller, “Nonlinear Component Analysis as a Kernel Eigenvalue Problem,” *Neural Comput.*, vol. 10, no. 5, pp. 1299–1319, 1996.

[121] B. Schölkopf, A. Smola, and K.-R. Müller, “Nonlinear component analysis as a kernel eigenvalue problem,” *Neural Comput.*, vol. 10, no. 5, pp. 1299–1319, 1998.

[122] J.-B. Li, S.-C. Chu, and J.-S. Pan, *Kernel Learning Algorithms for Face Recognition*. New York, NY: Springer New York, 2014.

[123] P. Honeine and C. Richard, “A Closed-form Solution for the Pre-image Problem in Kernel-based Machines,” *J. Signal Process. Syst.*, vol. 65, no.

3, pp. 289–299, 2011.

[124] W. Zheng, J. Lai, X. Xie, Y. Liang, P. C. Yuen, and Y. Zou, “Kernel Methods for Facial Image Preprocessing,” in *Pattern Recognition, Machine Intelligence and Biometrics*, Springer, 2011, pp. 389–409.

[125] J. T. Kwok and I. W. Tsang, “The Pre-Image Problem in Kernel Methods,” *IEEE Trans. Neural Networks*, vol. 15, no. 6, pp. 1517–1525, 2004.

[126] S. I. Kabanikhin, “Definitions and examples of inverse and ill-posed problems,” *J. Inverse Ill-Posed Probl.*, vol. 16, no. 4, pp. 317–357, 2008.

[127] T. J. Abrahamsen and L. K. Hansen, “Input Space Regularization Stabilizes Pre-Images for Kernel Pca De-Noising,” *IEEE Int. Work. Mach. Learn. Signal Process. 2009. MLSP 2009*, 2009.

[128] C. Leitner and F. Pernkopf, *The pre-image problem and kernel PCA for speech enhancement*. Springer Berlin Heidelberg, 2011.

[129] H. Lin and C. Lin, “A study on sigmoid kernels for SVM and the training of non-PSD kernels by SMO-type methods,” 2003.

[130] S. Boughorbel, J. Tarel, and N. Boujemaa, “Conditionally positive definite kernels for svm based image recognition.,” in *IEEE International Conference on Multimidea and Expo*, 2005, pp. 113–116.

[131] U. Von Luxburg, O. Bousquet, and B. Schölkopf, “A compression approach to support vector model selection,” *J. Mach. Learn. Res.*, vol. 5, no. Apr, pp. 293–323, 2004.

[132] P. t LIP, “Model selection for support vector machines,” 1999.

[133] Q. Wang, “Kernel principal component analysis and its applications in face recognition and active shape models,” *arXiv Prepr. arXiv1207.3538*, 2012.

[134] U. Mohammed, S. J. Prince, and J. Kautz, “Visio-lization : Generating

Novel Facial Images," *ACM Trans. Graph.*, vol. 28, no. 3, p. 57, 2009.

[135] A. Dessein, D. Bordeaux, A. Makejevs, and W. A. P. Smith, "Towards Synthesis of Novel , Photorealistic 3D Faces," no. c, p. 2813853, 2016.

[136] P. Aschwanden and W. Guggenbuhl, "Experimental results from a comparative study on correlation-type registration algorithms," *Robust Comput. Vis.*, pp. 268–289, 1992.

[137] A. Nakhmani and A. Tannenbaum, "A new distance measure based on generalized image normalized cross-correlation for robust video tracking and image recognition," *Pattern Recognit. Lett.*, vol. 34, no. 3, pp. 315–321, 2013.

[138] L. Wang, Y. Zhang, and J. Feng, "On the Euclidean distance of images," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 27, no. 8, pp. 1334–1339, 2005.

[139] J. Li and B.-L. Lu, "An adaptive image Euclidean distance," *Pattern Recognit.*, vol. 42, no. 3, pp. 349–357, 2009.

[140] P. Pérez, M. Gangnet, and A. Blake, "Poisson image editing," *ACM Trans. Graph.*, vol. 22, no. 3, p. 313, 2003.

[141] H. Abdi, "Partial least squares regression and projection on latent structure regression (PLS Regression)," *Wiley Interdiscip. Rev. Comput. Stat.*, vol. 2, no. 1, pp. 97–106, 2010.

[142] G. Guo and G. Mu, "Simultaneous dimensionality reduction and human age estimation via kernel partial least squares regression," *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 657–664, 2011.

[143] S. Yan, H. Wang, X. Tang, and T. S. Huang, "Learning Auto-Structured Regressor from Uncertain Nonnegative Labels," *2007 IEEE 11th Int. Conf. Comput. Vis.*, pp. 1–8, 2007.

[144] X. Geng, C. Yin, and Z.-H. Zhou, “Facial age estimation by learning from label distributions,” *Pattern Anal. Mach. Intell. IEEE Trans.*, vol. 35, no. 10, pp. 2401–2412, 2013.

[145] H. Han, C. Otto, and A. K. Jain, “Age estimation from face images: Human vs. machine performance,” in *2013 International Conference on Biometrics (ICB)*, 2013, pp. 1–8.

[146] K. Y. Chang, C. S. Chen, and Y. P. Hung, “Ordinal hyperplanes ranker with cost sensitivities for age estimation,” *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recognit.*, pp. 585–592, 2011.

[147] W. Chao, J. Liu, and J. Ding, “Facial age estimation based on label-sensitive learning and age-oriented regression,” *Pattern Recognit.*, vol. 46, no. 3, pp. 628–641, 2013.

[148] L. Hong, D. Wen, C. Fang, and X. Ding, “A new biologically inspired active appearance model for face age estimation by using local ordinal ranking,” in *Proceedings of the Fifth International Conference on Internet Multimedia Computing and Service*, 2013, pp. 327–330.

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

[150] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, and M. Bernstein, “Imagenet large scale visual recognition challenge,” *Int. J. Comput. Vis.*, vol. 115, no. 3, pp. 211–252, 2015.

[151] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” *arXiv Prepr. arXiv1512.03385*, 2015.

[152] G. Huang, Z. Liu, and K. Q. Weinberger, “Densely connected

convolutional networks," *arXiv Prepr. arXiv1608.06993*, 2016.

[153] Y. Zhong, R. Arandjelović, and A. Zisserman, "Faces in places: Compound query retrieval," in *BMVC-27th British Machine Vision Conference*, 2016.

[154] T. S. Huang, "Human age estimation using bio-inspired features," *2009 IEEE Conf. Comput. Vis. Pattern Recognit.*, pp. 112–119, Jun. 2009.

[155] A. Sharif Razavian, H. Azizpour, J. Sullivan, and S. Carlsson, "CNN features off-the-shelf: an astounding baseline for recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2014, pp. 806–813.

[156] S. Zha, F. Luisier, W. Andrews, N. Srivastava, and R. Salakhutdinov, "Exploiting image-trained CNN architectures for unconstrained video classification," *arXiv Prepr. arXiv1503.04144*, 2015.

[157] H. Azizpour, A. Sharif Razavian, J. Sullivan, A. Maki, and S. Carlsson, "From generic to specific deep representations for visual recognition," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2015, pp. 36–45.

[158] B. Athiwaratkun and K. Kang, "Feature Representation in Convolutional Neural Networks," *arXiv Prepr. arXiv1507.02313*, 2015.

[159] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning representations by back-propagating errors," *Cogn. Model.*, vol. 5, no. 3, p. 1, 1988.

[160] N. Srivastava, G. E. Hinton, A. Krizhevsky, I. Sutskever, and R. Salakhutdinov, "Dropout: a simple way to prevent neural networks from overfitting.," *J. Mach. Learn. Res.*, vol. 15, no. 1, pp. 1929–1958, 2014.

[161] Y. LeCun and Y. Bengio, "Convolutional networks for images, speech,

and time series," *Handb. brain theory neural networks*, vol. 3361, no. 10, p. 1995, 1995.

[162] A. Vedaldi and K. Lenc, "Matconvnet: Convolutional neural networks for matlab," in *Proceedings of the 23rd ACM international conference on Multimedia*, 2015, pp. 689–692.

[163] Y. LeCun, L. D. Jackel, L. Bottou, A. Brunot, C. Cortes, J. S. Denker, H. Drucker, I. Guyon, U. A. Muller, and E. Sackinger, "Comparison of learning algorithms for handwritten digit recognition," in *International conference on artificial neural networks*, 1995, vol. 60, pp. 53–60.

[164] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, 1998.

[165] N. Rusk, "Deep learning," *Nat. Methods*, vol. 13, no. 1, pp. 35–35, 2015.

[166] V. P. Plagianakos, G. D. Magoulas, and M. N. Vrahatis, "Learning rate adaptation in stochastic gradient descent," in *Advances in convex analysis and global optimization*, Springer, 2001, pp. 433–444.

[167] L. Bottou, "Stochastic gradient learning in neural networks," *Proc. Neuro-Nımes*, vol. 91, no. 8, 1991.

[168] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, 2009, pp. 248–255.

[169] Y. Bengio, "Practical recommendations for gradient-based training of deep architectures," in *Neural networks: Tricks of the trade*, Springer, 2012, pp. 437–478.

[170] X. Glorot and Y. Bengio, "Understanding the difficulty of training deep feedforward neural networks.," in *Aistats*, 2010, vol. 9, pp. 249–256.

[171] V. Nair and G. E. Hinton, “Rectified linear units improve restricted boltzmann machines,” in *Proceedings of the 27th international conference on machine learning (ICML-10)*, 2010, pp. 807–814.

[172] B. Graham, “Fractional max-pooling,” *arXiv Prepr. arXiv1412.6071*, 2014.

[173] K. Simonyan and A. Zisserman, “Very deep convolutional networks for large-scale image recognition,” *arXiv Prepr. arXiv1409.1556*, 2014.

[174] P. Sermanet, D. Eigen, X. Zhang, M. Mathieu, R. Fergus, and Y. LeCun, “Overfeat: Integrated recognition, localization and detection using convolutional networks,” *arXiv Prepr. arXiv1312.6229*, 2013.

[175] M. Oquab, L. Bottou, I. Laptev, and J. Sivic, “Learning and transferring mid-level image representations using convolutional neural networks,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2014, pp. 1717–1724.

[176] J. Yosinski, J. Clune, Y. Bengio, and H. Lipson, “How transferable are features in deep neural networks?,” in *Advances in neural information processing systems*, 2014, pp. 3320–3328.

[177] F. Schroff, D. Kalenichenko, and J. Philbin, “Facenet: A unified embedding for face recognition and clustering,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2015, pp. 815–823.

[178] Y. Sun, X. Wang, and X. Tang, “Deep learning face representation from predicting 10,000 classes,” in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1891–1898.

[179] L. Wolf, T. Hassner, and I. Maoz, “Face recognition in unconstrained videos with matched background similarity,” in *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, 2011, pp. 529–

534.

[180] H. Han, C. Otto, A. K. Jain, and E. Lansing, “Age Estimation from Face Images : Human vs . Machine Performance,” 2013.

[181] X. Zhu and D. Ramanan, “Face detection, pose estimation, and landmark localization in the wild,” in *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*, 2012, pp. 2879–2886.

[182] T.-J. Liu, K.-H. Liu, H.-H. Liu, and S.-C. Pei, “Age estimation via fusion of multiple binary age grouping systems,” in *Image Processing (ICIP), 2016 IEEE International Conference on*, 2016, pp. 609–613.

[183] G. Guo and G. Mu, “Human age estimation: What is the influence across race and gender?,” in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition-Workshops*, 2010, pp. 71–78.

[184] A. M. Bukar and H. Ugail, “On Automatic Age Estimation from Facial Profile View,” *Accept. Manuscr. IET Comput. Vis.*, 2017.