An Efficient Age Estimation System with Facial Makeover Images Based on Key Points Selection

Tamilb Vairavan[1] and Kaliyaperumal Vani[2]
[1]Department of Electronics and Instrumentation, R.M.K Engineering College, Chennai
[2]Department of Information Science and Technology College of Engineering, Anna University, Chennai

Abstract: *Age is one of the essential factors in establishing the identity of the person. Estimation of the human age is a procedure adopted by anthropologists, archaeologists and forensic scientists. Compared with other cognition problems, age estimation from face images is still very challenging. Predicting and estimating the age from facial images with makeup is an interesting task in digital entertainment. Estimating age from a facial image is an intriguing and exigent task. Aging changes both shape as well as texture and it is an irreversible, uncontrollable and personalized. The efficiency of the age estimation system degrades with respect to facial makeover. The main objective of this research is to estimate the age of a human from the facial image with makeup. Initially, the face image will be normalized by employing a face detection algorithm. After detecting the face exactly, we have extracted the unique features (key points) from the images such as texture, shape and regions. Estimating the age of a person with different makeovers is not an easy task. To overcome this difficulty, we have to identify the uniqueness of each image of a same person. The eye part does not change whatever the person having the makeup. So the eyes are same for the person with different makeover. For region area or key points, the eye portion will be segmented from the detected face image. The shape feature can be extracted by Active Appearance Model (AAM). Finally, based on the feature library, the image can be classified under a particular age group using Artificial Neural Network (ANN). After the classification the age can be predicted. The proposed approach will be implemented in MATLAB and planned to be evaluated using various facial makeover images.*

Keywords: *Age estimation system, AAM, ANN, LGXP.*

Received July 1, 2013; accepted March 20, 2014

1. Introduction

Human faces, as important visual cues, convey a significant amount of non-verbal information to facilitate the real-world human-to-human communication. As a result, the modern intelligent systems are expected to have the capability to accurately recognize and interpret human faces in real time [8]. As one of the main human facial attributes, aging plays a more complex role than other factors such as human identity, expression, gender and race. Besides, because age is a temporal property of people, it's hard to collect the same person's face image across ages. It's also tedious and laborious to label the exact or approximate ages of collected faces. Due to these difficulties, researches on human age are not as much as that on other facial attributes [4]. Compared with other facial variations, aging effects are very dependent on genetics, life style, and location of residence and weather conditions. Furthermore, males and females age differently, and the apparent effects of aging are often masked by makeup and facial accessories [15].

Factors such as angle, illumination, facial expressions, and makeup can reduce the robustness of the model. Further, variations between individual facial aging patterns make it difficult to use one global function for all people. Therefore, we try to build a model for each person, which falls into the category of personalized age estimation. It is generally agreed that personalized age estimation performs better than global age estimation [5]. Even a human observer can rarely guess the exact age of a face image [6]. Most automatic image-based age estimation systems are composed by combining two components: An image representation and an age estimation process [15]. Developing a facial appearance model entails characterizing certain attributes that are inherent to human faces such as: The 3D structure of human faces, the reflective properties of facial skin, the uniqueness of different facial features, and the bilateral symmetry in the configuration of facial features etc., in combination with scene-centric attributes such as: Illumination and view-point [12].

Although, automatic image-based age estimation is an important technique involved in many real-world applications, it is still a challenging problem to estimate human ages from face images. Since, different individuals age quite differently, the aging process is determined by not only the person's gene but also many external factors, such as health, living style, living location, and weather conditions. Males and females may also age differently due to the different extent in using make-up's and accessories [8]. Besides, facial aging is a sub-problem in face recognition, because simulating the appearance of a person across years may help recognizing his or her face [1].

Due to its potential in various kinds of applications, facial age estimation has been studied in fields as diverse as image processing, pattern recognition, and machine learning [5]. Facial attributes, such as: Identity, age, gender, expression, and ethnic origin, play a crucial role in real facial image analysis applications. Estimating human age automatically via facial image analysis has lots of potential real-world applications, such as multimedia communication, Human Computer Interaction (HCI), and security [8]. Age estimation can be used in a wide range of smart human-machine applications, e.g., limiting access to age-appropriate internet or television contents or creating general characteristics of a typical customer in a required age range to be used to develop a marketing strategy [1]. Facial age estimation uses computers to estimate age by facial images. It can be widely applied in such areas as age-based access control, age-adaptive human computer interaction, and age-based advertising [5].

2. Literature Review

Choi *et al*. [3] have proposed a new age estimation method using a hierarchical classifier method based on both global and local facial features. The experimental results have shown that the performance of the proposed method was superior to that of the previous methods when using the BERC, PAL and FG-Net aging databases.

The problem of eye detection in face images was very important for a large number of applications ranging from face recognition to gaze tracking. Ramezanpour *et al.* [13] have proposed a new algorithm for eyes detection. Experimental results have shown that a correct eye detection rate of 94% could be achieved on 300 FACE94 images with different facial expressions and lighting conditions. Due to using of GLHS color space, lighting condition didn't affect on algorithm output.

Hewahi *et al*. [9] have presented a methodology based on neural networks to estimate human ages using face features. Due to the difficulty of estimating the exact age, they have developed their system to estimate the age to be within certain ranges. Two public data sets were used to test the system, these are FG-NET and MORPH. To evaluate our system's performance, they have carried out a comparative study between their proposed system, human being and other research trails.

Roy *et al*. [14] have discussed a novel mechanism by which two images at different time periods of the individual life can be compared so that, it can be ascertained both images are of same individual. This was done by extracting feature points of a face from which a face triangle is formed. If the ratio of areas of the triangles of both images was within a specified range then we could say both images are of same person. Experimental results have shown that face recognition and age range estimation both may be effectively performed and which performs with low computational effort.

Chiang *et al*. [2] have described a real-time face detection algorithm for locating faces in images and videos. The algorithm was found not only the face regions, but also the precise locations of the facial components such as eyes and lips. According to the experimental results, the proposed algorithm was exhibited satisfactory performance in terms of both accuracy and speed for detecting faces with wide variations in size, scale, orientation, color, and expressions.

Li *et al.* [11] have proposed a discriminative model to address face matching in the presence of age variation. In that framework, they have represented each face by designing a densely sampled local feature description scheme, in which Scale Invariant Feature Transform (SIFT) and Multi-scale Local Binary Patterns (MLBP) serve as the local descriptors. Experimental results have shown that their approach outperforms a state-of-the-art commercial face recognition engine on two public domain face aging data sets: MORPH and FG-NET.

Tin and Sein [17] have developed automatic age dependent face recognition system. This approach was based on the Principle Component Analysis (PCA). The efficiency of the system could be confirmed through the experimental results. While recognition of most facial variations, such as identity, expression and gender, has been exten-sively studied, automatic age estimation has rarely been explored. In contrast to other facial variations, aging variation presents several unique characteristics which make age estimation a challenging task.

Geng *et al*. [7] have proposed an automatic age estimation method named Aging pattern Subspace (AGES). The basic idea was to model the aging pattern, which is defined as the sequence of a particular individual's face images sorted in time order, by constructing a representative subspace. In the experiments, AGES and its variants were compared with the limited existing age estimation methods (WAS and AAS) and some well-established classification methods (kNN, BP, C4.5, and SVM). Moreover, a comparison with human perception ability on age was conducted. It is interesting to note that the performance of AGES was not only significantly better than that of all the other algorithms, but also comparable to that of the human observers.

Thakur and Verma [16] have concerned with providing a methodology to estimate age groups using face features. The proposed method was based on the supervised neural network with backpropagation algorithm. Their proposed approach has been developed, tested and trained using the database FG-NET. The algorithm classifies subjects into four different age categories in 4 four key stages: Pre-processing, facial feature extraction wrinkle analysis, and age group classification. The obtained results were significant.

3. Proposed Age Estimation Methodology

3.1. Face Detection Algorithm

Face detection is the initial and important task for the estimation of age with the facial makeover. It is needed to know whether an image contains a face and if so, where it is-this is referred as "face detection". Detecting the exact face is very difficult one, because of some factors such as: The illumination variation, facial appearance, face expression, pose of head, image orientation, gender, wearing of glasses, different skin coloring, and facial makeup. Such factors results in the face distribution to be highly non-linear and complex in any space that is linear to the original image space and the original exact face may not be detected.

3.1.1. Estimation of the Bounding Box

To detect the face, a method to predict the bounding box of a face image using an efficient binary test and a decision tree is used. For this a bounding box estimator is used. The training data, binary test and tree construction and data stored in leaf node are defined in detail.

3.1.1.1. Training Data

For an image of the size $m \times n$ and the pitch size $m_p \times n_p$, the number of overlapping patches is $\frac{(m-m_p+1)(n-n_p+1)}{2}$.

A set of patches are represented as:

$$\left\{ P_i = (A_i, l_i) \right\} \tag{1}$$

Where A_i= the appearance of the patch, and l_i= the offset of the patch.

The offset vector of the patch l_i is a 2D vector that specifies shifts of (x, y) from the center face image or from a fixed point in the face image.

3.1.1.2. Binary Test

General binary test: A test in a decision tree DT has to be done at a node. Initially, consider a simple binary test, which is in the form of:

$$DT(A) = \begin{cases} 1, & \text{if } A(x,y) \leq A(x',y') \\ 0, & \text{otherwise} \end{cases} \tag{2}$$

Where (x, y) and (x', y') = Two locations in the A patch.
Proposed binary test:

$$DT_{\mu}(A) = \begin{cases} 1, & \text{if } A(x,y) \leq Avg(A) \\ 0, & \text{otherwise} \end{cases} \tag{3}$$

Where $Avg(A)$ = Average values of the pixels in the A patch.

In comparison with the general binary test, this proposed binary test only needed half amount of the total number of the pixel. Additionally, this method needed an integral image for the calculation of the average values, quickly.

3.1.1.3. Tree Construction

At the time of training, each non-leaf node in the decision tree chooses the binary test that splits the training samples in a best way. The offset uncertainty used in this is represented as follows:

$$U(L) = \sum_{i \in L} (l_i - l_L)^2 \tag{4}$$

Where l_L= Mean offset vector value of all the face image patches in the set $L = \{P_i = (A_i, l_i)\}$ patch.

For minimizing the equation given in below, we have to do a binary test DT^*:

$$DT^* = \arg \min_{b=1,\ldots,B} \left(U(L_{left}) + U(L_{right}) \right) \tag{5}$$

Where B= Number of binary tests to be possibly occurred; L_{left} *and* L_{right}= Subset of training samples that are reaching the *left* and *right* nodes, respectively.

Every leaf node *left*, in the decision tree that is constructed, stores a single offset vector value (x_{left}, y_{left}). Consider that the subset of training examples L_{left}, that arrives at the left node *left*, then the offset vector value, (x_{left}, y_{left}) will be as follows:

$$x_{left} = \frac{1}{|L_{left}|} \sum_{k \in L_{left}} x_k \tag{6}$$

$$y_{left} = \frac{1}{|L_{left}|} \sum_{k \in L_{left}} y_k \tag{7}$$

There are two stopping criteria for building the tree. They are (a) the maximum depth of the tree (b) minimum number of samples present at a node. Whenever, a node satisfies the second (b) stopping criteria, we add an additional constraint. This constraint is helpful for checking out the variance of the offset vectors with one particular noted threshold value. It is the optimal solution to provide a better estimate value of the offset at the leaf nodes. Thus, we have to detect the face from an image clearly.

3.2. Feature Extraction

After the detection of the face, we can extract the features from that detected face image. Normally, the facial features such as: Eyebrows, eyes, nose, mouth, hair-line, texture, surface, and shape are commonly extracted from the face. If the face is covered with makeup and mask, it is impossible to find out the correct original face parts. In order to find out the original face part, we extract the features of left and right eye portions, texture, and the shape of the face.

The reason for extracting only of these features is that the eye parts, texture, and the shape does not change whatever the person having the makeup. So the eyes, texture and the shape are same for the person with different makeover.

3.2.1. Shape Extraction using AAM

3.2.1.1. Structure of AAM

From the facial image, the feature, shape is extracted using Active Appearance Model (AAM). The method AAM is a statistical model of appearance, which is made by the combination of shape model and texture model. The features that are extracted consist of both the information of shape and texture. The variations in the shape are taken by aligning the landmark points and then PCA is performed on those points. The Eigen values are used for changing the shape by varying the elements of shape model parameters. Figure 1 depicts the shape changes that are made by applying changes in Eigen values and added to the mean shape. For a given means face shape, we can do the texture modeling by warping the images into the mean shape, to obtain a shape free patch. Texture modeling is also similar to the shape modeling. It is obtained by using PCA. Appearance model parameter can be made by combining shape model parameter and grey-level model parameter. PCA is applied on combined parameter vector and then the appearance parameter that controls both shape and texture of the model is calculated. It is possible to achieve changes in both shape and texture, by varying the appearance parameter.

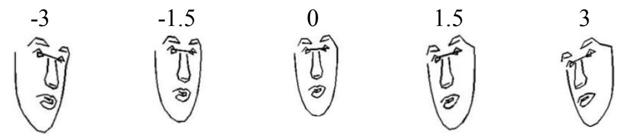
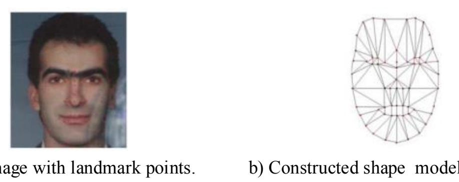


Figure 1. Shape changes obtained by varying mean shape.

3.2.1.2. Constructing the Shape Model

For this construction, a 2-dimensional shape model 'SM' with 'p' points is used. Lot of texture and shape information have presented in the points, which are placed in regions of the face. The shape model having number of instances 's'. Each of these instances is represented as a vector, which contains '$2p$' elements, that is the x, y co-ordinates of each of the 'p' points.

$$s = (x_1, y_1, x_2, \ldots, x_p, y_p)^T \tag{8}$$



a) An image with landmark points. b) Constructed shape model example.

Figure 2. Overview of shape model.

Apply generalized orthogonal procrustes analysis to align all the training shapes in the pre-processing step. With the help of this algorithm, all components that are caused by translation, rotation, scaling are eliminated from the data set. By mirroring the training shapes horizontally, this is also helpful to make additional shapes. This directs to a new training data set d of $D'=2D$ training examples. Calculate the mean shape s_0 as the mean of all the training examples, D' on the basis of training data set.

PCA is useful for finding the main components of all shapes in the training data set. Choose m components consisting of m biggest Eigen values as the shape components s_i, where i takes the values from 1 to m. Hence, the reconstruction of shapes of the training data set and the generation of new shapes are possible. The generation of new shapes is not part of the data set which are of the basic shape s_0 and a linear combination of the components s_i.

$$s = \frac{1}{D'} \sum_{i=1}^{D'} d_i' + \sum_{i=1}^{m} v_i s_i \tag{9}$$

Where the value of shape, $s_0 = \frac{1}{D'} \sum_{i=1}^{D'} d_i'$.

According to the number of used components, m the reconstruction of the shapes has the variation in quality. If we want to make new shapes that are not present in the training data set, we will need to have a diverse set of images for the training. In accordance with the training data set, the main components s_i, indicates global variations of the face like pitch and yaw, that are mostly invariant to facial emotions and also local changes like opening, closing of the eyes or mouth, which are involved in the emotional changes of the subjects. Figure 2-b shows the basic shape s_0.

3.2.1.3. Constructing the Appearance Model

Appearance model is also one of the parts in AAM. Conversion of the high dimensional input images into a linear subspace of Eigen faces are used by the appearance model. The output of this leads to drastic reduction of the dimension of the parameter space. In this, we need to eliminate the noise in the image by filtering all input images $I(x)$ with a Gauss filter. For piecewise affine transformation $W(x, p)$, the input image will be converted to the basic shape s_0 to $I(W(x, p))$. Then, apply the histogram equalization on these normalized images for the reduction of lighting influences. To normalize the input images in this manner, a PCA is applied on the input images. Select the k components that containing the Eigen values as the appearance components $A_1(x)$ to $A_k(x)$. The mean of all normalized input images will produce the mean appearance components $A_0(x)$.

$$A_0(x) = \frac{1}{D'} \sum_{i=1}^{D'} I(W(x, p)) \tag{10}$$

Depending on the appearance components $A_i(x)$, the generation of an image $A(x)$ with the basic shape s_0 is possible. The following equation shows this.

$$A(x) = A_0(x) + \sum_{i=1}^{m} \lambda_i A_i(x) \tag{11}$$

3.2.1.4. Instances of AAM

Model instance is made by the combination of both shape model as well as the appearance model. The model instance $M(W(x, p))$ indicates the combined appearance model and its shape. Because of this reason, the appearance parameters $\lambda = (\lambda_1, ..., \lambda_m)$ and the shape parameters $v = (v_1, \ldots, v_n)$ are needed. With the use of Equation 4, it is easy to calculate the image $A(x)$ in the form of the basic shape s_0. By the usage of warp $W(x, p)$, the image $A(x)$ can be converted into the shape s.

3.2.1.5. Adaptation of the Model

Adaptation of the trained model to an unknown input image $I(x)$ is important. For this, we need to identify the optimal parameters v and λ.

$$\arg\min_{v,\lambda} \sum_{x \in s_0} \left[A_0(x) + \sum_{i=1}^{m} \lambda_i A_i(x) - I(W(x,p)) \right]^2 \tag{12}$$

We can describe the following error function $E(x)$, for the problem of optimization.

$$E(x) = A_0(x) + \sum^{m} \lambda_i A_i(x) - I(W(x,p)) \tag{13}$$

A variant of the inverse compositional algorithm is used in this paper to solve this optimization problem. Projection algorithm is used in the inverse compositional algorithm. Projection algorithm allows the optimization of the shape parameters v and the appearance parameters λ simultaneously. The appearance parameters λ_i are not present in the optimization, which leads to the original form of the adaptation algorithm problem.

3.3. Texture Extraction Using LGXP

3.3.1. Texture Feature

Texture provides a high-order description of the local image content. The analysis of texture requires the identification of those texture attributes which can be used for segmentation, discrimination, recognition, or shape computation. The structural approach assumes that the texture is characterized by some primitives following a placement rule. In this view, in order to describe a texture one needs to describe both the primitives and the placement rule. The description should be sufficiently flexible so that a class of equivalent textures can be generated by using similar primitives in similar relationships. Therefore, textures suitable for structural analysis have been confined to quite regular textures rather than more natural textures in practice [7-11].

3.3.2. LGXP

In LGXP, phases are firstly quantized into different range, then LXP operator is applied to the quantized phases of the central pixel and each of its neighbors, and finally the resulting binary labels are concatenated together as the local pattern of the central pixel, Figure 3 shows the execution LGXP algorithm.

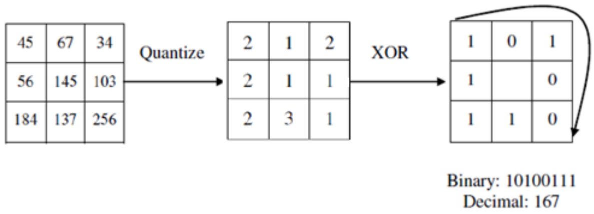


Figure 3. Encoding method of LGXP.

Formally, the pattern of LGXP in binary and decimal form is defined as follows:

$$Lgxp_{\mu,v}(Z_c) = [Lgxp_{\mu,v}^{S}, Lgxp_{\mu,v}^{S-1}, L, Lgxp_{\mu,v}^{1}]_{binary} \tag{14}$$

$$= [\sum_{i=1}^{s} 2^{i-1} \cdot Lgxp_{\mu,v}^{i}]_{decimal} \tag{15}$$

Where Z_C denotes the central pixel position in the Gabor phase map with scale v and orientation μ, S is the size of neighborhood $Lgxp^i_{\mu,v}(i=1, 2, \ldots, S)$, and denotes the pattern calculated between Z_C and its neighbor Z_i, which is computed as follows:

$$Lgxp_{\mu,v}^{i} = q(\Phi_{\mu,v}(z_c)) \otimes q(\Phi_{\mu,v}(z_i)), \quad i=1,2,\ldots,p \tag{16}$$

Where $q_{\mu,v}(\bullet)$ denotes the phase, $\otimes$ denotes the LXP operator, which is based on XOR operator, as defined in Equation 17; $q(\bullet)$ denotes the quantization operator, which calculates the quantized code of phase according to the number of phase ranges, as defined in Equation 17.

$$c - d = \begin{cases} 0, & if \quad c = d \\ 1, & else \end{cases} \tag{17}$$

$$q(\varphi_{\mu,v}(\bullet)) = i; \tag{18}$$

$$f \frac{360 * i}{b} n\, \Phi_{\mu,v}(\bullet) < \frac{360*(i+1)}{b}, i = 0, 1, L, b-1 \tag{19}$$

Where b denotes the number of phase ranges. It is important for LGXP to set appropriate in Equation 17. In order to make the patterns robust to the variations of Gabor phase, the value of cannot be too large. In this study, we find that LGXP performs well enough when b=4. The reason might be that this setting achieves a good balance between the robustness to phase variations and representation power of local patterns.

3.4. Eyes Detection

In our work, the left and right eyes are detected using cascade object detection method. This method helps to detect the objects as per the requirement of object detections. In our work, we need to detect the objects such as left and right eyes from an image. This cascade object detection method works based on an algorithm named as Viola-Jones.

Viola-Jones Algorithm is based on traveling around the input image through sub window proficient of detecting features. Initially, this window is leveled to detect faces of different sizes in the image. A scale invariant detector was extended by Viola Jones and which runs via the image several times with different size for every time that image runs. The detector needs same number of computations for being scale invariant and which not consider the image size. At the initial level, the effortless detectors are used, which discard the windows that do not have face parts. And the subsequent levels use complicated detectors that examine the features in further detail. If only after the face observation could be made via the whole cascade, the face can be detected. From the integral image and Haar resembling features only, these kinds of detectors are constructed.

The image is transformed into an integral image by formulating each pixel that equivalent to the whole sum of all pixels above and to the left of the corresponding pixel. At the last level of our process, we obtain high percentage of face objects. After the detection of face, this algorithm is again applied to the upper face parts for the detection of eye parts only from the detected face object. Thus, we can detect the eye parts using cascade object detector method.

3.5. Classification Using Neural Network

For classifying the age, there are two Feed Forward Neural Network (FFNN) classifiers [10] such as FFNN-1 and FFNN-2 are used. Neural network is a three-layer standard classifier with n input nodes, l hidden nodes and k output nodes. It is noted that if there are two hidden layers are used means, then one hidden layer is to correlate each pair in one meaningful unit and the second is considered to be the real hidden layer after organizing the input data in the first hidden layer. For the classifier FFNN-1, the shape feature points $\{S_1, S_2, \ldots, S_n\}$ that are extracted from the face are given as the input layer and also for the classifier FFNN-2, the make-up face image MF, left and right eyes LE, RE and texture features T_i are given as the input layer. The output layer is the age of the image A_1 and A_2 for both neural networks. FFNN-1 classifier for age classification is given in Figures 4 and 5, FFNN-2 classifier for age classification is given.

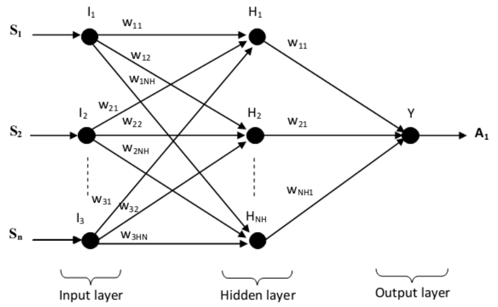


Figure 4. FFNN-1 classifier for age classification.

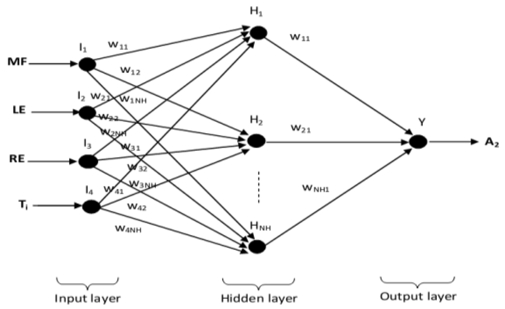


Figure 5. FFNN-2 classifier for age classification.

3.5.1. NN Function Steps

1. For both FFNN-1 and FFNN-2: Set weights for every neuron's except the neurons in the input layer.
2. For FFNN-1: Generate the neural network with the extracted shape feature points $\{S_1, S_2, \ldots, S_n\}$ as the input units, H_{NH} hidden units and age A_1 as the output unit.
3. For FFNN-2: Generate the neural network with the make-up face image MF, left and right eyes LE and RE and texture features T_i as the input units, H_{NH} hidden units and age A_2 as the output unit.
4. For FFNN-1: The calculation of the planned Bias function for the input layer.

$$X=\beta+\sum_{n=0}^{H_{NH}1} w'_{(n)1} S_1(n)+w'_{(n)2} S_2(n)+w'_{(n)3} S_3(n)+\ldots+w'_{(n)3} S_3(n) \tag{20}$$

The activation function for the output layer is calculated as:

$$Active(X)=\frac{1}{1+e^{-X}} \tag{21}$$

For FFNN-2: The calculation of the planned Bias function for the input layer is:

$$X=\beta+\sum_{n=0}^{H_{NH}1} w'_{(n)} MF(n)+w'_{(n)} LE(n)+w'_{(n)} RE(n)+w'_{(n)i} T_i(n) \tag{22}$$

The activation function for the output layer is calculated as:

$$Active(X)=\frac{1}{1+e^{-X}} \tag{23}$$

5. For both FFNN-1 and FFNN-2: Identify the learning error as given below.

$$LE = \frac{1}{H_{NH}} \sum_{n'=0}^{N_{NH}-1} Y_{n'} - Z_{n'} \tag{24}$$

Where LE: Learning rate of both FFNN-1 and FFNN-2. $Y_{n'}$ - Desired outputs, $Z_{n'}$-Actual outputs.

3.5.2. Learning Algorithm-Back Propagation Algorithm Used for Minimizing the Error

In both the FFNN-1 and FFNN-2, Back Propagation Algorithm is used as the Learning Algorithm.

Back Propagation Algorithm steps for both FFNN-1 and FFNN-2.

1. The weights for the neurons of hidden layer and the output layer are assigned by randomly choosing the weight. But the input layer has the constant weight.
2. The planned bias function and the activation function are calculated using Equations 20 and 21 for the FFNN-1 and by using Equations 22 and 23 for FFNN-2.
3. The back propagation error is found for each node and then the weights are updated as follows:

$$w_{(n')} = w_{(n')} + \Delta w_{(n')} \tag{25}$$

4. The weight $\Delta w_{(n')}$ is changed as given below:

$$\Delta w_{(n')} = \delta . X_{(n')} . E^{(BP)} \tag{26}$$

Where δ-Learning rate, which normally ranges from 0.2 to 0.5. $E^{(BP)}$-BP Error.

5. The process is repeated using (2) and (3) steps, until the BP error gets minimized. i.e., $E^{(BP)} < 0.1$.
6. If the minimum value is obtained, then the FFNN-1 and FFNN-2 is well trained for performing the testing phase.

Using the features, both the FFNN-1 and FFNN-2 classifiers are well trained and the Images are tested, accordingly. From the output produced from both the classifiers, the average age of the given image is identified:

$$Average\ Age = \frac{A_1 + A_2}{2} \tag{27}$$

To determine the age, different features are calculated such as: Shape, texture and eye location. These features are unique for each and every person and it does not changes, if the person having different makeups. So based on the unique features the image is first classified into different age groups and then the age can be estimated.

4. Experimental Results and Comparative Analysis

In this section, we have investigated the performance of our proposed novel age estimation system in facial makeover images. Our proposed approach is implemented in MATLAB (7.12). The results showed that our approach has an encouraging performance. We have generated a huge database which contains cini actor and actress images in different getups.

The gradual results attained by the proposed age estimation system are given below. Initially, the input images as in Figures 6-a and 7-b are subjected to the process of face detection. Figure 6-a contains various makeup images and Figure 7-a contains original face images without makeup. Only the face parts are detected from the face detection process and the resultant images are represented in Figures 6-b and 7-b with different makeup images and original images.

a) Original images.

b) Face detected images of the corresponding original images.

Figure 6. Sample output obtained from face detection process (with makeup).

a) Original images.

b) Face detected images of the corresponding original images.

Figure 7. Sample output obtained from face detection process (without makeup).

Figures 8 and 9 represents the shape feature extracted from the facial image using AAM. 50 points are plotted on the eyes, nose, mouth and chin for shape feature extraction using AAM as shown in Figures 8-b and 9-b.

a) Original images.

b) Landmark points of the original images.

Figure 8. Sample output obtained from the shape feature extraction process (with makeup).

a) Original images.

b) Landmark points of the original images.

Figure 9. Sample output obtained from the shape feature extraction process (without makeup).

The texture feature is extracted using local Gabor XOR pattern which is described in Figures 10 and 11. The texture features are also extracted for both the makeup images and original images without makeup.

a) Sample images from the database.

b) LGXP images.

Figure 10. Sample output obtained from the texture feature extraction process (with makeup).

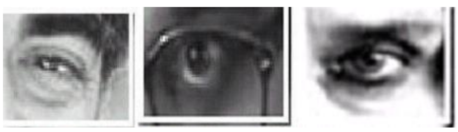
a) Sample images from the database.

b) LGXP images.

Figure 11. Sample output obtained from the texture feature extraction process (without makeup).

The highlight of our proposed method is the eye detection. Both the left and right eyes are detected using cascade object detection method, which works with the aid of Viola-Jones Algorithm. The detected eyes are shown in Figures 12 and 13 for both the makeup images and original images without makeup.

a) Sample images from the database.



b) Eye images.

Figure 12. Sample output obtained from the eye detection process (with makeup).

a) Sample images from the database.

b) Eye images.

Figure 13. Sample output obtained from the eye detection process (without makeup).

- *Comparative Analysis*

The performances of the age estimation are usually evaluated by two different measures:

Mean Absolute Error (MAE) and Cumulative Score (CS) are the evaluation measures. The MAE is defined as the mean of absolute error between the estimated ages and the ground truth.

$$MAE = \frac{\sum \left| l_k - l_k^* \right|}{N} \tag{28}$$

Where l^*_k the estimated age for the sample is, l_k is the ground truth age of the sample and N is the total number of testing images. The cumulative score is defined as:

$$CS(j) = N_{e \le j} / N * 100\% \tag{29}$$

Where $N_{e\le j}$ is the number of test images on which the age estimation makes the absolute error no higher than j years. We propose the estimation error over than 10 years is unacceptable and plots CS curves with $j \le 10$.

We conduct a comparative study on performances of four kinds of regressions on the designed feature set: Multi-Layer Perceptron (MLP), Support Vector Regression (SVR) [14] and MPCA.

Figure 14 represents the cumulative score for different existing methods and Table 1 represents the MAE for different existing methods. From MAE and CS curve of human beings on our database, one can see that there exists apparent ambiguity in age perception.

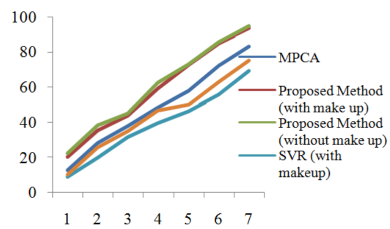


Figure 14. Cumulative score for different methods.

Table 1. MAE of different classifiers.

Methods	MAE in Years	
	Without Makeup	With Makeup
SVR	5.6	17.1
MLP	5.1	15.7
MPCA	3.9	14.3
Proposed	2.9	6.1

For the makeup images, the error rate MAE of Proposed method is low in compared with the existing methods. Our proposed method facilitates only 6.1 MAE value for makeup images but the existing methods SVR, MLP and MPCA gives 17.1, 15.7 and 14.3 MAE values respectively. And also, for the original images without makeup, our proposed method provides low MAE (2.9) than the existing methods. Other existing methods SVR, MLP and MPCA give 5.1, 3.9, and 5.6, MAE values, respectively for images without make up. The lower error rate in our proposed method leads to increase in the performance of estimating the age of a person. From Table 1 and Figure 15, we observed that our proposed method efficiently estimated the age of a human from the facial images with makeup.

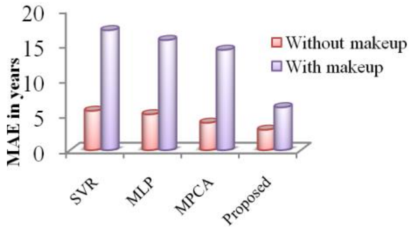


Figure 15. Comparison based on MAE.

5. Conclusions

In this paper, we have implemented a frame work for face image based automatic age estimation. In this paper, we have proposed the age estimation system based on AAM and unique features. The estimation process is based on the facial feature extracted from the normalized images. Unique features are efficiently estimating the image with less error rate. The age estimation process is done by employing ANN method. The implementation results illustrates that this apparent age estimation process effectively estimates the ages for the facial images from the database when compared to the age estimation systems that is in existence. This could be visualized from the comparative analysis. Thus, we conclude that an apparent age estimation based on AAM and unique feature set gives better age estimation than the other existing methods.

References

[1] Akeel M. and Mishra R., "ANN and Rule Based Method for English to Arabic Machine Translation," *The International Arab Journal of Information Technology*, vol. 11, no. 4, pp. 396-405, 2014.

[2] Chiang C., Tai W., Yang M., Huang Y., and Huang C., "A Novel Method for Detecting Lips, Eyes and Faces in Real Time," *Journal of Real-Time Imaging-*, vol. 9, no. 4, pp. 277-287, 2003.

[3] Choi S., Lee Y., Lee S., Park K., and Kim J., "Age Estimation Using a Hierarchical Classifier Based on Global and Local Facial Features," *ELSEVIER Journal of Pattern Recognition*, vol. 44, no. 6, pp. 1262-1281, 2011.

[4] Gao F. and Ai H., "Face Age Classification on Consumer Images with Gabor Feature and Fuzzy LDA Method," *in Proceedings of the 3rd International Conference on Advances in Biometrics*, Italy, pp. 132-141, 2009.

[5] Gao P., "Facial Age Estimation Using Clustered Multi-task Support Vector Regression Machine," *in Proceedings of the 21st International Conference on Pattern Recognition*, Tsukuba, pp. 541-544, 2012.

[6] Geng X., Smith-Miles K., and Zhou Z., "Facial Age Estimation by Nonlinear Aging Pattern Subspace," *in Proceedings of the 16th ACM International Conference on Multimedia*, NY, pp. 721-724, 2008.

[7] Geng X., Zhou Z., and Smith-Miles K., "Automatic Age Estimation Based on Facial Aging Patterns," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 12, pp. 2234-2240, 2007.

[8] Guo G., Fu Y., Dyer C., and Huang T., "Image-Based Human Age Estimation by Manifold Learning and Locally Adjusted Robust Regression," *Journal IEEE Transactions on Image Processing*, vol. 17, no. 7, pp.1178-1188, 2008.

[9] Hewahi N., Olwan A., Tubeel N., EL-Asar S., and Abu-Sultan Z., "Age Estimation Based on Neural Networks using Face Features," *Journal of Emerging Trends in Computing and Information Sciences*, vol. 1, no. 2, pp. 61-67, 2010.

[10] Kilinc M. and Akgul Y., "Human Age Estimation Via Geometric And Textural Features," *in Proceedings of the International Conference on Computer Vision Theory and Applications*, vol. 1, pp. 531-538, 2012.

[11] Li Z., Park U., and Jain A., "A Discriminative Model for Age Invariant Face Recognition," *IEEE Transactions on Information Forensics and Security*, vol. 6, no. 3, pp. 1028-1037, 2011.

[12] Ramanathan N., Chellappa R., and Biswas S., "Age Progression in Human Faces: A Survey," *Journal of Visual Languages and Computing*, vol. 15, pp. 3349-3361, 2009.

[13] Ramezanpour M., Azimi M., and Rahmati M., "A New Method for Eye Detection in Color Images," *Journal of Computer Engineering*, vol. 1, pp. 3-11, 2009.

[14] Roy H., Bhattacharjee D., Nasipuri M., and Basu D., "Age Range Estimation from Human Face Images Using Face Triangle Formation," *International Journal of Research and Reviews in Information Sciences*, vol. 2, no. 1, pp. 155-160, 2012.

[15] Ruiz-Hernandez J., Crowley J., and Lux, A., "'How Old are You?': Age Estimation with Tensors of Binary Gaussian Receptive Maps," *in Proceedings of British Machine Vision Conference*, UK, pp. 1-11, 2010.

[16] Thakur S. and Verma L., "Identification of Face Age Range Group using Neural Network," *International Journal of Emerging Technology and Advanced Engineering*, vol. 2, no. 5, pp. 250-254, 2012.

[17] Tin H. and Sein M., "Effective Method of Age Dependent Face Recognition," *International Journal of Computer Science and Informatics*, vol. 1, no. 2, pp. 17-21, 2011.

Tamilb Vairavan obtained her Bachelor's degree in Electronics and Instrumentation from Annamalai University. Then she obtained her Master's degree in Applied Electronics from University of Madras and pursuing Ph.D in Digital Image processing in Anna University, Chennai. Currently, she is an Associate Professor at the Faculty of Electronics and Instrumentation, R.M.K Engineering College, Chennai. Her current research interests are Server, Digital Image Processing, Biometrics and Virtual Simulation.

Kaliyaperumal Vani received the B.E. degree from Anna University, Chennai, the M.E degree from Anna University , Chennai and the Ph.D. degree from Anna University, Chennai. She is currently an Associate Professor in the Department of Information Science and Technology at College of Engineering, Anna University, Chennai. Her research interests are remote sensing and Digital Image Processing and Signal Processing.