University of Tartu

Faculty of Science and Technology

Institute of Technology

Ozan Bilici

Local Phase Quantization Feature Extraction based Age and Gender Estimation Using Convolutional Neural Network

Masters Thesis (30 ECTS)

Robotics and Computer Engineering

Supervisor:

Assoc. Prof. Gholamreza Anbarjafari

Tartu 2017

Abstract

Local Phase Quantization Feature Extraction based Age and Gender Estimation Using Convolutional Neural Network

Even though artificial neural networks are one of the oldest machine learning techniques, there were no many experiments on them by 2010s because of its computational complexity. Artificial neural networks got inspired by human neural anatomy, and try to achieve similar accuracy. Latest advances of silicon technology enable us to conduct experiments on all types of artificial neural networks. Convolutional Neural Networks are one of state-of-art neural network types. As a human, we all have great recognition, detection mechanism in our body. In this study, it will be attempted to gain similar ability with computer aid of CNNs. As all other supervised-learning methods, we need training and testing dataset. We are going to apply CNN on apparent age and gender estimation. There are few public dataset which are created for age estimation. One of them and the biggest one is IMDB-Wiki dataset which contains pictures of famous people from wikipedia and IMDB with their real-age label. In order to create real-age label, the creator used the time differences between photo-taken year and birth year. However for better accuracy, we need apparent age information. Because aging is a process that depends on many conditions. As it is going to be explained later, we collected Japanese dataset on the internet, and labeled their apparent ages by weighted voting. After collecting the image data sets, we pre-processed the images with face detection and alignment methods. Afterwards, we copied all images and used Local Phase Quantization(LPQ) method to extract their features. In CNN, it is always better to use pre-trained data and fine-tune it. Thus we used deep face recognition pre-trained data with almost 2 millions images. After that, we fine tuned images(with LPQ and without LPQ separately) with using the label distribution encoding. Finally we had 2 CNN data. For combining the results, we took the mean of all respective output neurons. At the end, expected values of all neurons are considered as apparent age information. For gender classification, we just trained the system in the similar way. Only difference is that we have only 2 output neurons for gender classification, besides LPQ is not applied in gender classification.

CERCS: T120 Systems engineering, computer technology; T125 Automation, robotics, control engineering; T111 Imaging, image processing

Keywords: histogram equalization, face detection, face alignment, label distributed encoding, local phase quantization, neural network, convolutional neural network, deep learning, VGG-16, VGG-19, age estimation, gender classification

Kokkuvõte

Lokaalse Faasi Kvantimise Tunnusjoonte Eraldamisel põhinev Vanuse ja Soo Hindamine Kasutades Konvulutsionaalset Närvivõrku

Tehisnärvivõrgud on ühed vanematest masinõppe meetoditest, kuid nende arvutusliku komplekssuse tõttu ei ole neid enne 2010ndaid väga palju rakendatud. Tehisnärvivõrkude peamiseks eeskujuks ja täpsuse eesmärgiks on inimese närvisüsteem. Uuemad saavutused räni tehnoloogias võimaldavad rakendada komplekssemaid tehisnärvivõrgu tüüpe, näiteks konvulutsionaalseid närvivõrke(CNN). Antud uuringus üritatakse arvutites saavutada sarnast eristus- ja tuvastusvõimekust kui inimestel kasutades CNN. Me rakendame näiva vanuse ja soo hindamiseks CNNi, mistõttu vajame treening- ja katseandmestikke nagu kõigis järelvalvega õppemeetodites. Hetkel ei ole olemas väga palju vanuse hinnangu andmestikke, kuid eksisteerivatest kõige suurem on IMDB-Wiki andmestik. See andmestik sisaldab pilte kuulsustest nende päris vanustega Wikipeediast ja IMDBst Andmestiku loojad kasutasid päris vanuste arvutamiseks pildi loomise kuupäeva ja pildil oleva isiku sünniaasta vahet. Parema täpsuse saavutamiseks on vaja näiva vanuse informatsiooni sellepärast, et vananemine on mitmetest faktoritest mõjutatud protsess. Me korjasime internetist jaapanlaste nägude andmestiku ja määrasime igale näole vastavad näivad vanused kasutades häälteenamust. Pärast andmete kogumist eeltöötlesime pilte näo tuvastus- ja joondamismeetoditega. Järgnevalt kopeerisime kõik pildid ja kasutasime lokaalse faasi kvantimis (LPQ) meetodit, et eraldada tähtsad tunnused. CNNde puhul on alati mõistlikum kasutada treenimiseks andmeid, mis on varasemalt välja töötatud, ja siis neid häälestada. Meie kasutasime kahest miljonist pildist koosnevaid andmeid, mis olid mõeldud näo tuvastus närvivõrgu treenimiseks. Järgnevalt me häälestasime pilte (LPQ'ga ja ilma) kasutades märgendite jaotus kodeeringut. Tulemuseks saime 2 CNN andmestikku ja nende kombineerimiseks kasutasime kõigi väljund neuronite keskmiseid. Kõigi neuronite oodatud väärtused määravad näiva vanuse. Soo tuvastussüsteemi puhul me ei kasutanud LPQ'd ja tulemuse määravad ainult kaks väljund neuronit.

CERCS: T120 Süsteemitehnoloogia, arvutitehnoloogia; T125 Automatiseerimine, robootika, control engineering; T111 Pilditehnika

Märksõnad: histogrammi ühtlustus, näo tuvastus, näo joondumine, märgendite jaotus kodeering, lokaalse faasi kvantimine, närvivõrk, konvulutsionaalne närvivõrk, süvaõpe, VGG-16, VGG-19, vanuse hindamine, soo hindamine

Contents

List of Figures

List of Tables

Abbreviations, constants, definitions

CNN - convolutional neural network

LAP - looking at people

DEX - deep expectation of apparent age from a single image

LUT - lookup tables

DCT - discrete cosine transform

KNN - k-nearest classifier

AGE - age and gender estimation

LBP - local binary pattern

HOG - histogram of oriented gradients

PSF - point spread function

STFT - short time fourier transform

LPQ - local phase quantization

CPU - central processing unit

GPU - graphics processing unit

BVLC - berkeley vision and learning center

BSD - berkeley software distribution

iCV - intelligent computer vision

MAE - mean absolute error

1 Introduction

Age and Gender Estimation (henceforth it will be referred as AGE) tries to estimate age and gender of a person on taken-photo. The system based on two subsystems such as training and classification (note that classification and estimation are used as equivalent in this study). Figure 1.1 and 1.2 show the system implementations. In addition Figure 1.3 illustrates the all image processing steps.

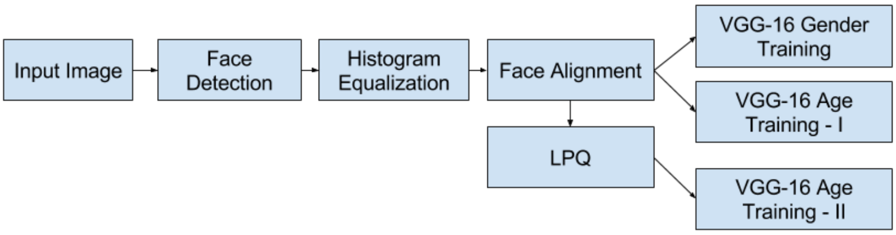


Figure 1.1: Training Block Diagram of AGE

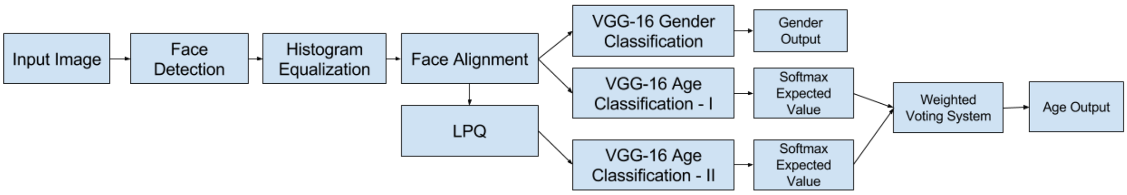


Figure 1.2: Classification Block Diagram of AGE

Nowadays everything becomes smart and intelligent such as our mobile phones, computers, and cars. Many companies try to produce product for some certain age and gender groups. So in order to increase the intelligence level of electronic devices, AGE system can be employed.

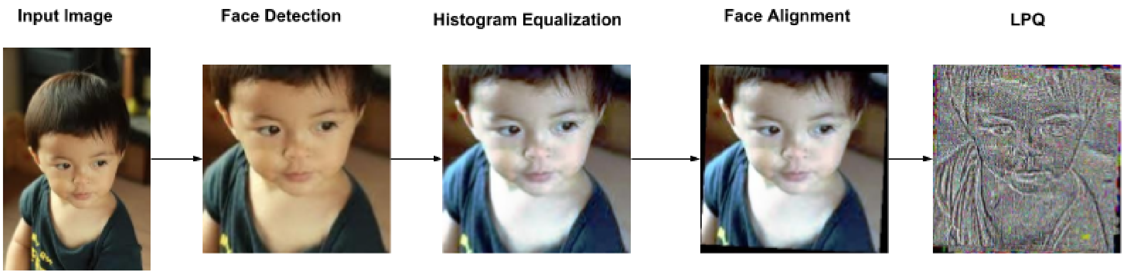


Figure 1.3: Image processing steps

AGE system would help cosmetic sector pretty well as much as some website restrictions for infants or teenagers. Another useful system can be vending machines, for example, in Japan, people can obtain various of products from vending machines. Let's assume that the person would like to buy alcohol by the vending machine, the vending machine would scan the person by AGE system due to the obligations of the local law.

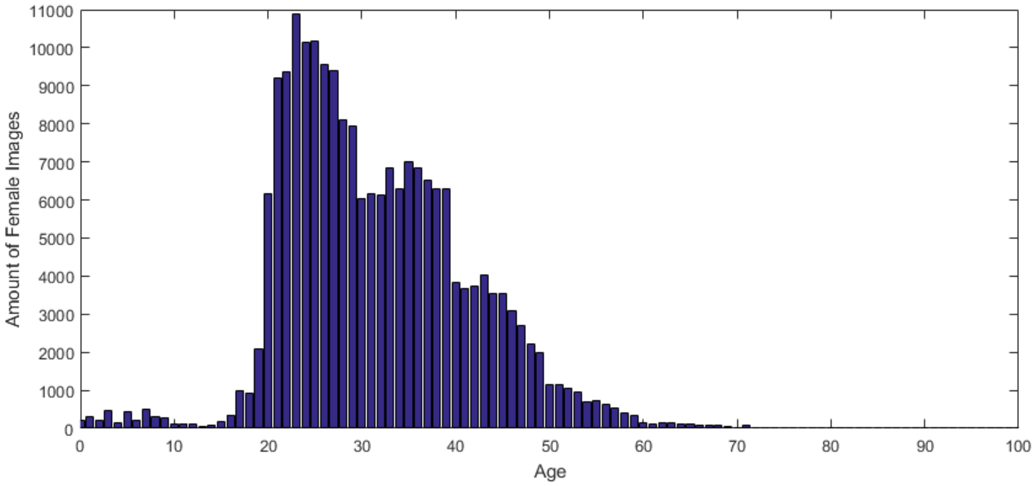
Detecting gender is classification problem because of its discrete nature, whereas age estimation could be considered as regression problem. In this study, AGE system considered both as classification problems. Age estimation is very challenging and important topic. One of the reason why age estimation is challenging because lack of dataset with age labels. Besides it can be considered that two types of age information exist such as apparent and real age. Aging is a process which also depends on environment and life style. So we will focus on apparent age information because estimating real age would need more information as input data.

In this study, a novel method was introduced to estimate age and gender using Japanese Dataset with the help of local phase quantization technique and by using label distribution encoding and convolutional neural network.

2 Japanese Age and Gender Dataset

AGE system is developed for Japanese people, besides there was no dataset specific for Japanese people. Thus dataset had to be collected over Internet. iCV group [10] aided for collecting and labelling it by manual work through the Internet.

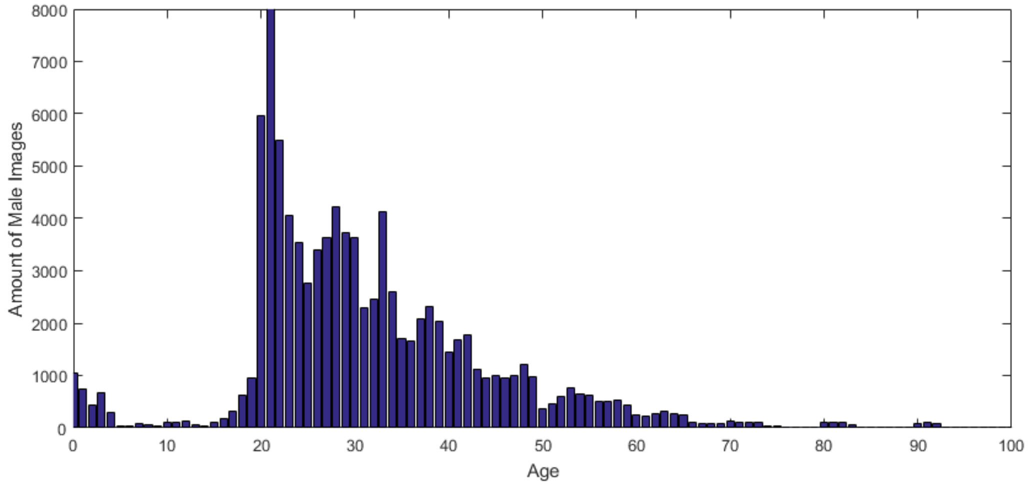
Due to developing both age and gender classification, dataset is labelled as apparent age for both Japanese females and males separately. Main difficulty was that we were few and images were many. Thus standard deviation of apparent age estimation are very high in this dataset. In result section, labelling will be discussed with more details. Figure 2.1 and 2.2 illustrate the age distribution in female and male group, respectively. Additionally Figure 2.3 shows the age distribution in both female and male group in total.



(a)

Figure 2.1: Age Distribution of Female Images

It is clearly seen that people, aged between 20 years old and 30 years old, outnumbered the other age groups on the Internet. In order to create accurate system, amount of the images in every age group should be very similar and high enough to train. As a matter of fact, our dataset is far from the ideal condition for very accurate system.



(a)

Figure 2.2: Age Distribution of Male Images

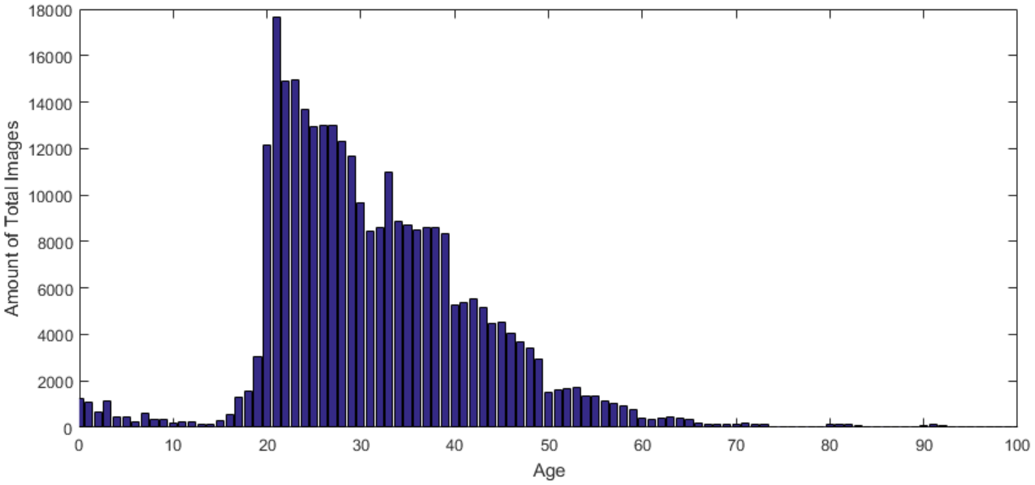


Figure 2.3: Age Distribution of Both Male and Female Images

Table 2.1 indicates the amount of images in male and female categories (refer to Appendix 1 for more details regarding to dataset). In our dataset, the amount of female images outnumbered male counterparts. While both gender and age training, dataset were reordered and some images were removed to prevent poor results in gender classification.

Table 2.1: Table to image statistics

Statistic of Images			
	Male Images	Female Images	Total Images
Amounts	96984	204916	301900

3 Related Work

3.1 Age Estimation

As it was mentioned earlier, there are two types of age estimation such as real and apparent. Even though many studies related to real age estimation have been done, apparent age estimation is still very much in its infancy stage.

In the following method [7], the author applied age estimation based on facial aging patterns. The author proposed a method basically used particular person's photos in chronological time order, and find the representative version of the image on subspace that can reconstruct the face image with minimum error.

In this paper [6], the author uses label distribution. Instead of assigning image to the single label, the author associate each image with label distribution. Triangle and Gaussian distributions are used and tested in this method.

The author [11] implemented real age estimation by using wrinkle feature extraction. Firstly wrinkle areas of face are located, then the feature extraction methods were applied. Each image were clustered by fuzzy c-mean clustering algorithm. The estimated age is calculated by clustering membership value and average of each cluster.

ChaLearn Looking At People is a competition which was held in 2015 [4] and 2016 [5]. After these competitions, apparent age estimation became more popular. They provide one of the largest dataset known for apparent age estimation. The dataset contains training data, validation data and finally test data to measure the accuracy of the methods were competed in the competition. Most of the methods used in the competition are based on convolutional neural networks.

Deep EXpectation of Apparent Age From a Single Image(DEX) [26], which uses the Convolutional Neural Network - VGG-16 [30], was the winner of ChaLearn LAP 2015 apparent age estimation challenge [4]. They considered the problem as a classification problem between 0 to 100 year(s) old, that's why they used 101 age labels. IMDB-WIKI [26] dataset was created with the images crawled from IMDB and Wikipedia webpages by them. They fine-tuned the VGG-16 model on already pre-trained ImageNet [28] dataset with the dataset they created. They splitted the ChaLearn LAP 2015 [4] dataset into 20 different groups, further they fine-tuned 20 models by using 90% of each groups for training and 10% for validation. The face alignment methods was not applied in their solution. Images were rotated by -10 to 10, translated by -10% to 10% of the size and scaled by 0.9 to 1.1 of original image. They did the augmentation ensuring there is no overlap between validation and training dataset. Then they trained and picked the best performance on the validation set. The final prediction was the average of outputs of the each model.

AgeNet [19] considered problem as both classification and regression problem. They trained respectively real-value based regression and gaussian label distribution based classification models. For both models, large-scale deep convolutional neural network is deployed. Firslty they pre-trained web collected face dataset with identity label, afterwards they fine-tune it real age dataset with noisy age label. At the end, they fine-tune one more time with the apparent age dataset which was provided by ChaLearn LAP 2015 [4].

Although Zhu *et al.* [34] applied CNNs also, the reason of it was different than the other methods. CNNs was conducted in order to extract features. Afterwards support vector machine, support vector regressor and random forest were used to estimate apparent age information.

In the second edition of the ChaLearn competition [5], number of images inside the dataset were increased dramatically. Age distribution of the dataset was also changed. Especially the percentage of the children images were significantly increased. Thus winner of the second competition [5] [2] had many improvement of the system in order to achieve better solution. Because of the success of CNN, they followed the same pipeline for their CNN. Firstly fine-tuning on pre-trained dataset with a large datasets, secondly fine-tuning it again with the competition dataset. The authors fine-tuned 2 separate CNNs. One was for all age labels but it was significantly differed than the DEX. They applied label distribution encoding [5] in their solution. Besides the second CNN was just for the children between 0 to 12 year(s) old. Firstly, test data was used in first type of CNN. Second type of CNN was just used in the case the first result was not above 12 years old. They outperformed in the competition.

Refik *et al.* [20] conducted their method by also using CNNs. As the other competitors of ChaLearn competition [5], they were also loyaled the [26] method. Main distinguish is that instead of using single label, they created three different age groups. Because of having these age groups, they trained their system as three models. In order to calculate the apparent age, average of these three models were used.

3.2 Gender Classification

Gender classification is basically binary classification problem [3], [15], [33]. In [21], support vector machine was used for gender classification. It was one of the first study about gender classification. According to the study, the results were outperformed to human.

Wu et al. [32] applied LUT-type weak classifier based on Adaboost learning method for face detection and gender classification. Local binary patterns are known as good for texture classifications. In the following paper [9], the author presents variant method for local binary patterns. After presented method were applied, histogram of local binary patterns were used for features for gender classification by applying support vector machine.

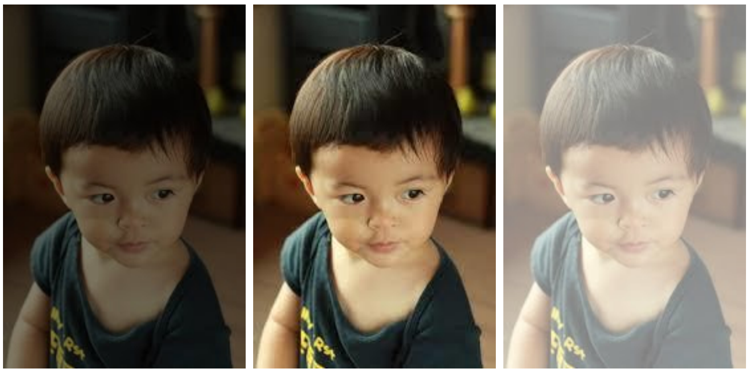
In [22], discrete cosine transform (DCT) is deployed for feature extraction and sorted the high the features with high variance. Afterwards K-nearest classifier (KNN) is applied for training. The author claimed that his method achieved over $\%99$ accuracy. In [18], four layer CNNs were applied. Convolution operation was replaced with cross-corelation in order to reduce the computational complexity. The results show above $\%98$ accuracy in two different dataset.

4 Methodology

In this section, the methods, that are used in system implementations, are going to be explained.

4.1 Histogram Equalization

The contrast and the light are very important in image processing operations. The differences between the image taken in dark environment and the image taken in light environment are very different in pixel level. Figure 4.1 shows the same image in different light condition.



(a) Darker Image (b) Normal Image (c) Lighter Image

Figure 4.1: Japanese infants in three different light condition [8]

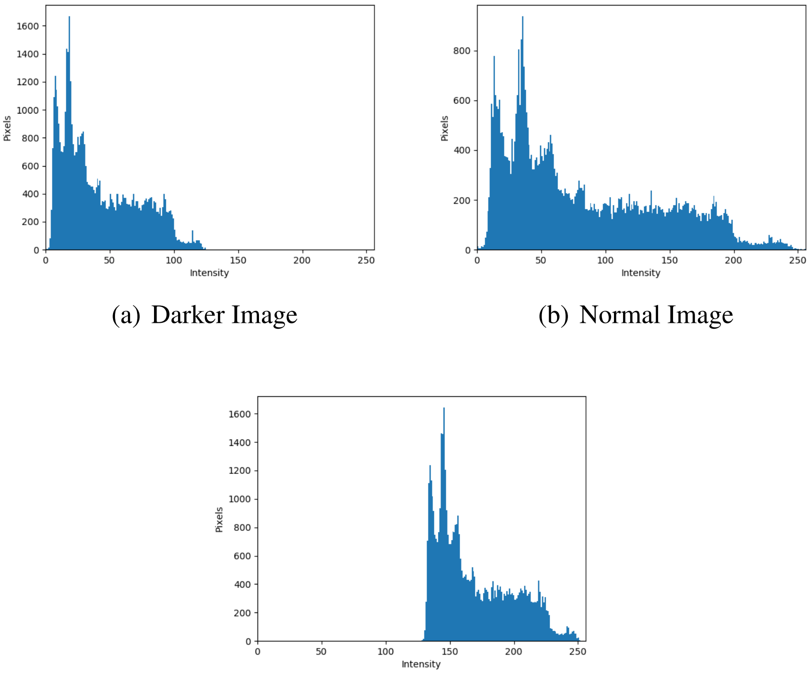
Figure 4.2 shows the histogram graph of corresponding images. Problem can be seen easily. We should avoid the distribution of pixels in such a small band in bright and dark image. We can do this by applying histogram equalization.

Histogram equalization is very simple. Let's assume f be a given image represented as a I_{xy} two dimensional matrix of integer pixel intensities ranging 0 to $L-1$. In gray scale image, or any channel of image, L value is usually 256. Let's also assume p denotes possibility value of each intensity pixel of the image. i is in range of 0 to $L-1$.

$$p_i = \left(\frac{number\ of\ pixels\ with\ intensity\ value\ of\ i}{total\ number\ of\ pixels} \right) \tag{4.1}$$

After finding the possibility values of pixels, we can obtain the $Ihist_{xy}$ by applying formula 4.2. Floor is the rounding down function. Figure 4.3 and 4.4 show the images and their corresponding histograms after applying histogram equalization. All images in Figure 4.3 look like identical for human eyes.

$$Ihist_{xy} = floor\left((L-1) \sum_{i=0}^{I_{xy}} p_n \right) \tag{4.2}$$

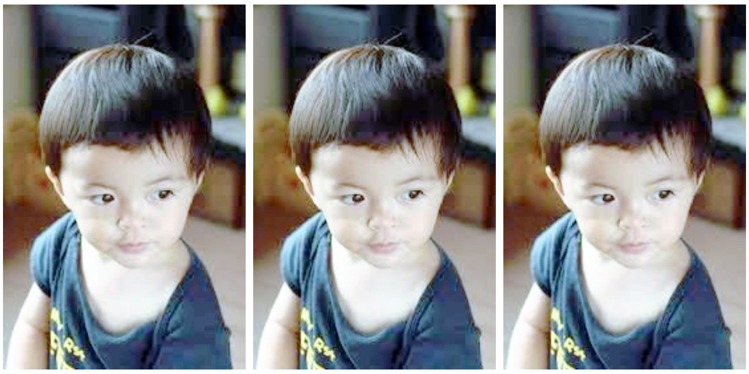


(a) Darker Image

(b) Normal Image

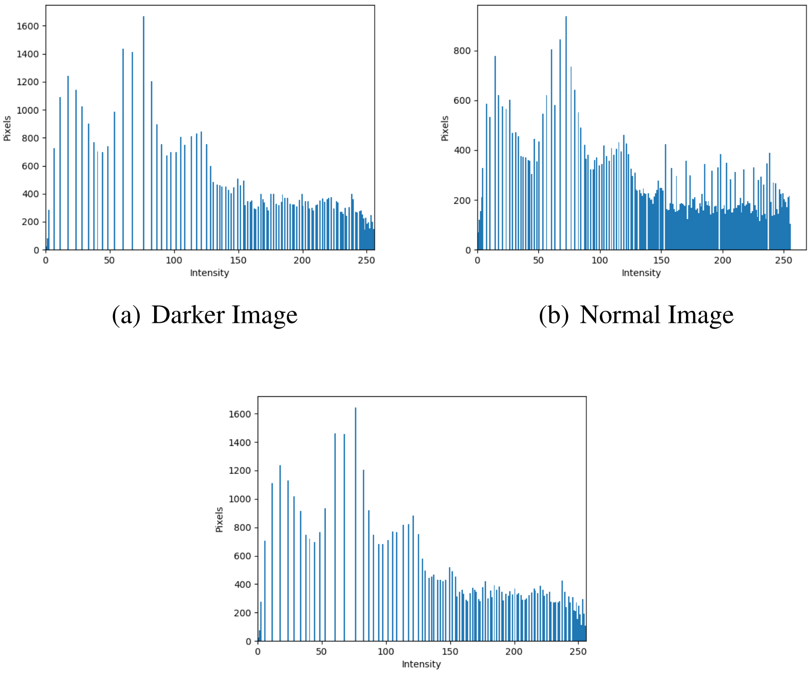
(c) Lighter Image

Figure 4.2: Histogram graphs of Japanese infants



(a) Darker Image (b) Normal Image (c) Lighter Image

Figure 4.3: Japanese infants after histogram equalization



(a) Darker Image (b) Normal Image

(c) Lighter Image

Figure 4.4: Histogram graphs of Japanese infants after histogram equalization

4.2 Face Detection

Face detection is very important step in age and gender estimation. There are various methods for detecting faces. In our early development phase, we were willing to use Viola-Jones face detector. Viola-Jones is supported by many image processing libraries such as OpenCV and MATLAB. Although it is very easy to use Viola-Jones, it performed very poor. The Japanese age dataset was collected (approximately 400.000 images) from Internet and labelled as a survey by iCV [10]. Applying the Viola-Jones face detector on our dataset, we lost more than 50% of the dataset. That's why we decided to use dlib library and OpenCV together. Face detector of dlib library is made using the Histogram of Oriented Gradients (HOG) feature combined with a linear classifier, an image pyramid, and sliding window detection scheme.

In order to increase the performance, we decided to rotate all the images in the range of [-45°, +45°]. Then applied face detector. Dlib library also produces the score. Thus we were able to choose faces with the highest score. Additionally, we cropped the faces in a box with a 20% margin in order to obtain additional features such as hairs.

Basic algorithm can be seen as following :

Algorithm 1 Face detection algorithm

```
$image\ \{image\}$
main:
$rotation\_angle \in \{-45, ...., +45\}$
loop
    $image\_new = rotate\_image(image, rotation\_angle)$
    $face, score[rotation\_angle\_index\_value] = face\_detection(image\_new)$
end loop
$max\_score\_angle = max\_value(score[rotation\_angle\_index\_value])$
$image\_new = rotate\_image(image, max\_score\_angle)$
```

Figure 4.5 illustrates the full images of the people, and Figure 4.7 shows after face detection algorithm.

As you can see Figure 4.7, resolution of the image is also increased to 256x256x3. The main reason will be explained in details, in the section of CNN and VGG-16.

Figure 4.5: Three Japanese Infant's Photos [8]

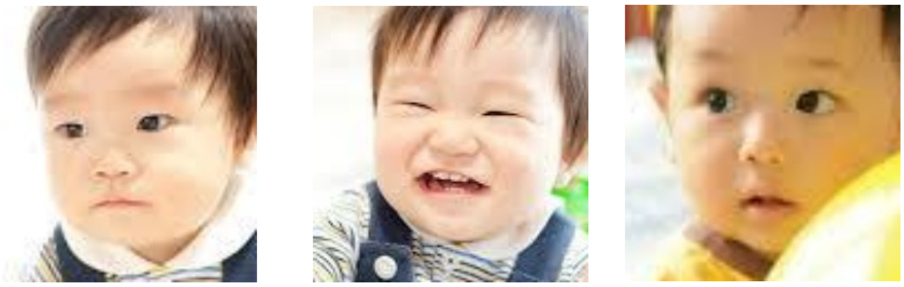


Figure 4.6: Detected Faces Photos [8]

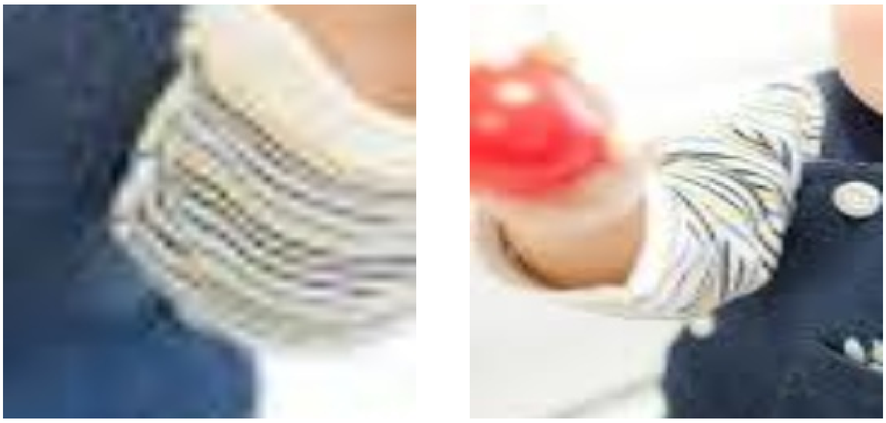


Figure 4.7: False Detected Faces [8]

4.3 Face Alignment

In order to train system well, there are few options. One would be to rotate images in all angles, and feed them to the CNNs. As it is obvious that it will cause extra computational complexity to system. Let's assume that when there is 100 images and they are rotated in the range [+180, -

180] by the step value 0.5. It is certain that we need to feed our CNNs by 100 images multiplied by $(+180 - (-180)) * (1/0.5)$ which is equal to 7200 images. As much as we increase the photos, the needed image amount will be increased by 720. Although similar approach is used in [26] and being successful, we would like to change that approach in our system. Therefore we decided to do face alignment to all face detected images. This approach will help us to reduce computational complexity dramatically that is needed by CNNs.

Face alignment is a technique that used to align all faces in similar angle. We can align all images upside down or normal or in any angle. Before applying face alignment, as it is obvious that we need some reference values. We applied face alignment algorithm inside dlib [14]. This library uses the article of ”One Millisecond Face Alignment with an Ensemble of Regression Trees” [13] to implement alignment. This algorithm extracts facial landmarks, then it very is easy to align faces. The algorithm will be called facial landmark detector by so on.

Facial landmark detector inside the dlib is used to estimate the location of 68 coordinates that is shown in Figure 4.10. These annotations are part of the 68 points iBUG 300-W dataset [29] that the dlib facial landmark detector was trained on.

Additionally in original article [13], the dataset used is different. By different, it is meant that it uses 194 points model HELEN dataset [16]. Dlib library provides training functionality. That’s why the user can apply HELEN dataset for better resolution. In our study, we do not need more landmarks to align faces. We sticked with dlib implementation.

Figure 4.8 illustrates, 68 facial landmark points on real examples on Japanese infants. After detecting these 68 facial landmark points, we need to rotate the images to the same order. As it is seen in Figure 4.10, coordinates of points 37 and 46 should be approximately on the same X axis as well as point 28 and 31 should be approximately on the same Y axis. Basic trigonometry enables us to find angle between these points. Then we just need to rotate image. Formula 4.3, 4.4, 4.5 would work and easy to implement. Figure 4.9 shows rotated image.

$$d_x = P_{46,x} - P_{37,x} \tag{4.3}$$

$$d_y = P_{37,y} - P_{46,y} \tag{4.4}$$

$$\theta = atan(-dy, dx) \tag{4.5}$$

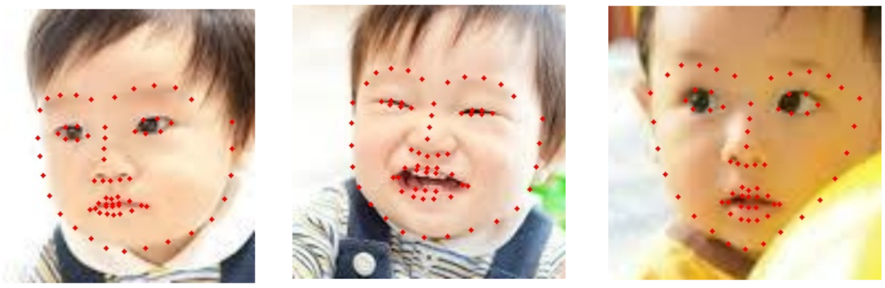


Figure 4.8: Facial Landmarks on Japanese Infants

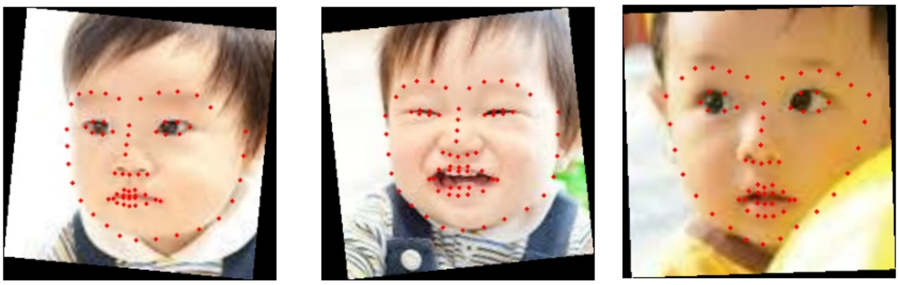


Figure 4.9: Facial Landmarks on Japanese Infants

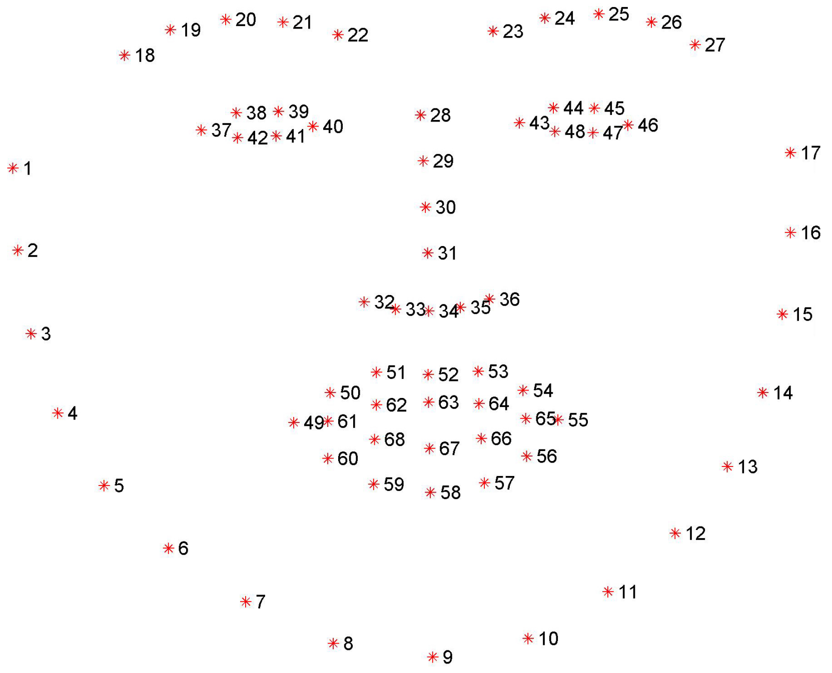


Figure 4.10: Facial Landmarks

4.3.1 Local Phase Quantization

Local phase quantization (LPQ) is a proposed descriptor for texture classification in 2008 [23]. According to their experiments, their method achieved higher classification accuracy of blurred texture image than the other methods such as local binary pattern (LBP) [1], [25] or Gabor filter bank methods [23]. This method is used also for face recognition in other paper [17].

In AGE system, we tend to apply blur invariant image as an input to CNNs, which will be discussed in the next sections. Thus we chose LPQ as feature extraction method. As a matter of fact in aging process, human face gets wrinkles etc. Generally speaking texture classification algorithm would be very similar to detect that similar features on human face. In this perspective, we applied LPQ as an aid system.

4.3.1.1 Mathematical Background

The following mathematical explanations are taken from [23] and [24]. For further explanation the reader shall look at these articles which mathematical equations deeply are covered by publishers of LPQ. Blurring can be represented by a convolution between intensity value of the image and point spread function (PSF) [23]. In Fourier domain, it corresponds to

$$G = F.H \tag{4.6}$$

where G, F and H are discrete Fourier transform of blurred image, original image and PSF respectively. The equation can be splitted into magnitude and phrase parts as 4.8, 4.7.

$$\|G\| = \|F\|\|H\| \tag{4.7}$$

$$\underline{/G} = \underline{/F} + \underline{/H} \tag{4.8}$$

When the assuming the PSF of the blur is centrally symmetric, Fourier transform of H is always real value [24]. By this, it is meant that $\underline{/H}$ can either be 0 or π. Additionally H functions shape for a regular PSF is similar to Gaussian or sinc function which ensures low frequency values of H are positives [24]. In these frequencies it causes to $\underline{/F}$ to be a property of blur invariant [24]. Local spectra of LPQ are computed by short-time fourier transform (STFT) [24].

Coefficients of local Fourier transform are computed at four frequency points such as:

$$u_1 = [a, 0]^T,\ u_2 = [0, a]^T,\ u_3 = [a, a]^T,\ u_4 = [a, -a]^T \tag{4.9}$$

where a is small enough scalar number to satisfy $H(u_i) > 0$. The result for each pixel position in a vector [24] :

$$F_x = \begin{bmatrix} F(u_1, x)\ F(u_2, x)\ F(u_3, x)\ F(u_4, x) \end{bmatrix} \tag{4.10}$$

The phase information is obtained by using a simple scalar quantization [24]:

$$g_n = \begin{cases} 1, & \text{if } g_n \geq 0 \\ 0, & \text{otherwise} \end{cases} \tag{4.11}$$

where g_n is the n-th component of the vector [24] :

$$G(x) = \begin{bmatrix} \Re(F(x))\ \Im F(x) \end{bmatrix} \tag{4.12}$$

As a result, eight binary coefficients g_n are encoded in a range of 0 and 255 as following [24] :

$$F_{LPQ}(x) = \sum_{n=1}^{8} g_n 2^{n-1} \tag{4.13}$$

Feature vector of 256-dimension are found by histogram of values which compose these values from all positions [24].

4.3.2 Encoding

There are few methods for encoding approach which are:

1. Real value encoding. When the problem takes into account as regression problem, this method is the one shall be used. In this method, real age values are assigned to the image. By this, it is meant that if the person is 40 years old, label value will be also 40 years old. You may recognize that in gender classification, the person may be male or female. So there are just two classes. Even in real value encoding, it turns it to classification problem.

2. Binary value encoding. In this method, it is obvious that the problem turns into classification problem. For gender classification, it is obvious that there should be single output value 0 or 1. But if we consider it as neural network problem, there should be two output neurons which are [1,0] or [0,1]. Let's get back to age classification binary value encoding. We can also call output neurons as vectors. So here, first number of classes should be defined such as 101. Every single vector will contain 100 values but in binary just only one value will be 1 and the rest will be 0. For the real age value of 0, vector will look like [1, 0, 0, ..., 0] and the real value of 1, vector will look like [0, 1, 0, 0 ...., 0].

3. Label distribution encoding. This method is used first time [6], and then it was successful in [5] competition. The logic behind label distribution encoding is very similar to binary value encoding. There is again certain number of classes. In our case it is 101. In this time, the vector values are not binary values, but the real values representing the probability distributions of the belonging to corresponding classes. It can be explained like that there is no differences between the day person becomes 26 years old and the day before.

Let's assume we have N classes of age and we encode the age $y \in \mathbb{R}$ with a label vector as multidimensional I_j, with j_t dimension in range of zero to N:

$$I_j = \frac{1}{\sigma\sqrt{2\pi}} e^{\frac{(j-y)^2}{2\sigma^2}} \tag{4.14}$$

Normal distribution of age label is shown in 4.14, where σ value is constant and predefined parameter. As it is noticeable, for gender classification, all 3 methods are giving the same result. First method is regression, but the second and the third method actually are the same. Third method is generalized version of the second method. So we can say when $\sigma \rightarrow 0$, it is binary value encoding. Figures 4.11, 4.12, 4.13 show the age distribution for different ages and different σ values.

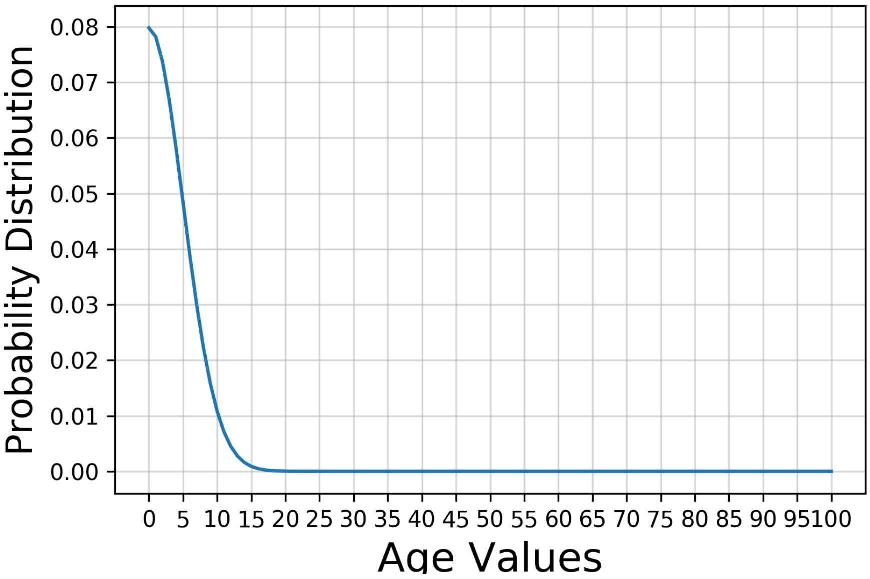


Figure 4.11: Real Age Value is 0 and standard deviation value is 5.

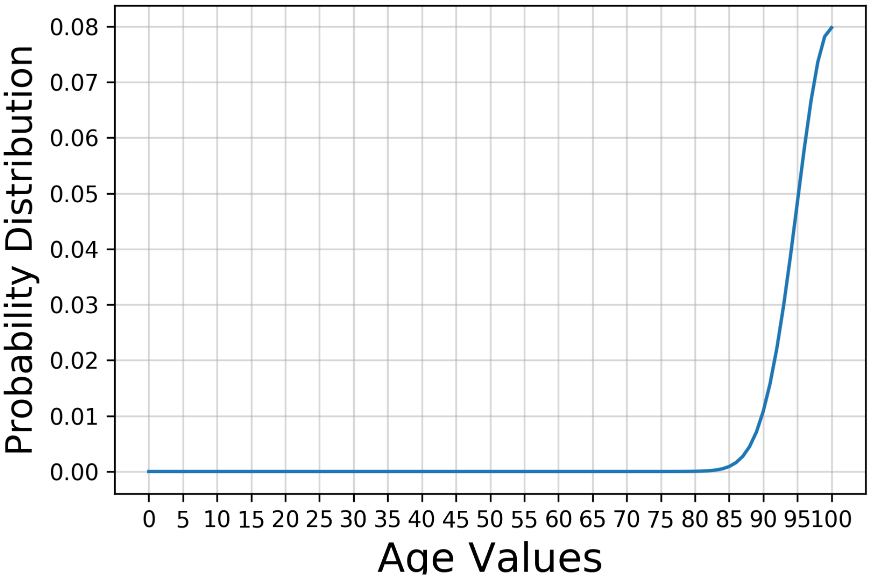


Figure 4.12: Real Age Value is 100 and standard deviation value is 5.

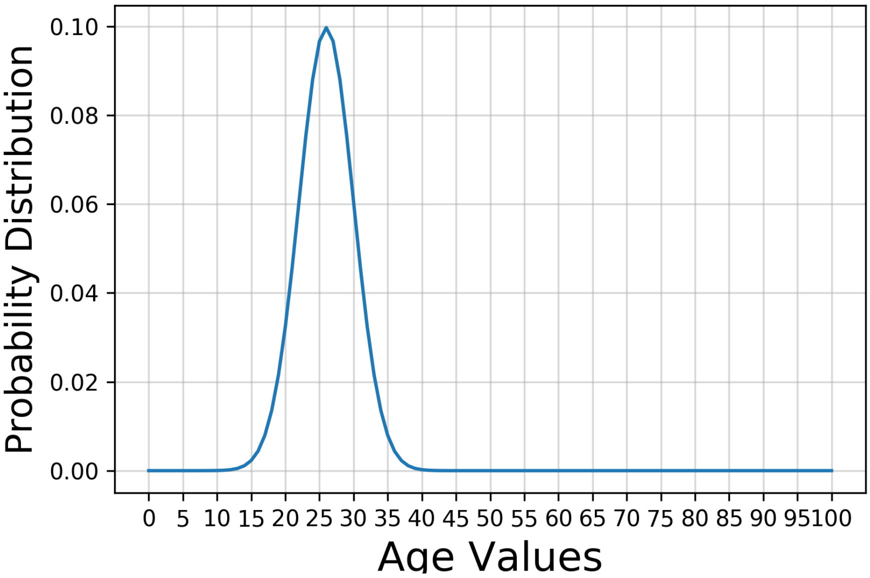


Figure 4.13: Real Age Value is 26 and standard deviation value is 4.

4.3.3 The Deep Learning

The deep learning is one of the state of the art machine learning method using the algorithm called artificial neural networks which is fully inspired by human neural system. Before starting to explain deep learning, artificial neural network is going to be explained.

Human brains consist of hundreds of billions of interconnected neurons that help us to recognize patterns, learn new languages, understand the environments, etc. The scientists thought that it is possible to implement similar system with help of silicon. Many silicon valley companies designed artificial neural networks and have a successful results.

An artificial neural network consists of one input layer, one or two hidden layer(s), and one output layer which are connected to each other. Figure 4.14 illustrates example of one hidden layer neural network.

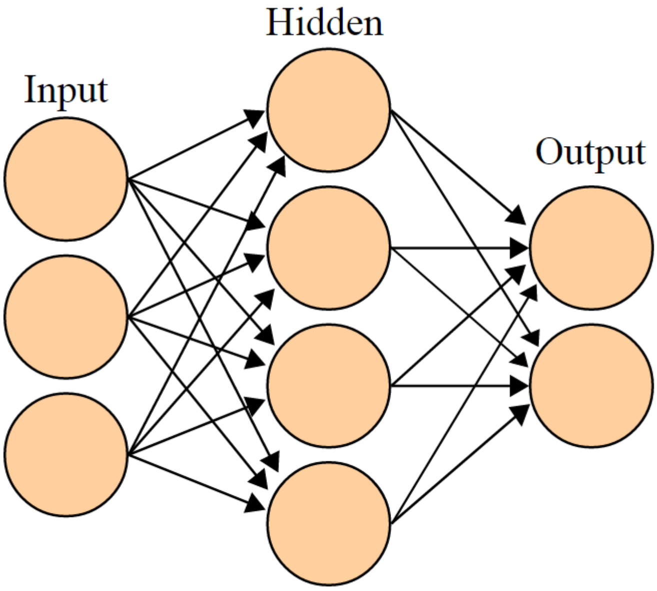


Figure 4.14: Simple neural network with one hidden layer

Each connection has their numerical value which is called weight. The output, h_i, of a neuron i in the hidden layer is,

$$h_i = \left(\sum_{j=1}^{N} W_j X j \right) \tag{4.15}$$

X_j is the input value of a neuron j in the input layer. W_j is the weight value of the connection between a neuron i in the hidden layer and a neuron j in the input layer. This is very basic example of a neural network. More complicated neural network can have bias value and activation function.

Artificial neural network is easy to implement by software as well as easy to compute. In many case, the developer, should adjust number of neurons inside the hidden layer in order to achieve best result.

Nowadays CPUs and GPUs are getting much more powerful. The researchers, in order to benefit of these improvements, try to increase the number of hidden layers inside the neural networks. Effect of these improvements helped deep learning to become famous. Deep learning may simply be considered as an artificial neural network which has many hidden layers. Figure 4.15 illustrates deep learning network.

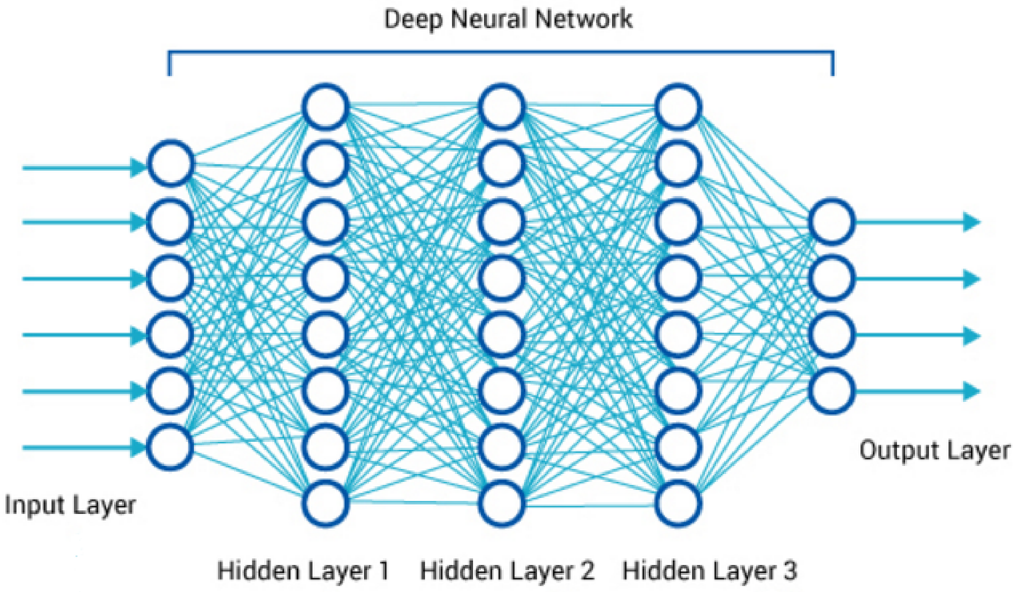


Figure 4.15: Deep learning network

4.3.3.1 Convolutional Neural Network

In this study, Convolutional neural networks (CNNs) are applied. CNNs are one of the deep learning methods that are mostly conducted on image processing. The difference of CNNs and simple deep learning network is that CNNs use convolution operators which can be understood by its name. So then the question is "what is the convolution operation and what does it do?".

Convolution is very important topic in image processing, it is also knows as filter in some topics. Convolution is mostly used for smoothing, sharpening and enhancing the image.

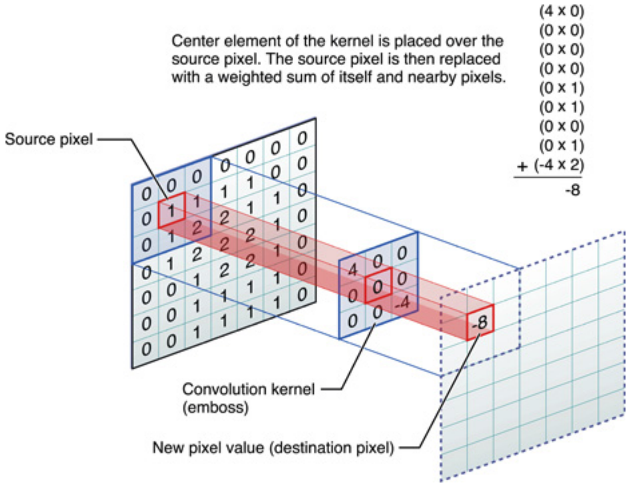

Center element of the kernel is placed over the source pixel. The source pixel is then replaced with a weighted sum of itself and nearby pixels.
(4 x 0)
(0 x 0)
(0 x 0)
(0 x 0)
(0 x 1)
(0 x 1)
(0 x 0)
(0 x 1)
+ (-4 x 2)
-8
Source pixel
Convolution kernel (emboss)
New pixel value (destination pixel)

Figure 4.16: Convolution operation on single pixel [31]

Convolution operation example is shown in Figure 4.16. Basically it finds the relationship between neighbourhood pixels. There are many predefined convolution kernels that does different improvements as mentioned earlier.

The aim of CNNs is not to use predefined kernels but to learn data specific kernels. Aid of this approach, the weight vectors are learnt by neurons spatially close to each other.

4.3.3.2 VGG-16

In ImageNet Challenge 2014, VGG-16 and VGG-19 are introduced. VGG-16 is thorough evolution of networks of increasing depth using an architecture with very small (3 x 3) convolution kernel [30]. This proved that significant improvement can be achieved by pushing the depth 16-19 weight layers [30]. This methods are ranked as first and second places in the localisation and classification respectively.

After the competition, they improved further their model. Table 4.1 shows the top-5 classification error on ILSVRC-2012 [27] results with the improved model.

Table 4.1: VGG ILSVRC-2012 Results [27]

Model	Validation Set	Test Set
16-layer	7.5%	7.4%

19-layer	7.5%	7.3%
model fusion	7.1%	7.0%

Even though authors published VGG-16 and VGG-19 neural network architecture on their webpage, model fusion is not published. The table shows that there is only small difference between 16 layers and 19 layers. The difference can be disregarded in order to decrease computational complexity.

4.3.3.3 Caffe Deep Learning Framework

In the previous section, training of the system is not explained. It is very complicated mathematical operations. As it would be very time-consuming task to re-implement all the functionality while already advanced ones exist.

In this study, we chose Caffe that is a deep learning framework. It is developed by the Berkeley Vision and Learning Center (BVLC) and by community contributors. Yangqing Jia created the project during his PhD at UC Berkeley and released it under the BSD 2-Clause license [12]. Caffe has very modular design which enables user to create and improve models and optimize them without hard-coding. The user can easily switch between GPU and CPU by setting single flag [12].

The framework has been forked by over 1000 developers since the first commit. That's why it has been updated to track state of the art in both code and models [12]. Another but the most important reason to use caffe would be its speed. Caffe can process 60M images per day with a single NVIDIA K40 GPU [12].

Caffe is developed under Linux operation system with C++ programming language. Thus it does not support natively Windows. The user, who would like to use Caffe on Windows machine, has two options. One of them would be the change the unix-like libraries to windows based inside the source code of Caffe. Other option would be the use Cygwin tool chain on windows machine. Both methods were applied in our study (Note that some volunteers produced Caffe for windows, besides they publish it in official website after this study already produced the similar one).

4.3.3.4 Scripts

Although Caffe framework is written entirely in C++, it also supports Python. Many deep learning framework uses python because it is interpret programming language rather than compiled one. By this, it is meant that there is no time consumption for compilation after every single change. People can think that Python is interpret programming language, so it is much more slower than the compiled one. This is very correct assumption unless one thing that python is able to use C++ libraries which means in low level, it is as fast as compiled code. So in our case, training is written in python that enables us to change and observe very rapid. Caffe deep learning framework also provides iPython example projects. Thus we used iPython and Python Notebook together.

4.3.4 Training/Validation

VGG-16 is very huge neural network. That's why it need high performance and high memory graphic card. Minimum memory of GPU card shall be 4GB. Even though 4GB would not train the system in recommended batch size. Thus batch size must be reduced to 10. Batch size plays very important role in training. Table 4.2 and 4.3 show configuration values that was used by training.

Gender Training/Validation: Table 2.1 illustrates that the amount of female images were outnumbered. Imbalanced classes are likely to give faulty results. In order to prevent the faulty results, $\%50$ of the female images were randomly removed from gender training. Randomly chosen $\%5$ of images from both classes were used for testing. $\%10$ of the images from each classes also were chosen randomly for validation, and the rest of the images were deployed by training. Table 4.2 shows the configuration of caffe framework for gender training.

Table 4.2: Table to gender training solver settings

Gender Training Solver Settings	
Feature	Value
Batch Size	10
Test Iteration	1000
Test Interval	10000

Base LR	0.003
LR Policy	”step”
Gamma	0.1
Step Size	10000
Display	200
Maximum Iteration	40000
Momentum	0.9
Weight Decay	0.0005

Age Training/Validation: Firstly at is already mentioned earlier, we pre-trained our network with Wiki-IMDB [4] dataset. Afterwards we splitted the Japanese Dataset as %70 training, %20 validation and %10 testing. In machine learning, validation is very important.

Table 4.3: Table to age training caffe solver settings

Age Training Solver Settings	
Feature	Value
Batch Size	16
Test Iteration	1000
Test Interval	20000
Base LR	0.003
LR Policy	”step”
Gamma	0.1
Step Size	25000
Display	200
Maximum Iteration	50000
Momentum	0.9
Weight Decay	0.0005

5 Tests

In machine learning, the data and its labels are playing very important roles. When the Japanese Age Dataset was collected, we had to label them manually. Because of hard-manual-work, labelling might give faulty results. In consequence, this would effect the training performance.

To sum up, we need to use good testing method in order to reduce the faulty labelling. Therefore we applied threshold value for detecting errors. When the error is lower than the threshold value, we omitted it as well as when the error is higher than threshold, we reduce the error by threshold value. In this study, the results are evaluated by using the standard mean absolute error (MAE) 5.1 measure with aid of threshold control mechanism.

$$MAE = \left(\frac{\sum_{i=1}^{N} |y_i - x_i|}{n} \right) \tag{5.1}$$

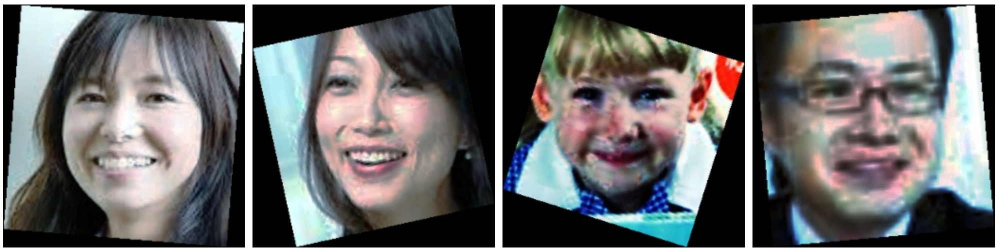
i represents the test number value in a range of all amount of test data. y_i and x_i indicate apparent age and estimated age value respectively. When the value of $|y_i - x_i|$ is lower than threshold, the value is taken as 0 in order to reduce the faulty results due to mislabelling issues.

The method, that described, is employed by age estimation testing, however test gender classification used basic measurement method due to its binary nature, ratio of correctly classified data and all tested data gave the results.

6 Result

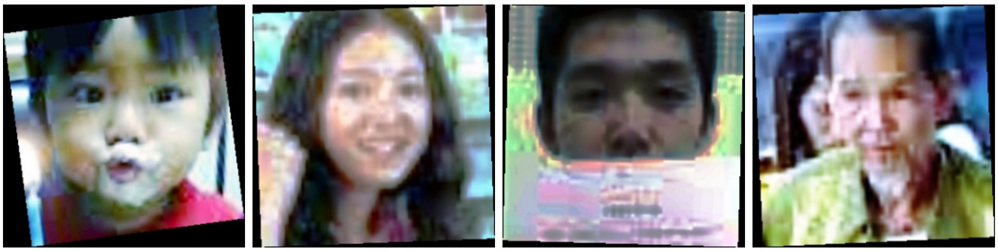
6.1 Gender Classification

Convolutional neural network give excellent results for gender classification. As it was already mentioned earlier, we split %5 of our data for testing. Test results show that it is very likely to reach %97 accuracy easily. Figure 6.1 and 6.2 illustrate, respectively, the correctly classified and misclassified photos.



(a) Female photo (b) Female photo (c) Male photo (d) Male photo

Figure 6.1: Correctly classified photos [8]



(a) Female photo (b) Female photo (c) Male photo (d) Male photo

Figure 6.2: Misclassified photos [8]

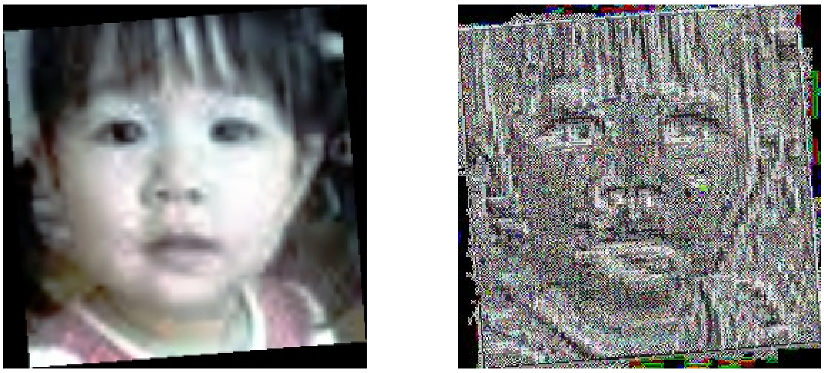
Main reason for the misclassified images are mostly because of poor resolution. It was also discovered that some data were misclassified in training as well due to the automated script which was collected images from google and automatically labelling their gender by using search keywords. In order to reach higher accuracy, resolution of the data must be increased and mislabelling problem must be solved. Table 6.1 shows the amount of correctly classified photos. As it can be seen that the accuracy is over %97.

Table 6.1: Gender classification results

	Amount of Images	Correctly Classified
Male	3112	2987
Female	2076	2047
Total	5188	5034

6.2 Age Estimation

We have trained our system with two type of images such as aligned normal image and its LPQ corresponding. Figure 6.3 illustrates the input data for estimation as well as training. LPQ corresponding is used for reducing the error by applying weighted voting. Additionally standalone LPQ produced very poor results.



(a) Normal Photo

(b) LPQ Corresponding

Figure 6.3: Input data for training and estimation [8]

Table 6.2 illustrates the results achieved by AGE system. All results were compared to Human Observer. Our AGE system, significantly outperformed the human observer.

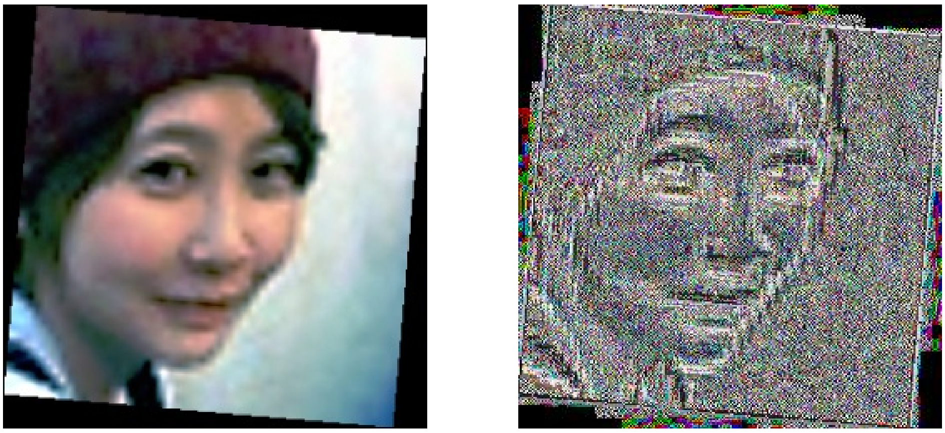
Table 6.2: Age estimation results

	Error Value - Normalized MAE [5.1] Results
Standalone VGG-16	0.78
Standalone LPQ	0.95

Weighted Voting	0.62
Human Observer	0.73

Figure 6.4, 6.5 and 6.6 show the estimation and their correct apparent age labels.

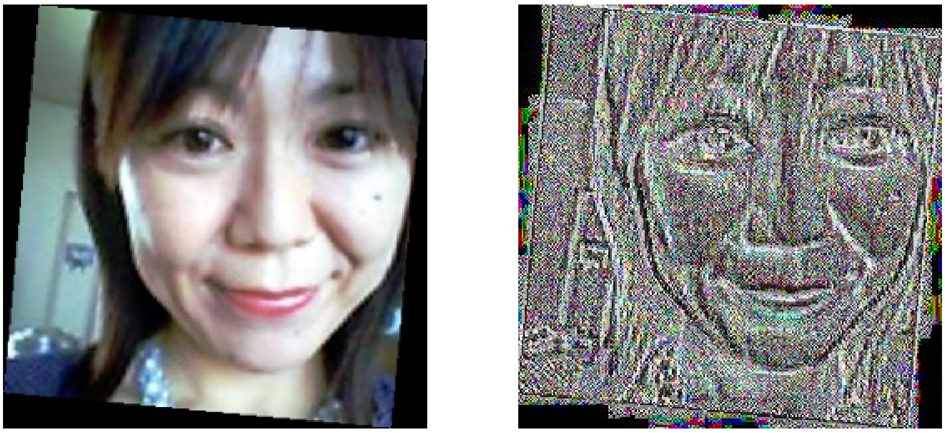
Results show that LPQ tend to give younger age values. In conclusion, our weighted voting system aids to achieve better accuracy.



(a) Apparent 30/ Estimated 33

(b) Apparent 30/ Estimated 20

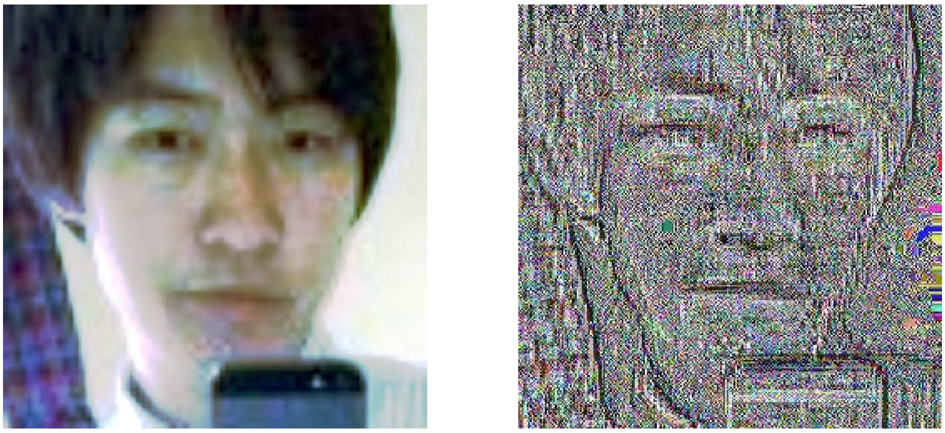
Figure 6.4: Weighted age estimation 30.5 years old



(a) Apparent 35/ Estimated 18

(b) Apparent 35/ Estimated 16

Figure 6.5: Weighted age estimation 17.6 years old



(a) Apparent 27/ Estimated 23

(b) Apparent 27/ Estimated 20

Figure 6.6: Weighted age estimation 22.4 years old

7 Summary

In this study, it was proposed that age and gender estimation based on local phase quantization and convolutional neural network. They were applied to Japanese dataset that was collected by iCV [10] and manually labelled. There were two main problem such as mislabelling for age and imbalance data for gender classification. The problem with gender classification was solved by removing the extra images from training. Because nature of mislabelling by human, we used MAE with threshold algorithm in testing (Section 5). Results show that the better accuracy is high likely to be achieved with high quality photos.

8 Acknowledgement

This thesis has been partially supported by Estonian Research Council Grant PUT638 and Super Apply Inc., *A.G.E. Project*, through the Research and Development VLTTI16119. The author also gratefully acknowledges the support of NVIDIA Corporation with the donation of the Titan X Pascal GPU.

References

[1] G. Anbarjafari, “Face recognition using color local binary pattern from mutually independent color channels”, *EURASIP Journal on Image and Video Processing*, vol. 2013, no. 1, p. 6, 2013.

[2] G. Antipov, M. Baccouche, S.-A. Berrani, and J.-L. Dugelay, “Apparent age estimation from face images combining general and children-specialized deep learning models”, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2016, pp. 96–104.

[3] A. Deniz, H. E. Kiziloz, T. Dokeroglu, and A. Cosar, “Robust multiobjective evolutionary feature subset selection algorithm for binary classification using machine learning techniques”, *Neurocomputing*, vol. 241, pp. 128–146, 2017.

[4] S. Escalera, J. Fabian, P. Pardo, X. Baró, J. Gonzalez, H. J. Escalante, D. Misevic, U. Steiner, and I. Guyon, “Chalearn looking at people 2015: Apparent age and cultural event recognition datasets and results”, in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2015, pp. 1–9.

[5] S. Escalera, M. Torres Torres, B. Martinez, X. Baró, H. Jair Escalante, I. Guyon, G. Tzimiropoulos, C. Corneou, M. Oliu, M. Ali Bagheri, *et al.*, “Chalearn looking at people and faces of the world: Face analysis workshop and challenge 2016”, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition Workshops*, 2016, pp. 1–8.

[6] X. Geng, C. Yin, and Z.-H. Zhou, “Facial age estimation by learning from label distributions”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 10, pp. 2401–2412, 2013.

[7] X. Geng, Z.-H. Zhou, and K. Smith-Miles, “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] *Google*, Accessed: 2016. [Online]. Available: `http://www.google.com/`.

[9] B. Gudla, S. R. Chalamala, and S. K. Jami, “Local binary patterns for gender classification”, in *Artificial Intelligence, Modelling and Simulation (AIMS), 2015 3rd International Conference on*, IEEE, 2015, pp. 19–22.

[10] *iCV Research Group at University of Tartu*, Accessed: 19-04-2017. [Online]. Available: `http://icv.tuit.ut.ee/`.

[11] R. Jana, D. Datta, and R. Saha, “Age estimation from face image using wrinkle features”, *Procedia Computer Science*, vol. 46, pp. 1754–1761, 2015.

[12] Y. Jia, E. Shelhamer, J. Donahue, S. Karayev, J. Long, R. Girshick, S. Guadarrama, and T. Darrell, “Caffe: Convolutional architecture for fast feature embedding”, *arXiv preprint arXiv:1408.5093*, 2014.

[13] V. Kazemi and J. Sullivan, “One millisecond face alignment with an ensemble of regression trees”, in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2014, pp. 1867–1874.

[14] D. E. King, “Dlib-ml: A machine learning toolkit”, *Journal of Machine Learning Research*, vol. 10, no. Jul, pp. 1755–1758, 2009.

[15] D. Krefting, C. Jansen, T. Penzel, F. Han, and J. W. Kantelhardt, “Age and gender dependency of physiological networks in sleep”, *Physiological Measurement*, vol. 38, no. 5, p. 959, 2017.

[16] V. Le, J. Brandt, Z. Lin, L. Bourdev, and T. Huang, “Interactive facial feature localization”, *Computer Vision–ECCV 2012*, pp. 679–692, 2012.

[17] J. Li, S. Li, J. Hu, and W. Deng, “Adaptive lpq: An efficient descriptor for blurred face recognition”, in *Automatic Face and Gesture Recognition (FG), 2015 11th IEEE International Conference and Workshops on*, IEEE, vol. 1, 2015, pp. 1–6.

[18] S. S. Liew, M. K. HANI, S. A. RADZI, and R. Bakhteri, “Gender classification: A convolutional neural network approach”, *Turkish Journal of Electrical Engineering & Computer Sciences*, vol. 24, no. 3, pp. 1248–1264, 2016.

[19] 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.

[20] R. C. Malli, M. Aygün, and H. K. Ekenel, “Apparent age estimation using ensemble of deep learning models”, in *Computer Vision and Pattern Recognition Workshops (CVPRW), 2016 IEEE Conference on*, IEEE, 2016, pp. 714–721.

[21] B. Moghaddam and M.-H. Yang, “Gender classification with support vector machines”, in *Automatic Face and Gesture Recognition, 2000. Proceedings. Fourth IEEE International Conference on*, IEEE, 2000, pp. 306–311.

[22] M. Nazir, M. Ishtiaq, A. Batool, M. A. Jaffar, and A. M. Mirza, “Feature selection for efficient gender classification”, in *Proceedings of the 11th WSEAS International Conference*, 2010, pp. 70–75.

[23] V. Ojansivu and J. Heikkilä, “Blur insensitive texture classification using local phase quantization”, in *International conference on image and signal processing*, Springer, 2008, pp. 236–243.

[24] V. Ojansivu, E. Rahtu, and J. Heikkila, “Rotation invariant local phase quantization for blur insensitive texture analysis”, in *Pattern Recognition, 2008. ICPR 2008. 19th International Conference on*, IEEE, 2008, pp. 1–4.

[25] P. Rasti, M. Daneshmand, and G. Anbarjafari, “Statistical approach based iris recognition using local binary pattern”, *Dyna*, vol. 92, no. 1, pp. 76–81, 2017.

[26] R. Rothe, R. Timofte, and L. Van Gool, “Dex: Deep expectation of apparent age from a single image”, in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2015, pp. 10–15.

[27] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, A. C. Berg, and L. Fei-Fei, “ImageNet Large Scale Visual Recognition Challenge”, *International Journal of Computer Vision (IJCV)*, vol. 115, no. 3, pp. 211–252, 2015. DOI: `10.1007/s11263-015-0816-y`.

[28] O. Russakovsky, J. Deng, H. Su, J. Krause, S. Satheesh, S. Ma, Z. Huang, A. Karpathy, A. Khosla, M. Bernstein, *et al.*, “Imagenet large scale visual recognition challenge”, *International Journal of Computer Vision*, vol. 115, no. 3, pp. 211–252, 2015.

[29] C. Sagonas, E. Antonakos, G. Tzimiropoulos, S. Zafeiriou, and M. Pantic, "300 faces in-the-wild challenge: Database and results", *Image and Vision Computing*, vol. 47, pp. 3–18, 2016.

[30] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition", *arXiv preprint arXiv:1409.1556*, 2014.

[31] *Single bit convolution example*, Accessed: 19-04-2017. [Online]. Available: `https://developer.apple.com/library/content/documentation/Performance/Conceptual/vImage/ConvolutionOperations/ConvolutionOperations.html`.

[32] B. Wu, H. Ai, and C. Huang, "Real-time gender classification", in *Third International Symposium on Multispectral Image Processing and Pattern Recognition*, International Society for Optics and Photonics, 2003, pp. 498–503.

[33] Z. Yang, M. Li, and H. Ai, "An experimental study on automatic face gender classification", in *Pattern Recognition, 2006. ICPR 2006. 18th International Conference on*, IEEE, vol. 3, 2006, pp. 1099–1102.

[34] Y. Zhu, Y. Li, G. Mu, and G. Guo, "A study on apparent age estimation", in *Proceedings of the IEEE International Conference on Computer Vision Workshops*, 2015, pp. 25–31.

Appendixes

Appendix 1: Table of Amount of Images

Table 8.1: Amount of images in each age category

Amount of Images		
Age	Male Count	Female Count
0	1036	205
1	750	313
2	437	221
3	676	465
4	290	143
5	33	430
6	40	222
7	85	498
8	51	319
9	39	284
10	110	100
11	110	129
12	121	103
13	65	51
14	38	90
15	110	168
16	182	354
17	308	1002

18	634	935
19	947	2073
20	5955	6176
21	8457	9198
22	5505	9376
23	4061	10889
24	3530	10131
25	2775	10177
26	3399	9572
27	3626	9395
28	4212	8118
29	3725	7956
30	3638	6048
31	2288	6154
32	2460	6123
33	4133	6841
34	2602	6289
35	1697	7018
36	1663	6840
37	2081	6528
38	2311	6290
39	2029	6286
40	1446	3823
41	1682	3686
42	1783	3736
43	1126	4030
44	961	3527
45	1003	3533
46	943	3081
47	999	2699
48	1219	2202

49	974	1990
50	376	1142
51	460	1161
52	600	1061
53	760	942
54	639	702
55	627	736
56	506	627
57	508	539
58	525	389
59	431	331
60	252	153
61	233	120
62	261	148
63	310	138
64	264	127
65	239	122
66	109	92
67	82	68
68	80	68
69	71	57
70	130	21
71	115	66
72	105	21
73	103	21
74	25	22
75	26	24
76	18	15
77	14	19
78	12	14
79	13	14

80	97	13
81	113	8
82	103	7
83	61	15
84	12	10
85	16	16
86	11	6
87	5	10
88	3	3
89	3	2
90	93	2
91	107	6
92	93	9
93	3	6
94	4	9
95	5	7
96	2	7
97	1	1
98	7	1
99	5	1
100	1	0

II. Licence

Non-Exclusive licence to reproduce thesis and make thesis public

I, **Ozan Bilici**,

1. herewith grant the University of Tartu a free permit (non-exclusive licence) to:

   1.1 reproduce, for the purpose of preservation and making available to the public, including for addition to the DSpace digital archives after 01.01.2019 until the expiry of the term of validity of the copyright, and

   1.2 make available to the public via the web environment of the University of Tartu, including via the DSpace digital archives after 01.01.2019 until expiry of the term of validity of the copyright,

   of my thesis

   Local Phase Quantization Feature Extraction based Age and Gender Estimation Using Convolutional Neural Network

   supervised by Dr. Gholamreza Anbarjafari

2. I am aware of the fact that the author retains these rights.

3. I certify that granting the non-exclusive licence does not infringe the intellectual property rights or rights arising from the Personal Data Protection Act.

Tartu, 16.05.2017