Experimental Analysis of Face Recognition from Infrared Camera using Retinex Algorithm and Linear Discriminant Analysis (LDA)

Mohamad Nurfakhrian Aziz, Anggunmeka Luhur Prasasti and Tito Waluyo Purboyo
Department of Computer Engineering, Faculty of Electrical Engineering,
Telkom University, Bandung, Indonesia

Abstract: Face recognition using infrared technology is increasingly being used for the security purpose, it is Closed Circuit Television (CCTV) camera. CCTV camera is generally using infrared technology to monitor the surrounding environment in every condition whether it's clear or even dark one without any lighting. Any evidence of crime captured by CCTV camera will be used for legal purposes. As we know that the image from infrared camera has poor quality, especially in terms of illumination and image detail, so, it's hard to recognize someone from infrared camera. Therefore, the image quality from infrared camera needs to be enhanced. Retinex algorithm works like retina (in the eyes) and cortex (in the brain) to adapt the illumination. SSR (Single-Scale Retinex) and MSR (Multi-Scale Retinex) methods has been used in this research to be analyzed their effectiveness in face recognition using LDA (Linear Discriminant Analysis) method. In the tests that have been done, people face recognition in very dark conditions can still be recognized compared to face recognition without retinex and MSR has better accuracy than SSR one.

Key words: Infrared camera, face recognition, image quality enhancement, linear discriminant analysis, retinex, MSR

INTRODUCTION

An image is a representation of various information contained therein such as pixel value, color and so on. Along with the development of technology, the image is increasingly applied in various fields. Digital image processing is needed to adjust human perception. Basically, image processing is categorized into two scopes. First, digital image processing in terms of image quality improvement in the characteristics of detail, visual effects, lighting and so on, so that, the quality of digital images better than the previous conditions. Second, digital image processing for image data processing, e.g., data storage, transmission/communication and to assist the interaction between man and machine (Li *et al.*, 2015; Madenda, 2015).

The use of a digital image is very influential in our life because we can find many informations that can be used for diagnostic assesment or even as evidence of the crime. For example, on the use of CCTV cameras. In general, this CCTV camera used infrared technology, so, it can observe environment in the dark. With this CCTV camera, we can find out all the crimes that operate during the night. The evidence used is usually the face of criminals who will be recognized using face recognition method. In order for the criminal's face to be recognized accurately, appropriate face recognition methods are required.

Based on previous research, the quality of infrared images tends to have bad contrast and poor detail. For example, research on the use of infrared cameras in the biomedical field, to trace blood vessels in obese people (Prasasti *et al.*, 2015). The result of the infrared image in this research still needs to be improved in its quality. Many methods of image quality improvement are used at this time. For example, the improvement of infrared image quality using the retinex method. Therefore, improving the quality of the infrared image is needed. In this research, the improvement of infrared image quality using retinex SSR and MSR method and face recognition method used is Linear Discriminant Analysis (LDA). Tests conducted on this research to compare Linear Discriminant Analysis (LDA) face recognition results using retinex and without using retinex.

Basic theory: The human face is a gift given by God with the face, we can recognize each other. The human face basically has important parts of the eyes, nose and mouth. Each person must have different eye,

Corresponding Author: Mohamad Nurfakhrian Aziz, Department of Computer Engineering, Faculty of Electrical Engineering,
Telkom University, Bandung, Indonesia

nose and mouth shape, although, there is a similarity between one person and another is a natural thing. The human face in digital images is a 2-Dimensional object (2D).

Face recognition is one part of the many types of digital image processing. Facial recognition is used to identify a person's face by paying attention to parts of the face such as the eyes, the nose, the mouth or the characteristic of a person's face. Not all faces can be recognized, for example, if the face image has poor quality such as less qualified resolution, less detail or more noise on the image. If you want to produce a good facial recognition required additional methods such as digital image enhancement or other methods.

Digital image: Visually, the image is a representation of in-formation, so that, humans can analyze and interpret the information. Information from images is divided into two parts: basic information and abstract information. Basic information is information that we can handle directly without requiring any special knowledge. For example, the image must have color, feature, shape, contrast or light level. As for the abstract information is information that required special knowledge. For example, in the picture, someone has a different facial expression which describes the various feelings of the person (feeling sad, happy, angry and so on) (Madenda, 2015).

Mathematically, the image can be defined as a two-dimensional function of $f(x, y)$ where x and y are spatial coordinates and f is the color intensity value of x and y coordinates. The color intensity in $f(x, y)$ is encoded in 24 bits for color images (three RGB color components: R = Red, G = Green and B = Blue), 8 bits for Gray-level image and 1 bit for the binary image. The value of $f(x, y)$ is finite. Digital images are generally two dimensions expressed in the matrix. The matrix is a representation of the pixels of the N×M matrix where the range of values is 1 bit = 2 color [0 and 1], gray-level 8 bit = 256 color [0-255] and RGB 24 bit = (R: [0-255], G: [0-255], B: [0-255]) (Madenda, 2015).

Infrared image: The human eye can see visible light or a fraction of the electromagnetic spectrum but does not see ultraviolet light and infrared light. Infrared radiation can be used to determine the temperature of an object. To be able to detect infrared radiation used the thermographic camera. With this camera infrared radiation can be detected in the range of electromagnetic spectrum about 0.9-14 μm and produce radiation that can be viewed by humans (Battalwar *et al.*, 2015).

The infrared radiation source is thermal. Because basically, any object that has a temperature above absolute zero (0 K) will emit radiation in the infrared region. According to black body's legal radiation, thermography makes it possible to see the environment

with or without real illumination. The higher body temperature has more infrared radiation which causes the object to emit radiation energy of a visible spectrum. Infrared cameras can work in total darkness because the ambient light levels are not being noticed. This camera can process infrared radiation with an algorithm to interpret the data, so that, the required information is displayed into the infrared image. This infrared image is generally represented in gray color which has 8 bits [0-255] (Battalwar *et al.*, 2015).

Image enhancement: Image enhancement is generally required to improve image quality to fit the human perception. Improved images usually have less detail, less light or a lot of noise on the image causing inconvenience to the human eye (Janani and Dinakaran, 2014; Shu-Xin *et al.*, 2013; Sun *et al.*, 2017). The retinex method is the first method that mimics the workings of the human eye such as the retina and cortex of the human eye.

Retinex theory: At night or in non-bright conditions, digital images have poor contrast and high noise ratios which causes the image to be unclear, especially, on image details. This causes problems for people's lives, especially in areas that use video surveil-lance applications. Based on the problem, it is proposed that image enhancement technology for images with low light intensity. Edwin Land first proposed the Retinex theory in 1977 with the basic idea that the perception of light perceived by humans depends on values in the surrounding region rather than the absolute value of the point of light (Yin *et al.*, 2014).

Referring to the problem of infrared images that always have low contrast, low dynamic range, weak detail information and low visual effects, this study uses Retinex theory for image improvement methods (Shu-Xin *et al.*, 2013). In recent years, retinex has been applied more and more widely. And some relevant algorithms are proposed gradually. Jobson *et al.* Dinata (2014), Liu *et al.* (2016) and Borade *et al.* (2016) proposed Retinex (SSR) Single-Scale, Multi-Scale Retinex (MSR) and Multi-Scale Retinex with Color Restoration (MSRCR) (Yin *et al.*, 2014). Here are the variants of retinex theory.

Single-Scale Retinex (SSR): Single-Scale Retinex (SSR) relies on the ratio of ambient light to average light that can provide dynamic range compression of images. This SSR has a sigma value that will affect the Gaussian value. The greater the Gaussian value the higher the color value. The sigma value used is [0-255]. Here is the equation of SSR (Li *et al.*, 2007):

$$R_i(x, y) = \log I_i(x, y) - \log[F(x, y) * I_i(x, y)] \qquad (1)$$

where, $R_i(x, y)$ is the retinex output, the subscript $i\epsilon R$, G, B represents the three colors, $I_i(x, y)$ is the image distribution in the ith color band, the sign*denotes the convolution operation and $F(x, y)$ is a Gaussian function. This algorithm uses a varied Gaussian function. Since, the Gaussian function is regional and provides a good dynamic range compression on a number of constant spaces, the Gaussian function is used as follows (Li *et al.*, 2007):

$$F(x, y) = e^{-(x^2+y^2)/\sigma^2} \tag{2}$$

where, σ is the Gaussian constant which is generally used to represent the standard deviation. The Gaussian constant σ is referenced as a scale to the SSR. The small value of σ provides excellent dynamic range compression but has a color value that is not very good (Li *et al.*, 2007).

SSR calculation example: First, determine the value of Gaussian filter. This value useful for determining SSR value (Dinata, 2014):

$$F(x,y) = e^{-(x^2+y^2)/s^2} \tag{3}$$

Where:
σ = Gaussian constant (sigma)
e = Exponential
(x, y) = Pixel coordinates

Example: $\sigma_1 = 10$, $\sigma_2 = 15$, $\sigma_3 = 20$

Example original image matrix:

125	250	100
89	70	125
255	175	80

$$F(x,y) = \begin{pmatrix} e^{-\frac{(-1)^2+(1)^2}{(10)^2}} & e^{-\frac{(0)^2+(1)^2}{(10)^2}} & e^{-\frac{(1)^2+(1)^2}{(10)^2}} \\ e^{-\frac{(-1)^2+(0)^2}{(10)^2}} & e^{-\frac{(0)^2+(0)^2}{(10)^2}} & e^{-\frac{(0)^2+(1)^2}{(10)^2}} \\ e^{-\frac{(-1)^2+(-1)^2}{(10)^2}} & e^{-\frac{(0)^2+(-1)^2}{(10)^2}} & e^{-\frac{(-1)^2+(1)^2}{(10)^2}} \end{pmatrix}$$

$$= \begin{pmatrix} 0.9801986733 & 0.9900498337 & 0.9801986733 \\ 0.9900498337 & 1 & 0.9900498337 \\ 0.9801986733 & 0.9900498337 & 0.9801986733 \end{pmatrix}$$

The total $\Sigma F(x, y)$ must be equal to 1, if not equal to 1, then it should be normalized by the following Eq. 4-6:

$$\sum F(x,y) = 8.880994027 \neq 1 \tag{4}$$

$$a(x,y) = a \times \frac{1}{\sum F(x,y)}$$

$$a(1,1) = \frac{0.9801986733}{8.880994027} = 0.1103703786$$

$$a(1,2) = \frac{0.9900498337}{8.880994027} = 0.1114796194$$

$$a(1,3) = \frac{0.9801986733}{8.880994027} = 0.1103703786$$

$$a(2,1) = \frac{0.9900498337}{8.880994027} = 0.1114796194$$

$$a(2,2) = \frac{1}{8.880994027} = 0.1126000082 \tag{5}$$

$$a(2,3) = \frac{0.9900498337}{8.880994027} = 0.1114796194$$

$$a(3,1) = \frac{0.9801986733}{8.880994027} = 0.1103703786$$

$$a(3,2) = \frac{0.9900498337}{8.880994027} = 0.1114796194$$

$$a(3,3) = \frac{0.9801986733}{8.880994027} = 0.1103703786$$

After all normalized insert into the matrix 3×3 (Gaussian matrix) then sum it all:

$$= \begin{pmatrix} 0.1103703786 & 0.1114796194 & 0.1103703786 \\ 0.1114796194 & 0.1126000082 & 0.1114796194 \\ 0.1103703786 & 0.1114796194 & 0.1103703786 \end{pmatrix} \tag{6}$$

$$\sum a(x,y) = 1$$

Because the number $\Sigma a(x, y)$ is already equal to 1, then the convolution process. Convolution of the original image matrix with a Gaussian matrix (Dinata, 2014):

0	0	0	0	0
0	125	250	100	0
0	89	70	125	0
0	255	175	80	0
0	0	0	0	0

Convolution with:

0.1103703786	0.1114796194	0.1103703786
0.1114796194	0.1126000082	0.1114796194
0.1103703786	0.1114796194	0.1103703786

Here's, partly the result of convolution with $\sigma_1 = 10$:

= (0×0.1103703786)+(0×0.1114796194)+(0×0.1103703786)+
(0×0.1114796194)+(125×0.1126000082)+(250×0.1114796194)+
(0×0.1103703786)+(89×0.1114796194)+(70×0.1103703786) =
59.5925185

Calculations are done by shifting each pixel, thus,
resulting the following convolution matrix (Dinata,
2014):

59.5925185	84.65575079	60.790784597
107.094640365	140.373037494	88.85231678
65.869548115	88.4735083124	50.177812978

SSR_1 results (Dinata, 2014):

$$R_1(x,y) = \log I_1(x,y) - \log[F(x,y)*I_1(x,y)]$$
$$R_1(1,1) = \log(125) - \log(59.5925185)$$
$$R_1(1,1) = 0.321718273$$

All completed to generate SSR1 $\sigma_1 = 10$ matrices:

0.321718273	0.470283543	0.2161622514
-0.08037773012	-0.3021856576	0.1482412567
0.5878554964	0.2962247996	0.2025782585

The same calculation is also done with $\sigma_2 = 15$ and $\sigma_3 = 20$.

SSR2 $\sigma_2 = 15$:

0.32276945	0.471205087	0.217096941
-0.080415011	-0.304018746	0.14814252
0.589417701	0.296911281	0.20343417

SSR3 $\sigma_3 = 20$:

0.323137649	0.47152711	0.217424514
-0.080427955	-0.304063617	0.148107635
0.589964451	0.297151415	0.203734095

Multi-Scale Retinex (MSR): In the Multi-Scale Retinex
(MSR) algorithm the input image uses the result of SSR
with different sigma values. The results of this different
sigma are then summed, so as to produce different color
values with SSR. Although, MSR uses a combination of
some SSR results, the detail on the image will not change,
since, retinex basically, only improves the image quality in
the lighting scope. In order for optimal retinex results then

need to add some other image enhancement method. The
retinex multi-scale equation can be defined as follows
(Li *et al.*, 2007):

$$R(x, y) = \sum_{k=1}^{N} W_k \{\log I(x, y) - \log[G_k(x, y)^* I(x, y)]\} \quad (7)$$

Where:
N = The scale value to be used
W_k = The Weights for the k-scale
$G_k(x, y)$ = The Gaussian function

where, $F_k(x, y) = K_k e^{-(x^2+y^2)/\sigma_k^2}$ where, K_k is an arithmetic
operator.

MSR calculation example: This MSR uses SSR results
that have been resolved by multiplying by the weight
value W_k. The weight value W_k used in this research is
(0.33, 0.34 and 0.33) where the value of this W_k should be
equal to 1 $\sum W_k = 1$, then divided into 3 values based on
the scale used (Liu *et al.*, 2016):

SSR1 $\sigma_1 = 10$:

0.321718273	0.470283543	0.216162251
-0.08037773	-0.302185658	0.148241257
0.587855496	0.2962248	0.202578259

SSR2 $\sigma_2 = 15$:

0.32276945	0.471205087	0.217096941
-0.080415011	-0.304018746	0.14814252
0.589417701	0.296911281	0.20343417

SSR3 $\sigma_3 = 20$:

0.323137649	0.47152711	0.217424514
-0.080427955	-0.304063617	0.148107635
0.589964451	0.297151415	0.203734095

$$R(x,y) = \sum_{k=1}^{N} W_k \{\log\log I(x,y) - \log[G_k(x,y)^* I(x,y)]\} \quad (8)$$

$$R(x,y) = W_1 \times SSR1 + W_2 \times SSR2 + W_3 \times SSR3 \quad (9)$$

0.322544067	0.471007245	0.216896593
-0.08040698	-0.303428634	0.148163591
0.589082601	0.296763986	0.203250694

Then normalized by the following Eq.8 and 9 (Dinata, 2014):

$$MSR_i = 255 \times \left| \frac{value-min}{max-min} \right| \qquad (10)$$

where, value is the current value in matrix $R(x, y)$, min is the smallest value in matrix $R(x, y)$, max is the largest value in matrix $R(x, y)$. Here's an example of the calculation:

$$MSR_1 = 255 \times \left| \frac{(0.322544067 - (-0.303428634))}{(0.589082601 - (-0.303428634))} \right|$$

$$MSR_1 = 178.84709177358425$$

Then the MSR result:

179	221	149
64	0	129
255	171	145

Viola-Jones: Pattern recognition on digital images is called computer vision. The main element in computer vision is the algorithm (or method) in detecting an object in a digital image. Detection of objects in the most influential digital image is the human face detection. In general, this face detection only detects whether the face is a human face or not. In contrast to face recognition that can recognize a person's face. Of course, face detection is usually paired with face recognition for more accurate results. The most widely used face detection method is the Viola-Jones method. Viola-Jones method is a combination of support vector machines, boosting algorithms and cascade classifier (Egorov *et al.*, 2018; Putro *et al.*, 2012).

The Viola-Jones method, proposed by Viola and Jones (2001) is based on integral image processing using a "weak" classification called Haar characteristic (Egorov *et al.*, 2018). Here are three basic methods used by the Viola-Jones algorithm (Nehru and Padmavathi, 2017).

The Haar-like feature is used for feature extraction to get an integral image. The machine-learning technique called AdaBoost is used for face detection. Because the process of classification is very complex, then made this basic improvement classification techniques. The last method is the cascade classification that can combine many features efficiently. In this case, the generated classifier has several filters.

How to calculate the pixel value in Fig. 1 is by adding up the pixels located in the white rectangle minus the number of pixels in the gray rectangle. The two rectangular features are shown in Fig. 1a-c
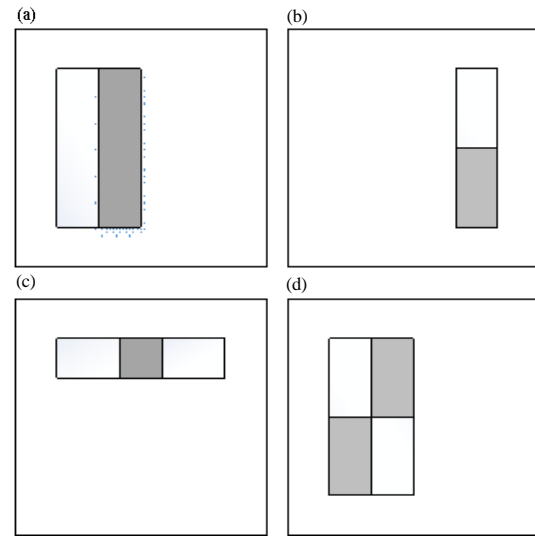

Fig. 1: a-d) Example features of Viola-Jones rectangle (Viola and Jones, 2001)

shows the three rectangular features and the image Fig. 1d features a rectangle (Nehru and Padmavathi, 2017).

Linear Discriminant Analysis (LDA): Face recognition is the most widely used biometric to recognize a person's face. This technology is used for some interests, for example, on the use of video surveillance that can keep an eye on someone for legal purposes as for face recognition is currently used as a password on smartphones, so, only those concerned can open a smartphone. This face recognition uses feature extraction as its main element. Among the most widely used feature extractions were Principal Component Analysis (PCA) and Linear Discriminant Analysis (LDA) (Borade *et al.*, 2016).

Fisher Linear Discriminant Analysis (FLD, a.k.a. Linear Discriminant Analysis or LDA) was introduced by Belhumeur in the field of artificial intelligence and pattern recognition (Liu and Wechsler, 2002). The basic idea of LDA is to project a high-dimensional sample to an optimal discriminant vector space to extract classification information and compressed feature dimensions by maximizing the between-class scatter value and minimizing within-class scatter (Wang and Chen, 2017). The classification of traits is performed by Principal Component Analysis (PCA) while data classification is performed by Linear Discriminant Analysis (LDA). Therefore LDA itself is a development of PCA which has a weak classification among the classes. The weakness of LDA requires large training samples (in sense of having to require extensive training data) for good generalization (Liu and Wechsler, 2002; Muntasa, 2015). Following face recognition steps with LDA (Borade *et al.*, 2016):

Take a picture M with size N×N (in 2D) then insert into training data. And also take another sample for test data. The total number of pixels of each image is N^2. There is a C number of people (explaining the number of samples to be tested/how many people will be tested). Change all 2D images in train data into 1D vector column matrix. Calculate the average image:

$$\mu = \frac{1}{M}\sum_{i=1}^{M} I_i \qquad (11)$$

Determine the standard deviation of each image from the average image:

$$\varphi = I_i - \mu \qquad (12)$$

Calculate the covariance matrix M×M:

$$L = A^T A \qquad (13)$$

L equals the formula for the general covariance matrix, i.e., (C = AA'). Then from L is searched Eigenvalue which Eigenvalue will be used to find Eigenvector.

Project the image vector centered on the Eigenspace to find the average Eigenspace grade and average Eigenspace. Then, calculate within-class Scatter (S_W) and between-class Scatter (S_B) with the following Eq. 14 and 15:

$$S_W = \sum_{j=1}^{C}\sum_{i=1}^{N_j}(I_i^j - \mu_j)(I_i^j - \mu_j)^T \qquad (14)$$

$$S_B = \sum_{j=1}^{C}(\mu_j - \mu)(\mu_j - \mu)^T \qquad (15)$$

Where:
I_i = The first sample of class j
N_j = The sample number of class j
C = The Class number
μ_j = The average of class j

The final result is to minimize the S_W value and increase the S_B value by increasing the ratio of det|S_B|/det|S_w|. This ratio can be enlarged on the eigenvector of the $S_W S_B$ column of the vector column of the matrix projection. To prevent the odd S_W value is applied dimensional reduction with (C-1).

Further measurement of similarities using euclidian distance in fact, there are still many methods for measuring similarities such as Chebyshev, Manhattan, Minkowski, Angular Separation and Hausdorff (Muntasa, 2015). But in this study using euclidian distance. Here, is the equation of euclidian distance (Muntasa, 2015):

$$d(x,y) = \sqrt{\sum_{i=1}^{m}\sum_{j=1}^{n}(x_{i,j} - y_{i,j})^2} \qquad (16)$$

Where:
m = The number of matrix columns
n = The number of rows of matrices
i and j = Classes and objects
x and y = The two objects to be measured for resemblance

The dimensions of these two objects must be of the same size because if the dimensions are not the same then the program will be an error.

LDA calculation example:

$$X_{X\in\omega 1} = \begin{pmatrix} 1 & 2 & 3 & 4 & 5 & 10 \\ 2 & 3 & 3 & 5 & 5 & 9 \end{pmatrix}, n_1 = 6$$

$$X_{X\in\omega 2} = \begin{pmatrix} 4 & 5 & 5 & 3 & 5 & 6 \\ 2 & 0 & 2 & 2 & 3 & 3 \end{pmatrix}, n_2 = n_2 = 6$$

First, calculate the average for class 1 and 2 (Muntasa, 2015).

For class 1:

$$\mu_1 = \frac{1}{n_1}\sum_{X\in\omega 2} X = \frac{1}{6}\left(\begin{pmatrix}1\\2\end{pmatrix}+\begin{pmatrix}2\\3\end{pmatrix}+\begin{pmatrix}3\\3\end{pmatrix}+\begin{pmatrix}4\\5\end{pmatrix}+\begin{pmatrix}5\\5\end{pmatrix}+\begin{pmatrix}10\\9\end{pmatrix}\right)$$

$$\mu_1 = \begin{pmatrix} 4.2 \\ 4.5 \end{pmatrix}$$

$$(17)$$

For class 2:

$$\mu_2 = \frac{1}{n_2}\sum_{X\in\omega 2} X = \frac{1}{6}\left(\begin{pmatrix}4\\5\end{pmatrix}+\begin{pmatrix}5\\0\end{pmatrix}+\begin{pmatrix}5\\2\end{pmatrix}+\begin{pmatrix}3\\2\end{pmatrix}+\begin{pmatrix}5\\3\end{pmatrix}+\begin{pmatrix}6\\3\end{pmatrix}\right) \quad (18)$$

$$\mu_2 = \begin{pmatrix} 4.2 \\ 7 \end{pmatrix}$$

Then calculate the covariance value of each data (Muntasa, 2015).

For class 1:

$$\left(X_i[:1]-\mu_1\right)\left(X_i[:1]-\mu_1\right)^T = \left(\begin{pmatrix}1\\2\end{pmatrix}-\begin{pmatrix}4.2\\4.5\end{pmatrix}\right)\left(\begin{pmatrix}1\\2\end{pmatrix}-\begin{pmatrix}4.2\\4.5\end{pmatrix}\right)^T =$$

$$\begin{pmatrix} -3.2 \\ -2.5 \end{pmatrix} \times (-3.2 \quad -2.5) = \qquad (19)$$

$$\begin{pmatrix} 10.24 & 8 \\ 8 & 6.25 \end{pmatrix}$$

Do the same for 2-6 data. Be careful when selecting data.

$$\left(X_1[:2]-\mu_1\right)\left(X_1[:2]-\mu_1\right)^T = \begin{pmatrix} 4.84 & 3.3 \\ 3.3 & 2.25 \end{pmatrix}$$

$$\left(X_1[:3]-\mu_1\right)\left(X_1[:3]-\mu_1\right)^T = \begin{pmatrix} 1.44 & 1.8 \\ 1.8 & 2.25 \end{pmatrix}$$

$$\left(X_1[:4]-\mu_1\right)\left(X_1[:4]-\mu_1\right)^T = \begin{pmatrix} 0.04 & -0.1 \\ -0.1 & 0.25 \end{pmatrix}$$

$$\left(X_1[:5]-\mu_1\right)\left(X_1[:5]-\mu_1\right)^T = \begin{pmatrix} 0.64 & 0.4 \\ 0.4 & 0.25 \end{pmatrix}$$

$$\left(X_1[:6]-\mu_1\right)\left(X_1[:6]-\mu_1\right)^T = \begin{pmatrix} 33.64 & 26.1 \\ 26.1 & 20.25 \end{pmatrix}$$

For class 2:

$$\left(X_2[:1]-\mu_2\right)\left(X_2[:1]-\mu_2\right)^T = \left(\begin{pmatrix} 4 \\ 2 \end{pmatrix}-\begin{pmatrix} 4.7 \\ 2 \end{pmatrix}\right)\left(\begin{pmatrix} 4 \\ 2 \end{pmatrix}-\begin{pmatrix} 4.7 \\ 2 \end{pmatrix}\right)^T =$$

$$\begin{pmatrix} -0.7 \\ 0 \end{pmatrix}\times\begin{pmatrix} -0.7 & 0 \end{pmatrix} = \qquad (20)$$

$$\begin{pmatrix} 0.49 & 0 \\ 0 & 0 \end{pmatrix}$$

Do the same for 2-6 data. Be careful when selecting data.

$$\left(X_2[:2]-\mu_2\right)\left(X_2[:2]-\mu_2\right)^T = \begin{pmatrix} 0.09 & -0.6 \\ -0.6 & 4 \end{pmatrix}$$

$$\left(X_2[:3]-\mu_2\right)\left(X_2[:3]-\mu_2\right)^T = \begin{pmatrix} 0.09 & 0 \\ 0 & 0 \end{pmatrix}$$

$$\left(X_2[:4]-\mu_2\right)\left(X_2[:4]-\mu_2\right)^T = \begin{pmatrix} 2.89 & 0 \\ 0 & 0 \end{pmatrix}$$

$$\left(X_2[:5]-\mu_2\right)\left(X_2[:5]-\mu_2\right)^T = \begin{pmatrix} 0.09 & 0.3 \\ 0.3 & 1 \end{pmatrix}$$

$$\left(X_2[:6]-\mu_2\right)\left(X_2[:6]-\mu_2\right)^T = \begin{pmatrix} 1.69 & 1.3 \\ 1.3 & 1 \end{pmatrix}$$

Then, count S_W but before calculating S_W first determine S_1 and S_2. Here's, the Eq. 21 (Muntasa, 2015):

$$S_1 = \frac{1}{6-1}\left(\begin{pmatrix} 10.24 & 8 \\ 8 & 6.25 \end{pmatrix}+\begin{pmatrix} 10.24 & 8 \\ 8 & 6.25 \end{pmatrix}+\begin{pmatrix} 1.44 & 1.8 \\ 1.8 & 2.25 \end{pmatrix}+\begin{pmatrix} 0.04 & -0.1 \\ -0.1 & 0.25 \end{pmatrix}+\begin{pmatrix} 0.64 & 0.4 \\ 0.4 & 0.25 \end{pmatrix}+\begin{pmatrix} 33.64 & 26.1 \\ 26.1 & 20.25 \end{pmatrix}\right) =$$

$$\begin{pmatrix} 10.168 & 7.9 \\ 7.9 & 6.3 \end{pmatrix}$$

$$S_2 = \frac{1}{6-1}\left(\begin{pmatrix} 0.49 & 0 \\ 0 & 0 \end{pmatrix}+\begin{pmatrix} 0.09 & -0.6 \\ -0.6 & 4 \end{pmatrix}+\begin{pmatrix} 0.09 & 0 \\ 0 & 0 \end{pmatrix}+\begin{pmatrix} 2.89 & 0 \\ 0 & 0 \end{pmatrix}+\begin{pmatrix} 0.09 & 0.3 \\ 0.3 & 1 \end{pmatrix}+\begin{pmatrix} 1.69 & 1.3 \\ 1.3 & 1 \end{pmatrix}\right) =$$

$$\begin{pmatrix} 1.068 & 0.2 \\ 0.2 & 1.2 \end{pmatrix}$$

$$S_w = S_1+S_2 = \begin{pmatrix} 10.168 & 7.9 \\ 7.9 & 6.3 \end{pmatrix}+\begin{pmatrix} 1.068 & 0.2 \\ 0.2 & 1.2 \end{pmatrix}$$

$$S_w = \begin{pmatrix} 11.236 & 8.1 \\ 8.1 & 7.5 \end{pmatrix}$$

Calculate the average of all classes (Muntasa, 2015):

$$\mu = \frac{1}{n_1+n_2}\sum_{X_{i\in 1,\,\text{row}}} X =$$

$$\frac{1}{12}\left(\begin{pmatrix} 1 \\ 2 \end{pmatrix}+\begin{pmatrix} 2 \\ 3 \end{pmatrix}+\begin{pmatrix} 3 \\ 3 \end{pmatrix}+\begin{pmatrix} 4 \\ 5 \end{pmatrix}+\begin{pmatrix} 5 \\ 5 \end{pmatrix}+\begin{pmatrix} 10 \\ 9 \end{pmatrix}+\begin{pmatrix} 4 \\ 5 \end{pmatrix}+\begin{pmatrix} 5 \\ 0 \end{pmatrix}+\begin{pmatrix} 5 \\ 2 \end{pmatrix}+\begin{pmatrix} 3 \\ 2 \end{pmatrix}+\begin{pmatrix} 5 \\ 3 \end{pmatrix}+\begin{pmatrix} 6 \\ 3 \end{pmatrix}\right) = \qquad (21)$$

$$\begin{pmatrix} 4.42 \\ 3.25 \end{pmatrix}$$

$$S_{B1} = n_1\times(\mu_1-\mu)\times(\mu_1-\mu)^T$$

$$S_{B1} = 6\times\begin{pmatrix} 4.2 \\ 4.5 \end{pmatrix}\times\begin{pmatrix} 4.2 & 4.5 \end{pmatrix}$$

$$S_{B1} = \begin{pmatrix} 105.84 & 113.4 \\ 113.4 & 121.5 \end{pmatrix}$$

$$S_{B2} = n_2\times(\mu_2-\mu)\times(\mu_2-\mu)^T$$

$$S_{B2} = 6\times\begin{pmatrix} 4.7 \\ 2 \end{pmatrix}\times\begin{pmatrix} 4.7 & 2 \end{pmatrix}$$

$$S_{B2} = \begin{pmatrix} 132.54 & 56.4 \\ 56.4 & 24 \end{pmatrix}$$

Calculate the SB (Muntasa, 2015):

$$S_B = S_{B1}+S_{B2} = \begin{pmatrix} 105.84 & 113.4 \\ 113.4 & 121.5 \end{pmatrix}+\begin{pmatrix} 132.54 & 56.4 \\ 56.4 & 24 \end{pmatrix}$$

$$S_B = \begin{pmatrix} 238.38 & 169.8 \\ 169.8 & 145.5 \end{pmatrix}$$

Calculate the Eigenvalue (Muntasa, 2015):

$$A = S_B \times S_w^{-1}$$

$$A = \begin{pmatrix} 238.38 & 169.8 \\ 169.8 & 145.5 \end{pmatrix} \times \begin{pmatrix} 11.236 & 8.1 \\ 8.1 & 7.5 \end{pmatrix}^{-1}$$

$$A = \begin{pmatrix} 238.38 & 169.8 \\ 169.8 & 145.5 \end{pmatrix} \times \begin{pmatrix} 0.4019 & -0.4340 \\ -0.4340 & 0.6021 \end{pmatrix} \quad (22)$$

$$A = \begin{pmatrix} 22.1117 & -1.2203 \\ 5.09562 & 13.91235 \end{pmatrix}$$

$$\det\det(\lambda I - A) = 0$$

$$\det\det\left(\lambda \begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \begin{bmatrix} 22.1117 & -1.2203 \\ 5.09562 & 13.91235 \end{bmatrix} \right) = 0$$

$$\det\det\left(\begin{bmatrix} \lambda & 0 \\ 0 & \lambda \end{bmatrix} - \begin{bmatrix} 22.1117 & -1.2203 \\ 5.09562 & 13.91235 \end{bmatrix} \right) = 0$$

$$\det\det\left(\begin{bmatrix} \lambda-22.1117 & 1.2203 \\ -5.09562 & \lambda-13.91235 \end{bmatrix} \right) = 0$$

$$\det\det \begin{bmatrix} \lambda-22.1117 & 1.2203 \\ -5.09562 & \lambda-13.91235 \end{bmatrix} = 0$$

$$\big((\lambda-22.1117)(\lambda-13.91235)\big) - \big((-5.09562)(1.2203)\big) = 0$$

$$\lambda^2 - 36.02405\lambda + 307.6257095 + 6.218185086$$

$$\lambda^2 - 36.02405\lambda + 313.8438946$$

$$\frac{-b \pm \sqrt{(b)^2 - 4ac}}{2a} = \frac{36.02405 \pm \sqrt{(36.02405)^2 - 4(1)(313.8438946)}}{2(1)}$$

$$= \frac{36.02405 \pm 6.508194834}{2}$$

$$\lambda_1 = 21.26612242, \lambda_2 = 14.75792758$$

$$(23)$$

λ_1 and λ_2 are the Eigenvalues obtained from the LDA method. Then calculate the Eigenvector. Calculate the Eigenvector (Muntasa, 2015):

$$\begin{pmatrix} \lambda-22.1117 & 1.2203 \\ -5.09562 & \lambda-13.91235 \end{pmatrix} \begin{pmatrix} x_1 \\ x_2 \end{pmatrix} = \lambda \begin{pmatrix} x_1 \\ x_2 \end{pmatrix}$$

$$(\lambda-22.1117)(x_1) + 1.2203x_2 = 0$$

$$-5.09562x_2 + (\lambda-13.91235)(x_1) = 0$$

For $\lambda_1 = 21.26612242$

$$(\lambda-22.1117)(x_1) + 1.2203x_2 = 0$$

$$\lambda x_1 - 22.1117x_1 + 1.2203x_2 = 0$$

Substitution $\lambda_1 = 21.26612242$ value

$$21.26612242x_1 - 22.1117x_1 + 1.2203x_2 = 0$$

$$-0.84557758x_1 + 1.2203x_2 = 0$$

$$1.2203x_2 = 0.84557758x_1$$

$$x_2 = \frac{0.84557758x_1}{1.2203}, x_1 = \frac{1.2203x_2}{0.84557758}$$

$$-5.09562x_2 + (\lambda-13.91235)(x_1) = 0$$

$$-5.09562x_2 + \lambda x_1 - 13.91235x_1 = 0$$

Substitution $\lambda_1 = 21.26612242$ value

$$-5.09562x_2 + 21.26612242x_1 - 13.91235x_1 = 0$$

$$-5.09562x_2 + 7.35377242x_1 = 0$$

$$7.35377242x_1 = 5.09562x_2$$

$$x_1 = \frac{5.09562x_2}{7.35377242}, x_2 = \frac{7.35377242x_1}{5.09562}$$

x_1 and x_2 that have been obtained are Eigenvectors:

For $\lambda_2 = 14.75792758$

$$(\lambda-22.1117)(x_1) + 1.2203x_2 = 0$$

$$\lambda x_1 - 22.1117x_1 + 1.2203x_2 = 0$$

$$\lambda_2 = 14.75792758$$

$$14.75792758x_1 - 22.1117x_1 + 1.2203x_2 = 0$$

$$-7.35377242x_1 + 1.2203x_2 = 0$$

$$1.2203x_2 = 7.35377242x_1$$

$$x_2 = \frac{7.35377242x_1}{1.2203}, x_1 = \frac{1.2203x_2}{7.35377242}$$

$$-5.09562x_2 + (\lambda-13.91235)(x_1) = 0$$

$$-5.09562x_2 + \lambda x_1 - 13.91235x_1 = 0$$

$$\lambda_2 = 14.75792758$$

$$-5.09562x_2 + 14.75792758x_1 - 13.91235x_1 = 0$$

$$-5.09562x_2 + 0.84557758x_1 = 0$$

$$0.84557758x_1 = 5.09562x_2$$

$$x_1 = \frac{5.09562x_2}{0.84557758}, x_2 = \frac{0.84557758x_1}{5.09562}$$

x_1 and x_2 that have been obtained are Eigenvectors.

Further projected the Eigenvector to be extracted to find the Euclidean distance value between the projection result and the input image. The Euclidean distance value is the value of similarity between the inputted images with training data.

Mean Square Error (MSE) and Peak Signal to Noise Ratio (PSNR): The Mean Square Error (MSE) and Peak Signal to Noise Ratio (PSNR) are the methods used for objective quality measurement. In this context, quality measurements are based on image quality by comparing the original image quality with the result of improving image quality. Before determining PSNR, first determine the value of MSE because PSNR and MSE are interconnected. So, if the value of MSE is smaller, then the value of PSNR will be greater and if the value of PSNR is

greater, then the image quality will be better and vice versa if the MSE value is greater, then the PSNR value will be smaller. Here is the equation of PSNR (Madenda, 2015):

$$MSE = \frac{1}{C \times M \times N} \sum_{i=0}^{M-1} \sum_{j=0}^{N-1} (I(i,j) - R(i,j))^2 \qquad (24)$$

where, I(i, j) is the pixel image value of the original image at position (i, j). Then R (i, j) is the pixel image value of the reconstruction image at position (i, j). M and N, respectively are the height and width of the image in pixel size whereas C represents the number of color components of the image: C = 1 for binary or grayscale image, C = 3 for color image.

As for PSNR is a mathematical approach of human perception of image quality. The unit for PSNR is decibels (dB). Here is the equation of PSNR (Madenda, 2015):

$$PSNR = 10\log_{10}\left(\frac{Max_I^2}{MSE}\right) \qquad (25)$$

where, Max_I is the maximum value of an original image pixel I. The value Max_I = 255 when the pixel image color component is encoded in 8 bits.

MATERIALS AND METHODS

This research aims to recognize a person's face with retinex image repair method of the infrared camera (Fig. 2).

System overview: The process flow from the above general description starts from: the image was taken using the Yi-Dome 720p infrared camera. For the number of samples in this research using 10 different people. There are two kinds of resolutions on the Yi-Dome camera that is 480 and 720p. In this research use 720p resolution with pixel dimensions 1280×720. After face image obtained then separated between train and test data using PC/Laptop. After train data and test data are ready, then tested using MATLAB application for digital image processing. The MATLAB application used is MATLAB 2017.

System design from the software side: The process flow from the design of the built system starts from (Fig. 3).

The image was taken using the infrared camera. The image taken is divided into two folders: train data folder and test data folder. In the data trainer and test, contains images with very dark, dim light and bright light conditions. Infrared images with different face samples
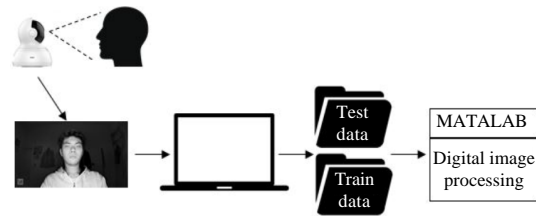

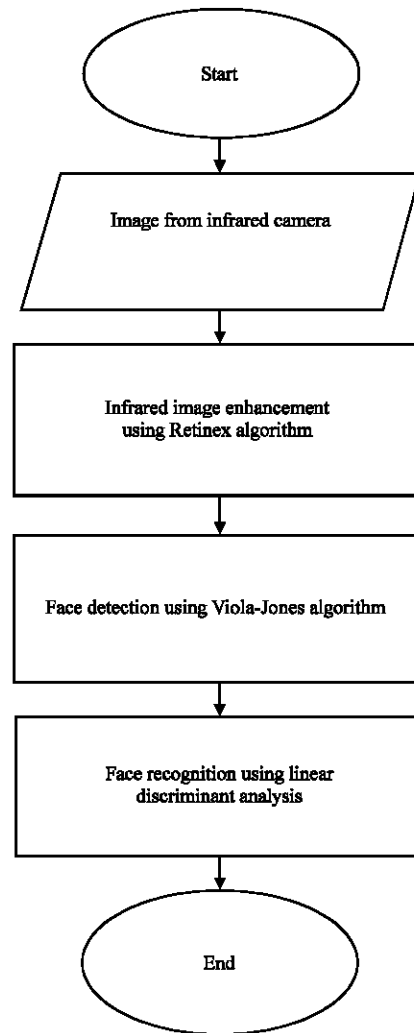Fig. 2: System overview of this research

Fig. 3: Flowchart of the system

will be processed with the Retinex algorithm for image repair and also, Retinex algorithm in this research is part of pre-processing. This image improvement aims to make the image with low quality and light to be better than the previous condition. When test data is inputted, the image will be resized with dimension size 200×180 pixels. Because If the test data is inputted using the image

Table 1: Detail of training data

Data types	Level of lighting (lux)	Pose	Amount of data
Bright light	20	Flat face and smiley face	30
Dim light	15	Flat face and smiley face	15
Very dark	0	Flat face and smiley face	15
Very dark+Dim light	0 and 15	Flat face and smiley face	30
Mixed training data	20, 15 and 0	Flat face and smiley face	60
Very dark+Face angle	0	Flat face and smiley face+Face angle of 30 and 60°	27
Total training data×10 persons	177×10 = 1770		

original size of 1280×720 pixels that occur is during the training process takes a long time. In condition cause the laptop crash/freeze. Next face detection is done with Viola-Jones algorithm. This face detector is used, so that, in the process of recognizing the face not the whole pixel is recognizable but more focused on the recognition of the human face. After the human face is detected then face recognition process with Linear Discriminant Analysis (LDA) algorithm. Then for the final result will be analyzed the effect of Retinex method to increase face recognition accuracy.

Train data specification and test data specification: Making the database in this research has two types of data, namely train data and test data. Train data is data from a collection of people faces who will be used as data for face recognition and test data is data from a collection of faces of people but with conditions slightly different from the data training to be used as test data in face recognition. All these data are face images of different people. For shooting using the Yi-Dome camera with a maximum resolution of 720 p (1280×720 pixels) is equipped with two camera modes, the regular camera and the infrared camera.

Train data: Train data is using 10 images of different people's face with each person is taking 30 data for normal poses for bright light conditions, 15 data for dim light and very dark. Each of the shooting conditions is determined by Lux unit. To determine the value of Lux is using the application android Lux Meter. For the condition of very light used 10 W lamp with 20 lux value for the condition of dim light used 5 W lamp with value 15 lux and for very dark condition do not wear lamp at all with value 0 lux (Table 1 and 2).

Test data: For the test data used still use the same 10 persons such as training data. Light conditions used as test data is very dark, dim light and bright light with the amount of test data each person amounted to 15 data. So, for the total test data used are 15 images of each light condition multiplied by 10 persons, i.e., 450 images in the test scenario without retinex whereas for the retinex test

Table 2: Sample of training data

Data types	Example of training data
Very dark	
Dim light	
Bright light	
Very dark with angle poses	

Table 3: Detail of test data

Data types	Level of lighting (lux)	Pose	Amount of data
Bright light	20	Flat face and smiley face	15
Dim light	15	Flat face and smiley face	15
Very dark	0	Flat face and smiley face	15
Very dark 80 cm	0	Flat face and smiley face	15
Very dark 120 cm	0	Flat face and smiley face	15
Very dark 160 cm	0	Flat face and smiley face	15
Very dark with angle poses	0	Face angle of 30 and 60°	12
Total test data×10 persons	102×10 = 1020		

scenario the test data used is only very dark with two different sigma values on the SSR test, i.e. (SSR1 Sigma = 2) were 15 persons and (SSR2 Sigma = 200) were 15 persons and two sigma types in the MSR test (MSR1 Sigma = 2, 80, 220) were 15 persons and (MSR2 Sigma = 4, 60, 200) were 15 persons. So, the total test data is using retinex 600 images. And test data with very dark, dim light and bright light conditions, each test data has been changed using MSR (Sigma 1 = 2, 80, 220) as a test for comparison of faces recognition without using retinex and not using retinex (Table 3 and 4).

Test scenario: In testing of this system, there are 5 test scenarios. This test uses train data and test data previously created. With this scenario can be seen the comparison between the test results of each scenario including the system performance and accuracy level of similarity of face recognition system that has been built.

Testing the 1st model: In this testing model was a test without using retinex (SSR and MSR). Light conditions used for testing are very dark, dim light and bright light. Then for train data use mixed training data (Table 5).

Testing the 2nd model: In this testing model, used very dark test data and there are 4 tests. First, testing SSR1 using (Sigma = 2), SSR2 using (Sigma = 200), MSR1 using (Sigma = 2, 80, 220) and MSR2 using (Sigma = 4, 60, 200). It's just added SSR and MSR image enhancement methods for face recognition. Then for the training data still use mixed training data (Table 6).

Testing the 3rd model: In this test model was a test of light conditions. Divided into 3 test groups, namely same light condition, cross crosses condition and mixed training data condition (Table 7).

Testing the 4th model: In this test model is a distance testing. Divided into 3 tests, the distance of 80, 120 and 160 cm. Then for train data use is mixed training data (Table 8).

Table 4: Sample of test data

Data types	Example of test data
Very dark	
Dim light	
Bright light	
Very dark with distance 80 cm	
Very dark with distance 120 cm	
Very dark with distance 160 cm	

Table 4: Continue

Data types	Example of test data
Very dark with angle poses	

Table 5: Details of 1st test scenario

ith test model/ Testing	Test data used	Level of test data lighting	Training data used
1st model			
1	Normal position	Very dark	Mixed training data
2	Normal position	Dim light	Mixed training data
3	Normal position	Bright light	Mixed training data

Table 6: Details of 2nd test scenario

ith test model/ Testing	Test data used	Level of test data lighting	Training data used
2nd model			
1	Normal position	Very dark SSR1	Mixed training data
2	Normal position	Very dark SSR2	Mixed training data
3	Normal position	Very dark MSR1	Mixed training data
4	Normal position	Very dark MSR2	Mixed training data

Table 7: Details of 3rd test scenario

ith test model	Test condition/ Testing	Level of test data lighting	Training data used
3rd model	**Same light**		
	1	Very dark MSR1	Very dark
	2	Dim light MSR1	Dim light
	3	Bright light MSR1	Bright light
3rd model	**Cross light**		
	4	Very dark MSR1	Bright light
	5	Dim light MSR1	Bright light
	6	Bright light MSR1	Very dark+dim light
3rd model	**Mixed training data**		
	7	Very dark MSR1	Mixed training data
	8	Dim light MSR1	Mixed training data
	9	Bright light MSR1	Mixed training data

Table 8: Details of 4th test scenario

ith test model/ Testing	Test data used	Level of test data lighting	Training data used
4th model			
1	80	Very dark MSR1	Mixed training data
2	120	Dim light MSR1	Mixed training data
3	160	Bright light MSR1	Mixed training data

Table 9: Details of 5th test scenario

ith test model/ Testing	Test data used	Level of test data lighting	Training data used
5th model			
1	30°	Very dark MSR1	Very dark+Face angle of 30 and 60°
2	60°	Very dark MSR1	Very dark+Face angle of 30 and 60°

Testing the 5th model: In this test model is an angle testing. Divided into 2 tests, the angle of 30 and 60°. Then for train data use is very dark+very dark with face angle of 30 and 60°. More details can be seen in Table 9.

RESULTS AND DISCUSSION

The software application to recognize the face is shown in Fig. 4. Based on Fig. 4 first step of face recoginition is train the data and save them to the database into*.mat format. To test any face recognition, select the picture or turn on the camera to get the picture. This recognition will be conducted in LDA method. All the ongoing process will be shown on the white column on the right side.

In Fig. 5 is an example of the process of face detection before face recognition using LDA. The detected face area varies depending on the distance from face recognition. If the distance of face recognition is farther away the area of the face detected will be smaller.

For face recognition results of each experiment performed in the form of an accuracy value on each data tested whether recognized or not. With 100% value for the recognizable face and 0% for unrecognized face, then calculated accuracy value based on the ith experiment on each type of experiment with the formula:

$$\text{Accuracy} = \frac{\sum \text{recognized test data}}{\sum \text{test data}} \times 100\% \qquad (7)$$

Result of 1st test scenario (without retinex): This first model scenario uses 3 types of testing with very dark, dim light and bright light test data and uses mixed training data. In this scenario, it does not use Retinex method. The test result on this first model is the level of facial recognition accuracy before using Retinex method.

In Fig. 6, the first scenario model test diagram shows the highest accuracy value that is on the 2nd test with dim data test and 28% accuracy. In this test, the level of illumination affects face recognition accuracy. Because in this dim condition the contrast of light is not too bright and not too dark.

Result of 2nd test scenario (retinex comparison): This second model scenario uses 4 types of testing ie SSR1 with sigma ($\sigma_1 = 2$), SSR2 with sigma ($\sigma_1 = 200$), MSR1 with sigma ($\sigma_1 = 2$, $\sigma_2 = 80$, $\sigma_3 = 220$) and MSR2 with sigma ($\sigma_1 = 4$, $\sigma_2 = 60$, $\sigma_3 = 200$). The training data used in the training data is mixed. All light conditions test data used very dark. This scenario aims to compare the results of Single-Scale Retinex (SSR) with Multi-Scale Retinex (MSR).
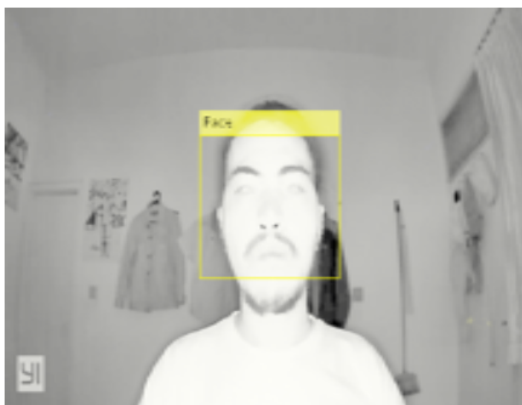
Fig. 4: Example of GUI

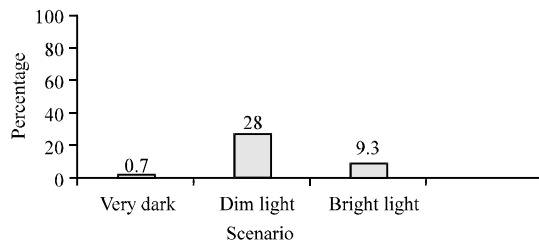Fig. 5: Face detection using Viola-Jones algorithm

Fig. 6: Result of 1st test scenario; Without Retinex testing

In Fig. 7, the second model scenario test diagram shows the highest accuracy value that is on the 3rd test with the test data very dark MSR with sigma value and an accuracy of 6.7%. Based on the test results, the MSR method is the best method for facial recognition using the LDA method because it uses 3 different sigma values.
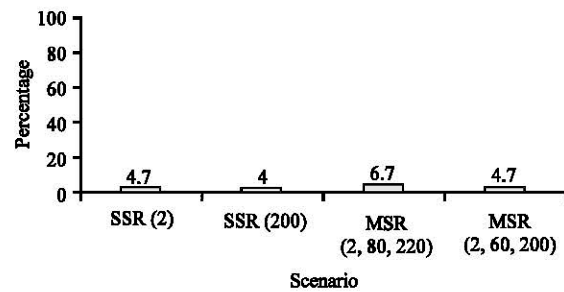

Fig. 7: Result of 2nd test scenario; Comparison of Retinex testing

Result of 3rd test scenario (light conditions): This third model scenario uses 3 different types of testing, namely the same light conditions, cross-light conditions and mixed data training conditions. Each test condition has 3 tests, so, the whole has 9 tests in this test scenario.

In Fig. 8, the diagram of the test results of the third model scenario shows the highest accuracy value that is in testing the condition of mixed training data compared with the same light conditions and cross-light conditions. This occurs because each test data on the data condition of mixed practice using the amount of data train more than the same light conditions and cross-light conditions, so that, the data used to train more complex.

Result of 4th test scenario (distance): This fourth model scenario uses 3 different distance tests which are 80, 120 cm distance and distance of 160 cm with very dark test data and mixed training data.
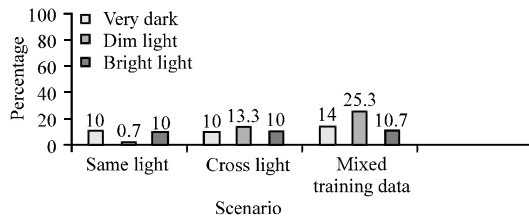
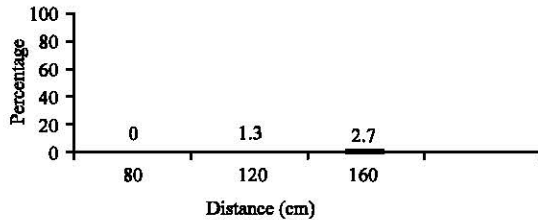Fig. 8: Result of 3rd test scenario; Light conditions testing

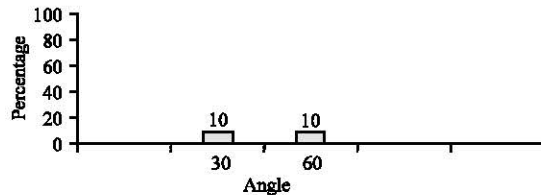Fig. 9: Result of 4th test scenario; Distance tseting

Fig. 10: Result of 5th test scenario; Angle testing

In Fig. 9, the 4th model scenario test diagram shows the highest accuracy value in the 3rd test of the test data very dark MSR sigma (σ_1 = 2, σ_2 = 80, σ_3 = 220) with distance 160 cm retrieval and an accuracy of 2.7%. From this test it can be concluded that the maximum result at a distance of 160 cm.

Result of 5th test scenario (angle): This fifth model scenario uses 2 angle-taking tests of 30 and 60°. All light conditions test data very dark MSR sigma (σ_1 = 2, σ_2 = 80, σ_3 = 220) using very dark+very dark with face angles training data.

In Fig. 10, the 5th model scenario model test diagram shows the same accuracy value of 10% of 2 tests with the test data very dark MSR sigma (σ_1 = 2, σ_2 = 80, σ_3 = 220) with angle 30 and 60°. Because at the recognition of this corner of the eyes, nose and mouth are not very clear in contrast to the normal test that has a clearer face, so that, the face will be easily recognizable.

Analysis from system performance: Here, is the analysis of system performance during face recognition process: the test data resolution affects the process time during the introduction. In this research, the resolution used is 1280×720 pixels by changing the pixel size to 200×180 pixels. If the resolution used is the original resolution, it

takes a longer introduction time can sometimes cause the laptop/PC to freezing. So, in this research, the resolution used is 200×180 pixels.

The resolution size between the test data and training data must be the same, if the resolution size is different, then the process of face recognition will have an error. This is due to the Euclidian distance calculation process requiring the dimensioned condition to be measured must have the same dimensions as the comparator (Yeganeh *et al.*, 2008). The amount of training data can also affect the processing time, if the amount of training data more and more, then the time required for face recognition is also more.

Analysis from face recognition accuracy: In this analysis, several scenarios of test models use Mean Square Error (MSE) and Peak Signal to Noise Ratio (PSNR) to measure image quality. Here is an analysis of each test scenario that has been done: in the first model scenario of testing without retinex, the best face recognition accuracy results in the dim data test with an accuracy score of 28% compared to other test data. In retinex comparison test the best accuracy result is sigma MSR method (σ_1 = 2, σ_2 = 80, σ_3 = 220) with an accuracy value of 6.7% because MSR uses 3 different sigma values than SSR using only 1 sigma.

In testing the condition of light of the best accuracy results is in the condition of data training mix because in this test condition the amount of training data used more and more complex than the other tests. In testing of distance, the best result is a distance of 160 cm with a face recognition value of 2.7% while the closer distance is not recognized. From the average PSNR calculation result that the 160 cm test image has a higher PSNR value of 60.36243 dB compared to the 80 cm test image with the value of PSNR 59.8849 dB and the 120 cm test image with the value of PSNR 59.92222 dB.

At the test angle, there is no best result. The face recognition accuracy of each test has the same value. Because parts of the eyes, nose and mouth are not clearly recognized. The mean value of both test images is 59.88638 dB for angles of 30° and 59.72208 dB for an angle of 60°. From the calculation of PSNR average the difference of these two test images is 0.1643 dB. With this small value difference it cannot make a significant difference.

CONCLUSION

After the test is done, the conclusions obtained from this research are: resolution of the image gives effect when the training data is done, if using the original image

resolution of 1280×720 pixels then the data training process and face recognition will take longer. If using a smaller pixel resolution in this research using a resolution of 200×180 pixels then the data training process and face recognition will be faster.

At face detection, Viola-Jones algorithm not only detects faces but can detect faces from existing posters behind, then this face detection is still not perfect. The amount of training data affect the time of training process data, if the amount of data train more, then the time required for training data will be longer.

Retinex method can improve face recognition accuracy in very dark light conditions. Based on the results of tests that have been done can be concluded that the value of accuracy in face recognition is very small. No accuracy value above 50%.

SUGGESTIONS

The researchers conclude many things that make the accuracy value very small, so, the researchers suggest a few things to improve face recognition accuracy including: The background of the image taken should be a clean background of everything such as clothing, home furnishings and so on. Because at the time of the recognition process many recognizable pixels face, so that, the algorithm does not focus on face pixels but all the pixels in the image will be recognized. It is recommended to use a white screen in the background if you do not have a room like a photo studio.

Using the high-resolution camera in this research the camera used only 1280×720 pixel resolution. Resize the image to be not too small from the original resolution in this study the original resolution of the image is 1280×720 pixels which are resized to 200×180 pixels. In this case, the decline in resolution is very extreme. The researcher suggests converting it to 480×360 pixels.

Does not change the dimension of the matrix because it would be more effective to use a 2D matrix rather than convert it into a 1D matrix. Addition of image repair method to remove noise on the infrared image because basically retinex method only improves image from the side of lighting, not to eliminate noise and increase image detail. Increase the amount of train data because the weakness of the LDA is to require a large amount of training data in order to produce better generalizations (Liu and Wechsler, 2002).

REFERENCES

Battalwar, P., J. Gokhale and U. Bansod, 2015. Infrared thermography and IR camera. Intl. J. Res. Sci. Eng., 1: 9-14.

Borade, S.N., R.R. Deshmukh and S. Ramu, 2016. Face recognition using fusion of PCA and LDA: Borda count approach. Proceedings of the 2016 24th Mediterranean Conference on Control and Automation (MED), June 21-24, 2016, IEEE, Athens, Greece, ISBN:978-1-4673-8347-9, pp: 1164-1167.

Dinata, S., 2014. Implementation of the multiscale retinex method for image enhancement. BS Thesis, Department of Computer Science and Technology Information, Sumatera Utara University, Medan, Indonesia.

Egorov, A.D., D.U. Divitskii, A.A. Dolgih and G.A. Mazurenko, 2018. Some cases of optimization face detection methodes on image (Using the Viola-Jones method as an example). Proceedings of the 2018 IEEE International Conference on Russian Young Researchers in Electrical and Electronic Engineering (EIConRus), January 29-February 1, 2018, IEEE, Moscow, Russia, ISBN:978-1-5386-4339-6, pp: 1075-1078.

Janani, V. and M. Dinakaran, 2014. Infrared image enhancement techniques-a review. Proceedings of the 2nd International Conference on Current Trends in Engineering and Technology (ICCTET 2014), July 8, 2014, IEEE, Coimbatore, India, ISBN:978-1-4799-7986-8, pp: 167-173.

Li, D., Y. Zhang, P. Wen and L. Bai, 2015. A retinex algorithm for image enhancement based on recursive bilateral filtering. Proceedings of the 2015 11th International Conference on Computational Intelligence and Security (CIS), December 19-20, 2015, IEEE, Shenzhen, China, ISBN:978-1-4673-8659-3, pp: 154-157.

Li, Y., C. Hou, F. Tian, H. Yu and L. Guo et al., 2007. Enhancement of infrared image based on the Retinex theory. Proceedings of the 29th Annual International Conference on Engineering in Medicine and Biology Society (EMBS 2007), Auguest 22-26, 2007, IEEE, Lyon, France, ISBN:978-1-4244-0787-3, pp: 3315-3318.

Liu, C. and H. Wechsler, 2002. Gabor feature based classification using the enhanced fisher linear discriminant model for face recognition. Image Proc. IEEE Trans., 11: 467-476.

Liu, H., X. Sun, H. Han and W. Cao, 2016. Low-light video image enhancement based on multiscale Retinex-like algorithm. Proceedings of the 2016 International Conference on Chinese Control and Decision Conference (CCDC), May 28-30, 2016, IEEE, Yinchuan, China, ISBN:978-1-4673-9715-5, pp: 3712-3715.

Madenda, S., 2015. [Digital Image and Video Processing: Theory, Application and Programming Using MATLAB]. Penerbit Erlangga, Jakarta Indonesia, ISBN:9786022985283, Pages: 312 (In Indonesian).

Muntasa, A., 2015. [Pattern Recognition: Application for Face Identification, Object Texture Analysis, Introduction to Vehicle Number Plates and Blood vessel Segmentation]. Graha Ilmu, Ruko Jambusari, Yogyakarta, ISBN:978-602-262-471-4, Pages: 265 (In Malay).

Nehru, M. and S. Padmavathi, 2017. Illumination invariant face detection using viola jones algorithm. Proceedings of the 2017 4th International Conference on Advanced Computing and Communication Systems (ICACCS), January 6-7, 2017, IEEE, Coimbatore, India, ISBN:978-1-5090-4560-0, pp: 1-4.

Prasasti, A.L., R.K.W. Mengko and W. Adiprawita, 2015. Vein tracking using 880 nm near infrared and CMOS sensor with maximum curvature points segmentation. Proceedings of the 7th International Conference on WACBE World Congress on Bioengineering 2015, July 6-8, 2015, Springer, Cham, Switzerland, Europe, pp: 206-209.

Putro, M.D., T.B. Adji and B. Winduratna, 2012. [Face detection system using the Viola-Jones method]. Proceedings of the National Seminar on Science, Engineering and Technology (SciETec 2012), February 22, 2012, University of Brawijaya, Malang, Jawa Timur, Indonesia, pp: 71-75 (In Indonesian).

Shu-Xin, L., L. Ji-Chen, Z. Zhi-Long and Y. Wei-Ping, 2013. Infrared image enhancement algorithm based on retinex theory and adaptive gain control. Proceedings of the 2013 IEEE International Conference on Signal Processing, Communication and Computing (ICSPCC 2013), August 5-8, 2013, IEEE, KunMing, China, ISBN:978-1-4799-1027-4, pp: 1-5.

Sun, L., J. Wang and S. Li, 2017. A new infrared image enhancement algorithm. Proceedings of the 2017 29th International Conference on Chinese Control and Decision Conference (CCDC), May 28-30, 2017, IEEE, Chongqing, China, ISBN:978-1-5090-4658-4, pp: 421-424.

Viola, P. and M. Jones, 2001. Rapid object detection using a boosted cascade of simple features. Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2001), December 8-14, 2001, IEEE, Kauai, Hawaii, USA., pp: 511-518.

Wang, Y. and Y. Chen, 2017. A new feature extraction algorithm based on fisher linear discriminant analysis. Proceedings of the 2017 3rd International Conference on Control, Automation and Robotics (ICCAR), April 24-26, 2017, IEEE, Nagoya, Japan, ISBN:978-1-5090-6089-4, pp: 414-417.

Yeganeh, H., A. Ziaei and A. Rezaie, 2008. A novel approach for contrast enhancement based on histogram equalization. Proceedings of the 2008 International Conference on Computer and Communication Engineering (ICCCE 2008), May 13-15, 2008, IEEE, Kuala Lumpur, Malaysia, ISBN:978-1-4244-1691-2, pp: 256-260.

Yin, J., H. Li, J. Du and P. He, 2014. Low illumination image Retinex enhancement algorithm based on guided filtering. Proceedings of the 2014 IEEE 3rd International Conference on Cloud Computing and Intelligence Systems (CCIS), November 27-29, 2014, IEEE, Shenzhen, China, ISBN:978-1-4799-4720-1, pp: 639-644.