Face Recognition Using the Direct GLCM and K-NN Methods

I Komang Astina Adiputra, Raditiana Patmasari and Rita Magdalena
School of Electrical Engineering, Telkom University,
Jl. Telekomunikasi, Jl. Terusan Buah Batu No. 01, Bandung, Indonesia 40257
E-mail:komangastina77@gmail.com, raditiana@telkomuniversity.ac.id, ritamagdalena@telkomuniversity.ac.id

***Abstract*—This paper introduces a new face recognition method based on the gray-level co-occurrence matrix (GLCM). This method directly uses GLCM by converting a matrix into a vector that can be used as a feature vector for the classification process, this method is called direct GLCM. The classification process used is K-Nearest Neighbor (K-NN), in which the classification process compares the features contained in K-NN namely Euclidean distance, Cityblock, Chebychev, and Minkowski. The results show that using direct GLCM as a feature vector in the recognition process using the K-NN classification with the Cityblock feature produces an accuracy of 84.29%, FAR 6.67% and FRR 9.05%.**

***Index Terms*—Face recognition, Gray-Level Co-occurrence Matrix, K-Nearest Neighbor, Cityblock.**

I. Introduction

Face Recognition has always attracted the attention of researchers as one of the most important techniques for human identification. One limitation of the real-time recognition system is the computational complexity of the existing approach. Many systems and algorithms have been introduced in the past few decades with high recognition rates. So, in this study the face image extraction method used is direct GLCM because it has a solution of the above problems, namely this method is very competitive and exceeds sophisticated facial recognition techniques such as PCA and LDA [1]. Using a small amount of gray level makes the algorithm faster and at the same time this method maintains a high level of recognition accuracy [2].

Direct GLCM is a method developed from the GLCM extraction method using hiralick features namely Energy, Entropy, Contrast, Variance, Homogeneity, Correlation, Sum Average, Sum Entropy, Sum Variance, Difference Variance, Difference Entropy, Maximum Correlation Coefficient, Information Measures of Correlation [3]. This method directly uses GLCM by converting a matrix into a vector that can be used as a feature vector for the classification process [1] [2].

The classification method used is K-Nearest Neighbor (K-NN), where in this classification process will compare the classification features that cause high levels of face recognition accuracy. Features used in the classification process are Euclidean distance, Cityblock, Chebychev, and Minkowski [4].

II. GRAY-LEVEL CO-OCCURENCE MATRIX

Gray Level Co-occurrence Matrix (GLCM) is a texture descriptor that is widely used and it is proven that the results obtained from the co-occurrence matrix are better than other texture discrimination methods. GLCM calculates statistical features based on the intensity of the gray level of the image. Features like GLCM are useful in texture recognition, image segmentation, image retrieval, color image analysis, image classification, object recognition, and texture analysis methods etc [5]. One of the simplest approaches to describe texture is to use statistical moments from the histogram of the intensity of an image or region [6]. Using histograms only in calculations, will produce texture sizes that only carry information about the intensity distribution, but not about the relative position of pixels towards each other in that texture. Using a statistical approach like a co-occurrence matrix will help to provide valuable information about the relative position of neighboring pixels in an image [1]. Given the image I, the size of N $\times$ N, the co-occurrence matrix P can be defined as follows:

$$P(i,j) = \sum_{x=1}^{N}\sum_{y=1}^{N}\begin{cases}1, if I\left(x,y\right)=i, I\left(x+\Delta_x+\Delta_y\right)=j\\ 0, otherwise\end{cases} \tag{1}$$

Where the offset (Δ_x, Δ_y), is the distance between pixel-of-interest and its neighbor. Note that offset parameters make the co-occurrence matrix very sensitive to rotation [1]. Fig 1 shows the shared empirical results using the level Ng = 5 and offset [0 1], [-1 1], [-1 -1] interpreted as one pixel passing in the specified four directions.

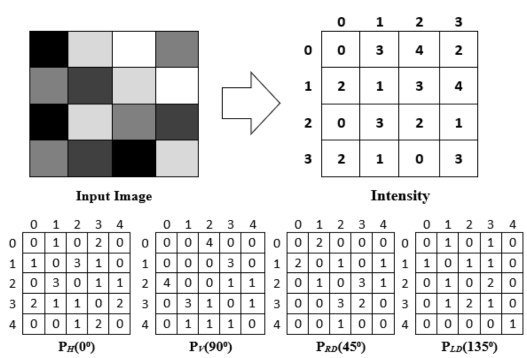

Fig. 1. The co-occurrence matrix with a level of Ng = 5 and four offsets: $P_H(0^o), P_V(90^o), P_{RD}(45^o), P_{LD}(135^o)$ [1].

The average gray-level co-occurrence matrix can be calculated by:

$$P = \frac{P_H + P_V + P_{RD} + P_{LD}}{4} \tag{2}$$

III. K-NEAREST NEIGHBOR

The K-NN algorithm uses neighborhood classification as the predictive value of the value of the new instance. K-NN works based on the minimum distance from the new data to the training data sample to determine the nearest K neighbor. After that, we will get the majority value as a result of predictions from the new data.

Calculation of distance in the K-NN method can be done using the Euclidean distance formula, Cityblock, Chebychev, and Minkowski [4].

- Euclidean Distance:

$$j(v_1, v_2) = \sqrt{\sum_{k=1}^{N} (v_1(k) - v_1(k))^2} \tag{3}$$

- Cityblock or Manhattan distance:

$$j(v_1, v_2) = \sum_{k=1}^{N} |v_1(k) - v_1(k)| \tag{4}$$

- Chebychev:

$$j(v_1, v_2) = max_{k=1 \to N}(|v_1(k) - v_1(k)|) \tag{5}$$

- Minkowsk:

$$j(v_1, v_2) = \sqrt[\frac{1}{p}]{\sum_{k=1}^{N} (v_1(k) - v_1(k))^p} \tag{6}$$

with:
j : a distance of test data to training data
$v_1(k)$: feature k test data, with k = 1,2,3, ..., N
$v_2(k)$: feature k training data, with k = 1,2,3, ..., N

IV. SYSTEM PLAN

System flow diagram there are two processes, namely the process of training data stored in the database and the process of testing data which ends with the classification process. The system flow diagram can be seen in Fig 2.

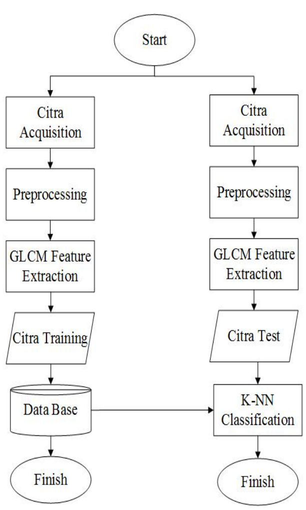

Fig. 2. Flow chart of the training process and test process.

V. SIMULATION RESULT

For the gray level value of Direct GLCM is 64 (Ng = 64). After going through all the processes according to the system flow diagram, the system outputs are accuracy, FAR, and FRR of the four K-NN features that will be compared. The closest neighbor to the classification process used is 1 (K = 1). The comparison of system accuracy from the four K-NN features can be seen in Table 1 and the comparison of system FAR and FRR from the four K-NN features can be seen in Table 2.

TABLE I
COMPARATIVE ANALYSIS

Recognition rate	**K-NN Feature**			
	Euclidean	***Cityblock***	***Chebychev***	***Minciwski***
Percentage	79.52%	84.29%	63.34%	66.67%

TABLE II
COMPARATIVE ANALYSIS FAR & FRR

Recognition rate	**K-NN Feature**			
	Euclidean	***Cityblock***	***Chebychev***	***Minciwski***
FAR	12.38%	6.67%	28.1%	29.52%
FRR	8.1%	9.05%	6.67%	3.81%

VI. CONCLUTION

After the design and implementation has been carried out and face recognition testing has been carried out, conclusions can be drawn, where using the GLCM direct feature extraction method and using the four K-NN feature classification methods (Euclidean distance, Cityblock, Chebychev, and Mincowski) get the best accuracy with features Cityblock. Using the Cityblock feature in the classification stage produces the best accuracy of 84.29 % with FAR of 6.67 % and FRR of 9.05 %. So in the face recognition process using GLCM direct feature extraction using 210 test images, it is recommended to use the K-NN classification method with Cityblock feature.

REFERENCES

[1] E. Alaa, D. Hasan, "Co-Occurrence based Statistical Approach for Face Recognition, IEEE, Turkish Republic of Northern Cyprus, 2009.
[2] E. Alaa, D. Hasan, Co-occurrence matrix and its statistical features as a new approach for face recognition, Tubitak, vol. 19, no. 1, p. 97 107, 2011.
[3] D. Gadkari, Image Quality Analysis Using GLCM, University of Central Florida, Orlando, 2004.
[4] A. Kadir and A.Susanto, Teori dan Aplikasi Pengolahan citra. Yogyakata: CV.Andi Offset, 2013.
[5] C. Nageswara Rao, S. Sreehari Sastry, K. Mallika, H. S. Tiong and K. Mahalaksmi. Co-Occurrence Matrix and Its Statistical Features as an Approach for Identification Of Phase Transitions Of Mesogens, International Journal of Innovative Research in Science, Engineering and Technology, vol. 2, no. 9, p. 4531 4538, 2013.
[6] R. C. Gonzalez, R. E. Woods, Digital Image Processing, 3rd Ed. Prentice Hall, 2008.