❏   2812

Real-time Multi-object Face Recognition Using Content Based Image Retrieval (CBIR)

**Muhammad Fachrurrozi[1], Saparudin[2], Erwin[3], Mardiana[4], Clara Fin Badillah[5],
Junia Erlina[6], Auzan Lazuardi[7]**

[1,2,5,6,7]Informatics Engineering Department, Faculty of Computer Science, Universitas Sriwijaya, Indonesia
[3]Computer Engineering Department, Faculty of Computer Science, Universitas Sriwijaya, Indonesia
[4]Law Department, Faculty of Law, Universitas Sriwijaya, Indonesia

Article Info	ABSTRACT
	Face recognition system in real time is divided into three processes, namely feature extraction, clustering, detection, and recognition. Each of these stages uses different methods, Local Binary Pattern (LBP), Agglomerative Hierarchical Clustering (AHC) and Euclidean Distance. Multi-face image search using Content Based Image Retrieval (CBIR) method. CBIR performs image search by image feature itself. Based on real time trial results, the accuracy value obtained is 61.64%.

Corresponding Author:

Muhammad Fachrurrozi,
Informatics Engineering Department, Faculty of Computer Science,
Universitas Sriwijaya, Indralaya, Indonesia.
Email: mfachrz@unsri.ac.id

1. INTRODUCTION

Face recognition on multi-object imagery can be identified and recognized. The face is one of human biological information in addition to fingerprints, eyes and voice. The process of the introduction of multi-object face image in real-time detection process begins with the face, then tracking to determine the location of the face to be recognized [1]. Face detection is done by separating the face and background and determining the location, size and number of face objects present in the image [2]. The detected face is extracted using Local Binary Pattern (LBP). Vector feature extraction results with trainer image is classified using Hierarchical Agglomerative Clustering (AHC), then compared with the feature vector of the test images using the Euclidean distance. This process is based Content Based Image Retrieval (CBIR) due to the search for vector features in the database is done in accordance with the visual content of the image itself [3].

Content Based Image Retrieval has several processes to get good results, one of the process is by feature extraction to get the value of each facial feature [4]. The Local Binary Pattern algorithm is capable of characterizing and distinguishing surface textures. Local Binary Pattern has a weakness in extracting the texture of the face, this is due to Local Binary Pattern algorithm to extract all images without distinguishing the face and background [5]. High accuracy can be achieved if the algorithm is implemented on a texture with low variance [6]. Image clustering using Hierarchical Clustering Algorithm can improve the speed and accuracy of image matching on Content Based Image Retrieval [7],[8]. Agglomerative Hierarchical Clustering Methods used in face recognition systems are the Single Linkage, Complete Linkage and Average Linkage.

Test image and training image is extracted using eigenface algorithm. Euclidean Distance plays a role to find the minimum distance between test image and train image, and obtained an accuracy of 94% [9]. changing facial expressions, light and pose differences is one of the main factors that greatly affect the process of facial recognition [10]. such changes may alter the non-convex face shape, self-occlusion and nonlinear form changes which may complicate the classification [11]. In this multi-object face recognition system, the face detection process is done before the face extraction process, so that the features obtained right on the face. Then the feature is identified by finding the distance of Euclidean distance between the test image vector feature and the training image resulting from the feature extraction stage using Local Binary Pattern (LBP).

2. FACE RECOGNITION
2.1. Local Binary Pattern (LBP)

Local Binary Pattern (LBP) represents a pixel which formed by a 3x3 matrix as a comparison between the center pixel and its surrounding pixel which then converted into binary numbers. The comparison assumes that if the surrounded pixel value is greater than the central pixel value than it will be 1 otherwise 0. After we get 8 binary numbers in each pixel then it will be replaced with the decimal form to get the result. The LBP algorithm equation can be expressed as the following formula:

$$LBP(x_c, y_c) = \sum_{p=0}^{7} f(g_p - g_c)2^p \tag{1}$$

where: g_p: central pixel value

   g_c : the pixel value around the center
   p: number of pixels around the center
And the function f (x) is defined as follows:

$$f(x) = \begin{Bmatrix} 1, x \geq 0 \\ 0, x < 0 \end{Bmatrix} \tag{2}$$

2.2. Hierarchical Clustering

Hierarchical clustering is used to group similar imagery into clusters to increase the speed of searching image [2]. In this algorithm, there is a hierarchal tree which provides a view of several levels of data abstraction known as dendogram [3]. In Hierarchical Clustering, there are two ways to cluster the data, namely agglomerative and divisive. Agglomerative clustering process based on the amount of data grouped into some hierarchy, it will be clustered into a hierarchical unit. Divisive performs the reverse process of agglomerative, that is cluster one hierarchy into several hierarchies.

Agglomerative Hierarchical clustering is a clustering algorithm based on the proximity distance between two images into a hierarchy. This process repeats itself until it gets some hierarchy. The hierarchy with the closest distance is combined into one hierarchy. The proximity to the new hierarchy then recalculated and the closest hierarchy is merged again. The process is repeated until all the data (object) clustered into one hierarchy. Calculating the spacing between two images using the Manhattan Distance formulated in equation 3:

$$d = \sum_{i=1}^{n} |u_i - v_i| \tag{3}$$

where: d : the distance between the image of u and v.

   n : number of variables.

   u_i : the value of u on i variable.

   v_i : the value of v on i variable

The distance between images is written into a matrix called distance matrix. In order to determine the distance between the two clusters, Agglomerative Hierarchical clustering has 3 methods of grouping data, namely:
a. Single Linkage

Single Linkage classifies data based on the closest distance (Min) between the hierarchy. Single Linkage can be formulated in equation 4:

$$d(uv)w = Min\big[d(uw), d(vw)\big] \tag{4}$$

where: u : the image of u

v : the image of v

w : the image of w

$d(uv)w$: the distance between the hierarchy uv and w

$d(uw)$: the distance between the hierarchy u and w

$d(vw)$: the distance between the hierarchy v and w

b. Complete Linkage

Complete Linkage categorizes data by the furthest distance (Max) or the maximum distance between hierarchies. Complete Linkage can be formulated in equation (5):

$$d(uv)w = Max\big[d(uw), d(vw)\big] \tag{5}$$

where: u : the image of u

v : the image of v

w : the image of w

$d(uv)w$: the distance between the hierarchy uv and w

$d(uw)$: the distance between the hierarchy u and w

$d(vw)$: the distance between the hierarchy v and w

c. Average Linkage

Average Linkage classifies data based on the average distance between the hierarchy. Average Linkage can be formulated in equation (6):

$$d(uv)w = \frac{d(uw) + d(vw)}{2} \tag{6}$$

where: u : the image of u

v : the image of v

w : the image of w

$d(uv)w$: the distance between the hierarchy uv and w

$d(uw)$: the distance between the hierarchy u and w

$d(vw)$: the distance between the hierarchy v and w

2.3. Euclidian Distance

Euclidean distance is an algorithm to calculate the difference or minimum distance between the test image and training image. An Object that will be used as a recognition object is the one which has the minimum distance. To calculate the minimum distance between the vector value of the test image and the training image present in the database, the Euclidean distance algorithm equation can be expressed by the formula:

$$J(A,B) = \sqrt{\sum_{e=1}^{p}\big(a_e - b_e\big)^2} = \|A - B\| \tag{7}$$

which: $J(A,B)$: the distance between imagery A and B

A : the value of the image vector present in the database

B : value of the test image vector

P : number of variables

a_e : value a on the e-variable

b_e : value b on the e-variable

3. METHODOLOGY

Figure 1 is a block diagram of a multi-object face recognition system based on image retrieval, where the path of the system is divided into 2 stages namely the phase of data capture phase of face recognition.
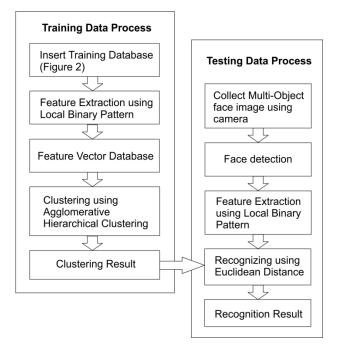
Training Data Process

Insert Training Database (Figure 2)

Feature Extraction using Local Binary Pattern

Feature Vector Database

Clustering using Agglomerative Hierarchical Clustering

Clustering Result

Testing Data Process

Collect Multi-Object face image using camera

Face detection

Feature Extraction using Local Binary Pattern

Recognizing using Euclidean Distance

Recognition Result

Figure 1. General system diagram

In general, the steps of data retrieval in this research is as follows:
a. Collect face image.
b. The feature extraction process using LBP method to get the characteristic of face image then transformed into vector feature form which will be stored in the database.
c. Then do the clustering process using the AHC method on the vector of the face image in the database. Clustering results that have been obtained later used as a comparator value of calculating the distance for face recognition.

In the face recognition phase the process of the system is as follows:
a. Open webcam to detect faces. The process of face detection and recognition is done in real-time.
b. The detected face then captured.
c. The feature extraction process is done using Local Binary Pattern method to get the feature of the face image then transformed into the vector feature form.

Then, do the process of recognition by calculating the distance between the new face image features and features of the existing on the database by using Euclidian distance which then matched with the clustering results.

4. IMPLEMENTATION AND RESULT

This study uses dataset as many as 50 images from 5 people, each taken 10 images. Images are taken from different sides, but with the same background. Face image determined still visible both eyes, nose and mouth. The image used as training data has dimensions of 150x150 pixels. Examples of face image data can be seen in Figure 2.

Figure 2. Example of image face data

Face image recognition done by comparing the distance from the test vector image with training image feature value using Euclidean. The smallest distance of vector feature between the test image and the training image will determine the outcome of the recognition. Figure 3 is a multi-object face image recognition form.
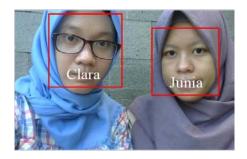

Figure 3. Result of face image recognition

Measurement accuracy rate is obtained by comparing the value of the number of objects to be identified and the number of objects contained in the test images. The level of accuracy of face recognition in realtime is smaller than facial recognition in non real time. On the introduction in on realtime with the same data and methods obtained an accuracy of 65.32% [4].

The research "A Face Recognition System Based on Eigenfaces Method", the eigenface with the smallest Euclidian distance is the one the person resembles the most. Simulations have been done using the Matlab program. The success rate for the large database used is found to be 94.74% [7]. Recognizing face on ARM processor using K-Nearest Neighbor (KNN) algorithm. This research sought best k-value to create proper face recognition with low-power processor. The proposed algorithm was tested on three datasets that were Olivetti Research Laboratory (ORL), Yaleface and MUCT. OpenCV was chosen as main core image processing library, due to its high-speed. Proposed algorithm was implemented on ARM11 700MHz. 10-fold cross-validation showed that KNN face recognition detected 91.5% face with k=1. Overall experiment showed that proposed algorithm detected face on 2.66 s on ARM processor [10].

Table 1. Testing of Real-time Face Recognition

No.	Test image	Recognized object	Accuracy (%)	Time execution (second)
1.	2 objects on image-1	1	50	0,0750
2.	2 objects on image-2	1	50	0,0740
3.	2 objects on image-3	2	100	0,0730
4.	2 objects on image-4	1	50	0,0720
5.	2 objects on image-5	2	100	0,0710
6.	3 objects on image-1	1	33.3	0,0730
7.	3 objects on image-2	2	66.6	0,0730
8.	3 objects on image-3	2	66.6	0,0720
9.	3 objects on image-4	2	66.6	0,0720
10.	3 objects on image-5	1	33.3	0,0710

5. CONCLUSION

Multi-object face recognition system can recognize single or multi object in real-time with an accuracy of 61.64%. The feature extraction process on the input image plays an important role in determining the success rate of face image recognition. The recognizable testing process has the same lighting, distance and other effects during training.

REFERENCES

[1] K. Chen and L. J. Zhao, "Robust Realtime Face Recognition and Tracking System," vol/issue: 9(2), pp. 82-88, 2009.

[2] C. Lin, *et al.*, "Gabor Filters and Feature Fusion," vol/issue: 11(10), pp. 5986-5994, 2013.

[3] D. H. Z. and P. D. D. F. Fuhui Long, "Fundamentals of Content-Based Image Retrieval," *Multimed. Inf. Retr. Manag. Technol. Fundam.*, pp. 1-26, 2003.

[4] M. Fachrurrozi, "Multi-Object Face Recognition Using Content Based Image Retrieval (CBIR)," vol. x, pp. 193-197, 2017.

[5] T. Ahonen, *et al.*, "Face Description with Local Binary Patterns: Application to Face Recognition," vol/issue: 28(12), pp. 2037-2041, 2006.

[6] V. S. V. S. Murthy, *et al.*, "Content Based Image Retrieval using Hierarchical and K-Means Clustering Techniques," *Int. J. Eng. Sci. Technol.*, vol/issue: 2(3), pp. 209-212, 2010.

[7] M. üg. Çarıkçı and F. Özen, "A Face Recognition System Based on Eigenfaces Method," *Procedia Technol.*, vol. 1, pp. 118-123, 2012.

[8] A. Katare, *et al.*, "Content Based Image Retrieval System for Multi Object Images using Combined Features," 2007.

[9] N. Alajlan, *et al.*, "Multi-object image retrieval based on shape and topology," *Signal Process. Image Commun.*, vol/issue: 21(10), pp. 904-918, 2006.

[10] E. Setiawan and A. Muttaqin, "Implementation of K-Nearest Neighbors Face Recognition on Low-power Processor," vol/issue: 13(3), 2015.

[11] M. E. Wibowo, *et al.*, "Improved Face Recognition across Poses using Fusion of Probabilistic Latent Variable Models," vol/issue: 15(4), pp. 1976-1986, 2017.