Register 6 (1) 2020 pp. 65-73
Jurnal Ilmiah Teknologi Sistem Informasi (Scientific Journal of Information System Technology)

ISSN 2502-3357 (Online)
ISSN 2503-0477 (Print)

Available online to www.journal.unipdu.ac.id

Unipdu

S2-Accredited – SK No. 34/E/KPT/2018
Journal page is available to www.journal.unipdu.ac.id:8080/index.php/register

Combination of fast hybrid classification and *k* value optimization in k-nn for video face recognition

Nuning Septiana [a], *Nanik Suciati* [b]

[a,b] *Department of Informatic Engineering, Institut Teknologi Sepuluh Nopember, Surabaya, Indonesia*

email: [a] *nuning13@mhs.if.its.ac.id* , [b] *nanik@if.its.ac.id*

ARTICLE INFO

Article history:
Received 27 July 2019
Revised 3 August 2019
Accepted 21 August 2019
Published 6 April 2020

Keywords:
face recognition
Fast Hybrid Classification
k-NN
video

IEEE style in citing this article:
N. Septiana and N. Suciati, "Combination of fast hybrid classification and k value optimization in k-nn for video face recognition," *Register: Jurnal Ilmiah Teknologi Sistem Informasi*, vol. 6, no. 1, pp. 65-73, 2020.

ABSTRACT

Nowadays, the need for face recognition is no longer include images only but also videos. However, there are some challenges associated with the addition of this new technique such as how to determine the right pre-processing, feature extraction, and classification methods to obtain excellent performance. Although nowadays the k-Nearest Neighbor (k-NN) is widely used, high computational costs due to numerous features of the dataset and large amount of training data makes adequate processing difficult. Several studies have been conducted to improve the performance of k-NN using the FHC (Fast Hybrid Classification) method by optimizing the local *k* values. One of the disadvantages of the FHC Method is that the *k* value used is still in the default form. Therefore, this research proposes the use of k-NN value optimization methods in FHC, thereby, increasing its accuracy. The Fast Hybrid Classification which combines the k-means clustering with k-NN, groups the training data into several prototypes called TLDS (Two Level Data Structure). Furthermore, two classification levels are applied to label test data, with the first used to determine the *n* number of prototypes with the same class in the test data. The second classification using the optimized *k* value in the k-NN method, is employed to sharpen the accuracy, when the same number of prototypes does not reach *n*. The evaluation results show that this method provides 86% accuracy and time performance of 3.3 seconds.

1. Introduction

The advancement in technology has made data collection for various purposes such as video data collection easier. As the speed in internet service becomes higher and more accessible, there is a huge demand for video data yearly, such as for security purpose. More than 45% of organizations across the worldwide use CCTV (Closed-Circuit Television) to easily supervise and monitor their employers, clients and strangers. Another research also stated that some criminal cases are solved through its usage [1]. Unfortunately, supervisors often find it difficult to recognize the person on the video recording, owing to blur quality, low resolution, position, and poor lighting. Therefore, a system capable of recognizing and correcting the faces from video data captured with CCTV using data features and classification algorithm is needed.

Many research on Video Face Recognition (VFR) has been conducted to solve this problem using various methods such as Support Vector Machine (SVM) [2, 3] and modified Neural Networks [4, 5, 6, 7]. However, some studies show that the VFR study using k-NN such as Local Mean-based k-Nearest Centroid Neighbour (LMKCN) which creates the centroid of each class and queried as the closest neighbour, has many opportunites for improvement [8].

Another research also used k-NN with Discrete Cosine Transform (DCT) as the features. The DCT features were utilized because it proved to have better performance than other data features [9].

N. Septiana and N. Suciati./Register 6 (1) 2020 pp. 65-73
ISSN 2502-3357 (Online)
ISSN 2503-0477 (Print)

Unfortunately, [9] it shows slower performance and its accuracy per frame is less than 80% with high computational cost. Therefore, this paper proposed another modified k-NN methods for video face recognition with the optimized k value known as the Fast Hybrid Classification (FHC) and DCT to increase accuracy and reduce computational cost.

2. Research Method

Research on Video Face Recognition (VFR) has been previously carried out on numerous occasions to solve various challenges using joint sparse representation [10]. It considers the face image of VFR as a set formulated, where two crucial things are obtained namely the representation of integrity and sparsity. This method was tested using three real-world video datasets and compared with methods that utilize set images such as Mutual Subspace Method, Discriminant Canonical Correlation Analysis (DCC), and Manifold-to-Manifold Distance (MMD).

Furthermore, another VFR study was conducted using bag of features and multi-class Support Vector Machine (SVM) [4]. However, this study describes the facial recognition system in videos with more than one face. Firstly, bag of words extraction was performed using the SURF (Speeded Up Robust Features) algorithm. Secondly, the extraction results were classified using multi-class SVM and tested with dataset Face-95 and Face-96. The proposed method was compared with the PCA (Principal component analysis) approach.

VFR studies were also developed by Witham et al. using Local Binary Pattern (LBP) and Local Discriminant Classification for rhesus macaques [11]. This method was developed under the conditions of training and processing which must be carried out by a standard Personal Computer (PC) in real time. This system was tested using a dataset image of 24 adult macaques under several different conditions.

Besides its extraction features, some researches also focused on classifying the algorithm such as [12] combining fuzzy ARTMAP; an architecture to achieve fuzzy logic and Adaptive Resonance Theory (ART) neural network; and Dynamic Particle Swarm Optimization to provide rapid adaptation when studying new classes used to categorize video. When new data existed, the system automatically adjusted the weights, architecture and parameters to obtain high classification accuracy.

Other studies applying modified Neural Network is Trunk-Branch Ensemble Convolutional Neural Networks [5]. Its algorithm provides a trunk network that studies the face shape of various holistic images and branch networks of a cropped patch images from one facial component. The datasets used in this research include PaSC, Cox Face, and Youtube Face, applied with the neural network tested with other models such as Visual Geometry Group (VGG) Face and the Hybrid Euclidean-and-Riemannian Metric Learning (HERML) algorithm. Another research using deep learning is carried out on face images using principal component neural network (PCNN) and Hebbian algorithm (GHA) [6].

In addition to neural network, the SVM (Support Vector Machine) modification algorithm is also widely applied to VFR such as Hybrid Multiclass SVM [2]. This study uses two different types of One Versus One SVM (OVO-SVM), OVO-SVM max win strategy and OVO-SVM bottom up decision tree for the classification stage. OVO-SVM max win strategy is used to determine the eight face images that are most similar to the input. Meanwhile, the bottom up decision tree is used to predict labels based on these facial images. This technique is compared to VFR which only uses the max win strategy or bottom up decision tree, while other research in face spoofing detection on videos uses KSVM (k-Means and SVM) [3].

Face spoofing is an attempt to get access to people using faces images or videos. This paper used IDA feature (Image Distortion Analysis) on each image/frame (for video) in correlation with k-Means and SVM. The data features were compared to the spoof database to form a ranking list, which acted as KSVM input system. k-Means algorithm performed clustering on data in the list, while SVM train executed the misclassified ones. Finally, SVM classified the image/frame as genuine or fake with its algorithm used in many VFR researches [13].

Research on the k-NN method has been widely carried out, one of which is based on a symmetrical visual grid and dynamic circumference [14]. In this study, the data object is combined with the query object assuming it has a symmetrical grid cell. During the search process, the query object becomes a centroid, while the largest distance in the virtual grid is considered a radius for creating

ISSN 2502-3357 (Online)
N. Septiana and N. Suciati./Register 6 (1) 2020 pp. 65-73 ISSN 2503-0477 (Print)

dynamic circles to determine data objects in the circle. This process is very suitable for data used in Geographic Information Systems.

Other researches focuses on optimizing the value of *k* [15] in each training data for different local and global effects. This technique does not require large computational costs, therefore it has the capability to run fast and produce higher accuracy than the usual k-NN method.

Classifications for large datasets require high computational costs [16]. It introduced a new method namely Fast Hybrid Classification (FHC), where Two Level Data Structure were made using the k-Means algorithm, which further increased the speed to access TLDS. First, each class contains a set of training data with similar class. The set is divided into several clusters with each k-Means formed considered a prototype. Furthermore, each prototype consists of class label and members. Figure 1 show the process of creating TLDS.

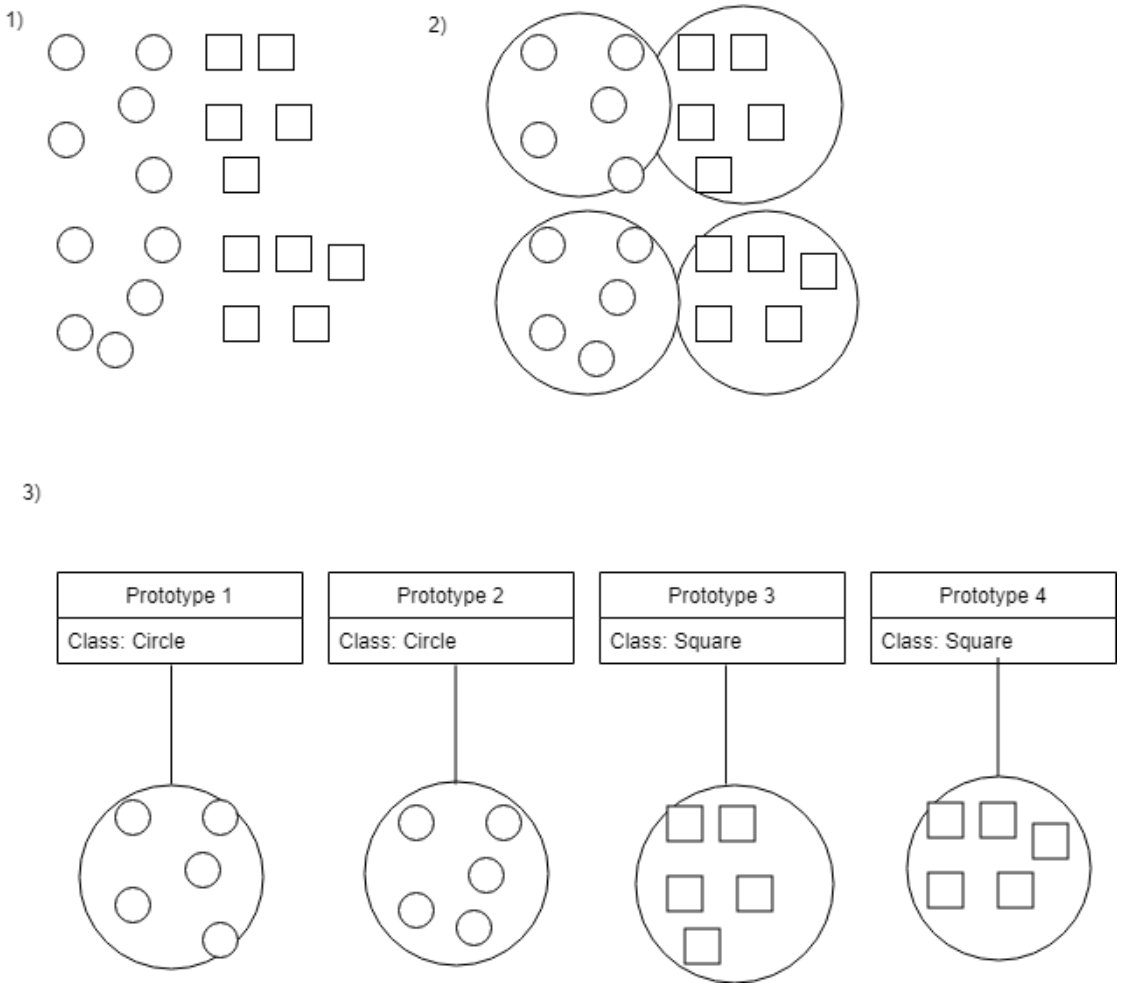

Figure 1. Process obtaining TLDS

FHC, which has been tested with image dataset such as Letter Recognition and Textures, proved to have better performance than k-NN. However, subsequent analysis indicated that it still carries out conventional k-NN. This step is optimized using *k* value optimalization from [15], though both have not been tested using facial images. Therefore, a combination of optimized *k* value for k-NN with FHC as the modified k-NN in video face recognition, is applied to determine the performance.

3. Proposed Methods

Generally, there are three main steps of video face recognition. The first is face detection where CAMShift provided by MATLab is utilized. After the face image were obtained by face detection, the extraction feature of DCT with the ability to give higher accuracy was divided into two, namely the training and classification phases [9]. Furthermore, the training phase was divided into two. The first training phase was used to determine the optimal *k* value of each data, while the other was to create a model called TLDS which was classified using FHC methods. Figure 2 illustrates the flowchart of video

N. Septiana and N. Suciati./Register 6 (1) 2020 pp. 65-73
ISSN 2502-3357 (Online)
ISSN 2503-0477 (Print)

face recognition using both DCT and FHC with the system evaluated by obtaining the accuracy of each video using confusion matrix compared with k-NN.

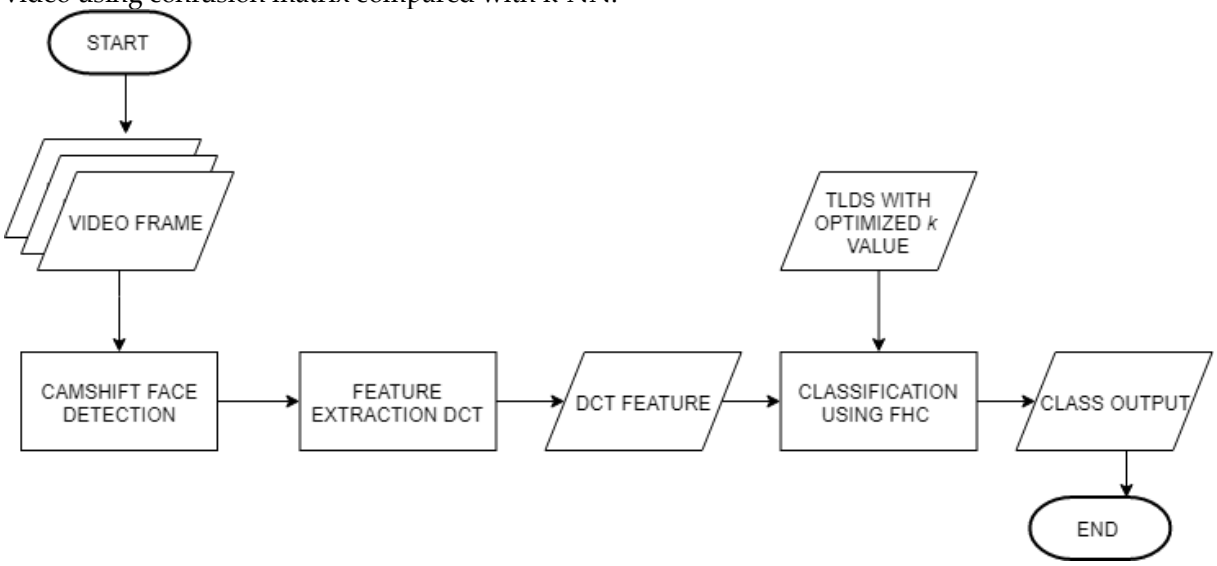

Figure 2. Flowchart system

3.1. Face detection

The face detection was conducted using Continuously Adaptive Meanshift (CAMShift), an algorithm used to determine objects in a video by applying meanshift in each frame. The CAMShift algorithm is available in MATLAB Library and has been widely used in various types of research due to its ease of applying algorithms [17]. Some studies use it for learning local features of face recognition online [18], hand gesture recognition [19], and player tracking system [20]. Figure 3 is an example of face detection in video using CAMShift.

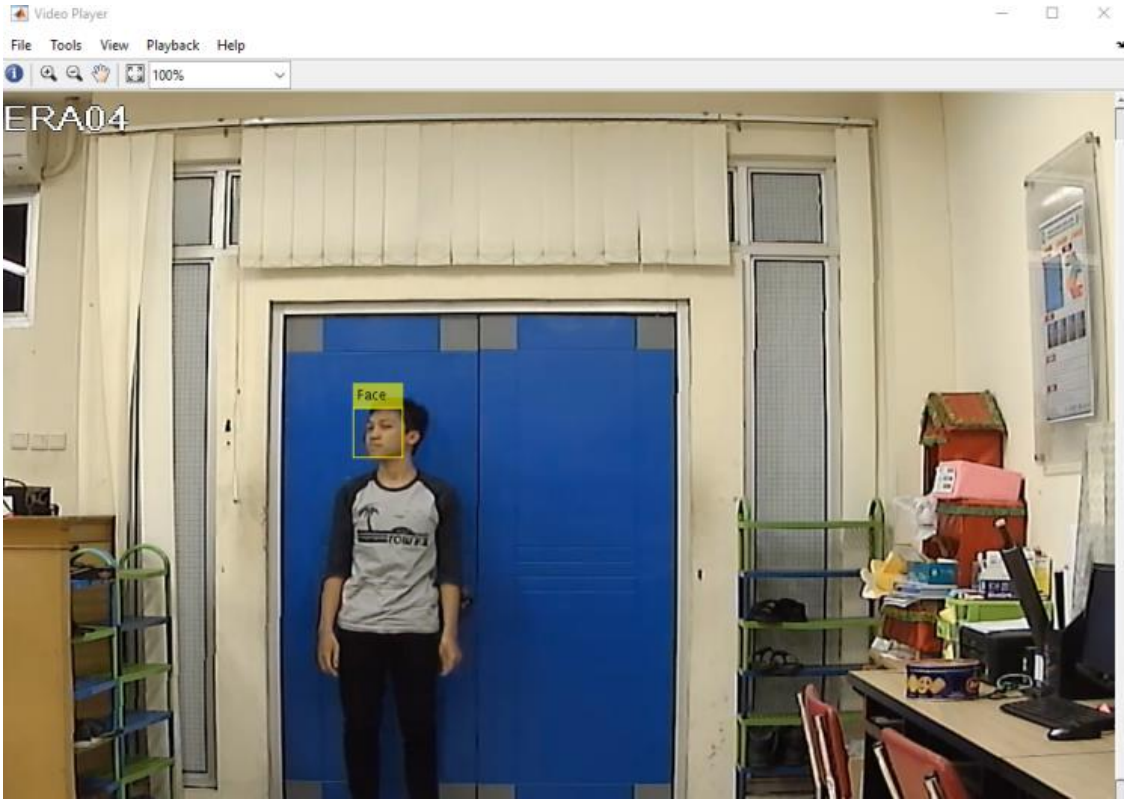

Figure 3. Face detection with CAMShift

3.2. Features extraction using DCT

This paper used DCT or Discrete Cosine Transform to extract its features, using similarities with Discrete Fourier Transform, which converts signals or images from the spatial to frequency domain.

http://doi.org/10.26594/register.v6i1.1668

ISSN 2502-3357 (Online)
N. Septiana and N. Suciati./Register 6 (1) 2020 pp. 65-73 ISSN 2503-0477 (Print)

DCT divides images into several parts with different quality levels [21]. It is applied in various fields such as error control [22], improve image quality [23] and increase the efficiency of video coding [24, 13].

Research on facial recognition for real-life applications shows that images using DCT in feature extraction, obtain better results than those that utilize other features such as Fourier, Wavelet, and Wash-hadamard. Below are the steps to extract DCT features:

1. Change the image size to 64 × 64 pixels
2. Divide images into blocks with size 8 × 8 pixel
3. Calculates the DCT coefficient matrix for each block
4. Sort the DCT coefficient matrix by zigzag scanning
5. Select the first five AC coefficients as local feature vectors from each block

In extracting its features for face images, the coefficients were not considered because they are representations of image intensity, not cosine waves. The DCT coefficient was calculated by equations 1 and 2:

$$F(u,v) = \left(\frac{2}{N}\right)^{\frac{1}{2}}\left(\frac{2}{M}\right)^{\frac{1}{2}}\sum_{i=0}^{N-1}\sum_{j=0}^{M-1}\alpha_i\alpha_j cos\left[\frac{\pi.u}{2.N}(2i+1)\right]cos\left[\frac{\pi.v}{2.M}(2j+1)\right].f(i,j) \quad (1)$$

$$\alpha_p = \begin{cases} \frac{1}{\sqrt{2}}, & p = 0 \\ 1, & otherwise \end{cases} \quad (2)$$

Where:

- The input image is $N \times M$
- $f(i,j)$ is the pixel intensity of row i column j
- $F(u,v)$ is the result of the transformation to the frequency domain coordinates u and v in the form of the DCT coefficient matrix

3.3. Combining fast hybrid classification and optimization k value for k-NN

FHC is a modified k-NN which utilizes centroids created using k-means as their model instead of data training. It is divided into two main phases. The first is training phase, which creates the centroids called Two-Level Data Structure (TLDS). It is implemented to determine the optimal k value for each data. The second is the classification process which is explained in more detail below.

1st training phase (obtaining optimal k value)

This process adds the optimal k value to the training data, which is used during the testing phase to access the FHC. This process changes the set of data training T such as $T = \{(x_1, y_1), ..., (x_n, y_n)\}$) into $T = \{(x_1, y_1, k_1), .., (x_n, y_n, k_n)\ \})$. Where x_n is the n-th value, y_n is the label, and k_n is the optimized k value.

Besides data training, this process also uses the range of k values (kmin and kmax) as its parameter. The kmin value starts at 3 while the kmax is 25. The steps to determine the k value are as follows:

1. Obtain a global performance value by calculating the average accuracy of each k value with 10-fold Cross-Validation using training data.
2. For each training data i
   For each k value
   a. Find k closest objects to the data training i
   b. Obtain a local performance value by calculating the percentage of the objects which have the same class with that of the data training.
   c. Get the value of eval (k) by adding the local and global performance.
3. Choose the k value which has the highest eval (k)
4. Associate optimal k values in data training i

2nd training phase (obtaining tlds)

In this phase, the system makes TLDS from the updated data training as the model for the classification phase. It contains some prototypes with each consisting of class attributes and member vectors. Below are the steps in creating TLDS:

ISSN 2502-3357 (Online)
N. Septiana and N. Suciati./Register 6 (1) 2020 pp. 65-73 ISSN 2503-0477 (Print)

1. Preparing TLDS empty structure data (prototype name, class name, member vector list).
2. For each class c
   a. Take all training data from with class c (sc).
   b. Obtain a k value for kmeans (nc), where $nc = |\frac{sc}{DRF}|$ then round it up.
   c. Obtain a cluster of nc numbers from the k-means process.
   d. For each CL cluster:
      Add a TLDS element that contains information on centroid values, data sets, and class names.
3. TLDS is successfully created.

Classification phase

When there was new data, the system checked the n closest prototype. Assuming from all n prototypes, more m that belongs to the same class originated, then the data was labeled A. However, assuming it was less than n, it went through the second search or k-NN process. The variable of m was dependent on the npratio, while n was equal to minimal number of same-class prototype. For example, class A has the least number prototype which is 10. Therefore, the value of n equals 10 assuming its value is set to npratio 0.7 This means that there are at least 7 in 10 nearest prototype that belong to class A.

When the new data moved to the second search, the system determined the nearest neighbour or 1-NN and checked the optimal k value associated to the nearest neighbour. Furthermore, it processed the k-NN query when k was equal to the optimal value.

4. Experiment and Results

This research utilizes dataset videos obtained from CCTV cameras in the KCV lab, ITS Informatics Engineering department. In each video, people coming in from the laboratory door were captured before they disappeared from the camera. In this dataset, eight classes consisting of 45 videos were classified into training and test data. All videos were in mp4 format and a resolution of 720 pixels. In addition, facial detection was conducted using the Camshift method. Figure 4 shows the result of sample face image.

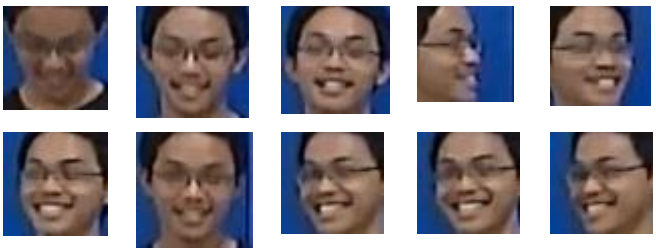

Figure 4. Sample image of face images

Each face image was extracted to obtain the DCT features using a total of 320 number of features. The proposed method was compared with conventional k-NN and modified FHC (without second search) technique with the accuracy calculated using confusion matrix. Table 1 displays the result of overall performance.

Table 1. Overall performance

Performance	Methods				
	3-NN	Optimized k-NN	Edited FHC (without k-NN)	FHC	FHC+Optimized k-NN
Overall accuracy (%)	85	85.5	82	82.8	86
Time execution (sec)	4.4	5.3	1.14	2.4	3.3

FHC+Optimized k-NN shows the best performance eventhough it is not the fastest, while FHC is faster than 3-NN with slightly lower accuracy. Furthermore, FHC used 3-NN as the second search assuming the first search showed lower threshold or the data testing obtained lower npratio. The accuracy of each video was also calculated, with each containing one label. The accuracy was calculated by the amount of face images (frame) which was classified into the actual label, according to Table 2. The lowest accuracy by FHC comes from Video 2 which only has 58%. In addition, Video 2, Video 4, and Video 6 shows that the FHC's accuracy is lower than 3-NN with gap around 12-13% due to the use of more data training by the 3-NN. The FHC generalizes the data training into TLDS. The difficulty,

ISSN 2502-3357 (Online)
N. Septiana and N. Suciati./Register 6 (1) 2020 pp. 65-73 ISSN 2503-0477 (Print)

associated with generalizing the data training occures because it comes with 320 number of features which is considered large.

However, assuming FHC is enhanced with optimizied *k* value, the overall accuracy becomes the highest. For instance, in Table 2, Video 2 and Video 4 increased faster than others because when the second searching occurred in the system, the optimized *k* value used the default value in FHC. This proves that choosing the right value of *k* improves the accuracy of classification system. Conversely, from video-based perspective, all of these methods have an accuracy of 100% and 50% per frames in the video.

Apart from the amount used to create the computational cost, a little was for the TLDS which was twice the FHC and lower than 3-NN in the testing phase. FHC with no k-NN even costed four times lower, while FHC+Optimized k-NN was slower. This was because when data testing went through second searching, the system executed the k-NN processing twice. First was 1-NN to determine the associated *k* value of the nearest data training. Second was k-NN processing using the *k* value.

From this result, the VFR system combined with the FHC optimized *k* value has better performance than k-NN and FHC due to its ability to produce good accuracy under limited time.

Table 2. Individual recognition rates per frame

Video	Accuracy (%)				
	3-NN	Optimized k-NN	Edited FHC (without *k*-NN)	FHC	FHC+Optimized *k* value
Video 1	**95.4**	95	93	94.3	94.3
Video 2	**71.6**	71.	59.2	58	71.6
Video 3	97	95	96	96	**98**
Video 4	80	80	67.5	67.5	**80.8**
Video 5	87.2	87.5	**100**	**100**	92.7
Video 6	**97.3**	97.3	71	84.2	94.7
Video 7	74.3	75	82	82	**87.5**
Video 8	87.5	87.5	**92.1**	**92.1**	86.7

5. Conclusions

In this paper, a combination of optimized *k* value for k-NN with fast hybrid classification and discrete cosine transform as features for face recognition based on video was proposed. Fast hybrid classification is a modified k-NN algorithm which produces centroids from data training as its new neighbors. The *k* value is optimized to choose its ideal value during k-NN processing. Discrete Cosine Transform was implemented as a feature used to extract video, with one frame of cropped face culminating in 320 features. k-NN is a painful method assuming the system had to compare all data with hundreds of features.

The experimental results show that the FHC method with optimized *k* value provides the highest accuracy of 86% even though the difference with other methods is not too high. This is because k-NN optimization process is able to detect the right *k* value for each data testing to provide better accuracy. In terms of computational costs, the proposed method is faster than k-NN but slower than the FHC which is 3.3 second. In the future, this research will be improved by implementing selection feature to decrease the number of optimized FHC algorithm that modified the clustering method to obtain better grouping results.

6. References

[1] H. Idrees, M. Shah and R. Surette, "Enhancing camera surveillance using computer vision: a research note," *Policing: An International Journal*, vol. 41, no. 2, pp. 292-307, 2018.

[2] M. H. Selamat and H. M. Rais, "Image face recognition using Hybrid Multiclass SVM (HM-SVM)," in *International Conference on Computer, Control, Informatics and its Applications (IC3INA)*, Bandung, Indonesia, 2015.

ISSN 2502-3357 (Online)
ISSN 2503-0477 (Print)

[3] T. Faseela and M. Jayasree, "Spoof Face Recognition in Video Using KSVM," *Procedia Technology,* vol. 24, pp. 1285-1291, 2016.

[4] A. Rikhtegar, M. Pooyan and M. T. Manzuri-Shalmani, "Genetic algorithm-optimised structure of convolutional neural network for face recognition applications," *IET Computer Vision,* vol. 10, no. 6, pp. 559-566, 2016.

[5] C. Ding and D. Tao, "Trunk-Branch Ensemble Convolutional Neural Networks for Video-Based Face Recognition," *IEEE Transactions on Pattern Analysis and Machine Intelligence,* vol. 40, no. 4, pp. 1002-1014, 2018.

[6] N. Sudha, A. R. Mohan and P. K. Meher, "A Self-Configurable Systolic Architecture for Face Recognition System Based on Principal Component Neural Network," *IEEE Transactions on Circuits and Systems for Video Technology,* vol. 21, no. 8, pp. 1071-1084, 2011.

[7] G. Goswami, M. Vatsa and R. Singh, "Face Verification via Learned Representation on Feature-Rich Video Frames," *IEEE Transactions on Information Forensics and Security,* vol. 12, no. 7, pp. 1686-1698, 2017.

[8] S. Damavandinejadmonfared, "Kernel Entropy Component Analysis using local mean-based k-nearest centroid neighbour (LMKNCN) as a classifier for face recognition in video surveillance camera systems," in *IEEE 8th International Conference on Intelligent Computer Communication and Processing*, Cluj-Napoca, Romania, 2012.

[9] B. Ko, J.-H. Jung and J.-Y. Nam, "View-independent object detection using shared local features," *Journal of Visual Languages & Computing,* vol. 28, no. June, pp. 56-70, 2015.

[10] Z. Cui, H. Chang, S. Shan, B. Ma and X. Chen, "Joint sparse representation for video-based face recognition," *Neurocomputing,* vol. 135, no. July, pp. 306-312, 2014.

[11] C. L. Witham, "Automated face recognition of rhesus macaques," *Journal of Neuroscience Methods,* vol. 300, no. April, pp. 157-165, 2018.

[12] J.-F. Connolly, E. Granger and R. Sabourin, "An adaptive classification system for video-based face recognition," *Information Sciences,* vol. 192, no. June, pp. 50-70, 2012.

[13] S. M. K. Hasan and M. Ahmad, "A new approach of sign language recognition system for bilingual users," in *International Conference on Electrical & Electronic Engineering (ICEEE)*, Rajshahi, Bangladesh, 2015.

[14] G. Li and J. Tang, "A new K-NN query algorithm based on the symmetric virtual grid and dynamic circle," in *International Conference on Artificial Intelligence and Education (ICAIE)*, Hangzhou, China, 2010.

[15] N. García-Pedrajas, J. A. R. d. Castillo and G. Cerruela-García, "A Proposal for Local k Values for k -Nearest Neighbor Rule," *IEEE Transactions on Neural Networks and Learning Systems,* vol. 28, no. 2, pp. 470-475, 2017.

[16] S. Ougiaroglou and G. Evangelidis, "Efficient k-NN classification based on homogeneous clusters," *Artificial Intelligence Review,* vol. 42, p. 491–513, 2014.

[17] F. Li, R. Zhang and F. You, "Fast pedestrian detection and dynamic tracking for intelligent vehicles within V2V cooperative environment," *IET Image Processing,* vol. 11, no. 10, pp. 833-840, 2017.

[18] A. Mian, "Online learning from local features for video-based face recognition," *Pattern Recognition,* vol. 44, no. 5, pp. 1068-1075, 2011.

[19] A. S. Kundu, O. Mazumder, P. K. Lenka and S. Bhaumik, "Hand Gesture Recognition Based Omnidirectional Wheelchair Control Using IMU and EMG Sensors," *Journal of Intelligent & Robotic Systems,* vol. 91, p. 529–541, 2018.

[20] M.-C. Hu, M.-H. Chang, J.-L. Wu and L. Chi, "Robust Camera Calibration and Player Tracking in Broadcast Basketball Video," *IEEE Transactions on Multimedia,* vol. 13, no. 2, pp. 266-279, 2011.

[21] W.-G. Chen, X. Wang and Y. Tian, "A two-stage algorithm for the early detection of zero-quantized discrete cosine transform coefficients in High Efficiency Video Coding," *EURASIP Journal on Image and Video Processing,* vol. 2017, p. 56, 2017.

ISSN 2502-3357 (Online)
N. Septiana and N. Suciati./Register 6 (1) 2020 pp. 65-73 ISSN 2503-0477 (Print)

[22] I. Natgunanathan, Y. Xiang, G. Hua, G. Beliakov and J. Yearwood, "Patchwork-Based Multilayer Audio Watermarking," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 11, pp. 2176-2187, 2017.

[23] L. Yuan-Yuan, C. He-Xin, Z. Yan and S. Hong-Yan, "Discrete cosine transform optimization in image compression based on genetic algorithm," in *8th International Congress on Image and Signal Processing (CISP)*, Shenyang, China, 2015.

[24] M. Abdelrasoul, M. S. Sayed and V. Goulart, "Real-time unified architecture for forward/inverse discrete cosine transform in high efficiency video coding," *IET Circuits, Devices & Systems*, vol. 11, no. 4, pp. 381-387, 2017.