Image Processing in Automatic Locking System on SS2 TNI AD Weapon Rack Using Fingerprint Sensor with K-Nearest Neighbor (KNN) Algorithm Method

Rafi Maulana Al-farizi[1], Indrazno Siradjuddin[2] and Sapto Wibowo [3]

[1] Electrical Engineering, State Polytechnic of Malang, Malang, Indonesia
[2] Electrical Engineering, State Polytechnic of Malang, Malang, Indonesia
[3] Electrical Engineering, State Polytechnic of Malang, Malang, Indonesia
rafi.maulana0048@gmail.com

Abstract. Weapon security is crucial in military environments. An automatic weapon rack locking system using fingerprint sensors can be an effective solution to prevent unauthorized access. However, in order to implement an automatic locking system an authorizing method using fingerprint is required. In the image processing stage, fingerprint images obtained from the sensor are processed to extract important features, such as core points and unique ridge orientations in the fingerprint. The K-Nearest Neighbor (KNN) algorithm compares the unknown fingerprint features with the fingerprints in the database to be classified. In the conducted tests, the image identification speed using 10 images was found to be 0.46 seconds, while it was 1.72 seconds for 50 images. Furthermore, the system achieved a False Acceptance Rate (FAR) of 0% and a False Rejection Rate (FRR) of 8% using an optimal Parameter K. The success rate reached 93%, with an accuracy level of 96%. Thus, optimal processing speed and accuracy were achieved in this automatic locking system. Consequently, this system can be effectively implemented to enhance weapon security in military environment.

Keywords: Fingerprint Sensor, Image Processing and K-Nearest Neighbor (KNN).

1 Introduction

Weapon security is a crucial aspect in military contexts[1]. However, the process of securing weapons is still done manually. In this process, all weapons on the weapon rack are locked using a long chain by inserting the chain into each gun's stock hole. When a weapon needs to be borrowed or used, the warehouse sergeant, with the approval of the warehouse head, retrieves the weapon by unlocking the chain from the gun, pulling them out one by one. This process takes time and involves friction between the chain and the weapon itself. Several personnel are provided keys to access the weapon rack security. This method has many weaknesses, one of which is the negligence of the warehouse sergeant who may lose the key due to it falling or being

R. Andrie Asmara et al. (eds.), *Proceedings of the 5th Annual Advanced Technology, Applied Science, and Engineering Conference (ATASEC) 2023*, Advances in Engineering Research 229,
https://doi.org/10.2991/978-94-6463-358-0_8

stolen. There-fore, a better and more modern weapon rack security system is needed. One of the security measures is to rec-ognize the fingerprint pattern of an individual's identity (Biometrics), so that the personnel who take and use the weapons can be monitored by authorized warehouse and staff personnel[2].

Fingerprint detection techniques have been widely studied by researchers using various methods and approaches. Ziqra Hanifah et al. applied the K-Nearest Neighbor (KNN) and Decision Tree methods, which yielded different accuracies, with KNN achieving 85% accuracy and Decision Tree achieving 89% accuracy[3]. Boldson Herdianto S. et al. conducted research on Biometric Identification Using Minutiae Extraction on Fingerprint Images and achieved an accuracy of 92.8% for a 5-megapixel camera and 95.3% for an 8-megapixel camera[4]. Katon A. implemented the Backpropagation Artificial Neural Network method for Fin-gerprint Pattern Classification. The system achieved a classification accuracy of 75.71%. The classification system was implemented in the form of a website application[5]. Novelita Dwi et al. applied the Convolutional Neural Network method using Resnet-50 for Fingerprint Classification, improving the accuracy performance of the fingerprint classification system by 11.79%. The validation accuracy obtained for images without Clahe was 83.26%, while for images with Clahe, the validation accuracy reached 95.05%[6]. Sofyan Saputra applied Fuzzy Mamdani Logic for fingerprint pattern detection using the Canny edge detector algorithm. The testing results using the Universal Image Quality Index showed a pattern detec-tion success rate of 90%, while the Canny edge detection had a success rate of only 30% and the Sobel opera-tor had a success rate of only 6%[7].

Fingerprint recognition is developed because it has advantages such as being unlosable, unforgettable, and unforgeable as it is inherent to humans. Basically, pattern recognition is used to solve problems, especially related to fingerprint classification through the characteristics and features of the fingerprints. One of the issues is that fingerprint characteristics have a high degree of similarity, requiring a very high level of accuracy for the pattern recognition of fingerprints themselves[8].

However, one of the challenges is that fingerprint characteristics themselves have a high degree of similarity. Thus, a high level of accuracy is required for the pattern recognition of fingerprints[9]. Therefore, a process for high-accuracy fingerprint pattern recognition using appropriate methods is needed. The method used is image processing on fingerprints, which plays a vital role in ensuring the accuracy and reliability of the fingerprint identification or verification process. These image processing techniques help optimize important features and reduce noise or disturbances that can affect fingerprint recognition results, along with the K-Nearest Neighbor (K-NN) algorithm for object classification based on the data process closest to the object itself.

2 Proposed Method

The K-Nearest Neighbor (K-NN) method in fingerprint image processing is used to classify new fingerprint im-ages based on their similarity to fingerprint images in a dataset. Next, K nearest fingerprint images with the smallest distances from the new fingerprint image are selected as the nearest neighbors. Based on the majority class label of these nearest neighbors, the class label of the new fingerprint image is determined to classify it into the most frequently occurring class.

2.1 Fingerprint Sensor

Fingerprint sensor is a piece of hardware that is used to collect and analyze unique patterns found on the skin surface of the human fingertip. This sensor works by recording an image or fingerprint pattern which is then converted into digital data that can be processed by a computer system [10].
Each individual has a unique fingerprint pattern, which is formed by rows of lines, patterns of peaks, and fissures on the skin surface. The fingerprint sensor works by taking an image or "recording" the fingerprint pat-tern using various methods such as optical, capacitance, or ultrasound. Fingerprint sensors have advantages in recognition speed, high security, and a good level of accuracy. This makes it one of the most commonly used biometric technologies in various security and identity authentication applications[11].

2.2 Image Processing

Image processing is a method or technique that can be used to process images or images by manipulating them into an image data that is loaded to obtain certain information about the object being observed. this prepro-cessing process consists of stages namely Cropping, Thresholding, Thinning and Resizing[12].

- Cropping

Cropping in image processing means cutting one part of the image so that the expected image is obtained. The size of the cropped image changes according to the size of the image taken. Cropping is done at coordinates (x, y) to coordinates (m, n)[13].

- Thresholding

Thresholding is the conversion from a black-and-white image to a binary image carried out by a thresholding operation[14].

$$f_B(i,j) = \begin{cases} 1, & f_g(i,j) < T \\ 0, & other \end{cases} \tag{1}$$

Where $f_B(i,j)$ represents a binary image, $f_g(i,j)$ represents a grayscale image, and T denotes the threshold value.

2.3 K-Nearest Neightbor (K-NN)

One of the most frequently used classification methods is the K-Nearest Neighbor method. The use of K-Nearest Neighbor aims to classify new objects based on learning data that are closest to the new object. In this case the amount of data or commonly called the nearest neighbor is determined by the user which is stated by K[17]. The steps in the K-Nearest Neighbor method are as follows:

- Calculates Euclidean distance
  Euclidean distance formula:

$$d = \sqrt{\sum_{i=1}^{p} (x_1 - x_2)^2} \tag{2}$$

  The equation states that d represents the distance, i represents the data index, p represents the data dimensions, x_1 represents the sample data, and x_2 represents the test data.

- Sort by Euclidean distance value
- Determine the k closest classification records
- The target output is the majority class

The formula for finding the K-Nearest Neighbor (K-NN) is as follows:

$$FAR = \frac{FR}{N} \times 100\% \tag{3}$$

$$FAR = \frac{FA}{N} \times 100\% \tag{4}$$

$$Accuracy = \frac{TA}{N} \times 100\,\% \tag{5}$$

Where FRR denotes False Rejection Rate, FAR expresses False Aceeptance Rate, FRexpresses False Rejection, FA expresses False Aceeptance, TA expresses True Acceptance and N represents the total data.

3 Research Database

In this research, it started with the determination of the object, conducting the study, designing the sys-tem under investigation, performing testing, and analysing the level of accuracy and processing speed ob-tained.

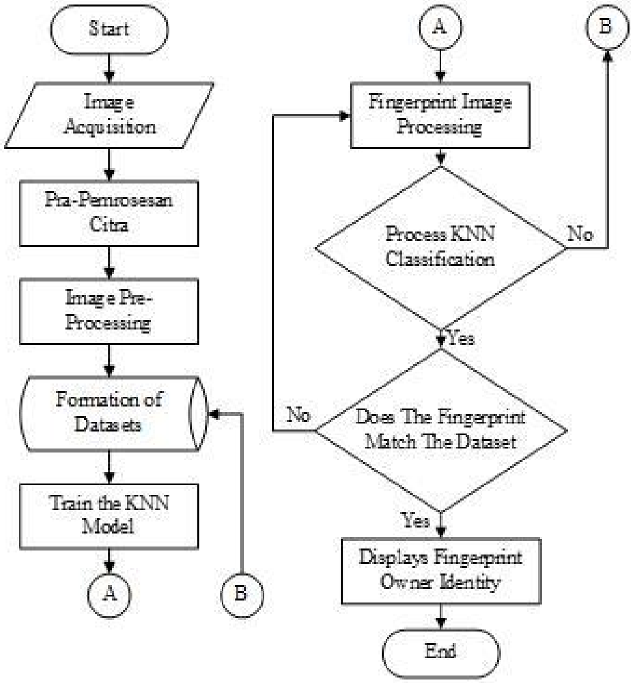

Fig. 1. Flowchart of fingerprint image and KNN

In Figure 3, the flowchart can be observed, starting with fingerprint image acquisition. The fingerprint image then undergoes preprocessing and feature extraction stages, including Grayscale, Thresholding, Thin-ning, and Resize processes. Afterwards, the processed fingerprint image is entered into the database. Once the fingerprint dataset is formed, training is conducted using the K-Nearest Neighbor (KNN) algorithm to achieve optimal results in fingerprint recognition and matching, as well as high-speed fingerprint image processing. Subsequently, KNN classifies the desired fingerprint data. If the detected fingerprint matches the fingerprint data in the dataset, the LCD display will show the owner's fingerprint identity

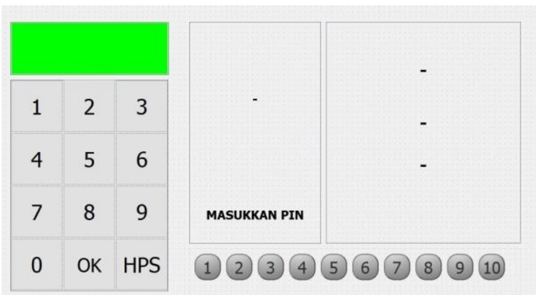

Fig. 2. User Interface Display

The user interface is displayed on the Raspberry Pi 7" Touch Screen Display. The interface shows num-bers as passwords, photos, names, rank/corps, NRP, and weapon owner's rack number. The preprocessing of fingerprint images is performed using the OpenCV application.

The purpose of this preprocessing is to enhance the ridge detection process in fingerprints. This process involves several stages, namely Grayscale, Thresholding, Thinning, and Resize. The Grayscale stage is performed to facilitate further analysis and processing. After the Grayscale stage, the Thresholding process is applied to separate the fingerprint ridges from the background of the image. Next, the Thinning stage is con-ducted to facilitate further analysis and feature extraction in the fingerprint.

Fig. 3. Thresholding and Thining Process

Afterwards, the resize process is performed to change the size of the image according to specific needs, such as reducing or enlarging the image size, as well as adjusting the desired image display.

The fingerprint images are stored in a dataset that will be used as training and validation data for the KNN model. This dataset is also used for testing and predicting fingerprint images. The dataset consists of five fingers: thumb, index finger, middle finger, ring finger, and little finger. Each finger has 10 images, resulting in a total of 50 fingerprint images. The dimensions of each image are 256×288 pixels.

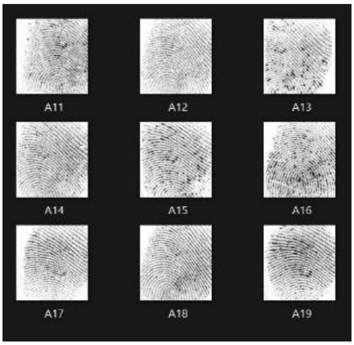

Fig. 4. Fingerprint image dataset

The design of K-Nearest Neighbor (K-NN) in fingerprint recognition aims to leverage the capabilities of this algorithm in performing fast and efficient fingerprint classification or comparison. The K-NN algorithm is utilized for various stages in the fingerprint recognition process, including identification, verification, and matching of fingerprint data.

1. The parameter K (K-value)

```
for p, q in matches:
    if p.distance <0.1 * q.distance:
        match_point.apppend(p)
keypoint = 0
```

Fig. 5. The nearest fingerprint data selection program

In Figure 5, the variable p represents the training image, q represents the validation image, and distance < 0.1 refers to a condition used to select the nearest neighbors in the classification or data grouping process. In the context of distance < 0.1, this parameter is used to measure the distance between the analyzed training image point and the matched validation image.

2. Majority Vote

```
if len(keypoints_1) < len(keypoints_2)
    keypoints = len (keypoints_1)
else:
        keypoints = len (keypoints_2)
```

Fig. 6. Majority Voting

Majority voting using keypoints refers to the features or key points in the data that are used to build the KNN model. These features serve as representations or descriptions of each data point and are used to select the class label with the highest frequency or number of neighbors as the predicted class label for the data being classified

4 Result and Discussion

In this discussion, the main focus is on the processing speed and matching accuracy in the fingerprint image processing system. The aim of this research is to measure the extent to which the system can recognize and match fingerprints with a high level of accuracy.

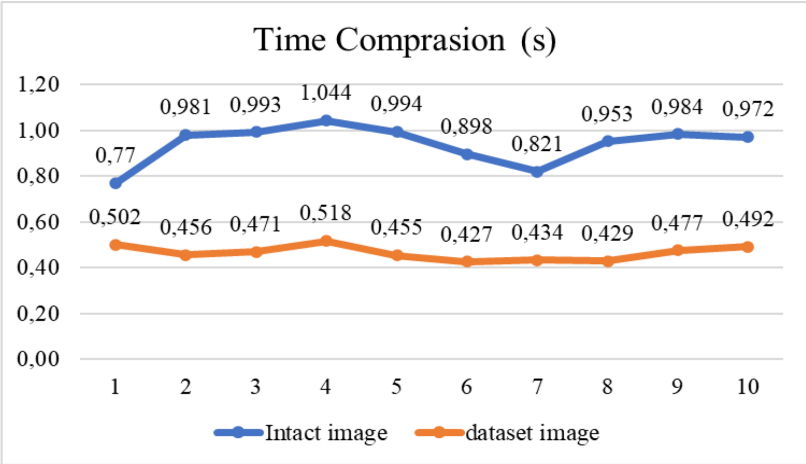

Fig. 7. Graph of fingerprint time comparison dataset of 10 images

In Figure 7, the comparison graph of fingerprint matching time using simulations with a dataset of 10 intact image is shown. From the graph, an average time value of 0.94 seconds is obtained. Meanwhile, for the dataset of 10 dataset image, an average time value of 0.46 seconds is obtained.

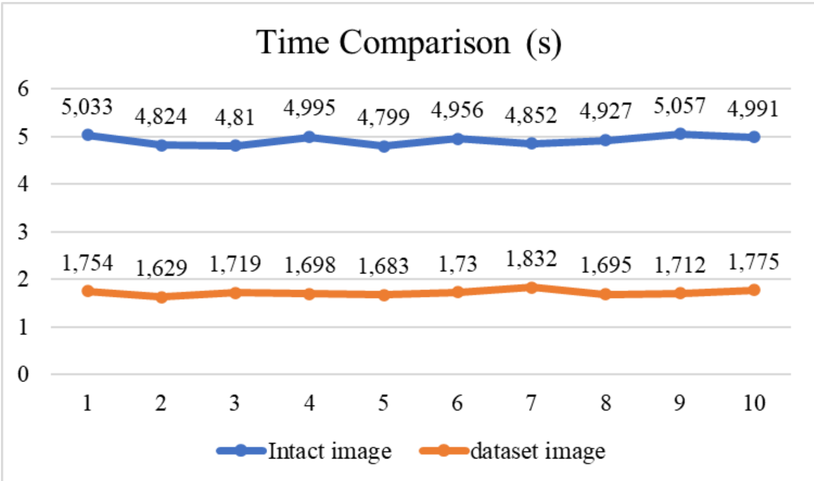

Fig. 8. Graph of fingerprint time comparison dataset of 50 images

In Figure 10, the comparison graph of fingerprint matching time using simulations with a dataset of 50 intact image is shown. From the graph, an average time value of 4.92 seconds is obtained. Meanwhile, for the dataset of 50 dataset image, an average time value of 1.72 seconds is obtained.

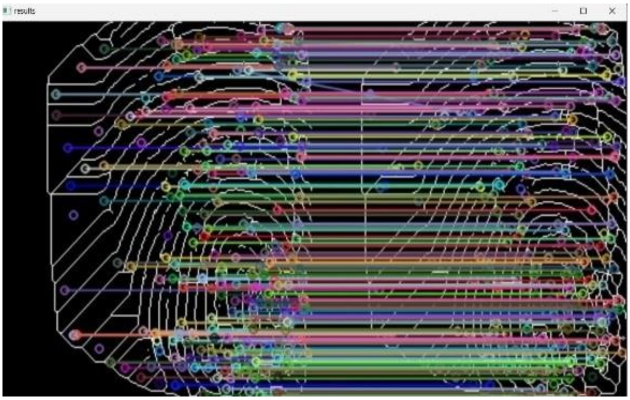

Fig. 9. Image distance reading process by KNN

In Figure 9, the process of capturing the training image (left) is compared with the validation image in the dataset (right), followed by the extraction of matching patterns from the fingerprint image.

Table 1. Parameter K, Percentage (ā), FRR and FAR

Identity	No (N)	Training (P)	Validation (Q)	Percentage ā (%)	FRR	FAR
Thumb	1	406,25	408,74	99,39	0	0
	2	393,05	394,60	99,61	0	0
	3	353,50	368,80	95,85	0	0
	4	387,56	393,33	98,53	0	0
	5	369,49	387,56	95,34	0	0
Index finger	6	374,67	382,23	98,02	0	0
	7	368,04	379,11	97,08	0	0
	8	354,14	368,04	96,22	0,69	0
	9	354,14	368,04	96,22	0,69	0
	10	354,14	368,04	96,22	0	0
Middle finger	11	397,13	399,94	99,30	0	0
	12	366,72	380,85	96,29	0	0
	13	366,72	380,85	96,29	0	0
	14	386,27	386,92	99,83	0	0
	15	393,55	386,92	98,91	0	0
Ring finger	16	394,67	397,48	99,29	0,62	0
	17	386,27	386,91	99,83	0	0
	18	388,20	389,74	99,60	0	0
	19	394,67	407,48	96,86	0	0
	20	355,99	391,80	90,86	0	0
Little finger	21	330,26	357,56	92,37	100	0
	22	5,29	373,17	1,42	0	0
	23	388,20	389,74	99,60	0	0
	24	401,97	424,36	97,81	0	0
	25	392,06	400,86	94,72	100	0

Based on Table 1, a testing was conducted on fingerprint detection and distance reading using 5 fingers, namely the thumb, index finger, middle finger, ring finger, and little finger. Each finger underwent 5 trials, with 24 successful attempts and 1 failure. Referring to formulas 3, 4, and 5, the test results obtained a FRR (False Rejection Rate) of 8%, FAR (False Acceptance Rate) of 0%, accuracy of 96%, and a parameter K of 93.4%.

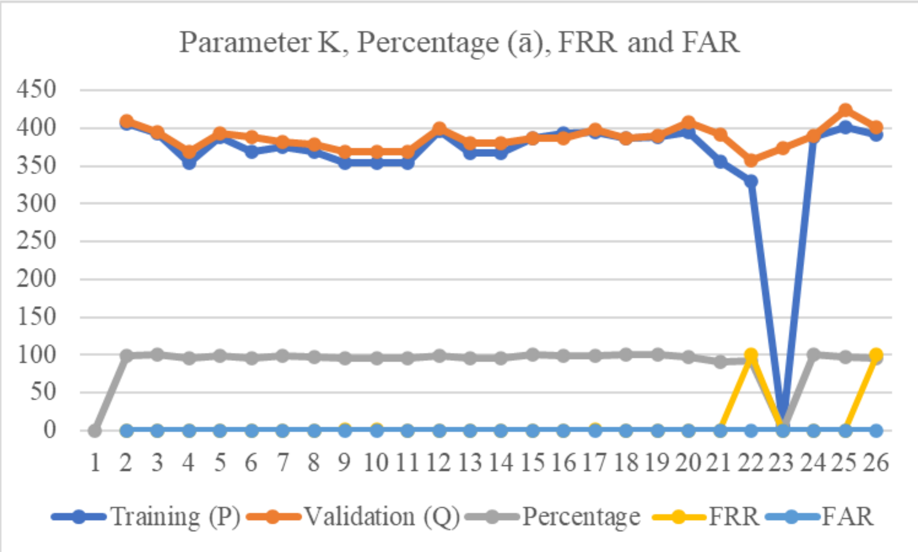

Fig. 10. Graph of Parameter K, Percentage (ā), FRR and FAR

From Figure 10, it can be concluded that to achieve a high level of accuracy in KNN reading of fingerprint images, it is important to ensure that the values of P and Q have a small difference. This means that if the program can process and provide a matching or validation rate above 90% on training and validation images, it will be considered as a match.

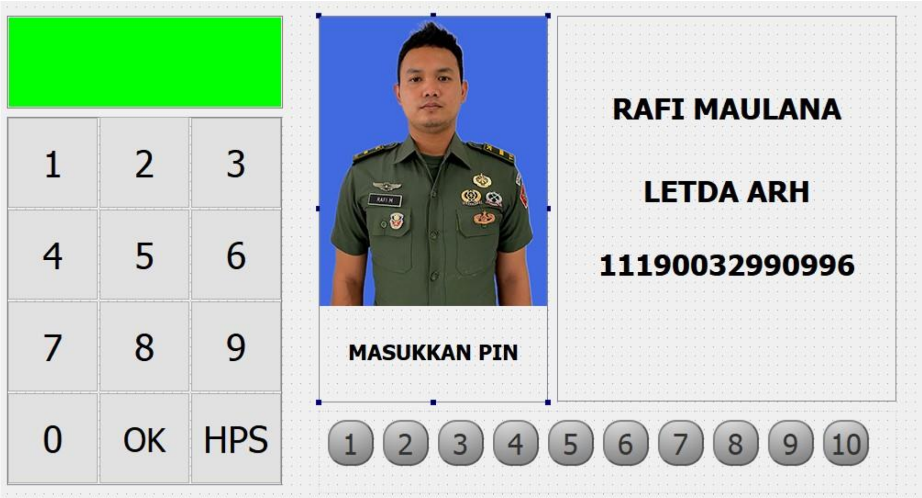

Fig. 11. Display of the User Interface on the LCD

In Figure 11, there is a User Interface display on the LCD that shows the identity information of the fingerprint owner, including a numerical password, photo, name, rank/corps, an NRP (identification number), and weapon owner's rack number/

5 Conclusion

The increased use of fingerprint data as an authentication method in the system requires enhanced security in storing the fingerprint template data in the database. This can be achieved by combining the KNN method, which provides fast processing and high detection accuracy, to recognize and match fingerprints.

The KNN method operates effectively in measuring distances and identifying fingerprint images generated through image processing. Based on this research, it can be concluded that the image identification speed using 10 images is 0.46 seconds, while for 50 images it is 1.72 seconds, with a 8% FRR and a 0% FAR using the optimal K parameter. The achieved success rate is 93%, with a high accuracy level reaching 96%. Therefore, an optimal level of processing speed and accuracy has been discovered.

References

1. J. Smith, A. Brown, "Secure Weapon System: A Case Study in Military Weapon Security", International Journal of Computer Science and Security (IJCSS), Vol. 10, No. 1, 2016.
2. N. Sudiana, A. Mufida, "Perancangan dan Implementasi Sistem Keamanan Senjata Militer dengan Menggunakan RFID dan Sensor Sidik Jari," Jurnal Teknologi dan Sistem Komputer, Vol. 5, No. 2, 2017.

3. Haniffah Z., Dirgantoro B., Setianingsih C., "Deteksi Kepribadian Anak dengan Sidik Jari Menggunakan Metode K-Nearest Neighbor (Knn) dan Decision Tree" ISSN : 2355-936, Bandung, 2019.
4. Situmorang B. H., " Identifikasi Biometrika Menggunakan Ekstraksi Minutiae Pada Citra Sidik Jari", Jurnal Teknoinfo, Volume 17, Nomor 1, Januari 2023, Page 289-302, Bogor, 2023.
5. Katon Agung A., "Klasifikasi Sidik Jari Menggunakan Jaringan Syaraf Tiruan Backpropagation", Jati (Jurnal Mahasiswa Teknik Informatik), Vol. 1 No. 1, Malang, 2017.
6. Miranda N. D., "Convolutional Neural Network pada Klasifikasi Sidik Jari Menggunakan Resnet-50", Jurnal Teknik In-formatika (JUTIF), Vol.1, No. 2, Desember 2020, hlm. 61-68, Bandung. 2020.
7. Saputra S., "Logika Fuzzy Metode Mamdani Untuk Pendeteksi Pola Sidik Jari Pada Algoritma Cany Edge Detector", Konferensi Nasional Sistem & Informatika 2017, Bali, 2017.
8. S. S. Kurniawan, A. F. Aulia, "Perancangan Sistem Keamanan Senjata Api pada Instansi Militer Berbasis Internet of Things (IoT)," Jurnal Ilmiah Teknologi Informasi Asia, Vol. 13, No. 2, 2020.
9. Cappelli, R., Lumini, A., & Maio, D. "Fingerprint Recognition: Recent Advances and Future Challenges". IEEE Transac-tions on Pattern Analysis and Machine Intelligence, 42(9), 2097-2112, 2020.
10. Tisse, C. L., & Dorizzi, B., "On the State of the Art of Fingerprint Recognition", IEEE Transactions on Biometrics, Be-havior, and Identity Science, 3(1), 2-20, 2021.
11. Ratha, N. K., Connell, J. H., & Bolle, R. M., "Adaptive Fingerprint Sensor Selection: A Framework for Improved Bio-metric Performance", IEEE Transactions on Pattern Analysis and Machine Intelligence, 43(2), 322-334, 2021.
12. Simonyan, K., & Zisserman, A. Very Deep "Convolutional Networks for Large-Scale Image Recognition", International Conference on Learning Representations (ICLR), 2015.
13. Zhang, Q., Lin, L., He, L., Liang, X., & Sun, J. "Content-Aware Deep Image Cropping", IEEE Transactions on Image Processing, 29, 1609-1621, 2020.
14. Le, V. H., & Lee, S. "A New Local Adaptive Thresholding Method for Document Image Binarization", Information Sci-ences, 529, 244-257, 2020.
15. Wang, Y., Chen, J., Li, B., & Liu, C. L., Rethinking Thinning in Chinese Calligraphy Character Recognition. Pattern Recognition, 108, 107456, 2020.
16. Elhoseiny, M., El-Melegy, M. T., & El-Saban, M., "Deep K-Nearest Neighbors: Towards Confident, Interpretable and Robust Deep Learning", IEEE Transactions on Pattern Analysis and Machine Intelligence, 41(4), 832-845, 2019.
17. Das, S., & Srivastava, S., "K-Nearest Neighbors Based Deep Feature Extraction for Image Retrieval", Expert Systems with Applications, 174, 114707, 2021