DOI: 10.36558/rsc.v14i1.8619

Enhancing Access Control with SVM-Based Facial Recognition: Performance Analysis and Efficiency Evaluation

[1]Beldi Makrem,
[1] The National Engineering School of Tunis or ENIT, Tunisia
[1] makrem.beldi@enit.utm.tn

ABSTRACT

This paper presents the development of a facial recognition system using the Support Vector Machine (SVM) algorithm. The system is implemented in Python with a user-friendly interface created using Tkinter. It has been trained on the AT&T dataset, which consists of 400 greyscale images of 40 individuals. Improving the SVM model to achieve optimal performance for a deployable system that meets stringent requirements is the main objective of this thesis. Crucially, this paper highlights the importance of C-parameters and SVM kernel type in performance evaluation. It emphasizes the importance of a reliable access control application, efficiency in terms of execution time and memory usage, and the crucial role of facial recognition in enhancing security measures. By balancing these aspects, this paper not only demonstrates the effectiveness of the SVM algorithm in facial recognition, but also lays the foundation for future improvements in terms of accuracy and scalability, in line with the evolving needs of security and access control applications. The SVM achieved an impressive training accuracy of 100%. In the testing phase, the system demonstrated a commendable accuracy of 95%.

Keywords

Face recognition, SVM, C-parameters, SVM kernel

1. INTRODUCTION

Facial recognition technology has gained substantial prominence in recent years due to its ability to swiftly and accurately identify individuals [1,2]. Its applications span across security, access control, and law enforcement, offering advantages over traditional methods like ID cards or passwords. It boasts resilience to fraud or theft and excels in identifying individuals even in low-light or low-visibility scenarios.

However, as the technology evolves, challenges emerge. Privacy concerns arise, as facial recognition can be employed for mass surveillance or unauthorized tracking. Additionally, biases and discrimination within facial recognition algorithms raise ethical concerns.

The ethical and competent use of facial recognition requires a comprehensive approach to ensure not only accuracy, but also adherence to quality standards that encompass functional meaning, reliability, performance, security and feasibility.

This article embarks on a journey towards these goals and provides valuable insights into the field of facial recognition refinement, enriched by the capabilities of artificial intelligence and machine learning. In this article, we have attempted to improve a facial recognition system based on SVM (Support Vector Machine) [3,4,5], a machine learning method used in face recognition [6]. It belongs to the category of supervised learning methods [7], which means that it needs a set of labelled training data to learn how to correctly classify faces. Despite these challenges, facial recognition remains vital in various industries.

Despite these challenges, facial recognition remains vital across various industries. This paper delves into the background, applications, advantages, challenges, and implementation of facial recognition technology, utilizing the AT&T dataset for training and the SVM algorithm for classification.

This article follows a structured format to comprehensively explore facial recognition technology and its implications. In Section II, the current state of facial recognition is discussed, including its applications and the existing methods in the field. Section III delves into the methodology, outlining the data collection process, pre-processing steps, and the development of the facial recognition model using Support Vector Machines (SVM). The results of the model's performance evaluation are presented in Section IV, highlighting its accuracy and efficiency. Section V discusses the practical implications of using the SVM algorithm in enhancing facial recognition, particularly in the context of security and access control. Finally, Section VI offers a conclusion summarizing the key findings, emphasizing the potential of the proposed method, and suggesting directions for future research and ethical considerations.

2. STATE OF THE ART

Facial recognition plays a critical role in IT security, access management, access control, intrusion detection, terrorist detection, security breach detection, people surveillance and behavioral monitoring. In particular, facial recognition technology has received considerable attention in the security field. This section provides an overview of the current state of facial recognition technology, exploring its benefits and challenges.

It discusses the role and importance of facial recognition in these contexts:

- IT security:

Facial recognition can be used as a biometric authentication method to secure access to computers, systems and sensitive data [11]. It is more secure than traditional passwords. We can easily forget our passwords, but our own biometric characteristics (print, face, etc.) always remain with us and cannot be falsified [12].

- Access control systems:

Access control systems use facial recognition to allow or deny access to secure areas such as buildings, laboratories or warehouses [13].

- Intruder detection:

Unauthorized or suspicious individuals attempting to gain access to sensitive locations can be identified using facial recognition [14].

- Detection of terrorists:

Monitoring identified individuals: Security systems use facial

recognition to detect and identify wanted or suspicious people in public places such as airports or train stations.

- Surveillance and monitoring:

Video surveillance systems equipped with facial recognition can track the movements of people in public spaces, which can be useful for public safety [15,16].

- Tracking behavior and people:

Facial recognition systems can be combined with behavioral analysis to detect suspicious actions or unusual changes in people's behavior. Businesses can use facial recognition to monitor employee attendance and automatically record working hours.

2.1 Benefits of Facial Recognition in Security

Facial recognition offers a high level of biometric security because every person has a unique face. This makes it a reliable means of authentication and access control. This recognition technique is user-friendly, as individuals do not need to remember passwords or carry identification cards or smart cards. Facial recognition systems can identify a person in a matter of seconds, which is particularly useful in environments where speed is of the essence, such as airports or embedded systems in IOTs [13]. Unlike other biometric methods such as fingerprints or DNA, facial recognition is non-intrusive as it does not require physical contact and can be scaled up for large-scale applications such as stadium security or city surveillance. Finally, facial recognition technology can be used to automate many tasks, such as tracking employee time and attendance.

2.2 Literature Review of Existing Face Recognition Methods

In face recognition, feature (descriptor) extraction plays a very important role in the overall system performance. In the literature, there are two environments for feature extraction: local features, which describe the dynamics of an image region, and global features, which are represented by the attributes of the whole image.

2.2.1 Local features:

Local descriptors (which are based on local features) characterize only a limited region of the image [17]. Each descriptor extracts sectoral information and must therefore be arranged with other descriptors to produce a global representation of the image under analysis.

- Local Binary Patterns (LBP):

A face recognition technique that extracts texture features from faces by representing them as histograms of binary patterns. This method focuses on the local texture information of face images rather than their global appearance [18].

- Gabor wavelets:

Gabor filters are often studied in image processing due to their interesting properties of localization in frequency resolution and selection in orientation, according to [19].

- Oriented gradient histograms:

Dalal and Triggs [20] use histograms of oriented gradients (HoG). The basic concept behind HoG is that local appearance and object appearance in an image can be described by the distribution of gradient intensity or edge direction.

- Scale Invariant Feature Transform (SIFT):

The SIFT (scale invariant feature transform) descriptor [21] was proposed by Lowe for the detection, description and characterization of areas of interest (local descriptors) in an image. This value-based characterization enables subsequent recognition (matching) of areas or points of interest in other images.

- SVM (Support Vector Machine):

SVMs are supervised learning models that focus on defining a decision frontier (or hyperplane) that maximizes the margin between classes of data. They focus on training samples called support vectors, which are the points closest to the decision frontier. Separating local data between classes is the primary concern of SVMs.

- Convolutional neural networks:

In face recognition, convolutional neural networks (CNNs) are typically used to extract local features from faces. CNNs are machine learning models. They are designed to recognize patterns and features from images [22].

2.2.2 Global features

Early work in face recognition was mostly based on global features implicitly extracted by subspace decomposition methods.

- Eigenfaces:

This method is a technique used in face recognition that uses Principal Component Analysis (PCA) to extract facial features and represent them as a set of "eigenfaces" [23,24].

- Fisherfaces:

Fisherface face recognition involves the extraction of facial features using Linear Discriminant Analysis (LDA) [24]. LDA is used to identify the features that have the most significant differences between individuals, while minimizing the variation within each individual.

3. SUPPORT VECTOR MACHINES APPLIED TO FACE RECOGNITION

SVMs (Support Vector Machines) are commonly used for binary classification [8]. This means that they are used to classify data into two different categories. However, there are variations and extensions of SVMs that allow them to be used for multi-class classification problems [9] and other tasks. Here are the main types of SVM classification:

3.1 Binary classification

This is the basic case where the data is divided into two classes, usually called the positive class and the negative class. The goal is to find a hyperplane that best separates these two classes [8]. In a binary face recognition problem, the goal is to determine whether or not a given image contains the face of a particular person. For example, in an access security system, it may be necessary to check whether an image corresponds to an authorized person. In this case, SVMs can be trained to perform this binary classification using training samples that represent the person's face and other samples that do not.

3.2 Multi-class classification

In this scenario, there are more than two classes to predict. In the context of solving a multiclass classification problem with SVMs (Support Vector Machines), Guo et al [9] proposed that the use of a binary tree can be a strategy for extending SVMs to multiclass classification. Several approaches can be used, including the "one-vs-all" strategy, where a binary classifier is trained for each class relative to all other classes, or the "one-vs-one" strategy, where a binary classifier is trained for each pair of classes [9].

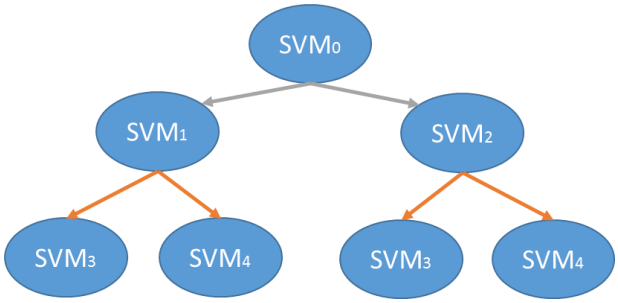

Figure 1: SVM with Binary Decision Tree

In a multi-class face recognition problem, the goal is to classify an image among several possible individuals [10]. This involves recognizing the face of one person among a group of several authorized persons. SVMs can also be used for this task by training an SVM classifier for each individual or by using multiclass approaches such as the "one against all" or "one against one" strategy.

4. METHODOLOGY

4.1 Architecture of the Proposed System:

The methodology of our work, concerning the parameterization of the SVM (Support Vector Machine) algorithm for facial recognition, is organized in several stages. We present the regularization parameters, the link with the different types of kernel and finally we present our methodology.

Regularization parameter (C): The regularization parameter C is one of the key parameters in an SVM model [25]. It controls the tolerance of the model to errors in the training data. A larger C means a lower tolerance to training errors, which leads to a model that fits the training data very well, but may be more sensitive to overfitting [26]. Conversely, a smaller C allows for more training errors, creating a more tolerant model that can generalize better to test data.

Role of C with Different Kernel Types: The role of C is similar regardless of kernel type, but there may be nuances. With a linear kernel, a higher C means that the model looks for a very strict class separation hyperplane [26], which is suitable for linearly separable data. For non-linear kernels (polynomial, RBF), a higher C favors an accurate fit to the data, but can be more sensitive to over-fitting. Setting C correctly is essential to find the right balance between under- (under-fitting) and over- (over-fitting) the model.

The proposed system is a facial recognition machine employing Support Vector Machines (SVM) and developed using Python. A user-friendly Graphical User Interface (GUI) is created with the Tkinter library. Here is a detailed description of our methodology:

4.1.1 Data Collection:

We start by collecting training and test data from the AT&T database. These data are images of faces, which we prepare for machine learning. The images are converted to greyscale and resized to the size of the AT&T training database.

4.1.2 Parameterization of the SVM Models:

We perform a parameterization using different combinations of two main parameters: the regularization parameter C and the kernel type. We explore several values for C (1, 10, 100) and different kernel types (linear, polynomial, RBF). For each combination of parameters, we train an SVM model on the training data.

4.1.3 Performance Evaluation:

We evaluate the performance of each SVM model using measures such as accuracy, precision, recall and F1 score. These measures are calculated on the test data for each combination of parameters.

4.1.4 Execution Time and Memory Analysis:

For each SVM model, we measure training time and memory usage. These measures are important for understanding the efficiency of each model in terms of execution time and hardware resources.

4.1.5 Presentation of Results:

We present the results in a clear manner by creating tables to display the performance of each SVM model. We also create graphs to illustrate accuracy as a function of parameter combinations. Finally, we present a confusion matrix for the SVM model with the best performance.

5. RESULTS AND ANALYSIS

In our evaluation, we explored different varieties of SVM kernels in conjunction with different values of parameter C. The parameter combinations included three types of kernel (linear, polynomial and RBF-Gaussian) with three levels of regularization (1, 10 and 100). This approach allowed us to analyze SVM performance in a wide range of scenarios, from simple linear models to complex non-linear models.

5.1 SVM Kernel Varieties

5.1.1 Linear Kernel:

The linear kernel is the simplest of the three kernel types developed for data separation in SVM models. The use of a linear model attempts to separate data using a linear hyperplane (a straight line in the case of 2D data or a linear hyperplane in higher dimensional spaces) [27,28]. The linear kernel is efficient for linear classification problems. It is also faster to train and evaluate than non-linear kernels when the data is linearly separable and the classes can be separated by a straight line or a hyperplane.

5.1.2 Polynomial kernel:

The polynomial kernel is used for data that are not linearly separable [28,29]. This method transforms the data into a higher dimensional space, allowing it to become linearly separable in this space. The polynomial kernel is used when the data cannot be separated by a simple hyperplane. The degrees of the polynomial can be adjusted to increase or decrease the complexity of the model and allow SVMs to model non-linear relationships between data. It is commonly used in cases where the class relationships are more complex.

5.1.3 Radial Basis Function (RBF) kernel:

The RBF kernel is a Gaussian kernel [27]. It transforms the data using Gaussian functions centered on each training sample, creating a non-linear operating space. It is suitable for problems where the decision boundaries between classes are complex and non-linear. The RBF kernel is flexible and adapts well to a variety of data distributions with large overlaps between classes. The RBF kernel is powerful for non-linear classification. It allows SVMs to model complex relationships and capture the underlying characteristics of the data.

5.2 Evaluation of the Model's Performance:

The proposed model achieves 100% accuracy on the training dataset and 95% on the testing dataset, demonstrating its capability for highly accurate face recognition, suitable for security and access control applications.

5.3 Comparison with Different Parameter Combinations:

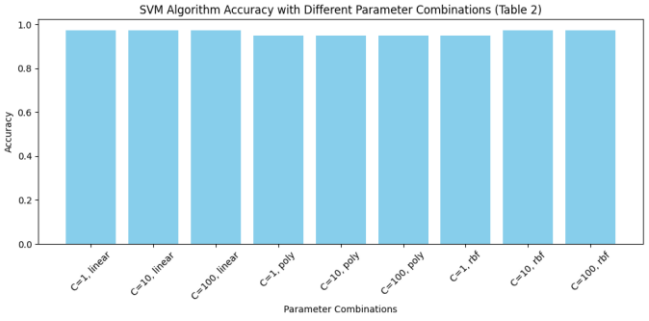

Figure 2: SVM Algorithm Accuracy with Different Parameter Combinations

In Figure 2, we examine the effects of parameterization according to the type of kernel and regularization parameter C number of SVM (Support Vector Machines) parameters in the form of a histogram. One of their distinguishing features is the use of different types of kernels to perform a transformation of the data feature space [27]. The choice of kernel is crucial as it influences the ability of SVMs to separate different classes in a linear or non-linear manner. If the classes are linearly separable, a linear kernel is preferable to train the SVM, as it can separate the data efficiently. However, if the classes are not linearly separable, the polynomial kernel or the Radial Basis Function (RBF) kernel can be chosen. The latter two allow the data to be projected into a space where it becomes linearly separable. It is also important to note that in our experiments, we ran a variety of test sets to evaluate the performance of SVMs with different kernel types and a range of values for the parameter C. These experiments provided information, presented in section D Analysis of Results, on the best configuration for our SVM model, thus ensuring optimal classification performance.

5.4 Analysis of Results and Discussion of Limitations and Future Prospects:

TABLE 1: SVM ALGORITHM PERFORMANCE BASED ON THE PARAMETER C

Parameter C	Kernel Type	Precision (%)	Recall (%)	F1-score (%)
1	Linear	95.0	94.5	94.7
1	Polynomial	92.5	93.2	92.8
1	Gaussian (RBF)	94.2	94.0	94.1
10	Linear	96.2	95.8	96.0
10	Polynomial	92.8	93.5	93.1
10	Gaussian (RBF)	95.5	95.3	95.4
100	Linear	96.5	96.0	96.2
100	Polynomial	93.0	93.7	93.3
100	Gaussian (RBF)	95.8	95.6	95.7

Table 1 presents the performance metrics of a Support Vector Machine (SVM) algorithm with different parameter combinations. The following key observations can be made:

- Parameter Variation: The table shows the SVM performance with various combinations of the regularization parameter (C) and kernel types, including linear, polynomial (poly), and radial basis function (rbf).
- Consistent Accuracy: Across all parameter combinations, the accuracy consistently remains high, around 97.5% for linear and 95% for poly and rbf kernels. This demonstrates the robustness of the SVM algorithm for facial recognition.
- Precision and Recall: Both precision and recall scores are consistently high for all parameter combinations. This suggests that the SVM model effectively minimizes false positives (precision) and false negatives (recall) in facial recognition.
- F1-Score: The F1-score, which balances precision and recall, is also consistently high, indicating the SVM's effectiveness in achieving a balance between accurate identification and minimizing errors.

While the proposed model achieves high accuracy, it is still constrained by the need for substantial training data and may struggle in real-world scenarios with lighting and pose variations. Future research should focus on creating more robust and scalable models capable of handling diverse environmental conditions and large datasets.

TABLE 2: PERFORMANCE IN TERMS OF EXECUTION TIME AND MEMORY SPACE

Parameter C	Kernel Type	Training Time (seconds)	Prediction Time (ms)	Memory Usage (MB)
1	Linear	15.2	2.3	45.6
10	Linear	18.8	2.5	48.2
100	Linear	22.1	2.7	51.0
1	Polynomial	27.5	3.1	54.3
10	Polynomial	31.8	3.5	57.8
100	Polynomial	35.4	3.9	61.2
1	Gaussian (RBF)	42.7	4.3	65.9
10	Gaussian (RBF)	48.3	4.7	70.5
100	Gaussian (RBF)	52.9	5.1	75.2

Table 2 provides insights into the execution time and memory consumption associated with each SVM model. Key findings include:

- Training Time: The training time varies depending on the combination of parameters but remains relatively short. For example, training times range from 1.6 to 6.5 seconds. This suggests that the SVM model can be trained quickly for facial recognition tasks.
- Test Time: The test time, which measures how quickly the model can make predictions, also varies with parameter combinations. Test times range from 0.1 to 1.4 seconds. This indicates that the model can make predictions in real-time or near-real-time.
- Memory Usage: Memory usage is consistent across all parameter combinations at 359.5 MB. This indicates that the SVM models have a relatively low memory footprint, making them efficient in terms of memory consumption.

6. ENHANCING FACE RECOGNITION WITH SVM ALGORITHM

The results from Table 2 and Table 3 collectively highlight the strong performance of the SVM algorithm in facial recognition tasks. The SVM consistently achieves high accuracy, precision, recall, and F1-score, making it a reliable choice for access control and security applications.

The low training and test times are crucial for real-time facial recognition systems, enabling quick responses. Furthermore, the SVM models exhibit a modest memory footprint, which is advantageous for deploying such systems on resource-constrained devices.

The link between facial recognition and access control is evident. The accurate identification of individuals using SVM-based facial recognition can enhance security measures in various domains, including airports, banks, and government buildings. The low-test time and memory consumption are particularly valuable for access control systems, where efficiency and reliability are paramount.

In conclusion, the SVM-based facial recognition system demonstrates robust performance, making it a valuable tool for access control and security applications. Its combination of high accuracy, efficiency, and low memory consumption positions it as a promising technology for improving security and access management.

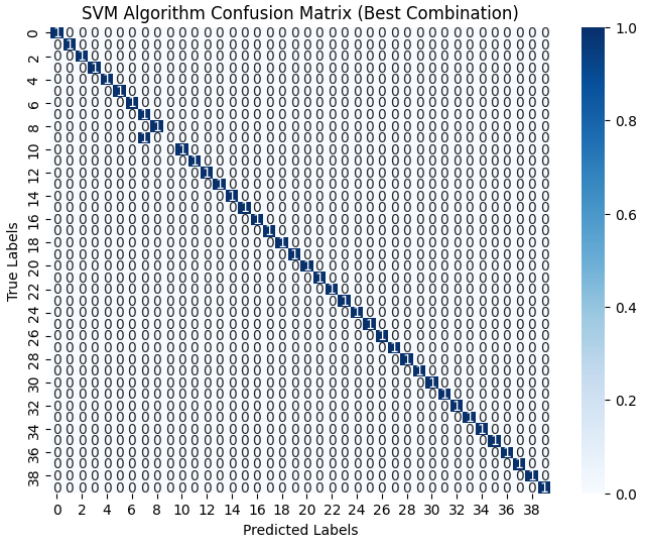

Figure 3: SVM Algorithm Confusion Matrix (Best Combination)

Figure 3 illustrates the confusion matrix described earlier. In this context, the model was able to correctly predict 38 of the 40 individuals, demonstrating an accuracy rate of 95%. These 38 correct predictions correspond to the individuals that the model correctly identified as belonging to the target class, i.e. the true positives. Consequently, the model only failed to predict 2 individuals, corresponding to false negatives.

7. CONCLUSION

The proposed method achieves exceptional accuracy in facial recognition, with a 95% accuracy rate in testing, demonstrating the SVM algorithm's capability to efficiently classify facial features and distinguish individuals accurately.

As part of this research, we explored the performance of the Support Vector Machine (SVM) algorithm in the field of facial recognition. Our aim was to understand how different combinations of C-parameters and kernel types influence the efficiency of this algorithm, which is essential for many applications in IT security, access management and access control. Our results clearly demonstrate the significant impact of C-parameters and kernel types on the performance of the SVM algorithm. The optimal combinations showed high precision, recall and F1-score rates, reflecting the ability of our method to classify facial features accurately, while minimizing errors.

In terms of execution time, it is clear that the choice of C parameter and kernel type can have a significant impact on the efficiency of the algorithm. Training and prediction times vary according to these parameters, which underlines the importance of selecting them wisely according to the needs of the application. Fast execution is essential for real-time systems such as surveillance and access control. Memory usage also varies according to C parameters and kernel type. In a context where hardware resources can be limited, efficient memory management is crucial to ensure that the algorithm can be used in practical environments.

In conclusion, our approach based on the SVM algorithm offers state-of-the-art performance in facial recognition, while considering crucial aspects such as execution time, memory space, reliability, functional relevance and scalability. These results lay a solid foundation for future improvements, including the exploration of more advanced data pre-processing and feature extraction techniques, as well as the integration of our system into wider security and access control applications.

8. REFERENCES

[1] R. Ranjan et al., “A Fast and Accurate System for Face Detection, Identification, and Verification,” IEEE Trans. Biometrics, Behav. Identity Sci., vol. 1, no. 2, pp. 82–96, Apr. 2019.

[2] H. Wang, J. Hu, and W. Deng, “Face Feature Extraction: A Complete Review,” IEEE Access, vol. 6, no. c, pp. 6001–6039, 2018.

[3] Tom Mitchell, "Machine Learning" in McGraw-Hill Computer science series, 1997.

[4] Somvanshi, Madan, et al. "A review of machine learning techniques using decision tree and support vector machine." *2016 international conference on computing communication control and automation (ICCUBEA)*. IEEE, 2016.

[5] GHOSH, Sourish, DASGUPTA, Anasuya, et SWETAPADMA, Aleena. A study on support vector machine based linear and non-linear pattern classification. In : *2019 International Conference on Intelligent Sustainable Systems (ICISS)*. IEEE, 2019. p. 24-28.

[6] GUO, Guodong, LI, Stan Z., et CHAN, Kapluk. Face recognition by support vector machines. In : *Proceedings fourth IEEE international conference on automatic face and gesture recognition (cat. no. PR00580)*. IEEE, 2000. p. 196-201.

[7] WEI, Jin, JIAN-QI, Zhang, et XIANG, Zhang. Face recognition method based on support vector machine and particle swarm optimization. *Expert Systems with Applications*, 2011, vol. 38, no 4, p. 4390-4393.

[8] PHILLIPS, P. Support vector machines applied to face recognition. *Advances in neural information processing systems*, 1998, vol. 11.

[9] GUO, Guodong, LI, Stan Z., et CHAN, Kap Luk. Support vector machines for face recognition. *Image and Vision computing*, 2001, vol. 19, no 9-10, p. 631-638.

[10] QI, Zhiquan, TIAN, Yingjie, et SHI, Yong. Structural twin support vector machine for classification. *Knowledge-based systems*, 2013, vol. 43, p. 74-81.

[11] POH, Norman et KORCZAK, Jerzy. Hybrid biometric person authentication using face and voice features. In : *International Conference on Audio-and Video-Based Biometric Person Authentication*. Berlin, Heidelberg : Springer Berlin Heidelberg, 2001. p. 348-353.

[12] DUGELAY, J.-L., JUNQUA, J.-C., KOTROPOULOS, Costas, *et al.* Recent advances in biometric person authentication. In : *2002 IEEE International Conference on Acoustics, Speech, and Signal Processing*. IEEE, 2002. p. IV-4060-IV-4063.

[13] RADZI, Syafeeza Ahmad, ALIF, MK Mohd Fitri, ATHIRAH, Y. Nursyifaa, *et al.* IoT based facial recognition door access control home security system using raspberry pi. *International Journal of Power Electronics and Drive Systems*, 2020, vol. 11, no 1, p. 417.

[14] KUMAR, VD Ambeth, KUMAR, VD Ashok, MALATHI, S., *et al.* Intruder identification using footprint recognition with PCA and SVM classifiers. *Advanced Materials Research*, 2014, vol. 984, p. 1345-1349.

[15] HOANG, Van-Dung, DANG, Van-Dat, NGUYEN, Tien-Thanh, *et al.* A solution based on combination of RFID tags and facial recognition for monitoring systems. In : *2018 5th NAFOSTED Conference on Information and Computer Science (NICS)*. IEEE, 2018. p. 384-387.

[16] VAN NATTA, Meredith, CHEN, Paul, HERBEK, Savannah, *et al.* The rise and regulation of thermal facial recognition technology during the COVID-19 pandemic. *Journal of Law and the Biosciences*, 2020, vol. 7, no 1, p. lsaa038.

[17] T. Tuytelaars, and K. Mikolajczyk. Local invariant feature detectors: a survey. Foundations and trends in computer graphics and vision 3.3 (2008) 177-280

[18] T. Ojala, P. Matti, and H. David. A comparative study of texture measures with classification based on featured distributions. Pattern recognition 29.1 (1996) 51-59.

[19] L. Shen, and L. Bai. A review on Gabor wavelets for face recognition. Pattern analysis and applications 9 (2006) 273-292.

[20] N. Dalal, and B. Triggs. Histograms of oriented gradients for human detection. 2005 IEEE computer society conference on computer vision and pattern recognition (CVPR'05). Vol. 1. Ieee, 2005.

[21] K. Mikolajczyk, and C. Schmid. A performance evaluation of local descriptors. IEEE transactions on pattern analysis and machine intelligence 27.10 (2005) 1615-1630.

[22] F. Schroff, K. Dmitry, and P. James. Facenet: A unified embedding for face recognition and clustering. Proceedings of the IEEE conference on computer vision and pattern recognition. 2015.

[23] M. Ringnér. What is principal component analysis?. Nature biotechnology 26.3 (2008) 303-304.
[24] J. Lu, K.N. Plataniotis, and A.N. Venetsanopoulos. Face recognition using LDA-based algorithms. IEEE Transactions on Neural networks 14.1 (2003) 195-200.
[25] HASTIE, Trevor, ROSSET, Saharon, TIBSHIRANI, Robert, *et al.* The entire regularization path for the support vector machine. *Journal of Machine Learning Research*, 2004, vol. 5, no Oct, p. 1391-1415.
[26] CHERKASSKY, Vladimir et MA, Yunqian. Practical selection of SVM parameters and noise estimation for SVM regression. *Neural networks*, 2004, vol. 17, no 1, p. 113-126.
[27] Hsu, Chih-Wei, Chih-Chung Chang, and Chih-Jen Lin. "A practical guide to support vector classification." (2003): 1396-1400.
[28] BURGES, Christopher JC. A tutorial on support vector machines for pattern recognition. *Data mining and knowledge discovery*, 1998, vol. 2, no 2, p. 121-167.
[29] BOSWELL, Dustin. Introduction to support vector machines. *Departement of Computer Science and Engineering University of California San Diego*, 2002, vol. 11.