https://doi.org/10.24867/GRID-2018-p52 Professional paper

THE MODELING OF FACIAL RECOGNITION PROCESS IN PROSPECTIVE OF SIMULATION TECHNIQUES

(A methodical elaboration through the built-in modules of Matlab)

Jozef Bushati [1], Virtyt Lesha [2], Dea Strica [3]
[1]University of Shkodra, Shkodra, Albania
[2]Polytechnic University of Tirana, Tirana, Albania
[3]University of Tirana, Tirana, Albania

Abstract: *Biometrics is a field of study and dissemination that brings a host of contributing domains in an integrated way to generate products that find multiple applications particularly in creating an individual's identity. Modeling biometric processes reveals a series of discussions in various fields, mainly in medicine and in the digital image processing field. These discussions range from the simulation levels of these models to those of application and concrete products that use biometric systems for identifying and recognizing individuals. Indeed, over many biological parameters of an individual to be identifiable, there is the face that generates a lot of information about the physical data of the individual. According to specific studies, facial components combined according to different methods can be used to design simulation models that come in aid of generating closed-source and dedicated programs of biometric recognition (Das, 2018).*
This paper deals with a simulation model, through the Matlab test-bench, of the facial recognition process. The model is created at the proposal level for the subject field and gives the limits and recommendations for improvements and further developments of this simulation method.
The modeling methodology includes two parts; the first part is the Matlab program that has been coded in such a way that it takes some images that represent the faces of different people published in the open-source database on the Internet; the second part is the creation of a database that retains these images to enable face identification when entering a new image program. The facial search and retrieval process are built through some built-in functions that the latest versions of Matlab offer.
Finally, the simulation model provides a new facial identification model that leaves room for further discussion and limits to the relatively small number of face images that can support the database in Matlab.

Key words: biometrics, test-bench, database, identification

1. INTRODUCTION

Face recognition is a process under which persons are subject to biometric verification and authentication for purposes such as camera surveillance, biometric passports, biometric facial identification for research and development purposes, and so on (Buhrow, 2016). As one of the biometric identification methods, facial identification is more popular and has a lot of advantages over others where the main one consists in relatively large distance identification compared to other methods such as eye iris, sound, trace of fingers, etc (Bourlai, 2016).
Among the 6 biometric identifying methods considered, facial identification has shown greater suitability in a Machine-Readable Travel Documents (MTRD) system based on several evaluation factors such as: perception and suitability to the population, adaptability to electronic data processing equipment, etc. (Chityala et al, 2014). The following figure shows the suitability of facial identification in comparison to other biometric methods (Datta et al, 2015). As a biometric system, a system of face recognition operates in two methods: face verification and face identification. Face verification includes matching a one-to-one approach that compares an image to another image whose identity is known. An application of this method is the use of a biometric passport. Face identification includes one-to-many matching where a face image is compared to some face images that are stored on a database. Also, this method finds use when it is necessary to identify a face image that is closer to the face image being tested (Newman, 2009).
Further face recognition development has brought a lot of development of image processing software in the sense of the parameters of these information processing machines, such as development of memory suitable for storing images with higher levels than just textual information. In addition, requests and advances in this regard are also represented in database developments, their management as well as storage of visual and multimedia data; it is natural that in this case the budgets and the financial capacities for these purposes are imposed.

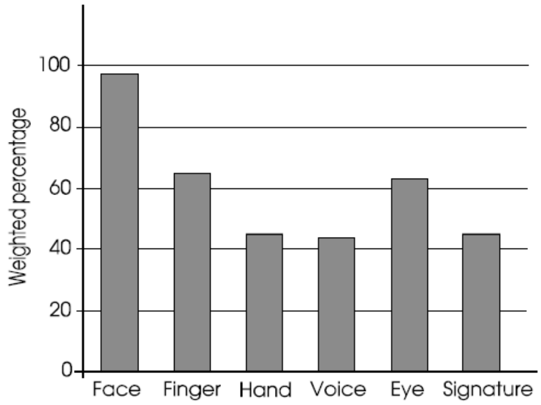

Figure 1: Level of suitability to an MTRD system of 6 biometric Identification attributes

The use of face recognition includes a series of defined algorithms which, through the development and simulation by respective programs, move on these identification processes

The algorithms developed in face recognition processes: are Eigenface, Gabor jets, Fisherface, etc. In this paper we discuss and elaborate the Eigenface method in terms of its improvement in the simulation software environment and the provision of further proposals for the performance of this method.

Finally, automatic face identification faces multiple challenges with various technical aspects such as when we have an environmental impact on an image such as damage to the figures that will be identified, photographing under undesirable effects of light, making pixel elements prevent to perform with images, etc.

2. METHODOLOGY

The face images, presented as multidimensional matrices, are subject to image processing through algorithms that have shown high interest in the computer vision in general. Surveys on this subject include algebraic and statistical methods for extracting and further elaboration these images.

In concrete applications, rather than simulation environments, the facial identification consists on some processes that are taken into consideration. In principle, as input of the system, the facial image is subjected to observation by a sensor that records the values of facial image pixels; image pixels can be monochromatic or colored and for each image type, specific attributes are required for image identification and recording as inputs in the system.

Facial image is considered as a two-dimensional matrix that, after switching to the scanning of the sensor, requires normalization in fixed dimensions of the *mxn* size of matrix and this is predetermined for each facial image that may have different dimensions, depending on the distance when the face image was taken from the sensor.

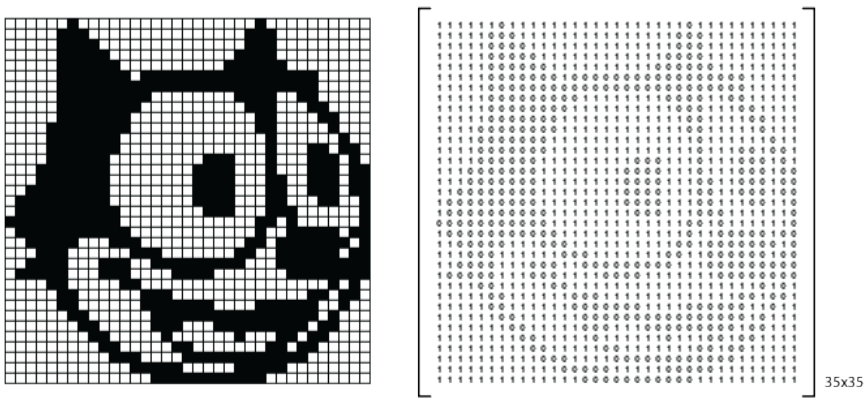

Figure 2: Monochromatic image expressed in bites

Face image and image processing in general, specifies and analyzes each pixel of the image in question and the dimensions of the images that are subject to normalization and greatly affect the processing capabilities of the computing machines; the purpose of the computing machine is to normalize the image in fixed dimensions *mxn*. As a result, processing methods face this challenge because in high values of dimensions, several disadvantages arise where the most commonly required costly calculation capabilities in case of these high-dimensional images.

The most common face image processing model is the Principal Component Analysis (PCA) dimensioning technique; the first component of this approach is the linear combination of the original image dimensional dimensions.

The most common case in simulation and application development facilities is multiple analysis assuming that the extraction processing and the principal manifold are linear. Once facial image is normalized and extracted from the facial image database, it behaves as a linear subspace from the image subspace.

In the case of our simulation model, we have considered one of the most common methods of linear analysis that is the Eigenface technique which is subject to probabilistic models.

The Eigenface method was proposed for the first time by Turk/Pentland. In the case of our study, an improved version was taken into account by highlighting the advantages it includes and the limits it pursues under this modified model.

The simulation environment is Matlab R2017, which has already built-in functions that support image processing applications and, consequently, the biometric case.

The simulation takes into consideration a database of 400 face images that are included in the program; this database is valid for performing the identification of face images that will be inserted for matching.

The developed code adapts the Eigenface method with a treatment other than classic approach in the sense of the size of the database in terms of the number of facial images supported. Also, the other modification that is applied to the classical Eigenface method considers of the time of execution of face recognition.

The simulation model consists of two parts: the creation of the database execution code and the main code used to read this database required for the recognition process (Jain et al, 2011).

The face recognition simulation considers 400 different persons in the modeling process. The modification consists on intervening the two cycles needed to complete the creation of the database and introducing a subset for each cycle leading to the minimization of the time of reading the database in the main code of the face recognition and further increasing the number of supported of face images in the database.

Also, for each execution of 400 face images, the time of execution was measured as in the case of the Eigenface class method and the improved version of this method. Finally, the graphical presentation of each case is enabled thus showing the comparison between the two Eigenface methods regarding the time execution.

3. RESULTS

In the simulation discussed in this paper, the methodology is reflected in detail in the results section where the findings and determining elements discussed in the above sections are provided.

Specifically, in our simulation we have used 400 face images which are placed as input on the improved model of Matlab simulation; against the standard simulation model in question, the performance of each simulation model is determined in terms of the time needed for executing the face image recognition process for the database at all.

In function of this approach, we have found the execution time for each image's recognition of 400 images per standard and improved algorithm. Moreover, through Matlab simulations we have defined the trend-line for each simulation model; this trend-line is of the "General Model Gauss8" type of the eighth order.

The approximation equation by the "General Model Gauss8" method is theoretically the following:

$$f(x) = a1 \times \exp(-\left(\frac{x-b1}{c1}\right)^2 + a2 \times \exp(-\left(\frac{x-b2}{c2}\right)^2 + a3 \times \exp(-\left(\frac{x-b3}{c3}\right)^2 + a4 \times \exp(-\left(\frac{x-b4}{c4}\right)^2 + a5 \times \exp(-\left(\frac{(x-b5)}{c5}\right)^2 + a6 \times \exp(-\left(\frac{x-b6}{c6}\right)^2 + a7 \times \exp(-\left(\frac{x-b7}{c7}\right)^2 + a8 \times \exp(-\left(\frac{x-b8}{c8}\right)^2 \quad (1)$$

where, the coefficients a1, b1, etc. consist of the approximation line determinants according to the "General Model Gauss8" method. The values of the above constants are presented below for both the simulation model and the improved simulation.

Table 1: Trendline coefficients according to the General Gauss8 algorithm for the classic simulation model

Coefficients (with 95% confidence bounds)	The values Standard Algorithm	The values Improved Algorithm
a1	2.229 (-114.7, 119.1)	1.491 (-24.18, 27.16)
b1	320.4 (200.2, 440.5)	339.3 (281, 397.5)
c1	320.4 (200.2, 440.5)	38.3 (-127.3, 203.9)
a2	320.4 (200.2, 440.5)	0.7382 (-15.77, 17.25)
b2	107.4 (-1490, 1705)	166.5 (122.5, 210.6)
c2	128.4 (-3073, 3330)	25.69 (-80.65, 132)
a3	2.336 (-40.91, 45.59)	0.2948 (-3.196, 3.786)
b3	218.1 (134.5, 301.8)	79.68 (40.61, 118.8)
c3	27.72 (-110.5, 165.9)	21.83 (-44.45, 88.11)
a4	2.136 (-38.29, 42.56)	1.278 (-27.25, 29.81)
b4	-5.647 (-146, 134.7)	200 (126.7, 273.3)
c4	51.78 (-287.9, 391.4)1.086 (-1.765, 3.937)	37.51 (-237.7, 312.8)
a5	1.086 (-1.765, 3.937)	2.746 (2.54, 2.952)
b5	172 (165, 178.9)	22.5 (-16.04, 61.03)
c5	13.04 (-5.923, 32)	105.3 (-34.39, 245)
a6	1.192 (-42.31, 44.69)	1.591 (-6.449, 9.632)
b6	367.4 (316.5, 418.2)	129.1 (-35.42, 293.6)
c6	21.2 (-159, 201.4)	42.5 (-242.1, 327.1)
a7	1.498e+12 (-4.008e+17, 4.008e+17)	2.426 (0.3012, 4.551)
b7	4823 (-4.472e+07, 4.473e+07)	396.1 (345.9, 446.2)
c7	858.8 (-4.351e+06, 4.353e+06)	42.06 (-48.42, 132.5)
a8	2.552 (-103.3, 108.4)	2.687 (-2.356, 7.729)
b8	273.9 (206.3, 341.4)	267.4 (168.7, 366.1)
c8	30.49 (-197.2, 258.2)	69.94 (-738.5, 878.4)

Below we have presented the relevant trend charts for each model, the standard and the improved one.

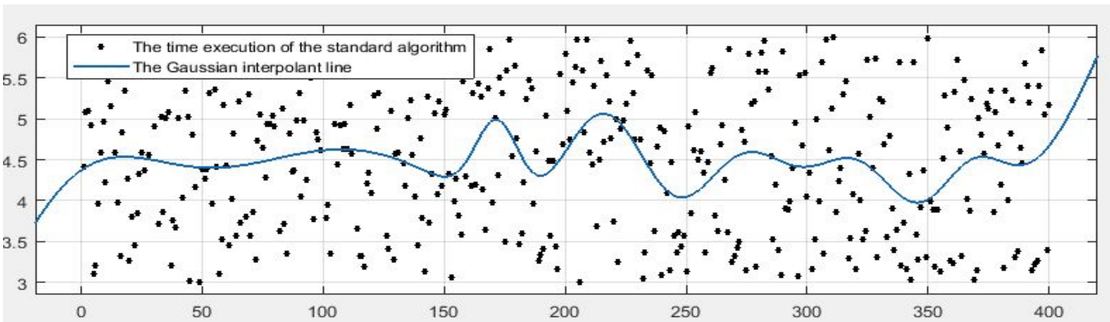

Figure 3: A graph showing the dependency of the execution time for the recognition of each face image for the standard model

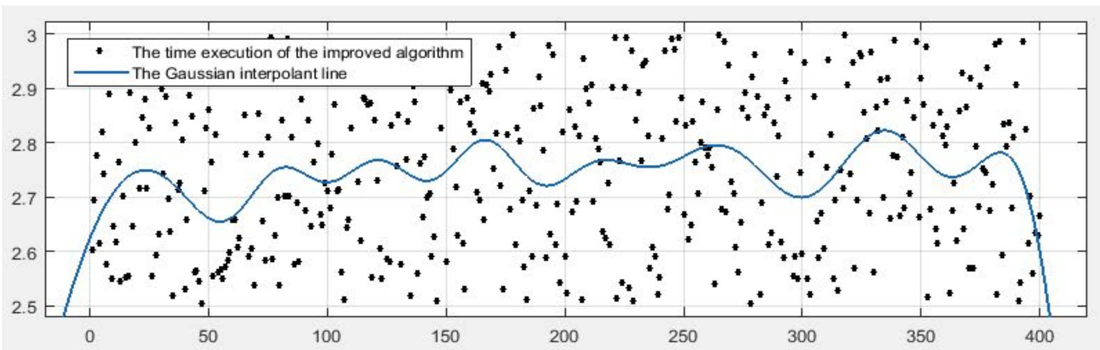

Figure 4: A graph showing the dependency of the execution time for the recognition of each face image for the improved pattern

We can confirm, from the graphs presented above, the time execution needed to perform the recognition process for the improved model of the simulation, is lower in comparison to the classical model.
In the corresponding graph of the improved simulation model the biggest value of the time execution needed to perform the recognition does not exceed 2.9 sec whereas in the standard model graph the lowest execution time is above 3.5 sec.
Also, presented below are the supporting data of the trend-line performance and the execution time of which 400 imaging images of the simulation model.

Table 2: A table showing the performance data of two models

	SSE_Standard Algorithm	R-squared_Standard Algorithm	Adjusted R-square_Standard Algorithm	RMSE_Standard Algorithm
Standard Algorithm	268.8	0.06352	0.006235	0.8456
Improved Algorithm	7.484	0.06752	0.01048	0.1411

Both simulation models support the "NonLinearLeastSquares" method and are not robust. The algorithm used to program the approximation method "Gauss8" is "Trust-Region". Further details are as follows:

1. *DiffMinChange* – 1.0e-8
2. *DiffMaxChange* – 0.1
3. *MaxFunEvals* – 600
4. *MaxIter* – 400
5. *TolFun* – 1.0e-6
6. *TolX* – 1.0e-6

4. CONCLUSIONS AND DISCUSSIONS

In this paper we discussed about the field of biometric face recognition regarding performance of the Eigenface algorithm (Gonzalez et al, 2018).
The importance of the Eigenface model, as an algorithm for biometric facial identification processes, plays an important role and provides space for continuous reconfigurations and modifications that improve the performance of the models in question in different terms where one of them is the time of facial identification execution versus a certain database.
In our model we have implemented a simulation model through Matlab software (R2017a), which realizes the face recognition process including an open source database of 400 facial images (Solomon et al, 2011).
The purpose of this paper was to present a simulation model that deals inside it with a modified version of the Eigenface algorithm; the modeling was intended to give the results of this improvement of the Eigenfaces algorithm in terms of the execution time of identifying a facial image.
The modification, carried out in the Eigenface algorithm, was the intervening in the respective cycles by introducing a subset into the algorithm. Also, the modification of this modeling was the intervention standard code that makes it possible to read the database that is further applied in the basic model algorithm (Nixon et al, 2013).
Also, to reflect the results in time, we simulated through Matlab a “Gauss8 approximation” algorithm model that generates a threefold reflection performance of the standard Eigenface algorithm versus the improved algorithm (Bolle et al, 2004). Along with the graphical data of Gauss8 approximation, specific and supporting parameters are also given, where they are considered as computable coefficients (Petrou et al, 2015).
Because of the Gauss8 method simulations, it is concluded that the execution time of the facial image identification process for the standard Eigenface algorithm has extremes that show greater time values compared to the improved pattern (Vacca, 2007).
Disputes and discussions that arise from this study leave spaces for improvements in the terms of time needed to perform facial identification process performances as well as the size of the database needed to keep the face images; in relation to the database arises the challenge of designing models that go hand in hand with the improvement of the model's quality in relation to the execution.

5. REFERENCES

[1] Bolle, R. M., Connell, J. H., Pankanti, S., Ratha, N. K., Senior, A. W.: “Guide to biometrics”, (Springer-Verlag, New York, 2004), doi: 10.1007/978-1-4757-4036-3.

[2] Bourlai, T.: “Face Recognition Across the Imaging Spectrum”, (Springer International Publishing, Switzerland, 2016), doi: 10.1007/978-3-319-28501-6.

[3] Buhrow, W. C.: “Biometrics in support of military operations: Lessons from the battlefield”, (CRC Press, Florida, USA, 2016).

[4] Chityala, R., Pudipeddi, S.: “Image processing and acquisition using Python”, (CRC Press, Florida, USA, 2014).

[5] Das, R.: “The science of biometrics: Security technology for identity verification”, (Routledge, New York, 2018), doi: 10.4324/9780429487583.

[6] Datta, A. K., Datta, M., Banerjee, P. K.: “Face detection and recognition: Theory and practice”, (Taylor & Francis, Florida, USA, 2015), doi: 10.1201/b19349.

[7] Gonzalez, R. C., Woods, R. E.: “Digital image processing”, 4th ed, (Pearson, New York, 2018).

[8] Jain, A. K., Li, S. Z.: “Handbook of face recognition”, (Springer-Verlag, London, 2011), doi: 10.1007/978-0-85729-932-1

[9] Newman, R.: “Security and access control using biometric technologies”, (Cengage Learning, US, 2009).

[10] Nixon, M. S., Aguado, A. S.: “Feature extraction and image processing for computer vision”, 3rd ed, (Academic Press, Oxford, UK, 2013), doi: 10.1016/C2011-0-06935-1.

[11] Petrou, M., Petrou, C.: “Image processing: The fundamentals”, 2nd ed, (John Wiley & Sons, Chichester, 2015), doi: 10.1002/9781119994398.

[12] Solomon, C., Breckon, T.: “Fundamentals of digital image processing: A practical approach with examples in Matlab”, (John Wiley & Sons, Chichester, 2011), doi: 10.1002/9780470689776.

[13] Vacca, J. R.: “Biometric technologies and verification systems”, (Elsevier - Butterworth Heinemann, Amsterdam, 2007).