

Building Face Detection with Face Divine Proportions

<https://doi.org/10.3991/ijoe.v17i04.19149>

Mohamed Oualla (✉)

Moulay Ismail University, Errachidia, Morocco
m.oualla@umi.ac.ma

Khalid Ounachado, Abdelalim Sadiq
Ibn Tofail University, Kenitra, Morocco

Abstract—In this paper, we proposed an algorithm for detecting multiple human faces in an image based on haar-like features to represent the invariant characteristics of a face. The choice of relevant and more representative features is based on the divine proportions of a face. This technique, widely used in the world of beauty, especially in aesthetic medicine, allows the face to be divided into a set of specific regions according to known mathematical measures. Then we used the Adaboost algorithm for the learning phase. All our work is based on the Viola and Jones algorithm, in particular their innovative technique called Integral Image, which calculates the value of a Haar-Like feature extracted from a face image. In the rest of this article, we will show that our approach is promising and can achieve high detection rates of up to 97%.

Keywords—Face detection, Haar-like features, Adaboost, Facial proportions, Golden ratio

1 Introduction

Over the past two decades, with the development of information technology, face detection has become a key technology in computer vision through its various interesting applications, such as face recognition, expression recognition, digital video processing, video detection, video surveillance, human computer interface, video retrieval and face image database management etc. The human face is a dynamic object and has a high degree of variability in its appearance, which makes face detection a difficult problem in computer vision. The mission of a face detection system is to find and locate each human face in an image. However, face detection from a single image is a challenging task because of variability in scale, location, orientation (up-right, rotated), and pose (frontal, profile). Facial expression, occlusion, and lighting conditions also change the overall appearance of faces.

To meet all these requirements, an object detection system must involve two main steps (1) extracting features (Face representation), and (2) training a classifier using the extracted features to detect and locate faces. Indeed, for building an object detection system, we must rely on a more robust feature extraction system. The good feature set allows the detector to cleanly discriminate the human face even in cluttered

backgrounds. Hence, how to extract robust and discriminating features that can differentiate a face from a non-face, remains a central and difficult problem. A variety of representations of facial features have been offered in recent years namely HaarLike Features [1][20], which is the basis of the Viola and Jones algorithm which is the most known in the past two decades, LBP [2], SURF [3], HOG [4][18], NPD [5], Color and Skin color [6], Eigen face [7], etc.

The selection of robust and defining features is a major challenge for any face detection system. For example, the Viola and Jones algorithm chooses these features from a set containing thousands of features (160000) [1], and this has a great influence on the duration of the learning phase. The main idea of our algorithm is to exploit the divine measurements of a face [9], to manually determine these robust and determining features (HaarLike features). These measures consist of dividing the face into specific regions with well-known measures valid for all types of face. Then, we use Adaboost learning algorithm to determine the ideal correlation between these different features. This correlation constitutes our detection system.

The rest of the paper is organized as follows: In Section 2, we review the problem of feature selection, emphasizing different search and evaluation strategies. An overview of the proposed method is also presented. In Section 3 We present a brief explanation of the divine face proportions technique. Section 4 explores our approach in detail, and to prove the performance of our algorithm, section 5 presents some tests and experiments carried out during this work. Finally, we summarize this article with a conclusion.

2 Related Work

The first phase of a computer vision system is undoubtedly detection, and more particularly, face detection. A lot of research has been done on this subject. These searches can be grouped into four categories: knowledge-based methods, feature invariant methods, template matching methods, and appearance-based methods [6]. According to several studies including that done by [6], appearance-based algorithms are the most popular which use local feature representation and classifier learning. Among the most recent face detection algorithms that has gained increasing attention due to its remarkable results is that of Viola and Jones [1][19], which has been integrated into Intel Open Computer Vision library [8] with five haar-cascade classifiers. The amazing real-time speed and high detection accuracy of this face detector can be attributed to three factors: The first factor is the use of filters in rectangular form called HaarLike features with their integration function called Integral Image that is simple and fast (see figure 1). The second factor is the adaptation of the Adaboost algorithm to select the best features. The third factor is the representation of the features selected by Adaboost as a cascade, which they called Attentional cascade. In fact, this algorithm encourages several researchers to invest themselves in the search for other solutions for an ideal face detector, and especially in the axis concerning the extraction of features [15][16]. Indeed, several other types have been proposed as Local Binary Pattern (LBP) [2], SURF [3], HOG [4], and NPD [5]. Although some of

the features cited above offer significant discriminatory qualities than the HaarLike features for face detection, they generally increase the computational cost [5].

All these methods extract the features without being wary of their relevance, it is during the learning phase that the significant features will be chosen automatically, and this wastes the learning time and the chosen features are not necessarily relevant. This issue is our motivation to develop this work. The objective is to propose a system allowing to choose relevant and significant features without going through a learning algorithm and this based on the facial proportions [9,10,11] that thanks to it we can easily divide a face into well-known regions. and invariant whatever the face, then represent them by the Haar-Like features illustrated in figure 1.

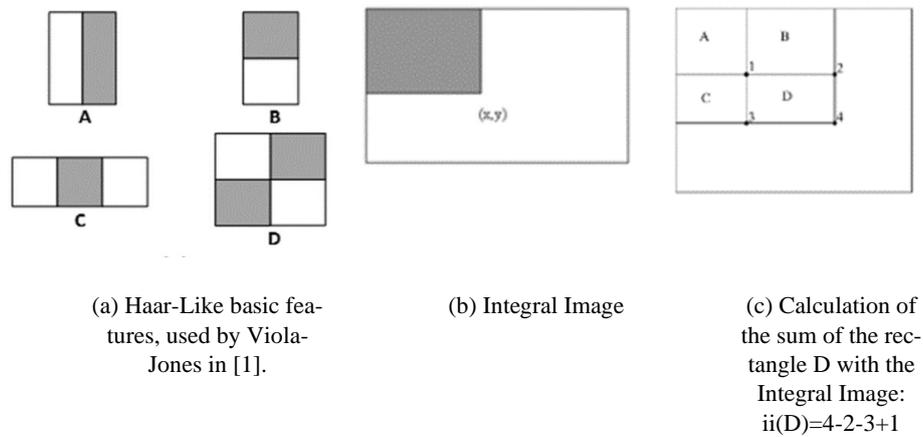


Fig. 1. Haar-Like features and Integral Image calculation

3 Facial Proportions

Several centuries ago, certain mathematical rules were discovered which would govern the mathematical beauty of a person according to the proportions of his face. Moreover, it would even be possible to assess the symmetry as well as the level of mathematical perfection of a human face from these, so-called divine proportions. Among the first to address this subject Leonardo da Vinci in his writings on human anatomy [11].

Aesthetic medicine specialists, namely orthodontists and maxillofacial surgeons use its proportions for the treatment and analysis of the face and for the planning of surgical rejuvenation.

In the world of aesthetic medicine, the face is divided, horizontally into three symmetrical parts (Figure 2 (a)), and vertically into five symmetrical parts (Figure 2 (b)) [9,10,11]. The three parts of the horizontal division are: The upper part extends from the trichion (point where the hairline meets the middle of the forehead) to the glabella, the middle part between the glabella and the sub-nasal and the last part ex-

tends from the sub-nasal to the chin. For vertical division, the width of each eye, intercanthal distance and nasal width all measuring one fifth (Figure 2 (b)). Well, in general and according to several criteria like, age, sex and ethnicity its parts are not always equal.

Golden ratio: The golden ratio or mysterious number also called Phi, is a ratio observed in nature, art, and architecture. Its approximate value is $\text{Phi} = 1.618$. It is also used to measure the beauty of the human face. It is considered a mysterious key to determine beautiful, ideal, and harmonious forms, it is also called divine proportion.

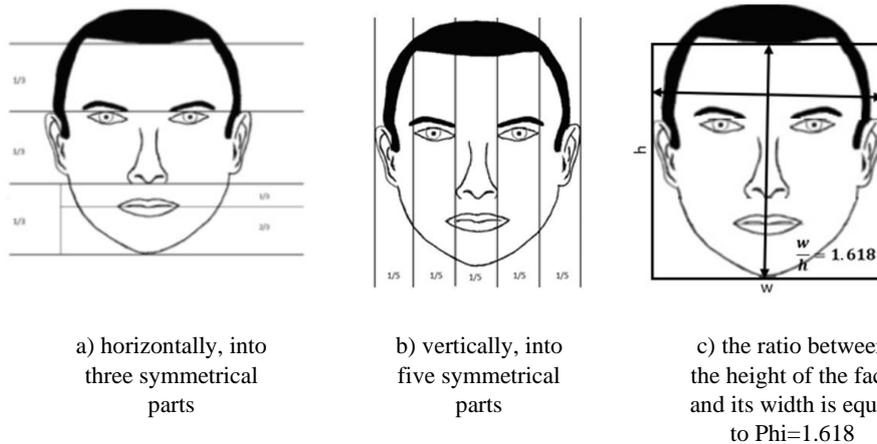


Fig. 2. The principle of face division

In this context, beauty specialists apply it to several proportions and structures of the face, such as the ratio of the width of the mouth to the width of the nose or the ratio of the height of the tooth to the width of the tooth, or the ratio between the height of the face and its width which is equal to 1.618 as shown in figure 2 (c). For this work, this is the latest report that interests us.

4 Building Face Detector with Face Proportions

The construction of our final detector goes through five main phases as shown in Figure 3. In the first phase, we build the training and test data from our dataset via the principle of data splitting, then each image undergoes preprocessing before to move on to the third phase, which consists in extracting the features determined manually in advance according to the rules of face proportions. Then the learning phase with Ada-boost which gives us as a result a detector with a high rate. The learning phase is repeated several times and the resulting detectors are stored to form the final detector. In the rest of this paper, each phase is discussed in detail.

4.1 Dataset splitting

In supervised learning, a prediction model is built to predict the results of an unknown target function. This target function is represented by a finite set T . T gathers pairs $[\vec{x}_i, \vec{d}_i]$ such that \vec{x}_i is an input data and \vec{d}_i its desired output: $T = \{[\vec{x}_1, \vec{d}_1], \dots, [\vec{x}_n, \vec{d}_n]\}$ where $n > 0$ is the number of ordered pairs of input/output samples [13].

In the ML world, Data splitting is a common strategy that involves taking all the data in the dataset and dividing it into two (see three [13]) learning and evaluation subsets. Usually with a ratio of 70% to 80% for learning and 20% to 30% for evaluation.

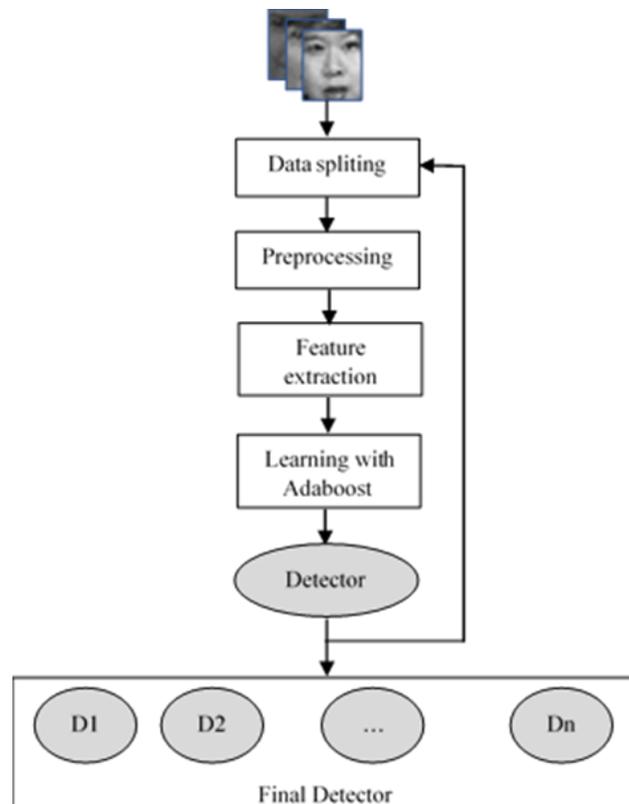


Fig. 3. Main steps for building a face detection system using face proportions

Our goal is to create a robust and reliable detector with a high detection rate, for this purpose our system performs repetitive training and with each iteration, it divides the dataset, randomly, into two sets namely learning and validation. The result of each division operation will be a detector which will participate in the formation of the final detector.

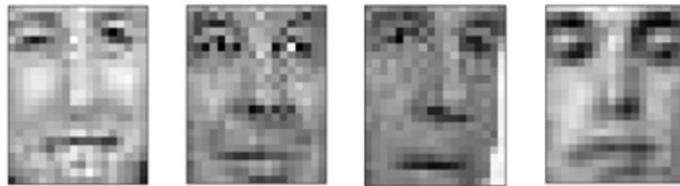
4.2 Preprocessing

The image of a face may have low contrast and may also have uneven lighting caused by the position of light sources. All of these can affect feature extraction. This is because we improve the image of a face and we remove noise by filtering the image with a Gaussian low pass filter. This filter has shown its effectiveness in the preprocessing of the faces.

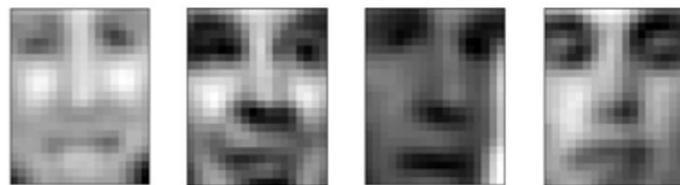
Figure 4 shows faces before application of the Gaussian filter and others after application. The characteristics of a face, namely the eyes, nose, and mouth, are very clear and can play an essential role in the classification phase.

4.3 Features extraction

Principle of dividing the face into regions: Our idea is based on the principle of dividing a face according to face proportions, shown in figure 2. According to several studies this division is almost perfect and valid for all face types with some differences in the measurements of each region [9]. So, the idea revolves around two axes; The first is the division of the image into small regions (boxes) according to the principle shown in section III. The second is to extract information from these portions, but only the most critical ones, such as the eye portion, the nose portion, and the mouth portion, using haar-like features. Figure 5 (a) shows the area, in gray, which contains the determining portions, this area has a height of 23 and a width of 18.



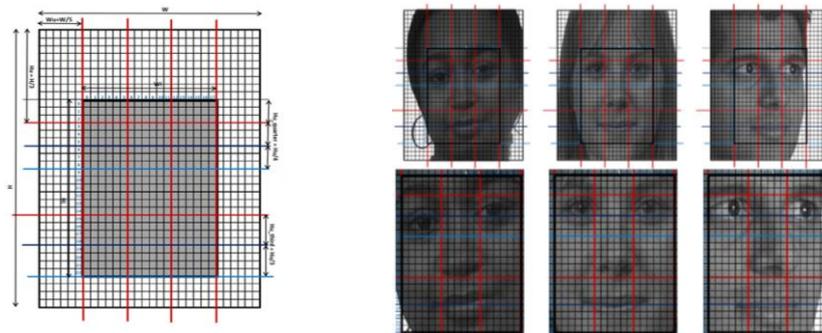
a) Original images with size 18x23.



b) Application of the Gaussian filter to the images of (a)

Fig. 4. Application of the Gaussian filter

If we consider an image with a given size, then this image is classified as a face if the determining portions correctly understand the characteristics of the face (eyes, nose and mouth). Indeed, instead of looking for the existence of these characteristics in the whole face, as Viola and Jones do in their algorithm, our system only focuses on a well-defined area and in this way, we minimize the cost and also the resources used in the learning phase. Instead of playing with thousands of features during the learning phase, our system allows this number to be minimized down to only 32 because we know where to look, exactly, each element (eyes, mouth, nose).



The first image (a) shows the outline of a face with the different measurements adopted by our system.

In (b), the first row shows face images cropped at 30x36 size. The second line illustrates the area of the determining portions of size 18x23.

Fig. 5. These examples of images from the FERET database, show clearly that whatever the face, the eyes, nose and mouth are always in the same portions.

Each image of the dataset must be cut, manually, in a size of 30x36 (Figure 5 (a)) so that each portion contains the correct element (eyes, nose, mouth...) and respecting the division rules shown in section III. Then each image will be cropped, but this time automatically, into a 18x23 image that contains only the invariant characteristics of the face. The choice of size is made in such a way to facilitate the calculations resulting from the operation of the division; 36 for the height because this number is divisible by 3 and also by 4, then $36/3 = 12$ is also divisible by 3. Finally, 30 for the width because it is divisible by 5. These dimensions are detailed in the table 1.

Table 1. The dimensions adopted to cut and divide our face images

Dimension	Value
W	30
H	36
Wi	18
Hi	23
Wu	W/5=6
Hu	H/3=12
Hu_third	Hu/3 =4
Hu_half	Hu/2=6
Hu_quarter	Hu/4 =3

Haar-Like features: The Haar-Like features, proposed by Viola and Jones, identify faces among several other objects in a given image. Viola and Jones [1] used three types of features, as shown in Figure 1 (a); two-rectangle feature, three-rectangle feature and four-rectangle feature. A HaarLike feature contains white and black areas, and its value is defined by the difference between the pixel values of the white areas and those of the black areas. The value of a Haar-Like feature reflects the change in the gray scale of the image. For example, some characteristics of the face can be described simply as rectangular features; the eyes are darker than the cheeks, the sides of the nose are darker than the bridge of the nose, and the mouth is darker than the surrounding color. Figure 6 shows how our system exploits these advantages.

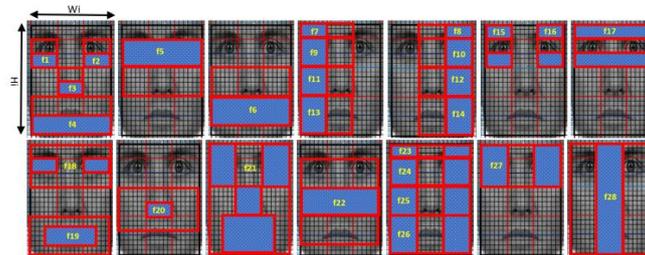


Fig. 6. The different Haar-Like features applied to different regions of the face.

The main motivation for using these features rather than pixels, for face detection, is that they are much faster to compute using a new image representation called Integral Image. The Integral Image of a given point in the image is equal to the sum of all the pixels located in the upper left corner of the point, as shown in Figure 1 (b). Let i be an image of dimension $M \times N$, $i(x,y)$ is the intensity of pixel (x,y) of image i . The Integral Image is another image ii such that for each pixel (x,y) :

$$ii(x,y) = \sum_{x' \leq x} \sum_{y' \leq y} i(x',y') \quad (1)$$

Then, the sum of the pixels of all the regions of the image can be obtained by a simple scan of the image, which considerably improves the efficiency of calculating

the eigenvalue of the image. Therefore, the integration of each rectilinear rectangle $R = [(x,y),w,h]$ is calculated only in four references (see Figure 1 (c)):

$$ii(R) = ii(x + w, y + h) - ii(x, y + h) - ii(x + w, y) + ii(x, y) \quad (2)$$

Such as (x,y) is the upper left corner of R , w and h represent, respectively, its width and height.

Features extraction: For face detection, the face representation must capture not only the entire appearance of a face, but also the distinct appearance of different parts of the face and the contextual appearance. And we believe that such a rich representation would help the face detector to improve performance under complex conditions. To achieve this, each part of the face, according to the proposed model, is represented by one or more basic features offered by Viola and Jones.

Our model uses 32 features divided into three groups. Features $f1$ to $f21$ are extracted, directly, from the image according to the principle of Viola and Jones. $f22$ to $f28$ represent features whose values can be deduced from the features of the 1st group. This type of features can be represented by Haar-Like features exactly like those of the 1st group. Figure 6 illustrates these different types and Table 2 shows the equations making it possible to calculate the values of the features of the 2nd group. The third group has four features $f29$ to $f32$ deduced from $f1$, $f2$, $f15$ and $f16$, but which cannot be represented with the Haar-Like features (See table 2).

Table 2. The distribution of the different features and the way to calculate their value.

Group	Features	Value
1	$f1$ to $f21$	Integral Image (Viola & Jones)
2	$f22$ to $f28$	$ii(f22) = ii(f5) + ii(f6)$ $ii(f23) = ii(f7) + ii(f8)$ $ii(f24) = ii(f9) + ii(f10)$ $ii(f25) = ii(f11) + ii(f12)$ $ii(f26) = ii(f13) + ii(f14)$ $ii(f27) = ii(f23) + ii(f24)$ $\quad = ii(f7) + ii(f8) + ii(f9) + ii(f10)$ $ii(f28) = ii(f27) + ii(f25) + ii(f26)$
3	$f29$ to $f32$	$ii(f29) = \text{abs}(ii(f1)) - \text{abs}(ii(f2))$ $ii(f30) = \text{abs}(ii(f15)) - \text{abs}(ii(f16))$ $ii(f31) = ii(f1) / ii(f2)$ $ii(f32) = ii(f15) / ii(f16)$

4.4 Learning with adaboost

The AdaBoost algorithm, developed by Freund and Schapire in 1995 [12], is used to improve the robustness of a weak learning algorithm. Its principle consists in combining a set of weak classifiers to form a more efficient classification function (strong classifiers). A classifier is said to be weak in the language of Boosting when it does not have good classification performance. A weak classifier can only classify learning base data correctly at 51% [1]. For each feature, a weak classifier determines the optimal threshold classification function, so that the minimum number of examples is

misclassified. A weak classifier h_j consists of a feature f_j , and a threshold θ_j (formula 3):

$$h_j(x) = \begin{cases} 1 & \text{if } p_j \cdot f_j(x) < p_j \cdot \theta_j \\ 0 & \text{otherwise} \end{cases} \quad (3)$$

The main idea of the AdaBoost algorithm is since T iterations we select T weak classifiers $h_t(x)$ each with a weight α_t to finally obtain a strong classifier $f(x)$ defined by the following expression:

$$f(x) = \sum_{t=1}^T \alpha_t h_t(x) \quad (4)$$

The strong classifiers trained by AdaBoost are applied sequentially, and each of them decides to accept or reject. If the window has been classified positive on one stage, containing the sought object, it will be processed by the following classifier. On the contrary, if it is negative, it is definitively excluded from the study, and will therefore not be processed by the following classifiers. A window which crosses all the stages will therefore be considered as containing the sought object. This principle is called by Viola and Jones; the Attentional Cascade. Figure 7 clearly explains this operation for a 3-stage cascade.

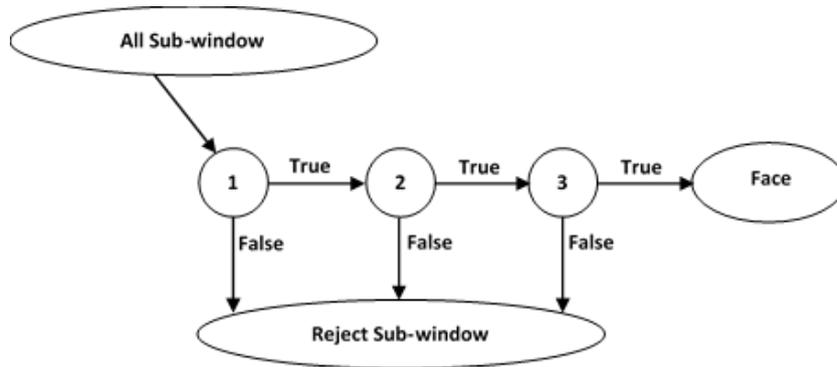


Fig. 7. Schematic description of the cascade classification tree. [1].

Our principle: The principle of our learning algorithm consists in repeating the learning with AdaBoost several times and at the level of each iteration, the algorithm randomly divides the dataset into two sets, namely learning and validation. Then, it is verified that the detection rate, the precision, and the recall of each detector exceed a threshold fixed in advance, if applicable, it will be added to the final detector. Algorithm 1 details this system.

Algorithm 1 Learn Final Detector	
Require:	<i>ds</i> : learning dataset. <i>min_rate</i> : The minimum detection rate. <i>min_precision</i> : the minimum precision allowed. <i>min_recall</i> : the minimum recall allowed. <i>n</i> : The number of detectors to choose from.
Ensure:	
Initialization:	$i \leftarrow 0$
Do	$ds_i = \text{split}(ds)$ $di = \text{Adaboost}(ds_i)$ if $(\text{rate}(di) \geq \text{min_rate} \text{ AND } \text{precision}(di) \geq \text{min_precision} \text{ AND } \text{recall}(di) \geq \text{min_recall})$ - add(<i>di</i>) to <i>Df</i> - increment <i>i</i>
end if	
While	$(i < n)$
Return	$D_f = \{d_1, d_2, \dots, d_n\}$

The final detector $D_f = \{d_1, d_2, \dots, d_n\}$ comprises *n* detector d_i . Each d_i detector is the result of the Adaboost algorithm applied to the dataset with a given splitting operation. This detector is added to the final detector if its detection rate exceeds the minimum *min_rate* fixed in advance and its precision and recall exceed a given value. The number *n* of the detectors elected is fixed by the programmer.

This repeating characteristic shows the robustness of our feature extraction system. Indeed, the number of iterations performed by our algorithm to choose *n* detectors is always closer to this number *n*. Figure 8(a) shows the variation in the total number of iterations, to choose 20 detectors, for 100 final detectors. We notice that the number of iterations varies between 20 and 36, but also according to figure 8(b) the most realized values vary between 20 and 25. This clearly shows that the detectors comprising the final detector have detection rates high.

To detect a face in a given image with our final detector, we opted to use the majority vote prediction. In this principle, we combine the labels of the predicted classes of each individual detector, d_i , and select the label of the class that received the most votes. In fact, in our case, we are talking about the binary classification where class1 = -1 and class2 = +1, therefore, we can write the majority vote prediction as follows:

$$D_f(x) = \text{sign}[\sum_{i=1}^n d_i(x)] = \begin{cases} 1 & \text{if } \sum_{i=1}^n d_i(x) \geq 0 \\ -1 & \text{otherwise} \end{cases} \quad (5)$$

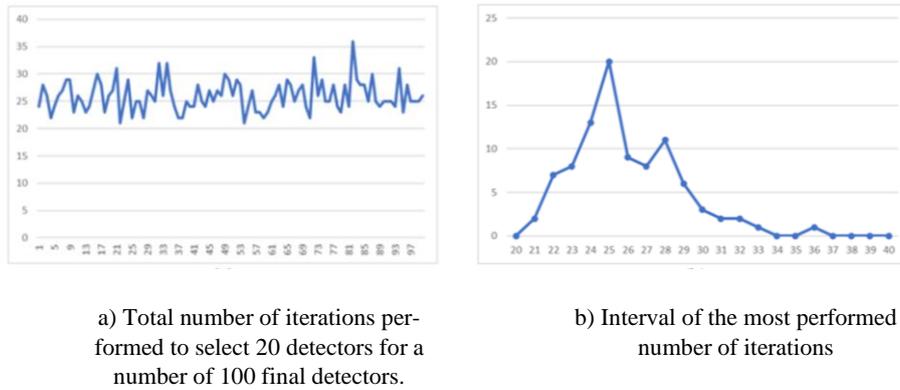


Fig. 8. Variation of the total number of iterations.

5 Experiments and Results

5.1 Dataset

The image databases used to evaluate the performance of face detection algorithms are very numerous and varied. For this project, we have chosen to use the Color FERET [14] face database thanks to its diversity of images. The Color FERET database is an expansion with color images and higher resolution of the original FERET database in gray tones. This database allows the study of a wide range of acquisition conditions as images are acquired, for everyone, in various poses, levels of illumination, as well as in the presence of expression or not. The base is made up of various ethnicities, which makes it possible to assess the performance of the system against a diverse population. In the context of this work, we cut, manually, the images from this database according to the measurements exposed in section IV.C. The result is a 30x36 sized image subbase. From this subset, we created 200 positive images of size 18x23 which we used to train our final detector. Figure 9 shows a sample of the positive images used by our system.



Fig. 9. Example of positive set of images.

5.2 Feature analysis

The choice of features was not made at random. Indeed, we based ourselves on the invariant characteristics of a face, namely the eyes, nose, mouth, eyebrows, cheeks, and the combination of two or more of these characteristics as shown by feature f21. To measure the performance of our 32 selected features, we used two more well-known methods, namely feature importance property and Correlation Matrix with Heatmap.

Figure 10 shows the importance of our features. We can clearly see that the features extracted directly from one of the invariant characteristics like the eyes and the eyebrows are more determining, while those extracted from the nose and the mouth are less important because of the different positions of the faces used in our dataset. For example, the nose is not always in the same portion.

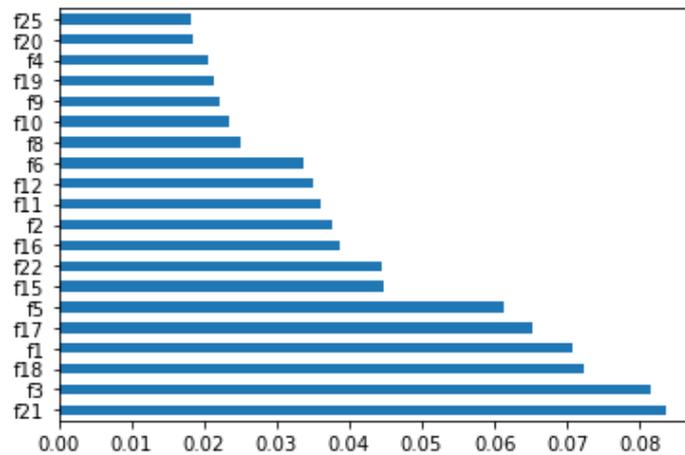


Fig. 10.Feature importance value.

Figure 11 shows that the main features of group 1 present a weak collinearity while the features of group 2 present a high correlation with those of group 1, which gives the impression that they illustrate redundancies but our experiments show that the addition of these features (group 2) increases the performance of our final detector.

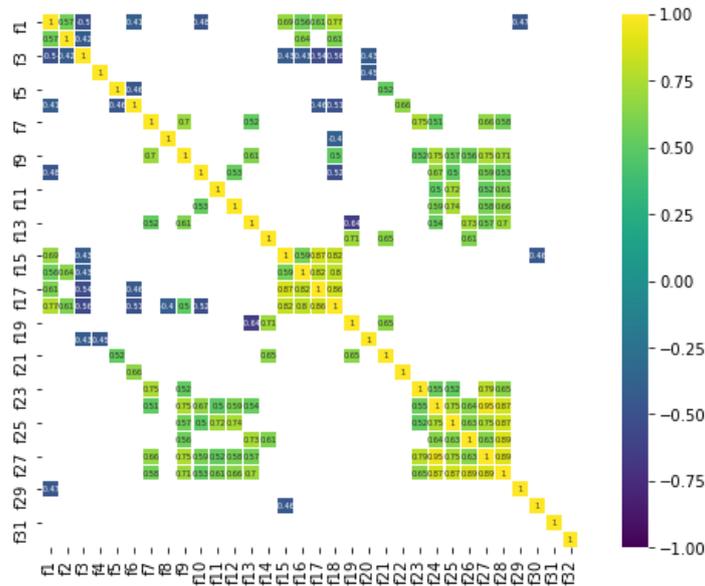


Fig. 11.Dataset Attributes Correlation Heatmap.

5.3 Performance and results

To test our algorithm, we created 100 final detectors which we tested and rated their performance using a test basis in this regard. This test database includes images collected from the web. It contains 101 images of celebrity faces and 71 negative images that do not contain faces. Figure 12 shows the detection rate of these different detectors. We can see that this rate is high and proves the performance of our approach. Table 3 shows the averages of the different measurements showing that our approach is promising and will give impressive results in face detection.

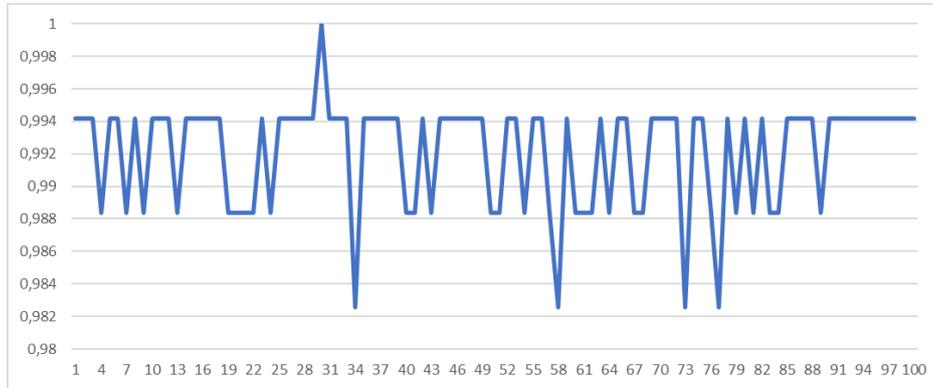


Fig. 12. Detection rate of 100 final detectors.

Table 3. The averages of the different performance measures

Accuracy	Recall	Precision	f_measure
0,9921512	0,9978218	0,9889241	0,9933453
TN (%)	FP (%)	FN (%)	TP (%)
98,4	1,6	0,2	99,8

To have performance tests comparable to those carried out by Viola & Jones [1], we used the MIT-CMU test base [17] which consists of 117 images and 511 faces. Viola and Jones used 130 images and 507 faces. A difference that we overlooked as most of the images have been the same or have had a few slight changes.

Table 4. Comparison with the Viola & Jones algorithm.

	Detection rate	False alarm rate	Detection speed
Our Detector	97%	18%	0.57s
VIOLA & JONES	94%	33%	0.7s

Figure 13 shows some examples of detection using sample images from the MIT-CMU test database.



Fig. 13. Some examples of detection.

6 Conclusion

In this article, we have introduced a face detection algorithm, based on Haar-like features and their application to specific regions of a face. These regions are deduced by applying the principle of face proportion. The detection rate is 97%, and the average of false positives is 18% based on MIT-CMU test. Due to the process of cutting faces according to the rules we presented above, we are not able to train our final detector on a larger number of training images, which may improve the performance of our detector. In the later work we will prepare a dataset with more images and test our algorithm with other learning algorithms (SVM, Decision Tree, etc.).

7 References

- [1] Viola, P., Jones, M.J. “Robust Real-Time Face Detection”. *International Journal of Computer Vision* 57, 137–154 (2004). <https://doi.org/10.1023/b:visi.0000013087.49260.fb>
- [2] H. Jin, Q. Liu, H. Lu, and X. Tong, “Face detection using improved LBP under bayesian framework”. In *Proceedings of the 3rd International Conference on Image and Graphics, 2004*.<https://doi.org/10.1109/icig.2004.62>
- [3] J. Li, T. Wang, and Y. Zhang, “Face detection using SURF cascade”. In *ICCV BeFIT work-shop, 2011*.

- [4] Yang, H.C., Wang, X.A, “Cascade face detection based on histograms of oriented gradients and support vector machine”. In 2015 10th International Conference on P2P, Parallel, Grid, Cloud and Internet Computing (3PGCIC), pages 766–770, 2015. <https://doi.org/10.1109/3pgcic.2015.14>
- [5] Shengcai Liao, Anil K. Jain, and Stan Z. Li, “A Fast and Accurate Unconstrained Face Detector”. Article in IEEE Transactions on Pattern Analysis and Machine Intelligence, August 2014, DOI: [10.1109/TPAMI.2015.2448075](https://doi.org/10.1109/TPAMI.2015.2448075).
- [6] S. Zafeiriou, C. Zhang, Z. Zhang, “A Survey on Face Detection in the wild: past, present and future”. Computer Vision and Image Understanding (2015), <https://doi.org/10.1016/j.cviu.2015.03.015>
- [7] M. Turk and A. Pentland, "Eigenfaces for Recognition", Journal of Cognitive Neuroscience, vol.3, no. 1, pp. 71-86, 1991. <https://doi.org/10.1162/jocn.1991.3.1.71>
- [8] Intel-Cooperation: Intel open-source computer vision library (OpenCV) (2011). Available at <http://sourceforge.net/projects/opencvlibrary/>.
- [9] Prendergast P.M. “Facial Proportions”. In: Erian A., Shiffman M. (eds) Advanced Surgical Facial Rejuvenation. Springer, Berlin, Heidelberg, 2012. https://doi.org/10.1007/978-3-642-17838-2_2
- [10] [10] J. Milutinovic, K. Zelic, N. Nedeljkovic, "Evaluation of Facial Beauty Using Anthropometric Proportions", The Scientific World Journal, vol. 2014, Article ID 428250, 8 pages, 2014. <https://doi.org/10.1155/2014/428250>
- [11] Hashim PW, Nia JK, Taliercio M, Goldenberg G. “Ideals of facial beauty”. Cutis. 2017 Oct;100(4):222-224. PMID: 29136056.
- [12] Y. Freund et R. Schapire, “A decision-theoretic generalization of on-line learning and an application to boosting”, Journal of Computer and System Sciences (JCSS), vol. 55, no 1, p. 119-139, 1997. <https://doi.org/10.1006/jcss.1997.1504>
- [13] Z. Reitermanov, “Data Splitting”. In WDS'10 Proceedings of Contributed Papers, Part I, 31–36, 2010, ISBN 978-80-7378-139-2.
- [14] National Institute of Standards and Technology. (2003) The Color FERET Database [Online]. Available: <https://www.nist.gov/programs-projects/face-recognition-technology-feret>.
- [15] M. Oualla, K. Ounachad, A. Hnini, A. Sadiq, “The Fast Integration of a Rotated Haar-Like Feature for Face Detection”. In International Journal of Advanced Trends in Computer Science and Engineering, 9(3):4055-4062, 2020, <https://doi.org/10.30534/ijatcse/2020/232932020>
- [16] M. Oualla, A. Sadiq, S. Mbarki, “A new approach to represent rotated haar-like features for objects detection”. In Journal of Theoretical & Applied Information Technology (JATIT), 78(1) (10th August 2015), 15-23, URL:<http://www.jatit.org/volumes/Vol78No1/3Vol78No1.pdf>. <https://doi.org/10.1109/cist.2014.7016645>
- [17] K.K. Sung, T. Poggio. "Example-Based Learning for View-Based Human Face Detection", IEEE PAMI, Vol. 20, No. 1, pp. 39-51, 1998. <https://doi.org/10.1109/34.655648>
- [18] Q. Tian, L. Zhang, Y. Wei, W. Zhao, W. Fei, “Vehicle Detection and Tracking at Night in Video Surveillance”, iJOE – Volume 9, Special Issue 6: "AIAIP2012", p. 61-64, July 2013. <https://doi.org/10.3991/ijoe.v9is6.2828>
- [19] W. G. Quilachamin, J. E. Sánchez, L. P. Concepción, “Kitchengam’ Criteria on the Use of Algorithms in a Person’s Pattern Detection, which Contribute to Safety, Surveillance and Energy Efficiency: Study of Art”, iJOE – Vol. 16, No. 7, p. 49-64, 2020. <https://doi.org/10.3991/ijoe.v16i07.14291>

- [20] W. Yimyam, M. Ketcham, “The System for Driver Fatigue Monitoring Using Decision Tree via Wireless Sensor Network for Intelligent Transport System”, *iJOE* – Vol. 14, No. 10, p.21-39, 2018. <https://doi.org/10.3991/ijoe.v14i10.7507>

8 Authors

Mohamed Oualla received his BS in Software Engineering from the Sciences and Technologies Faculty (FST), Moulay Ismail University, Errachidia in 1999, DESA in Computer Network and Telecommunication from the National School of Computer Science and Systems Analysis (ENSIAS), Mohammed V University, Rabat, Morocco in 2002 and PhD in 2017 from the Faculty of Science, Ibn Tofail University of Kenitra, Morocco. Currently, he is an Associate Professor in Computer Science with the Faculty of Science and Technologies (FST), Errachidia, Morocco. His research interests include image processing, pattern recognition, machine learning, big data and IoT. Email: m.oualla@umi.ac.ma

Khalid Ounachad received his DESS in Information System Engineering from The Faculty of Sciences Semlalia (FSSM) Cadi Ayyad University, Marrakech, Morocco in 2007. He is currently a PhD candidate in the Department of Computer Science in Faculty of Science, Ibn Tofail University, Kenitra, Morocco. His research interests include image processing, multimedia information retrieval, machine learning, sentiments analysis and artificial intelligence.

Abdelalim Sadiq received his BS in Software Engineering from the Sciences and Technologies Faculty, Moulay Ismail University, Errachidia in 1999, DESA in Computer Network and Telecommunication from the National School of Computer Science and Systems Analysis (ENSIAS), Mohammed V University, Rabat, Morocco in 2002 and PhD in Computer Science from the ENSIAS, Mohammed V University, Rabat, Morocco, in 2007. He is currently an Associate Professor in Computer Science, Department of Sciences Faculty, Ibn Tofail University, Kenitra, Morocco and team leader in Information System and Multimedia (SIM). His research interests include multimedia information retrieval and processing, sentiments analysis, IoT and data science. He has served as a reviewer for several international conferences and journals.

Article submitted 2020-10-11. Resubmitted 2021-01-15. Final acceptance 2021-01-19. Final version published as submitted by the authors.