



Facial Analysis Method for Pain Detection

Oana Subea^{1,2} and George Suciuc²

¹ Faculty of Electronics, Telecommunications and Information Technology,
University Politehnica of Bucharest, Bucharest, Romania

`oana.subea@beia.ro`

² R&D Department BEIA Consult International, Bucharest, Romania
{`oana.subea, george`}@beia.ro

Abstract. Facial expression recognition has been an active research topic for many years. In this paper a method for automatically recognizing pain intensity in images with facial expressions will be implemented. The method presented will contain a first step in which the face and the important points on the face will be located using the DLIB library. The second step consists of the calculation of HOG-type traits in order to describe the face found. The traits will be used to train a Random Forest (RF) regressor that will estimate the intensity of the pain. Training and testing will be done on the public UNBC-McMaster shoulder Pain Expression Archive database, using Python programming.

Keywords: Image processing · Facial recognition · HOG(Histogram of oriented Gradients) · Random Forest · DLIB

1 Introduction

Charles Darwin's book in 1872, entitled "Expression of Emotion in Man and Animals", has had a great impact on nowadays facial expression recognition technology. This area has been later studied by other scientists who have demonstrated that emotions are communicated in different facial and vocal ways, and this is available for every culture and species. Researchers in this field, such as Keltner, Ekman, Gonzaga, & Beer, reviewed the literature regarding how emotion can be expressed in facial features. Scherer, Johnstone, & Klasmeyer wrote about the expression of emotion by voice, the theme of language as a way in which people can express their emotions was approached by Reilly & Seibert, and Snowdon studied different displays amongst various nonhuman species.

It is a fact that one of the stronger and most immediate means of communication for individuals is the facial expression. Sometimes, human beings can show involuntarily their intentions or emotions through facial features and they can do that faster than they would using words [1].

20 years ago the situation of research and literature in terms of emotional expression was largely based on facial expression [2]. Keltner, Ekman, Gonzaga, and Beer showed that each emotion is characterized by a distinct facial expression. Also, the types of emotions on which these studies were based were limited [4].

The community of scientists became more interested in emotions and facial expression recognition in the last 10 years, the number of surveys on this topic increasing every year. Paul Ekman is the one who laid the foundation for the detection of facial expression by formulating in 1970 the “theory of basic emotions”. The basic emotions are the following: surprise, joy, fear, sadness, anger and disgust. This theory has subsequently led at the appearance of the Facial Action Coding System (FACS) which is based on 44 Action Units (AUs). These Action Units are labeled according facial muscles movements and more Action Units form a facial expression [3]. The Facial Action Coding System is presented in Fig. 1 [9]. In Fig. 2 are presented Upper Face Action units and some combinations.

Upper Face Action Units					
AU 1	AU 2	AU 4	AU 5	AU 6	AU 7
					
Inner Brow Raiser	Outer Brow Raiser	Brow Lowerer	Upper Lid Raiser	Cheek Raiser	Lid Tightener
*AU 41	*AU 42	*AU 43	AU 44	AU 45	AU 46
					
Lid Droop	Slit	Eyes Closed	Squint	Blink	Wink
Lower Face Action Units					
AU 9	AU 10	AU 11	AU 12	AU 13	AU 14
					
Nose Wrinkler	Upper Lip Raiser	Nasolabial Deepener	Lip Corner Puller	Cheek Puffer	Dimpler
AU 15	AU 16	AU 17	AU 18	AU 20	AU 22
					
Lip Corner Depressor	Lower Lip Depressor	Chin Raiser	Lip Puckerer	Lip Stretcher	Lip Funneler
AU 23	AU 24	*AU 25	*AU 26	*AU 27	AU 28
					
Lip Tightener	Lip Pressor	Lips Part	Jaw Drop	Mouth Stretch	Lip Suck

Fig. 1. FACS action units (AU). AUs with “*” indicate that the criteria have changed for this AU, that is, AU 25, 26, and 27 are now coded according to criteria of intensity (25A-E), and AU 41, 42, and 43 are now coded according to criteria of intensity









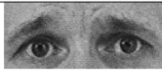






<i>NEUTRAL</i>	AU 1	AU 2	AU 4	AU 5
				
Eyes, brow, and cheek are relaxed.	Inner portion of the brows is raised.	Outer portion of the brows is raised.	Brows lowered and drawn together	Upper eyelids are raised.
AU 6	AU 7	AU 1+2	AU 1+4	AU 4+5
				
Cheeks are raised.	Lower eyelids are raised.	Inner and outer portions of the brows are raised.	Medial portion of the brows is raised and pulled together.	Brows lowered and drawn together and upper eyelids are raised.
AU 1+2+4	AU 1+2+5	AU 1+6	AU 6+7	AU 1+2+5+6+7
				
Brows are pulled together and upward.	Brows and upper eyelids are raised.	Inner portion of brows and cheeks are raised.	Lower eyelids cheeks are raised.	Brows, eyelids, and cheeks are raised.

Fig. 2. Upper face action units and some combinations

The method for estimation of pain intensity will be implemented using a database which contains facial images annotated with pain-specific expressions and action units. Part of this database will be used for the training of the classifier and the rest will help us to test the classifier.

Having an input image, we will be able to detect the face and its important facial points, in this way, in the area of the facial points of interest, we will calculate features using HOG (Histogram of Oriented Gradients) descriptor. The resulting traits will be used to train a classifier, such as Random Forest (RF). After the training, the method will be tested to see the result to different types of training (we will use random images).

As an application of this method, we will take real-time images from a webcam and we will display next to the detected face the corresponding action unit represented by an emoticon.

The rest of the paper is organized as follows: Sect. 2 describes related work in this area, like pain assessment in patients who aren't unable to verbally communicate with medical staff or face analysis for security and surveillance applications. Section 3 presents the principle of HOG (Histogram of Oriented Gradients) descriptor. Section 4 represents the description of the process, and Sect. 6 is the conclusion and future work of this project.

2 Related Work

One of the most challenging problem in patient care that medical staff experiences everyday is pain assessment in patients who are not able to verbally communicate. In [5] are described 2 Sparse Kernel Machine Algorithms used for pain assessment in infants using their facial expressions: Support Vector Machine Algorithm (SVM) and Relevance Vector Machine Algorithm (RVM). For this research, it has been used Infant Classification of Pain Expressions (COPE) database. This database contains 204 color photographs of 26 Caucasian neonates with age between 18 h and 3 days. From this database have been selected 21 subjects and for each of these subjects at least one photo corresponded to pain and one to non-pain.

The SVM algorithm is used to classify images which contains faces into two groups of “pain” and “non-pain” with accuracy in range of 82% and 88%. The Toolbox used for running the SVM classification algorithm is SVM MATLAB. The classification accuracy for this algorithm with a linear kernel was 90%.

When the RVM algorithm has been applied with a linear kernel to the same data set the obtained result was a classification with an accuracy almost identical, namely 91%, while the number of relevance vectors has been reduced to 2. The RVM algorithm can also be used to determine the posterior probability of the membership to a class of a test image.

Further, the results obtained using the RVM method were compared to human assessment. For measuring the agreement in the main intensity assessment between the human examiners and the RVM algorithm the weighted kappa coefficient has been used. This coefficient has a value of 0.48 for human experts and 0.52 for non-experts as compared with the RVM. This result shows that the agreement is moderate, and the classification is almost identical for binary classification [5].

Besides the importance that Face Analysis has in the medical field, it can be also very useful in other areas, such as security and surveillance, human-computer interaction, biomechanical applications, customized applications, etc.

In [6] it's presented a system framework of facial expression recognition for video surveillance purpose. The system is composed of four modules:

1. Face processing
2. Domain transformation
3. Feature extraction

Expression recognition

The first module, Face Processing, consists 3 steps. The first one is face detection which is performed by adopting the Viola-Jones face detection system. This face detector is used to select the desired face area. Next step is face cropping in order to obtain the required facial part of the image. The third and final process is scale normalization which is made in order to reduce the feature extraction area and to improve the execution time. This stage supposes the normalization of the cropped facial part to a window of size 128×128 pixels. For the scaling of the image, the method of the nearest neighbor is used.

Domain transformation involves applying the Discrete Wavelet Transform (DWT) in order to obtain local information by decomposing image into low and high frequency subbands. These subbands can be further combined with descriptors like HOG. The orientation information of the image is obtained by following combinations of scaling functions $\phi(x)$ and wavelet functions $\psi(x)$.

$$\bullet \quad \phi_{LL}(x, y) = \phi(x)\phi(y) \quad (1.1)$$

$$\bullet \quad \psi_{LH}(x, y) = \phi(x)\psi(y) \quad (1.2)$$

$$\bullet \quad \psi_{HL}(x, y) = \psi(x)\phi(y) \quad (1.3)$$

$$\bullet \quad \psi_{HH}(x, y) = \psi(x)\psi(y) \quad (1.4)$$

The feature extraction process is based on the use of a histogram of oriented gradients (HOG) feature. It counts the number of occurrences of gradient orientation in the local patch of an image. The gradient information changes meaningfully near edges and corners. At every pixel, the gradient has a magnitude G and a direction θ which is determined using LL subband (ϕ_{LL}) of DWT.

$$G = \theta \sqrt{\phi_{LL}^2(x) + \phi_{LL}^2(y)} = \tan^{-1} \frac{\phi_{LL}(y)}{\phi_{LL}(x)} \quad (1.5)$$

where:

- $\phi_{LL}(x)$ – is x – direction derivative of LL subband
- $\phi_{LL}(y)$ – is y – direction derivative of LL subband

The most important step is expression recognition module. The expressions are classified using a Support Vector Machine (SVM) classifier with OVA architecture. After feature computation, SVM classifier is trained and tested. For the recognition of expressions in the testing videos sequences, the training labels have been used. Experiments were performed in Matlab 8.3 environment. The results show that the proposed approach is effective towards recognition of angry, disgust, fear, happy, sad and surprised expression [6].

3 HOG Descriptor

A features descriptor is a representation of an image, or of a portion of an image, which simplifies the given image by extracting the useful information and by eliminating the unnecessary information. Usually, this kind of features descriptor converts an image with width \times length \times 3 dimensions, where 3 represents the 3 color channels, into a dimension vector. This vector can be further used for image recognition or objects detection in images.

Such a descriptor is also HOG (Histogram of Oriented Gradients). It can be used for images of any size, the only condition being that the images or the region of the image have a fixed aspect ratio. For example, for 1:2 report, the used image has to be 100×200 , 128×256 , 1000×2000 dimension. For this kind of descriptor, the input image has commonly $64 \times 128 \times 3$ dimension. In this case, the vector will have the length: $n = 3780$ [7].

3.1 The Principle of the Gradient Method

The principle of this method consists of defining contour points. These points represent the image's pixels whose gray levels present big changes. Therefore, we will use derivative gradient operators. If the input image has continuous space support, the derivative will have the maximum value in the direction of an edge. The linear combination of partial derivatives in horizontal and vertical directions represents the derivative of an image in the r direction, which forms θ with the horizontal direction.

$$\frac{\partial f}{\partial r} = \frac{\partial f}{\partial x} \frac{\partial x}{\partial r} + \frac{\partial f}{\partial y} \frac{\partial y}{\partial r} = \frac{\partial f}{\partial x} \cos \theta + \frac{\partial f}{\partial y} \sin \theta \quad (1.6)$$

$$\frac{\partial f}{\partial r} = f_x \cos \theta + f_y \sin \theta \quad (1.7)$$

The maximum value of this derivative is given by the equation below:

$$\frac{\partial}{\partial \theta} \left(\frac{\partial f}{\partial r} \right) = -f_x \sin \theta + f_y \cos \theta = 0 \quad (1.8)$$

The solution to the equation is:

$$\theta_0 = \tan^{-1} \left(\frac{f_y}{f_x} \right) \quad (1.9)$$

In the direction of the θ angle the gradient module is:

$$\left(\frac{\partial f}{\partial r} \right)_{max} = \sqrt{f_x^2 + f_y^2} \quad (2.1)$$

Practically, the implementation of this method involves the calculation of partial derivatives f_x and f_y , the calculation of the maximum gradient module and its direction, for each point of the image. The maximum value of the gradient that we have obtained will be compared to a fixed threshold. If the gradient is greater than the threshold the pixel for which we have calculated the gradient will be a contour pixel.

To calculate the partial derivatives f_x and f_y , we need first to make their translation in discrete time as follows.

$$f_x = \frac{\partial f}{\partial x} = \frac{\Delta f(m, n)}{\Delta m} \tag{2.2}$$

$$f_y = \frac{\partial f}{\partial y} = \frac{\Delta f(m, n)}{\Delta n} \tag{2.3}$$

These derivatives can be implemented in many ways using linear combinations of pixels in the image that are positioned in the vicinity of the current pixel with the coordinates (m,n). All these operations can be achieved by linear filtering using special masks. These masks are presented in Eqs. (2.5), (2.7), (2.9).

$$1. \quad \begin{cases} f_x = f(m, n) - f(m + 1, n) \\ f_y = f(m, n) - f(m, n + 1) \end{cases} \tag{2.4}$$

$$\text{The mask used : } \begin{cases} W_x = (-1 \quad \boxed{1}) \\ W_y = \begin{pmatrix} \boxed{1} \\ -1 \end{pmatrix} \end{cases} \tag{2.5}$$

$$2. \quad \begin{cases} f_x = f(m - 1, n) - f(m, n) \\ f_y = f(m, n - 1) - f(m, n) \end{cases} \tag{2.6}$$

$$\text{The mask used : } \begin{cases} W_x = (1 \quad \boxed{-1}) \\ W_y = \begin{pmatrix} 1 \\ \boxed{-1} \end{pmatrix} \end{cases} \tag{2.7}$$

$$3. \quad \begin{cases} f_x = f(m - 1, n) - f(m + 1, n) \\ f_y = f(m, n - 1) - f(m, n + 1) \end{cases} \tag{2.8}$$

$$\text{The mask used : } \begin{cases} W_x = (1 \quad \boxed{0} \quad -1) \\ W_y = \begin{pmatrix} 1 \\ \boxed{0} \\ -1 \end{pmatrix} \end{cases} \tag{2.9}$$

Because of the small size of the derivation masks, W_x , W_y , the results can be very easily affected by noise. In this case, it is necessary to combine the derivation filtering with smoothing filtering so that the noise effects will be reduced. If we consider that the noise is Gaussian and additive type, when we will apply the smoothing filter we will obtain a lower value of the contrast (the blurry effect). To minimize these effects, the mediation operation for smoothing filtering has to be in a perpendicular direction to that of the contours, more exactly, for vertical derivation, we will use a horizontal mask for the smoothing operation, and for horizontal derivation, we will use a vertical mask.

Another alternative, used in most cases, is the use of a weighted mediation for the smoothing process. This one allows us to give greater importance to the currently processed pixel. It is preferably to use the (2.9) type derivative operators. In this way, we obtain the following horizontal and vertical masks for the derivative operators.

$$W_x = \begin{pmatrix} 1 & 0 & -1 \\ c & \boxed{0} & -c \\ 1 & 0 & -1 \end{pmatrix} \tag{3.1}$$

$$W_y = \begin{pmatrix} 1 & c & -1 \\ 0 & \boxed{0} & 0 \\ -1 & -c & -1 \end{pmatrix} \tag{3.2}$$

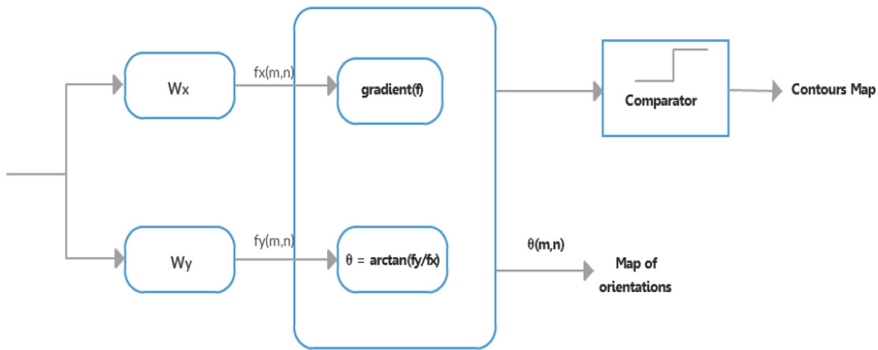


Fig. 3. Contour extractor based on the gradient method

Where: c = weight constant

Depending on the value of the weight constant we have several types of contour extraction operators: Prewitt ($c = 1$), Izotrop ($C = \sqrt{2}$), Sobel ($c = 2$).

In Fig. 3 is presented the diagram of the principle of contour extractor based on the gradient method.

3.2 The Orientations Map

An important part of this process it's the orientations map. It represents an image which contains, for each pixel, the orientation of the gradient with the maximum value of the module. It can be useful for further processing contours.

Contours map it's a binary image where the marked points correspond with the position of contour points (the points which have a gradient with a high value of the module) [8]. An example of a contours map using the HOG method is presented in Fig. 4 [11].

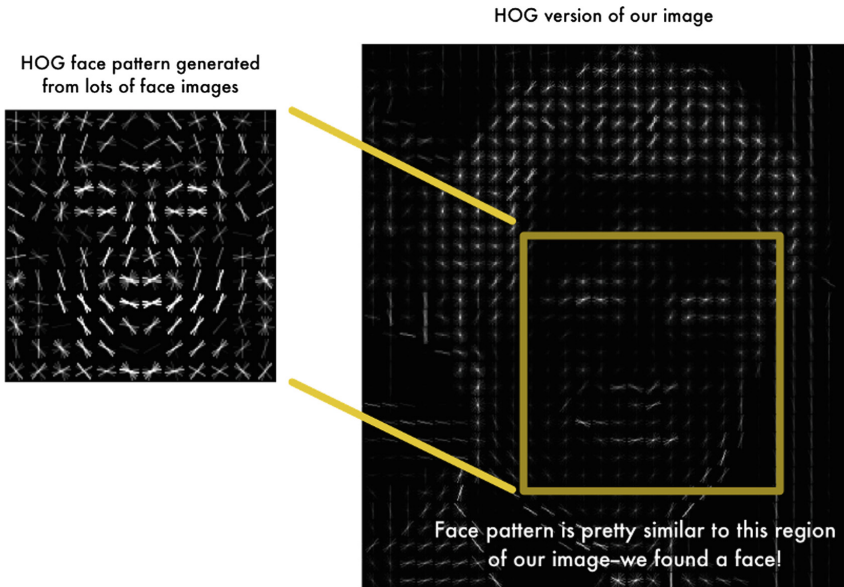


Fig. 4. The result of the HOG method

4 Experiments with FaceReader Software

FaceReader is a program for facial analysis. It can detect emotional expressions in the face. It can identify six basic expressions: happy, sad, angry, surprised, scared, disgusted, a neutral state. The program works with many independent variables: age and gender, ethnicity and facial hair. These variables can be automatically estimated from a video, camera input or an image.

The facial expression detection is made in three steps:

- Face finding – in this step is found the position of the face
- Face modeling – in this step the artificial face model is synchronized and used to describe the location of 500 key points and the texture of the face. The results are combined with those of the Deep Face algorithm for a higher classification accuracy.
- Face classification – in this step are presented the six basic expressions and a neutral state [13].

The results obtained by using FaceReader on different participants are presented in Figs. 5 and 6. In both situations the feeling with the highest weight that has been detected is happiness. Furthermore, heart rate detection can be performed using facial analysis [14].

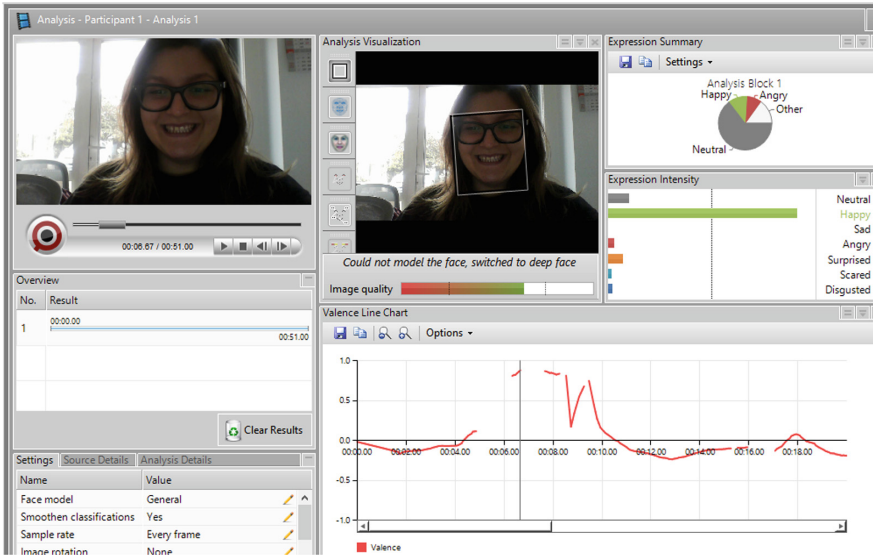


Fig. 5. Face analysis using FaceReader

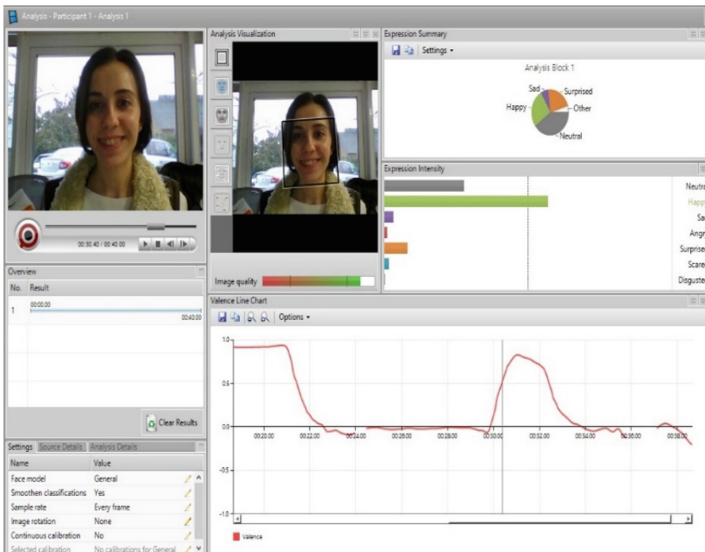


Fig. 6. Face analysis using FaceReader

The process of detecting emotions can become more complex by adding a stimulus. This stimulus can be a video, an image or a stimulus given by an external program. In this way, the emotion analysis can be done while the subject is watching this stimulus. For this experiment we used as stimulus three images, for which we set up a duration of 10 s. The results obtained using these stimuli are presented in Fig. 7.

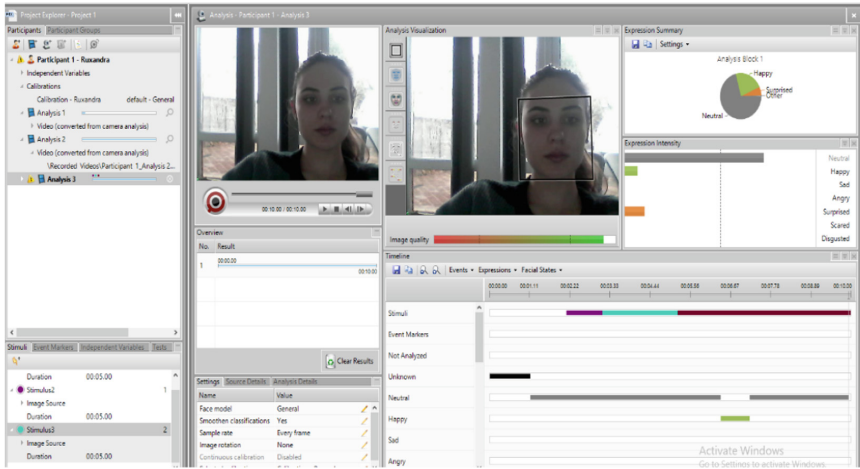


Fig. 7. Face analysis using FaceReader software after three stimulus have been applied

As is shown in Fig. 7, the emotions that have been detected on the face of the subject, after that we have applied three stimulus, are intuitively presented in a separate box under the images taken from the webcam. In most of the time the subject was neutral.

5 Evaluation of the Facial Analysis Process

This method involves the existence of a database which contains facial images annotated with facial expressions specific to pain, and with action units. We will use part of this database to train the Random Forest classifier and the other part to test this classifier. The operating principle of the Random Forest classifier is presented in the Fig. 8 [10].

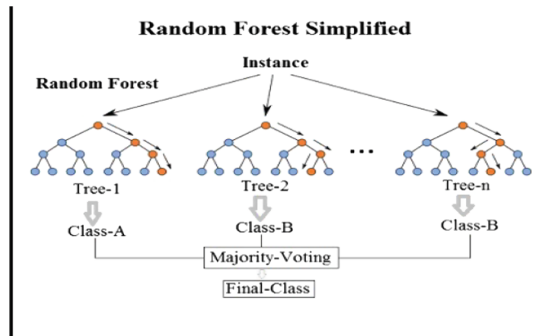


Fig. 8. The operating principle of random forest classifier

Firstly, we have to detect the facial points in the image. For facial landmarks detection, DLIB library will be used. The detection of facial landmarks represents a big challenge for shapes prediction process. Having an input image (and of course a Region Of Interest – ROI – which is specific for the object of interest) a shape predictor tries to detect the points of interest of the shape.

The detection of facial landmarks has 2 steps:

- Locate the face inside the image
- Detect the key facial-structures on the face from the region of interest (ROI)

The most important regions of the face are:

- The mouth
- Eyebrows
- The eyes
- The nose
- Maxillary

The facial landmarks detector is presented in Fig. 9 [12].

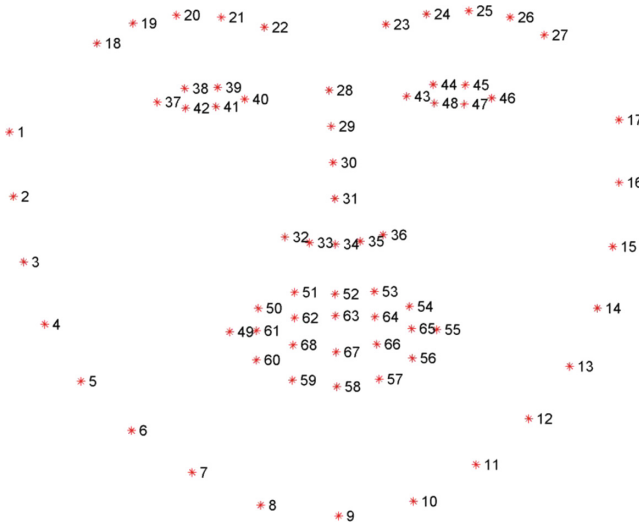


Fig. 9. 68 facial landmark coordinates from the iBUG 300-W dataset

These annotations are part of the set of 68 points iBUG 300-W. This set has been used to train the dlib Predictor.

After the facial points detection, we will use HOG descriptor to calculate the traits in the areas of the important points we have just detected. These traits will be used for the Random Forest classifier’s training. We will test further what happens and which are the results that we obtain for random images and different faces.

The results obtained with our program can be further compared with the results obtained using FaceReader software. In this way we can obtain the accuracy of this implementation.

The application will be implemented using the Python programming language. It involves taking real-time images from a webcam, and display on the screen, near the detected face, the correspondent action-unit and, eventually, an emoticon to illustrate the pain emotion. This method would be more intuitive for those who are not specialist in this domain. We can also use a video from the database where we can detect the faces, instead of using live images from the webcam. In Fig. 6 is shown the facial landmarks obtained using the DLIB library in a Python code. In Fig. 10 is presented the results obtained after applying facial landmarks detection on the picture.

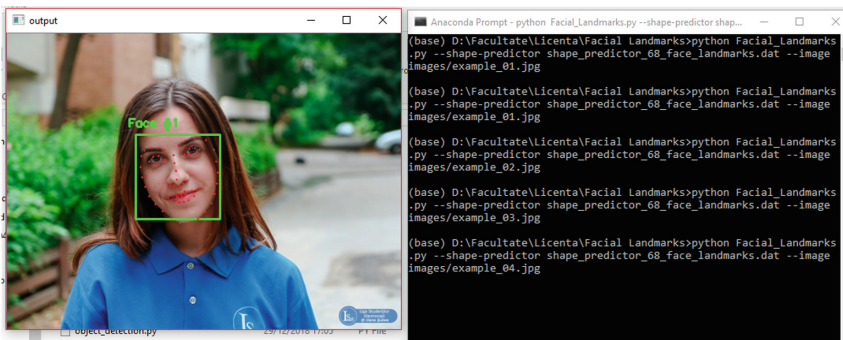


Fig. 10. Facial landmarks detection

6 Conclusion and Future Work

Facial expression recognition and the intensity of emotions have been an active research topic for many years. It is one of the most powerful and immediate means for human beings to communicate their emotions and intentions. The face can express emotion sooner than people verbalize or even realize their feelings. In this paper is presented a method for facial analysis and the estimation of pain intensity based on HOG descriptor, the Random Forest classifier and the UNBC-McMaster shoulder Pain Expression Archive database. Future research will focus on analyzing the influence of ambient light on the accuracy of the method.

Acknowledgment. This work has been supported in part by UEFISCDI Romania and MCI through projects VIRTUOSE, EmoSpaces and PAPUD, funded in part by European Union's Horizon 2020 research and innovation program under grant agreement No. 777996 (Sealed-GRID) and No. 787002 (SAFECARE).

References

1. Tian, Y.-L., Kanade, T., Cohn, J.F.: Recognizing action units for facial expression analysis. *IEEE Trans. Pattern Anal. Mach. Intell.* **23**(2), 97–115 (2001)
2. Zhang, D., et al. (Eds.) *Advanced Data Mining and Applications*. In: *Proceedings 8th International Conference, ADMA 2012, Nanjing, China, 15–18 December 2012* (2012)
3. Tornincasa, S., et al.: Department of management and production engineering, politecnico di torino. In: *3D Facial Action Units and Expression Recognition using a Crisp Logic* (2019)
4. Davidson, R.J., Scherer, K.R., Goldsmith, H.H. (eds.): *Introduction: Expression of Emotion, Handbook of Affective Sciences*. Oxford University Press, New York (2003)
5. Gholami, B., Haddad, W.M., Tannenbaum, A.R.: Agitation and pain assessment using digital imaging. In: *31st Annual International Conference of the IEEE EMBS Minneapolis, Minnesota, USA, 2–6 September* (2009)
6. Nigam, S., Singh, R., Misra, A.K.: Efficient facial expression recognition using histogram of oriented gradients in wavelet domain, Springer Science + Business Media, LLC, part of Springer Nature, pp. 5–10 (2018)
7. Histogram of Oriented Gradients. <https://www.learnopencv.com/histogram-of-oriented-gradients/>. Accessed 01 Feb 2019
8. Vertan, C., Ciuc, M., Zamfir, M.: *Analiza Imaginilor: Îndrumar de laborator*, pp. 32–35 (2001)
9. what-when-how, In *Depth Tutorials and Information*. <http://what-when-how.com/face-recognition/facial-expression-recognition-face-recognition-techniques-part-1/>. Accessed 21 Feb 2019
10. Medium. <https://medium.com/@williamkoehrsen/random-forest-simple-explanation-377895a60d2d>. Accessed 21 Feb 2019
11. StackExchange. <https://datascience.stackexchange.com/questions/21926/application-of-histogram-of-oriented-gradients-in-colored-image>. Accessed 21 Feb 2019
12. pyimagesearch. <https://www.pyimagesearch.com/2017/04/03/facial-landmarks-dlib-opencv-python/>. Accessed 26 Feb 2019
13. Nodus. <https://www.nodus.com/facereader/facial-expression-analysis>. Accessed 26 Feb 2019
14. Carmen, N., Poenaru, V., Suciú, G.: Heart rate measurement using face detection in video. In: *2018 International Conference on Communications (COMM)*, pp. 131–134. IEEE (2018)