Facial Expression Recognition: Comparison of Feature Extraction Methods

Diandra A. A. Kubo*, Olga R. P. Bellon† and Luciano Silva‡
IMAGO Research Group
Universidade Federal do Paraná
*daakubo@inf.ufpr.br, †olga@ufpr.br, ‡luciano@ufpr.br

Patrick Flynn
Computer Vision Research Lab
University of Notre Dame
flynn@nd.edu

***Abstract*—Human facial expressions are one of the most important communication channels, being used with trust to better understand one's state of mind in a variety of applications, for instance, emotion recognition. As a result, various algorithms and methods have been developed for facial expression recognition. On this context, we review the literature and conduct tests on different algorithms regarding facial feature extraction, in order to evaluate their performance on the BU-3DFE database. This database was chosen because it is widely used and all emotions are annotated for each image. Therefore BU-3DFE is suitable for the proposed benchmarking. The best result was achieved by a combination of Eigenfaces and SVM as classifier.**

I. Introduction

In 1872, Darwin established the principle of associated habits of the human expression, affirming that "Complex actions are of direct or indirect service under certain states of the mind, in order to relieve or gratify certain sensations, desires" [1]. This comment urges us to use expressions to understand someone's internal feelings.

In some occasions, the verbal expression, the main channel of communication, is not possible or not an option. In [2], it is emphasized that one of the most important aspects of nonverbal communication is the facial expression. The task of recognizing facial expressions has been transformed into recognizing one of the six basic emotions [3], that are: happiness, sadness, anger, fear, surprise, and disgust, as can be seen in Figure 1.

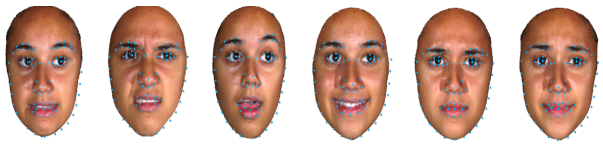

Fig. 1. Ekman's six basic emotions on the BU3DFE database [4]. From left to right: fear, disgust, surprise, happiness, anger, and sadness.

This task can help healthcare professionals in charge of neonatal [5] and adult [6] care, identifying the facial expression of pain. Likewise, is possible to measure someone's engagement while doing an assignment, which is frequently used by marking companies to get a feedback on advertising [7]. Moreover, recognizing facial expressions can be used to monitor a patient's well-being [8] or to assess the learning of children with autism [9].

In this work, we analyze the existing literature on automatic facial expression recognition, aiming to find the most used and relevant approaches in feature extraction, in order to compare different methods. In doing so, we hope to guide the future construction of a robust method that carries the strong points of different methods.

II. Concepts

The task of automatically classifying human facial expressions is composed of three main steps: face detection, features extraction from the face and expression classification. The expression can be classified as one of the six basic emotions defined by Ekman [3]. As this work focuses on the feature extraction step, a more in deep explanation of face detection can be found in [10], [11].

A. Convolutional Neural Networks

With Neural Networks, the goal is to mimic the human brain learning process, simulating neurons interactions. The idea is to reinforce a pathway when it is successful and weaken it when it is not, which is represented through weights and biases along the network [12]. The network is then composed by neurons that are arranged in layers and said layers are also connected. Each neuron produces the result of a chosen function on its input and passes it along to the next set of neurons it is connected to [13]. Due to these connections, a high nonlinearity can be expected from a neural network depending on the chosen architecture. When the data reaches the last layer, a class or value is reached and outputted.

Dealing with image processing, the inputs are, naturally, images, and images can become large vectors due to its 3-dimensionality for colored images. This can be a problem since at least the first layer of the network would have to hold 3 neurons for every pixel on an image. Furthermore, usually, in architectures such as the Multilayer Perceptron [**?**], all neurons of two connected layers must be connected, which makes the processing of images even more costly.

Thinking about this, the Convolutional Neural Networks (CNNs) were created, that already assume the input data are images, so the layers are already designed for them [14]. It also does not have only fully connected layers, because it takes into consideration the spatial relation of the pixels that should not be treated as random, as it is an image after all. Other than

fully connected layers, the other two main types of layers are convolutional and pooling layers.

On convolutional layers, instead of having a neuron for each pixel like in fully connected layers, we have a neuron representing the result of multiple kernels passed on the previous layer. In other words, you convolve a filter across the input, hence the layer name. These filters that are learned can be later used for feature extraction.

In the other hand, the pooling layers are used to reduce parameters and therefore reduce computation. The rule most used for these layers is the maximum, so for each block of the input being analyzed, only the maximum value passes to the next layer. A convolutional neural network is then defined mainly by the mixture of these types of layers and any other fitting the problem being solved.

B. Eigenfaces

The Eigenfaces of a set of images are nothing more than the eigenvectors from said images when they are treated as vectors instead of matrices.

This method was originally used for facial recognition [15], to differentiate identities. But it can also be used to differentiate facial expressions [16].

The first thing that needs to be done is to transform all images in vectors with the same dimension, so that a matrix with all of them can be created, with each row representing an image. On this matrix, the mean of every column is calculated, that can be seen as the mean image of the dataset. Every row (image) of the matrix has the mean values calculated subtracted. Following, the covariance matrix (Equation 1) is calculated on the original matrix M.

$$C = M^T M \tag{1}$$

From the covariance matrix, the eigenvectors can be extracted, according to the eigenvalues calculated. To do that, the Eigenvectors are then ordered according to their associated eigenvalues. An analysis can be done in order to check the desired number of eigenvectors needed so that the data representation is enough for the problem. Following the eigenvectors selection, the original matrix M containing the data is multiplied by each eigenvector, and the new resulting matrix is the new face space FS where all images from the training and test will be represented.

Finally, the matrix M with the images is multiplied by the face space FS, so they can be in this new representation (Equation 2).

$$P = FS^T \times M \tag{2}$$

C. Geometric Information

Extracting the geometrical information from a face means to somehow obtain the shape and position of facial components [2]. One way to achieve this can be seen on [17], where facial landmarks are detected and from them, a Delaunay triangulation [18] is done. This way, a mesh of triangles is computed on the face, with each vertice being a landmark. The internal angles of these triangles can also be calculated, representing the relationship among the landmarks, that deform depending on the facial expression.

Another way of extracting features from a face is to simply calculate the distance, usually Euclidian, between each pair of landmarks. This is also done with the intent of tracking changes on the face caused by a change in the facial expression. All distances must be normalized using a measure from each person, like the interocular distance, i.e. the distance from the outer points of the left and right eye.

D. Gabor Filters

Gabor Filters are linear filters usually used for edge detection but also for texture analysis, therefore it makes sense to use it in order to examine facial expressions. This filter is based on the visual cortex assimilation of images [19] and follows the Equation 3. An in deep explanation of the physics involved can be seen in [**?**].

$$g(x, y; \gamma, \theta, \psi, \sigma, \gamma) = exp\left(- \frac{x'^2 + \gamma^2 y'^2}{2\sigma^2} \right) cos\left(2\pi \frac{x'}{\lambda} + \psi \right) \tag{3}$$

The parameter theta (θ) defines the filter's direction. The sigma (σ) is the standard deviation of the used Gaussian function. Lambda (λ) defines the sinusoidal factor wave length. The gamma (γ) parameter represents the relation of the spatial aspect. Finally, the psi (ψ) is the phase offset.

After the feature extraction from the face on the image, it is necessary for this data to be classified in order to obtain the facial expression information. For the classification of a new context to be made, it is indispensable to have a previous process of context learning to have been done [13]. The SVM (Support Vector Machine) method has a goal to build hyperplanes that divide these contexts being learned in the best way possible. When these examples are linearly separable, the method search for a mapping function that projects the examples on a new data space so that they can be separated [20]. The optimum hyperplane will be the one that keeps an equal distance to all classes existing in the examples.

III. Related Work

There are a lot of ways to characterize a facial expression classifier, but as the focus of this work is feature extraction, it will be through this perspective we will analyze other work.

One of the first works in this area can be seen in [21], where the idea was to identify facial deformations caused by facial expressions analyzing changes from the neutral expression. This was done by calculating distances between previously selected points on a face. The classification was very unsophisticated, done by checking the changes on an empirically built table of distances.

In [22] a performance comparison between AdaBoost, support vector machines (SVM), and linear discriminant analysis (LDA) as classifiers is done. Features were represented as

Gabor filters. The best results were achieved by a combination of AdaBoost and SVM.

Other approach used in [23] deals with the problem of detecting the amount of deformation differently. They first estimate a person's neutral expression using Gaussian mixture models and then use the features remaining from the subtraction of the neutral face from the expressive one. The classification is made using SVM.

Gabor-wavelet labeled elastic graph matching approach of the von der Malsburg group and the Eigenface/Fisherface algorithm were used for feature extraction in [24], followed by LDA-based classification.

The work in [17] uses the geometrical information of a face by calculating the distances between facial landmarks and also the angles between triangles created through a Delaunay triangulation where each vertice is a facial landmark.

In [**?**], a Convolutional Neural Networks is used. The architecture proposed has 5 layers: 2 convolutional layers, 2 pooling layers and one fully-connected layer, and in this work they showed how a CNN deals better with translations on the input images.

All these reviewed works were the basis for the chosen methodology of this work.

IV. Methodology

The face detection, the first stage of facial expression analysis, was performed using the Viola-Jones method, that can be seen with a deeper explanation in [11]. After the obtention of the face location, the feature extraction by different methods was executed as follows

A. Feature Extraction

The first characteristic chosen for the analysis was the Gabor filter. Following what was said in Section II, the kernel size used was 30×30. The theta (θ), that defines the direction varied from 0 to π with a step of $\frac{\pi}{8}$. The sigma (σ) was defined as 4.0 and represents the standard deviation of the Gaussian function used. Following, the lambda (λ) was used with a 10.0 value, representing the sinusoidal wave length. Gamma (γ) was set to 0.5, which is the ratio of the spatial aspect. Finally, the psi (ψ) was set to 0.0, which is the phase offset.

The second set of characteristics were the Eigenfaces and according to the database size, the 150 first Eigenfaces were extracted from the covariance matrix. The third set was the characteristics extracted from a convolutional neural network and the architecture chosen was the state-of-the-art defined in [25].

As for the geometric information, at first the angles from the Delaunay triangulation were used, and later the Euclidian distances between facial landmarks. With 68 landmarks, this yielded 114 triangles and consequently 342 angles. Calculating the distance between each pair of landmarks, we also obtained 2278 distances.

B. Database

The facial expression database used was the BU-3DFE from the Binghamton University. This database contains images of 100 participants, 56 women, and 44 men. The participant's age ranges from 18 and 70 years old. Each participant performed the six basic emotions [4].

V. Results

On the initial performance analysis, the Euclidian distances and the angles from the Delaunay triangulation did not present satisfactory accuracy. As they are two characteristics used often in the literature and as the initial vectors were larger than the ones in said literature, a Principal Components Analysis (PCA) was made. After this transformation, a better result was obtained for the angles with only 200 components in the place of the original 342. The distances vector, originally composed of 2278 distances (a combination of all landmarks), was transformed to a vector of only 10 distances, a number consistent with the literature in facial expression as it can be seen in [5], [21]. The following results consider the angles and distances after the PCA transformation.

On Table I, the performance results of each method can be seen. The Eigenfaces achieved the best performance, but although it was the best, it only had an accuracy of 77.71%. For a characteristic alone combined with the SVM classifier, this result is relatively good, seeing as recent work focuses on methods that require much more training data and time. This result could be expected due to the nature of the data used here, as convolutional neural networks outperform methods like Eigenfaces on data from non-controlled environments.

On an attempt to improve accuracy, using the best method, the Eigenfaces, instead of using the whole face as an input, we divided the face into six regions: left eye, left eyebrow, right eye, right eyebrow, nose, and mouth. We then trained a PCA for each region (same procedure as for eigenfaces in the whole image) and in addition to that, calculated a weight for the expression vote for each region, seeing as some should be taken into account more than others (the mouth, for example). The results were not satisfactory, seeing as accuracy improved only 1%, and as the training time grew exponentially, we concluded it to not be worth it.

TABLE I
Methods performance by expression and mean performance. Values represent percentage of accuracy.

	Eigenfaces	**Angles**	**Distances**	**CNN**	**Gabor**
Disgust	67.1	78.2	73.1	71.4	69.1
Surprise	93.5	94.4	89.1	80.8	81.7
Anger	83.3	68.4	57.1	70.3	69.4
Fear	76.2	58.3	48.4	54.2	55.9
Sadness	63.7	57.3	42.5	68.9	57.7
Happiness	82.5	80.2	79.8	81.7	75.2
Mean Score	77.71	72.8	65.0	71.21	68.16

This is still an intrinsically hard problem. On Figure 2 this can be observed. On the top row, we have images classified

as Sadness, but the expression asked of the participants was Anger. One could easily say it was, in fact, Sadness. The same thing happened when the expression asked was Sadness, on the bottom row. These images were all classified as Anger. We see here there is a visible difficulty even for humans to make the distinction between them.

Fig. 2. Similar expressions that are labeled as different. Top row: Anger images classified as Sadness; bottom row: Sadness images classified as Anger.

VI. Conclusion

The facial expression recognition field has its limitation in part due to the fact that most databases are composed of only posed expressions, which is affected by two main facts: different people do not express emotions the same way and posed expression often differ from authentic ones. The first situation is influenced by the fact that different cultures can express the same emotion in a different way, but also people in the same culture with different experiences and influences can respond differently to the same stimulus. On the second situation, we have the dilemma that the training data usually follows a different distribution than the test data. In one hand, if the training data comes from a base with posed expressions, the model trained may not be able to work properly in a non-controlled environment application. In the other hand, if the training data also comes from a non-controlled environment, there is no way to know the correct labels for sure in order to train the model. To top of all of that, this field also suffers from the main problems that happen at the face detection stage, the first stage of the facial expression recognition pipeline.

The next step for this work is to build a robust method that aggregated all the strong points reviewed here in order to achieve better performance. Furthermore, we plan to use 3D images, since the database used here also has this information available, which makes the rendering and alignment easier.

Acknowledgment

The present work was made possible with the support of the Science Without Borders program from CAPES. The authors would also like to thank CNPQ for their support.

References

[1] C. Darwin, *The expression of the Emotions in Man and Animals* . Oxford University Press, 1872.

[2] Y.-L. Tian, T. Kanade, and J. F. Cohn, "Facial expression analysis," in *Handbook of Face Recognition*. Springer, 2005, pp. 247–275.

[3] P. Ekman and W. V. Friesen, "Constants across cultures in the face and emotion." *Journal of personality and social psychology*, vol. 17, no. 2, p. 124, 1971.

[4] X. Zhang, L. Yin, J. F. Cohn, S. Canavan, M. Reale, A. Horowitz, and P. Liu, "A high-resolution spontaneous 3d dynamic facial expression database," in *International Conference and Workshops FG*. IEEE, 2013, pp. 1–6.

[5] M. Schiavenato, J. F. Byers, P. Scovanner, J. M. McMahon, Y. Xia, N. Lu, and H. He, "Neonatal pain expression: Evaluating the primal face of pain," *Pain*, vol. 138, pp. 240–271, 2008.

[6] G. C. Littlewort, M. S. Bartlett, and K. Lee, "Automatic coding of facial expressions displayed during posed and genuine pain," *Image and Vision Computing*, vol. 27, no. 12, pp. 1797–1803, 2009.

[7] J.-U. Garbas, T. Ruf, M. Unfried, and A. Dieckmann, "Towards robust real-time valence recognition from facial expressions for market research applications," in *Humaine Association Conference on ACII*. IEEE, 2013, pp. 570–575.

[8] B. Lee, J. Chun, and P. Park, "Classification of facial expression using svm for emotion care service system," in *International Conference SNPD*. IEEE, 2008, pp. 8–12.

[9] T. Pradi, O. Bellon, L. Silva, and G. M. S. Dória, "Ferramentas de computação visual para apoio ao treinamento de expressões faciais por autistas: Uma revisão de literatura," in *Seminário Integrado de Software e Hardware*, 2016.

[10] P. Viola and M. Jones, "Robust real-time face detection," *International Journal of Computer Vision*, vol. 57, no. 2, pp. 137–154, 2004.

[11] A. G. F. Fior, M. P. Segundo, O. R. P. Bellon, and L. Silva, "Detecção e reconhecimento facial em sequências de vídeo," in *Revista Eletrônica de Iniciação Científica*. SIBIGRAPI, 2008, pp. 33–43.

[12] L. Fausett, *Fundamentals of Neural Networks: Architectures, Algorithms, and Applications*. Prentice-Hall, Inc., 1994.

[13] D. Michie, D. J. Spiegelhalter, and C. C. Taylor, "Machine learning, neural and statistical classification," pp. 84–101, 1994.

[14] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "Imagenet classification with deep convolutional neural networks," in *Advances in Neural Information Processing Systems*, 2012, pp. 1097–1105.

[15] M. Turk and A. Pentland, "Eigenfaces for recognition," *Journal of Cognitive Neuroscience*, vol. 3, no. 1, pp. 71–86, 1991.

[16] C. Padgett and G. W. Cottrell, "Representing face images for emotion classification," *Advances in Neural Information Processing Systems*, pp. 894–900, 1997.

[17] C. Fabian Benitez-Quiroz, R. Srinivasan, and A. M. Martinez, "Emotionet: An accurate, real-time algorithm for the automatic annotation of a million facial expressions in the wild," in *CVPR*. IEEE, 2016, pp. 5562–5570.

[18] D.-T. Lee and B. J. Schachter, "Two algorithms for constructing a delaunay triangulation," *International Journal of Computer & Information Sciences*, vol. 9, no. 3, pp. 219–242, 1980.

[19] S. E. Grigorescu, N. Petkov, and P. Kruizinga, "Comparison of texture features based on gabor filters," *Trans. Image Process.*, vol. 11, no. 10, pp. 1160–1167, 2002.

[20] L. H. Hamel, *Knowledge Discovery With Support Vector Machines*. John Wiley & Sons, 2011, vol. 3.

[21] M. Suwa, N. Sugie, and K. Fujimora, "A preliminary note on pattern recognition of human emotional expression," *IJCPR*, pp. 408–410, 1978.

[22] M. S. Bartlett, G. Littlewort, M. Frank, C. Lainscsek, I. Fasel, and J. Movellan, "Recognizing facial expression: Machine learning and application to spontaneous behavior," *CVPR*, vol. 2, pp. 568–573, 2005.

[23] S. Ulukaya and Ç. E. Erdem, "A hybrid facial expression recognition method based on neutral face shape estimation," in *SIU*. IEEE, 2012, pp. 1–4.

[24] M. J. Lyons, J. Budynek, and S. Akamatsu, "Automatic classification of single facial images," *Trans. Pattern Anal. Mach. Intell.*, vol. 21, no. 12, pp. 1357–1362, 1999.

[25] G. Levi and T. Hassner, "Emotion recognition in the wild via convolutional neural networks and mapped binary patterns," in *Proceedings of the ACM on International Conference on Multimodal Interaction*, 2015, pp. 503–510.