

# A Facial Expression Recognition System Using Convolutional Networks

Andre Teixeira Lopes, Edilson de Aguiar, Thiago Oliveira-Santos\*

Federal University of Espirito Santo, Brazil

\*Corresponding Author: todsantos@inf.ufes.br







Expression	Angry	Disgust	Fear	Happy	Sad	Surprise
						
Recognition Rate	85.19%	97.74%	85.33%	98.55%	79.76%	98.80%

Fig. 1. This work proposes a simple solution for facial expression recognition that uses a combination of standard methods, like Convolutional Network and specific image pre-processing steps. To the best of our knowledge, our method achieves the best result in the literature, 97.81% of accuracy, and takes less time to train than state-of-the-art methods.

**Abstract**—Facial expression recognition has been an active research area in the past ten years, with a growing application area like avatar animation and neuromarketing. The recognition of facial expressions is not an easy problem for machine learning methods, since different people can vary in the way that they show their expressions. And even an image of the same person in one expression can vary in brightness, background and position. Therefore, facial expression recognition is still a challenging problem in computer vision. In this work, we propose a simple solution for facial expression recognition that uses a combination of standard methods, like Convolutional Network and specific image pre-processing steps. Convolutional networks, and the most machine learning methods, achieve better accuracy depending on a given feature set. Therefore, a study of some image pre-processing operations that extract only expression specific features of a face image is also presented. The experiments were carried out using a largely used public database for this problem. A study of the impact of each image pre-processing operation in the accuracy rate is presented. To the best of our knowledge, our method achieves the best result in the literature, 97.81% of accuracy, and takes less time to train than state-of-the-art methods.

**Keywords**—Facial Expression; Convolutional Networks; Computer Vision; Machine Learning; Expression Specific Features;

## I. INTRODUCTION

Facial expression is one of the most important features of human emotion recognition [1]. It was introduced as a research field by Darwin in his book "The Expression of the Emotions in Man and Animals" [2]. According to Li and Jain [3], it can be defined as the facial changes in response to person's internal emotional state, intentions, or social communications. Automated facial expression recognition has a large variety of applications nowadays, such as data-driven animation, neuromarketing, interactive games, entertainment, sociable robotics

and many others human-computer interaction systems.

Expression recognition is a task that humans perform daily and effortlessly [3], but that is not yet easily performed by computers. A lot of work has tried to make computers reach the same accuracy as humans, and some examples of these works are highlighted here. Facial expression recognition systems can be divided in two main categories, those that work with static images [4], [5], [6] and those that work with dynamic image sequences [7], [8]. Static-based methods do not use temporal information, i.e. the feature vector comprises information about the current input image only. Sequence based methods, in the other hand, use temporal information of images to recognize the expression based on one or more frames. Automated systems for facial expression recognition receive the expected input (static image or image sequence) and give as output one of the six basic expressions (anger, sad, surprise, happy, disgust and fear, for example). Some systems also recognize the neutral expression. This work will focus on methods based on static images and it will consider the six basic expressions only.

As described in [3], automatic facial expression analysis comprises three steps: face acquisition, facial data extraction and representation, and facial expression recognition. Face acquisition can be separated in two major steps: face detection [9], [10], [11], [12] and head pose estimation [13], [14], [15]. After the face is located, the facial changes caused by facial expressions need to be extracted. These changes are usually represented as geometric feature-based methods [16], [17], [18], [12] or appearance-based methods [16], [6], [19]. Geometric feature-based methods work with shape and location of facial components like mouth, eyes, nose and eyebrows. The

feature vector that represents the face geometry is composed of facial components or facial feature points. Appearance-based methods work with feature vectors extracted from the whole face, or from specific regions, and are acquired using image filters applied to the face image [3].

The facial expression recognition is the last stage of this system. According to Liu *et al.* [4], expression recognition systems basically use a three-stage training procedure: feature learning, feature selection and classifier construction, in this order. The feature learning stage is responsible for the extraction of all features related to the facial expression variation. The feature selection chooses the best features to represent the facial expression. They should minimize the intra-class variation of expressions while maximizing the inter-class variation [5]. Minimizing the intra-class variation of expressions is a problem because images of different individuals with the same expression are far from each other in the pixel's space. Maximizing the inter-class variation is also difficult because images of the same person in different expressions may be very close to each other in the pixel's space [20]. At the end of the process, one classifier (or more classifiers with one for each expression) are used to infer the facial expression, given the selected features.

Recently, a lot of work has been employed in the facial expression recognition research field [4], [5], [12]. Methods using convolutional neural networks (CNN) for face recognition [21] can also be found in the literature. CNN's have a high computation cost in terms of memory and speed, but can achieve some degree of shift and deformation invariance and are also highly parallelizable. This network type has demonstrated being able to achieve high recognition rates in various image recognition tasks like character recognition [22], handwritten digit recognition [23], object recognition [24], and facial expression recognition [25], [26], [27], [7].

Although there are many methods proposed in the literature, some points still deserve attention, for example, accuracy rate could be higher in [28], [1], validation methods could be improved in [13], [28], [25] and others limitations in general. Trying to cope with some of these limitations while keeping a simple solution, we present an approach combining standard methods, like image normalizations, synthetic training samples (i.e. real images with artificial rotations) generation and Convolutional Network, into a simple solution that is able to achieve a very high accuracy rate (97.81%) as can be seen in Fig. 1. Our training and experiments were carried out using the Extensive Cohn-Kanade (CK+) database of static images [29] that is largely used in the literature. In addition, we present a complete validation method and a reduced training time compared with some of the state-of-the-art methods.

The remainder of this paper is organized as follows: the next section presents the most recent related works that are followed by a description of the proposed approach. In section IV, the experiments are explained, the results are presented and compared with the state-of-the-art. Finally, a conclusion is presented.

## II. RELATED WORK

There have been several expression recognition approaches developed in the last decade and a lot of progress has been made in this research area recently. A full survey can be found in [3], [30]. This section focuses on methods closely related to the approach proposed in this work.

In [4], the authors propose a novel approach called Boosted Deep Belief Network (BDBN). Their approach performs the three learning stages (feature learning, feature selection and classifier construction) iteratively in a unique framework. The BDBN focus on the six basic expressions. Their experiments were conducted using two public databases of static images, Cohn-Kanade [29] and JAFFE [31], and achieved an accuracy of 96.7% and 68.0%, respectively. The training and the testing adopted a one-versus-all classification strategy, creating a binary classifier for each expression. The time required to train the network was about 8 days. The online recognition is calculated in function of the weak classifiers. In their method they use seven classifiers, one for each expression. Each classifier took 30 ms to recognize each expression, with a total online recognition time of about 0.21 s.

In [5], the authors perform a deep study using Local Binary Patterns (LBP) as feature extractor. They compare and combine different machine learning techniques like template matching, Support Vector Machine, Linear Discriminant Analysis and linear programming to recognize facial expressions. The authors also conduct a study to analyze the impact of image resolution in the accuracy result and conclude that methods based on geometric features do not handle low resolution images very well, while those based on appearance, like Gabor Wavelets and LBP, are not so sensible to the image resolution. The best result achieved in their work was an accuracy rate of 95.1% using SVM and LBP in the Cohn-Kanade [29] database. The testing setting used was a 10-fold cross validation scheme. The training time and the online recognition time was not mentioned by the authors.

An approach using Histogram of oriented Gradients (HoG) features is presented in [32]. The authors used a set of feature extractors like LBP, PCA and HoG with a SVM to classify static face images as one of the six expressions. The proposed method achieves an accuracy rate of 70% when classifying only five expressions (anger, fear, joy, relief and sadness). The training time and the online recognition time was not mentioned by the authors.

Convolutional neural networks (CNN) were firstly used by Lecun *at al.* [33], which was inspired by the early work of Hubel and Wiesel [34]. One of the main advantages of CNN is that the models' input is a raw image rather than hand-coded features. CNNs are able to learn the set of features that best model the desired classification. In general, this type of hierarchical network has alternating types of layers, convolutional layers and sub-sampling layers. CNNs architectures vary in how convolutional and sub-sampling layers are applied and how the networks are trained. Convolutional layers are mainly parametrized by the number of generated maps and the kernels

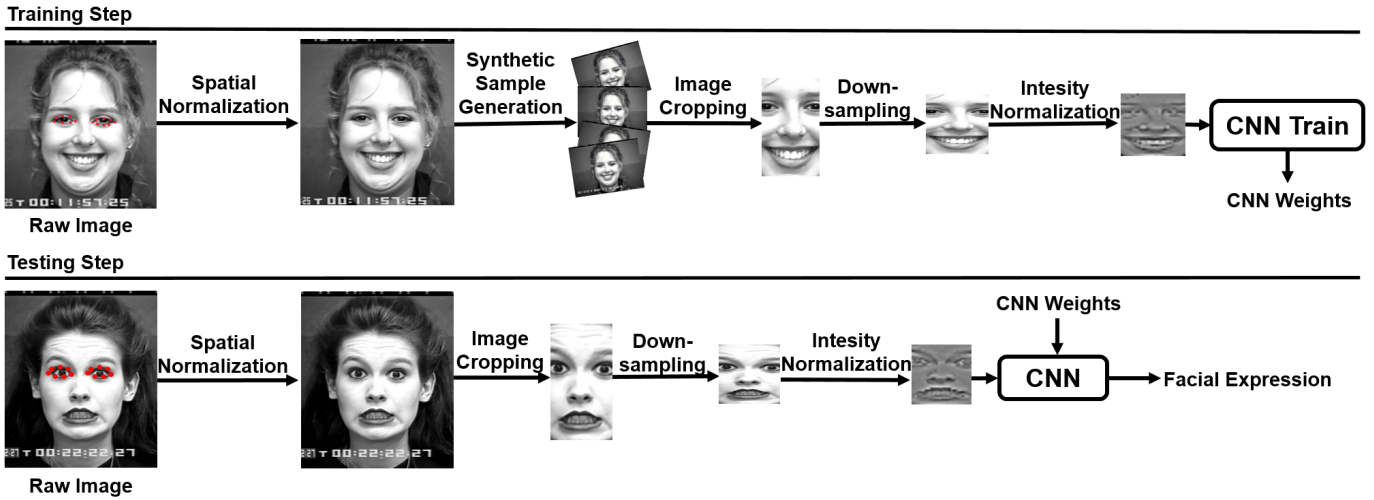


Fig. 2. Overview of the proposed facial expression recognition system. The system is divided in two main steps: training and testing. The training step takes as input an image with a face and its eyes location. Then, a spatial normalization is carried out to align the eyes with the horizontal axis. After that, new images are synthetically generated with rotation angles based on a Gaussian distribution to increase the database size. Then, a cropping is done using the inter-eyes distance to remove background information keeping only expression specific features. A downsampling procedure is carried out to get the features in different images in the same location. Thereafter, an intensity normalization is applied to the image. The normalized images are used to train the Convolutional Network. The output of the training step is a set of weights that achieve the best result with the training data. The testing step use the same methodology as the training step: spatial normalization, cropping, downsampling and intensity normalization. Its output is a single number that represents one of the six basic expressions.

size. The kernel is shifted over the valid region of the input image generating one map. The learning procedure of CNNs consists of getting the best weights associated to the kernel. The learning can use a descendant gradient method, like the one proposed in [33]. Sub-sampling layers are used to increase the position invariance of the kernels [35] by reducing the map size. The new reduced map is generated considering a function of each pixel neighborhood in the higher resolution version. The main types of sub-sampling layers are maximum-pooling and average pooling [35]. In the maximum-pooling, the new map will keep only the maximum pixel value of the respective neighborhood region, whereas in the average pooling the new pixel value will be an average of the neighbors.

In [27], a method that uses a combination of two methods, convolutional networks to detect faces and a rule based algorithm to recognize the expression, is presented. The rule based algorithm is proposed to enhance the subject independent facial expression recognition. This procedure uses rules like distance between eyes and mouth, length of horizontal line segment in mouth and length of horizontal line segment in eyebrows. The experiments were carried out with 10 persons only and the authors report the results of detecting smiling faces only (i.e. happiness), which was 97.6%. The training time and the online recognition time was not mentioned by the authors.

A video-based facial expression recognition system is proposed in [7]. They developed a 3D-CNN having an image sequence (from the neutral to the final expression) using 5 successive frames as 3D inputs. Therefore, the CNN input will be  $H * W * 5$  - where  $H$  and  $W$  are the image height and width respectively, and 5 is the number of frames. The authors claim that the 3-D CNN method can handle some degrees

of shift and deformation invariance. With this approach they achieved an accuracy of 95%, but the method relies on a sequence containing the full movement from the neutral to the expression. The experiments were carried out with 10 persons only. The training time and the online recognition time was not mentioned by the authors.

### III. FACIAL EXPRESSION RECOGNITION SYSTEM

In this section, an efficient method that performs the three learning stages in just one classifier (CNN) is presented. The only additional step required is a pre-process to normalize the images. The proposed method comprises two main phases: training and testing. During training, the system receives a database of grayscale image of faces with their respective expression and eye center locations and learns a set of weights that better separates the facial expressions for classification. During test, the classifier receives a grayscale image of a face along with the respective eye center locations, and outputs the predicted expression by using the weights learned during training.

An overview of the method is illustrated in Fig. 2. The training and the testing have slightly different workflows. For training, the system applies a sequence of: spatial normalization, synthetic samples generation, image cropping, downsampling and intensity normalization. For recognizing an unknown image (i.e. testing phase), the system applies a sequence of: spatial normalization, image cropping, downsampling and intensity normalization. The only difference between the image pre-processing steps of training and testing is the synthetic samples generation that is used in training only. The output of the trained CNN is a single label number that express one of the six basic expressions.

### A. Spatial Normalization

The images in the database vary in rotation, brightness and size even for images of the same person. These variations are not related to the face expression and can affect the accuracy rate of the system. To address this problem, a spatial normalization of the face region is performed to correct possible geometric issues like rotations. Basically, a rotation is applied to align the eyes with the horizontal axis of the image. This rotation makes the angle formed by the line segment going from one eye center to the other, and the horizontal axis to be zero. Rotations in the images are not related to the facial expression and therefore should be removed to avoid negatively affecting the accuracy rate of the system. The spatial normalization procedure is shown in Fig. 3.

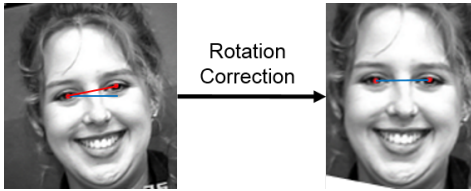


Fig. 3. Spatial Normalization example. The non-corrected input image (left) is rotated (right) using the line segment going from one eye center to the other (red line) and the horizontal axis (blue line).

### B. Synthetic Sample Generation

One of the main problems of CNN methods is that they usually need of a lot of data in the training phase to achieve a good accuracy rate. To address this problem Simard *et. al.* in [36] propose the generation of synthetic images (i.e. real images with artificial rotations) to increase the database. The authors show the benefits of applying combinations of translations, rotations and skewing for increasing the database. Following this idea, in this paper, we use a 2D Gaussian distribution ( $\sigma = 3^\circ$ ) to introduce random noise in the locations of the center of the eyes. Synthetic images are generated by considering the normalized versions of noisy eyes locations. For each image, 30 synthetic images are generated.

As it can be seen in Fig. 4, a Gaussian distribution of points are generated for both eyes. Using this distribution, the new locations are concentrated in regions near the original points and just few locations are far away the original points. Based on the angle generated by pairs of synthetic points (one for the left and one for the right), the image is rotated around the middle point between the synthetic eye centers. Using a Gaussian distribution with a small standard deviation ( $\sigma = 3$ ), the majority of the synthetic samples are generated very close to the original one resulting mostly small rotation angles. Otherwise, it could affect the learning method and decrease the accuracy of the CNN if too many images with high rotation angles (e.g.  $90^\circ$ ,  $180^\circ$ , etc) were used.

### C. Image Cropping

As shown in Fig. 2, the original image has a lot of background information that is not important to the expression

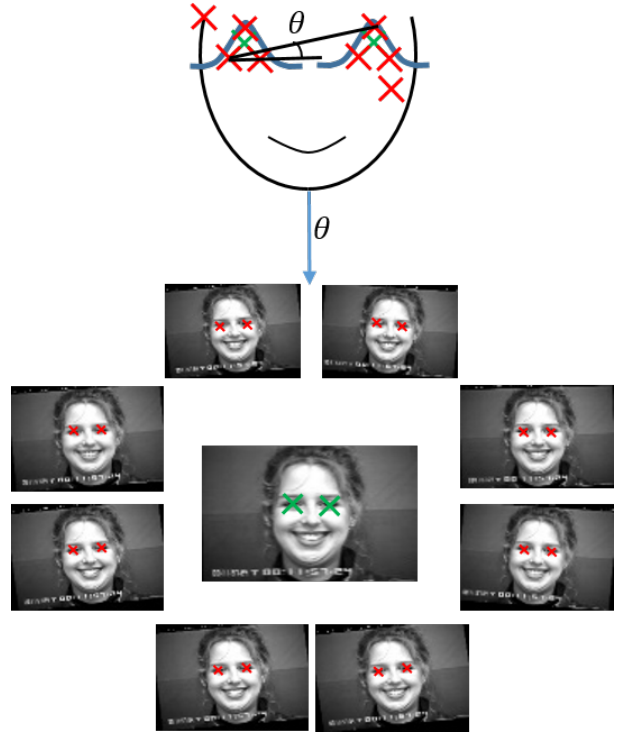


Fig. 4. Illustration of the synthetic sample generation. The Gaussian synthetic sample generation procedure increases the database size and variation, adding rotation noise  $\theta$  in the images considering a controlled environment. The angle of rotations are small and are generated by a pair of points coming from two different Gaussian distributions with  $\sigma = 3^\circ$ , one for each eye. The green crosses are the original eyes points, while the red ones are the synthetic points.

classification procedure. This information could decrease the accuracy of the classification because the classifier has one problem more to solve, discriminating between background and foreground. After the cropping, all image parts that do not have expression specific information are removed. The cropping region also tries to remove facial parts that do not contribute for the expression (e.g. ears, part of the forehead, etc.). Therefore, the region of interest is defined based on a ratio of the inter-eyes distance. Consequently, our method is able to handle different persons and image sizes without human intervention. To get the region shown in Fig. 5, a 4.5 multiplying factor based on the inter-eyes distance was used in the vertical axis and a 2.4 factor was used in the horizontal axis. These values were empirically chosen.

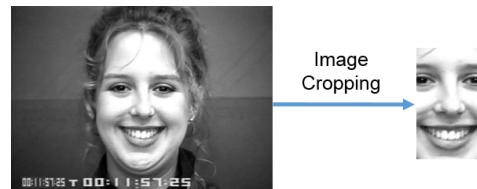


Fig. 5. Image cropping example. The spatial normalized input image (left) is cropped (right) to remove all non-expression features, such as background and hair.

#### D. Downsampling

The downsampling operation is performed to ensure the same location for the face components (eyes, mouth, eyebrow, etc) of every image. This procedure helps the CNN to learn which regions are related to each specific expression. The downsampling also enables the convolutions to be performed in the GPU since most of the graphics card nowadays have limited memory. The final image is 32x32 pixels.

#### E. Intensity Normalization

The image brightness and contrast can vary even in images of the same person in the same expression increasing, therefore, the variation in the feature vector. Such variations increase the complexity of the problem that the classifier has to solve for each expression. In order to reduce these issues an intensity normalization was applied. A method adapted from a bio-inspired technique described in [37] was used. Basically, the normalization is a two step procedure: firstly a subtractive local contrast normalization is performed; and secondly, a divisive local contrast normalization is applied. In the first step, the value of every pixel is subtracted from a Gaussian-weighted average of its neighbors. In the second step, every pixel is divided by the standard deviation of its neighborhood. The neighborhood for both procedures uses a kernel of 7x7 pixels (empirically chosen). An example of this procedure is illustrated in Fig. 6.

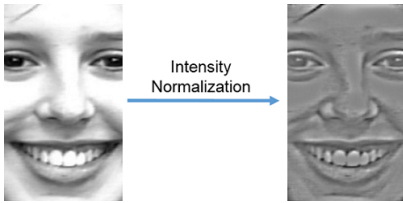


Fig. 6. Illustration of the intensity normalization. The figure shows the image with the original intensity (left) and its intensity normalized version (right).

Equation 1 shows how each new pixel value is calculated in the intensity normalization procedure:

$$x' = \frac{x - \mu_{nhg x}}{\sigma_{nhg x}} \quad (1)$$

where,  $x'$  is the new pixel value,  $x$  is the original pixel value,  $\mu_{nhg x}$  is the Gaussian-weighted average of the neighbors of  $x$ , and  $\sigma_{nhg x}$  is the standard deviation of the neighbors of  $x$ .

#### F. Convolutional Network

The architecture of our Convolutional Network is represented in the Fig. 7. The network receives as input a 32x32 grayscale image and outputs the confidence of each expression. The class with the maximum value is used as the expression in the image. Our CNN architecture comprises 2 convolutional layers and 2 subsampling layers. The first layer of the CNN is a convolution layer, that applies a convolution kernel of 7x7 and outputs an image of 28x28 pixels. This layer is followed

by a subsampling layer that uses max-pooling (with kernel size 2x2) to reduce the image to half of its size. Subsequently, a new convolution is applied to the feature vector and is followed by another subsampling. The output is given to a fully connected layer that has 256 neurons. The network has six output nodes (one for each expression that outputs their confidence level) that are fully connected to the previous layer.

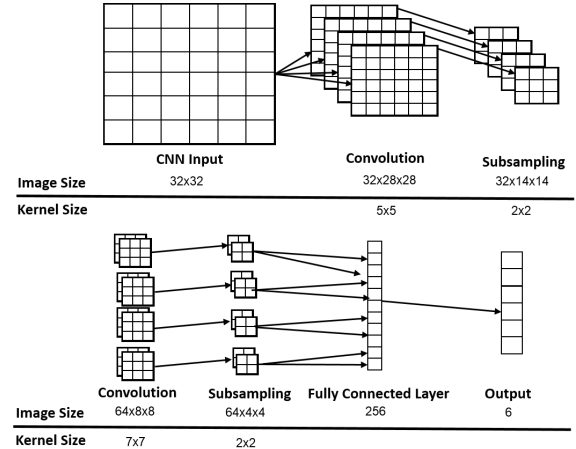


Fig. 7. Architecture of the proposed Convolutional Neural Network. It comprises five layers: the first layer (convolution type) outputs 32 maps; the second layer (subsampling type) reduces the map size by half; the third layer (convolution type) outputs 64 maps for each input; the fourth layer (subsampling type) reduces the map once more by half; the fifth layer (fully connected type) and the final output with 6 nodes representing each one of the expression are responsible for classifying the facial image.

## IV. RESULTS AND DISCUSSION

The experiments were performed using the Extended Cohn-Kanade (CK+) database [29], and using the training and testing methodology described in [4]. Accuracy is computed considering one classifier to classify all learned expressions. In addition, in order to perform a fair comparison with the method proposed in [4], accuracy is also computed considering one binary classifier for each expression.

The implementation of the normalization steps was done in-house using C++ and OpenCV, and we used a GPU based CNN framework developed in C++ together with a Matlab wrapper as classifier (developed by Demyanov *et. al.* [38]). All the experiments were carried out using an Intel Core i7 3.4 GHz with a NVIDIA GeForce GTX 660 CUDA Capable that has 1.5Gb of memory in the GPU. The normalization and classification steps took in average 0.11 and 0.23 second respectively. The convolutional network parameters are presented in Table

In this Section, a study is carried out showing the impact of every normalization step in the accuracy of the method. Firstly, we describe the database used for the experiments. Secondly, each experiment is presented and discussed in details. Thirdly, a comparison with the state-of-the-art methods is presented. Finally, the limitations are discussed.

TABLE I  
CNN PARAMETERS

Parameter	Value
Epochs	300
Loss Function	Logistic Regression
Momentum	0.95
Learning Rate	0.05

### A. Test Database

The presented system was trained and tested using the Extended Cohn-Kanade dataset (CK+) [29]. This dataset comprises 100 university students with age between 18 and 30 years old. The subjects in the dataset are 65% female, 15% are African-American and 3% are Asian or south American. The images were captured from a camera located directly in front of the subject. The students were instructed to perform a series of expressions. Each sequence begins and ends with the neutral expression. All images in the dataset are 640 by 480 pixel arrays with 8-bit precision for grayscale values.

To do a fair comparison with the state-of-the-art methods [4], [5], in our experiments the neutral and the contempt expression images were not used. In addition, only the last 3 frames of each expression sequence were selected to compose the training/testing dataset. The dataset was divided in 8 groups without subject overlap between groups. This methodology was used in [4] to ensure that the testing groups do not have subjects from the training group.

### B. Experiments

*a) No Preprocessing:* This first experiment was carried out using the original database, without any intervention or image pre-processing. The training and the testing for this, and all subsequent experiments, use a 8-Fold procedure. The training was performed 8 times, each time leaving one group out for tests. The training of the proposed CNN uses a gradient descendant method. Gradient descendant methods rely on the order of the given samples to search for the local minimum. To avoid this variation, each experiment configuration is run 10 times (training and testing) with different image presentation order. With this change, the descending gradient method can take additional paths and increase (or decrease) the recognition rate. We present the accuracy as an average and as the best value for all 10 runs. Using the accuracy as an average, we show that if even the presentation order is not the best, the method still achieves high accuracy rates. In this experiment, the best accuracy of all 10 runs was 61.70%, and the average was 56.93%. The average accuracy per expression is shown in Table II. This average is calculated using the amount of hits in all runs divided by the amount of all images in all runs.

As it can be seen in Table II, using only the CNN without any image pre-processing, the recognition rate is very low compared to the state-of-the-art methods.

*b) Spatial Normalization:* As explained in Section III, a spatial normalization is applied to get the features in the same pixel space, in both training and testing steps. This normalization took only 0.09 seconds to be applied to an image. This

experiment is done using the same methodology described before. The best accuracy of all 10 runs was 90.00%, and the average was 86.44%. The average accuracy per expression is shown in Table II. Compared with the result shown before, we can note a significantly increase of the recognition rate by adding the spatial normalization and cropping processes.

*c) Intensity Normalization:* The intensity normalization is used to remove brightness variation in the images in both steps, training and testing. This experiment was performed using just the intensity normalization and cropping. It uses the same methodology described before. This normalization took only 0.02 second to be applied to an image. The best accuracy of all 10 runs was 86.10%, and the average was 82.39%. The average accuracy per expression is shown in Table II.

*d) Spatial Normalization and Intensity Normalization:* Putting both normalization steps together, spatial and intensity, we remove a big part of the variations unrelated to the facial expression leaving just the expression specific variation that is not related to the person. This experiment is done using the same methodology described before. The normalizations together take 0.11 seconds to be performed. The best accuracy of all 10 runs was 87.81%, and the average was 84.68%. The average accuracy per expression is shown in Table II.

As it can be seen, the accuracy of applying both normalization procedures is lower than the one that uses only the spatial normalization. The result of the disgust and happy expressions have a very low accuracy, which reduces the overall recognition average. To verify the need for the intensity normalization, a new experiment using only the spatial normalization procedure and the synthetic sample generation was performed and is presented below.

*e) Spatial Normalization and Synthetic Samples:* The result of the spatial and intensity normalization and only the spatial normalization gives a false impression that the intensity normalization might decrease the accuracy of the method - since the result of applying only the spatial normalization is better than the result with both normalizations. To verify this suspicion, a new experiment was conducted, using only the spatial normalization procedure and the synthetic samples. This experiment is done using the same methodology described before. The best accuracy of all 10 runs was 90.83%, and the average was 87.54%. The average accuracy per expression is shown in Table II.

*f) Spatial Normalization, Intensity Normalization and Synthetic Samples:* The best result achieved in our method applies the three image pre-processing steps: spatial normalization, intensity normalization and synthetic samples generation. For the synthetic sample generation, thirty more samples were generated for each image. This experiment is done using the same methodology described before. The total training time for this experiment was four and a half days. The best accuracy of all 10 runs was 93.74%, and the average was 91.46%. The average accuracy per expression is shown in Table II. The accuracy of this experiment shows that joining the three techniques (spatial normalization, intensity normalization and synthetic samples) is better than using only the spatial nor-

malization and the synthetic samples presented previously.

Table II shows the mean accuracy for each expression using all the preprocessing steps already discussed. In *a*) no preprocessing is used, in *b*) just the spatial normalization is employed, in *c*) just the intensity normalization is used, in *d*) both normalizations (spatial and intensity) are used, in *e*) the spatial normalization using the synthetic samples are used and in *f*) both normalizations and the synthetic samples are used.

TABLE II  
PREPROCESSING STEPS ACCURACY DETAILS

	Angry	Disgust	Fear	Happy	Sad	Surprise
a)	31.40%	63.67%	11.60%	68.30%	24.64%	81.08%
b)	79.70%	87.40%	68.00%	94.00%	64.52%	96.06%
c)	70.14%	90.50%	43.00%	94.78%	53.21%	94.65%
d)	90.96%	44.40%	95.16%	57.38%	94.73%	84.18%
e)	80.22%	88.98%	71.86%	95.74%	69.04%	94.61%
f)	82.22%	96.55%	74.93%	98.06%	76.57%	97.38%

### C. Comparisons

Table III summarizes all results presented in this section, showing the evolution of the proposed method by changing the image pre-processing steps.

TABLE III  
PREPROCESSING COMPARISON

Preprocessing	Average	Best
None	56.93±5%	61.70%
Spatial Normalization	86.44±3%	90.00%
Intensity Normalization	82.39±1%	86.10%
Both Normalizations	84.68±2%	87.81%
Spatial Normalization and Synthetic Samples	87.54±1%	90.83%
<b>Both Normalizations and Synthetic Samples</b>	<b>91.46±1%</b>	<b>93.74%</b>

As it can be seen in Table III, the best performance was achieved using both, normalization procedures and the synthetic samples. Using the training weights obtained with the best accuracy, the confusion matrix shown in Table IV was built. The recognition of the disgust, happy and surprise expressions achieves an accuracy rate higher than 97%. While the angry and fear expression was about 85%. The sad expression achieves the smallest recognition rate, with only 79.76%. The fear expression was confused in the majority of the time with the sad expression. This shows that the features of these two expression are not well separated in the pixel space, i.e. they are very similar to each other in some cases.

TABLE IV  
CONFUSION MATRIX USING BOTH NORMALIZATIONS AND SYNTETIC SAMPLES

	Angry	Disgust	Fear	Happy	Sad	Surprise
Angry	<b>85.19%</b>	0.56%	0%	0.48%	3.57%	0%
Disgust	3.70%	<b>97.74%</b>	0%	0%	0%	0%
Fear	5.19%	0%	<b>85.33%</b>	0%	9.25%	0%
Happy	0%	0%	5.33%	<b>98.55%</b>	0%	1.20%
Sad	3.70%	0%	4.00%	0.97%	<b>79.76%</b>	0%
Surprise	2.22%	1.69%	5.33%	0%	7.14%	<b>98.80%</b>

As discussed before, we adapted our method to perform binary classifications for each expression, i.e. do a one-versus-all classification as in [4]. In this method, all images are

presented to six binary classifiers, each one responsible to recognize the presence, or not, of one specific expression. For example, if we present an image of a subject smiling, just the classifier of the happy expression should answers "yes", and all other classifiers should answer "no". Using this methodology, our accuracy was 97.81%. The comparison of this accuracy with other methods is shown in Table V.

TABLE V  
PERFORMANCE COMPARISON

Method	Performance
CSPL [12]	89.90%
AdaGabor [39]	93.30%
3D-CNN [7]	95.00%
LBPSVM [5]	95.10%
BDBN [4]	96.70%
<b>Our Method</b>	<b>97.81%</b>

### D. Limitations

As discussed before, the presented method needs the locations of each eye for the image pre-processing steps. The eye detection can be easily included to the system adopting the method shown in [40]. In addition, as shown in Table II, the accuracy of some expressions, like fear and sad, was less than 80%, while the accuracy of the whole method was about 93%. This suggests that the variation between these classes are not enough to separate them. One approach to address this problem is to create a specialized classifier for those expressions, to be used as a second classifier.

### V. CONCLUSION

In this paper, we propose a facial expression recognition system that uses a combination of standard method, like Convolutional Network and specific image pre-processing steps. Experiments showed that the combination of the normalization procedures improves significantly the method's accuracy. As shown in the results, in comparison with the state-of-the art methods that use the same facial expression database, our method achieves a better accuracy, and presents a simpler solution. In addition, it takes less time to train. As future work, we want to test this approach in others databases, and perform a cross database validation.

### ACKNOWLEDGMENT

We would like to thank Universidade Federal do Espírito Santos - UFES (project SIEEPEF, 5911/2015), Fundação de Amparo Pesquisa do Espírito Santo - FAPES (grants 65883632/14, 53631242/11, and 60902841/13) and Coordenação de Aperfeiçoamento de Pessoal de Nível Superior - CAPES (grant 11012/13-7, and scholarship).

### REFERENCES

- [1] Y. Wu, H. Liu, and H. Zha, "Modeling facial expression space for recognition," in *2005 IEEE/RSJ International Conference on Intelligent Robots and Systems, 2005. (IROS 2005)*, 2005, pp. 1968–1973.
- [2] C. Darwin, *The Expression of the Emotions in Man and Animals*. CreateSpace Independent Publishing Platform, 2012.

- [3] S. Z. Li and A. K. Jain, *Handbook of Face Recognition*. Springer Science & Business Media, 2011.
- [4] P. Liu, S. Han, Z. Meng, and Y. Tong, "Facial expression recognition via a boosted deep belief network," in *2014 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2014, pp. 1805–1812.
- [5] C. Shan, S. Gong, and P. W. McOwan, "Facial expression recognition based on local binary patterns: A comprehensive study," *Image and Vision Computing*, vol. 27, no. 6, pp. 803–816, 2009.
- [6] W. Liu, C. Song, and Y. Wang, "Facial expression recognition based on discriminative dictionary learning," in *2012 21st International Conference on Pattern Recognition (ICPR)*, 2012, pp. 1839–1842.
- [7] Y.-H. Byeon and K.-C. Kwak\*, "Facial expression recognition using 3d convolutional neural network," in *International Journal of Advanced Computer Science and Applications*, vol. 5, no. 12, 2014.
- [8] J.-J. J. Lien, T. Kanade, J. Cohn, and C. Li, "Detection, tracking, and classification of action units in facial expression," *Journal of Robotics and Autonomous Systems*, 1999.
- [9] C.-R. Chen, W.-S. Wong, and C.-T. Chiu, "A 0.64 mm real-time cascade face detection design based on reduced two-field extraction," *IEEE Transactions on Very Large Scale Integration (VLSI) Systems*, vol. 19, no. 11, pp. 1937–1948, 2011.
- [10] C. Garcia and M. Delakis, "Convolutional face finder: a neural architecture for fast and robust face detection," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 26, no. 11, pp. 1408–1423, 2004.
- [11] Z. Zhang, D. Yi, Z. Lei, and S. Li, "Regularized transfer boosting for face detection across spectrum," *IEEE Signal Processing Letters*, vol. 19, no. 3, pp. 131–134, 2012.
- [12] M. Bartlett, G. Littlewort, M. Frank, C. Lainscsek, I. Fasel, and J. Movellan, "Recognizing facial expression: machine learning and application to spontaneous behavior," in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2005. CVPR 2005*, vol. 2, 2005, pp. 568–573 vol. 2.
- [13] P. Liu, M. Reale, and L. Yin, "3d head pose estimation based on scene flow and generic head model," in *2012 IEEE International Conference on Multimedia and Expo (ICME)*, 2012, pp. 794–799.
- [14] W. W. Kim, S. Park, J. Hwang, and S. Lee, "Automatic head pose estimation from a single camera using projective geometry," in *Communications and Signal Processing (ICICS) 2011 8th International Conference on Information*, 2011, pp. 1–5.
- [15] M. Demirkus, D. Precup, J. Clark, and T. Arbel, "Multi-layer temporal graphical model for head pose estimation in real-world videos," in *2014 IEEE International Conference on Image Processing (ICIP)*, 2014, pp. 3392–3396.
- [16] Z. Zhang, M. Lyons, M. Schuster, and S. Akamatsu, "Comparison between geometry-based and gabor-wavelets-based facial expression recognition using multi-layer perceptron," in *Third IEEE International Conference on Automatic Face and Gesture Recognition, 1998. Proceedings*, 1998, pp. 454–459.
- [17] P. Yang, Q. Liu, and D. Metaxas, "Boosting coded dynamic features for facial action units and facial expression recognition," in *IEEE Conference on Computer Vision and Pattern Recognition, 2007. CVPR '07*, 2007, pp. 1–6.
- [18] S. Jain, C. Hu, and J. Aggarwal, "Facial expression recognition with temporal modeling of shapes," in *2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops)*, pp. 1642–1649.
- [19] Y. Lin, M. Song, D. T. P. Quynh, Y. He, and C. Chen, "Sparse coding for flexible, robust 3d facial-expression synthesis," *IEEE Computer Graphics and Applications*, vol. 32, no. 2, pp. 76–88, 2012.
- [20] S. Rifai, Y. Bengio, A. Courville, P. Vincent, and M. Mirza, "Disentangling factors of variation for facial expression recognition," in *Computer Vision ECCV 2012*, ser. Lecture Notes in Computer Science, A. Fitzgibbon, S. Lazebnik, P. Perona, Y. Sato, and C. Schmid, Eds. Springer Berlin Heidelberg, 2012, no. 7577, pp. 808–822.
- [21] S. Lawrence, C. Giles, A. C. Tsoi, and A. Back, "Face recognition: a convolutional neural-network approach," *IEEE Transactions on Neural Networks*, vol. 8, no. 1, pp. 98–113, 1997.
- [22] G. Lv, "Recognition of multi-fontstyle characters based on convolutional neural network," in *2011 Fourth International Symposium on Computational Intelligence and Design (ISCID)*, vol. 2, 2011, pp. 223–225.
- [23] X.-X. Niu and C. Y. Suen, "A novel hybrid CNN SVM classifier for recognizing handwritten digits," *Pattern Recognition*, vol. 45, no. 4, pp. 1318–1325, 2012.
- [24] Y. LeCun, F. J. Huang, and L. Bottou, "Learning methods for generic object recognition with invariance to pose and lighting," in *Proceedings of the 2004 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004. CVPR 2004*, vol. 2, 2004, pp. II–97–104 Vol.2.
- [25] B. Fasel, "Robust face analysis using convolutional neural networks," in *16th International Conference on Pattern Recognition, 2002. Proceedings*, vol. 2, 2002, pp. 40–43 vol.2.
- [26] F. Beat, "Head-pose invariant facial expression recognition using convolutional neural networks," in *Fourth IEEE International Conference on Multimodal Interfaces, 2002. Proceedings*, 2002, pp. 529–534.
- [27] M. Matsugu, K. Mori, Y. Mitari, and Y. Kaneda, "Subject independent facial expression recognition with robust face detection using a convolutional neural network," *Neural Networks: The Official Journal of the International Neural Network Society*, vol. 16, no. 5, pp. 555–559, 2003.
- [28] P. Zhao-yi, W. Zhi-qiang, and Z. Yu, "Application of mean shift algorithm in real-time facial expression recognition," in *International Symposium on Computer Network and Multimedia Technology, 2009. CNMT 2009*, 2009, pp. 1–4.
- [29] P. Lucey, J. Cohn, T. Kanade, J. Saragih, Z. Ambadar, and I. Matthews, "The extended cohn-kanade dataset (CK+): A complete dataset for action unit and emotion-specified expression," in *2010 IEEE Computer Society Conference on Computer Vision and Pattern Recognition Workshops (CVPRW)*, 2010, pp. 94–101.
- [30] C.-D. Caeleu, "Face expression recognition: A brief overview of the last decade," in *2013 IEEE 8th International Symposium on Applied Computational Intelligence and Informatics (SACI)*, 2013, pp. 157–161.
- [31] M. Lyons, S. Akamatsu, M. Kamachi, and J. Gyoba, "Coding facial expressions with gabor wavelets," in *Third IEEE International Conference on Automatic Face and Gesture Recognition, 1998. Proceedings*, 1998, pp. 200–205.
- [32] M. Dahmane and J. Meunier, "Emotion recognition using dynamic grid-based HoG features," in *2011 IEEE International Conference on Automatic Face Gesture Recognition and Workshops (FG 2011)*, 2011, pp. 884–888.
- [33] Y. Lecun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," vol. 86, no. 11, pp. 2278–2324, 1998.
- [34] D. H. Hubel and T. N. Wiesel, "Receptive fields and functional architecture of monkey striate cortex," *The Journal of Physiology*, vol. 195, no. 1, pp. 215–243, 1968.
- [35] D. C. Cirean, U. Meier, J. Masci, L. M. Gambardella, and J. Schmidhuber, "Flexible, high performance convolutional neural networks for image classification," in *Proceedings of the Twenty-Second International Joint Conference on Artificial Intelligence - Volume Volume Two*, ser. IJCAI'11. AAAI Press, 2011, pp. 1237–1242.
- [36] P. Simard, D. Steinkraus, and J. C. Platt, "Best practices for convolutional neural networks applied to visual document analysis," in *Seventh International Conference on Document Analysis and Recognition, 2003. Proceedings*, 2003, pp. 958–963.
- [37] B. A. Wandell, *Foundations of Vision*, 1st ed. Sunderland, Mass: Sinauer Associates Inc, May 1995.
- [38] S. Demyanov, J. Bailey, R. Kotagiri, and C. Leckie, "Invariant back-propagation: how to train a transformation-invariant neural network," *arXiv:1502.04434 [cs, stat]*, 2015.
- [39] L. Zhong, Q. Liu, P. Yang, B. Liu, J. Huang, and D. Metaxas, "Learning active facial patches for expression analysis," in *2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2012, pp. 2562–2569.
- [40] J. M. Saragih, S. Lucey, and J. F. Cohn, "Deformable model fitting by regularized landmark mean-shift," *International Journal of Computer Vision*, vol. 91, no. 2, pp. 200–215, 2010.