FERAtt: Facial Expression Recognition with Attention Net

Pedro D. Marrero Fernandez[1], Fidel A. Guerrero Peña[1,2], Tsang Ing Ren[1], Alexandre Cunha[2]
[1]Centro de Informática (CIn), Universidade Federal de Pernambuco (UFPE), Brazil
[2]Center for Advanced Methods in Biological Image Analysis (CAMBIA)
California Institute of Technology, USA
[1]{pdmf, tir}@cin.ufpe.br, [2]{fagp, cunha}@caltech.edu

Abstract

We present a new end-to-end network architecture for facial expression recognition with an attention model. It focuses attention in the human face and uses a Gaussian space representation for expression recognition. We devise this architecture based on two fundamental complementary components: (1) facial image correction and attention and (2) facial expression representation and classification. The first component uses an encoder-decoder style network and a convolutional feature extractor that are pixel-wise multiplied to obtain a feature attention map. The second component is responsible for obtaining an embedded representation and classification of the facial expression. We propose a loss function that creates a Gaussian structure on the representation space. To demonstrate the proposed method, we create two larger and more comprehensive synthetic datasets using the traditional BU3DFE and CK+ facial datasets. We compared results with the PreActResNet18 baseline. Our experiments on these datasets have shown the superiority of our approach in recognizing facial expressions.

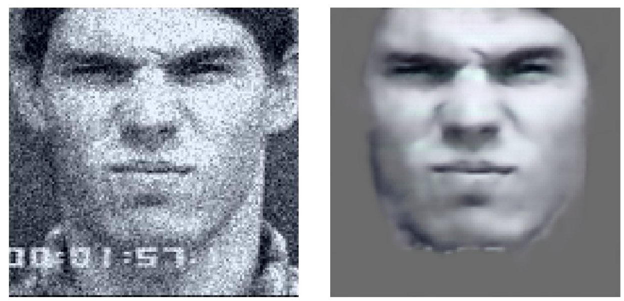


Figure 1: Example of attention in an image. Facial expression is recognized on the front face which is separated from the less prominent components of the image by our approach. The goal is to jointly train for attention and classification where a probability map of the faces are created and their expressions learned by a dual–branch network. By focusing attention on the face features, we try to eliminate the detrimental influence possibly present on the other elements in the image during the facial expression classification. In this formulation, we explicitly target learning expressions solely on learned faces and not on other irrelevant parts of the image (background).

1. Introduction

Human beings are able to express and recognize emotions as a way to communicate an inner state. Facial expression is the main form to convey this information and its understanding has transformed the treatment of emotions by the scientific community. Traditionally, scientists assumed that people have internal mechanisms comprising a small set of emotional reactions (*e.g.* happiness, anger, sadness, fear, disgust) that are measurable and objective. Understanding these mental states from facial and body cues is a fundamental human trait, and such aptitude is vital in our daily communications and social interactions. In fields such as Human-Computer Interaction (HCI), Neuroscience, and Computer Vision, scientists have conducted extensive research to understand human emotions. Some of these studies aspire to create computers that can understand and respond to human emotions and to our general behavior, potentially leading to seamless beneficial interactions between humans and computers [29, 12]. Our work aims to contribute to this effort, more specifically in the area of Facial Expression Recognition, or FER for short.

Deep Convolutional Neural Networks (CNN) have recently shown excellent performance in a wide variety of image classification tasks [17, 32, 35, 34]. The careful design of local to global feature learning with a convolution, pooling, and layered architecture produces a rich visual representation, making CNN a powerful tool for facial expression recognition [18]. Research challenges such as the Emotion Recognition in the Wild (EmotiW) series[1] and

[1] https://sites.google.com/view/emotiw2018

Kaggle's Facial Expression Recognition Challenge[2] suggest the growing interest of the community in the use of deep learning for the solution of this problem.

Recent developments for the facial expression recognition problem consider processing the entire image regardless of the face crop location within the image [42]. Such developments bring in extraneous artifacts, including noise, which might be harmful for classification as well as incur in unnecessary additional computational cost. This is problematic as the *minutiae* that characterizes facial expressions can be affected by elements such as hair, jewelry, and other environmental objects not defining the actual face and as part of the image background. Some methods use heuristics to decrease the searching size of the facial regions to avoid considering objects beyond the face itself. Such approaches contrast to our understanding of the human visual perception, which quickly parses the field of view, discards irrelevant information, and then focus the main processing on a specific target region of interest – the so called *visual attention* mechanism [14, 39]. Our approach tries to mimic this behavior as it aims to suppress the contribution of surrounding deterrent elements by segmenting the face in the image and thus concentrating recognition solely on facial regions. Figure 1 illustrates how the attention mechanism works in a typical scene.

Attention mechanisms have recently been explored in a wide variety of contexts [38, 15], often providing new capabilities for known neural networks models [7, 8, 4]. While they improve efficiency [26] and performance on state-of-the-art machine learning benchmarks [38], their computational architecture is much simpler than those comprising the mechanisms in the human visual cortex [2]. Attention has also been long studied by neuroscientists [36], who believe it is crucial for visual perception and cognition [1] as it is inherently tied to the architecture of the visual cortex and can affect its information.

Our contributions are summarized as follows: (1) We propose a CNN-based method using attention to jointly solve for representation and classification in FER problems; (2) We introduce a new dual-branch network to extract an attention map which in turn improves the learning of kernels specific to facial expression; (3) A new loss function is formulated for obtaining a facial manifold represented as a Gaussian Mixture Model; and (4) We offer a new synthetic generator to render face expressions which significantly augments training data and consequently improves the overall classification.

2. Related Works

Liu *et al.* [22] introduced a facial expression recognition framework using 3DCNN together with deformable action parts constraints to jointly localize facial action parts and learn part-based representations for expression recognition. Liu *et al.* [21] followed by including the pre-trained Caffe CNN models to extract image-level features.

In 2015, Yu and Zhang [43] achieved state-of-the-art results in the EmotiW challenge using CNNs. They used an ensemble of CNNs each with five convolutional layers and showed that randomly perturbing the input images yielded a 2-3% boost in accuracy. Specifically, they applied transformations to the input images at training time. At testing time, their model generated predictions for multiple perturbations of each test example and voted on the class label to produce a final answer. They used stochastic pooling [6] rather than max pooling due to its good performance on limited training data. Mollahosseini *et al.* [27] have also obtained state of the art results with a network consisting of two convolutional layers, max-pooling, and four inception layers, the latter introduced by GoogLeNet.

Another recent method the De-expression Residue Learning (DeRL) [40], trains a generative model to create a corresponding neutral face image for any input face. Then, another model is trained to learn the deposition (or residue) that remains in the intermediate layers of the generative model for the classification of facial expression.

Zhang *et al.* [46] proposed an end-to-end learning model based on Generative Adversarial Network (GAN). The architecture incorporates a generator, two discriminators, and a classifier. The GAN is used for generating multiples variation of one image, which is used to train a convolutional neural network.

3. Methodology

In this section, we describe our contributions in designing a new network architecture, in the formulation of the loss function used for training, and in the method to generate synthetic data.

3.1. Network architecture

Given a facial expression image I, our objective is to obtain a good representation and classification of I. The proposed model, Facial Expression Recognition with Attention Net (FERAtt), is based on the dual-branch architecture [9, 19, 28, 48] and consists of four major modules: (i) an attention module G_{att} to extract the attention feature map, (ii) a feature extraction module G_{ft} to obtain essential features from the input image I, (iii) a reconstruction module G_{rec} to estimate a proper attention image I_{att}, and (iv) a representation module G_{rep} that is responsible for the representation and classification of the facial expression image. An illustration of the proposed model is shown in Figure 2.

Attention module. We use an encoder-decoder style network, which has been shown to produce good results for many generative [33, 48] and segmentation tasks [31]. In

[2]https://www.kaggle.com/c/challenges-in-representation-learning-facial-expression-recognition-challenge

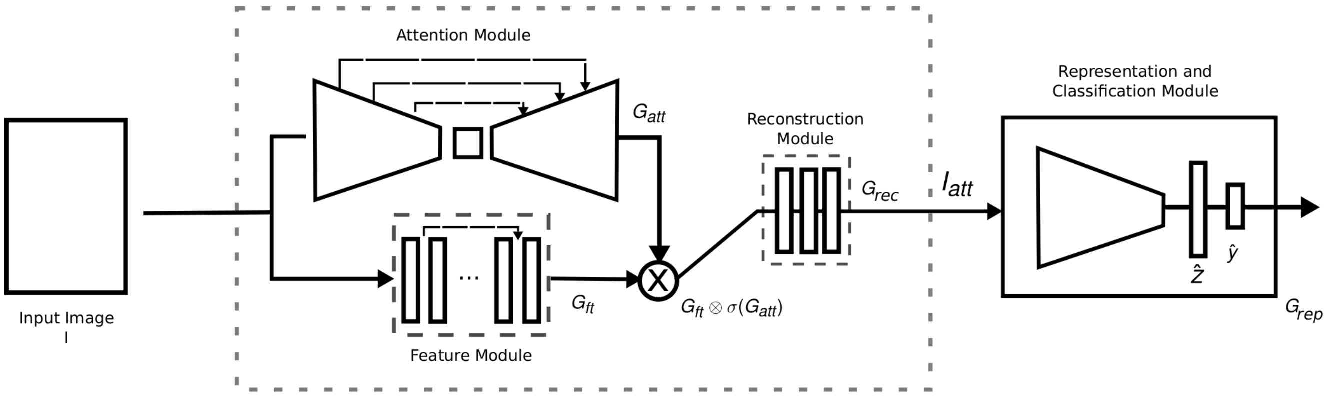


Figure 2: **Architecture of FERAtt**. Our model consists of four major modules: attention module G_{att}, feature extraction module G_{ft}, reconstruction module G_{rec}, and classification and representation module G_{rep}. The features extracted by G_{att}, G_{ft} and G_{rec} are used to create the attention map I_{att} which in turn is fed into G_{rep} to create a representation of the image. Input images I have 128×128 pixels and are reduced to 32×32 by an Averaging Pooling layer on the reconstruction module. Classification is thus done on these smaller but richer representations of the original image.

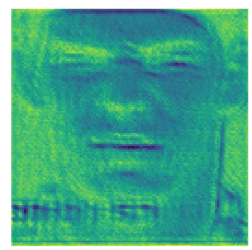
particular, we choose a variation of the fully convolutional model proposed in [31] for semantic segmentation. Also, we applied four layers in the coder with skip connections and dilation of 2x. The decoder layer is initialized with pre-trained ResNet34 [10] layers. This significantly accelerates the convergence. The output features of the decoder are denoted by G_{att}, which is used to determine the attention feature map. This is a probability map that is not the same as a simple segmentation procedure.

Feature extraction module. Four ResBlocks [20] were used to extract high-dimensional features for image attention and to maintain spatial information; no pooling or strided convolutional layers were used. We denote the extracted features as G_{ft} – see Figure 3b.

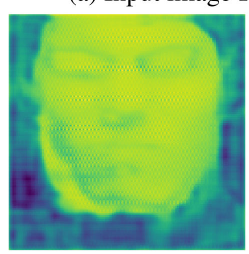
Reconstruction module. The reconstruction layer adjusts the attention map to create an enhanced input to the representation module. This module has two convolutional layers, a Relu layer, and an Average Pooling layer which, by design choice, resizes the input image of 128×128 to 32×32. This reduced size was chosen for the input of the representation and classification module (PreActivation-ResNet [11]), the image size number we borrowed from the literature to facilitate comparisons. We plan to experiment with other sizes in the future. We denote the feature attention map as I_{att} – see Figure 3d.

Representation and classification module. For the representation and classification of facial expressions, we have chosen a Fully Convolutional Network (FCN) of PreActivateResNet [11]. This architecture has shown excellent results when applied on classification tasks. The output of the FCN, vector z, is evaluated in a linear layer to obtain a vector $\hat{z} \in \mathbb{R}^d$ with the desired dimensions. $f_\Theta : \mathbb{R}^D \to \mathbb{R}^d$, the network function, builds a representation for a sample

(a) Input image I



(b) G_{ft}



(c) G_{att}

(d) I_{att}

Figure 3: Generation of attention map I_{att}. A 128×128 noisy input image (a) is processed by the feature extraction G_{ft} and attention G_{att} modules whose results, shown, respectively, in panels (b) and (c), are combined and then fed into the reconstruction module G_{rec}. This in turn produces a clean and focused attention map I_{att}, shown on panel (d), that is classified by the last module G_{rep} of FERAtt. The I_{att} image shown here is before reduction to 32×32 size.

image $x \in \mathbb{R}^D$, (e.g. $D = 128 \times 128$ pixels) in an embedded space of reduced dimension $\mathbb{R}^d$ (we use $d = 64$ in our experiments). Vector $\hat{z}$ is then evaluated in a regression layer to estimate the probability $p(w|\hat{z})$ for each class w_j, $w = [w_1, w_2, \ldots, w_c]$.

3.2. Loss functions

The FERAtt network generates three outputs: a feature attention map $\hat{I}_{att}$, a representation vector $\hat{z}$, and a classification vector $\hat{w}$. In our training data, each image I has an associated binary ground truth mask I_{mask} corresponding to a face in the image and its expression class vector w. We train the network by jointly optimizing the sum of attention, representation, and classification losses:

$$\min_{\Theta} \{\mathcal{L}_{att}(I_{att}, I \otimes I_{mask}) + \mathcal{L}_{rep}(\hat{z}, w) + \mathcal{L}_{cls}(\hat{w}, w)\} \tag{1}$$

where Θ represents the collective parameters that need be optimized. We use the pixel-wise MSE (Mean Square Error) loss function for $\mathcal{L}_{att}$, and for $\mathcal{L}_{cls}$ we use the BCE (Binary Cross Entropy) loss function. We propose a new loss function $\mathcal{L}_{rep}$ for the representation, defined below.

3.3. Structured Gaussian Manifold Loss

Let $S = \{x_i | x_i \in \mathbb{R}^D\}$ be a collection of *i.i.d.* samples x_i we want to classify into c classes, and let w_j represent the j–th class, for $j = 1, \dots, c$. The class function $l(x) = \arg\max p(w|f_\Theta(x))$ returns the class w_j of sample x – maximum *a posteriori* probability estimate – for the neural net function $f_\Theta : \mathbb{R}^D \rightarrow \mathbb{R}^d$ drawn independently according to probability $p(x|w_j)$ for input x. Suppose we separate S in an embedded space such that each set $C_j = \{x | x \in S, l(x) = w_j\}$ contains the samples belonging to class w_j. Our goal is to find a Gaussian representation for each C_j which would allow a clear separation of S in a reduced space, $d \ll D$.

We assume that $p(f_\Theta(x)|w_j)$ has a known parametric form, and it is therefore determined uniquely by the value of a parameter vector θ_j. For example, we might have $p(f_\Theta(x)|w_j) \sim N(\mu_j, \Sigma_j)$, where $\theta_j = (\mu_j, \Sigma_j)$, for $N(.,.)$ the normal distribution with mean μ_j and variance Σ_j. To show the dependence of $p(f_\Theta(x)|w_j)$ on θ_j explicitly, we write $p(f_\Theta(x)|w_j)$ as $p(f_\Theta(x)|w_j, \theta_j)$. Our problem is to use the information provided by the training samples to obtain a good transformation function $f_\Theta(x_j)$ that generates embedded spaces with a known distribution associated with each category. Then the *a posteriori* probability $P(w_j|f_\Theta(x))$ can be computed from $p(f_\Theta(x)|w_j)$ by the Bayes' formula:

$$P(w_j|f_\Theta(x)) = \frac{p(w_j)p(f_\Theta(x)|w_j, \theta_j)}{\sum_i^c p(w_i)p(f_\Theta(x)|w_i, \theta_i)} \tag{2}$$

In this work, we are using the normal density function for $p(x|w_j, \theta_j)$. The objective is to generate embedded subspaces with a defined structure. We use Gaussian structures:

$$p(f_\Theta(x)|w_j, \mu_j, \Sigma_j) = \frac{1}{(2\pi)^{n/2}|\Sigma_j|^{1/2}} \exp(-\frac{1}{2}X^T\Sigma_j^{-1}X) \tag{3}$$

where $X = (f_\Theta(x) - \mu_j)$. For the case $\Sigma_j = \sigma^2 I$, where I is the identity matrix:

$$p(x|w_j, \mu_j, \sigma_j) = \frac{1}{\sqrt{(2\pi)^n}\sigma_j} \exp(-\frac{||f_\Theta(x) - \mu_j||^2}{2\sigma_j^2}) \tag{4}$$

In a supervised problem, we know the *a posteriori* probability $P(w_j|x)$ for the input set. From this, we can define our structured loss function as the mean square error between the *a posteriori* probability of the input set and the *a posteriori* probability estimated for the embedded space:

$$\mathcal{L}_{rep} = \mathbb{E}\left\{||P(w_j|f_\Theta(x_i)) - P(w_j|x_i)||_2^2\right\} \tag{5}$$

3.4. Synthetic image generator

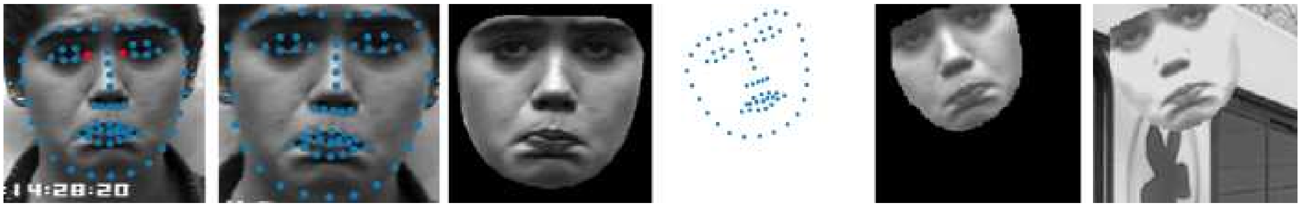
A limiting problem of currently available face expression datasets for supervised learning is the reduced number of correctly labeled data. We propose a data augmentation strategy to mitigate this problem in the lines of what has been introduced in [5]. Our image renderer R creates a synthetic larger dataset using real face datasets by making background changes and geometric transformations of face images. The example in Figure 4 shows a synthetic image generated pipeline by combining an example face of the CK+ dataset and a background image.

The generator method is limited to make low-level features that represent small variations in the facial expression space for the classification module. However, it allows creating a good number of examples to train our end-to-end system, having a larger contribution to the attention component. In the future we plan to include high-level features using GAN from the generated masks [13].

The renderer R adjusts the illumination of the face image so that it is inserted in the scene more realistically. An alpha matte step is applied in the construction of the final composite image of face and background. The luminance channel of the image face model I_{face} is adjusted by multiplying it by the factor $\frac{I_r}{I_{face}}$ where I_r is the luminance of the region that contains the face in the original image.

4. Experiments

We describe here the creation of the dataset used for training our network and its implementation details. We discuss two groups of experimental results: (1) Expression recognition result, to measure the performance of the method regarding the relevance of the attention module and the proposed loss function, and (2) Correction result, to analyze the robustness to noise.



(a) (b) (c) (d) (e) (f)

Figure 4: The pipeline of the synthetic image generation. The horizontal alignment of the image (b) is based on the inner points of the eyes (red points in (a)). The face is obtained as the convex hull of the landmarks set (c) and a random transform matrix is generated (d). The face image is projected on the background image (e). A face image and a general cropped background image are combined to generate a composite image (f). By using distinct background images for every face image, we are able to generate a much larger training data set. We create a large quantity of synthetic new images for every face of a database: approximately 9,231 synthetic images are generated for each face in the CK+ database, and 5,000 for the BU-3DFE database. This covers a great variety of possible tones and different backgrounds.

4.1. Datasets

We employ two public facial expression datasets, namely Extended Cohn-Kanade (CK+) [24] and BU-3DFE [41] to evaluate our method. we apply in all experiments person-independent FER scenarios [45]. Subjects in the training set are completely different from the subjects in the test set, i.e., the subjects used for training are not used for testing. The CK+ dataset includes 593 image sequences from 123 subjects. We selected 325 sequences of 118 subjects from this set, which meet the criteria for one of the seven emotions [24]. The selected 325 sequences consist of 45 Angry, 18 Contempt, 58 Disgust, 25 Fear, 69 Happy, 28 Sadness and 82 Surprise [24] facial expressions. In the neutral face expression case, we selected the first frame of the sequence of 33 random selected subjects. The BU-3DFE dataset is known to be challenging mainly due to a variety of ethnic/racial ancestries and expression intensity [41]. A total of 600 expressive face images (1 intensity x 6 expressions x 100 subjects) and 100 neutral face expression images, one for each subject, were used [41].

We employed our renderer R to augment training data for the neural network. R uses a facial expression dataset (we use BU-3DFE and CK+, which were segmented to obtain face masks) and a dataset of background images chosen from the COCO dataset. Figure 5 shows some examples of images generated by the renderer on the BU-3DFE dataset.

Figure 5: Examples from the synthetic BU-3DFE dataset. Different faces are transformed and combined with randomly selected background images from the COCO dataset. We then augment images after transformation by changing brightness and contrast and applying Gaussian blur and noise.

4.2. Implementation and training details

In all experiments, we considered the neural network architecture PreActResNet18 for the classification and representation processes. We adopted two approaches: (1) a model with attention and classification, FERAtt+Cls, and (2) a model with attention, classification, and representation, FERAtt+Rep+Cls. These models were compared with the classification results. For representation, the last convolutional layer of PreActResNet is evaluated by a linear layer to generate a vector of selected size. We have opted for 64 dimensions for the representation vector $\hat{z}$.

All models were trained on Nvidia GPUs (P100, K80, Titan XP) using PyTorch[3] for 60 epochs for the training set with 200 examples per mini batch and employing Adam optimizer. Face images were rescaled to 32×32 pixels. The code for the FERAtt is available in a public repository[4].

[3] http://pytorch.org/

[4] https://github.com/pedrodiamel/ferattention

		Synthetic				Real			
Database	Method	Acc.	Prec.	Rec.	F1	Acc.	Prec.	Rec.	F1
BU-3DFE	Baseline	69.37 ±2.84	71.48 ±1.46	69.56 ±2.76	70.50 ±2.05	75.22 ±4.60	77.58 ±3.72	75.49 ±4.68	76.52 ±4.19
	FERAtt+Cls	75.15 ±3.13	77.34 ±1.40	75.45 ±2.57	76.38 ±1.98	80.41 ±4.33	82.30 ±2.99	80.79 ±3.75	81.54 ±3.38
	FERAtt+Rep+Cls	**77.90** ±2.59	**79.58** ±1.77	**78.05** ±2.34	**78.81** ±2.01	**82.11** ±4.39	**83.72** ±3.09	**82.42** ±4.08	**83.06** ±3.59
CK+	Baseline	77.63 ±2.11	68.42 ±2.97	68.56 ±1.91	68.49 ±2.43	86.67 ±3.15	81.62 ±7.76	80.15 ±9.50	80.87 ±8.63
	FERAtt+Cls	84.60 ±0.93	74.94 ±0.38	76.30 ±1.19	75.61 ±0.76	85.42 ±2.89	75.65 ±2.77	78.79 ±2.30	77.18 ±2.55
	FERAtt+Rep+Cls	**85.15** ±1.07	**74.68** ±1.37	**77.45** ±0.55	**76.04** ±0.97	**90.30** ±1.36	**83.64** ±5.28	**84.90** ±8.52	**84.25** ±6.85

Table 1: Classification results for the Synthetic/Real BU-3DFE database (6 expression + neutral) and CK+ database (7 expression classes + neutral). Baseline: PreActResNet18[11], Acc.: Accuracy, Prec.: Precision, Rec.: Recall, F1: F1 measurement. Leave-10-subjects-out cross-validation is used for all experiments.

4.3. Expression recognition results

This set of experiments makes comparisons between a baseline architecture and the different variants of the proposed architecture. The objective is to evaluate the relevance of the attention module and the proposed loss function.

Protocol. We used different metrics to evaluate the proposed methods. Accuracy is calculated as the average number of successes divided by the total number of observations (in this case each face is considered an observation). Precision, recall, F1 score, and confusion matrix are also used in the analysis of the effectiveness of the system. Demšar [3] recommends the Friedman test followed by the pairwise Nemenyi test to compare multiple data. The Friedman test is a nonparametric alternative of the analysis of variance (ANOVA) test. The null hypothesis of the test H_0 stipulates that models are equivalent. Similar to the methods in [30], Leave-10-subject-out (L-10-SO) cross-validation was adopted in the evaluation.

Results. Tables 1 shows the mean and standard deviation for the results obtained on the real and synthetic datasets. For the BU-3DFE database the Friedman nonparametric ANOVA test reveals significant differences ($p = 0.0498$) between the methods. The Nemenyi post-hoc test was applied to determine which method present significant differences. The result for the Nemenyi post-hoc test (two-tailed test) shows that there are significant differences between the FERAtt+Cls+Rep and all the others, for a significance level at $\alpha < 0.05$.

In the CK+ database case, the Friedman test found significant differences between the methods with a level of significance of $p = 0.0388$ for the Synthetic CK+ dataset and $p = 0.0381$ for Real CK+ dataset. In this case, we applied the Bonferroni-Dunn post-hoc test (one-tailed test) to strengthen the power of the hypotheses test. For a significance level of 0.05, the Bonferroni-Dunn post-hoc test did not show significant differences between the FERAtt+Cls and the Baseline for Synthetic CK+ with $p = 0.0216$. When considering FERAtt+Rep+Cls and Baseline methods, it shows significant differences for the Real CK+ dataset with $p = 0.0133$.

Methods	Accuracy	NE
Lopes[23]	72.89	7†
Jampour[16]	78.64	7†
Zhang[47]	80.10	7†
Zhang[46]	80.95	7†
Our	**82.11**	7†

Table 2: Comparison of the average recognition accuracy with state-of-the-art FER methods for the BU-3DFE database. NE: number of expressions, †: six basic expressions + neutral class. Leave-10-subjects-out cross-validation is used for all methods.

Table 2 and 3 show the comparisons results between the different FER methods for the BU-3DFE database [40, 23, 16, 47, 46] and for the CK+ database [25, 44, 40]. Although some results cannot be directly compared due to different experimental setups, different expression classes and different preprocessing methods (e.g. face alignment), it is demonstrated that the proposed method can yield a feasible and promising recognition rate (around 82.11 percent for the BU-3DFE database and 90.30 for the CK+ database) with static facial images under person-independent recognition scenario.

The results shown in Figure 6 present the 64-dimensional

Methods	Accuracy*	NE
IACNN[25]	95.37	7‡
DSAE[44]	95.79 (93.78)	7‡
DeRL[40]	97.30 (96.57)	7‡
Our	**97.50**	7‡
DSAE[44]	89.84 (86.82)	8†
Our	**90.30**	8†

Table 3: Comparison of the average recognition accuracy with state-of-the-art FER methods for the CK+ database. NE: number of expressions, †: six basic expressions + neutral class and contempt class, ‡: six basic expressions + contempt class (neutral is excluded). $*$: the value in parentheses is the mean accuracy, which is calculated with the confusion matrix given by the authors. Leave-10-subjects-out cross-validation is used for all methods.

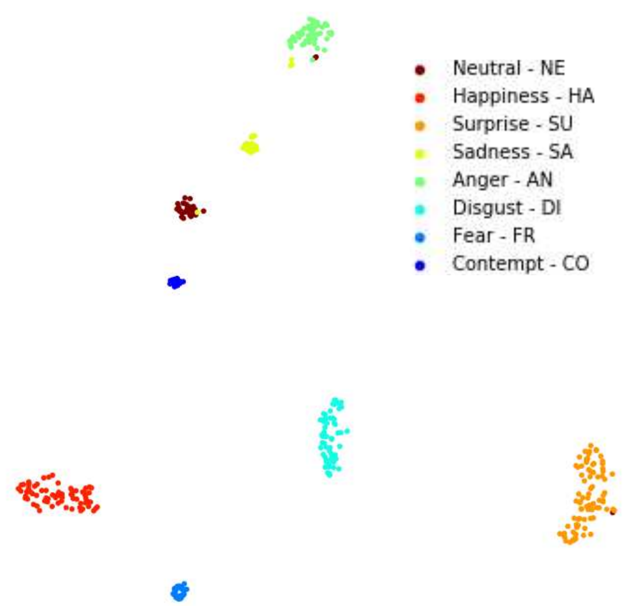


Figure 6: Barnes-Hut t-SNE visualization [37] of the Gaussian Structured loss for the Real CK+ database. Each color represents one of the eight emotions including neutral. Observe the clear separation of classes with a really small amount of misclassified neutral images. The distance between classes presented in the figure shows the expected. For example, happiness and anger are far apart while neutral appears approximately halfway between them.

embedded space using the Barnes-Hut t-SNE visualization scheme [37] of the Gaussian Structured loss for the Real CK+ dataset. Errors committed by the network are mostly due to the neutral class which is intrinsically similar to the other expressions we analyzed. Surprisingly, we observed intraclass separations into additional features, such as race, that were not taken into account when modeling or training the network. Figure 7 shows the top-5 retrieved images for random actor of the queries on Real CK+ database. We can see, that some actors are repeated in each query (not all actors have all expressions image on the database). This shows that the global features of the face are present in the obtained representation. It is also observed that these actors do not always appear in the same positions, which shows that there are lower level features that determine the similarity degree.

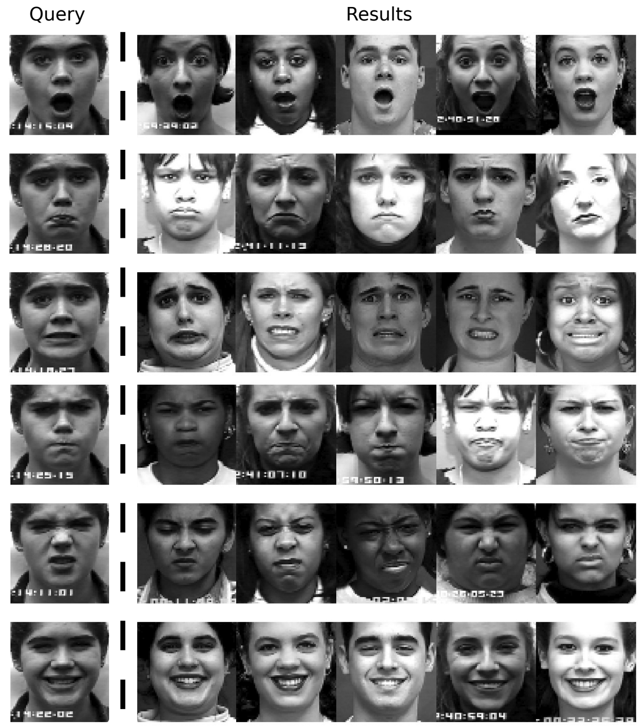


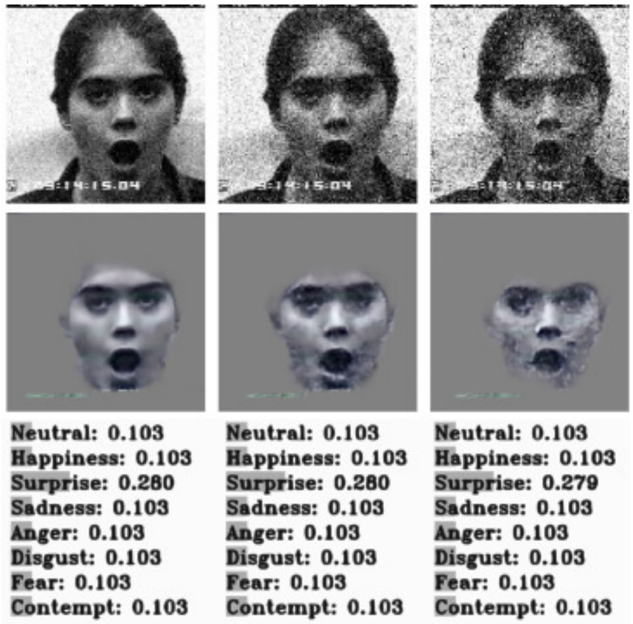
Figure 7: Top-5 images retrieved using FERAtt+Rep+Cls for the Real CK+ database embedded vectors.

4.4. Robustness to noise

The objective of this set of experiments is to demonstrate the robustness of our method to the presence of image noise when compared to the baseline architecture PreActResNet18.

Protocol. To carry out this experiment, the Baseline, FERAtt+Class, and FERAtt+Rep+Class models were trained on the Synthetic CK+ dataset. Each of these models was readjusted with increasing noise in the training set ($\sigma \in [0.05, 0.30]$). We maintained the parameters in the training for fine-tuning and used the real database CK+, so that 2000 images were generated for the synthetic dataset for test.

Results. One of the advantages of the proposed approach is that we can evaluate the robustness of the method under different noise levels by visually assessing the changes in the attention map I_{att}. Figure 8 shows the attention maps



(a) $\sigma = 0.10$ (b) $\sigma = 0.20$ (c) $\sigma = 0.30$

Figure 8: Attention maps I_{att} under increasing noise levels. We progressively added higher levels (increasing variance σ) of zero mean white Gaussian noise to the same image and tested them using our model. The classification numbers above show the robustness of the proposed approach under different noise levels, $\sigma = 0.10, 0.20, 0.30$, where the Surprise and all other scores are mostly maintained throughout all levels, with only a minor change of the Surprise score, from 0.280 to 0.279, occurring for the highest noise contamination of $\sigma = 0.30$.

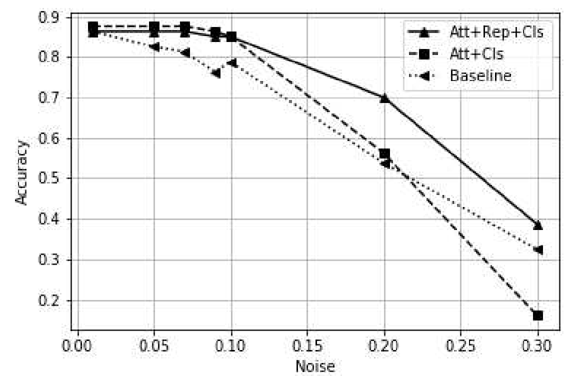


Figure 9: Classification accuracy after adding incremental noise on the Real CK+ dataset. Our approach results in higher accuracy when compared to the baseline, specially for stronger noise levels. Our representation model clearly leverages results showing its importance for classification. Plotted values are the average results for all 325 images in the database.

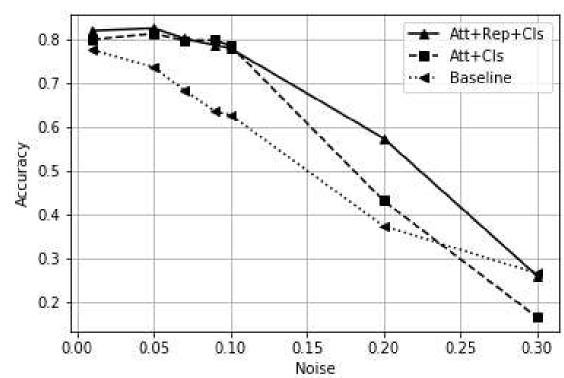


Figure 10: Average classification accuracy after adding incremental noise on the Synthetic CK+ dataset. The behavior of our method in the synthetic data replicates what we have found for the original Real CK+ database, *i.e.*, our method is superior to the baseline for all levels of noise. Plotted average values are for 2,000 synthetic images.

for an image for white zero mean Gaussian noise levels $\sigma = [0.01, 0.05, 0.07, 0.09, 0.1, 0.2, 0.3]$. We observe that our network is quite robust to noise for the range of 0.01 to 0.1 and maintains a distribution of homogeneous intensity values. This aspect is beneficial to the subsequent performance of the classification module. Figures 9 and 10 present classification accuracy results of the evaluated models in the Real CK+ dataset and for 2000 synthetic images. The proposed method FERAtt+CLs+Rep provides the best classification in both cases.

5. Conclusions

In this work, we present a new end-to-end neural network architecture with an attention model for facial expression recognition. We create a generator of synthetic images which is used for training our models. The results show that, for these experimental conditions, the attention module improves the system classification performance in comparison to other methods from the state-of-the-art. The loss function presented works as a regularization method on the embedded space. For future work, we plan to incorporate a transformer component in the architecture for automatic alignment of the face. We want to train the network for extreme condition such as dark light and occlusion.

6. Acknowledgment

The authors thanks the financial support from the Brazilian funding agency FACEPE and CETENE for usage of the computational facility.

References

[1] Brian Cheung, Eric Weiss, and Bruno Olshausen. Emergence of foveal image sampling from learning to attend in visual scenes. *arXiv preprint arXiv:1611.09430*, 2016.

[2] Peter Dayan, LF Abbott, et al. Theoretical neuroscience: computational and mathematical modeling of neural systems. *Journal of Cognitive Neuroscience*, 15(1):154–155, 2003.

[3] Janez Demšar. Statistical comparisons of classifiers over multiple data sets. *The Journal of Machine Learning Research*, 7:1–30, 2006.

[4] S. M. Ali Eslami, Nicolas Heess, Theophane Weber, Yuval Tassa, David Szepesvari, koray kavukcuoglu, and Geoffrey E Hinton. Attend, infer, repeat: Fast scene understanding with generative models. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 3225–3233. Curran Associates, Inc., 2016.

[5] Pedro D Marrero Fernández, Fidel A Guerrero Peña, Tsang Ing Ren, and Jorge JG Leandro. Fast and robust multiple colorchecker detection using deep convolutional neural networks. *Image and Vision Computing*, 81:15–24, 2019.

[6] Benjamin Graham. Fractional max-pooling. *arXiv preprint arXiv:1412.6071*, 2014.

[7] Alex Graves, Greg Wayne, Malcolm Reynolds, Tim Harley, Ivo Danihelka, Agnieszka Grabska-Barwińska, Sergio Gómez Colmenarejo, Edward Grefenstette, Tiago Ramalho, John Agapiou, et al. Hybrid computing using a neural network with dynamic external memory. *Nature*, 538(7626):471, 2016.

[8] Karol Gregor, Ivo Danihelka, Alex Graves, Danilo Jimenez Rezende, and Daan Wierstra. Draw: A recurrent neural network for image generation. *arXiv preprint arXiv:1502.04623*, 2015.

[9] Kaiming He, Georgia Gkioxari, Piotr Dollár, and Ross Girshick. Mask r-cnn. In *Computer Vision (ICCV), 2017 IEEE International Conference on*, pages 2980–2988. IEEE, 2017.

[10] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep residual learning for image recognition. *CoRR*, abs/1512.03385, 2015.

[11] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Identity mappings in deep residual networks. In *European conference on computer vision*, pages 630–645. Springer, 2016.

[12] Robert Highfield, Richard Wiseman, and Rob Jenkins. How your looks betray your personality. *New Scientist*, 201(2695):28–32, 2009.

[13] Yuchi Huang and Saad M Khan. Dyadgan: Generating facial expressions in dyadic interactions. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2017 IEEE Conference on*, pages 2259–2266. IEEE, 2017.

[14] Laurent Itti and Christof Koch. Computational modelling of visual attention. *Nature Reviews Neuroscience*, 2(3):194, 2001.

[15] Max Jaderberg, Karen Simonyan, Andrew Zisserman, and koray kavukcuoglu. Spatial transformer networks. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 2017–2025. Curran Associates, Inc., 2015.

[16] Mahdi Jampour, Thomas Mauthner, and Horst Bischof. Multi-view facial expressions recognition using local linear regression of sparse codes. In *Proceedings of the 20th Computer Vision Winter Workshop Paul Wohlhart*, 2015.

[17] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 25*, pages 1097–1105. Curran Associates, Inc., 2012.

[18] Shan Li and Weihong Deng. Deep facial expression recognition: A survey. *arXiv preprint arXiv:1804.08348*, 2018.

[19] Yijun Li, Jia-Bin Huang, Narendra Ahuja, and Ming-Hsuan Yang. Deep joint image filtering. In *European Conference on Computer Vision*, pages 154–169. Springer, 2016.

[20] Bee Lim, Sanghyun Son, Heewon Kim, Seungjun Nah, and Kyoung Mu Lee. Enhanced deep residual networks for single image super-resolution. In *The IEEE conference on computer vision and pattern recognition (CVPR) workshops*, volume 1, page 4, 2017.

[21] Mengyi Liu, Ruiping Wang, Shaoxin Li, Shiguang Shan, Zhiwu Huang, and Xilin Chen. Combining multiple kernel methods on riemannian manifold for emotion recognition in the wild. In *Proceedings of the 16th International Conference on Multimodal Interaction*, pages 494–501. ACM, 2014.

[22] Ping Liu, Shizhong Han, Zibo Meng, and Yan Tong. Facial expression recognition via a boosted deep belief network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1805–1812, 2014.

[23] André Teixeira Lopes, Edilson de Aguiar, Alberto F De Souza, and Thiago Oliveira-Santos. Facial expression recognition with convolutional neural networks: coping with few data and the training sample order. *Pattern Recognition*, 61:610–628, 2017.

[24] Patrick Lucey, Jeffrey F Cohn, Takeo Kanade, Jason Saragih, Zara Ambadar, and Iain Matthews. The Extended Cohn-Kanade Dataset (CK+): A complete dataset for action unit and emotion-specified expression. In *Computer Vision and Pattern Recognition Workshops (CVPRW), 2010 IEEE Computer Society Conference on*, pages 94–101. IEEE, 2010.

[25] Zibo Meng, Ping Liu, Jie Cai, Shizhong Han, and Yan Tong. Identity-aware convolutional neural network for facial expression recognition. In *2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017)*, pages 558–565. IEEE, 2017.

[26] Volodymyr Mnih, Nicolas Heess, Alex Graves, and koray kavukcuoglu. Recurrent models of visual attention. In Z. Ghahramani, M. Welling, C. Cortes, N. D. Lawrence, and K. Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27*, pages 2204–2212. Curran Associates, Inc., 2014.

[27] Ali Mollahosseini, David Chan, and Mohammad H Mahoor. Going deeper in facial expression recognition using

deep neural networks. In *Applications of Computer Vision (WACV), 2016 IEEE Winter Conference on*, pages 1–10. IEEE, 2016.

[28] Jinshan Pan, Sifei Liu, Deqing Sun, Jiawei Zhang, Yang Liu, Jimmy Ren, Zechao Li, Jinhui Tang, Huchuan Lu, Yu-Wing Tai, et al. Learning dual convolutional neural networks for low-level vision. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3070–3079, 2018.

[29] R.W. Picard. *Affective Computing*. MIT Press, 2000.

[30] Raymond Ptucha and Andreas Savakis. Manifold based sparse representation for facial understanding in natural images. *Image and Vision Computing*, 31(5):365–378, 2013.

[31] O. Ronneberger, P.Fischer, and T. Brox. U-net: Convolutional networks for biomedical image segmentation. In *Medical Image Computing and Computer-Assisted Intervention (MICCAI)*, volume 9351 of *LNCS*, pages 234–241. Springer, 2015. (available on arXiv:1505.04597 [cs.CV]).

[32] Olga Russakovsky, Jia Deng, Hao Su, Jonathan Krause, Sanjeev Satheesh, Sean Ma, Zhiheng Huang, Andrej Karpathy, Aditya Khosla, Michael Bernstein, Alexander C. Berg, and Li Fei-Fei. Imagenet large scale visual recognition challenge. *International Journal of Computer Vision*, 115(3):211–252, Dec 2015.

[33] Assaf Shocher, Nadav Cohen, and Michal Irani. Zero-shot super-resolution using deep internal learning. In *Conference on computer vision and pattern recognition (CVPR)*, 2018.

[34] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, 2014.

[35] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2015.

[36] Sabine Kastner Ungerleider and Leslie G. Mechanisms of visual attention in the human cortex. *Annual review of neuroscience*, 23(1):315–341, 2000.

[37] Laurens Van Der Maaten. Accelerating t-sne using tree-based algorithms. *Journal of machine learning research*, 15(1):3221–3245, 2014.

[38] Oriol Vinyals, Ł ukasz Kaiser, Terry Koo, Slav Petrov, Ilya Sutskever, and Geoffrey Hinton. Grammar as a foreign language. In C. Cortes, N. D. Lawrence, D. D. Lee, M. Sugiyama, and R. Garnett, editors, *Advances in Neural Information Processing Systems 28*, pages 2773–2781. Curran Associates, Inc., 2015.

[39] Fei Wang, Mengqing Jiang, Chen Qian, Shuo Yang, Cheng Li, Honggang Zhang, Xiaogang Wang, and Xiaoou Tang. Residual attention network for image classification. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3156–3164, 2017.

[40] Huiyuan Yang, Umur Ciftci, and Lijun Yin. Facial expression recognition by de-expression residue learning. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2168–2177, 2018.

[41] Lijun Yin, Xiaozhou Wei, Yi Sun, Jun Wang, and Matthew J Rosato. A 3D facial expression database for facial behavior research. In *Automatic face and gesture recognition, 2006. FGR 2006. 7th international conference on*, pages 211–216. IEEE, 2006.

[42] Zhiding Yu and Cha Zhang. Image based static facial expression recognition with multiple deep network learning. In *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*, ICMI '15, pages 435–442, New York, NY, USA, 2015. ACM.

[43] Zhiding Yu and Cha Zhang. Image based static facial expression recognition with multiple deep network learning. In *Proceedings of the 2015 ACM on International Conference on Multimodal Interaction*, pages 435–442. ACM, 2015.

[44] Nianyin Zeng, Hong Zhang, Baoye Song, Weibo Liu, Yurong Li, and Abdullah M Dobaie. Facial expression recognition via learning deep sparse autoencoders. *Neurocomputing*, 273:643–649, 2018.

[45] Zhihong Zeng, Maja Pantic, Glenn Roisman, Thomas S Huang, and others. A survey of affect recognition methods: Audio, visual, and spontaneous expressions. *Pattern Analysis and Machine Intelligence, IEEE Transactions on*, 31(1):39–58, 2009.

[46] Feifei Zhang, Tianzhu Zhang, Qirong Mao, and Changsheng Xu. Joint pose and expression modeling for facial expression recognition. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 3359–3368, 2018.

[47] Tong Zhang, Wenming Zheng, Zhen Cui, Yuan Zong, Jingwei Yan, and Keyu Yan. A deep neural network-driven feature learning method for multi-view facial expression recognition. *IEEE Transactions on Multimedia*, 18(12):2528–2536, 2016.

[48] Shizhan Zhu, Sifei Liu, Chen Change Loy, and Xiaoou Tang. Deep cascaded bi-network for face hallucination. In *European Conference on Computer Vision*, pages 614–630. Springer, 2016.