



Performance comparisons of artificial neural network algorithms in facial expression recognition

Amira ElsirTayfour ^{1*}, Altahir Mohammed ², Moawia Elfaki Eldow ³

¹ PhD Research Scholar, King Khalid University

² Sudan University of Science & Technology

³ University of Khartoum

*Corresponding author E-mail: amtyfour@gmail.com

Copyright © 2015 Amira ElsirTayfour et al. This is an open access article distributed under the [Creative Commons Attribution License](#), which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Abstract

This paper presents methods for identifying facial expressions. The objective of this paper is to present a combination of texture oriented method with dimensional reduction and use for training the Single-Layer Neural Network (SLN), Back Propagation Algorithm (BPA) and Cerebellar Model Articulation Controller (CMAC) for identifying facial expressions. The proposed methods are called intelligent methods that can accommodate for the variations in the facial expressions and hence prove to be better for untrained facial expressions. Conventional methods have limitations that facial expressions should follow some constraints. To achieve the expression detection accuracy, Gabor wavelet is used in different angles to extract possible textures of the facial expression. The higher dimensions of the extracted texture features are further reduced by using Fisher's linear discriminant function for increasing the accuracy of the proposed method. Fisher's linear discriminant function is used for transforming higher-dimensional feature vector into a two-dimensional vector for training proposed algorithms. Different facial emotions considered are angry, disgust, happy, sad, surprise and fear are used. The performance comparisons of the proposed algorithms are presented.

Keywords: Fisher's Linear Discriminant Function; Wavelet Gabor Filter; Artificial Neural Network.

1. Introduction

Identification of facial expressions has been a main factor used in clinical treatments, security, information finding from unknown persons and interacting with fellow beings. In many cases, it would be difficult to identify the expression using the manual method, especially when lots of expressions are available in different images. Hence, an automatic method is required to be provided that can accommodate the change in expression and accurately identify the expression.

2. Related work

Standard methods like static and dynamic techniques have been used earlier by researchers in identifying emotions. Bayesian technique has been used as an important static method. Ravi et al., [5], gave a comparative study and analysis of 'Facial Expression Recognition Technology' along with its progressive growth and developments. Oliveira et al.,[4], proposed a novel method called two-dimensional discriminant locality preserving projections (2D-DLPP) is proposed that can best discriminate different pattern classes. Cheng et al, [2], proposed a Gaussian Process model for the facial expression recognition in the Japanese female facial expression data set and found successful classification of facial expression.

Klaus and Ursula, [3], report the development of a rapid test of emotion recognition ability, the Emotion Recognition Index (ERI), consisting of two subtests: one for facial and one for vocal emotion recognition. Ruffman, [6], presents that recognition of emotion in still photos provides important information about young-old differences and has

sufficient ecological validity to explain age differences in a number of social insights. Bänziger Bänziger et al., [1], discusses an overview of some of the major emotion expression (EE) corpora currently available for empirical research and introduces a new, dynamic, multimodal corpus of emotion expressions, the Geneva Multimodal Emotion Portrayals Core Set (GEMEP-CS). Schlegel et al., [7], studied on emotion recognition ability (ERA) that can inform the measurement of the emotion perception component in emotional intelligence.

Katja et al., [11] state that existing tests to measure the ability to recognize other people's emotional expressions (emotion recognition ability [ERA]) mostly focus on a single modality (usually the face) and include only a small number of emotions, restricting their ecological validity. Further, their reliability is often unsatisfactory.

Scott and Honglak, [12] had used deep neural networks in recognizing emotion recognition.

3. Problem definition

Automated facial expression analysis is a vast research field. Pantic, 2000, [8], [9], provided a comprehensive overview of the state of the art and define problems to expression analysis.

- 1) Face detection in an image or image sequence.
- 2) Facial expression data extraction.
- 3) Facial expression classification. Face detection is dealt with implicitly. For still images, it is assumed that these are faces. For live video, face tracker automatically localizes facial features only if a face is present in the video stream.
- 4) The pose of the subject, the position of the face can cause some parts of the face such as eyes and nose occluded in the captured image, Yang, 2002, [10].
- 5) The system should be invariant to different lightning conditions and distraction as glasses, change in hair style, facial hair, mustache, and beard. The features like beard, mustache, and glasses may not result in the correct facial expression because these characteristics are subjects to change.
- 6) Person's affective natural changes with time could cause some difficulties in the machine's correct classification
- 7) Faces may be partially occluded by other objects such as scarf and hat.

4. The system setup

The implementation of the expression identification system includes detection of the image, training and testing the images for identification.

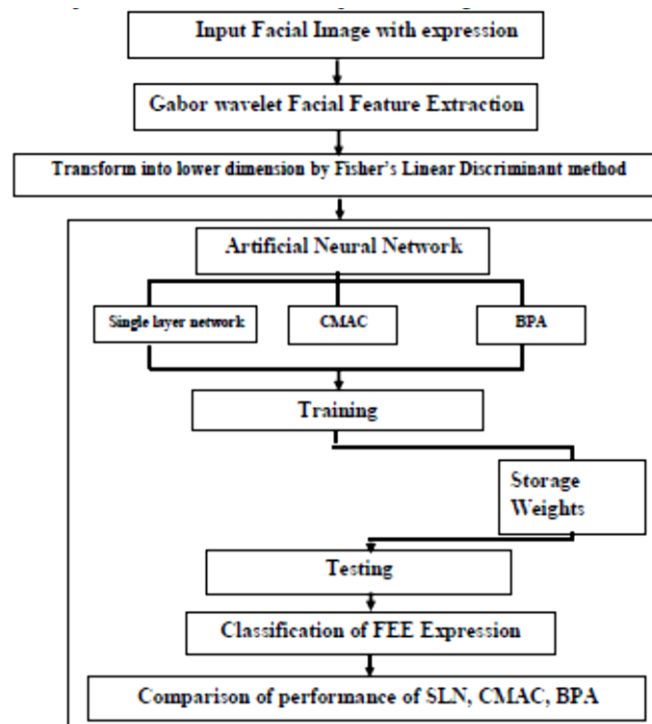


Fig. 1: FEE Identification Sequence.

Working Details of implemented Architecture (Figure 1):

Step 1: Frames are extracted from an image.

Step 2: Important features are extracted from successive frames that belong to one second.

Step 3: The features are trained using SLN/ CMAC/BPA and final weights are stored in a database.

Step 4: In the testing process, step 1 and step 2 are adopted. The extracted features are processed with final weights of the SLN/ CMAC/BPA, to get an output in the output layer of the SLN/ CMAC/BPA.

Step 5: The output is compared with a threshold value, to decide the category to which the particular emotion the facial expression belongs.

4.1. Image detection

A face has to be detected in a captured image. Once detected, the image region containing the face is extracted and geometrically normalized. Images have been acquired using a standard digital camera. The expression of a person under different conditions are presented in Figures 1-6 shows some of the training images.

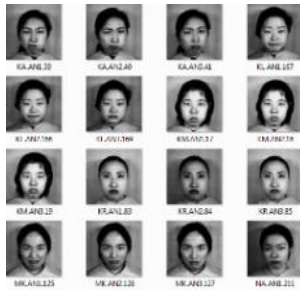


Fig. 2: Angry Expression for Different Situations.

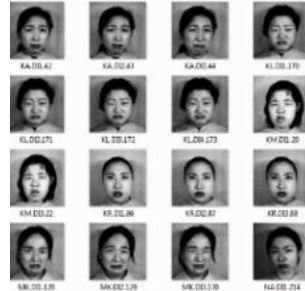


Fig. 3: Disgust Expression in Different Situations.

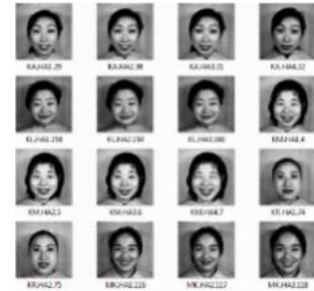


Fig. 4: Happy Expression in Different Situations.



Fig. 5: Sad Expression in Different Situations.

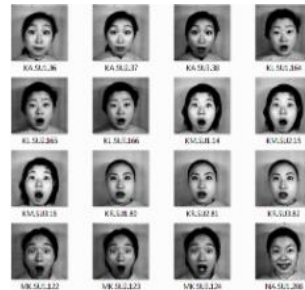


Fig. 6: Surprise Expression in Different Situations.



Fig. 7: Fear Expression in Different Situations.

4.2. Feature extraction by rotational wavelet Gabor filter

Facial texture segmentation plays a significant role in recognizing and identifying a material, type of characteristic for a particular image. Wavelets are employed for the computation of single and multi-scale roughness features because of their ability to extract information at different resolutions. Features are extracted in multiple directions using directional wavelet obtained from the partial derivative of the Gaussian distribution function. The first and second derivative wavelets are used to get the characteristics of the textured image at different orientations like 0°, 90°, 45° and 135° and scales such as 1, 2 and 4.

Facial segmentation procedure partition an image into a constituent object that is used to find the regions of interests using K-means algorithm.

4.2.1. Preprocessing

Preprocessing is done for the removal of noise from an image using the Gaussian smoothing function. In designing Gaussian filters, the mask weights are computed directly from the Gaussian distribution, given by equation (1)

$$\phi(x, y, s) = \exp\left\{-\frac{x^2 + y^2}{2s^2}\right\} \tag{1}$$

Where x, y are directions, and s is the scale?

An overlapping moving window size of NxN is used for preprocessing. The coefficient value at the center of the mask is made equal to one by a suitable multiplication factor. When performing the convolution, the output pixel values must be normalized by the sum of the mask weights to ensure that regions of uniform intensity are not affected.

Figure 8 shows the final output of the Gabor wavelet output.

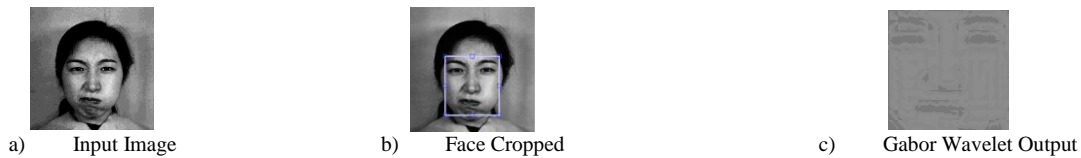


Fig. 8: Final Output of Wavelet Gabor Texture Extraction.

The output of wavelet Gabor filter is used as input for Fisher's linear discriminant function for obtaining transformed higher dimensional feature vector into two-dimensional vector by the computation of discriminant vectors ϕ_1 and ϕ_2 . The Fisher's criterion is given by equation (2).

$$J(\phi) = \frac{\phi^T S_b \phi}{\phi^T S_w \phi} \quad (2)$$

The 2-dimensional vectors set are denoted by Y_i . The vector Y_i is given by equation (3).

$$Y_i = (u_i, v_i) = \{X_i^T \phi_1, X_i^T \phi_2\} \quad (3)$$

The vector set Y_i , is obtained by projecting the original vector 'X (output of Gabor wavelet)' of the patterns onto the space spanned by ϕ_1 and ϕ_2 by using equation 3.

4.2.2. Training and testing using artificial neural network algorithms

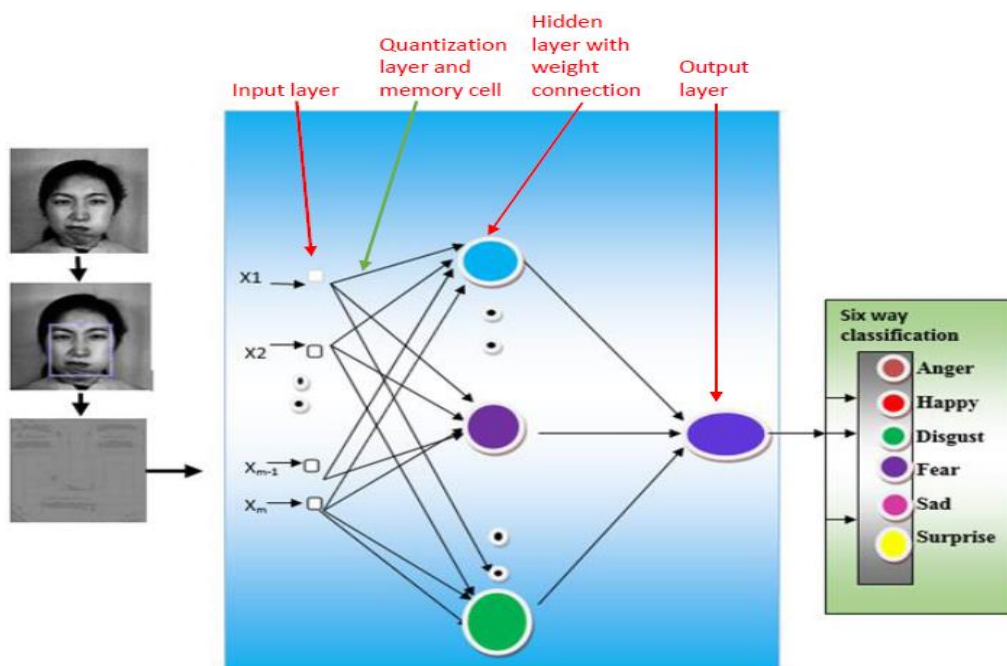


Fig. 9: Flow of Information in ANN.

The Figure 9 presents the flow of information in the ANN. It represents a general topology of ANN with one input layer, one hidden layer, and one output layer. The input layer takes the reduced two-dimensional input vector obtained by FLD.

a) Back propagation algorithm (BPA)

The BPA works based on reaching global minimum criteria. It uses steepest-descent method for minimizing the error among target values and the values obtained in the output layer. Figure 9 is a common ANN topology. In this topology, except quantization layer and memory cell all the layers are present for implementing BPA. The training of BPA consists of two stages. Stage-1 involves forward computation from input layer till output layer. If there is an error between target value and the obtained value, then error is propagated backwards from output layer till input layer for adjusting the weights. This results in reduced error in the output layer. All the patterns under training are presented to the input layer of BPA and the two stages of computation are implemented. This forms one iteration. Iterations are increased by presenting the training patterns to the network until the error is minimized to a specified value.

The number of nodes in the hidden layer of BPA is fixed based on the maximum number of emotion images recognized. Simulation of the number of nodes in the hidden layer is changed from 2 nodes to 20 nodes. More than 20 nodes is not simulated in the BPA training.

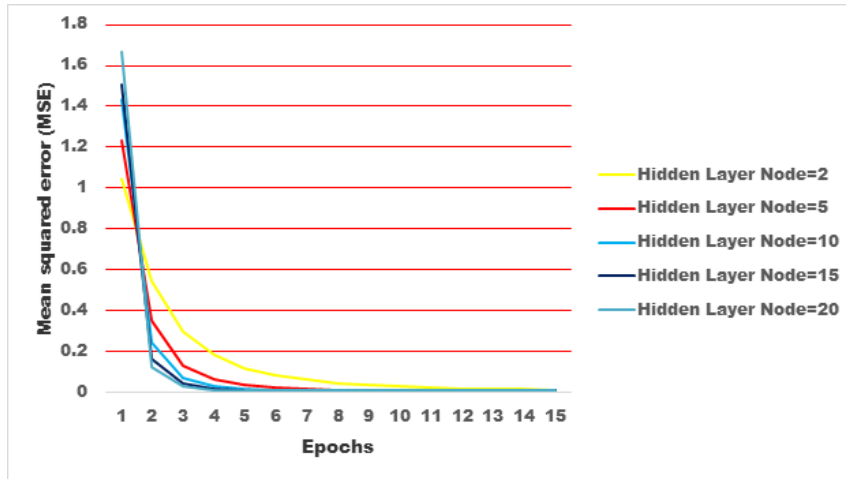


Fig. 10: Epochs versus Mean Squared Error.

Figure 10 presents the convergence curve for BPA with different number of nodes used in the hidden layer for training the FLD feature obtained from the angry images. The hidden layer with 20 nodes results in more convergence of the error. Other convergence curve indicates less convergence and hence more epochs taken to reach the desired error value.

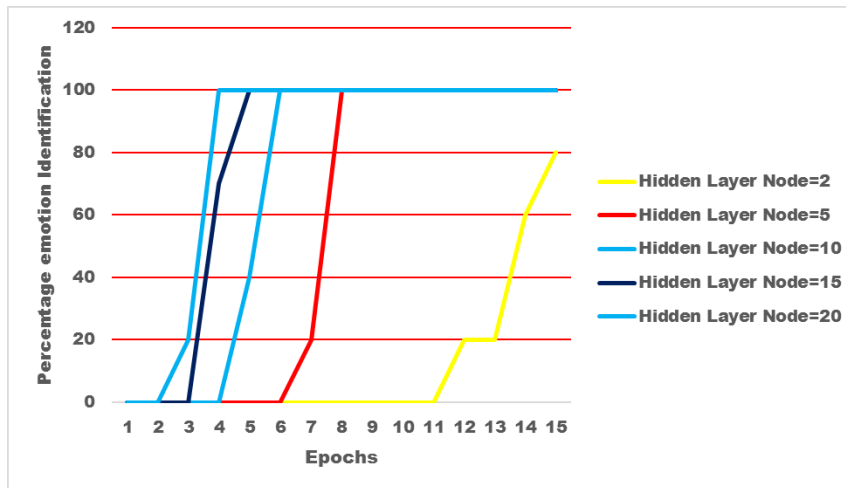


Fig. 11: Epochs versus Percentage of FEE Classification.

Figure 11 shows the improvement in correct recognition of total number of Facial emotion expression considering all test images as the number of iterations increase to reach specified convergence criteria. The recognition performance of BPA for different number of nodes in the hidden layer is shown in Figure 11.

b) Single layer network (SLN)

The SLN ANN topology does not contain hidden layer. The SLN is also called functional update method. In Figure 9, when only input layer and output layer are considered then it is called single layer network (SLN). During the presentation of each training pattern, the weights of the network are updated, when a functional criterion is satisfied. Otherwise, no updation of the weights is done. The patterns are presented in the binary form.

The algorithm for the functional update is as follows:

- Step 1: The weights and thresholds of the network are initialized.
- Step 2: The inputs and outputs of a pattern are presented to the network.
- Step 3: The output of each node in the successive layers is calculated by:

$$x_i = 1 / (1 + \exp(\sum w_{ij} x_j + \theta_i)) \tag{4}$$

Step 4: The number of nodes in the output layer, which are misclassified, are denoted by 'nm'. A node is misclassified, if it does not satisfy the equation:

$$1 - \epsilon > D \geq 0.5 \tag{5}$$

Where ϵ is the value fixed by the programmer, and

$$D = | \text{Desired output} - \text{Network output} | \quad (6)$$

If 'nm' is empty, i.e., not even one node satisfies equation 3.2, step 2 is adopted.

Step 5: If 'nm' is not empty, the objective function 'j' is computed by:

$$J = \frac{1}{2} \sum \sum_{Xi \in nm} D^2 \quad (7)$$

Step 6: The weights and thresholds are updated.

Step 7: The steps(2-6) are adopted, until the total MSE of all the patterns is below a specified value.

c) Cerebellar Model Articulation Controller (CMAC)

CMAC is an associative memory artificial neural network algorithm which maps inputs and corresponding outputs as any other network would do. However, the specialty of CMAC is that, the input pattern is further quantized in an overlapping fashion. Figure 9 shows the complete topology for implementing CMAC. In the testing process, input patterns are presented in the input layer of CMAC architecture. The values of the patterns are searched in the already quantized space and the corresponding space numbers of the different values of the input pattern is noted. The locations of the different spaces corresponding to the original input pattern is now treated as the input pattern for subsequent processing. Weights corresponding to the different space locations are summarized linearly to produce required outputs.

Steps involved in implementing CMAC:

Step 1: Present a pattern with two dimensional values obtained from FLD.

Step 2: quantize each value of the pattern into many ranges. For example, 0.1 is quantized into (0.01 -0.05), (>0.05 till 0.08), (>0.08 till <0.12). So, in this example, three quantization ranges are created. The value can be associated to range 3 (>0.08 till <0.12). Hence, 0.1 can be [0, 0, and 1]. Similarly, if the second dimensional values obtained from FLD is 0.07, then 0.07 can be [1, 0, and 0]. Hence, the two FLD value given in the input layer with two input nodes, has been converted into 6 valued pattern [0, 0, 1, 1, 0, and 0]. This pattern is further used to propagate forward till the output layer of the CMAC topology. An error is calculated in the output layer which is used to propagate backwards for updating the connection weights.

Step 3: In the testing process, new pattern is presented and the quantization process is carried out. All the weight connections already updated during the training process are selectively chosen based on the presence of 1 in the quantized pattern to obtain an output in the output layer. The output is further compared with a template to identify the FEE.

4.3. Performance analysis

The accuracy refers to how correctly; the proposed algorithms classify the FEE in a video. Different measures like precision and recall can be used to evaluate classification of emotion expressions. However, in this work the facial emotion expression classification accuracy is expressed as follows:

$$\text{Sensitivity} = \frac{TP}{TP+FN} \quad (8)$$

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \quad (9)$$

Detection refers to presence of an FEE.

Classification refers to labelling Angry/sad/happy/fear/surprise/disgust.

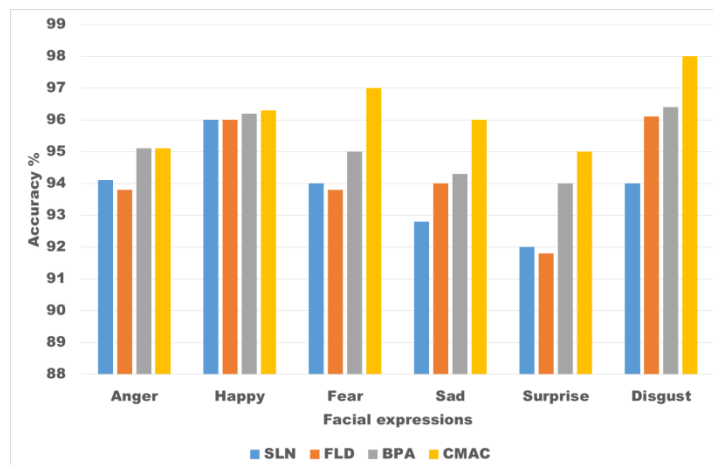


Fig. 12: Performance Comparison of 4 Algorithms for 6 FEE.

Figure 12 presents the performance comparisons of FLD and three ANN algorithms: SLN, BPA and CMAC. CMAC presents highest accuracy in identifying all the 6 expressions better when compared to other three algorithms. The reason for CMAC to be better is that, sub quantization methods are used before training the input patterns. Hence, more memory space is used to store the input information.

5. Conclusion

In this paper, sample images with emotions are used for training and testing the proposed system of facial expression recognition. Wavelet Gabor filter, Fisher's linear discriminant function and SLN, BPA ANN are used to implement the system. The performance of the system is purely based on the quality of the images. Most of the ANN algorithms work based on the multilayer architecture. In this manuscript, in addition to the multilayer architecture (BPA, CMAC), a comparison with the single layer architecture (SLN) has been done. This is to show if there is any possibility of showing improved performance using SLN. Hence, we used SLN also for comparison purposes. Commercial facial emotion recognition software incorporates many algorithms for processing the image. If the proposed neural network algorithms are implemented in the commercial software, there will be improvements in the software performance. The future work includes in analyzing the proposed system and its suitability for people with different origins.

References

- [1] Bänziger, T., Mortillaro, M., & Scherer, K. R. (2012). Introducing the Geneva Multimodal expression corpus for experimental research on emotion perception. *Emotion*, 12, 1161–1179. <http://dx.doi.org/10.1037/a0025827>.
- [2] Cheng, F., Yu, J., & Xiong, H. (2010). Facial expression recognition in jaffe dataset based on Gaussian process classification. *IEEE Transactions on Neural Networks*, 21(10), 1685–1690. <http://dx.doi.org/10.1109/TNN.2010.2064176>.
- [3] Klaus R. Scherer and Ursula Scherer, 2011, Assessing the Ability to Recognize Facial and Vocal Expressions of Emotion: Construction and Validation of the Emotion Recognition Index, *J Nonverbal Behav.*, Vol.35, pp.305–326., DOI 10.1007/s10919-011-0115-4. <http://dx.doi.org/10.1007/s10919-011-0115-4>.
- [4] Oliveira, L. E. S., Koerich, A. L., Mansano, M., & Britto, A. S. Jr., (2011). 2d principal component analysis for face and facial expression recognition. *Computing in Science and Engineering*, 13(3), 9–13. <http://dx.doi.org/10.1109/MCSE.2010.149>.
- [5] Ravi S., and Mahima S., 2011, Study of the Changing Trends in Facial Expression Recognition, *International Journal of Computer Applications* (0975-8887), Vol.21. No.5, pp.10-16.
- [6] Ruffman, T. (2011). Ecological validity and age-related change in emotion recognition. *Journal of Nonverbal Behavior*, 35, 297–304. <http://dx.doi.org/10.1007/s10919-011-0116-3>.
- [7] Schlegel, K., Grandjean, D., & Scherer, K. R. (2012). Emotion recognition: Unidimensional ability or a set of modality- and emotion-specific skills? *Personality and Individual Differences*, 53, 16–21. <http://dx.doi.org/10.1016/j.paid.2012.01.026>.
- [8] Pantic M., and Rothkrantz., 2000, Automatic analysis of facial expressions: The state of the art, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol.22, No.12, pp.1424-1445. <http://dx.doi.org/10.1109/34.895976>.
- [9] Pantic M., and Rothkrantz L.J.M., 2000, Expert system for Automatic Analysis of facial Expressions, *ELSEVIER, Image and Vision Computing*, Vol.18, pp.881-905 [http://dx.doi.org/10.1016/S0262-8856\(00\)00034-2](http://dx.doi.org/10.1016/S0262-8856(00)00034-2).
- [10] Yang M. H., Kriegman D. J., and Ahuja N., 2002, Detecting Faces in Images: A Survey, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, Vol. 24, No. 1, pp.34-58. <http://dx.doi.org/10.1109/34.982883>.
- [11] Katja Schlegel, Didier Grandjean, and Klaus R. Scherer, 2014, Introducing the Geneva Emotion Recognition Test: An Example of Rasch-Based Test Development, *Psychological Assessment*, Vol.26, No.2, pp.666–672. <http://dx.doi.org/10.1037/a0035246>.
- [12] Scott E. Reed and Honglak Lee, Training deep neural networks on noisy labels with bootstrapping, Accepted as a workshop contribution at ICLR 2015, pp.1-11.