SFEW Dataset Based Facial Emotion Classification Using BiDirectional Neural Network and Convolutional Neural Network

Chujie Wu

Research School of Computer Science
Australian National University, Canberra Australia
u6920829.com

Abstract. The Static Facial Expressions in the Wild (SFEW) database is of significant research value in the field of facial expression analysis for its high similarity to real world scenarios and covering unconstrained object features like varied facial expressions, head poses, large age range etc. We construct two classification models to classify facial expressions into seven classes including angry, disgust, fear, happy, sad, surprise and the neutral, in SFEW. The first model is based on a bidirectional neural network and it takes principle components of both descriptors local phase quantization(LPQ) and pyramid of histogram of oriented gradients (PHOG) representing the images in SFEW as explanatory variables. And we found the bidirectional neural network model achieves better performance for classification task than the non-bidirectional model, which indicates the bidirectional transmission's effective role in neural network model training. The second model is based on a deep learning approach that uses a convolutional neural network to train the centered face image data of SFEW. Experiments show the deep learning approach achieved similar accuracy results to BDNN method. In this classification task, CNN presents a better performance in handling larger scale image data.

Keywords: Convolutional Neural Network · Bidirectional Neural Network · Facial Emotion Classification· Facial Recognition· Static Facial Expressions in the Wild (SFEW)

1 Introduction

Facial expressions are one of the most natural, powerful, and immediate ways for people to communicate their emotions[1]. Within the past decade, developing methods of facial expression analysis has been a big focus and there have been significant effort taken in. Facial expression analysis includes both the measurement of facial motion and the recognition of facial expressions, which are generated by the change in a person's facial muscles[2]. It has a broad application space and prospect in fields like human computer interaction(HCI), medical psychology therapy and human behavior research.

Most facial expression data are captured in lab-controlled environment and collected by asking subjects consciously perform certain expressions. However, in reality, facial expression is more complex that it is often closely related to external conditions like the angle of head pose, the posture of the limbs, the lighting of the realistic scene and so on, which are hard to be precisely simulated and captured in lab-controlled conditions[3]. Therefore, the static facial expression database Static Facial Expressions in the Wild (SFEW) is of significant research value in the field of facial expression analysis for it is captured based on movie frames that has high similarity to real world scenarios and covers unconstrained object features like varied facial expressions, head poses and movement, varied illumination, age ,gender and occlusion etc.

Although the facial expression way can vary from culture to culture, it is widely accepted that there are six universal expressions including happiness, sadness, disgust, anger, surprise and fear don't change among cultures[4]. These six classes can be combined with the neutral class and regarded as a general facial expression classification labels.

1.1 Dataset Details

The Static Facial Expressions in the Wild (SFEW) dataset we use contains 675 image extracted from 37 different movies [2] and they're all labeled for seven basic expression classes. Different from most datasets recorded in lab environments, SFEW addresses the issue of static facial expressions in difficult conditions that are approximating real world conditions. It is more realistic and covers unconstrained facial expressions, varied head poses, large age range, occlusions, varied focus, different resolution of face and close to real world illumination.

Each image can also be represented by five principle components of descriptor local phase quantization LPQ along with five principle components of the descriptor pyramid of histogram of oriented gradients (PHOG). LPQ is a feature commonly in texture classification and is invariant to blur and illumination[5]. The PHOG feature counts occurrences of gradient orientation in localized portion of an image and it has been proved to have good performance in object recognition[6,7]. Both descriptors are reliable and representative and we take them as explanatory variables in classification task. And the label of each image is the truth value of expression class.

1.2 Outline of Investigation

Neural networks are made of interconnected processing neurons working in parallel and have profound success and wide application in performing classification, pattern recognition or prediction tasks on the basis of input data.

In our first attempt, we implement a bidirectional neural network based on two-layer perceptron trained by error back-propagation algorithm and use it to classify facial expressions in SFEW based on the values of descriptors. The investigating procedure is illustrated in Figure 1.

A bidirectional neural network can be regard as an extension of basic neural network as it simulates bidirectional electrical signals transmissions in human brains and is able to remember input patterns as well as output vectors, given either of them[8].

We use z-score normalization to preprocessing the data. It converts all indicators to a common scale with an average of zero and standard deviation of one which helps avoids introducing aggregation distortions stemming from differences in indicators' means. Five-fold cross validation is used to effectively avoid the overfitting and underfitting and evaluate estimator performance.

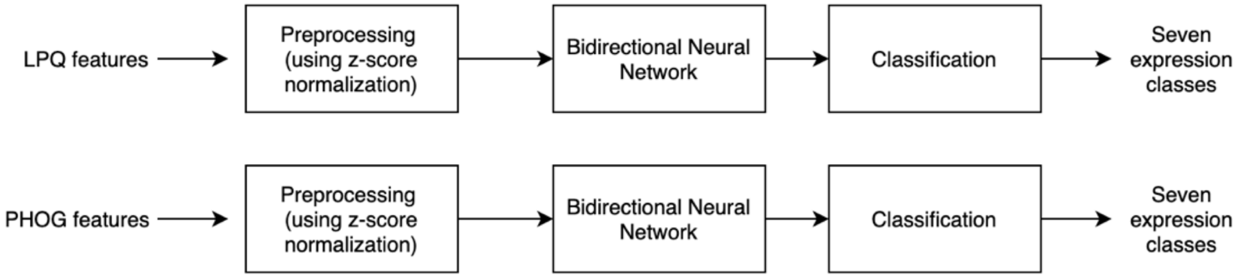

LPQ features
Preprocessing (using z-score normalization)
Bidirectional Neural Network
Classification
Seven expression classes
PHOG features
Preprocessing (using z-score normalization)
Bidirectional Neural Network
Classification
Seven expression classes

Fig. 1: Structure of general procedure of using BDNN for facial expression classification based on descriptors LPQ and PHOG

In our second attempt, we construct a classic 2-D convolutional neural network which takes the centered face images detected in the SFEW dataset as input and output the predicted expression class labels.

The convolutional neural network (CNN) is a class of deep learning neural networks and it is able to extract successively larger features in a hierarchical set of layers. CNN is most commonly used to analyze visual imagery and is frequently working behind the scenes in image classification[9].

We build up a frontal face detector to recognize faces in the SFEW image dataset and extract the centered faces from it. Those extracted faces are cropped and resized with the same shape so that they could be used as the input to the classic CNN network. The network is trained by training images and outputs predicted expression class labels. Five-fold cross validation is used to effectively avoid the overfitting and underfitting and evaluate estimator performance.

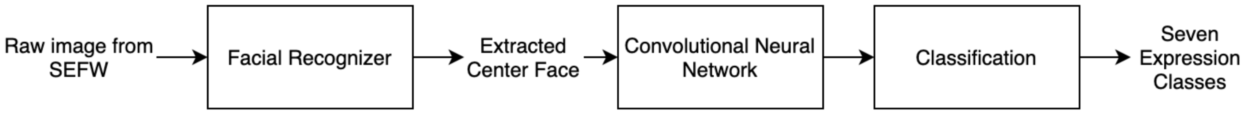


Fig. 2: Structure of general procedure of using CNN for facial expression classification based on SFEW database

We compare the classification performance between the basic non-bidirectional neural network model and the bidirectional model to reveal the contributes of bidirectional transmissions. Furthermore, we also compare the BDNN model's performance with SVM classification model's performance that published in the dataset research paper and make conclusions about the general ability of BDNN in facial expression classification tasks. Finally, we analyzed the CNN classification performance and compare its result with the former approaches.

2 Methodology

2.1 BDNN Approach

We implement a bidirectional neural network with one hidden layer to accomplish the facial expression classification work.

In bidirectional neural networks, learning is performed in both directions. It is trained as taking one modality as an input and the other modality as the expected output, while at the same time the second one is presented as input and the first one as expected output. This process is equivalent to using two separate neural networks that have sharing specific weight variables and are symmetrical in the structure and shape. By repeatedly alternating the training process of both forward neural network and the reversed directional network and keep sharing the weights, learning in both directions can be achieved.

Variables representing the weights are shared across the two networks and are in fact the same variables. The weight matrix W trained from the first neural network, which is also recognized as the forward directional network, can be shared to the second neural network, which is recognized as the reversed directional network, by flipping the matrix W over its diagonal and switching the row and column indices, and get the transposed matrix as the weight matrix W'.

Figure3.(a) illustrates the structure of the forward direction neural network. There are five neurons in the input layer that are able to store five principle components of the descriptors in the dataset. And the hidden layer contains five hundred of neurons so that it can learn features more precisely. The output layer has seven neurons which are corresponding to the seven expression classes. I use $ReLu$ as the activation function between input layer and hidden layer and use $SoftMax$ to determine the predicted expression class. Weights and bias are updated in the neural network by back propagation and the loss is defined as cross entropy loss. Each epoch of training in the forward neural network can represent the training process from right to left in the Bidirectional Neural Network.

Figure3.(b) illustrates the structure of the reversed direction neuron network. It is symmetrical in the shape to the forward direction neuron network that has seven neurons in the input layer, five hundred neurons in the hidden layer and five neurons in the output layer. The reversed direction neural network takes the output neuron values from the forward network as the input and its training process can be regarded as training from right to left in the Bidirectional Neuron Network.

There is no activation functions set between layers so that the weights and bias variables can be better stored and transformed between the forward direction neuron network and the reversed direction neural network. The weight and bias variables are firmly related to the forward direction neural network. As the figure(b) illustrated, the weight matrix between input layer and hidden layer are set as the transpose of the weight matrix between hidden layers and output layers in the forward direction neural network. Similarly, the weight matrix between hidden layer and output layer is set as the transpose of the weight matrix between input layer and hidden layer in the forward neural network. We didn't share the bias between hidden layer and output layers in both networks but the bias between input layer and hidden layer are transferred between those two networks.

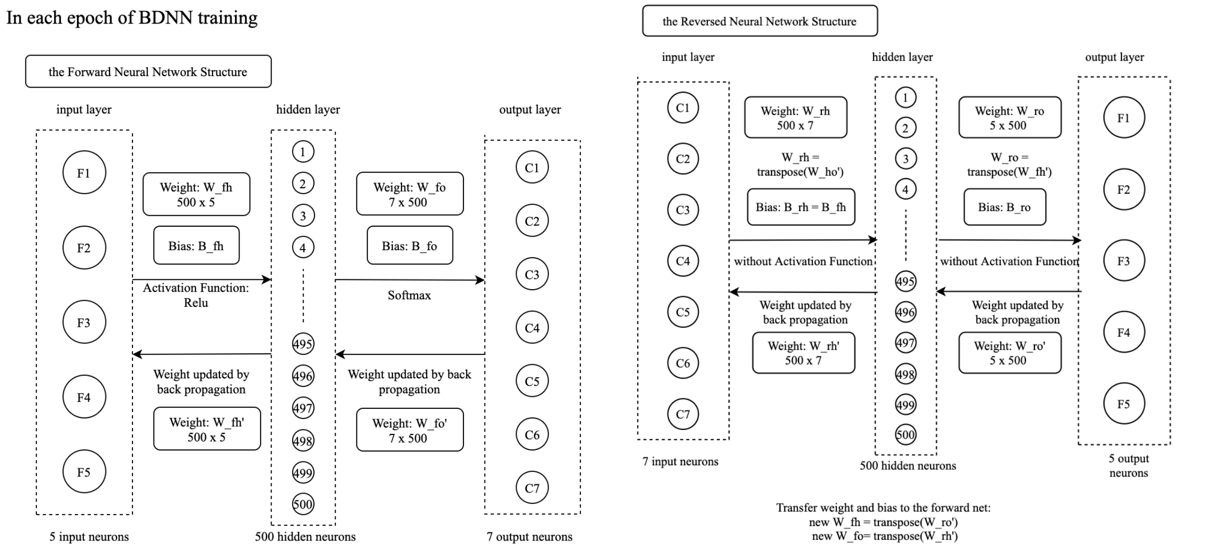
The reversed direction neural network takes mean square error and also updates its weight and bias variable by back propagation. After training, the weight matrix values are transferred back to the forward direction neural network in the form of transpose.

2.2 Deep Learning Approach

We construct a classic convolutional neural network containing two 2-D convolutional layers, a max pooling layer and two fully connected layers for the classification task.

The input data to the CNN is the preprocessed centered face images extracted from the SFEW dataset. We use a pre-trained classifier cascade to do the front face detection and extraction work. The detected face are converted to grayscale format and then are cropped and resized with the same shape, which is 32×32 in this case, so that they could be used as the input to the CNN network. The facial detection and extraction process is illustrated in Figure4.

We generate a total of 367 centered face image attached with its corresponding expression class labels, which includes 49 Angry Face, 41 Disgust Face, 46 Fear Face, 60 Happy Face, 61 Neutral Face, 54 Sad Face and 56 Surprise Face. In order to match the output labels in the neural networks, we denote $(0, 1, 2, 3, 4, 5, 6)$ to represent the expression classes $angry, disgust, fear, happy, neutral, sadandsurprise$ respectively. As the dataset is relatively small, we only randomly use 5 images from each expression class to be the test images and based on the total amount of images in each class, generally the ratio of test set to training set is 1:9.



(a) Structure of the forward direction neural network

(b) Structure of the reversed direction neural network.

Fig. 3: The BDNN model structure(represented by two symmetric networks)

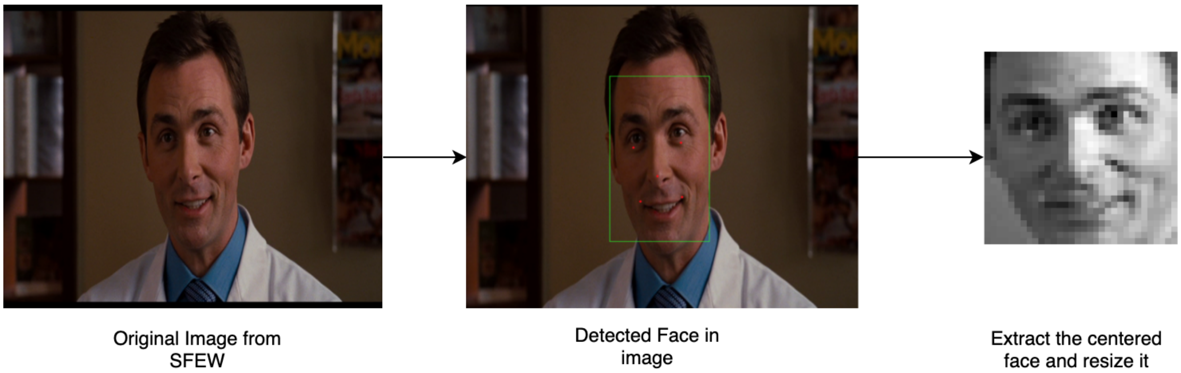


Fig. 4: The center face detection and extraction process

The convolution neural network's structure is illustrated in Figure5 and 6. The first convolutional layer contains input as an image (1 channel, i.e. grayscale map), output as 6 feature maps, kernal as 5×5 square. After max pooling with stride 2 and kernal size 2, the second convolutional layer contains input as 6 feature maps, output as 16 feature maps and a kernal as 5×5 quare. Three fully connected layer are following the convolution parts, and the final layer contains seven neurons which corresponds to the expression class labels. The fully connected layers are linearly connected and activated by ReLu function. The predicted label is determined by SoftMax function.

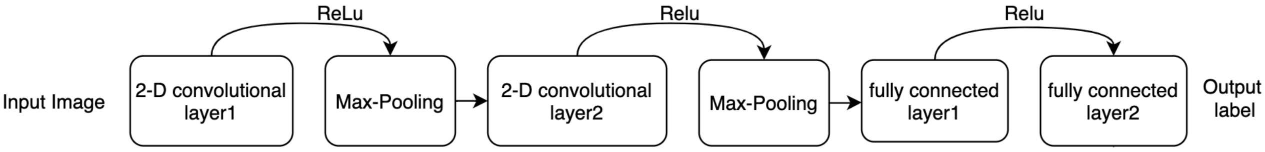


Fig. 5: Convolutional Neural Network Structure used in classification

3 Results and Discussion

The table briefly summarized our experiments result in a straightforward way.

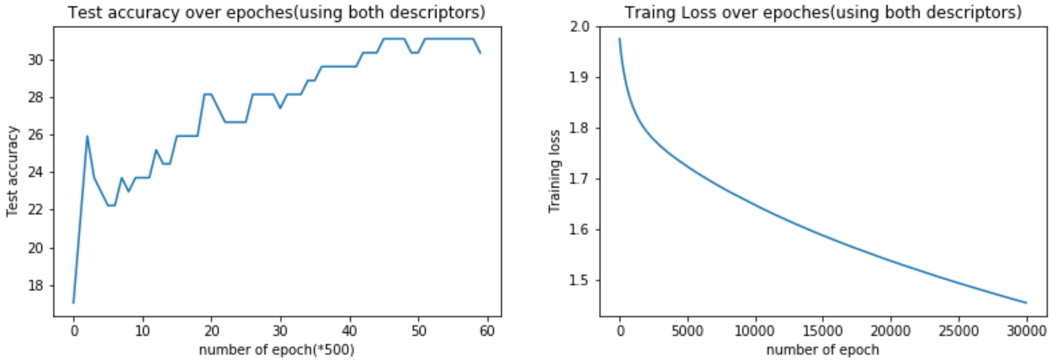
Table 1: Facial Expression Classification Performance Among Different Approaches.

Approach	Training Loss	Test Accuracy
Vanilla Neural Network(PHOG,LPQ)	1.940	15.37%
BDNN(PHOG,LPQ)	1.476	31.11%
BDNN(PHOG)	1.941	21.48%
BDNN(LPQ)	1.701	17.04%
SVM(LPQ)	-	43.71%
SVM(PHOG)	-	46.28%
CNN	0.25	29.10%

In terms of the BDNN approach, we did two sets of classification tasks. The first experiment is training a Bidirectional neural network using both descriptors LPQ and PHOG. And compare its classification performance with the non-bidirectional neural network model. The input to the Bidirectional Neural Network is 10 features including 5 principle components of LPQ and 5 principle performance of PHOG.

The figure 6 show the average testing accuracy and training loss over epochs using Bidirectional Neural Network for classification (using both descriptors LPQ and PHOG). The testing accuracy reached around 31.11% and the training loss is 1.476 after 30000 training epochs.

We define the non-bidirectional neural network to be the vanilla neural network having the same structure as the forward direction neural network. When doing the same classification task in the non-bidirectional neural network, we finally get the testing accuracy to be around 15.37% and the training loss is 1.940 after 30000 training epochs.



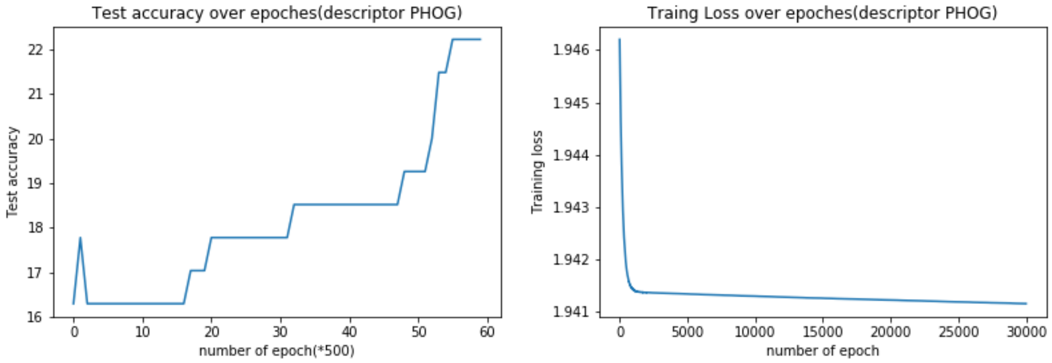
(a) Test Accuracy

(b) Training Loss

Fig. 6: The model test accuracy and training loss over epochs using Bidirectional Neural Network for classification (using both descriptors LPQ and PHOG)

We also trained Bidirectional neural network using one descriptor(LPQ and PHOG respectively). And compare their classification performance with the support vector machine model using C-SVC, with a radial basis function (RBF) kernel [2].

The figure 5 shows the classification performance of Bidirectional Neural Network for classification using descriptor PHOG. It indicates that the test accuracy reaches around 21.48% and the training loss get 1.941 after training.



(a) Test Accuracy

(b) Training Loss

Fig. 7: The model test accuracy and training loss over epochs using Bidirectional Neural Network for classification (using descriptor PHOG)

As for the classification performance of Bidirectional Neural Network for classification using descriptor LPQ. Experiment results indicates that the test accuracy reaches around 17.04% and the training loss get 1.701 after training. In the classification task using support vector machine model using C-SVC, with a radial basis function (RBF) kernel , the test accuracy is 43.71% for using the descriptor LPQ and 46.28% for PHOG[2].In terms of the deep learning approach using CNN, Figure 8 illustrate the average training loss over epochs in the classification process and the average test accuracy reaches 29.10%.

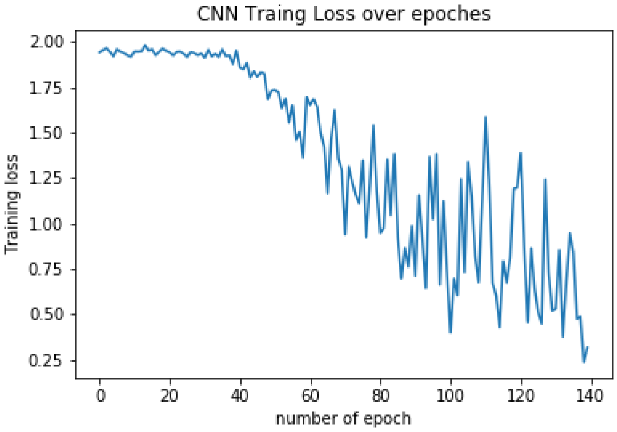


Fig. 8: Convolutional Neural Network Average Training Loss over epochs

Comparing the performance of the Bidirectional Neural Network model and the non-Bidirectional Neural Network model, it's clear that the bidirectional training can increase the classification accuracy which indicates the bidirectional transmission's effective role in neural network model training. Besides, The classification performance using single descriptor is not as good as the training using both descriptors. Compared with traditional approaches like Support Vector Machine, the BDNN Classifier is still unstable and has lower performance. It has shortcomings including the high computational cost, lower training speed, complicated structure and so on. But its enhancement in the training accuracy, especially compared to the non-bidirectional neural network can not be neglected. It's a beneficial and inspiring perspective to consider the bidirectional training as an active action into practice since the transmission not only exists in the basic neural network but also many more complex structures like convolutional neural networks, autoencoders and so on. Also, adding more interconnected layers and hidden neuron to the bidirectional neural network could definitely contribute to a higher robustness.

The convolutional neural network classifier reaches the test accuracy to be around 30%, which is not very high and didn't meet the expectations that a CNN could perform ideally in image classification tasks. We think the main reason is that the input image dataset size is still too small for training a robust convolutional neural network. The facial detector we use is not very effective that it only extracts around 55% of the facial images from the SFEW database. The total number of training image is only 332 which is very small for building up a trustful convolutional neural network model. Therefore, if a more effective facial extraction or construction can be taken for generating the training data, the performance of this CNN model will be improved a lot. Moreover, a more complex and stable CNN network structure needs to be investigated to achieve a better classification results.

4 Conclusion and Future Work

According to the experiment results we get, the bidirectional transmission's effective role in neural network model training is proved as it unable the neural network model achieves better performance for the facial expression classification task. However, the BDNN model still has many shortcomings like it doubles the size and the complexity of a neural network and takes higher computational cost in training. There is still a lot of space in enhancing performance of the BDNN model and use it to perform classification, pattern recognition or prediction tasks, for example, adding more interconnected layers, adding more hidden neurons or varying activation functions between layers. Also, a more effective way for transferring weights and parameter variables between the forward direction neural network and the reversed direction neural network is worth investigating as it can improve the efficiency and training speed of BDNN model. Bidirection is an inspiring perspective we can take when designing models and the effective bidirectional transmission will contributes to a more powerful structure.

The convolutional neural network classifier takes the centered face images as input and is trained to learn the successively features in a hierarchical set of layers. However, in our experiments, the test accuracy we achieved is not very high and it is mainly because the size of the input centered face image dataset is still too small for training a robust convolutional neural network. A more effective and accurate way of capturing and extracting face in images should be taken to generate a better dataset for training the network. Furthermore, more advanced deep learning methods like Resnet, transfer learning and so on are worth investigation in the facial expression classification work.

References

1. F. Siraj, A. Ab Aziz, M. S. Sainin, and Mohd Hafiz Mohd Hassin, "The Design of Emotion Detection System to regulate Human Agent Interaction," Proceedings of the Second International Conference on Artificial Intelligence and Engineering Technology. Kota Kinabalu, Sabah. 3-5 August. pp. 1-7, 2004.
2. Dhall, Abhinav, Roland Goecke, Simon Lucey, and Tom Gedeon. "Static facial expression analysis in tough conditions: Data, evaluation protocol and benchmark." In 2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops), pp. 2106-2112. IEEE, 2011.
3. Kanade, Takeo, Jeffrey F. Cohn, and Yingli Tian. "Comprehensive database for facial expression analysis." In Proceedings Fourth IEEE International Conference on Automatic Face and Gesture Recognition (Cat. No. PR00580), pp. 46-53. IEEE, 2000.
4. J. D. Velazquez. (1997). Modeling Emotions and Other Motivations in Synthetic Agents. In Proceedings of the Fourteenth National Conference on Artificial Intelligence, 10-16. Menlo Park, Calif.: American Association for Artificial Intelligence, 1997.
5. Ojansivu, Ville, and Janne Heikkilä. "Blur insensitive texture classification using local phase quantization." In International conference on image and signal processing, pp. 236-243. Springer, Berlin, Heidelberg, 2008.
6. A. Saïdani and A. K. Echi, "Pyramid histogram of oriented gradient for machine-printed/handwritten and Arabic/Latin word discrimination," 2014 6th International Conference of Soft Computing and Pattern Recognition (SoCPaR), Tunis, 2014, pp. 267-272.
7. V. Ojansivu and J. Heikkil. Blur Insensitive Texture Classification Using Local Phase Quantization. In Proceedings of the 3rd International Conference on Image and Signal Processing, ICISP'08, pages 236–243, 2008.
8. Nejad, A. F., and T. D. Gedeon. "Bidirectional neural networks and class prototypes." In Proceedings of ICNN'95-International Conference on Neural Networks, vol. 3, pp. 1322-1327. IEEE, 1995.
9. Lawrence, Steve, C. Lee Giles, Ah Chung Tsoi, and Andrew D. Back. "Face recognition: A convolutional neural-network approach." IEEE transactions on neural networks 8, no. 1 (1997): 98-113.