

Procedia Computer Science 00 (2016) 1–13



Procedia

Computer

Science

Convolutional Neural Network Models for Facial Expression Recognition using BU-3DFE Database

Xuan-Phung Huynh*a* , Tien-Duc Tran*b* , Yong-Guk Kim*a,*∗

*aDept. of Computer Engneering, Sejong University*

*bHCMC University of Technology and Education*

# Abstract

We present a convolutional neural network (CNN) for 2D + 3D feature-based facial expression recognition approach and present its performance using BU-3DFE database. Our network consists of two CNNs: one for the 3D face shape and the other for the face appearance with color in order to achieve efficiency and robustness. The network consists of three convolutional layers including max pooling as well as normalization layers, and two fully connected layers, totaling over 26.5 million parameters and 45504 neurons. A 6-way softmax is used for the outputs on the final layer. Performance evaluation suggests that the facial expression recognition rate reaches to excellent 88.29%.

*Keywords:* Deep learning, convolutional neural network, BU-3DFE, 2D + 3D feature-based facial expression recognition

# Introduction

Facial Expression Recognition (FER) has fascinated due to its potential applications, such as human-computer interaction, multimedia, surveillance, security, medicine, behavior science, communication, and education. However, FER is a very challenging problem mainly because it needs to assort diverse human expressions among different subjects in different cultures, genders and contexts [1]. In the past decades, varieties of FER approaches have been proposed [2, 3, 4, 1]. Most existing technologies for FER utilize off-the-shelf feature extraction methods for recog- nition and focus on 2D images [3, 4].Some successful approaches have been applied on 3D databases [2]. And others propose to use a convolutional neural network in recognizing facial expressions but they apply on 2D databases [3, 4, 5, 6].

In recent years, deep neural networks have become the state-of-the-art algorithms in the pattern recognition and machine learning areas. Convolutional Neural Networks (CNNs) are an alternative type of neural network that can be used to classify the objects in a scene [7, 8, 9]. Therefore, the quality of image recognition and object detec- tion had been progressing with a dramatic pace [9] by utilizing CNNs. Not only the result of powerful hardware, larger databases and bigger models, but also a consequence of new ideas, algorithms and improved network architec- tures has encouraged in this field. Our network combines two CNNs with two different inputs to elevate the system performance: one is a 3D facial shape model and the other a frontal-view appearance model.

CNN becomes more successful with LeNet [8], AlexNet [7], and GoogleNet [9]. In addition, high performance GPUs, paired with a highly optimized implementation of 2D convolution, facilitate the training of interestingly large CNNs. Moreover, many libraries, such as Theano, Caffe, and Torch, have supported for training as well as testing on GPUs. In our experiment, we utilize Theano to train our network with a standard facial expression BU-3DEF database.

*Email address:* ∗[ykim@sejong.ac.kr](mailto:ykim@sejong.ac.kr) (Yong-Guk Kim*a,*∗)

/ *Procedia Computer Science 00 (2016) 1–13* 2

* 1. *Related work*

Similar to 2D, 3D FER can be roughly divided into two categories: template-based and feature-based [10, 11]. For template-based approaches, deformable face models are utilized to extract the model parameters as expression features for recognition. For example, the various expressions of an individual using a face scanner that constructs textured 3D meshes as the morphable 3D model is proposed [12]. In addition, bilinear deformabel [13], shape deformation model [2], and statistical feature model [14] are also successful with template-based techniques. Nonetheless, establishing one-to-one correspondence between 3D face scans is the main disadvantage of template-based approaches. Not only time consuming procedures like dense 3D face registration but also model fitting are very challenging issues [11]. On another hand, feature-based approaches utilize variety of facial surface geometric or differential quantities to extract 3D expression cues. Facial landmarks are the most efficient features to distinguish between expressions or action units. For instance, the distances between 3D facial landmarks are proposed in [5, 6, 15]. Furthermore, batch-based features [16], 3D facial curves [17], deep representation [18], and 2D representations, such as local depth-SIFT features [19], expressive maps [20], facial conformal images [21], facial surface normal [22], curvatures [16]. Especially, the effectiveness of 2D+3D multimodal FER has been well presented as in [11]. Nevertheless, 3D facial landmark localization is still a very difficult task so that feature-based approaches are not accurate as well as robust [11]. Even so, these approaches generally accomplish better than template-based ones.

* 1. *Contributions*

In this paper, we propose to utilize multi-convolutional layers that can learn features automatically based on 2D image as well as 3D model of the face. The inputs include a color frontal face image and a gray depth image, which is rendered from 3D model. We propose a special structure to combine two inputs with two deep sub-network, which results in significant improvements when comparing with single one. Our network utilizes only the image, which is more simple than 3D models. In addition, our approach represents the facial features automatically through out deep learning process.

In contrast with majority of deep learning algorithms, which require the big data, our network performs on a small database, BU-3DFE. To avoid over-fitting problems, pre-trained models are utilized [18]. Even though we train our network without using pre-trained model, our model achieves the best result on BU-3DFE with common protocols that prevent over-fitting on this database. We propose a data augmentation on training as well as testing by cropping randomly. In addition, early stopping is utilized to reduce over-fitting.

The rest of the paper is organized as follow. In section 2 and 3, we review the BU-3DFE database and Convo- lutional Neural Network, respectively. In next section, we describe our network in detail. In section 5, we present experimental results. Finally, Section 6 summarizes this paper.

# 3D Facial Expression Database

* 1. *BU-3DFE*

BU-3DFE is a database of both prototypical 3D facial expression shape and 2D facial texture, consisting of 2500 models from 100 subjects with different racial ancestries and genders [23]. This database was designed to sample facial behaviors with seven universal emotional states: neutral (NE), happiness (HA), surprise (SU), fear (FE), sadness (SA), disgust (DI), and angry (AN). Four intensities, which reflect different levels of spontaneity, represent each expression. Based on the average of expert recognition rate, it was reported that it was 94.1% for the low intensity, 95.7% for the middle intensity, 96.8% for the high intensity, and 98.1% for the highest intensity, respectively [23]. Most researchers used the high and highest intensity when their models were evaluated.

We truncate the boundary to generate a face model, using only the face region. The cropped face region includes 13,000 - 21,000 polygons [23]. The database has included frontal view textures (512 by 512 pixels). Based on facial shape model, we generate depth image, i.e. 3D shape model, by utilizing Phong shading interpolation. Fig. 2 depicts the preprocessed database for 3D shape model, which is one of the inputs (gray scale image) of our network. The transformation of color image to gray scale image can be made by using equation [24]

*Iy* = 0*.*333*Fr* + 0*.*5*Fg* + 0*.*1666*Fb* (1)

/ *Procedia Computer Science 00 (2016) 1–13* 3

where *Fr* , *Fg*, and *Fb* are the intensity of R, G, B component respectively. *Iy* is the intensity of equivalent gray level image.

In addition, Fig. 1 shows the examples from two subjects with frontal view textures and depth faces. In our experiment, we keep the samples up six prototypical facial expressions (angry, disgust, fear, happiness, sadness, and surprise) of two higher intensity levels.



Figure 1: Two sample subjects showing six expressions (angry, disgust, fear, happiness, sadness, and surprise). The facial shape models (the 2nd and 4th row) and frontal-view textures (the 1st and 3rd row) are produced.

* 1. *Construction of depth image*

In this paper, we utilize Phong shading model [25] based on OpenGL library to render depth image represen- tation. BU-3DFE database has provided the 3-D mesh representation in type of WRLM v2.0 format [26]. It uses pre-computed and indexed local information about the 3-D surface [27]. Each 3-D polygonal mesh is presented as a collection of mesh elements: vertices (points), edges (connectors between vertices) and polygons (shapes formed by edges and vertices) [27]. Fig. 2a shows 3-D mesh representation for an exemplary face belonging to BU-3DFE database. To construct 3-D models accurately, the 2-D texture coordinates are also embedded in the vertex informa- tion. However, [23] provided the 2D frontal images in BU-3DFE, we only consider the construction of 3D shape to present depth images.

Utilizing OpenGL library, we have reproduced illusions of depth using lighting and shading. Shape From Shading (SFS) captures the shape recovery problem form a gradual variation of shading in the image [27]. In 1975, Phong proposed the new technique that resolved SFS problem requires the computation of the normal to the displayed surface at each point [25]. Considering a point along the edge of a polygonal model, the surface normal is the outcome of a linear interpolation to the normals at two vertices of that edge. Thus, the normal *Nt* to the surface a point between two vertices *P*0 and *P*1 is [25]:

*Nt* = *tN*1 + (1 − *t*)*N*0 (2)

where *t* = 0 at *N*0 and *t* = 1 at *N*1; otherwise *t* ∈ (0*,* 1).

Expanding to surface of a polygon, the normal at a point is determined in the same way as the computation of

the shading at that point with the Gouraund technique [25]. Based on the normal vector, the Phong reflectance model comprises a diffuse term, and ambient term, and a specular term:

*L*(*p*¯*,*→−*d* ) = *r I max*(0*,*→−*s* · →−*n* ) + *r I* + *r I max*(0*,*→−*m* · →−*d* )*α* (3)

*e d d a a s s e*

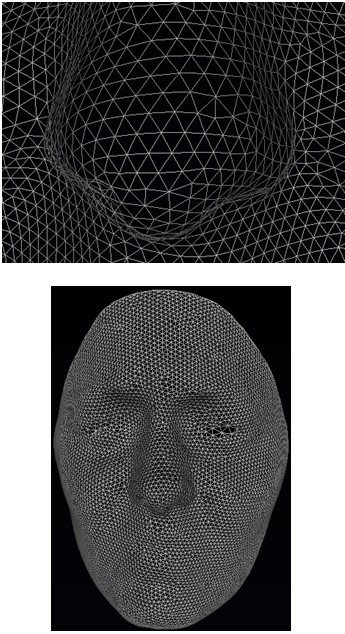
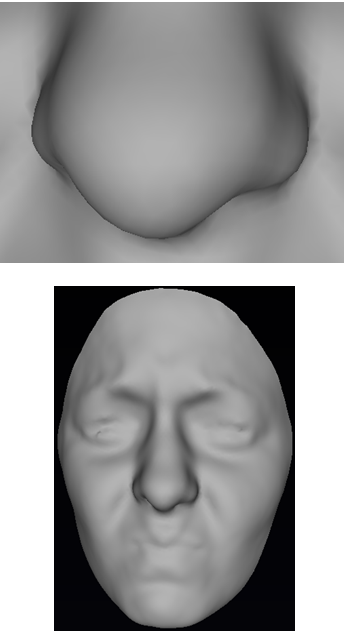
where

* + - *Ia*, *Id* , and *Ir* are parameters that correspond to the power of the light sources for the ambient, diffuse, and specular terms;

/ *Procedia Computer Science 00 (2016) 1–13* 4

* + - *ra*, *rd* , and *rs* are scalar constants, which are reflection coefficients that determine the relative magnitudes of the three reflection terms;
    - *α* determines the spread of the specular highlights;
    - →−*n* is the surface normal at *p*¯;
    - →−*s* is the direction of the distant point source;
    - →−*m* is the perfect mirror direction, given →−*n* and →−*s* ;
    - →−*de*

is the emittant direction of interest.

(a). 3-D mesh (b). Depth image from SFS.

Figure 2: 3-D face representations and construction of depth image

Fig. 2b demonstrates the result of SFS by using Phong shading interpolation with OpenGL library.

# Convolutional Neural Network

* 1. *Convolutional Layer*

The layer performs a convolution filter over the input. This layer computes the output as follows:

*k l*

*C*(*xu,v*) = � � *w*(*s, t*)*xu s,v t* (4)

− −

*s*=−*k t*=−*l*

where *w* is the filter of size *mxn*; *x* is input; *k* = (*m* − 1)*/*2, *l* = (*n* − 1)*/*2. In general, any convolution filter acts very much like an edge detector. Convolutional layers in a CNN play a major role in extracting features from the inputs [28, 29].

/ *Procedia Computer Science 00 (2016) 1–13* 5

* 1. *Max Pooling*

Max Pooling performs 2D max-pooling over the two trailing axes of input x. The output is a set of non-overlapping rectangles, and, for each such sub-region, output is the maximum value. Max-pooling is useful in vision due to:

* + - To reduce computation for upper layers, it eliminates any non-maximal values.
    - It features translational invariance with esteem to the filter size.
  1. *Rectified Linear Unit*

A Rectified Linear Unit (ReLU) has a non-linear activation function for artificial neurons as follows:

*R*(*x*) = *max*(0*, x*) (5)

Using ReLUs prevents overfitting and increases learning speed [30, 7].

* 1. *Local Response Normalization*

Local Response Normalization aids generalization by the expression [7]:

*min*(*N*−1*,i*+ *n* )

2

*bi* *i*

� *j* 2 *β*

*x,y* = *ax,y /* (*k* + *α*

*j*=*max*(0*,i*− *n* )

2

(*ax,y*) )

(6)

where *ai*

*x,y*

is the activity of neuron that is computed by applying kernel i at position (*x, y*) and then applying the ReLU

nonlinearity; *bi*

*x,y*

is the response-normalized activity. The sum runs over *n* ”adjacent” kernel maps at the same spatial

position, and *N* is the total number of kernels in the layer. The constants *k*, *n*, *α*, and *β* are hyper-parameters that determined in a validation set; we use *k* = 2, *n* = 5, *α* = 1*e* − 4, and *β* = 0*.*75. Because we do not subtract the mean, this layer is more correctly termed ”brightness normalization” [7].

* 1. *Fully Connected Layer*

Fully Connected Layer or Multilayer Perceptron connects all neurons of lower layer to every neuron of its layer as follows:

*F*(*x*) = *σ*(*W* ∗ *x* + *b*) (7)

where *σ* is activation function, *W* and *b* are the weights of layer. In our network, *σ* is a ReLU.

* 1. *Dropout*

Dropout is powerful method of regularizing although computation is inexpensive [30, 7, 31]. Let be the vector of outputs from layer l in a standard neural network. With dropout, the vector of outputs becomes:

*y*˜(*l*) = *r*(*l*) ∗ *y*(*l*) (8)

where *r*(*l*) ∼ *Berboulli*(*p*) with *p* is the rate of dropout.

* 1. *Output Layer*

The output layer is a one-hot vector representing the likelihood of the class given input. The resulting class for the output vector is the maximum likelihood value. In practice, we compute the cross-entropy between an approximating distribution and a true distribution. This is loss function, and the minimize of it is the maximum likelihood value of output. Actually, the cross entropy measures the average number of bits that required to identify an event from a set of possibility.This function computes:

*C* = *H*(*p, q*) = − � *p*(*x*) *log*(*q*(*x*)) (9)

*x*

where *p* is the distribution of the ground truth and *q* is approximating distribution.

/ *Procedia Computer Science 00 (2016) 1–13* 6

* 1. *Softmax Layer*

The Softmax Layer propagates back over the error. Let *K* is the total number of neurons in this layer. For each neuron *j* in [1*, K*], the output is

*exj*

*S* (*x*) *j* =

*K*

*i*=1 *exi*

(10)

# Our network

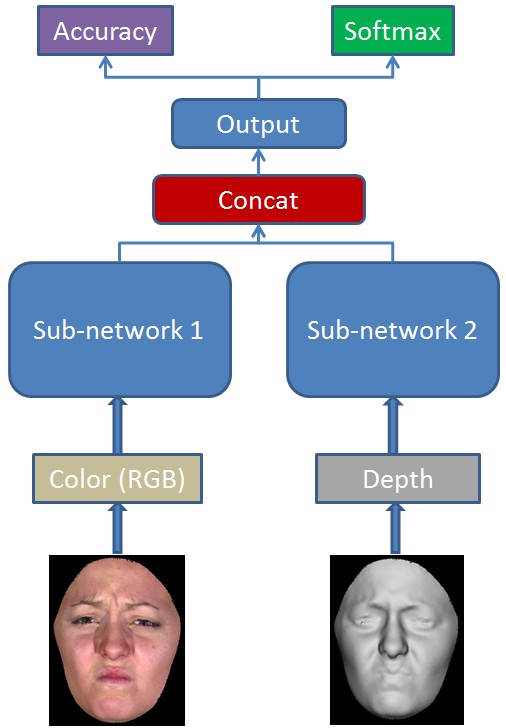
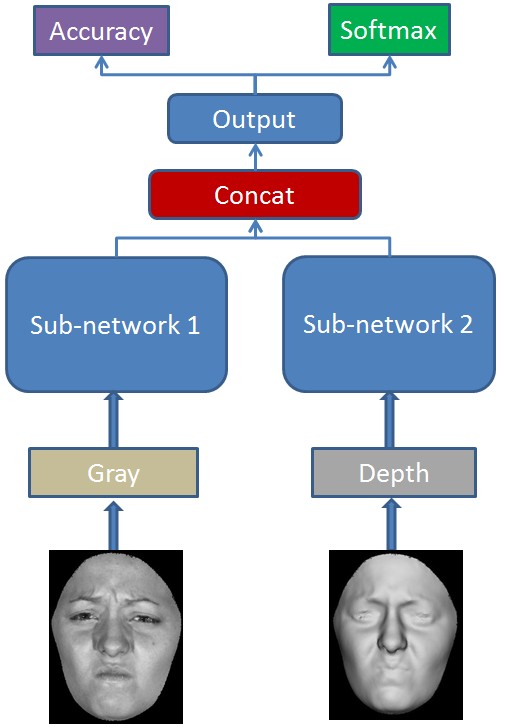
* 1. *Architecture*

Fig. 5 illustrates the high-level architecture of our network while Fig. 4 shows its detail structure. It contains two sub-networks and each sub-network consists of three convolutional layers and two fully-connected layers. Each sub-network has five convolutional and two fully-connected. When the depth face is the input in sub-network 2, sub-network 1 handles the frontal view texture with color (RGB) image as Fig. 5a and gray scale image as Fig. 5b, respectively. The difference between color image and gray scale image is the number channels of the convolution filter in the first convolutional layer. Color image utilizes three channels, and gray scale image uses only one channel.

All the convolutions, including those inside the inception modules, utilize rectified linear activation. The size of the receptive field in our network is 448 x 448 taking RGB color channels (frontal-view) and gray scale image (facial shape) with normalization between 0.0 and 1.0. *Tanh* units normally take several times than Rectified Linear Units (ReLUs) in training the given network [7]. Since our network is so large that faster learning has a great influence on the performance. Each convolutional layer contains a Max Pooling and a Local Response Normalization. These layers prevent the overfitting problem with complex network. We use normalization after applying the pooling in a certain convolutional layer.

We utilize two fully-connected layers in each sub-network. To improve the generalization error, a Dropout involves each fully-connected layer with a rate of 0.5. We combine the last layer in each sub-network by utilizing maximum element-wise operator. Then output layer is built on the top of this layer with six units that corresponds six facial expressions. The error is propagated back over a Softmax layer.

The first convolutional layer filters the 448 x 448 x 3 input images with 16 kernels of size 11 x 11 x 3 (if gray scale image is input, the sizes are 448 x 448 x 1 and 11 x 11 x 1, respectively). The second convolutional layer takes input as the output (pooled and response-normalized) of the first convolutional layer and filters it with 128 kernels of size 7 x 7 x 16. Similarly, the third convolutional layer has 512 kernels of size 5 x 5 x 128. The fully-connected layers have 2048 and 1024 neurons respectively.

(a) Input is a color image (RGB) (b) Input is a gray scale image

Figure 3: Our network model with different inputs.

/ *Procedia Computer Science 00 (2016) 1–13* 7

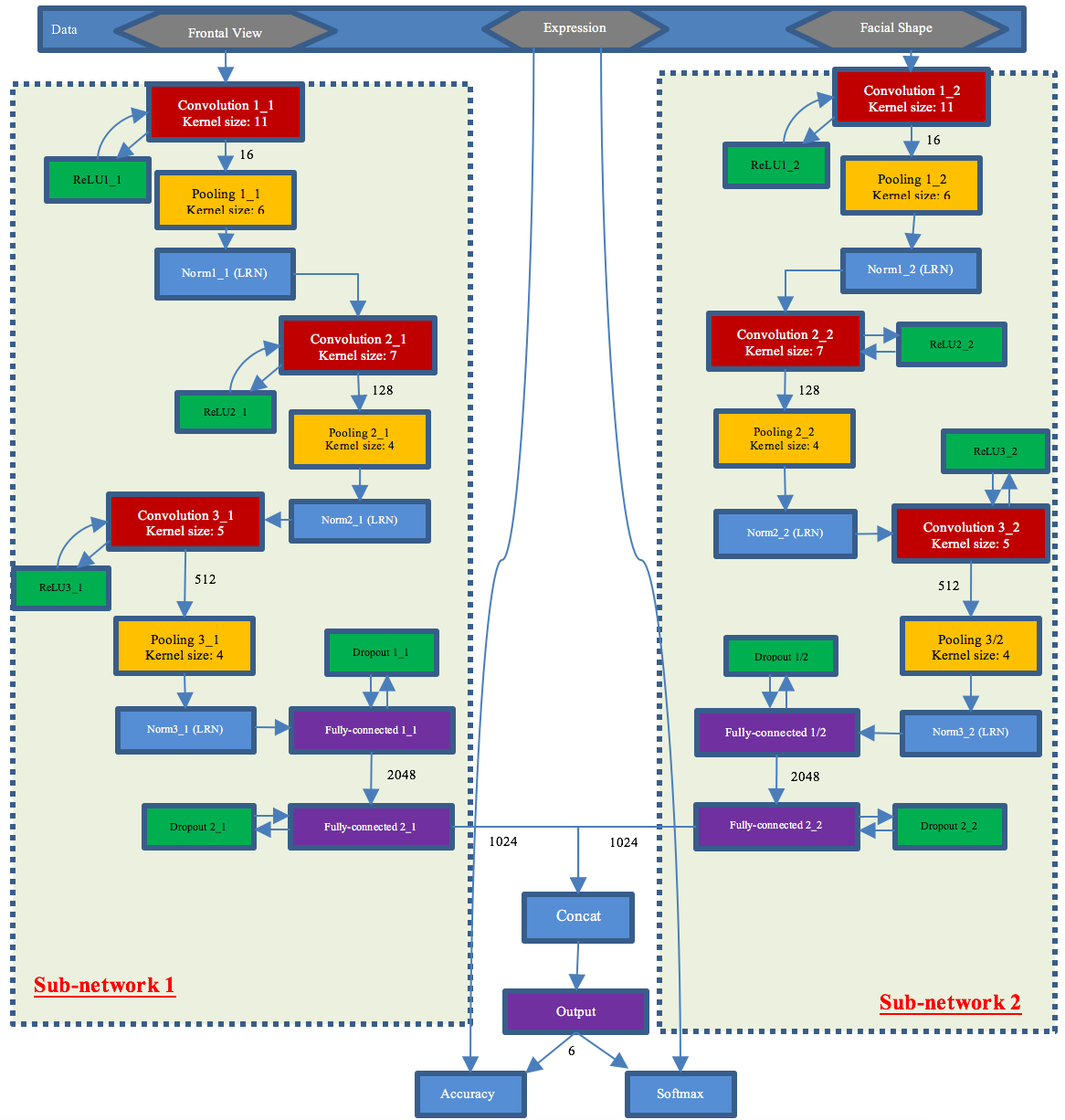


Figure 4: Our network structure in detail. We combine two sub-networks that have the same structure.

/ *Procedia Computer Science 00 (2016) 1–13* 8

* 1. *Reducing Overfitting*

Our network architecture has more than 26 million parameters. In addition, we have only 60 subjects (with 720 samples) for training and testing, and this turns out to be insufficient to train so many parameters without considerable overfitting. We utilize three ways to combat overfitting.

First of all, data augmentation is the easiest and most common method since image data is artificially enlarged us- ing label-preserving transformations [7]. We employ simple form of data augmentation, which consists of generating image translations and horizontal reflections. We extract random 448 x 448 patches from 512 x 512 images and flip a half of patches randomly by their horizontal reflections. This not only increases the size of our training set but also facilitate our approach to be invariant with translation. Without this scheme, our network can not apply for subject dependent protocol with BU-3DFE database due to substantial overfitting, which would have forced us to decrease image size and use much smaller network. At test time, our network predicts testing data by extracting sixty 448 x 448 patches (only randomly cropped) and averaging the predictions made by the network’s softmax layer on these patches. Secondly, ”dropout” is also a successful technique to reduce test errors in the network. Dropout neurons contribute to the forward pass and do not participate in back propagation. Due to contribute with many different random subsets of the other neurons, they are forced to learn more robust features. At test time, dropout neuron is deactivated since prediction is deterministic process. Finally, we apply regularization to the weights in a network. Therefore, the loss function (9) is become

*C* = *H*(*p, q*) + *λ* � *w*2 (11)

*w*

where *w* is the parameter in each layer of the network, and *λ* is regularization parameter. This technique is sometimes known as weight decay or L2 regularization. In our experiment, *λ* = 0*.*0004 is the best for empirical training. Although BU-3DFE database is small, these techniques decrease the substantial overfitting dramatically. Nevertheless, they also increase the number of iterations required to converge. Since the library for deep learning, such as Theano, is developed based on GPU computation, the time of training is reduced rapidly.

* 1. *Details of Learning*

We trained our models using Nesterov’s accelerated gradient [32] with a batch size of 81, and momentum of 0.9.

The weight update formulas is

*Vt*+1 = 0*.*9 ∗ *Vt* − *ε*v *f* (*wt* + 0*.*9 ∗ *Vt* )

*wt*+1 = *wt* + *Vt*+1

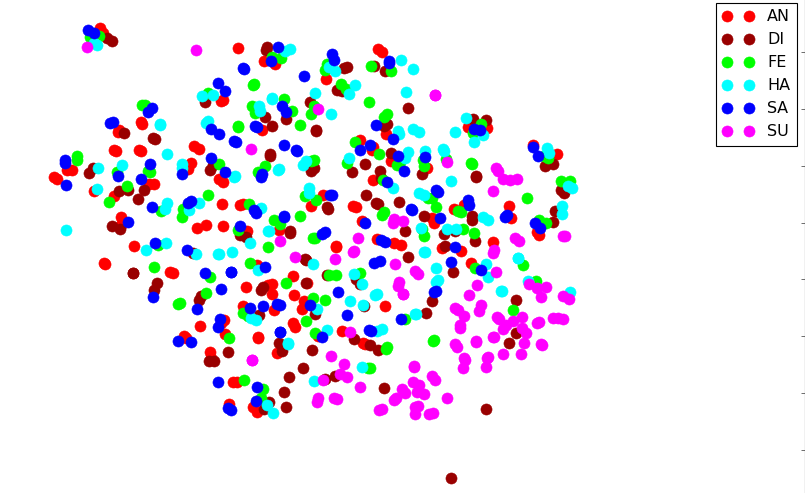
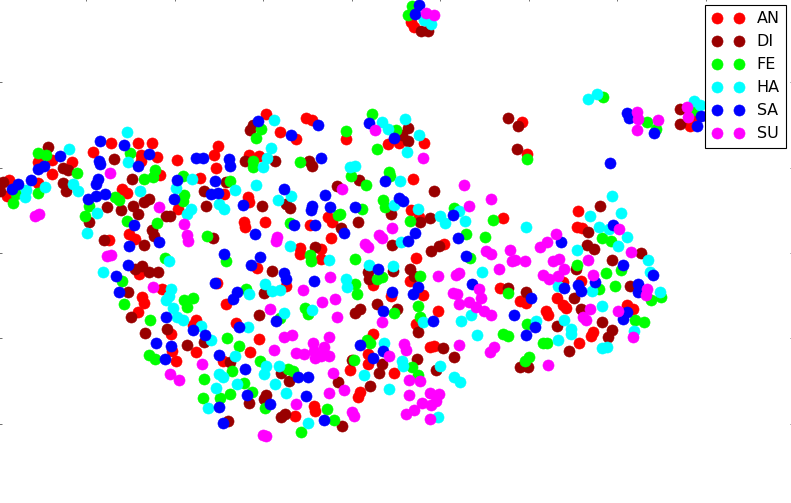
(12)

where *t* is the iteration index, *V* is the momentum variable, *ε* is the learning rate, and v *f* (*.*) is the gradient at *wt* . We utilize an equal learning rate for all layers as well as all epochs. The learning rate, *ε* is 0.025. In addition, we initialize the weights in each layer form a Glorot weight initialization, which is also known as Xavier initialization [33], with gain 1.0, and standard deviation 0.01. On the other hand, the neuron biases are initialized with constant 0.

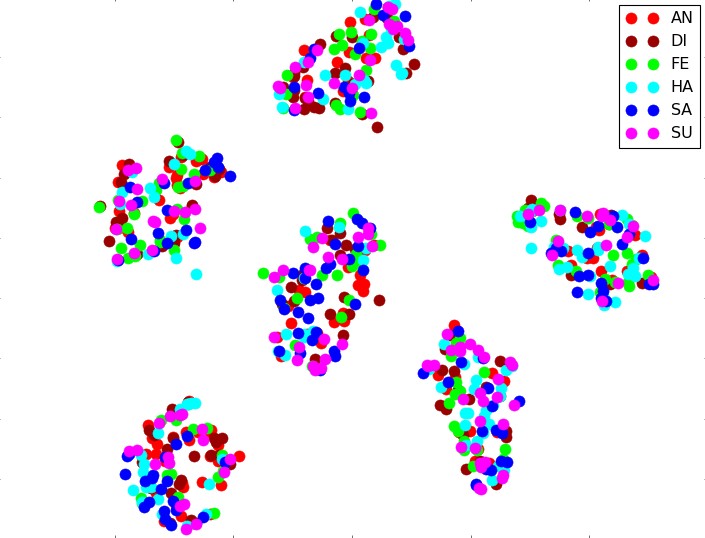
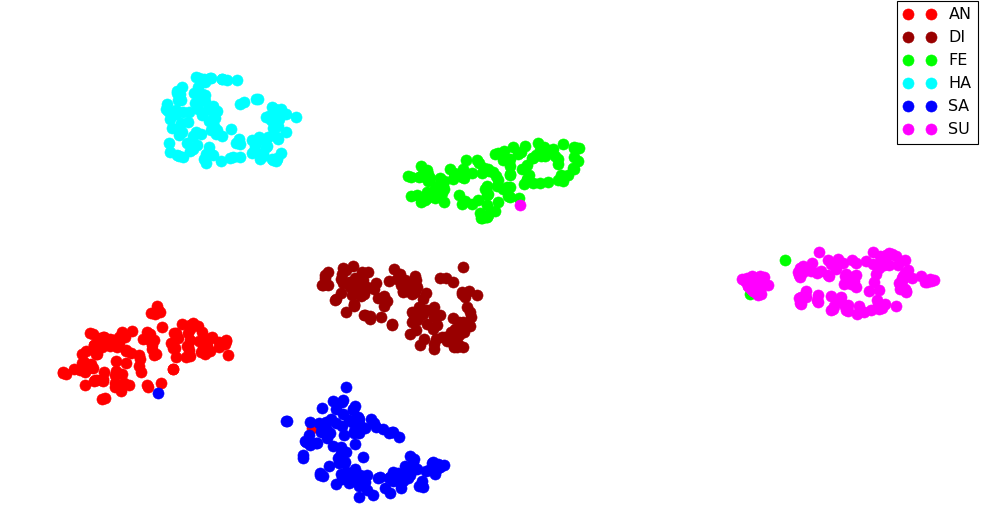
* 1. *Visualization of action units*

The CNNs extract multi-layer deep features from each attribute map. In our approach, we utilized different inputs with two type of networks and combined network. Learning by back-propagation, convolutional filter plays a vital role in feature extraction. These deep representations of filter have intuitive meaning. In order to illustrate the characteristics of the features extracted from the inputs, both color as well as depth image, concat layer, and output layer, we visualized the feature vectors using t-SNE, which is a efficient tool for visualization of high dimensional data [34]. Fig. **??** depicts the spread out of the output data from final layer and concat layer according to their label. Both input data are randomize while two layers have seperated data into six group. In addition, concat layer mixes some labels together. In contrast, output layer seperates six categories by each action unit. The network has learned a variety of action units [35], as well as various salient regions on face. Therefore, applying the softmax operating for this layer achieves the excellent results in our experiments.

/ *Procedia Computer Science 00 (2016) 1–13* 9

(a) Inputs are color images (b) Inputs are depth images

(c) Concat layer (d) Output layer

Figure 5: Visualization of action units. (a,b) Visualization of color inputs and depth inputs. (c) Visualization of concat layer’s output, using t-SNE [34]. (d) Visualization of the outputs in the final layer.

# Experiments

* 1. *Experimental setup*

We utilize the evaluation protocol in [2] to compare fairly with another FER approach. More precisely, we select 720 3D faces of 60 subjects, and keep each subject with 12 expressional samples (2 higher intensities for every kind of expression). Then, we subdivide randomly 60 subjects into two subsets a training set with 54 subjects (648 samples) and test set with 6 subjects (720 samples). To achieve stable recognition accuracy, we conducted 100 rounds of 10- fold subject-independent cross validation. Based on these data partition strategies and the constructed depth images, we use our network with the Convolutional Neural Network for expression classification. For training phase, we randomly cropped frontal images and depth images from 512 x 512 to 448 x 448 in each epoch. While testing phase, the results is averaged for 60 times of randomized crop with each testing sample. In addition, to save the time of training, we cancel process of training after 200 epochs that did not improve the accuracy of testing set. To finish training process with 100 round of 10-fold cross-validation, we spent roughly three months. Our systems are Intel(R) Core i7-5930K CPU, 64 GB of RAM, and NVIDIA Geforce GTX TITAN 12 GB GPU.

* 1. *Frontal image* + *depth image and their fusions*

In this section, we indicate that two type of frontal images (color and gray scale images) and depth images influence the feature extraction of CNN, and thus their fusion largely improves the expression recognition accuracies when combining color and depth image.

To fairly compare these fusions, we conduct 10 rounds of 10-fold cross-validation of each input case. Table 1 list the average expression recognition results of fusing within sub-network 1, sub-network 2, and combined network.

/ *Procedia Computer Science 00 (2016) 1–13* 10

Especially, we test sub-network 1 with two type of frontal images: color image and gray scale images, which are converted from color images. Therefore, we mention the type of input to indicate the network structure. For instance, Table 1a is the results of only sub-network 1 that processes on gray scale images.

Compared with the results in table 1a and 1b as well as 1c and 1d, we can see that fusions of color image are very efficient, with an improvement up to 1.14% and 0.77% respectively. On the other hand, only depth images’s fusion is too small as table 1e, 79.86%.Nonetheless, when combining with sub-network 1, our network improves the fusion significantly. In particular, contributed with gray scale image, table 1c shows an improvement up to 1% when considering table 1a. In the case of color images, the improvement is slightly from table 1b to 1d. More precisely, the confusion matrices of these score indicate that the color information contributes significantly in FER and depth information makes the network more robust.

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| (a) Only Gray Scale Image (87.01%) | | | | | | | (b) Only Color Image (88.15%) | | | | | |
| % | AN | DI | FE | HA | SA | SU | AN | DI | FE | HA | SA | SU |
| AN | 86.75 | 1.75 | 1.50 | 1.67 | 8.33 | 0.00 | 88.33 | 3.25 | 0.67 | 0.92 | 6.83 | 0.00 |
| DI | 3.83 | 86.17 | 3.58 | 3.33 | 1.00 | 2.08 | 4.25 | 88.33 | 2.00 | 1.83 | 1.25 | 2.33 |
| FE | 4.58 | 4.33 | 75.83 | 6.25 | 4.50 | 4.50 | 4.33 | 5.17 | 76.67 | 3.83 | 5.50 | 4.50 |
| HA | 0.58 | 0.00 | 2.83 | 95.58 | 0.17 | 0.83 | 0.67 | 0.25 | 2.50 | 95.75 | 0.00 | 0.83 |
| SA | 10.75 | 1.67 | 3.17 | 1.00 | 83.33 | 0.08 | 10.25 | 2.25 | 3.50 | 0.00 | 83.83 | 0.17 |
| SU | 0.08 | 0.42 | 3.25 | 1.67 | 0.17 | 94.42 | 0.00 | 0.42 | 3.17 | 0.33 | 0.08 | 96.00 |

(c) Only Depth Image (79.86%)

|  |  |  |  |  |  |  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- | --- |
| % | | | | AN | DI | FE | HA | SA | SU |  | | |
|  | | | AN | 82.42 | 3.50 | 2.25 | 1.00 | 10.42 | 0.42 |
| DI | 5.42 | 78.00 | 4.75 | 4.92 | 2.33 | 4.58 |
| FE | 4.92 | 8.25 | 60.17 | 13.92 | 6.00 | 6.75 |
| HA | 0.50 | 0.17 | 5.75 | 92.25 | 0.00 | 1.33 |
| SA | 16.25 | 3.08 | 2.67 | 0.92 | 75.08 | 2.00 |
| SU | 0.08 | 2.08 | 3.58 | 2.50 | 0.42 | 91.33 |
| (d) Gray Scale Image + Depth Image (87.93%) | | | | | | | (e) Color Image + Depth Image (88.70%) | | | | | |
| % | AN | DI | FE | HA | SA | SU | AN | DI | FE | HA | SA | SU |
| AN | 85.42 | 2.00 | 1.92 | 0.83 | 9.75 | 0.08 | 87.25 | 3.08 | 0.75 | 0.83 | 8.08 | 0.00 |
| DI | 4.08 | 88.50 | 2.08 | 2.00 | 1.00 | 2.33 | 4.08 | 88.92 | 2.08 | 1.42 | 1.25 | 2.25 |
| FE | 3.67 | 4.83 | 76.83 | 5.33 | 4.83 | 4.50 | 4.17 | 5.42 | 78.17 | 3.58 | 4.67 | 4.00 |
| HA | 0.67 | 0.33 | 3.00 | 95.17 | 0.00 | 0.83 | 0.58 | 0.25 | 3.17 | 95.17 | 0.00 | 0.83 |
| SA | 9.83 | 1.50 | 3.25 | 0.25 | 85.08 | 0.08 | 7.33 | 1.67 | 4.17 | 0.00 | 86.75 | 0.08 |
| SU | 0.00 | 0.25 | 2.75 | 0.08 | 0.33 | 96.58 | 0.00 | 0.33 | 3.33 | 0.42 | 0.00 | 95.92 |

Table 1: The effectiveness of fusing different inputs on BU-3DFE database.

* 1. *Comparison with di*ff*erent convolutinal layers*

Figure 4 illustrates the detail of our network, which includes three convolutional layers for each sub-network. Convolutional layers are the most critical parts in deep learning to extract the feature. To evaluate the performance of combining this layer, we conduct the same 10-fold validation with different number of layers. Table 2 shows the

|  |  |
| --- | --- |
| Number of layer | Accuracy (%) |
| 1 layer | 60.69 |
| 2 layer | 86.25 |
| 3 layer | 88.33 |
| 4 layer | 85.56 |

Table 2: Different number of convolution layers performs different result. Three layers reach the highest achievement.

/ *Procedia Computer Science 00 (2016) 1–13* 11

experimental results. It is clear to see that the network with three convolutional layer achieves the best accuracy. In addition, the network performance is the worst for only one layer. Even though increasing to four layers, the performance is lower than two layers because the more layers network includes, the more over-fitting it makes. On the other hands, increasing the number of layers takes long time to train. Therefore, we utilize three convolutional layers to achieve the best result in BU-3DFE database.

* 1. *Comparison with other methods*

To validate the effectiveness of our method, we compare it with the state-of-the-art methods on the BU-3DFE dataset. To give a comprehensive analysis, we compare four aspects: data modality, facial landmark, expression classifier, and recognition accuracy [11]. However, in our experiments, we use the raw data of frontal and depth image and do not consider the facial landmark. Actually, this landmark is extracted automatically by Convolutional layers [18].

In the literature, there are three FER protocols on BU-3DFE. The first protocol choses 60 subjects and average the accuracies of one or two rounds of 10-fold cross-validation [16, 5, 6, 15, 36]. This protocol has indicated very sensitive to the identity variations of training and testing sets [2]. Therefore, protocol II, in which 60 subjects are chosen and average the accuracy is average of 100 rounds of 10-fold cross-validation, is proposed [2]. In addition, if 60 subjects is selected randomly for each round of 10-fold cross validation, we have protocol III [19]. The accuracies of the same methods are dropped more than 20% from protocol I to protocol II, while protocol II and protocol III are close to each other [11]. From table 3, our, proposed CNN network reaches the highest average accuracy (88.53%) in protocol II.

In addition, when comparing with another deep learning method on FER, our network not only reaches the highest accuracy but also approaches simple model without any external resources. In 2D aspect, receptive fields are proposed to construct group-wise sub-networks for higher-level representations [3, 4]. In spite of convolutional layers to extract features, a set of local appearance variations produced by specific Action Units is proposed. This approach is good for the case of occlusion when the candidate region is hided. Therefore, this is inappropriate for BU-3DFE. On the other hand, some CNNs were proposed with this database [2, 37]. In this aspect, our network reach state-of-the-art of accuracy. Furthermore, we

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Method | Modality | Landmark | Classifier | Accuracy in protocol (%) | | |
| I | II | III |
| Wang et al. (2006) [16] | 2D/3D | 64 manual | LDA | 83.60 | 61.79 | - |
| Soyel et al. (2007) [5] | 3D mesh | 11 manual | NN | 91.30 | 67.52 | - |
| Soyel et al. (2008) [6] | 3D mesh | 83 manual | NN | 93.72 | - | - |
| Tang et al. (2008) [15] | 3D mesh | 83 manual | LDA | 95.10 | 74.51 | - |
| Tang et al. (2008) [36] | 3D mesh | 83 manual | SVM | 87.10 | - | - |
| Mpiperis et al. (2008) [13] | 3D mesh | global registration | ML | 90.50 | - | - |
| Gong et al. (2009) [2] | 3D depth | global registration | SVM | - | 76.22 | - |
| Zhao et al. (2010) [14] | 2D+3D | 19 automatic | BBN | 82.30 | - | - |
| Berretti et al. (2010) [19] | 3D depth | 27 manual | SVM | - | - | 77.54 |
| Lemaire et al. (2011) [38] | 3D mesh | 21 automatic | SVM | - | 75.76 | - |
| Li et al. (2012) [39] | 3D depth | global registration | MKL | - | - | 80.14 |
| Zeng et al. (2013) [21] | 3D depth | 3 automatic | SRC | - | - | 70.93 |
| Zhen et al. (2015) [22] | 3D mesh | global registration | SVM | - | 84.50 | 83.20 |
| Yang et al. (2015) [40] | 3D mesh | global registration | SVM | - | 84.80 | 82.73 |
| Li et al. (2015) [11] | 2D+3D | 49 automatic | SVM | - | 86.32 | - |
| Li et al. (2015) [18] | 2D+3D | automatic | CNN+SVM | - | 84.87 | 83.48 |
| Ours (Deep Learning) | 2D+3D | automatic | CNN | - | 88.53 | 85.0 |

Table 3: Performance comparison with the state-of-the-art methods on BU-3DFE database.

/ *Procedia Computer Science 00 (2016) 1–13* 12

# Conclusions

In this paper, we present a new convolutional neural network that combines two models in 2D + 3D feature-based facial expression database: frontal view texture and depth image. The main advantage of CNNs is the significant features that are learned automatically by process of training. Actually, we automatically localize facial landmarks by utilizing convolutional kernel. Furthermore, the efficient of color image, gray scale image, depth image, and combined input have been conducted. The color information contributes significantly on FER while depth information make our model more robust. Our results show that our network achieves state-of-the-art results on BU-3DFE database.

# References

[1] A. S. Mian, M. Bennamoun, R. Owens, An efficient multimodal 2d-3d hybrid approach to automatic face recognition, Pattern Analysis and Machine Intelligence, IEEE Transactions on 29 (11) (2007) 1927–1943.

[2] B. Gong, Y. Wang, J. Liu, X. Tang, Automatic facial expression recognition on a single 3d face by exploring shape deformation, in: Proceed- ings of the 17th ACM international conference on Multimedia, ACM, 2009, pp. 569–572.

[3] M. Liu, S. Li, S. Shan, X. Chen, Au-aware deep networks for facial expression recognition, in: Automatic Face and Gesture Recognition (FG), 2013 10th IEEE International Conference and Workshops on, IEEE, 2013, pp. 1–6.

[4] M. Liu, S. Li, S. Shan, X. Chen, Au-inspired deep networks for facial expression feature learning, Neurocomputing 159 (2015) 126–136. [5] H. Soyel, H. Demirel, Facial expression recognition using 3d facial feature distances, in: Image Analysis and Recognition, Springer, 2007,

pp. 831–838.

[6] H. Soyel, H. Demirel, 3d facial expression recognition with geometrically localized facial features, in: Computer and Information Sciences, 2008. ISCIS’08. 23rd International Symposium on, IEEE, 2008, pp. 1–4.

[7] A. Krizhevsky, I. Sutskever, G. E. Hinton, Imagenet classification with deep convolutional neural networks, in: Advances in neural informa- tion processing systems, 2012, pp. 1097–1105.

[8] Y. LeCun, Y. Bengio, Convolutional networks for images, speech, and time series, The handbook of brain theory and neural networks 3361 (10).

[9] C. Szegedy, W. Liu, Y. Jia, P. Sermanet, S. Reed, D. Anguelov, D. Erhan, V. Vanhoucke, A. Rabinovich, Going deeper with convolutions, arXiv preprint arXiv:1409.4842.

[10] T. Fang, X. Zhao, O. Ocegueda, S. K. Shah, I. A. Kakadiaris, 3d facial expression recognition: A perspective on promises and challenges, in: Automatic Face & Gesture Recognition and Workshops (FG 2011), 2011 IEEE International Conference on, IEEE, 2011, pp. 603–610.

[11] H. Li, H. Ding, D. Huang, Y. Wang, X. Zhao, J.-M. Morvan, L. Chen, An efficient multimodal 2d+ 3d feature-based approach to automatic facial expression recognition, Computer Vision and Image Understanding 140 (2015) 83–92.

[12] S. Ramanathan, A. Kassim, Y. Venkatesh, W. S. Wah, Human facial expression recognition using a 3d morphable model, in: Image Processing, 2006 IEEE International Conference on, IEEE, 2006, pp. 661–664.

[13] I. Mpiperis, S. Malassiotis, M. G. Strintzis, Bilinear models for 3-d face and facial expression recognition, Information Forensics and Security, IEEE Transactions on 3 (3) (2008) 498–511.

[14] X. Zhao, D. Huang, E. Dellandre´a, L. Chen, Automatic 3d facial expression recognition based on a bayesian belief net and a statistical facial feature model, in: Pattern Recognition (ICPR), 2010 20th International Conference on, IEEE, 2010, pp. 3724–3727.

[15] H. Tang, T. S. Huang, 3d facial expression recognition based on automatically selected features, in: Computer Vision and Pattern Recognition Workshops, 2008. CVPRW’08. IEEE Computer Society Conference on, IEEE, 2008, pp. 1–8.

[16] J. Wang, L. Yin, X. Wei, Y. Sun, 3d facial expression recognition based on primitive surface feature distribution, in: Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on, Vol. 2, IEEE, 2006, pp. 1399–1406.

[17] A. Maalej, B. Ben Amor, M. Daoudi, A. Srivastava, S. Berretti, Local 3d shape analysis for facial expression recognition, in: Pattern Recognition (ICPR), 2010 20th International Conference on, IEEE, 2010, pp. 4129–4132.

[18] H. Li, J. Sun, D. Wang, Z. Xu, L. Chen, Deep representation of facial geometric and photometric attributes for automatic 3d facial expression recognition, arXiv preprint arXiv:1511.03015.

[19] S. Berretti, A. d. Bimbo, P. Pala, B. B. Amor, M. Daoudi, A set of selected sift features for 3d facial expression recognition, in: Pattern Recognition (ICPR), 2010 20th International Conference on, IEEE, 2010, pp. 4125–4128.

[20] O. Ocegueda, T. Fang, S. K. Shah, I. A. Kakadiaris, Expressive maps for 3d facial expression recognition, in: Computer Vision Workshops (ICCV Workshops), 2011 IEEE International Conference on, IEEE, 2011, pp. 1270–1275.

[21] W. Zeng, H. Li, L. Chen, J.-M. Morvan, X. D. Gu, An automatic 3d expression recognition framework based on sparse representation of conformal images, in: Automatic Face and Gesture Recognition (FG), 2013 10th IEEE International Conference and Workshops on, IEEE, 2013, pp. 1–8.

[22] Q. Zhen, D. Huang, Y. Wang, L. Chen, Muscular movement model based automatic 3d facial expression recognition, in: MultiMedia Model- ing, Springer, 2015, pp. 522–533.

[23] L. Yin, X. Wei, Y. Sun, J. Wang, M. J. Rosato, A 3d facial expression database for facial behavior research, in: Automatic face and gesture recognition, 2006. FGR 2006. 7th international conference on, IEEE, 2006, pp. 211–216.

[24] T. Kumar, K. Verma, A theory based on conversion of rgb image to gray image, International Journal of Computer Applications 7 (2) (2010) 7–10.

[25] B. T. Phong, Illumination for computer generated pictures, Communications of the ACM 18 (6) (1975) 311–317. [26] L. Lemay, J. Couch, K. Murdock, 3D Graphics and VRML 2.0, Sams. net Publishing, 1996.

[27] M. Daoudi, A. Srivastava, R. Veltkamp, 3D face modeling, analysis and recognition, John Wiley & Sons, 2013.

/ *Procedia Computer Science 00 (2016) 1–13* 13

[28] Y. Bengio, A. Courville, P. Vincent, Representation learning: A review and new perspectives, Pattern Analysis and Machine Intelligence, IEEE Transactions on 35 (8) (2013) 1798–1828.

[29] M. D. Zeiler, R. Fergus, Visualizing and understanding convolutional networks, in: Computer Vision–ECCV 2014, Springer, 2014, pp.

818–833.

[30] G. E. Dahl, T. N. Sainath, G. E. Hinton, Improving deep neural networks for lvcsr using rectified linear units and dropout, in: Acoustics, Speech and Signal Processing (ICASSP), 2013 IEEE International Conference on, IEEE, 2013, pp. 8609–8613.

[31] N. Srivastava, G. Hinton, A. Krizhevsky, I. Sutskever, R. Salakhutdinov, Dropout: A simple way to prevent neural networks from overfitting, The Journal of Machine Learning Research 15 (1) (2014) 1929–1958.

[32] I. Sutskever, J. Martens, G. Dahl, G. Hinton, On the importance of initialization and momentum in deep learning, in: Proceedings of the 30th international conference on machine learning (ICML-13), 2013, pp. 1139–1147.

[33] X. Glorot, Y. Bengio, Understanding the difficulty of training deep feedforward neural networks, in: International conference on artificial intelligence and statistics, 2010, pp. 249–256.

[34] L. Van der Maaten, G. Hinton, Visualizing data using t-sne, Journal of Machine Learning Research 9 (2579-2605) (2008) 85.

[35] Y.-l. Tian, T. Kanade, J. F. Cohn, Recognizing action units for facial expression analysis, Pattern Analysis and Machine Intelligence, IEEE Transactions on 23 (2) (2001) 97–115.

[36] H. Tang, T. S. Huang, 3d facial expression recognition based on properties of line segments connecting facial feature points, in: Automatic Face & Gesture Recognition, 2008. FG’08. 8th IEEE International Conference on, IEEE, 2008, pp. 1–6.

[37] L. Tian, C. Fan, Y. Ming, J. Shi, Srdanet: An efficient deep learning algorithm for face analysis, in: Intelligent Robotics and Applications, Springer, 2015, pp. 499–510.

[38] P. Lemaire, B. Ben Amor, M. Ardabilian, L. Chen, M. Daoudi, Fully automatic 3d facial expression recognition using a region-based approach, in: Proceedings of the 2011 joint ACM workshop on Human gesture and behavior understanding, ACM, 2011, pp. 53–58.

[39] H. Li, L. Chen, D. Huang, Y. Wang, J. M. Morvan, 3d facial expression recognition via multiple kernel learning of multi-scale local normal patterns, in: Pattern Recognition (ICPR), 2012 21st International Conference on, IEEE, 2012, pp. 2577–2580.

[40] X. Yang, D. Huang, Y. Wang, L. Chen, Automatic 3d facial expression recognition using geometric scattering representation, in: Automatic Face and Gesture Recognition (FG), 2015 11th IEEE International Conference and Workshops on, Vol. 1, IEEE, 2015, pp. 1–6.