Smiling and Non-smiling Emotion Recognition Based on Lower-half Face using Deep-Learning as Convolutional Neural Network

Fahad L. Malallah[1], Ahmed A. Al-Jubouri[1], Abdulbasit M. Sabaawi[1], Baraa T. Shareef[2],
Mustafa G. Saeed[3], Khaled N. Yasen[4]
{*Fahad.malallah@uoninevah.edu.iq*, ahmed.mohammed@*uoninevah.edu.iq*,
abdulbasit.ahmed@uoninevah.edu.iq}

[1]Computer and Information, College of Electronics Engineering, Ninevah University, Mosul, Iraq
[2]Department of Information Technology, College of Information Technology, Ahlia University, Manama, Kingdom of Bahrain
[3]Cihan-Slemani University, College of Science, Department of Computer Science, Iraq
[4]Department of Computer Science, Cihan University-Erbil, Erbil 44001, Iraq

Abstract. Image understanding is considered important among researchers. In this paper, a new technique is proposed to classify a detected face into two classes as a smile or non-smile category. First, the system detects and segments only the face. Then, it converts the image from RGB to Gray-Scale and enhances the image via an equalization technique. Where the contribution of this research is depending only on the lower half of the face since most of the smiling information can be perceived from the mouth and its perimeter. Then, a convolutional neural network (CNN) is applied to generate two output nodes. Public GENKI-4K database is used for the experiments, which contains 4000 challenged face images. The results demonstrate that the accuracy with 4-Fold cross-validation is 91%. This approach achieves a promising performance as compared with the state-of-the-art techniques in both accuracy and processing time.

Keywords: Computer Vision, Classification, Convolutional, Emotion recognition, Machine Learning, Neural Network.

1 Introduction

Recently, people are spending more time on digital devices such as laptops, smart mobile devices, etc, in which their digital activities generate large amounts of data that can be fed to several learning algorithms to be part of applications of human-computer interaction (HCI)[1], which is mainly to improve the quality of interaction between user and computer. One of these applications is human emotion recognition, which is considered as an attempt for the computing system to understand human face emotion. Generally, human emotion can be carried out via speech emotion[2] or face emotion systems [3, 4]. Face emotion is the closest to shape recognition, as it is considered important in such applications of computer vision, entertainment, and robotics[5] and remains the focus of this paper. A learning computing system is carried out via Machine Learning (ML) techniques, which have been used for decades to detect objects or regions of interest (ROI) in images, then to classify or identify classes of different objects. This technique extracts features representing regions, points, or objects of interest and afterward uses those features to train a model to classify or learn patterns in the image data. In machine learning, feature selection is a time-consuming manual process that usually involves processing each image with one or more image processing operations, such as calculating principal components

IMDC-SDSP 2020, June 28-30, Cyberspace

DOI 10.4108/eai.28-6-2020.2298175

analysis (PCA) or calculating gradient to extract the discriminative information from each image. On the other hand, deep learning (DL) algorithms [6, 7] is one of the important types of ML that can learn features, representations, and tasks directly from images, sound, text without feature extraction or feature selection stages in the computer vision system. One of the most prevalent DL algorithms is named Deep Neural Networks (DNNs), which outperforms traditional models in numerous recognition missions used with Human-Machine Interaction (HMI) applications[8].

Generally, facial expression recognition is one of the most important parts of Human-Computer Interaction (HCI). For example, it has been widely explored as a way to understand and communicate with children as they may not utter a clear speech [9]. This application attracted a large number of research interests in the field of computer vision. It possesses several potential applications in gaming, human-to-human interaction and human-to-computer as well. However, various types of feelings have been recognized namely anger, boredom, fear, disgust, happiness, neutral and sadness with a generalized feature set in real-time [10].

Additionally, lightweight devices such as smartphones have been used to assess the emotion, where emotion-aware mobile applications have been increasing because of the user acceptability. At the same time, the emotion recognition system should be in real-time by processing frame by frame of video as performed in this work [11]. Since mobile devices have limited processing power, the algorithm in the recognition system must be implemented using fewer computing resources compared to the existing work [12]. Accordingly, it requires a smaller matrix of the face image to be processed to detect the face then recognize whether it is a smiling face or non-smiling. This is an important criterion especially in the emerging field of big data science. Therefore, it is necessary to design a system with a low processing time requirement. In this study, we only focus on the lower half of the face. In other words, the recognition is depending on the mouth and its perimeters. Such a move provides for a smaller amount of data. However, careful design is needed to ensure that the system is accurate and efficient. Here, the human face is divided into two parts horizontally: upper and lower, and the lower part, which contains the mouth, will be the target (ROI). The objective of this research is to minimize the image matrix of the computation in which it will be very beneficial especially in the case of using big data, and for lightweight devices.

The current paper is organized as follows; Section 2 covers related work, Section 3 presents the methodology of the proposed idea with testing and analysis. Section 4 comprises the experiment setup; Section 5 discusses the results. Finally, Section 6 concludes the proposed method with this research which includes possible directions for future work.

2 Related work

In this section, recent techniques related to smile detection and recognition have been discussed. In a study conducted by [13], a pyramid histogram of oriented gradient (PHOG) and local binary pattern for image feature representation using an adaptive local weight assignment have been used as feature extraction. Where a correlation-based filter feature selection approach is adopted to minimize redundancy and extract the most relevant facial information from the feature vectors. In terms of classification, kernel-based classifiers such as support vector machine (SVM) [14] and kernel extreme learning are utilized for performing classification. Another work as stated [15], using convolutional neural networks for facial emotion detection. It uses auto-encoders to construct a unique representation of each emotion, as well as 8-layer CNN. Furthermore, the JAFFE database was used for training processing. For the testing part, the result is tested in the JAFFE Test Set, which has a set of 852 digital images, where they achieved a performance of up to 86.38%. However, the dataset, which utilized in this study contains a few numbers of digital images. Another study which is done by [16, 17], a smiling face detection framework named SmileNet is proposed, which uses a Fully Convolutional Neural Network (FCNN) to detect multiple smiling faces in a given image of varying resolution as claimed. For testing, the GENKI-4K database has been used to conduct experiments and the results showed that Smile-Net can deliver a performance of 95.76%, even under occlusions, and variances of a pose, scale, and illumination as claimed in the paper. However, in terms of our perception that the accuracy has been obtained is suitable with the image size $[300 \times 300 \times 3\ ]$as a colored image, but the training speed shown is slow performance, especially in the real-time training and testing. Also, smile detection in the wild with deep learning is proposed in [18], in which the deep convolutional network is exploited to handle this problem to perform feature learning and smile detection. Experiments were conducted on the GENKI-4K database demonstrate and scored a performance in smile detection up to 90.6%. Another smile detection in the wild is proposed in [19]. Here, the Gabor filters with multi-scale and multi-orientation are applied to extract facial textures from the input face image. Afterward, Histograms of Oriented Gradients (HOG) are then employed to encode these extracted Gabor faces to capture and characterize the facial appearance characteristics, then, a pooling strategy to transform the multiple HOG features into a global visual feature called Gabor-HOG. Regarding the classification, SVM is trained by using the GENKI-4K database to distinguish a smiling face from a non-smile face. This method also achieved a performance 91.6%. Also, smile detection work is presented in [20]. Here, faces are detected using a multi-view face detector and aligned and scaled using automatically detected eye locations. Then, a convolutional neural network (CNN) is exploited to determine whether it is a smiling face or not. Experimental results showed that the overall accuracy without an alignment is 92.5% by using the GENKI-4K database for training and testing with face image size is $[32 \times 32]$ for the full face image. Also, it is used for fire detection as in [21].

It is worth to mention that the contribution knowledge of the proposed technique is that the training and testing can be done on the face image for recognizing whether it is a smiling face or not, based on only the lower portion of the tested digital image, in case it is divided into two equal parts upper and lower. This assumption differs from all aforementioned existing techniques for smile and non-smile which are depending on the whole face image. The benefit of this contribution is used to save time computation, as well as at the same time having the same recognition accuracy compared to the state-of-the-art techniques, which has a very important advantage to the big data processing field and suitable for lightweight devices.

3 Methodology

The proposed work starts by handling the cropped face image by using one of the existing algorithms for real-time face detection such has Voila-Jonse [22]. Figure 1 depicts the steps of the proposed algorithm. The proposed smile classification largely depends on the shape, therefore, converting from RGB to the gray-scale image is proposed to simplify the computation for further steps. Next, image enhancement tools are used to highlight the possible details of the face image, which is equalization operation. Furthermore, for normalization, resizing to 2-D matrix as [26×26], because some image has dimensions less than [26×26] and other tested images contain bigger matrix size, therefore, to unify the process, the aforementioned size has been selected, which is also suitable for the neural network classification in terms of computation times. In the next step, the image will be divided into two parts: lower and upper parts. Intuitively, the human visual perception will recognize the smiling individual from non-smile by looking and concentrating in the lower part of the face, which is the mouth shape. In other words, if the mouth is relatively wide compared to the specific face and both upper and lower lips are separated from each other, as well as teeth might appear, then, all that can be considered as conditions for referring to the smiling individual. Accordingly, the lower part of the face image is essential for the recognition in this algorithm. Besides that, making recognition operation exclusively for the lower part of the face image has another advantage, which is a fast running or saving the computing time.

According to the proposed method, we have found that the upper part of the face image is not significant and it can be ignored, and this is experimentally approved in the result and discussion section. After that, for feature extraction and classification, CNN [23, 24] will be exploited in the proposed classification as using Deep Learning [25], which is no need to extract features before the classification stage. Finally, CNN will output two classes “1” for the smile category and ”0” for the non-smile category.

A. Face Image Pre-Processing

As the first step for the input image preparation is using row digital image of human face, which is cropped by using Viola-Joins, then some tools of image processing such as converting from RGB to gray-scale as the recognition does not depend on the color, as well as, image equalization for the sake of image enhancement to boost the image by highlighting details and more edges. The next step is to unify the image size to [26x26] as it has been selected based on the database used in this paper named GENKI-4K database [26, 27].

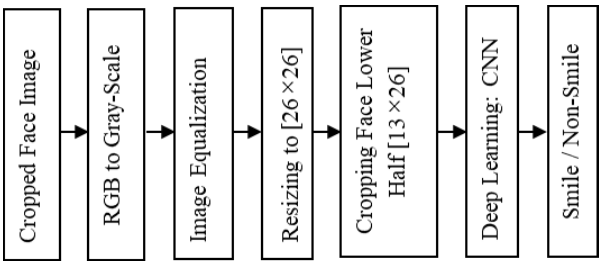
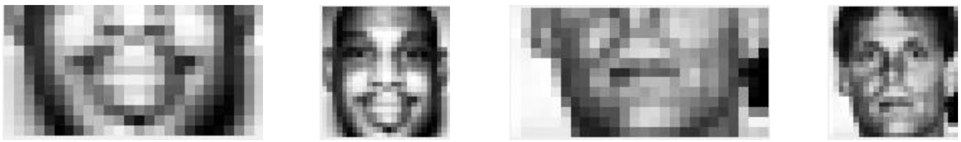


Fig 1. Framework Diagram for Smile and Non-smile Proposed Algorithm.

Practically, to speed up the training computing, a region of interest (ROI) will be only the lower part of the face image after dividing it into two equal parts in the horizontal direction as shown in Figure 2. The reason for choosing only the lower part as the most smile emotional face will be predicted based on the mouth and its borders. Figure 2 illustrates some individuals taken by the database with the whole detected and cropped digital face image for both smile and non-smile types. Features of an image can be extracted powerfully by using Convolution Neural Networks (CNNs) to identify digital images. CNNs are used primarily to find out about patterns in an image.



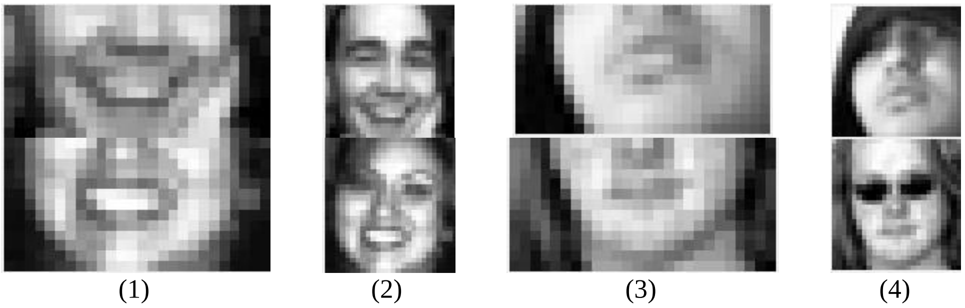
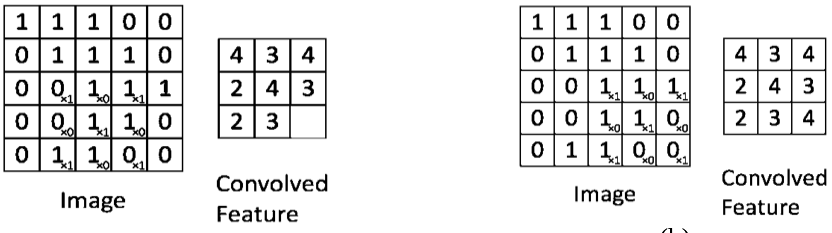


Fig 2. Depicting face image taken from Database, (1)Lower part cropped smile face [13 x 26], (2) Un-cropped smile face [26 x 26], (3)Lower part cropped non-smile face [13 x 26], (4)Un-cropped non-smile face [26 x 26].

B. Convolutional Neural Network (CNN)

Features are extracted by convoluting over an image and looking for patterns. In the first few layers of CNNs, the network can identify lines and corners. Furthermore, it can be passed these patterns down through the neural network and starts recognizing more complex features as it is called getting deeper. Therefore, this property makes CNNs so robust tool for identifying objects in digital images. In terms of the structure of CNNs, typically it contains several kinds of layers as follows, convolutional layer, pooling layer, activation layers as illustrated in Figure 3.

The "window" that moves over the image is called a mask or a kernel, which is typically square as 3x3 or might be an odd number as 5x5 or 7x7, etc. The distance the window moves each time is called the stride. Also, images are sometimes padded with "0" or "1" around the perimeter during performing convolutions. It can be concluded that the main goal of the convolutional layers is to conduct filtering processes.



1	1	1	0	0
0	1	1	1	0
0	$0_{\times1}$	$1_{\times0}$	$1_{\times1}$	1
0	$0_{\times0}$	$1_{\times1}$	$1_{\times0}$	0
0	$1_{\times1}$	$1_{\times0}$	$0_{\times1}$	0

Image

4	3	4
2	4	3
2	3	

Convolved Feature

(a)

1	1	1	0	0
0	1	1	1	0
0	0	$1_{\times1}$	$1_{\times0}$	$1_{\times1}$
0	0	$1_{\times0}$	$1_{\times1}$	$0_{\times0}$
0	1	$1_{\times1}$	$0_{\times0}$	$0_{\times1}$

Image

4	3	4
2	4	3
2	3	4

Convolved Feature

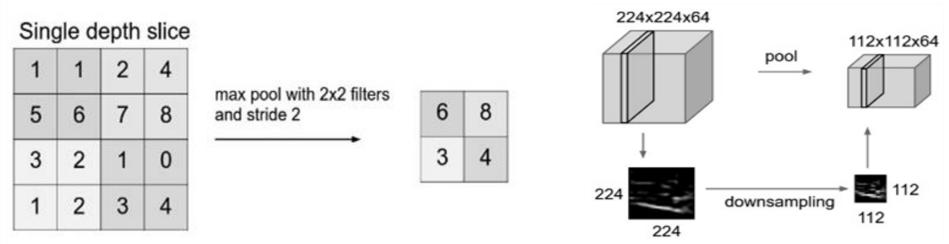
(b)

Fig 3. Convolution layer, (a) Convolution window of the image with the mask for before-last step, (b) Convolution window of the image with the mask for the last step.

Assume that CNNs consists of convolutional layers, which are characterized by an input image as Gray-scale (I), and filters (K) and biases (b), and filter (kernel) with a dimensional area of k1×k2 has m by n as in formula (1).

$$(I * K)_{i\,j} = \sum_{m=0}^{k_1-1} \sum_{n=0}^{k_2-1} K_{mn} \bullet I_{(i+m)(j+n)} + b \tag{1}$$

The next layer of CNNs is named the pooling layer as in Fgiure 4. It is specified as a form of non-linear down-sampling process, which has several types as max pooling and average pooling.



1	1	2	4
5	6	7	8
3	2	1	0
1	2	3	4

6	8
3	4

(a) (b)

Fig 4. Depicting Max-pooling layer, (a) Image 2-D matrix pooling with [2x2] filter and strides 2, (b) Illustrating pooling [224×224] with 64 channels down to [112×112] size.

Max-pooling as in Figure 4 takes the largest value from the window of the image covered by the kernel, while average pooling takes the average of all values in the window. Max-pooling partitions the input image into a set of non-overlapping sub-region, then the process is selecting the maximum value of this sub-region as explained and depicted in Figure 4. The reason why that max-pooling is useful due to the purposes, first by eliminating non-maximal values, it reduces computation for upper layers. Second, it provides a form of translation invariance.

Since it provides additional robustness to the position, max-pooling is a “smart” way of reducing the dimensionality of intermediate representations. The next layer of CNNs is named activation layers that work exactly as in other neural networks, a value is passed through a function that limits the value into a range, typically [0,1] or [-1,1].

The most used activation function in CNNs is the ReLu (Rectified Linear Unit). The main aim of using ReLus, as it is cheap to perform. The implementation of ReLu is as follows: if the input number is negative, then set it to zero (when the input is less than zero), or select the same number (when the number is more than zero). This function makes faster to train networks as in formula (2).

$$f(x) = \left\{ \begin{matrix} 0, \; forx < 0 \\ x, \; forx \geq 0 \end{matrix} \right\} \tag{2}$$

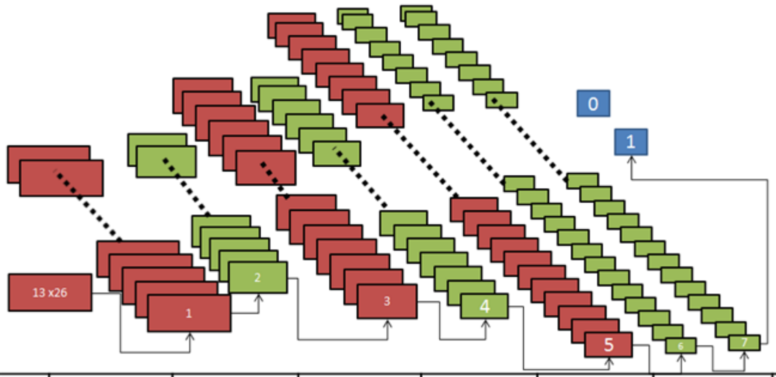
It is worth to mention that after each convolution layer, batch normalization operation is implemented so as to accelerating deep network training by reducing internal covariate shift as in [28].

The proposed CNNs architecture of this research, which is designed intentionally to classify tested face images as smile and non-smile, is illustrated in Figure 5. As it is clear that the input digital image size is [13×26], for the process number *1 as referred in Figure 5, the image size remains fixed, but it has 13 channels (convolution filters), number 13 has been selected during setting up the tuned parameters.

Afterward, process number *2, where the digital image size becomes [8×14×13], which undergoes a max_pooling operation in order to reduce the matrix size with the effect of padding operation. For next process number *3, the image size becomes the same but with increasing the channels up to 28.

For the process number *4, the size reduced to [4×7×28], which is the same number of channels but with reducing the matrix size. For the process number *5, third convolution operation occurred having 68 channels, and process number *6 the image size has been reduced to [3×4×68].

Finally, the fully-connected operation is applied in the remaining digital image that results [3×68], in other words, it is 204 features, which are representing the input digital image of the neural network as MLP to output two nodes that are the two categories: “0” for the non-smiling and “1” for the smiling category.



Face Image Size:	*1 13×26×13	*2 8 × 14×13	*3 8× 14×28	*4 4×7 × 28	*5 4×7×68	*6 3×4×68	*7 2×204

Fig 5. The proposed CNN structure and layers with its details.

The process numbers aforementioned in Figure 5 are well elaborated in Table 1. For instance, process number *1 has the following processes: 'conv_1', for convolution operation, 'batchnorm_1' for Batch Normalization and 'relu_1' for ReLU. The rest process numbers such *2, *3,*4,*5, *6 and *7 are explained in Table I. Eventually, there will be two categories as two “0” for non-smiling and “1” for the smiling category.

Table 1. Shows an explanation of the CNN 16-layers as process operation for each face image.

Input Image Size	*1 13×26×13	*2 8 × 14×13	*3 8× 14×28	*4 4×7 × 28	*5 4×7×68	*6 3×4×68	*7 2×204	Output Categories
1) 'image input' Image Input 13x26x1 images with 'zerocenter' normalization.	2) 'conv_1' Convolution with 13 3x3x1 convolutions with stride [1 1] and padding: [1 1 1 1] 3) 'batchnorm_1' Batch normalization with 13 channels 4) 'relu_1' ReLU.	5) 'maxpool_1' Max Pooling 2x2 with stride [2 2] and padding: [0 0 0 0]	6) 'conv_2' Convolution with 28 3x3x13 convolutions with stride [1 1], padding [1 1 1 1] 7) 'batchnorm_2' Batch normalization with 28 channels 8) 'relu_2' ReLU	9) 'maxpool_2' Max Pooling 2x2 max pooling with stride [2 2] and padding [0 0 0 0]	10) 'conv_3' Convolution with 68 3x3x28 convolutions with stride [1 1] and padding [1 1 1 1] 11) 'batchnorm_3' Batch normalization with 68 channels 12) 'relu_3' ReLU	13) 'maxpool_3' Max Pooling 2x2 max pooling with stride [2 2] and padding [0 0 0 0]	14) 'FC' Fully Connected 2 fully connected layer 15) 'softmax' Softmax	16) two classes '0' and '1'

4 Experiments

To evaluate the proposed recognition algorithm for classifying the digital face image into two categories as smile or non-smile, GENKI-4K database has been used [26], which is specified as an expanding database of images containing faces spanning a wide range of illumination conditions, geographical locations, personal identity, and ethnicity. The GENKI-4K contains

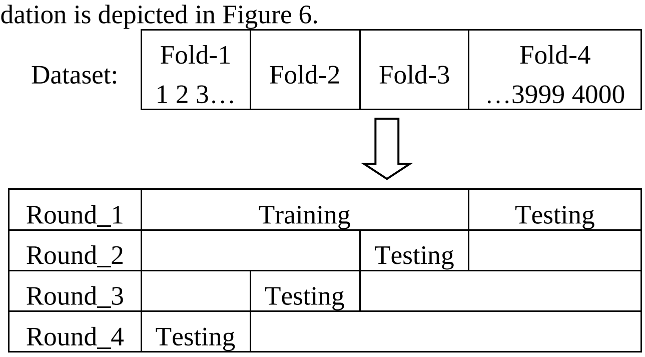
4000 face images labeled as either “smile” indexed from 1 to 2162 face image sample, and another label named “non-smile” indexed from 2163 to 4000. In this research, three experimental types have been conducted to test the performance of the proposed algorithm. First, in terms of the accuracy and performance of the various structures of CNNs are conducted to choose the optimum structure of the CNN. Secondly, after settling down the best CNNs architecture with their parameters, two types of experiments have been conducted in terms of percentage of training and testing samples, At first, the training percentage is 70% and testing is 30% of the 4000 sample face images , and secondly, experiment has the percentage of training and testing as 50% and 50 % respectively. Furthermore, to get more robust and trusted results, cross-validation with 4-fold has been implemented to get the result, which is the closest to reality.

About the cross-validation, four running experiments have been conducted, as each experiment has one fold for testing and the three remaining folds for training (each fold contains smiling and non-smiling image samples, divided equally among the folds). Each fold contains the following equation (3) and (4):

$$Smile_{no.} = \frac{2162(total\_Smile\_Samples}{4(fold)} \approx 540 \tag{3}$$

$$Non\_Smile_{no.} = \frac{4000 - 2162(total\_Smile\_Samples}{4(fold)} \approx 459 \tag{4}$$

After that, each fold contains 540 as smile samples, as well as to the 459 as non-smile samples. One fold has 999 samples, which will be used in the training and testing operations. Now, for the second experiment, the second fold is used for testing and the three remaining folds for training. Similarly, for the third experiment, the third fold is used to test the full set and the rest for training. Finally, for the fourth experiment, the fourth fold will be used for testing and the rest for training. Now, the overall accuracy is considered the average of the four experiments, this is the meaning of cross-validation as 4-fold, in which it is very paramount because there is no enough data to train a model and test as perfectly, so that it is a method that provides ample data for training the model and also leaves ample data for validation. A full description of the 4-Fold cross-validation is depicted in Figure 6.



Dataset:	Fold-1 1 2 3…	Fold-2	Fold-3	Fold-4 …3999 4000

Round_1	Training			Testing
Round_2			Testing	
Round_3		Testing		
Round_4	Testing			

Fig 6. Dataset (4-Fold) cross-validation distribution.

Once there is a 4000 image sample, then, it is assumed that each fold has 999 image samples constituting both smiling and non-smiling faces images as explained above. To give the details about the indices that have been used from the GENKI-4K database, Table 2 summarizes that.

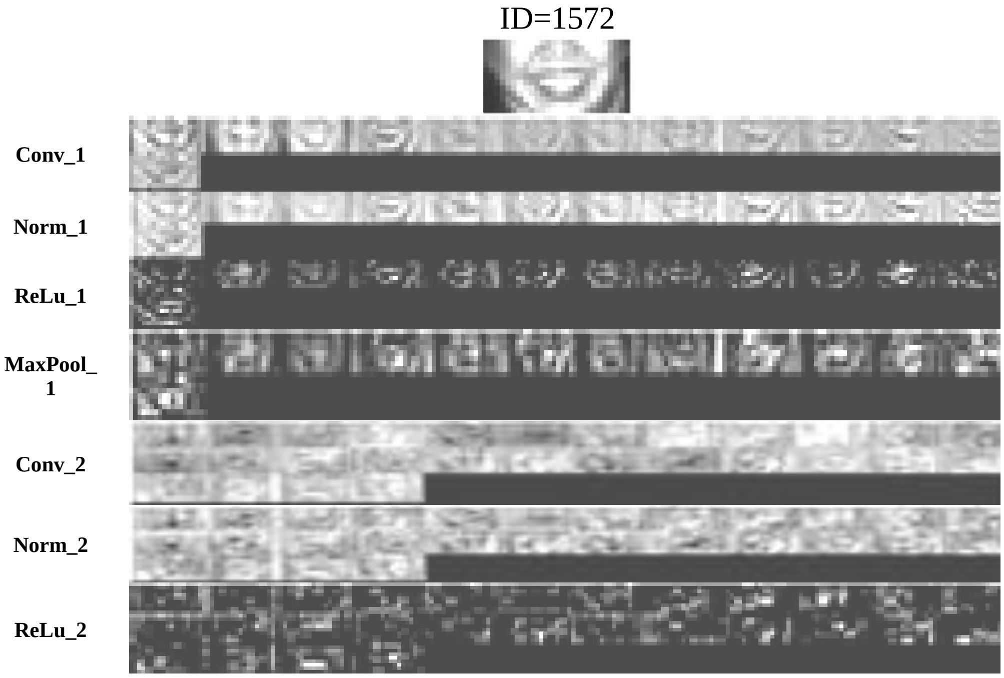
Table 2. Shows the detail of the indices in GENKI-4K on how to build the four folds in the Cross-validation.

Fold	Smile index in GENKI-4K	Non-Smile index in GENKI-4K
Fold-1	1-540	2163-2621
Fold-2	541-1080	2622-3080
Fold-3	1081-1620	3081-3539
Fold-4	1621-2160	3540-3998

The third experiment is used to consolidate this research result, an experiment is conducted to prove that the processing for the only lower part of the face image is better in terms of both accuracy and training speed. This will be accomplished by performing a comparison between the full-face image size [26×26] versus the mouth cropped face image [13×26] (lower half of the image as ROI). In other words, the full face image has been divided horizontally into equal parts then the ROI will be only for the lower part. Here, to concentrate and get more focusing on the mouth and lips of the human, because logically most of the smile and non-smile predication depends on the mouth and its perimeters shapes.

5 Result and Discussion

Visualizing a set of layers of the CNNs for random samples of smile and non-smile is preferable in this research article paper to elaborate on the result. Therefore, Figure 7 illustrates the 12 layers among the 16 layers as imaging. In terms of the CNN structure, the parameters that have been selected in the first layer named the convolution layer is 13 channels. As can be seen 13 channels (filters) image in the low-level features in Conv_1 in Figure 7, and in normalization, the layer is also visualized as it is seen in Norm_1.



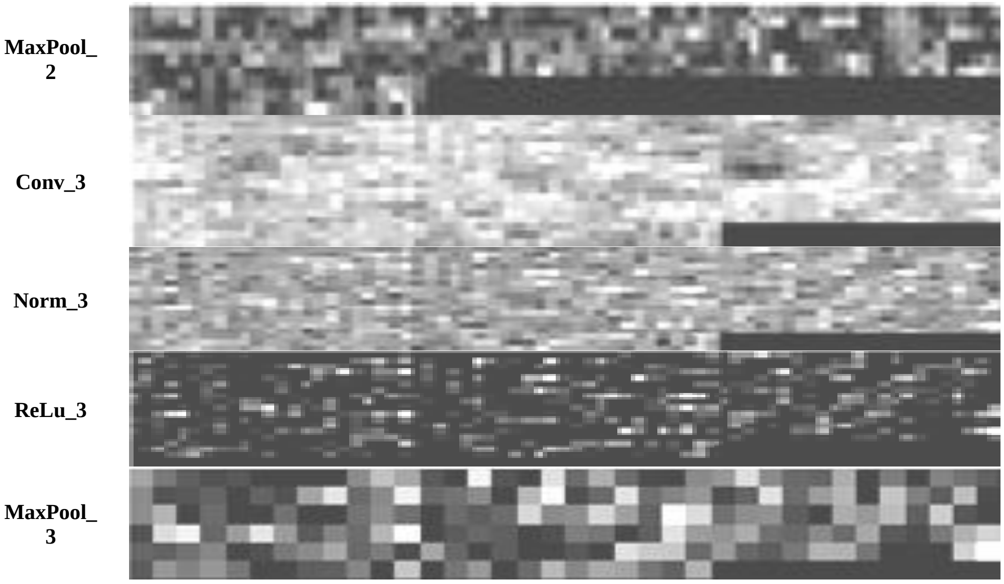


Fig 7. Visualizing CNN for the 12 layers of smiling digital face image, Id=1572 of the database.

Also, ReLu_1 is visualized, which is a function according to the sigmoid function that affects on the weights. Then, max-pooling, which is MaxPool_1 that has stride 2, is also illustrated. After that, the same operations are done but with second turn layers using a different number of convolutional channels as 28. Next, the third turn of layers, which are repeating the convolution Conv_3, normalization Norm_3, ReLu_3, and MaxPool_3 with batch processing layers, is also applied using 68 convolutional channels on the features taken from the second layers, to be passed later on to the fully connected layers. In other words, getting deeper coefficient features, as it is seen in Figure 7, the MaxPool_3 cannot be recognized by the human eyes.

However, this will be submitted along to the fully connected layers to be undergone as a Multi-layer perceptron (MLP) neural network. Identically, we have also carried out the random non-smile face that is not described here, which has also shown the same layers and structure of 13, 28 and 68 channels as explained above. It is worth to mention that, these two examples show that the extracted features have been conducted by CNN directly and going deeper and deeper by using convolutional operations layers gradually.

In terms of accuracy, Table 3 reports the recorded accuracy by using the dataset. The distribution here is 70% for training and 30% for testing. In other words, for training set contains 3001 face samples (number of smile sample is 1622, and for non-smile is 1379). About the testing set, it has 999 for both the smile category, which has 540 and Non-Smile, which has 459 digital image samples. Here, in the training and testing is a gray-scale image as [13×26×1] dimensions as image size (it is selected after extensive experiments).

The changing parameters here are the number of the convolution layers (channels) with max-pooling and ReLu., as well as, the number of Epoch. As it is clear from Table 3 the best accuracy that has been reported is 91.45 with 250 epoch training, in which the structure of the CNN having three times of a set of convolution, normalization, ReLu., and max-pooling, then finally pass the features to the fully connected layers as MLP neural network. For the first convolutional time, 13 channels of the filter have been implemented and 28 channels for the second convolutional time and regarding the third time 68 channels have been exploited for gaining a better recognition rate. This is the meaning of deep learning as going deep to get features.

Table3. Accuracy reported in this research with different CNN structures.

No. of Epoch	CNN Structure	Accuracy %
350	16-28-64	90.99
250	13 28 68	91.09
350	13 28 68	90.89
250	13 28 56 68	89.80
250	13 64	90.29
250	14 68	88.79
250	13 20 24 28	89.39
250	13 80	89.39

In this accuracy, the number of the failed recognized face sample is 86 face samples, while 913 face samples are correctly recognized whether it is a smiling or non-smiling face. To boost the recognition rate cross-validation method for the test has been used for both cropped as only the lower part having image size [13×26×1] and non-cropped face image that has image size [26×26×1]. As shown in Table 4 two types of results have been reported as both: lower-half image face and full image face. Firstly, the accuracy of the four experimental rounds of the cross-validation with their training times in second (s), are reported. As it is shown in Table 4, the first-round experiment, which has the first three folds as training and the fourth fold is used for testing, has 91.49% accuracy with the time taken for training up to 4916 seconds (s) (almost 1.5 hours). The second and third round experiments have the same accuracy as 91.29%. Finally, the fourth cross-validation round experiment has 90.09%. As the role of the cross-validation, the overall accuracy is considered the average of the four folds, which is 91.04% and the average time running is approximately 1.36-hour running on a workstation specified as Core2 Duo CPU, 2.0 GHz, 2,5 with 4G RAM. It is worth to mention that the detail of each round division has been explained in the Experiment section. Secondly, regarding the results of the full-face image, experiments have been reported in Table 4 in the last two columns.

Table 4. Cross-validation experiment with 4-Fold for cropped face image (Lower part & Full Face image).

Cross-validation	Dataset (4-Fold) Cropped Mouth (lower part) [13×26]		Dataset (4-Fold) Full Face Image [26×26]	
	Accuracy %	Training Time(s)	Accuracy %	Training Time(s)
Round_1	91.49	4916	90.49	8916
Round_2	91.29	4901	91.59	8713
Round_3	91.29	5068	91.09	8732
Round_4	90.09	4749	90.49	8652

Besides, the accuracy of each round by using the full image sample (without cropping the lower part ROI). As it is clear that the reported accuracy has a slight reduction than that reported in the Cropped Mouth (lower part) [13×26×1]. Moreover, the time consuming for the training neural network is longer, as the best-recorded accuracy, which is 91.59% for the second fold experiment, has training time almost 2.4 hours. The overall accuracy as the average of the four rounds is 90.9% with time-consuming as average approximately up to 8753 second or 2.43 hours. Accordingly, in terms of comparison, the performance of the half-face image as the lower part horizontally has a better performance than that of the full-face image. Thus, this is an aspect of the contribution of this article that taking the training of mouth region of the face with its perimeters is the outstanding discovery in terms of the smile and non-smile recognition while leaving the upper half part of the face image, as it is insignificant for the classification and do nothing just wasting time for the extra process.

Table 5. Comparison accuracy between the lower part image face and the full image face.

Type of Face Training	Percentage 70%	Percentage 50%
Lower half	91.4	90.3
All face	89.9	89.4

Other experiments, which have been conducted in the research, show that the comparison between full image face training and the lower part of the face ROI training for both 70% percent of the data training and the rest is 30% for testing, as well as another percentage as 50% for training and other 50% percent for testing. The results of the described experiments are listed in Table 5. It is clear that the accuracy resulted in the experiment of half image face has a better accuracy than the full image face in terms of both experiments of 70% and 50%. Furthermore, it is faster in terms of time computing of training. However, the obtained accuracy of the proposed methodology certainly has less time-consuming in terms of training and testing. This can be justified as half of the face image has been used with matrix dimension [13×26×1]. In comparison, most of the research papers in the literature review are taking the full image. Accordingly, it is concluded the proposed method is faster than the state-of-the-art techniques if using the same computing workstation and CPU speed, as well as with relatively, it has the same or a noticeable less accuracy. This is very beneficial in terms of big data science to save the storage devices of the trained image matrix and speed-up the training and testing time in the real-time execution. Moreover, a comparison has been carried out with the recently published works as explained in Table 6, which is based on two factors: face image size and accuracy of the performance. The comparison revealed that the proposed work has the smallest face image matrix used in our experiments, which is useful for the field of big data processing for time-saving. However, it is clear from Table 6 that there is a slight decreasing for the performance accuracy of our proposed techniques as compared with the existing smile classification work, this is because reducing the face image matrix down to [13×26×1], where it is only exploiting the mouth and its border as ROI for the training the testing. Overall, the efficiency of decreasing a large number of image pixels is sufficient versus losing a small percentage of the performance accuracy.

Table 6. Shows a comparison between the proposed and the state-of-the-art techniques in terms of both face image size and accuracy.

Article, Technique, Citation	Face Image Size	Accuracy %
Jang, 2018,CNN, [17]	*[300×300×3]*	95.7
Chen, 2017, CNN, [18]	*[64×64×1]*	90.6
Li,2016, Gabor-HOG, SVM, [19]	*[64×64×1]*	91.6
Bianco, 2016, CNN, [20]	*[32×32×3]*	92.5
Proposed Technique, CNN.	***[13×26×1]***	**91.0**

6 Conclusion

A new algorithm is proposed and tested for recognizing smiling and non-smiling face emotions. The algorithm is exploiting only the lower part of the image face of the individual, which is taking advantage of the texture of the mouth and the perimeter. This algorithm shows an efficient performance with big data science and real-time rendering due to the smallest image matrix as [13×26×1] used as compared with the existing techniques at the same time gained relatively the same accuracy. In the following steps, the image matrix is passed to the Deep-Learning as a convolutional neural network (CNN) with proposing a new structure of the CNN consisting of 16 layers and channels 13, 28 and 68. The proposed algorithm is tested by conducting several experiments described in the experiment section using the famous database named GENKI-4K. The accuracy has been achieved in this article paper up to 91% the result is supported by using the Cross-validation method. As future work, the algorithm can be developed by designing the framework to recognize multiple faces sentimental simultaneously as well as improving the recognition success rate.

References:

[1] S. M. Pablos, *et al.*, "Dynamic facial emotion recognition oriented to HCI applications," *Interacting with Computers,* vol. 27, pp. 99-119, 2013.

[2] A. Mohanta and U. Sharma, "Detection of Human Emotion from Speech—Tools and Techniques," in *Speech and Language Processing for Human-Machine Communications*, ed: Springer, 2018, pp. 179-186.

[3] N. Fragopanagos and J. G. Taylor, "Emotion recognition in human–computer interaction," *Neural Networks,* vol. 18, pp. 389-405, 2005.

[4] S. Lugović*, et al.*, "Techniques and applications of emotion recognition in speech," in *Information and Communication Technology, Electronics and Microelectronics (MIPRO), 2016 39th International Convention on*, 2016, pp. 1278-1283.

[5] S. J. Goyal*, et al.*, "Real-Life Facial Expression Recognition Systems: A Review," in *Smart Computing and Informatics*, ed: Springer, 2018, pp. 311-331.

[6] C. F. Higham and D. J. Higham, "Deep Learning: An Introduction for Applied Mathematicians," *arXiv preprint arXiv:1801.05894,* 2018.

[7] S. Khan and T. Yairi, "A review on the application of deep learning in system health management," *Mechanical Systems and Signal Processing,* vol. 107, pp. 241-265, 2018.

[8] N. Jain, *et al.*, "Hybrid Deep Neural Networks for Face Emotion Recognition," *Pattern Recognition Letters,* 2018.

[9] S. Kamal, *et al.*, "Facial emotion recognition for human-computer interactions using hybrid feature extraction technique," in *Data Mining and Advanced Computing (SAPIENCE), International Conference on*, 2016, pp. 180-184.

[10] S. Lalitha, *et al.*, "Emotion Recognition through Speech Signal for Human-Computer Interaction," in *Electronic System Design (ISED), 2014 Fifth International Symposium on*, 2014, pp. 217-218.

[11] O. A. Arigbabu, *et al.*, "Estimating body related soft biometric traits in video frames," *The Scientific World Journal,* vol. 2014, 2014.

[12] M. S. Hossain and G. Muhammad, "An emotion recognition system for mobile applications," *IEEE Access,* vol. 5, pp. 2281-2287, 2017.

[13] O. A. Arigbabu, *et al.*, "Smile detection using hybrid face representation," *Journal of Ambient Intelligence and Humanized Computing,* vol. 7, pp. 415-426, 2016.

[14] N. H. ABBAS, *et al.*, "Offline Handwritten Signature Recognition using Histogram Orientation Gradient and Support Vector Machine," *Journal of Theoretical and Applied Information Technology,* vol. 96, pp. 2075-2084, 2018.

[15] P. R. Dachapally, "Facial Emotion Detection Using Convolutional Neural Networks and Representational Autoencoder Units," *arXiv preprint arXiv:1706.01509,* 2017.

[16] Y. Jang, *et al.*, "SmileNet: Registration-Free Smiling Face Detection In The Wild," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 1581-1589.

[17] Y. Jang, *et al.*, "SmileNet: Registration-Free Smiling Face Detection In The Wild," 2018.

[18] J. Chen, *et al.*, "Smile detection in the wild with deep convolutional neural networks," *Machine vision and applications,* vol. 28, pp. 173-183, 2017.

[19] J. Li, *et al.*, "Smile detection in the wild with hierarchical visual feature," in *Image Processing (ICIP), 2016 IEEE International Conference on*, 2016, pp. 639-643.

[20] S. Bianco, *et al.*, "Robust smile detection using convolutional neural networks," *Journal of Electronic Imaging,* vol. 25, p. 063002, 2016.

[21] A. Namozov and Y. Im Cho, "An efficient deep learning algorithm for fire and smoke detection with limited data," *Advances in Electrical and Computer Engineering,* vol. 18, pp. 121-129, 2018.

[22] P. Viola and M. J. Jones, "Robust real-time face detection," *International journal of computer vision*, vol. 57, pp. 137-154, 2004.

[23] A. Krizhevsky, *et al.*, "Imagenet classification with deep convolutional neural networks," in *Advances in neural information processing systems*, 2012, pp. 1097-1105.

[24] C. Cernazanu-Glavan and S. Holban, "Segmentation of bone structure in X-ray images using convolutional neural network," *Adv. Electr. Comput. Eng,* vol. 13, pp. 87-94, 2013.

[25] Y. Lv, *et al.*, "Traffic flow prediction with big data: a deep learning approach," *IEEE Transactions on Intelligent Transportation Systems,* vol. 16, pp. 865-873, 2015.

[26] J. Whitehill, *et al.*, "Toward practical smile detection," *IEEE transactions on pattern analysis and machine intelligence,* vol. 31, pp. 2106-2111, 2009.

[27] G. Littlewort, *et al.*, "The computer expression recognition toolbox (CERT)," in *Automatic Face & Gesture Recognition and Workshops (FG 2011), 2011 IEEE International Conference on*, 2011, pp. 298-305.

[28] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," *arXiv preprint arXiv:1502.03167,* 2015.