Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Date of publication xxxx 00, 0000, date of current version xxxx 00, 0000.

Digital Object Identifier 10.1109/ACCESS.2017.Doi Number

SMEConvNet: A Convolutional Neural Network for Spotting Spontaneous Facial Micro-Expression from Long Videos

Zhihao Zhang[1,2], Tong Chen[1,2,3*], Hongying Meng[1,4], Guangyuan Liu[1,2], Xiaolan Fu[3,5]

[1]Chongqing Key Laboratory of Non-linear Circuit and Intelligent Information Processing, Southwest University, Chongqing, 400715, China,
[2]Chongqing Key Laboratory of Artificial Intelligence and Service Robot Control Technology, Chongqing, 400715, China
[3]Institute of Psychology, Chinese Academy of Sciences, Beijing, 100101, China
[4]Department of Electronic and Computer Engineering, Brunel University London, UB8 3PH, UK
[5]Department of Psychology, University of Chinese Academy of Sciences, Beijing, 100049, China

Corresponding author: Tong Chen (e-mail: c_tong@ swu.edu.cn).

This work was partially funded by the National Natural Science Foundation of China (Grant No. 61301297, 61502398), and the National Natural Science Foundation of China (NSFC) and the German Research Foundation (DFG) under project Cross Modal Learning(NSFC 6162113608/DFG TRR-169).

ABSTRACT Micro-expression is a subtle and involuntary facial expression that may reveal the hidden emotion of human beings. Spotting micro-expression means to locate the moment when the micro-expression happens, which is a primary step for micro-expression recognition. Previous work in micro-expression expression spotting focus on spotting micro-expression from *short video*, and with hand-crafted features. In this paper, we present a methodology for spotting micro-expression from *long videos*. Specifically, a new convolutional neural network named as SMEConvNet (Spotting Micro-Expression Convolutional Network) was designed for extracting features from video clips, which is the first time that deep learning is used in micro-expression spotting. Then a feature matrix processing method was proposed for spotting the apex frame from *long video*, which uses a sliding window and takes the characteristics of micro-expression into account to search the apex frame. Experimental results demonstrate that the proposed method can achieve better performance than existing state-of-art methods.

INDEX TERMS Spotting Micro-Expression, Apex Frame, Convolutional Neural Network, Deep Learning

I. INTRODUCTION

Facial expression analysis has been a topic of interest for many years [1, 2, 3]. As a special form of facial expression, micro-expression is getting more and more attention [4, 5, 6]. Micro-expression is a subtle and involuntary facial expression that is not subject to people's consciousness [7, 8]. There are usually two characteristics of micro-expression. One is short duration: it only lasts for 1/25 to 1/2 second [9]. Another is low intensity in facial muscle movement: not all corresponding facial muscles have movement for a specific expression, and the movement is very weak [10].

As micro-expression only occurs when people are trying to conceal their emotions, the recognition of micro-expression can uncover people's real emotion or hidden intent. The recognition of micro-expression finds application in many fields, such as emotion monitoring [8], lie detection [11], and homeland security [12].

In real applications, micro expression happens between neutral expressions because it is the result of failure of suppressing the facial muscles' movement. When a person tries to conceal his/her emotion, his/her facial expression is forced into a neutral state. In the moment when the suppression fails, the micro-expression happens. After that moment, the face will be back to neutral expression again. Thus, the recognition of micro-expression has two steps. The first step is to locate the moment when the micro-expression occurs, and the second one is to determine which category the micro-expression belongs to. This first step is called micro-expression spotting, that is a primary step for micro-expression recognition research and a focus of this paper.

The starting frame of the moment when micro-expression happens is called onset frame. And the ending frame is

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2879485, IEEE Access

named as offset frame. The frame where micro-expression reaches climax is called apex frame. To facilitate the research of micro-expression spotting, the video part from onset frame to offset frame was named as "*short video*" [13]. And the "*long video*" is the raw video sequence that may include irrelevant motion out of the *short video* (shown in Fig. 1).

Since the apex frame is the most expressive in a *short video*, instead of spotting the facial micro-movement, some researchers chose to spot apex frame [13, 15, 16]. The effectiveness of apex frame spotting can be determined by using the Mean Absolute Error (MAE) and Apex Spotting Rate (ASR), which were also used in [13]. MAE indicates the average frame distance between the ground truth and the spotted apex frame. ASR calculates the success rate in spotting apex frame within *short video*.

In CASME-RAW [26] and CASMEII-RAW [14] databases, the apex frames were manually labeled by coders from psychology department. These indexes serve as the ground truth for the micro-expression spotting research. However, to perform the labeling, at least two coders have to work separately to inspect the video clips frame by frame, which is time consuming and tedious. An automatic spotting method can save much time and energy of human coders.

Some automatic spotting works only spot apex frame from the *short video*. However, in real application, the apex frame needs to be spotted from *long videos*, not from well-segmented *short video* with clear onset and offset annotation. It is very difficult to spot micro-expression from *long videos*, because unwanted facial movements may be present on raw videos, which are outside of the *short video* and can be falsely detected as micro-expression. In this paper, we present a method that can spot the micro-expression from *long videos* without ground truth indexes of the onset and offset frames.

The automatic micro-expression spotting relies on the feature extracted from each frame in the raw video. Currently, most features used in micro-expression spotting are hand-crafted features and the selection of feature depends heavily on the experience of researchers. In this paper, we present a convolutional neural network (CNN) for automatically extracting features from frames. To the best of our knowledge, this is the first time deep learning is used for micro-expression spotting.

To find the apex frame, some researchers calculate the feature difference between current frame and reference fame, and locate apex frame as the frame that has the largest difference [13, 15]. In this paper, we use a sliding window over a feature matrix to locate the apex frame. An index value is calculated. The apex frame is located within sliding window that produces the largest index value.

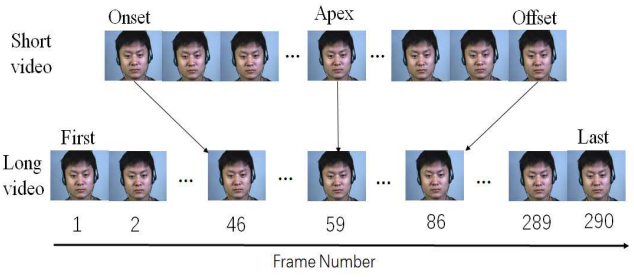

FIGURE 1. An example of a *long* and *short video* with annotated ground-truth labels indicating the onset, apex, and offset frame [14].

As described above, the main contributions of this paper are three-fold. Firstly, we present a method for spotting apex frame from *long videos* without knowing the indexes of the onset and offset frames. Secondly, we designed a CNN, named as Spotting Micro-Expression Convolutional Network (SMEConvNet) for automatically spotting apex frame from neutral expression frames and extracting features from the apex frame. Thirdly, we use a sliding window over feature matrix for locating apex frame.

The remaining parts of this paper are organized as follows: The related work is given in Section 2; the proposed method is explained in Section 3; experimental results and discussions for micro-expression spotting are presented in Section 4; the conclusion is given in Section 5.

II. RELATED WORK

Automatic facial micro-expression analysis has attracted increasing attention in recent years. However, only a few studies have focused on automatic micro-expressions spotting.

Shreve et al [17, 18] used an optical flow method [19] to compute the optical strain magnitude to spot both macro and micro-expression. Polikovsky et al [20, 21] employed 3D gradient histograms as feature to recognize the onset, apex, and offset of micro-expression. These methods could inspire the research community for developing new micro-expression spotting methods. However, they were only tested on posed micro-expression data. For the recording of posed micro-expression videos, participants will control their behavior according to the instructions. Therefore, posed data may exclude unwanted head movement and more clear-cut onset and offset, which makes the task of spotting posed data easier than that of spontaneous data.

Yan et al [15] used Constraint Local Model (CLM) [22] and Local Binary Pattern (LBP) [23] as feature extractors for searching the apex frame from spontaneous micro-expression video. The apex frame is located at the frame that has the largest feature difference in comparison with the first frame. Liong et al [16] employed optical flow [17] as feature extractor for locating apex frame from *short video*.

Moilanen et al [24] used LBP histogram to obtain temporal locations and spatial locations for micro-expression

spotting. The appearance-based features between average frame and current frame were used for the spotting.

Li et al [25] proposed an approach based on deep multi-task learning with Histograms of Oriented Optical Flow (HOOF) feature for micro-expression detection. However, they used CNN to detect the facial landmark localization and spilt the facial area into regions of interest, which is only the pre-processing stage of the micro-expression data. Liong et al [13] introduced an automatic approach to micro-expression analysis from *long video* that combines both spotting and recognition methods. The apex frame in a *long video* sequence was identified by applying optical strain feature extractor after eye masking and regions of interest selection techniques.

III. MATERIALS AND METHOD

A. DATABASE

A long video database is required for this research where the index of apex frame should be known in order to train the model and evaluate the performance. There are only two publicly available databases meeting the requirements, which are CASME-RAW [26] and CASMEII-RAW [14]. However, the data in CASME-RAW was obtained in the environment where illumination light may be flicking. The flicking light could produce noisy and dark video clips [14]. Therefore, the CASMEII-RAW was the only suitable one as we focus on the situation where the intensity of illumination source is constant.

There are spontaneous micro-expression video clips from 26 subjects in the CASMEII-RAW. The average age of the subjects is 22.03 years. The spatial resolution is 640×480 pixels and the frame rate is 200 fps. There are five categories of micro-expression in the database. The ground-truths provided include the onset, apex, offset frame indexes.

The database is randomly divided into two parts to simulate cross-database evaluation [27]. The first part consists of 150 *long video* clips, which are used as training set. The second part consists of 97 *long video* clips, which are employed as testing set. In order to avoid fortuity interference, we used five-fold cross-validation in the training set. Therefore, the training set of the database was randomly divided into five sub-sections, each of which contains 30 video clips. In each fold, the proposed network was trained on the four sub-sections (120 *long* video clips) and validated on the rest sub-section (30 *long* video clips). After the network was determined on the training set, it was then tested on the testing set (97 *long* video clips). After five fold, each of the five sub-sections was used once as a validation set, which produces five spotting results shown in Table V in Section IV. The final spotting result are given as the average of these five spotting results.

B. METHOD OUTLINE

The proposed spotting method consists of three steps that are outlined in Fig. 2. They are: (1) pre-processing the *long video* to obtain aligned and cropped video; (2) extracting high level features from the processed *long video* by using SMEConvNet; (3) Processing the feature matrixes further using a sliding window.

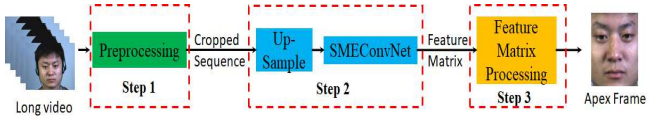

FIGURE 2. Outline of the proposed micro-expression spotting method

C. PRE-PROCESSING

Before applying spotting algorithms, we carried out pre-processing [28, 14] on the raw sample clips, which had three steps. Let R be the set of micro-expression clips:

$$R=\left[r_i \middle| r\in R, i=1,2\cdots n\right] \quad (1),$$

$$r_i=\left[f_{i,j} \middle| f\in r_i, j=1,2\cdots l_i\right] \quad (2).$$

The sample r_i represents the *i*-th micro-expression video clip, where l_i is the frame number of r_i, $f_{i,j}$ is the *j*-th frame of the sample r_i.

Firstly, a frontal face with neutral expression M was selected as the model face. Two inner eye corners and a spine landmark point of M were detected by the robust detector Discriminative Response Map Fitting (DRMF) [29], these three points are $\psi(M)$.

Secondly, the first frame $f_{i,1}$ of the micro-expression sequence r_i was transformed into the model face by using a non-reflective similarity transformation (NST) [14] to achieve the face alignment. The transform matrix T is represented as Equ. 3:

$$T_i=NST\left(\psi(M),\psi\left(f_{i,1}\right)\right), i=1,2\cdots n \quad (3),$$

where $\psi\left(f_{i,1}\right)$ is the coordinates of inner eye corners and nasal spine point of $f_{i,1}$. Then all frames of r_i were transformed by using T_i. The main reason why we detected landmark points only on the first frame but not on all frames is that the landmark points detected by DRMF might not be accurate enough. The transformed image f^T was computed by Equ. 4:

$$f_{i,j}^T=T_i\times f_{i,j}, j=1,2\cdots l_i \quad (4).$$

Thirdly, the inner eye corners and nasal spine point coordinates U_i of the first frame of each transformed micro-expression sequence $f_{i,1}^T$ were detected by DRMF, and then

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2879485, IEEE Access

the face of r_i^T each frame of was cropped out by using a rectangle determined by U_i. Each cropped face was then resized to 224×224 pixels. Fig. 3 illustrates the process of the preprocessing.

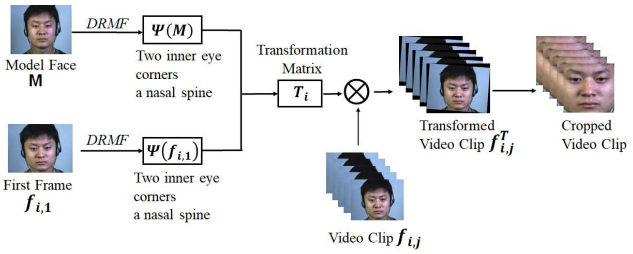

FIGURE 3. An illustration of process of pre-processing

D. UP-SAMPLEING

In order to train a CNN with the ability to distinguish apex frame from neutral expression frames, a large data is needed. In a *long video* including one *short video*, some neutral expression frames can be selected for the training. However, there is only one apex frame in the *long video*, which is far too few for the training.

Due to the frame rate of the database is high, there is little difference between the apex frame and its nearby frames. Therefore, we replace these frames within the *short video* with the apex frame. So that there are enough apex frames in the *long video*. We treated the apex frame as a positive sample and the neutral expression as a negative sample. With positive and negative samples, therefore, we can train the CNN. The up-sampling of apex frame is illustrated in Fig. 4.

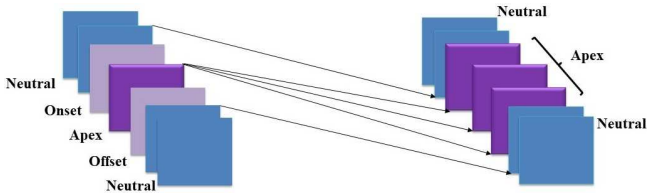

FIGURE 4. The illustration of up-sampling apex frame

E. SMEConvNet

In CNN design, convolutional and pooling layer pairs are stacked and then followed by fully connected (FC) layer at the end [30, 31, 32]. In design of the SMEConvNet, we followed this widely used structure. Different from normal design (only considering one FC layer), the SMEConvNet has three FC layers, which is inspired by the structure of AlexNet [33]. According to previous research work [34, 35, 36, 37], a deep neural network is not suitable for medium size dataset. Therefore, we design the SMEConvNet where medium size dataset can be used for training.

The number of convolutional and pooling layer pairs was determined experimentally using different numbers of pairs (experiment results are given in Section 4.1). The performance of the network increased as the number of pairs increased, and reached the best when the number was four. The reason might be that the four convolutional and pooling layer pairs is the most suitable for the size of the database, i.e. the less number of pairs cannot extract high enough abstract features, the more number of pairs will be redundant for the dataset so that the performance becomes worse.

Therefore, the proposed SMEConvNet has four convolutional and pooling layer pairs and three FC layers. The first two FC layers have 500 channels each, and the third layer (Fc7) performs apex frame and neutral expression classification that contains 2 channels (one for each class). The structure of the SMEConvNet is shown in Fig. 5 and the detailed information on configuration is given in Table I.

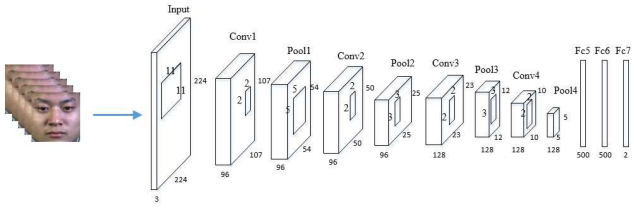

FIGURE 5. The structure of SMEConvNet

TABLE I: CONFIGURATION OF SMECONVNET

Layer	Kernel Parameter	Output Size
Data		224×224×3
Conv 1	K=11 S=2	96×107×107
Pool 1	K=2 S=2	96×54×54
Conv 2	K=5 S=1	96×50×50
Pool 2	K=2 S=2	96×25×25
Conv 3	K=3 S=1	128×23×23
Pool 3	K=2 S=2	128×12×12
Conv 4	K=3 S=1	128×10×10
Pool 4	K=2 S=2	128×5×5
Fc 5		500
Fc 6		500
Fc 7		2

During the training, the network weight parameters are learned using mini-batch stochastic gradient descent with momentum of 0.9. Each 64 images batch is sent to the CNN with weight decay 0.005. The base learning rate is 10^{-4} and the value is further dropped when the loss stops changing. The network iterates 30 epochs in each fold.

F. FEATURE MATRIX PROCESSING

The output of the Fc6 layer of SMEConvNet is a feature matrix (F) with dimension $X \times Y$, where X is the frame number of the input *long video* and Y is the number of features in the Fc6 layer in SMEConvNet. Each row of the feature matrix corresponds to a frame in the *long video*.

The feature matrix F was further processed into two more matrixes. The steps and method used for the processing are illustrated in Fig6.

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2879485, IEEE Access

The first frame of Matrix F was chosen as the reference frame. Therefore, the elements of each row of Matrix F were made difference with the corresponding elements in the first row of Matrix F. Then each difference value was squared. Finally, all the square values were summed in each row to obtain Matrix A with dimension $X \times 1$ (shown in Fig. 6). The j-th element of Matrix A was calculated by using

$$A_j = \sum_{n=1}^{Y} (F_{jn} - F_{1n})^2 \quad (5),$$

where F_{jn} is the element of matrix F.

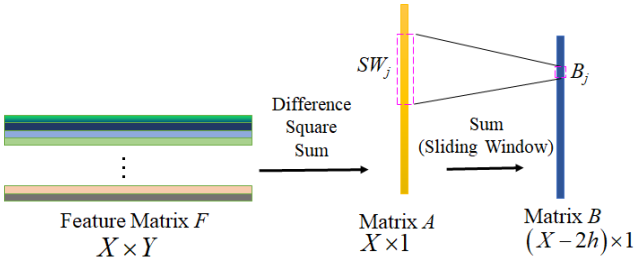

FIGURE 6. Illustration of feature matrix processing

SW_j: the j-th sliding window with L length covering part of Matrix A

B_j: the j-th element in Matrix B

The Matrix A was further processed by using a sliding window (SW). The length of the sliding window is L (an odd number), where:

$$L = 2h + 1 \quad (6).$$

The L was set as the average duration of micro-expression. Because CASME2-RAW has a frame rate of 200 fps. The average duration of micro-expressions is 0.335s corresponding to frame length of 67. We made L as 67 (h =33).

All the values in the sliding window (SW) were summed to produce one value as shown in Fig 6. After the sliding window processing, the Matrix B with dimension of $(X-2h) \times 1$ was obtained (shown in Fig. 6). The j-th element of matrix B was calculated by using:

$$B_j = SW_j(A) = \sum_{p=j-h}^{j+h} A_p \quad (7),$$

where SW_j is the j-th sliding window for generating B_j and A_p is the p-th element of Matrix A.

To locate the apex frame, the largest element in matrix B was found and the sliding window (SWL) producing this largest value was firstly located by using:

$$SWL = SW_j, \quad if \quad SW_j(A) = Max(B) \quad (8),$$

where $Max(B)$ is the largest element in matrix B

Then the largest value in SWL was found, and the index of this largest value in Matrix A was located. Suppose that the largest value is the m-th element in Matrix A:

$$A_m = Max(SWL) \quad (9).$$

Finally, the apex frame is located as the frame corresponding to this largest value, i.e. the apex is the m-th frame in the *long video*.

Fig. 7 gives two examples to illustrate the sliding window method. The central frame in the sliding window is current frame (CF), the frame h-th frame ahead of the CF is tail frame (TF), and h-th frame after the CF is head frame (HF).

In the first example shown in Fig. 7 (a), the CF is the 35th frame, and TF is the second frame of the *long video*. The sum produced by this sliding window is small since the window contains no or only small part of *short video*. In the second example shown in Fig. 7 (b), the CF is located near to the apex frame. The sum produced by this sliding window should be much larger than the value in the first example, since the window covers much more *short videos*. Therefore, the more the sliding window covers *short video*, the larger the value is in the Matrix B.

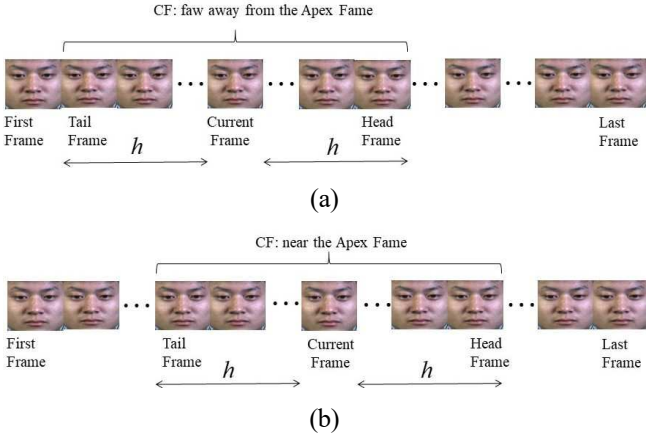

FIGURE 7. (a) CF is far from the apex frame, (b) CF is close to the apex frame

When the sliding window coincides with the *short video*, it produces the maximum value in the Matrix B. Therefore, the apex frame should be located in the sliding window that has maximum value in Matrix B. More discussion about the effectiveness of this feature matrix processing method will be given in the next section.

IV. RESULTS AND DISSCUSSION

A. COMPARISION OF VARIOUS STRUCTURES AND PARAMETERS

The experimental results under different CNN structures are outlined in Table II, one per column. The AlexNet is also listed for comparison. Different network structure are denoted by A, B, C, D, and E. During training, the input to CNN is a fixed-size 224×224 RGB image. Three fully-connected layers are in the end of every CNN: the first two have 500 channels each, the third performs 2-way classification and contain 2 channels (one for each class). The final layer of every network is a soft-max layer. All networks have the same configuration of the fully connected layers. All hidden layers are equipped with the Rectified Linear Unit (RELU) function.

TABLE II: VARIOUS CNN STRUCTURES. THE DEPTH OF THE NETWORK INCREASES FROM LEFT (A) TO THE RIGHT (D). THE CONVOLUTIONAL LAYER IS DENOTED AS 'CONV'. THE RELU ACTIVATION FUNCTION IS NOT SHOWN FOR BREVITY

CNN Configuration					
A	B	C	D	E	AlexNet
Input (224×224 image)					
Conv	Conv	Conv	Conv	Conv	Conv
Pool					
Conv	Conv	Conv	Conv	Conv	Conv
Pool					
	Conv	Conv	Conv	Conv	Conv
	Pool				Conv
		Conv	Conv	Conv	Conv
		Pool			Pool
			Conv		
			Pool		
Fc-500					Fc-4096
Fc-500					Fc-4096
Fc-2					
Softmax					

Table III shows the configuration of each network. Each column of Table III specifies the size of kernel ('K') and stride ('S'), and the number of kernels ('N'). The padding value was set as zero in each layer and not shown in the table. The CNN depth increased from A to D. The network E has the same structure as the network C. However, network E has more kernels.

In all networks, we use kernel size of 11 (K=11) at the first convolutional layer. This kernel size is much larger than that of rest layers and will increase the computational complexity of the network. However, many researches [38, 39] demonstrated that large kernel size can cover more part of certain important facial regions, such as eye and mouth region, and thus improve the performance of facial expression recognition.

In fully-connected layers (Fc), 500 is the number of the feature length and 2 is the number of class (apex frame or non-apex frame). The feature dimensionality is only 500 in order to reduce the number of parameters in the model, and prevents over-fitting.

TABLE III. CNNS CONFIGURATION.

Net structure	Conv	Pool	Conv	Pool	Conv	Pool	Conv	Pool	Conv	Pool	Fc	Fc	Fc
A	K=11 S=2 N=96	K=2 S=2	K=5 S=1 N=96	K=2 S=2							500	500	2
B					K=3 S=1 N=128	K=2 S=2							
C							K=3 S=1 N=128	K=2 S=2					
D									K=3 S=1 N=128	K=2 S=2			
E	K=11 S=2 N=96	K=2 S=2	K=5 S=1 N=256	K=2 S=2	K=3 S=1 N=384	K=2 S=2	K=3 S=1 N=384	K=2 S=2					
Net structure	**Conv**	**Pool**	**Conv**	**Pool**	**Conv**	**Conv**	**Conv**	**Pool**			**Fc**	**Fc**	
AlexNet	K=11 S=4 N=96	K=3 S=2	K=5 S=2 N=256	K=3 S=2	K=3 S=1 N=384	K=3 S=1 N=384	K=3 S=1 N=256	K=3 S=2			4096	4096	

Table IV shows the accuracy of classifying apex frame and non-apex frame. From the accuracy achieved by network A to D, we can see that the accuracy increases as the network depth increase, reaches maximum as the CNN (Network C) has 4 convolutional-and-pooling pairs, and then begins to decrease from Network D that has 5 convolutional-and-pooling pairs. Therefore, we chose Network C as the SMEConvNet.

The Network E has the same structure as the Network C (see Table III), but E has more convolutional kernels, which means E extracted more features. Compared with the accuracy that C archives (71.97%), E can only achieve accuracy of 68.19%. More kernels in E means more features extracted. This result indicates that more feature maps may led to decrease in accuracy rate and suggesting that the width of the network should also be considered when design the CNN.

It is also seen that the proposed networks (A-E) has better performance than AlexNet. This results support the point that properly sized network is likely to achieve better results [35, 36].

TABLE IV. THE ACCURACY OF CLASSIFYING APEX FRAME AND NON-APEX FRAME

	Fold 1 (%)	Fold 2 (%)	Fold 3 (%)	Fold 4 (%)	Fold 5 (%)	Average Value (%)
A	67.38	68.23	65.15	68.47	65.27	**66.94**
B	70.16	68.31	67.04	71.79	68.59	**69.18**
C	72.06	72.70	68.86	76.91	69.32	**71.97**
D	63.05	64.16	64.28	68.57	61.75	**64.36**
E	68.18	68.69	66.25	69.37	68.44	**68.19**
AlexNet	59.36	63.25	58.07	65.81	61.65	**61.62**

B. SPOTTING APEX FRAME

1) COMPARISON OF DIFFERENT FEATURE MATRIX PROCESSING METHODS

The effectiveness of apex frame spotting is evaluated by using the Mean Absolute Error (MAE) and Apex Spotting Rate (ASR) [13]. MAE is the average frame distance between the ground truth and the spotted apex frame, and can be computed by using:

$$MAE = \frac{1}{M}\sum_{j=1}^{M}\left|e_j\right| \tag{10}$$

where M is the total number of video sequence, and e is the distance (in frames) between the ground-truth apex and the spotted apex. The ASR is the success rate in spotting apex frame within the *short video*. If the spotted apex frame is within the *short video*, the spotting is successful. The ASR can be computed by using:

$$ASR = \frac{1}{M}\sum_{j=1}^{M}\alpha \tag{11}$$

$$\alpha = \begin{cases} 1 & if\ f^* \in \left(f_{j,onset}, f_{j,offset}\right) \\ 0 & otherwise \end{cases} \tag{12}$$

The algorithm described in section 3.6 was used to spot apex frame from *long video* sequences. In order to show the advantage of the proposed method, we compared three methods or the spotting, i.e. Maximum Frame (Max), Sliding Window Current Frame (SW-CF), and Sliding Window Maximum Frame (SW-Max) method.

The Max method regards the frame that has the maximum value in Matrix A (see Fig. 6) as the apex frame.

The SW-CF method finds the sliding window that produces the largest values in Matrix B (see Fig. 6) and takes the current frame (central frame) of the sliding window as the apex frame.

The SW-Max method finds the sliding window that produces the largest values in Matrix B (see Fig. 6) and takes the frame that has maximum value in Matrix A within the sliding window as the apex frame. This method is the method used in the paper.

Table V gives the experimental results. It is observed that the sliding window based method (SW-CF and SW-Max) can achieve better performance (smaller MAE and larger ASR) than what the Max method can achieve in each fold of test. The maximum value in matrix A should correspond to the apex frame in normal case, because the apex frame is the instant when the micro-expression reaches its climax (the most intense movement). However, in the case where irrelevant movement, such as eye blinking and head movement, appears outside of the *short video*, the maximum value in Matrix A may correspond to the irrelevant movement. In this situation, the Max method will produce wrong result. However, the sliding window based methods take full advantage of the characteristics that micro-expressions last a period of time, to locate the period, in which apex frame locates, and then locate the apex frame. Thus the sliding window based methods are more robust than Max method when irrelevant movements appear outside the *short video*.

It is also seen from Table V that SW-Max can achieve better performance than what SW-CF can achieve in each fold of test. The average MAE of SW-Max (22.36) is 4 frames smaller than that of SW-CF (26.55), and ASR of SW-Max (0.8280) are 0.03 smaller than that of SW-CF (0.7932). This indicates that SW-Max is a better method than SW-CF for micro-expression spotting.

The reason why SW-Max is better than SW-CF is that the apex frame is not always located in the center of sliding window. Shown in Fig. 8 is the potting of the Matrix A of a subject with vertical axis as the matrix element value. The *short video* range is from the 71st frame to the 161st frame shown as black rectangular. The ground-truth apex is the 91st frame shown as magenta triangle dot. When the sliding window (shown as yellow rectangle) is at 80th to 146th frame, the sum produced by the window is the largest. In this case, SW-CF takes the center point of the sliding window, i.e. 133rd frame (shown as red triangle dot), as the apex frame. However, SW-Max did not simply use the central frame of the sliding window, it search the maximum frame in the sliding window again to locate the apex frame, and thus takes the 87th frame (shown as green triangle dot) as the apex frame, which is much closer to the ground-truth apex frame.

One of reasons why SW-Max is better than Max is that Max tends to find other irrelevant facial movements, such as eye blinking. In the Fig 8, the Max method regards the 152nd frame as the apex frame (shown as blue triangle dot), which is far from the ground-truth apex. The 152nd frame is the peak of a spike in Fig 8. This spike is a result of eye blinking, which has even higher value than the ground-truth apex but much narrower width. By using SW-Max method, the sliding window with proper width can avoid locating this spike as micro-expression.

TABLE V. THE EXPERIMENTAL RESULTS OF SPOTTING APEX FRAME.

	Fold 1		Fold 2		Fold 3		Fold 4		Fold 5		The Average Value	
	MAE	ASR	MAE	ASR	MAE	ASR	MAE	ASR	MAE	ASR	MAE	ASR
Max	33.05	0.6186	31.18	0.7010	34.59	0.6392	30.25	0.7293	33.52	0.6495	**32.51**	**0.6675**
SW-CF	26.35	0.7835	25.55	0.8023	29.04	0.7707	25.36	0.8188	26.48	0.7910	**26.55**	**0.7932**
SW-Max	22.41	0.8232	22.47	0.8388	23.64	0.8072	21.24	0.8441	22.08	0.8267	**22.36**	**0.8280**

This article has been accepted for publication in a future issue of this journal, but has not been fully edited. Content may change prior to final publication. Citation information: DOI 10.1109/ACCESS.2018.2879485, IEEE Access

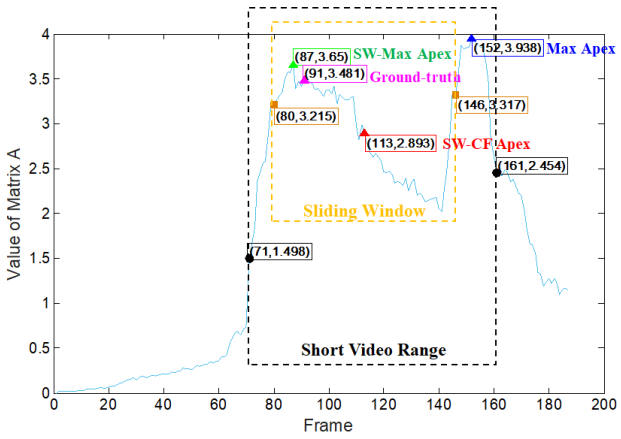

FIGURE 8. The plotting of matrix A

***Short Video* Range: from 71st to 161st frame shown as black rectangular**

Sliding Window: from 80th to 146th frame shown as yellow rectangular, which produces the largest sum

Ground-truth: the ground-truth apex frame, the 91st frame shown as magenta triangle dot

SW-Max Apex: the apex frame predicted by SW-Max method, the 87th frame shown as green triangle dot

SW-CF Apex: the apex frame predicted by SW-Max method, the 113rd frame shown as green triangle dot

Max Apex: the apex frame predicted by Max method, the 152nd frame shown as green triangle dot

2) INDEPENDENT OF FEATURE EXTRACTION METHOD

To further illustrate the SW-Max method is feature-extraction-method-independent. We used LBP to extract features of *long video* sequences, and then used Max, SW-CF, and SW-Max to spot apex frame. Table VI shows the result of LBP method. Again, the sliding window based method is better than Max method, and SW-Max method is better than SW-CF method. These results suggest that the proposed SW-Max method is independent of feature extraction method.

TABLE VI. THE RESULT OF SPOTTING APEX FRAME USING LBP

	MAE	ASR
Max	50.42	0.5361
SW-CF	30.75	0.7423
SW-Max	25.80	0.7838

3) INDEPENDENT OF FEATURE EXTRACTION METHOD

The spotting results by using proposed method (CNN+SW-Max) are also compared to a traditional method (LBP) and a state-of-art method [13]. The method in [13] is the only method that we can find and that has been recently developed for spotting apex frame from *long video* due to the few research in this area.

The experimental results (average MAE and ASR) are summarize in Table VII. It was observed that the method in [13] can achieve higher ASR (0.8230) than that (0.7838) of LBP-SW-Max, but larger MAE (27.21) than that (25.80) of LBP-SW-Max. This indicates that [13] has higher successful rate of locating apex frame within the *short video* range, but less successful with respect to locating apex frame close to the ground-truth apex frame.

In all three methods, the proposed method (CNN+SW-Max) has the highest ASR (0.8280) and smallest MAE (22.36). Compared with method [13], the CNN+SW-Max has 5 frames smaller MAE that is 18.5% smaller than that of method [13]. Compared with LBP+SW-Max, the CNN+SW-Max has 4% higher ASR and 3 frame smaller MAE that is 12% smaller than that of LBP+SW-Max). This may suggest that the CNN+SW-Max is a good method for locating apex frame in terms of both locating apex frame within the *short video* range and locating apex frame close to the ground-truth apex frame.

TABLE VII. COMPARISON OF DIFFERENT METHODS FOR SPOTTING APEX FRAME FROM *LONG VIDEO*

	MAE	ASR
LBP+Max	50.42	0.5361
LBP+SW-CF	30.75	0.7423
LBP+SW-Max	25.80	0.7838
CNN+Max	32.51	0.6675
CNN+SW-CF	26.55	0.7932
CNN+SW-Max	**22.36**	**0.8280**
[13]	27.21	0.8230

V. CONCLUSION

Micro-expression spotting is a primary step for micro-expression recognition. In this paper, we proposed a new method for spotting micro-expression from *long video*. A convolutional neural network named as SMEConvNet was designed. This is the first time that deep learning technique was used in micro-expression spotting. The SMEConvNet has four convolutional and pooling layer pairs followed by three fully connected layers. The number of feature extracted by using the SMEConvNet from each frame is 500. A feature matrix can then be built from a *long video*. Then, a feature matrix processing method was proposed. The feature matrix was processed by using difference, squared, and sum operation firstly, and then was manipulated by a sliding window. The sliding window producing the largest value was located, and then the maximum value within the sliding window was located as the apex frame. By combing the proposed feature extraction method (CNN) and feature matrix processing method (SW-Max), the proposed method can achieve higher ASR (0.8280) and smaller MAE (22.36) than LBP+SW-Max and state-of-art method [13].

REFERENCES

[1] A. Ucar, Y. Demir, and C. Güzelis, "A new facial expression recognition based on curvelet transform and online sequential extreme

learning machine initialized with spherical clustering," Neural Computing and Applications, vol. 27, no. 1, pp. 131-142, Jan. 2016.

[2] C. A. Corneanu, M. O. Simon, J. F. Cohn, and S. E. Guerrero, "Survey on RGB, 3D, Thermal, and Multimodal Approaches for Facial Expression Recognition: History, Trends, and Affect-related Applications," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 38, no. 8, pp. 1548-1568, Jan. 2016.

[3] G. Y. Zhao, X. H. Huang, M. Taini, S. T. Li, and M. Pietikainen, "Facial expression recognition from near-infrared videos," Image and Vision Computing, vol. 29, no. 9, pp. 607-619, Aug. 2011.

[4] S. J. Wang, H. L. Chen, W. J. Yan, X. and L. Fu, "Face Recognition and Micro-expression Recognition Based on Discriminant Tensor Subspace Analysis Plus Extreme Learning Machine," Neural Processing Letters, vol. 39, no. 1, pp. 25-43, Feb. 2014.

[5] X. H. Huang, G. Y. Zhao, X. P. Hong, W. M. Zheng, and M. Pietikainen, "Spontaneous facial micro-expression analysis using Spatiotemporal Completed Local Quantized Patterns," Neurocomputing, vol. 175, no. PA, pp. 564-578, Oct. 2015.

[6] F. Xu, J. P. Zhang, and J. Z. Wang, "Microexpression Identification and Categorization Using a Facial Dynamics Map," IEEE Transactions on Affective Computing, vol. 8, no. 2, pp. 254-267, May. 2017.

[7] P. Ekman, "Darwin, deception, and facial expression," Ann N Y Acad Sci, vol. 1000, no. 1, pp. 205-221, Dec. 2003.

[8] P. Ekman, and W. V. Friesen, "Nonverbal leakage and clues to deception," Psychiatry-interpersonal and Biological Processes, vol. 32, no. 1, pp. 88-106, 1969.

[9] W. J. Wen, Q. Wu, J. Liang, Y. H. Chen, and X. L. Fu, "How Fast are the Leaked Facial Expressions: The Duration of Micro-Expressions," Journal of Nonverbal Behavior, vol. 37, no. 4, pp. 217-230, Jul. 2013.

[10] S. Porter, and B. L. Ten, "Reading between the lies: identifying concealed and falsified emotions in universal facial expressions," Psychological Science, vol. 19, no. 5, pp. 508-514, May. 2008.

[11] P. Ekman, "Lie catching and microexpressions," in The Philosophy of Detection, ed C. W. Martin, pp. 118-136.

[12] S. Weinberger, "Airport security: Intent to deceive," Nature, vol. 465, no. 7297, pp. 412-415, May. 2010.

[13] S. T. Liong, J. See, K. S. Wong, and C. W. Phan, "Automatic Micro-expression Recognition from Long Video Using a Single Spotted Apex," in. *Proc. Adv. Asian. Conf. Comput. Vision.*, Mar, 2017, pp. 345-360.

[14] W. J. Yan, X. B. Li, S. J. Wang, and X. L. Fu, "CASME II: An Improved Spontaneous Micro-Expression Database and the Baseline Evaluation," Plos One, vol. 9, no. 1, Jan. 2014.

[15] W. J. Yan, S. J. Wang, Y. H. Chen, G. Y. Zhao, and X. L. Fu, "Quantifying Micro-expressions with Constraint Local Model and Local Binary Pattern," in. *Proc. Adv. European. Conf. on Comput. Vision,* Mar. 2014, pp. 296-305.

[16] S. T. Liong, J. See J, K. S. Wong et al, "Automatic apex frame spotting in micro-expression database," in *Proc. IEEE Conf. Pattern Recognin,.* Nov, 2016, pp. 3-6.

[17] M. Shreve, S. Godavarthy, V. Manohar, and D. Goldgof, "Towards macro- and micro-expression spotting in video using strain patterns," in *Proc. IEEE Conf. Applications of Comput Vision.*, July, 2009, pp. 1-6.

[18] M. Shreve, S. Godavarthy, D. Goldgof, and S. Sarker, "Macro- and micro-expression spotting in long videos using spatio-temporal strain," in *Proc. IEEE Conf. Auto Face and Gesture Recognit.*, Apr. 2011, pp. 51-56.

[19] M. J. Black, and P. Anandan, "The Robust Estimation of Multiple Motions: Parametric and Piecewise-Smooth Flow Fields," Computer Vision and Image Understanding, vol. 63, no. 1, pp. 75-104, Jan. 1996.

[20] S, Polikovsky, Y. Kameda, and Y. Ohta, "Facial micro-expressions recognition using high speed camera and 3D-gradient descriptor," in *proc. Adv. Crime. Detect. and. Prevent,.* 2010, pp. 1-6.

[21] S. Polikovsky, Y. Kameda, and Y. Ohta, "Facial Micro-Expression Detection in Hi-Speed Video Based on Facial Action Coding System (FACS)," IEICE Transactions on Information and Systems, vol. E96-D, no. 1, pp. 81-92, Jan. 2013.

[22] D. Cristinacce, and T. Cootes, "Automatic feature localisation with constrained local models," Pattern Recognition, vol. 41, no. 10, pp. 3054-3076, Oct. 2008.

[23] T. Ojala, M. Pietikainen, and D. Harwood I, "A Comparative Study of Texture Measures with Classification Based on Feature Distributions," Pattern Recognition, vol. 29, no. 1, pp. 51-59, Jan. 1996.

[24] A. Moilanen, G. Y. Zhao, M. Pietikainen, "Spotting Rapid Facial Movements from Videos Using Appearance-Based Feature Difference Analysis," in. *Proc. Adv. Inter. Conf. on Pattern. Recognit,* Aug. 2014, pp. 296-305.

[25] X. Li, J. Yu, S. Zhan, "Spontaneous facial micro-expression detection based on deep learning," in *Proc. IEEE Conf. Signal Process.*, Mar, 2017, pp. 1130-1134.

[26] W. J. Yan, Q. Wu, Y. J. Liu, S. J. Wang, and X. L. Fu, "CASME database: A dataset of spontaneous micro-expressions collected from neutralized faces," in Proc. IEEE Conf. Auto Face and Gesture Recognit., Apr, 2013, pp. 1-7.

[27] Oh, Yee-Hui, et al. "A Survey of Automatic Facial Micro-expression Analysis: Databases, Methods and Challenges," *arXiv preprint arXiv:1806.05781,* 2018.

[28] S. J. Wang, S. Wu, and X. L. Fu, "A main directional maximal difference analysis for spotting micro-expressions," in. *Proc. Adv. Asian. Conf. on Comput. Vision,* Mar. 2016, pp. 449-461.

[29] A. Asthana, S. Zafeiriou, S. Cheng, and M. Panetic, "Robust Discriminative Response Map Fitting with Constrained Local Models," in. *Proc. Adv. Comput. Vision. and Pattern. Recognit,* Jun. 2013, pp. 3444-3451.

[30] Y. Lécun, L. Bottou, Y. Bengio, H. Patrick, "Gradient-based learning applied to document recognition," Proceedings of the IEEE, vol. 86, no. 11, Dec. 1998.

[31] K. Simonyan, and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," *arXiv preprint arXiv:1409.1556,* 2014.

[32] M. D. Zeile, and R. Fergus, "Visualizing and Understanding Convolutional Networks," in Proc. Adv. European. Conf. Comput. Vision., 2014, pp. 818-833.

[33] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet classification with deep convolutional neural networks," in *proc. Adv. Neural Inf. Process. Syst.*, 2012, pp. 1097-1105.

[34] M. Peng, C. Y. Wang C, Chen T, G. Y. Liu, X. L. Fu, "Dual Temporal Scale Convolutional Neural Network for Micro-Expression Recognition," Frontiers in Psychology, 8:1745, Oct. 2017.

[35] M. Peng, C. Y. Wang, T. Chen, G. Y, Liu, "Nirfacenet: A convolutional neural network for near-infrared face identification," Information, vol.7, no. 4, Oct. 2016.

[36] Wu, Zhan, Min Peng, and Tong Chen, "Thermal face recognition using convolutional neural network," in *Proc. IEEE Conf. Optoelectronics and Image Process.,* June, 2016, pp. 6-9.

[37] Z. Wu, T. Chen, Y. Chen, Z. H. Zhang, and G. Y. Liu, "NIRExpNet: Three-Stream 3D Convolutional Neural Network for Near Infrared Facial Expression Recognition," Applied Sciences, vol. 7, no. 11, pp. 1184, Nov, 2017.

[38] A. Mollahosseini, D. Chan, and M. H. Mahoo, "Going deeper in facial expression recognition using deep neural networks," in *Proc. IEEE Conf. Applications of Comput Vision.*, Nov. 2016, pp. 1-10.

[39] L. Wang, R. F. Li, K. Wang, and J. Chen, "Feature Representation for Facial Expression Recognition Based on FACS and LBP," International Journal of Automation and Computing, vol. 11, no. 5, pp. 459-468, Oct. 2014.