

$$(x_{l_1}, y_{l_1}) = \left(\frac{x_1^M + x_2^M}{2}, \frac{y_1^M + y_2^M}{2} \right) (3)$$

Where (x_1^{RE}, y_1^{RE}) and (x_3^{RE}, y_3^{RE}) are the coordinates of first and third corners of right eye landmark region, (x_2^{LE}, y_2^{LE}) and (x_4^{LE}, y_4^{LE}) are the coordinates of second and fourth corners of left eye landmark region, and (x_1^M, y_1^M) and (x_2^M, y_2^M) are the coordinates of first and second corners of mouth landmark region. Based on these three landmark points, we perform face alignment. The face alignment removes the influence of head postures over the micro expression recognition. Based on the coordinates of right eye corner and left eye inner corner, we compute a rotation matrix R as follows;

$$R = \begin{bmatrix} R_1 & R_2 \\ R_3 & R_4 \end{bmatrix} (4)$$

Where

$$R_1 = \frac{x_{l_2} - x_{l_1}}{\sqrt{(y_{l_2} - y_{l_1})^2 + (x_{l_2} - x_{l_1})^2}} (5)$$

$$R_2 = \frac{y_{l_1} - y_{l_2}}{\sqrt{(y_{l_2} - y_{l_1})^2 + (x_{l_2} - x_{l_1})^2}} (6)$$

$$R_3 = \frac{y_{l_2} - y_{l_1}}{\sqrt{(y_{l_2} - y_{l_1})^2 + (x_{l_2} - x_{l_1})^2}} (7)$$

$$R_4 = \frac{x_{l_2} - x_{l_1}}{\sqrt{(y_{l_2} - y_{l_1})^2 + (x_{l_2} - x_{l_1})^2}} (8)$$

After the computation of rotation matrix R, the in-plane rotation and facial size variations within the facial region are varied as

$$(x'_{l_i}, y'_{l_i}) = (x_{l_i}, y_{l_i}) * R^T (9)$$

After facial alignment, the new coordinates of three landmark points are obtained as (x'_{l_1}, y'_{l_1}) , (x'_{l_2}, y'_{l_2}) and (x'_{l_3}, y'_{l_3}) of Right eye inner corner, left eye inner corner and nasal spine point respectively. Based on these new coordinates we measure the size of each block into which the facial region has to divide. The size (i.e., width and height) of each block is computed as follows;

$$w = \frac{(x'_{l_2} - x'_{l_1})}{2} (10)$$

And

$$h = \frac{(y'_{l_3} - y'_{l_1})}{2} (11)$$

Further, the starting point of each block is computed as

$$SP = (2x'_{l_1} - x'_{l_2}, 2y'_{l_1} - y'_{l_3}) (12)$$

Based on the width, height and starting point, the entire facial region is divided into equal sized blocks. Figure.3 shows a simple demonstration about the process of facial alignment.

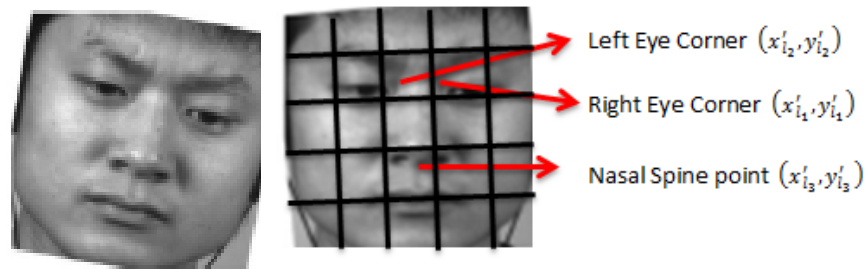


Figure.3 Facial Alignment & block division

3.4 Feature Extraction

Once the facial alignment and partitioning is completed, then we apply feature extraction over every block through Local Binary Pattern (LBP) based method. LBP was first introduced in the mid of 1990's [] and has been successfully applied in different fields of computer vision like human action recognition, object detection and texture analysis. Further, it was introduced into the face recognition and classification of micro expressions []. According to the standard methodology, the LBP computation is as follows; for each pixel in the frame, it is compared with its P neighbor pixels along a circle. During this comparison, if the gray value of neighbor pixel is greater than or equal to the gray value of center pixel, then the output is 1 otherwise the output is 0. This process derives a P -point number that can be represented as a decimal number by weighing with the power of two. Based on this we can state that the LBP characterizes the structures of P pixels, those were distributed evenly in angle over a circle of radius r and centered at pixel q_c . For a center pixel q_c and its P neighbor pixels $\{q_{r,p,n}\}_{n=0}^{P-1}$ on the circle of radius r , the LBP pattern is computed as

$$LBP_{r,p}(q_c) = \sum_{n=0}^{P-1} s(q_{r,p,n} - q_c) 2^n \quad (13)$$

Where

$$s(x) = \begin{cases} 1, & x \geq 0 \\ 0, & x < 0 \end{cases} \quad (14)$$

Where $s(\cdot)$ is a sign function. If the pixels are not observed to be at the center, then they are predicted through bi-linear interpolation. As shown in Figure.4, the decimal value of LBP pattern is given by the binary sequence of circular neighborhood, such as $241 = (11110001)_2$. LBP is gray scale invariant and it can encode the significant local patterns like blobs, edges and lines, because it measures the differences between center pixel and its neighbor pixels. For a given $M * N$ texture image, the LBP pattern $LBP_{r,p}(q_c)$ is computed for every pixel thereby a textured image can be represented by the distribution of LBP values, and make the image represented by a LBP histogram vector. By altering the value of r and the P , we can encode more information about the center pixel q_c .

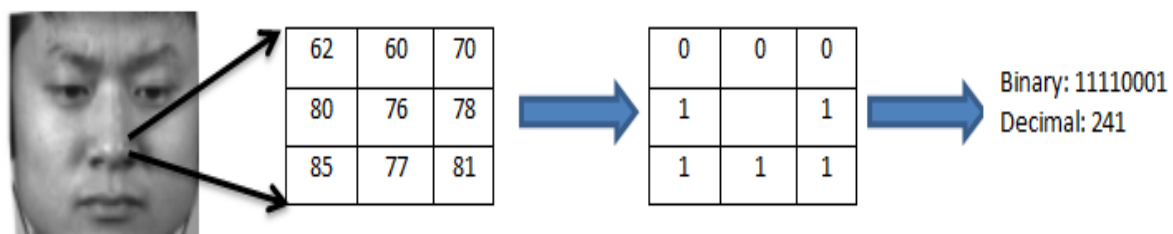


Figure.4 LBP computation

Even though the LBP has gained a better representation in ME, the main drawback is information loss due to the concept of thresholding with center pixel. Further, the proportion of uniform patterns may be too small to acquire the variations. Hence we propose an extended version of LBP called as Radial LBP (RLBP) which encodes the radial information between pixels at different radii. Unlike the conventional LBP which encodes the information between

center pixel and its neighbor pixels on a same circle (single scale), the RLBP considers the pixels on different circles (different scales) for encoding. Due to the consideration of pixels on the single scale, the LBP fails to encode the second order information of neighbor pixels between different circles. For every pixel in the image, we consider two circles or radius r and $r - \delta$ centered on the pixel q_c and P pixels are distributed on each ring evenly. For the computation of RLBP codes, we first measure the radial differences between pixels on two circles and then threshold against 0. Based on the formal definition, the computation of RLBP is done as follows;

$$RLBP_{r,p,\delta}(q_c) = \sum_{n=0}^{P-1} s(q_{r,p,n} - q_{r-\delta,p,n})2^n \quad (15)$$

Where r and $r - \delta$ are the radii of outer and inner circles respectively. Figure.5 shows the simple process of RLBP computation.

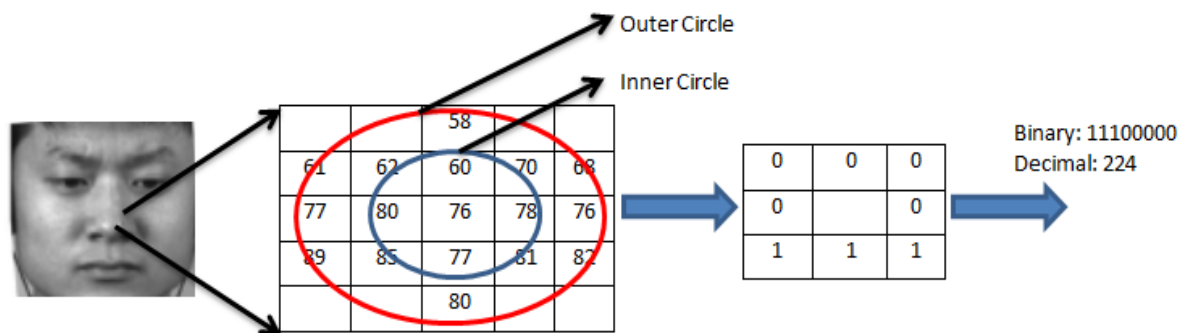


Figure.5 RLBP computation

After the computation of LBP and RLBP for every frame, then we apply histogram computation to extract the histogram features. Finally the obtained histogram of both LBP and RLBP are integrated and formulated into a single feature vector called as Composite Binary Pattern (CBP). Let's consider LBP_H as the histogram of LBP featured image and $RLBP_H$ as the histogram of RLBP featured image, the CBP is obtained by the horizontal concatenation of these two features. In this manner, every frame is represented with CBP vectors and then fed to feature difference analysis to find the exact ME spotting frames.

3.5 Feature Difference Analysis

The main intention to perform the feature difference analysis is to find the final ME spotting frames (onset, apex and offset frames) from an entire ME video sequence. Here we consider three notations such as Back Frame (BF), Front Frame (FF) and Current Frame (CF) for simplified representation. Here if we consider the k as the index of current frame, $k-1$ is the index of back frame and $k+1$ is the index of front frame. Except the first and last frames, all the remaining frames have BF and FF. For feature difference analysis, we seek the help of Chi-Squared (χ^2) distance and it is computed between CF and Average Feature Frame (AFF). Here the AFF is measured by averaging the BF and FF. the difference between CF and AFF indicates the level of changes in the facial area. Furthermore, due to rapid and spontaneous movements of ME, this can help to distinguish the rapid facial movements from temporally longer events. The entire process is repeated for entire frames except first and last frames of ME video sequence. For normalized histograms A and B with same number of bins, the Chi-Squared (χ^2) distance is computed as

$$\chi^2(A, B) = \sum_i \frac{(A_i - B_i)^2}{A_i + B_i} \quad (16)$$

Where the index i refers to the i -th bin of histogram. Here the calculation of dissimilarity check is done through blocks, means the $\chi^2(A, B)$ denotes a two blocks in the CF and AFF at same position. Once the feature difference is computed for every block of each frame, then we compute an initial difference vector by taking an average of M greatest

difference values for each frame. For every frame, M value is calculated as $1/3^{\text{rd}}$ of the entire blocks. For example, if the total number of blocks is 36, then the value of M is 12. Hence 12 blocks with maximum difference is extracted from every frame and the initial difference vector F is computed as

$$F = \frac{1}{M} \sum_{i=1}^M d_i (17)$$

Where d_i is the block difference value of i th block. From the obtained F value, we can find the local peaks which can provide the variations in local magnitude and background noise. The newly obtained local peak vector is called as local difference vector L_i and obtained as

$$L_i = F_i - \frac{1}{2}(F_{i+k} - F_{i-k}) (18)$$

The computation of L_i for each and every frame consequences to L_i of entire video sequence. Based on the obtained L_i values we compute the L_{mean} and L_{max} . Based on these values, we compute a threshold T and the local peak vectors of each frame are compared with threshold. The threshold computation is done as

$$T = L_{mean} + (L_{max} - L_{mean}) (19)$$

The set of frames those have higher L_i than the threshold are only selected as ME spotting frames. The first frame in the obtained resultant set of frames is called as onset frames and the end one is called as offset frame. The frame at the mid position is called as Apex frame.

4. EXPERIMENTAL ANALYSIS

In this section, we explore the details of simulation experiments conducted over the proposed model with the help of MEVIEW micro Expression dataset. For simulation purpose, we used the MATLAB 2015 software with image processing toolbox. Initially we explain the details of dataset and then we explore the details of simulation results obtained.

4.1 Dataset

For simulation purpose, we consider a standard ME dataset called as MEVIEW which was acquired by Peter Husak et al. [24]. This dataset consists of videos collected from TV interviews and poker games. Since the poker game is a stressful game and the candidates try to hide the internal emotions. Players try to fake or conceal their real emotions in which the ME likes to be appearing. The MEs are still rarely available because the post production department removes the most valuable movements such as the details of players face when the cards are being uncovered or somebody raises, folds or calls his/her cards. In this dataset, the average length of each video is observed as 3 seconds. The video clip is captured by switching the camera often on and captures a complete shot with single face. This dataset have totally 31 video clips and they are acquired from 16 subjects. The frame rate of each video is 25 frames per second. Totally five emotions classes (contempt, surprise, fear, anger, happy) are observed in this dataset and every video is FACS encoded. The onset and offset frames of ME are labeled in long videos, FACS coded and the type of emotion is also annotated. Some examples of MEVIEW dataset are shown in the following Figure.6-8.

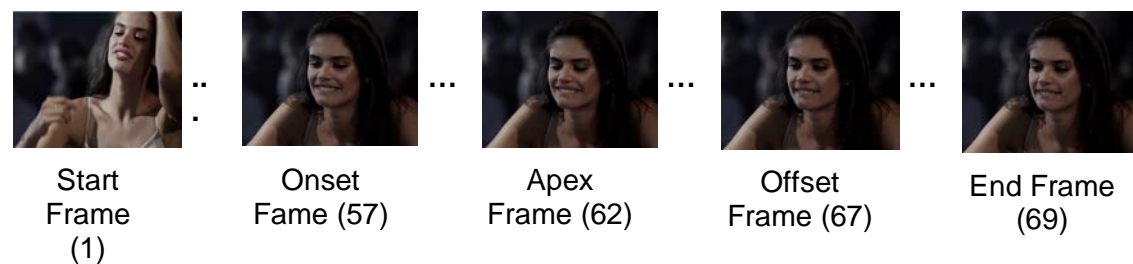


Figure.6 Video sample of Subject 11 for second time

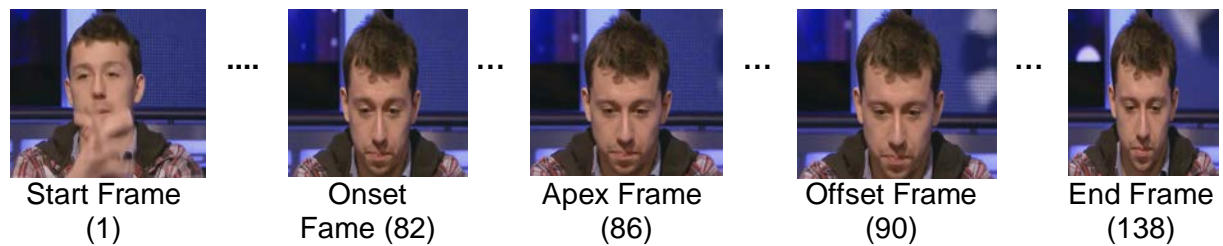


Figure.7 Video sample of Subject 3



Figure.8 Video sample of Subject 5 for second time

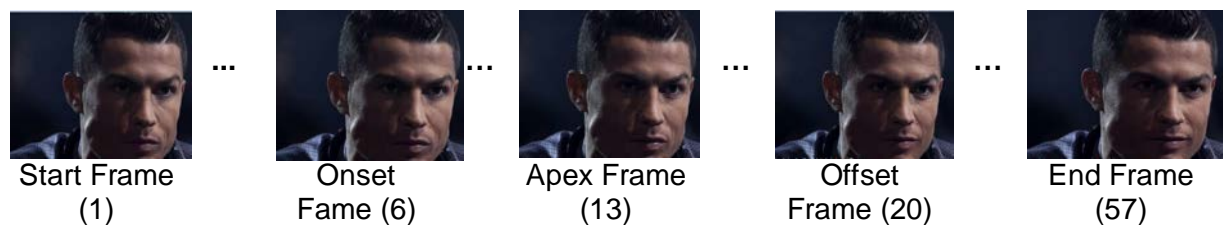


Figure.9 Video sample of Subject 13 for second time

As shown in the Figure.6, the video sample is belongs to subject 11 and acquired from a poker game. According o the annotations available in the dataset, the onset frame is numbered at 57 and offset frame is at 67. Even though, there is no information about the apex frame, for every test video, we defined the apex frame manually by closely observing the entire video samples. The total number of frames present in this video is 69 and the time span is about 2 seconds. Similarly we add more two videos samples those are belong to subject 3 and subject 5. The total number of frames present for the video sample shown in Figure.7 is 138 and for video sample shown in Figure.8 are 174. Finally for the video sample shown in Figure.9, the onset frames is at 6 and offset frame is at 20. The annotations represented at respective samples are taken from the information provided by the authors at the database website.

4.2 Results

For valuation, we consider the entire video samples of MEVIEW dataset. For a given video sample, our method tries to find the ME spotting (frames having Micro expressions). Here the output is the frames of a video sample those have effective micro expressions. With the availability of onset and offset annotations for frames of the ME, then a frame is considered to be correctly detected if it lies in the range of $\left[Onset - \left(\frac{N}{4}\right), Offset + \left(\frac{N}{4}\right)\right]$, where N is the maximum considered length of ME. Here we considered the length of N as 8 because the frame rate is 25 Hz. To balance the uncertainty of the annotations, here we have expanded the range to a small margin.

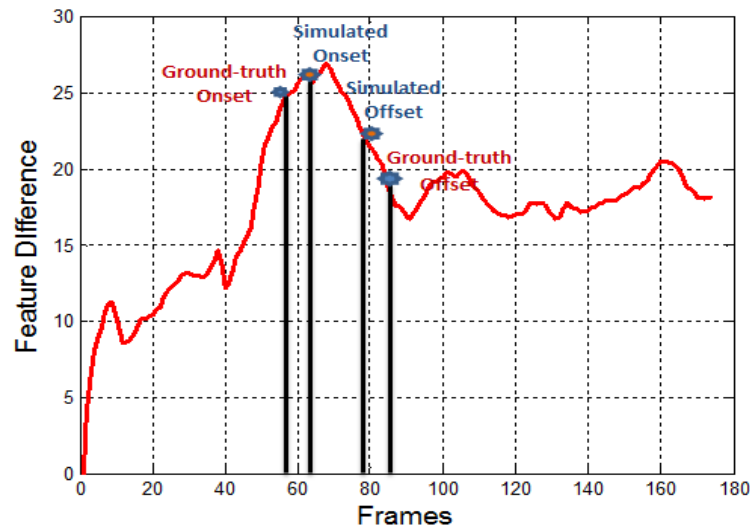


Figure.10 Feature difference analysis and annotations of ground truth and simulated ME spotting

In the above figure.10, we had shown a simple for feature difference analysis and annotated onset and offset frames. This figure is belongs to the video clip of subject 5, where the ground-truth ME spotting is declared as 75-84. From this figure, we can see that the obtained ME spotting interval also lies within the ground-truth range hence it is counted as TP. Moreover, we can also observe that with an increase of frame number along x-axis, the feature difference along y-axis also increasing. From this analysis, we can understand that the ME spotting frames interval has larger difference with first and last frames. Since the ME have peak magnitude sin the spotting interval, the frames shows larger deviations with starting and ending frames.

Next, we analyze the performance analysis of proposed approach through several performance metrics. Here the performance is measured with the True Positives and False Positives. The TP per interval in one video is defined based on the intersection between the spotted interval and ground truth interval. Consider the proposed spotted interval is $P_{Spotted}$ and ground-truth spotted interval is $G_{Spotted}$, then the TP is measured as

$$TP = \frac{(P_{Spotted}) \cap (G_{Spotted})}{(P_{Spotted}) \cup (G_{Spotted})} \geq k(20)$$

Where the k value is set as 0.5, and $G_{Spotted}$ represents the ground-truth interval of micro-expression (onset to offset). For a given test video, if the obtained ME interval satisfies the above condition, then it is considered as TP otherwise it is considered as False Positive (FP). Consider a video have x ground-truth intervals and y spotted intervals, then the TP is counted as z ($z \leq x$ & $z \leq y$), hence $FP = z - x$ and $FN = z - y$. Then the spotting performance for once video can be measured as

$$DR = \frac{z}{x} \text{ and } PPV = \frac{z}{y}(21)$$

And

$$F - Score = \frac{2z}{x+y}(22)$$

The above expressions for performance metrics are to for only one video. For an entire dataset, the overall TP is obtained by the accumulation of all TPs. Consider the test dataset have L videos with X micro-expressions and Y ME intervals, the performance is measured as

$$DR = \frac{Z}{X} \text{ and } PPV = \frac{Z}{Y}(23)$$

Where

$$Z = \sum_{i=1}^L z_i, X = \sum_{i=1}^L x_i \text{ and } Y = \sum_{i=1}^L y_i \quad (24)$$

And

$$F - Score = \frac{2 * DR * PPV}{DR + PPV} \quad (25)$$

Table.1 Performance analysis for MEVIEW dataset

Method	Recall (%)	Precision (%)	F-Score (%)
LBP	45.2314	19.8520	27.6285
RLBP	40.9965	16.4785	23.7017
CBP	53.4412	22.1012	31.2682

The results shown in the Table.1 are obtained based on metrics specified in Eq.(23-24) after the simulation of entire videos available in MEVIEW dataset. In the MEVIEW dataset, the total number of available videos is 40. Irrespective of the notes specified in the dataset information, we consider the entire videos for simulation. In the notes at some videos, they specified that there is no micro expressions but have macro expression or eye blinking etc. For such kind of videos also they specified the ground-truth spotting intervals and hence we consider such kind of videos also for simulation. Since our main objective is to spot the peak expressions, we simulated all the videos and validated with ground-truth interval in every videos. Here we employed both LBP and CBP feature extractors for feature extraction and simulated individually. Absolutely the proposed CBP is observed to have more recall, precision as well as F-Score compared to the conventional LBP. Due to the very limited accomplishment of MEVIEW dataset for ME spotting validation, we didn't provide a comparison with the existing method in literature survey. Compared to CBP and LBP, the RLBP have very less performance due to the consideration of only neighbor pixels information excluding the center pixel.

5. Conclusion

Micro-Expression spotting is much important for Micro Expression recognition. Without an appropriate spotting of ME in lengthy videos, the recognition accuracy is much affected. Moreover, the ME is very spontaneous and for such kind of objects, processing an entire video creates a huge computational burden. Hence, in this paper, we proposed a simple and effective method for ME spotting based on Local Binary Patterns. A new feature descriptor called as CBP is proposed here by combining the conventional LBP with the improved version of LBP called as RLBP. Before feature extraction, as a preprocessing method, we proposed a new landmarks detection followed by facial alignment. Finally after the CBP, we apply feature difference analysis for ME spotting. The recall rate obtained after the simulation of proposed method on MEVIEW dataset proves that performance effectiveness.

REFERENCES

- [1] Ekman, P. (2009b). *Telling Lies: Clues to Deceit in the Marketplace, Politics, and Marriage* (revised edition). W.W. Norton & Company.
- [2] P. Ekman, "Darwin, deception, and facial expression," *Ann. New York Acad. Sci.*, vol. 1000, no. 1, pp. 205-221, 2003
- [3] Q. Wu, X.-B. Shen, and X.-L. Fu, "Micro-expression and its applications," *Adv. Psychol. Sci.*, vol. 18, no. 9, pp. 1359-1368, 2010.
- [4] Ekman, P. (2002). *Microexpression Training Tool (METT)*. University of California, San Francisco, CA.
- [5] Frank, M. G., Maccario, C. J., and Govindaraju, V. (2009b). "Behavior and security," in *Protecting Airline Passengers in the Age of Terrorism*, eds P. Seidenstat and X. Francis, and F. X. Splane (Santa Barbara, CA: Greenwood Pub Group), 86-106.

- [6] S. J. Wang, W. J. Yan, X. Li, G. Zhao, and X. Fu, (2014), "Micro-expression recognition using dynamic textures on tensor independent color space", in *ICPR*, 2014.
- [7] Yan, W.-J., Wu, Q., Liang, J., Chen, Y.-H., and Fu, X. (2013a). How fast are the leaked facial expressions: the duration of micro-expressions. *J. Nonverb. Behav.* 37, 217–230.
- [8] Valstar, M. F., and Pantic, M. (2012). Fully automatic recognition of the temporal phases of facial actions. *IEEE Trans. Syst. Man Cybern. Part B* 42, 28–43.
- [9] Polikovsky, S., and Kameda, Y. (2013). Facial micro-expression detection in hi speed video based on facial action coding system (facs). *IEICE Trans. Inform. Syst.* 96, 81–92.
- [10] Polikovsky, S., Kameda, Y., and Ohta, Y. (2009). "Facial micro-expressions recognition using high speed camera and 3d-gradient descriptor," in 3rd International Conference on Crime Detection and Prevention (ICDP 2009) (London, UK), 1–6.
- [11] Wu, Q., Shen, X., and Fu, X. (2011). "The machine knows what you are hiding: an automatic micro-expression recognition system," in *Affective Computing and Intelligent Interaction ACII 2011* (Memphis), 152–162.
- [12] Shreve, M., Godavarthy, S., Goldgof, D., and Sarkar, S. (2011). "Macro-and micro-expressionspotting in long videos using Spatio-temporal strain," in 2011 *IEEE International Conference on Automatic Face & Gesture Recognition and Workshops (FG 2011)* (Santa Barbara), 51–56.
- [13] Shreve, M., Godavarthy, S., Manohar, V., Goldgof, D., and Sarkar, S. (2009). "Towards macro-and micro-expression spotting in video using strain patterns," in 2009 *Workshop on Applications of Computer Vision (WACV)* (Snowbird), 1–6.
- [14] Shreve, M., Brizzi, J., Fefilatye, S., Luguev, T., Goldgof, D., and Sarkar, S. (2014). Automatic expression spotting in videos. *Image Vis. Comput.* 32, 476–486.
- [15] Moilanen, A., Zhao, G., and Pietikainen, M. (2014). "Spotting rapid facial movements from videos using appearance-based feature difference analysis," in 2014 *22nd International Conference on Pattern Recognition (ICPR)* (Stockholm), 1722–1727.
- [16] Li, X., Xiaopeng, H., Moilanen, A., Huang, X., Pfister, T., Zhao, G., et al. (2017). Towards reading hidden emotions: a comparative study of spontaneous microexpressionspotting and recognition methods. *IEEE Trans. Affect. Comput.*
- [17] Wang, S.-J., Wu, S., Qian, X., Li, J., and Fu, X. (2016). A main directional maximal difference analysis for spotting facial movements from long-term videos. *Neurocomputing* 230, 382–389.
- [18] Davison, A. K., Yap, M. H., and Lansley, C. (2015). "Micro-facial movement detection using individualized baselines and histogram-based descriptors," in 2015 *IEEE International Conference on Systems, Man, and Cybernetics (SMC)* (Kowloon), 1864–1869.
- [19] Davison, A., Lansley, C., Costen, N., Tan, K., and Yap, M. H. (2016). SAMM: a spontaneous micro-facial movement dataset. *IEEE Trans. Affect. Comput.* 9, 116–129. doi: 10.1109/TAFFC.2016.2573832
- [20] Davison, A. K., Lansley, C., Ng, C. C., Tan, K., and Yap, M. H. (2016). Objective micro-facial movement detection using facs-based regions and baseline evaluation. *arXiv preprint arXiv:1612.05038*.
- [21] Xia, Z., Feng, X., Peng, J., Peng, X., and Zhao, G. (2016). Spontaneous micro-expression spotting via geometric deformation modeling. *Comput. Vis. Image Understand.* 147, 87–94.
- [22] Duque, C., Alata, O., Emonet, R., Legrand, A.-C., and Konik, H. (2018). "Microexpression spotting using the Riesz pyramid," in *WACV 2018* (Lake Tahoe).
- [23] Zhihao Zhang, Tong Chen, Hongying Meng, Guangyuan Liu, and Xiaolan Fu, "SMEConvNet: A Convolutional Neural Network for Spotting Spontaneous Facial Micro-Expression from Long Videos", *IEEE Access*, Vol. 6, 2018, pp.71143-71151.
- [24] Petr Husa'k, Jan C'ech, Jiří Matas, "Spotting Facial Micro-Expressions "In the Wild", 22nd Computer Vision Winter Workshop Nicole M. Artner, Ines Janusch, Walter G. Kropatsch (eds.) Retz, Austria, February 6–8, 2017.