Haphazard Cuboids Feature Extraction for Micro-Expression Recognition

GANG WANG1, SHUCHENG HUANG1, AND ZIZHAO DONG2

1School of Computer Science, Jiangsu University of Science and Technology, Zhenjiang 212003, China
2Key Laboratory of Behavior Sciences, Institute of Psychology, Chinese Academy of Sciences, Beijing 100101, China

Corresponding author: Shucheng Huang (schuang6@126.com)

This work was supported by the National Natural Science Foundation of China under Grant 62276118, Grant 61772244, Grant U19B2032, and Grant 62106256.

ABSTRACT Facial micro-expressions can reveal a person’s actual mental state and emotions. Therefore, it has crucial applications in many fields, such as lie detection, clinical medicine, and defense security. However, conventional methods have extracted features on designed facial regions to recognize micro-expressions, failing to effectively hit the micro-expression critical regions since micro-expressions are localized and asymmetric. Consequently, we propose the Haphazard Cuboids (HC) feature extraction method, which generates target regions by haphazard sampling technique and then extracts micro-expression spatio-temporal features. HC consists of two modules: spatial patches generation (SPG) and temporal segments generation (TSG). SPG is assigned to generate localized facial regions, and TSG is dedicated to generating temporal intervals. Through extensive experiments, we demonstrate the superiority of the proposed method. Afterward, we analyze two modules with conventional and deep-learning methods and find that they can significantly improve the performance of micro-expression recognition, respectively. Thereinto, we embed the SPG module into deep learning and experimentally demonstrate the effectiveness and superiority of our proposed sampling method in comparison with state-of-the-art methods. Furthermore, we analyze the TSG module with the maximum overlapping interval (MOI) method and find its coherence with the maximum interval of the apex frame distribution in CASME II and SAMM. Therefore, analogous to the human face’s region of interest (ROI), micro-expressions also inherit similar ROI in the temporal dimension, whose positions are highly relevant to the intensive moment, i.e., the apex frame.

INDEX TERMS Feature extraction, haphazard sampling, micro-expression recognition, ROI.

I. INTRODUCTION

Expressions are visual reflections of human emotions, and non-verbal communication through facial expressions has proven effective and has become a powerful means for humans to deliver affective messages. Facial expressions can be divided into two categories: macro-expressions and micro-expressions. Macro-expressions are considered regular facial expressions, such as happiness, anger, or surprise, consistent with a person’s deeds and lasting, ranging from 1/2s to 4s [1]. Micro-expressions are spontaneously generated expressions that emerge when a person tries to conceal an internal emotion, which cannot be suppressed, and are faint muscle movements of the face, with a duration of 1/25s-1/3s [2], [3], [4], [5].

Compared to macro-expression, micro-expression research has a relatively late start. The concept of micro-expressions was initially introduced by Haggard et al. in 1966 [6]. Then shortly, Ekman et al. [7] reported on the case of micro-expressions, where they observed a video of a conversation between a psychiatrist and a depressed patient, during which the patient tried to conceal his suicidal thoughts by smiling and yet there occasionally appeared several frames of extremely painful expressions. Researchers refer to such rapid, unconscious, spontaneous facial motions as micro-expressions when a person is...
undergoing intensive emotions. Ekman and his colleagues have developed the Facial Action Coding System (FACS) [8] and Micro-Expression Training Tool (METT) over decades of research. FACS describes facial activity based on facial action units (AU) and METT is used to train coders to detect and recognize micro-expressions.

Nevertheless, despite the trained coders, their recognition accuracy is not good, with only 47% reported in the known literature [9]. In addition, relying on a human to recognize micro-expressions is limited by professional training and time cost, making it challenging to be large-scale put into practice. Nevertheless, with the successful development of computer vision in recent years, many researchers have employed it to explore micro-expression recognition methods.

As a typical pattern recognition task, micro-expression recognition can be roughly divided into two parts. The first part is micro-expression feature extraction, which aims to extract features of micro-expressions from a micro-expression clip. The second part is micro-expression classification, which categorizes micro-expressions by designing a classifier, e.g., a support vector machine (SVM). In the last decade, most researchers in micro-expression recognition have dedicated themselves to the feature extraction part. Such an approach is undoubtedly essential since designing reliable micro-expression feature extraction methods which capture tiny facial movements can effectively advance the micro-expression recognition task. The micro-expression feature extraction part can be roughly subdivided into two steps. The first step is selecting specific facial regions, and the second is applying spatio-temporal descriptors to the selected regions to extract micro-expression features. In traditional micro-expression recognition, facial region selection has been based on designed $n \times n$ non-overlapping blocks [10], [11] which can describe local variations better, or ROIs [12], [13] selected based on a priori knowledge for many years. These methods, in the field of macro-expressions, can yield promising results. Nevertheless, micro-expressions are rapid and asymmetric; and are derived from local facial variations. Therefore, we assume that employing the whole face, electing a specific ROI, or using the whole micro-expression clip may not be the optimum scheme. Exploiting the characteristics of the spatio-temporal dimension of micro-expressions is a critical breakthrough in the future.

Therefore, we explore the facial region selection problem in micro-expression recognition as well as investigate the temporal frame selection problem from a new perspective, making the following contributions.

1) We propose an HC feature extraction method containing SPG and TSG modules, which facilitates the extraction of local micro-expression motions by haphazard sampling on the spatial and temporal domains of micro-expressions, and extensive experiments demonstrate that our method is superior to traditional facial selection with non-overlapping blocks and ROIs.

2) We embed the facial haphazard sampling method into deep learning and experimentally demonstrate the effectiveness and superiority of our proposed sampling method in comparison with state-of-the-art methods.

3) Analysis of TSG module by MOI method, it is found that micro-expressions also inherit similar ROI in the temporal dimension, whose positions are highly relevant to the intensive moment, i.e., the apex frame.

This paper reviews the related work on the micro-expression classification in Section II. Then in Section III, the proposed framework and our HC feature extraction method are presented in depth. Section IV covers the experiments conducted, the experimental results, and discussion, and finally Section V concludes this work.

II. RELATED WORK

Micro-expression analysis is a relatively new field. In the last decade, it has attracted much attention, but little work has been fulfilled up to now. Primarily, much of the work was mostly borrowed from macro-expression recognition. However, due to the short duration and subtle variations of micro-expressions, borrowed methods’ performance has been less impressive.

The work of Pfister et al. [10] was one of the very first attempts at automatic micro-expression recognition. Their approach is highly emblematic and presents a comparative benchmark for subsequent work on micro-expression recognition, where dynamic texture features of micro-expressions are extracted by applying local binary patterns from three orthogonal planes (LBP-TOP) [14]. Considering the redundancy and sparsity of LBP-TOP methods, many refined methods [15], [16], [17], [18] have been proposed. Zong et al. [19] designed a hierarchical spatial division scheme and employed kernelized group sparse learning (KGS) model to process hierarchical scheme based spatiotemporal descriptors such that they are more effective for extracting the low-intensity facial muscle movement. Afterward, in [20], they proposed a kernelized two-groups sparse learning (KTGSL) model to learn two groups of weights for two groups of features and select more discriminative features according to the learned weights. Huang et al. [21] proposed an effective EDLPP algorithm, which uses matrix exponential to keep discriminant and local information. Very recently, In MEGC 2019, Liu et al. [22], the first place winner, adopted special data enhancement strategies while utilizing deep neural networks with adversarial training and expression amplification to boost recognition performance. Zhou et al. [23] proposed the Dual-Inception network, which combines the vertical and horizontal components of optical flow for micro-expression recognition. Liong et al. [24] proposed STSTNet, which encodes the horizontal and vertical components of the optical flow, in addition to the optical strain, to learn more compelling features. Quang et al. [25] applied Capsule Networks based on the apex frames to recognize micro-expression. Wang et al. [26] proposed a 2+1D
spatio-temporal convolutional network, which uses 2D convolution to extract spatial features and 1D convolution to extract temporal features.

However, expressions are composed of basic facial action units [8], [27], which correspond to specific facial muscles and correlate with different facial regions. In other words, different facial regions contribute to facial expression unequally, which is particularly valid for micro-expressions derived from local, faint facial motions of muscles. Accordingly, many researchers have explored methods related to facial region selection.

Conventional handcrafted feature extraction methods on micro-expression recognition [19], [28] always separate the whole face into several equal overlapping blocks to better portray local variations. However, some researchers reduced the influence of useless regions by extracting features from ROI [11], such as the forehead, eyebrows, eyes, nose, and mouth, as shown in Fig. 1. Merghani et al. [13], [29] opted for ROIs based on FACS [8], and to eliminate the noise caused by blink movements, glasses, and stationary regions, Le et al. [30] proposed to mask the eye and cheek regions. However, eye movements contribute to the micro-expressions, such as narrowed orbits correlating with disgust highly. Later, in [31], Li et al. constructed a deep local holistic network to learn features from local and global face regions. Cen et al. [32] proposed joint temporal local cube binary pattern and multi-task facial activity patterns learning framework to explore the relationship between action units and emotional states. In recent years, many researchers have used deep learning to explore micro-expressions due to its excellent performance in computer vision tasks. Xia et al. [33] found that the regions around the eyes, nose, and mouth are mostly micro-expression intensive regions which can be elected as ROIs by analyzing the heatmap differences on the micro-expression dataset. Song et al. [34] used a three-stream convolutional neural network to recognize micro-expression, where its block-based segmentation of a face is designed to learn local spatial features better. Zhang et al. proposed a key facial sub-region delineation method based on AU location and facial feature points [35], [36] proposed a fusion model to introduce the action unit’s matrix into the learned facial graph representation, to capture these subtle local variations. [37] proposed a micro-attention cooperating with residual network to focus on facial areas of interest covering different action units. Meanwhile, Li et al. [38] proposed to extract the small facial patches consisting of facial landmarks. In this way, the dimensionality of the learning space can be significantly reduced, which facilitates deep model learning on micro-expression datasets. Xie et al. [39] and [40] employed graph attention convolutional neural network to extract the relationship between feature points and local optical flow patch to improve micro-expression recognition performance.

The above research demonstrated that micro-expressions are local in the spatial dimension. In the meantime, some traits exist in the temporal dimension: from the onset to the apex frame, micro-expressions motion becomes increasingly magnified, and from apex to offset frame, gradually weakens. In [41], they proposed to utilize only two images per video, namely, the apex frame and the onset frame. Afterward, [42] found that the apex frame correlates significantly with the amplitude change in the frequency domain and uses a deep convolutional neural network (DCNN) on the apex frame to recognize micro-expression. Much of the post-production work [24], [43] is based on the apex and neutral frames. In this paper, we analyzed both micro-expressions traits in the spatial and temporal dimensions by leveraging two modules of the proposed HC on the CASME II [44], SAMM [45] and 3DB-combined [46] datasets.

III. PROPOSED METHOD

The overall framework of our proposed micro-expression recognition is shown in Fig. 2. Firstly, we use dlib software [47] to detect face landmarks, perform face alignment, and simultaneously normalize the face to eliminate various confounding factors unrelated to facial expressions. Then we extract micro-expression features using the proposed HC method and finally train an SVM model for classification. The input of the framework is a micro-expression clip, and the output is a specific micro-expression category.

A. FACE ALIGNMENT AND NORMALIZATION

Face alignment is the task of identifying the geometric structure of faces in digital images and attempting to obtain a canonical alignment of the face based on translation, scale, and rotation. Suppose \( DB = \{V_1, V_2, \ldots, V_n\} \) is the selected database, and \( n \) is the total number of micro-expression clips. Where \( V_j = \{V_{j,1}, V_{j,2}, \ldots, V_{j,m}\} \), \( m \) denotes the total number of frames of the \( j \)-th micro-expression clip. First, we use the dlib tool to detect 68 landmarks (as adopted in [40]) in the first frame of each micro-expression video clip, which is denoted as \( \Psi \). Then, the transformation matrix \( T \) is calculated based on the coordinates of the detected landmarks corresponding to the left and right eyes in the image, so that the two eyes can be aligned on the same horizontal line. Since micro-expressions have few head movements in the whole clip, \( T \) can be applied uniformly for all frames in the clip, i.e., \( V_{j,t} = T_j \times V_{j,t}, t = 1, 2, \ldots, m \), where \( T_j \) denotes the transformation matrix of the first frame of the \( j \)-th micro-expression clip.

Face normalization is the processing of facial images to eliminate various confounding factors irrelevant to facial expressions. For this reason, we similarly determine the facial cropping region by the first frame of each micro-expression.
FIGURE 2. An overall framework of HC feature extraction.

clip and then employ the delineated region of the first frame as the facial cropping region of the whole micro-expression clip to reduce the effect of background noise. Suppose $\Psi_j$ is the 68 landmarks points of the first frame of the $j$th micro-expression clip, where we choose the maximum and minimum values of the landmarks points’ coordinates $x$ and $y$, i.e., $\Psi_{j,1}, \Psi_{j,17}, \Psi_{j,9}$ as the left, right and below of the face region. For the upper facial region, in order to eliminate the effect of broken hair, we determine it with the value $\Psi_{j,30, y} - (\Psi_{j,9, y} - \Psi_{j,30, y})/1.3$ (nose and chin) based on the facial construction and the effect of the experimentally cropped image. After performed above steps, the normalized database $DB' = \{V'_1, V'_2, \ldots, V'_n\}$ is obtained.

B. HAPHAZARD CUBOIDS FEATURE EXTRACTION METHOD

Our proposed Haphazard Cuboids (HC) feature extraction method consists of the following steps. At first, spatial patches generation (SPG) module is initialized to generate haphazard patches on the cropped image. Simultaneously temporal segments generation (TSG) module is initialized to generate haphazard intervals in the temporal dimension. Then, each haphazard patch is assembled with temporal segments to form haphazard cuboids, which are utilized to extract the LBP-TOP features from the database $DB'$. Ultimately we concatenate features of each haphazard patch to compose the final feature vector. To further illustrate the effect of TSG module, we employ the degraded version of HC, i.e., Haphazard Patches (HP), which uses entire continuous micro-expressions clips, as an algorithmic comparison. In parallel, we investigate how distinct sizes of HC and the range of temporal segments can influence the performance of micro-expression recognition. The specific processes of feature extraction are described below.

FIGURE 3. An example of SPG with $\rho \in \{0.1, 0.15, 0.2\}$.

1) SPATIAL PATCHES GENERATION

Given an image $I \in \mathbb{R}^{w \times h}$, where $w$ is the width and $h$ is the height. We assign a patch size factor $\rho$, which is employed to generate the width and height of the haphazard patches, where $\rho$ is the percentage of $w$ and $h$, and $\rho \in (0, 0.5)$. To reach approximately the same area as the whole face, we calculate the number of haphazard patches by applying the formula 1.

$$num\_patches = \lceil (1/\rho^2) \rceil$$  \hspace{1cm} (1)

Under natural conditions, the human face has a large-to-smaller pattern from top to bottom, making the cropped face region a partial background noise. To overcome this effect, we horizontally divide the face into three equivalents, and then for each equivalent, we further constrain the horizontal interval of patches by jointly the maximum and minimum values of the horizontal coordinates of the face landmarks in that region to generate patches of the target area and finally merge the patches of each equivalent. The formula is described in lines 2 to 9 of Alg. 1. The SPG with $\rho \in \{0.1, 0.15, 0.2\}$ is illustrated in the Fig. 3, respectively.

2) TEMPORAL SEGMENTS GENERATION

For a micro-expression clip, we assume its length is $t$. Analogous to the SPG, we set the temporal segment factor as $\eta$, where $\eta$ is the percentage of $t$. In order to match the fragment
length generated with the clip’s length relatively, the number of segments can be calculated by the formula 2.

\[ \text{num\_segments} = \lceil 1/\eta \rceil \]  

(2)

First, we scale the length of all micro-expression clips to unit length, then generate segments in the scope of unit length and eventually map them to the original clips. The formula is described in lines 10 to 12 of Alg. 1. The temporal segments generation with \( \eta \in \{0.3, 0.5, 0.8\} \) is illustrated in the Fig. 4, respectively.

3) HAPHAZARD PATCHES FEATURE EXTRACTION METHOD

To explore the temporal aspects of micro-expressions, we wield the HP method, which is invoked to perform feature extraction on the unbroken succession of micro-expression clips. HP is initialized with the SPG module and then uses spatio-temporal descriptors to extract features. As described in Section II, the LBP-TOP method is adopted to extract texture appearance and variations of micro-expressions in this paper due to its typicality and applicability. For the video, there are three dimensions, \( XY \), which refers to the width, height, and time dimensions. A video can be visualized as stacking \( XY \) plane in \( T \) dimension, \( XT \) plane in \( Y \) dimension, or \( YT \) plane in \( X \) dimension. The \( XY \) plane can depict the spatial features, and \( XT \) and \( YT \) can deliver the dynamic motion features in temporal aspects.

LBP-TOP extracts LBP histogram features in \( XY \), \( XT \), and \( YT \) planes, respectively, and finally concatenates the histogram features of the three planes into a final feature vector. The details are shown below.

Assume that HPF is the extracted HP features for the whole database \( DB' \), and \( \text{hpf}_i = [\text{sp}_{i,1}, \text{sp}_{i,2}, \ldots, \text{sp}_{i,m}] \) is the feature of \( ith \) micro-expression clip, where \( m \) is the number of patches, and \( \text{sp}_{i,j} \) denotes the feature of \( jth \) spatial patch of \( ith \) micro-expression clip. Then HPF is concatenated to form the feature of the \( ith \) clip, i.e., \( \text{hpf}_i = \text{sp}_{i,1} \oplus \text{sp}_{i,2} \oplus \ldots \oplus \text{sp}_{i,m} \). For each clip, we extract HPF as the above steps, and then get the final features \( \text{HPF} = \{\text{hpf}_1, \text{hpf}_2, \ldots, \text{hpf}_n\} \). The visualization process is shown in the Fig. 5.

4) HAPHAZARD CUBOIDS FEATURE EXTRACTION METHOD

With SPG and TSG modules initialized, we hereafter embark on extracting HC features using spatio-temporal descriptors. The details are illustrated below.

Suppose HCF is the whole database \( DB' \) extracted HC features. For the \( ith \) micro-expression clip, the feature is \( \text{hcf}_i = [\text{sp}_{i,1}, \text{sp}_{i,2}, \ldots, \text{sp}_{i,m}] \), where \( m \) is the number of patches. For the \( jth \) haphazard cuboid \( \text{sp}_{i,j} = \{\text{ts}_{i,j,1}, \text{ts}_{i,j,2}, \ldots, \text{ts}_{i,j,k}\} \) of the \( ith \) patch, where \( k \) is the number of temporal segments generated. We concatenate \( \text{ts}_{i,j,1}, \text{ts}_{i,j,2}, \ldots, \text{ts}_{i,j,k} \) to form \( \text{sp}_{i,j} \), i.e., \( \text{sp}_{i,j} = \text{ts}_{i,j,1} \oplus \text{ts}_{i,j,2} \oplus \ldots \oplus \text{ts}_{i,j,k} \). Then we similarly concatenate \( \text{sp}_{i,1}, \text{sp}_{i,2}, \ldots, \text{sp}_{i,m} \) to form the feature of the \( ith \) clip, noted as \( \text{hcf}_i \). For each video, we extract HCF according to the above steps, and then get the final features \( \text{HCF} = \{\text{hcf}_1, \text{hcf}_2, \ldots, \text{hcf}_n\} \), and the detailed algorithm is shown in Alg. 1. The visualization flows are illustrated in the Fig. 6.

IV. EXPERIMENTS

In this section, we described the database used, preprocessing process, LBP-TOP parameters, evaluation metrics, model hyperparameter tuning, and experimental results.

A. DATASETS

Considering the brief duration of micro-expressions, we elected databases with high frames per second (fps) from existing micro-expression datasets to explore micro-expressions better in the temporal dimension.

1) SMIC [48]

The SMIC database contains three sub-databases, including SMIC-HS (recorded by a high-speed camera at 100 fps), SMIC-VIS (recorded by a general vision camera at 25 fps), and SMIC-NIR (recorded by a near-infrared camera). SMIC-HS has 164 micro-expression clips from 16 subjects, while SMIC-VIS/SMIC-NIR consisted of 71 samples from 8 participants. Additionally, these samples were annotated as...
Algorithm 1 HC Feature Extraction

Input: video $V \in \mathbb{R}^{h \times w \times t}$, patch size factor $\rho$, segment factor $\eta$, patches collection $\Omega_{SPG}$, segments collention $\Omega_{TSG}$ and feature extractor $\Theta$

Output: feature $hc_f$

1. Detect 68 landmarks of 1th frame of $V$ as $\Psi$;
2. Compute number of patches using (1) as $k$;
3. Initialize range of height $H$ and width $W$;
4. for $j = 1; j \leq 3$ do
   5. $H \leftarrow [h(j-1)/3 : hj/3 - \rho h]$;
   6. $W \leftarrow [\text{min}(\Psi_j) : \text{max}(\Psi_j)]$;
5. Generate and append $[k/3]$ patches in range($H,W$) to $\Omega_{TSG}$;
6. $j \leftarrow j + 1$;
7. end
8. Compute number of segments using (2) as $s$;
9. Set temporal range to $[0 : 1 - \eta]$;
10. Generate $s$ temporal intervals to $\Omega_{TSG}$;
11. for $m = 1; m \leq [k/3 \times 3]$ do
   12. for $n = 1; n \leq s$ do
   13. $hc_f \leftarrow hc_f + \Theta(\Omega_{SPG,m} + \Omega_{TSG,n})$;
   14. $n \leftarrow n + 1$;
   15. end
   16. $m \leftarrow m + 1$;
17. end
18. return $hc_f$;

TABLE 1. CASME II samples per class with five categories.

| Expressions | Number of Samples |
|-------------|-------------------|
| Disgust     | 63                |
| Happy       | 32                |
| Surprise    | 25                |
| Repression  | 27                |
| Other       | 99                |

negative, positive and surprise. In particular, SMIC-HS was used for our experiments.

2) CASMEII [44]

The CASMEII [44] database, recorded by a 200 fps high-speed camera, contains 255 micro-expression samples from 26 participants. These micro-expression clips contains natural micro-expressions labeled with onset, apex, offset, AU, and seven emotional categories. Each face resolution was cropped to $280 \times 340$ pixels. Since CASME II provides higher temporal resolution but remains subject to category imbalance, we only considered micro-expression clips with categories labeled happy, disgust, repression, surprise, and others. The total number of micro-expressions selected for the experiment is 246 (5 classes), and the category details are shown in the Table 1.

3) SAMM [45]

Analogous to the CASME II database, SAMM was performed in a well-controlled laboratory environment and recorded with a high-speed camera at 200 fps. The face resolution was cropped to $400 \times 400$ pixels. It contains 159 micro-expression samples with emotion classes, including contempt, disgust, fear, anger, sadness, happiness, and surprise, generated by 32 subjects. The total number of micro-expressions selected like SAMM was 136 (5 classes), and the breakdown of categories is shown in Table 2.

4) 3DB-COMBINED [46]

A 3DB-combined database is proposed for composite database evaluation (CDE) by MEGC 2019 [46], in which all ME samples from the above three datasets are combined into a composite dataset, which aggregates them into three categories, i.e., negative (including depression, anger, contempt, disgust, fear, and sadness), positive (happy), and surprised. The sample distribution summaries for the datasets and their combinations are shown in Table 3.

B. DATA PREPROCESSING

As mentioned in Section III-A, we carried out face alignment and normalization on the micro-expression clips and resized the shape of each frame to $224 \times 224$ pixels for better feature extraction.

C. EXPERIMENTAL SETUP

To evaluate the effectiveness of each module of our proposed method, we designed two types of experiments, including the traditional hand-crafted method and the deep-learning method. Moreover, extensive experiments are conducted on a single dataset (CASME II and SAMM, 5 classes) and 3DB-combined dataset (3 classes) to validate the effectiveness of our method. All the experiments were conducted with Leave-One-Subject-Out (LOSO) cross-validation, where samples from one subject are held out as the testing set while all remaining samples are for training. Accuracy and F1 score were employed to measure the performance of micro-expression recognition for a single dataset (CASME II and SAMM, 5 classes). Additionally, according to the MEGC 2019, CDE protocol [46] (Unweighted F1-score (UF1) and Unweighted Average Recall (UAR)) is used to measure the
performance of various methods on composite and individual databases with 3 classes.

1) HAND-CRAFTED FEATURE
We conducted experiments on the publicly released databases CASME II [44], SAMM [45] and 3DB-combined dataset [46] respectively. Simultaneously, to validate our proposed method, we took the classical feature extraction algorithm LBP-TOP [14] as our benchmark method and applied the LBP uniform patterns to extract LBP-TOP features, of which the parameters \( R_x = R_y = R_t = 1 \) and \( P = 8 \) were adopted to all experiments below. SVM was deployed to categorize the micro-expressions, and it used radial basis kernel function with penalty factor \( C = [1, 2, \ldots, 50] \) and \( \gamma = 1/(\text{num_features} \times \text{var}(X)) \), in which \text{num_features} denotes the number of features and \text{var}(X) refers to the variance of input \( X \). Hyperparameters were obtained using the grid-search method, and all experiments were accomplished using Python 3.9.4.

2) DEEP-LEARNING FEATURE
To further validate the recentness of our method, we embedded the TSG module of HC into the deep-learning method and performed extensive experiments. Since almost all models designed by researchers in recent papers on deep learning-based micro-expression recognition input entire sequences, grayscale images, or optical flows, however, our manuscript discussion focuses on the impact of facial region selection scheme on micro-expression recognition performance. Thus, to fairly compare, we retrieved papers on deep learning-based micro-expression recognition from recent years and solely found one, namely [34].

In [34], Song et al. proposed Three-Stream Convolutional Neural Network (TSCNN), which extracts the spatio-temporal features of micro-expressions by three components, including the static-spatial component (extracting the appearance and overall outline information of the gray-scale apex frame), the local-spatial component (extracting the local features in the non-overlapping blocks of the apex frame), and the temporal component (extracting the dynamic deformation of micro-expressions). To verify the effectiveness of SPG module, we replicated the full architecture of the TSCNN, only altering the input of its local-spatial component with our facial sampling scheme. For more details, please turn to [34].

Since the apex frame annotation in SMIC is not available, D&C-RoIs method is applied to spot the apex frame (for more details, please turn to [49]). For the CASME II and SAMM datasets, the ground truth of the apex frame is directly used.

As in [34], we set the input image size for each recognition stream to \( 48 \times 48 \) pixels, the basic learning rate to \( 10^{-3} \), the epoch to 30, and each batch to 128. The experiments are conducted with Ubuntu 7.5.0, Python 3.9.4 with Pytorch 1.7.1 on NVIDIA GeForce GTX 1080 (12GB).

D. RESULTS AND DISCUSSION
We conducted baseline experiments (using LBP-TOP [14] method) based on \( 5 \times 5 \), \( 8 \times 8 \), and \( 10 \times 10 \) non-overlapping blocks and 8 ROIs using preprocessed micro-expression clips, respectively, and their experimental results on CASME II and SAMM are shown in Tables 4 and 5. For the baseline on the 3DB-combined dataset, they are individually presented inside a configuration-specific table for convenient comparison.

1) ANALYSIS OF SPG MODULE
a: HAND-CRAFTED FEATURE
Furthermore, to demonstrate the feasibility and effectiveness of our method step by step, we performed the experiments using a degraded version of HC, i.e., the HP method. To begin with, we initialized the SPG module using different parameters \( \rho \). Then, ten experiments for each \( \rho \) were done on two databases. As well, to examine the validity of the HC method, we selected the SPG parameter \( \rho \) with promising recognition performance based on the results of the HP method, and then, considering the comparative analysis of the two methods, we employed the patches already generated by the SPG module in HP, and on top of that, added TSG module. First, by setting the parameter \( \eta \), we initialized the TSG module and then carried out a succession of experiments based on each \( \eta \) separately to circumvent the coincidence of our method.

First of all, we performed experiments on single dataset (CASME II and SAMM) using the HP method, in which we set the parameters \( \rho \in \{0.1, 0.15, 0.2\} \) for the SPG module and performed 10 sets of experiments individually. The details of the experimental results are shown in Tables 6 and 7, respectively, where the maximum values of accuracy and F1 score for each group of experiments arebolded, which implies a quantification of the performance with the potential to generate thriving patches using the corresponding parameters. From Table 6, i.e., the CASME II results, when \( \rho = 0.2 \), the best results for the ten sets of experiments diverged marginally from the baseline results we obtained and occasionally were notably worse afterward, bringing no performance gains. The bulk of this is attributed to the failure to extract the localized features of the micro-expressions, i.e.,

| Method     | Accuracy | F1 score |
|------------|----------|----------|
| 5 x 5 blocks | 0.4065   | 0.3809   |
| 8 x 8 blocks | 0.4146   | 0.3690   |
| 10 x 10 blocks | 0.3821  | 0.3067   |
| 8 regions   | 0.4024   | 0.3885   |

| Method     | Accuracy | F1 score |
|------------|----------|----------|
| 5 x 5 blocks | 0.2868   | 0.2199   |
| 8 x 8 blocks | 0.2794   | 0.2249   |
| 10 x 10 blocks | 0.3382  | 0.2527   |
| 8 regions   | 0.2794   | 0.2327   |
the spatial representation regions. However, the recognition performance of micro-expressions improved significantly when \( \rho = 0.1 \) and \( \rho = 0.15 \). In particular, when \( \rho = 0.15 \), the recognition performance of micro-expressions improved the most. Both the accuracy and F1 score were enhanced by 4\%. From Table 7, i.e., the SAMM results, it could be likewise identified that the recognition performance failed to boost when \( \rho = 0.15 \) and \( \rho = 0.2 \), whereas when \( \rho = 0.1 \), the experimental results of the group 9 exhibited a relatively large performance pickup. Regarding the 60 sets of experimental results on the two databases jointly, it was observed that the experimental results of certain groups were relatively poor because the patches were too large or the representation regions of micro-expressions were not extracted. Therefore, to better explore the feasibility and effectiveness of the HC method proposed in this paper, in the following experiments, we selected the SPG module with parameter \( \rho \in \{0.1, 0.15\} \), having the potential to yield promising results.

**b: DEEP-LEARNING FEATURE**

We conducted experiments with two sizes of non-overlapping blocks (8 \( \times \) 8 and 10 \( \times \) 10) as the baseline, and two types of facial configuration options (SPG module and non-overlapping blocks) using the identical TSCNN model. Ten sets of experiments were conducted for each group. The experimental results in details are shown in Tables 8 and 9. In published papers such as MEGC 2019 [22], [23], [24], [25], they obtained good performance using the network, or aided by data augmentation, to learn features of the entire micro-expression optical flows. However, our manuscript focuses on facial region selection scheme, which is far from theirs. We throw away data augmentation and employ purely deep learning to extract features to verify the superiority of our proposed SPG module over the baseline (non-overlapping blocks).

The experimental results show that the performance of SPG module improves quite a bit compared to the baseline for both configurations. When \( \rho = 0.15 \), the best experimental results showed an improvement of about 7\% for UFI and 8\% for UAR over the baseline. The worst experimental results showed an improvement of about 3\% over the baseline for both UFI and UAR. Also, when \( \rho = 0.1 \), the best group of experiments showed a 4\% improvement in UFI and UAR versus baseline. The worst experimental results showed no elevation as against the baseline, which is attributed to the fact that SPG did not generate an effective micro-expression-active region by chance. We also present the confusion matrix when \( \rho = 0.15 \), where Figure 7 shows the recognition performance for the baseline facial configuration and Figure 8 shows the results for SPG module. Based on the confusion matrix, it is clear that extracting the facial patches (generated by the SPG module) features through TSCNN is more discriminative for Positive and Surprise samples. From the two groups of experimental results, it is apparent that for deep learning, \( \rho = 0.15 \) yields more substantial results. Moreover, embedding our proposed facial sampling method into a comparable model under the same experimental conditions can outperform SOTA algorithms of the same type.

2) ANALYSIS OF TSG MODULE

HC method contains two modules: the SPG module and the TSG module. Above all, based on the experimental results of HP, we noticed that with the SPG module, the recognition performance improved effectively when SPG generated the critical local regions of micro-expressions. As described in the previous paragraph, \( \rho \in \{0.1, 0.15\} \) were utilized to initialize the SPG module. For each \( \rho \), we performed 5 rounds of experiments. Later, we assigned the parameters \( \eta \) of TSG module to 0.3, 0.5, and 0.8. For each \( \eta \), 5 trials were executed, then average accuracy and F1 score were calculated. In other words, for each \( \rho \), 15 experiments were performed on a single database. To counter the casualty of the HC method, 150 experiments on single dataset (CASME II and SAMM) were carried out separately, with a total of 300 groups recorded. The detailed experimental results are shown in Tables 10, 11, 12 and 13.

Tables 10, 11, 12 and 13 showed that compared to the best results at baseline, HC method achieved a 4.9\% improvement in accuracy and a 5.5\% improvement in F1 score in CASME II, and a 3.7\% improvement in accuracy and a 4.6\% improvement in F1 score in SAMM. For a fairer performance comparison, we took the patch size as a criterion and
TABLE 6. Experimental results (Accuracy, F1 score) of HP method with $\rho \in \{0.1, 0.15, 0.2\}$ in CASME II. 1 to 10 represent repeated experiments with the identical parameters.

| $\rho$ | Accuracy | F1 score |
|-------|----------|----------|
| 0.1   | 0.4106   | 0.3922   |
| 0.15  | 0.4228   | 0.4042   |
| 0.2   | 0.3862   | 0.3849   |

TABLE 7. Experimental results (Accuracy, F1 score) of HP method with $\rho \in \{0.1, 0.15, 0.2\}$ in SAMM. 1 to 10 represent repeated experiments with the identical parameters.

| $\rho$ | Accuracy | F1 score |
|-------|----------|----------|
| 0.1   | 0.3308   | 0.3280   |
| 0.15  | 0.2867   | 0.2843   |
| 0.2   | 0.2941   | 0.2928   |

TABLE 8. Experimental results (UF1, UAR) of TSCNN model [34] with $10 \times 10$ non-overlapping blocks and SPG module ($\rho = 0.15$) in 3DB-combined database. 1 to 10 represent repeated experiments with the identical parameters.

| $\rho$ | FULL | CASME II | SAMM | SMIC |
|-------|------|----------|------|------|
| 0.15  | 0.6227 | 0.5624   | 0.7042 | 0.4634 |
| 0.2   | 0.5805 | 0.5800   | 0.7042 | 0.4634 |

TABLE 9. Experimental results (UF1, UAR) of TSCNN model [34] with $10 \times 10$ non-overlapping blocks and SPG module ($\rho = 0.1$) in 3DB-combined database. 1 to 10 represent repeated experiments with the identical parameters.

| $\rho$ | FULL | CASME II | SAMM | SMIC |
|-------|------|----------|------|------|
| 0.1   | 0.5939 | 0.5998 | 0.7046 | 0.4634 |

calculated the mean value of multiple experiments. In this rule, $10 \times 10$ non-overlapping blocks corresponds to $\rho = 0.1$ and $8 \times 8$ corresponds to $\rho = 0.15$. The mean value of experiments in single dataset was calculated. In CASME II, when $\rho = 0.15$, the accuracy was 0.4143, and the F1 score was 0.4011. Compared with $8 \times 8$ non-overlapping blocks, accuracy was equal, but the F1 score improved by 3.2%. When $\rho = 0.1$, the average accuracy obtained was 0.4125, and the average F1 score was 0.3647. Compared with $10 \times 10$ non-overlapping blocks, accuracy improved by 3%, while the F1 score improved by 5.8%. Similarly, in SAMM, when $\rho = 0.15$, the accuracy improved by 2% with an equal F1 score. When $\rho = 0.1$, the accuracy improved by 0.8%, and the F1 score by 1.3%. From the above results, it was clear that the HC method achieved an effective performance improvement.
Table 10. Experimental results (Accuracy / F1 score) of HC method with $\rho = 0.15$, $\eta \in \{0.3, 0.5, 0.8\}$ in CASME II. 1 to 5 represent repeated experiments with the identical parameters.

| $\rho = 0.15$ | 1       | 2       | 3       | 4       | 5       | Average |
|--------------|---------|---------|---------|---------|---------|---------|
| $\eta = 0.3$  | 0.4605/0.3935 | 0.3943/0.3817 | 0.4024/0.4015 | 0.4065/0.3924 | 0.3984/0.3866 | 0.4016/0.3911 |
| $\eta = 0.5$  | 0.3984/0.3970 | 0.4024/0.4001 | 0.3894/0.3993 | 0.4065/0.3990 | 0.3984/0.3981 | 0.4008/0.3987 |
| $\eta = 0.8$  | 0.3984/0.3956 | 0.3902/0.3944 | 0.3943/0.3926 | 0.3984/0.3946 | 0.4065/0.4040 | 0.3976/0.3962 |

Table 11. Experimental results (Accuracy / F1 score) of HC method with $\rho = 0.1$, $\eta \in \{0.3, 0.5, 0.8\}$ in CASME II. 1 to 5 represent repeated experiments with the identical parameters.

| $\rho = 0.1$ | 1       | 2       | 3       | 4       | 5       | Average |
|--------------|---------|---------|---------|---------|---------|---------|
| $\eta = 0.3$  | 0.4372/0.4248 | 0.4472/0.4428 | 0.4634/0.4409 | 0.4431/0.4324 | 0.4472/0.4398 | 0.4488/0.4388 |
| $\eta = 0.5$  | 0.4512/0.4462 | 0.4593/0.4294 | 0.4593/0.4478 | 0.4593/0.4404 | 0.4553/0.4319 | 0.4569/0.4391 |
| $\eta = 0.8$  | 0.4431/0.4340 | 0.4472/0.4434 | 0.4431/0.4332 | 0.4431/0.4396 | 0.4512/0.4399 | 0.4455/0.4380 |

Likewise, the results on the 3DB-combined database are shown in Tables 14 and 15, from which it can be seen that for combining micro-expression datasets of different ethnic groups, our facial sampling as well as time-domain selection schemes, i.e., $SPG$ and $TSG$, are superior to the non-overlapping blocks and entire micro-expression clip. When $\rho = 0.1$, UFI boosts nearly 3% and UAR 2% over the baseline at most; however, when $\rho = 0.15$, UFI improves by 4.5% and UAR 8% up the maximum. Apparently, $\rho = 0.15$ and $\eta \in \{0.5, 0.8\}$ bring more promising results. It is still evident that our proposed method is effective for a broader range of datasets.

From another perspective, we compared the $HP$ method with the $HC$ method in correspondence. By Tables 10, 11, 12 and 13, it was evident that the micro-expression recognition performance obtained by $HC$ method was generally higher than that of $HP$ method, especially for experiments on CASME II. The experiments indicated that, compared with the $HP$ method, $HC$ could achieve a relatively substantial performance improvement, either accuracy or F1 score, by incorporating the $TSG$ module. For instance, in Table 10, the accuracy was up to 0.4634, and the corresponding F1 score was up to 0.4409. For the comparative results on the CASME II database, i.e., Table 11 and 10 suggested...
that the TSG parameter $\eta \in \{0.3, 0.5\}$ further improve the accuracy and F1 score significantly. Likewise, for the SAMM database, Table 13 identified that the recognition performance of micro-expressions gained overall promotion, though the presence of some groups without performance gains. When $\rho = 0.1$, $HC$ reached its peak.

Next, to further investigate why the $HC$ method works, we visualized some well-performing micro-expression segments generated by the TSG module in the $HC$ method. The normalized temporal segments on CASMEII and SAMM are shown in Fig. 9 and 10. Moreover, we applied the maximum overlapping interval ($MOI$) method, which aims to locate temporal intervals with the most overlaps, to certify the effectiveness of the TSG module more precisely. The $MOI$ method can be illustrated directly with an example, i.e., suppose there are four intervals $[0.1, 0.6], [0.2, 0.7], [0.3, 0.8], [0.4, 0.9]$, and then the result obtained by $MOI$ is $[0.4, 0.6]$. Without bells and whistles, $MOI$ reflects the common traits of numerous well-performing $HC$ methods.

On the CASMEII database (Fig. 9), we statistically obtained the $MOI$ ranging from 0.45 to 0.49. However, on the SAMM database (Fig. 10), the $MOI$ was 0.36 to 0.51. The reasons for the different overlapping intervals may be the source of the database samples, such as ethnicity, age, etc. Moreover, the $TSG$ module was stochastically initialized, which may

\[
\begin{array}{|c|c|c|c|c|c|c|}
\hline
\text{Table 12. Experimental results (Accuracy / F1 score) of } HC \text{ method with } \rho = 0.15, \eta \in \{0.3, 0.5, 0.8\} \text{ in SAMM. 1 to 5 represent repeated experiments with the identical parameters.} \\
\hline
\text{ } & \text{1} & \text{2} & \text{3} & \text{4} & \text{5} & \text{Average} \\
\hline
\text{\rho=0.15} & \\
\eta=0.3 & 0.2941/0.2196 & 0.2794/0.2127 & 0.3088/0.2295 & 0.3015/0.2252 & 0.2941/0.2217 & 0.2956/0.2217 \\
\eta=0.5 & 0.2721/0.2136 & 0.2647/0.2077 & 0.2721/0.2134 & 0.2721/0.2139 & 0.2647/0.2095 & 0.2691/0.2116 \\
\eta=0.8 & 0.2721/0.2111 & 0.2721/0.2098 & 0.2721/0.2111 & 0.2721/0.2111 & 0.2721/0.2108 & 0.2721/0.2108 \\
\hline
\text{\rho=0.2528} & \\
\eta=0.3 & 0.3309/0.2456 & 0.3235/0.2396 & 0.3162/0.2364 & 0.3456/0.2553 & 0.3382/0.2503 & 0.3309/0.2454 \\
\eta=0.5 & 0.3382/0.2229 & 0.3382/0.2472 & 0.3456/0.2371 & 0.3309/0.2447 & 0.3456/0.2497 & 0.3397/0.2403 \\
\eta=0.8 & 0.3309/0.2479 & 0.3309/0.2472 & 0.3382/0.2521 & 0.3382/0.2501 & 0.3325/0.2411 & 0.3323/0.2477 \\
\hline
\text{\rho=0.2288} & \\
\eta=0.3 & 0.3235/0.2340 & 0.3235/0.2421 & 0.3235/0.2440 & 0.3309/0.2475 & 0.3325/0.2431 & 0.3250/0.2421 \\
\eta=0.5 & 0.3088/0.2244 & 0.3088/0.2202 & 0.3088/0.2202 & 0.3325/0.2213 & 0.3170/0.2209 & 0.3170/0.2209 \\
\eta=0.8 & 0.3162/0.2330 & 0.3162/0.2333 & 0.3088/0.2288 & 0.3015/0.2192 & 0.3162/0.2330 & 0.3118/0.2294 \\
\hline
\text{\rho=0.2286} & \\
\eta=0.3 & 0.2941/0.2122 & 0.2868/0.2083 & 0.3015/0.2132 & 0.3015/0.2136 & 0.2941/0.2085 & 0.2956/0.2112 \\
\eta=0.5 & 0.2941/0.2109 & 0.2941/0.2117 & 0.2868/0.2061 & 0.2868/0.2049 & 0.2941/0.2181 & 0.2912/0.2114 \\
\eta=0.8 & 0.2941/0.2105 & 0.2941/0.2105 & 0.2941/0.2194 & 0.2941/0.2194 & 0.2941/0.2116 & 0.2941/0.2143 \\
\hline
\text{\rho=0.2072} & \\
\eta=0.3 & 0.2868/0.2072 & 0.2794/0.2032 & 0.2574/0.1899 & 0.2721/0.2010 & 0.2647/0.1945 & 0.2721/0.1992 \\
\eta=0.5 & 0.2721/0.2068 & 0.2647/0.2015 & 0.2647/0.2026 & 0.2574/0.1911 & 0.2574/0.1911 & 0.2633/0.1986 \\
\eta=0.8 & 0.2721/0.2084 & 0.2647/0.2084 & 0.2647/0.2036 & 0.2647/0.2027 & 0.2500/0.1945 & 0.2632/0.2035 \\
\hline
\end{array}
\]
omit the crucial segments occasionally owing to relatively limited experiments. In addition, we tallied the apex frame distribution of the samples in both databases, as shown in Fig. 11 and 12. The intervals with the highest frequency of apex frame positions for CASME II and SAMM were [0.45, 0.55] and [0.3, 0.40], respectively. We found that the maximum frequency interval of the apex frame distribution tends to be consistent with the interval obtained from the MOI method on CASME II and SAMM. The consistency bias regarding SAMM is since our TSG module generates a relatively large range of temporal frames but contains the critical region of micro-expressions.

Based on the performance pickup brought by the TSG module, we demonstrated the critical interval of micro-expressions in the temporal dimension, which holds significant representative information of micro-expressions. From another side, we also identified that [43] barely considering the apex and reference frame to extract optical flow features is reasonable since the apex frame portrays the highest intensive muscle motion of micro-expressions. The results suggested that, similar to the ROI in the spatial dimension, the micro-expressions equally possess ROI in the temporal domain.

TABLE 14. Experimental results (UF1, UAR) of HC method with $\rho = 0.1$, $\eta \in \{0.3, 0.5, 0.8\}$ in 3DB-combined database. 1 to 5 represent repeated experiments with the identical parameters.

| METHOD | FULL | CASME II | SAMM | SMIC |
|---|---|---|---|---|
| $10 \times 10$ blocks | UF1 | UAR | UF1 | UAR | UF1 | UAR | UF1 | UAR |
| 0.3 | 0.4075 | 0.4206 | 0.3464 | 0.3468 | 0.3155 | 0.3556 | 0.2600 | 0.3574 |
| 1 | 0.4279 | 0.4239 | 0.4793 | 0.4750 | 0.3649 | 0.3797 | 0.3548 | 0.3559 |
| 2 | 0.4222 | 0.4223 | 0.5005 | 0.4938 | 0.3253 | 0.3539 | 0.3525 | 0.3564 |
| 3 | 0.4237 | 0.4219 | 0.5017 | 0.4930 | 0.3295 | 0.3575 | 0.3569 | 0.3647 |
| 4 | 0.4177 | 0.4153 | 0.4873 | 0.4891 | 0.3332 | 0.3611 | 0.3315 | 0.3320 |
| 5 | 0.4315 | 0.4411 | 0.4711 | 0.5026 | 0.3000 | 0.3338 | 0.2965 | 0.3436 |
| 0.5 | 0.4367 | 0.4434 | 0.4856 | 0.5073 | 0.2655 | 0.3188 | 0.3042 | 0.3483 |
| 2 | 0.4163 | 0.4140 | 0.4695 | 0.4674 | 0.3546 | 0.3725 | 0.3347 | 0.3374 |
| 3 | 0.4331 | 0.4430 | 0.4693 | 0.4959 | 0.3293 | 0.3539 | 0.2881 | 0.3382 |
| 4 | 0.4254 | 0.4346 | 0.4532 | 0.4788 | 0.3041 | 0.3374 | 0.2962 | 0.3430 |
| 5 | 0.4337 | 0.4398 | 0.4642 | 0.4864 | 0.2655 | 0.3188 | 0.3110 | 0.3549 |

TABLE 15. Experimental results (UF1, UAR) of HC method with $\rho = 0.15$, $\eta \in \{0.3, 0.5, 0.8\}$ in 3DB-combined database. 1 to 5 represent repeated experiments with the identical parameters.

| METHOD | FULL | CASME II | SAMM | SMIC |
|---|---|---|---|---|
| $8 \times 8$ blocks | UF1 | UAR | UF1 | UAR | UF1 | UAR | UF1 | UAR |
| 0.3 | 0.4111 | 0.4091 | 0.4823 | 0.4758 | 0.3488 | 0.3657 | 0.2974 | 0.3065 |
| 1 | 0.4282 | 0.4441 | 0.4454 | 0.4762 | 0.3168 | 0.3558 | 0.3125 | 0.3668 |
| 2 | 0.4296 | 0.4444 | 0.4412 | 0.4667 | 0.3168 | 0.3558 | 0.3164 | 0.3734 |
| 3 | 0.4254 | 0.4387 | 0.4407 | 0.4667 | 0.3193 | 0.3934 | 0.3072 | 0.3591 |
| 4 | 0.4278 | 0.4223 | 0.4436 | 0.4705 | 0.3168 | 0.3538 | 0.3187 | 0.3698 |
| 5 | 0.4384 | 0.4573 | 0.4532 | 0.4924 | 0.3120 | 0.3258 | 0.3373 | 0.3924 |
| 0.5 | 0.4378 | 0.4548 | 0.4568 | 0.4687 | 0.3168 | 0.3558 | 0.3238 | 0.3811 |
| 2 | 0.4323 | 0.4494 | 0.4596 | 0.4933 | 0.3168 | 0.3538 | 0.3102 | 0.3668 |
| 3 | 0.4384 | 0.4574 | 0.4684 | 0.5000 | 0.3168 | 0.3538 | 0.3157 | 0.3763 |
| 4 | 0.4294 | 0.4463 | 0.4514 | 0.4829 | 0.3168 | 0.3538 | 0.3121 | 0.3668 |
| 5 | 0.4312 | 0.4467 | 0.4466 | 0.4762 | 0.3193 | 0.3934 | 0.3179 | 0.3716 |

TABLE 16. Experimental results (UF1, UAR) of HC method with $\rho = 0.15$, $\eta \in \{0.3, 0.5, 0.8\}$ in 3DB-combined database. 1 to 5 represent repeated experiments with the identical parameters.

| METHOD | FULL | CASME II | SAMM | SMIC |
|---|---|---|---|---|
| $8 \times 8$ blocks | UF1 | UAR | UF1 | UAR | UF1 | UAR | UF1 | UAR |
| 0.8 | 0.4559 | 0.467 | 0.4559 | 0.4687 | 0.3103 | 0.3285 | 0.3134 | 0.3763 |
| 2 | 0.4247 | 0.4427 | 0.4424 | 0.4753 | 0.3103 | 0.3285 | 0.3100 | 0.3686 |
| 3 | 0.4321 | 0.4504 | 0.4514 | 0.4829 | 0.3168 | 0.3538 | 0.3149 | 0.3746 |
| 4 | 0.4245 | 0.4427 | 0.4424 | 0.4753 | 0.3103 | 0.3285 | 0.3087 | 0.3686 |
| 5 | 0.4259 | 0.4437 | 0.4463 | 0.4791 | 0.3127 | 0.3321 | 0.3095 | 0.3668 |

FIGURE 9. The temporal intervals of selected well-performing micro-expression experiments in CASME II.
In this paper, we propose an $HC$ feature extraction method including $SPG$ and $TSG$ modules, which facilitates the extraction of local micro-expression movements by haphazardly sampling the spatial and temporal dimensions of micro-expressions, and extensive experiments (applying LBP-TOP features [10]) demonstrate that our method is superior to the baseline (i.e., traditional facial selection with LBP-TOP features [10]). We also embed the $SPG$ module into a deep-learning method and experimentally demonstrate its superiority in comparison with state-of-the-art methods (i.e., TSCNN [34]). In the temporal aspect, we assume that, like facial-specific ROIs, micro-expressions have specific ROIs, and investigate the temporal frame selection problem from a new perspective.

Further, comparing our proposed method ($HC$) to its degraded version ($HP$), we found that the general superiority of the $HC$ method over the $HP$ method by 300 groups of experiments is attributed to the intro of the $TSG$ module. From the overall experimental results, the best performance configuration is available when $\rho = 0.15$ and $\eta = 0.5$. Then, we selected the groups with superior performance in the $HC$ method compared to the corresponding $HP$ method, tallied the $MOI$ of temporal segments in the $TSG$ module, and drew a comparison with the apex frame distribution of the database. The results show an intrinsic consistency between the two. In other words, analogous to the ROI of the human face, the micro-expressions also inherit a similar ROI in the temporal domain, and their positions are highly relevant to the moment when the most intensive facial motion of the micro-expression occurs, i.e., the apex frame.

V. CONCLUSION

In this paper, we propose an $HC$ feature extraction method including $SPG$ and $TSG$ modules, which facilitates the extraction of local micro-expression movements by haphazardly sampling the spatial and temporal dimensions of micro-expressions, and extensive experiments (applying LBP-TOP features [10]) demonstrate that our method is superior to the baseline (i.e., traditional facial selection with non-overlapping blocks and ROIs). Afterwards, we embed the $SPG$ module into a deep-learning method and experimentally demonstrate its superiority in comparison with state-of-the-art methods (i.e., TSCNN [34]). In the temporal aspect, we assume that, like facial-specific ROIs, micro-expressions have specific ROIs, and investigate the temporal frame selection problem from a new perspective.

Further, comparing our proposed method ($HC$) to its degraded version ($HP$), we found that the general superiority of the $HC$ method over the $HP$ method by 300 groups of experiments is attributed to the intro of the $TSG$ module. From the overall experimental results, the best performance configuration is available when $\rho = 0.15$ and $\eta = 0.5$. Then, we selected the groups with superior performance in the $HC$ method compared to the corresponding $HP$ method, tallied the $MOI$ of temporal segments in the $TSG$ module, and drew a comparison with the apex frame distribution of the database. The results show an intrinsic consistency between the two. In other words, analogous to the ROI of the human face, the micro-expressions also inherit a similar ROI in the temporal domain, and their positions are highly relevant to the moment when the most intensive facial motion of the micro-expression occurs, i.e., the apex frame.

REFERENCES

[1] P. Ekman, “Darwin, deception, and facial expression,” Ann. New York Acad. Sci., vol. 1000, no. 1, pp. 205–221, 2003.
[2] P. Ekman and W. V. Friesen, “Constants across cultures in the face and emotion,” J. Personality Social Psychol., vol. 17, no. 2, p. 124, 1971.
[3] P. Ekman, “Lie catching and microexpressions,” Philosophy Deception, vol. 1, no. 2, p. 5, 2009.
[4] W.-J. Yan, Q. Wu, J. Liang, Y.-H. Chen, and X. Fu, “How fast are the leaked facial expressions: The duration of micro-expressions,” J. Nonverbal Behav., vol. 37, no. 4, pp. 217–230, 2013.
[5] J. Li, Z. Dong, S. Lu, S.-J. Wang, W.-J. Yan, Y. Ma, Y. Liu, C. Huang, and X. Fu, “CAS(ME)3: A third generation facial spontaneous micro-expression database with depth information and high ecological validity,” IEEE Trans. Pattern Anal. Mach. Intell., early access, May 13, 2022, doi: 10.1109/TPAMI.2022.3174895.
[6] E. A. Haggard and K. S. Isaacs, “Micromomentary facial expressions as indicators of ego mechanisms in psychotherapy,” in Methods of Research in Psychotherapy, Boston, MA, USA: Springer, 1966, pp. 154–165.
[7] P. Ekman and W. Friesen, “Nonverbal leakage and clues to deception,” Psychiatry, vol. 32, no. 1, pp. 88–106, 1969.
[8] P. Ekman and W. Friesen, “Facial action coding system,” in Environmental Psychology & Nonverbal Behavior, Washington, DC, USA: APA PsychTests, 1978.
[9] M. Frank, M. Herbasz, K. Sinuk, A. Keller, and C. Nolan, “I see how you feel: Training laypeople and professionals to recognize fleeting emotions,” in Proc. Annu. Meeting Int. Commun. Assoc., New York, NY, USA, 2009, pp. 1–35.
[10] T. Pfister, X. Li, G. Zhao, and M. Pietikainen, “Recognising spontaneous facial micro-expressions,” in Proc. Int. Conf. Comput. Vis., Nov. 2011, pp. 1449–1456.
[11] X. Li, X. Hong, A. Moilanen, X. Huang, T. Pfister, G. Zhao, and M. Pietikäinen, “Towards reading hidden emotions: A comparative study of spontaneous micro-expression spotting and recognition methods,” IEEE Trans. Affective Comput., vol. 9, no. 4, pp. 563–577, Oct. 2017.
[12] S.-J. Wang, W.-J. Yan, G. Zhao, X. Fu, and C.-G. Zhou, “Micro-expression recognition using robust principal component analysis and local spatiotemporal directional features,” in Proc. Eur. Conf. Comput. Vis., Cham, Switzerland: Springer, 2014, pp. 325–338.
[13] W. Merghani and M. H. Yap, “Adaptive mask for region-based facial micro-expression recognition,” in Proc. 15th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG), Nov. 2020, pp. 765–770.
[14] G. Zhao and M. Pietikäinen, “Dynamic texture recognition using local binary patterns with an application to facial expressions,” IEEE Trans. Pattern Anal. Mach. Intell., vol. 29, no. 6, pp. 915–928, Jun. 2007.
[15] Y. Wang, J. See, R. C.-W. Phan, and Y.-H. Oh, “LBP with six intersection points: Reducing redundant information in LBP-TOP for micro-expression recognition,” in Proc. Asian Conf. Comput. Vis., Cham, Switzerland: Springer, 2014, pp. 525–537.
[16] Y. Wang, J. See, R.-C.-W. Phan, and Y.-H. Oh, “Efficient spatio-temporal local binary patterns for spontaneous facial micro-expression recognition,” *PLoS ONE*, vol. 10, no. 5, May 2015, Art. no. e0124674.

[17] S.-J. Yan, W.-J. Yan, X. Li, G. Zhao, C.-G. Zhou, X. Fu, M. Yang, and J. Tao, “Micro-expression recognition using color spaces,” *IEEE Trans. Image Process.*, vol. 24, no. 12, pp. 6034–6047, 2015.

[18] X. Huang, G. Zhao, X. Hong, W. Zheng, and M. Pietikäinen, “Spontaneous facial micro-expression analysis using spatiotemporal completed local quantized patterns,” *Neurocomputing*, vol. 175, pp. 564–578, Jan. 2016.

[19] Y. Zong, X. Huang, W. Zheng, Z. Cui, and G. Zhao, “Learning from hierarchical spatiotemporal descriptors for micro-expression recognition,” *IEEE Trans. Multimedia*, vol. 20, no. 11, pp. 3160–3172, Nov. 2018.

[20] J. Wei, G. Li, J. Yan, and Y. Zong, “Learning two groups of discriminative features for micro-expression recognition,” *Neurocomputing*, vol. 479, pp. 22–36, Mar. 2022.

[21] S. Huang and L. Zhuang, “Exponential discriminant locality preserving projection for face recognition,” *Neurocomputing*, vol. 208, pp. 373–377, Oct. 2016.

[22] Y. Liu, H. Du, L. Zheng, and T. Gedeon, “A neural micro-expression recognizer,” in Proc. 14th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG), May 2019, pp. 1–4.

[23] L. Zhou, Q. Mao, and L. Xiao, “Dual-inception network for cross-database micro-expression recognition,” in Proc. 14th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG), May 2019, pp. 1–5.

[24] S.-T. Liong, Y. S. Gan, J. See, H.-Q. Khor, and Y.-C. Huang, “Shallow triple stream three-dimensional CNN (STSTNet) for micro-expression recognition,” in Proc. 14th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG), May 2019, pp. 1–5.

[25] N. Van Quang, J. Chuan, and T. Tokuyama, “CapsuleNet for micro-expression recognition,” in Proc. 14th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG), May 2019, pp. 1–7.

[26] S.-J. Wang, Y. He, J. Li, and X. Fu, “MESNet: A convolutional neural network for spotting multi-scale micro-expression intervals in long videos,” *IEEE Trans. Image Process.*, vol. 30, pp. 3956–3969, 2021.

[27] E. L. Rosenberg and P. Ekman, *What the Face Reveals: Basic and Applied Studies of Spontaneous Expression Using the Facial Action Coding System* (FACS). Oxford, U.K.: Oxford Univ. Press, 2020.

[28] S.-J. Wang, S. Wu, X. Qian, J. Li, and X. Fu, “A main directional maximal local binary patterns for spontaneous facial micro-expression recognition,” *IEEE Trans. Image Process.*, vol. 24, no. 4, pp. 1571–1580, Apr. 2015.

[29] D. E. King, “Dlib-Ml: A machine learning toolkit,” *J. Mach. Learn. Res.*, vol. 10, pp. 1755–1758, Jan. 2009.

[30] S.-T. Liong, J. See, K. Wong, and M. H. Yap, “SAMMI: A spontaneous Micro-Facial movement dataset,” *IEEE Trans. Affective Comput.*, vol. 9, no. 1, pp. 116–129, Jun. 2016.

[31] J. See, M. H. Yap, J. Li, X. Hong, and S.-J. Wang, “MEGC 2019—The second facial micro-expressions grand challenge,” in Proc. 14th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG), May 2019, pp. 1–5.

[32] D. E. King, “‘Dlib-Ml: A machine learning toolkit,’” *J. Mach. Learn. Res.*, vol. 10, pp. 1755–1758, Jan. 2009.

[33] S. Huang and L. Zhuang, “Exponential discriminant locality preserving projection for face recognition,” *Neurocomputing*, vol. 208, pp. 373–377, Oct. 2016.

[34] Y. Liu, H. Du, L. Zheng, and T. Gedeon, “A neural micro-expression recognizer,” in Proc. 14th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG), May 2019, pp. 1–4.

[35] L. Zhou, Q. Mao, and L. Xiao, “Dual-inception network for cross-database micro-expression recognition,” in Proc. 14th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG), May 2019, pp. 1–5.

[36] S.-T. Liong, Y. S. Gan, J. See, H.-Q. Khor, and Y.-C. Huang, “Shallow triple stream three-dimensional CNN (STSTNet) for micro-expression recognition,” in Proc. 14th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG), May 2019, pp. 1–5.

[37] J. See, M. H. Yap, J. Li, X. Hong, and S.-J. Wang, “MEGC 2019—The second facial micro-expressions grand challenge,” in Proc. 14th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG), May 2019, pp. 1–5.

[38] D. E. King, “‘Dlib-Ml: A machine learning toolkit,’” *J. Mach. Learn. Res.*, vol. 10, pp. 1755–1758, Jan. 2009.

[39] S. Huang and L. Zhuang, “Exponential discriminant locality preserving projection for face recognition,” *Neurocomputing*, vol. 208, pp. 373–377, Oct. 2016.

[40] Y. Liu, H. Du, L. Zheng, and T. Gedeon, “A neural micro-expression recognizer,” in Proc. 14th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG), May 2019, pp. 1–4.

[41] L. Zhou, Q. Mao, and L. Xiao, “Dual-inception network for cross-database micro-expression recognition,” in Proc. 14th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG), May 2019, pp. 1–5.

[42] S.-T. Liong, Y. S. Gan, J. See, H.-Q. Khor, and Y.-C. Huang, “Shallow triple stream three-dimensional CNN (STSTNet) for micro-expression recognition,” in Proc. 14th IEEE Int. Conf. Autom. Face Gesture Recognit. (FG), May 2019, pp. 1–5.

[43] D. E. King, “Dlib-Ml: A machine learning toolkit,” *J. Mach. Learn. Res.*, vol. 10, pp. 1755–1758, Jan. 2009.

[44] S. Liong, J. See, K. Wong, and R. C. Phan, “Less is more: Micro-expression recognition—spreading over a single apex frame,” in Proc. 5th IEEE Int. Conf. Image Process. (ICIP), Oct. 2018, pp. 3094–3098.

[45] Y. S. Gan, S.-T. Liong, W.-C. Yau, Y.-C. Huang, and L.-K. Tan, “OFF-ApexNet on micro-expression recognition system,” *Signal Process., Image Commun.*, vol. 74, pp. 129–139, May 2019.

[46] S.-T. Liong, J. See, K. Wong, and R. C. Phan, “Less is more: Micro-expression recognition—spreading over a single apex frame,” in Proc. 5th IEEE Int. Conf. Image Process. (ICIP), Oct. 2018, pp. 3094–3098.

[47] A. K. Davison, C. Lansley, N. Costen, K. Tan, and M. H. Yap, “SAMMI: A spontaneous micro-expression database and the baseline evaluation,” *Proc. 10th IEEE Int. Conf. Workshops Autom. Face Gesture Recognit. (FG)*, Apr. 2013, pp. 1–6.

[48] S.-T. Liong, J. See, K. Wong, A. C. Le Ngo, Y.-H. Oh, and R. Phan, “Automatic apex frame spotting in micro-expression database,” in Proc. 3rd IAPR Asian Conf. Pattern Recognit. (ACPR), Nov. 2015, pp. 665–669.

**GANG WANG** was born in Jiangsu, China, in 1997. He received the bachelor’s degree from the Jiangsu University of Science and Technology, Zhenjiang, China, in 2016, where he is currently pursuing the master’s degree with the College of Computer Science. His research interests include machine learning and facial expression (micro-expression) recognition.

**SHUCHENG HUANG** received the bachelor’s and master’s degrees in computer application from the China University of Mining and Technology, Xuzhou, China, in 1991 and 2001, respectively, and the Ph.D. degree in computer application from Southeast University, Nanjing, China, in 2007. He is currently a Professor with the Jiangsu University of Science and Technology, Zhenjiang, China. His current research interests include computer vision and multimedia analysis.

**ZIZHAO DONG** received the B.S.Ed. degree in applied psychology major from China Women’s University, China, in 2018. She is currently pursuing the M.Psy. degree with the Institute of Psychology, Chinese Academy of Sciences. Her current research interests include facial micro-expression, visual psychophysics, and neurophysiology.