Article

SL-Swin: A Transformer-Based Deep Learning Approach for Macro- and Micro-Expression Spotting on Small-Size Expression Datasets

Erheng He [1,†], Qianru Chen [2,†] and Qinghua Zhong [1,2*]

[1] School of Physics and Telecommunication Engineering, South China Normal University, Guangzhou, 510006, P.R. China.
[2] School of Electronics and Information Engineering, South China Normal University, Foshan, 528225, P.R. China.
* Correspondence: zhongqinghua@m.scnu.edu.cn
† These authors contributed equally to this work.

Abstract: In recent years, the analysis of macro- and micro-expression has drawn the attention of researchers since they provide visual cues of an individual's emotions for a broad range of potential applications such as lie detection and criminal detection. In this paper, we address the challenge of spotting facial macro- and micro-expression from videos and present compelling results by using a deep learning approach to analyze the optical flow features. Different from other deep learning approaches that are mainly based on Convolutional Neural Networks (CNNs), we propose a Transformer-based deep learning approach that predicts a score indicating the probability of a frame being within an expression interval. In contrast to other Transformer-based models that achieve high performance by being pre-trained on large datasets, our deep learning model, called SL-Swin, which incorporates Shifted Patch Tokenization and Locality Self-Attention into the backbone network Swin Transformer, effectively spots macro- and micro-expressions by being trained from scratch on small-size expression datasets. Our evaluation outcomes surpass the MEGC 2022 spotting baseline result, obtaining an overall F1-score of 0.1366. Additionally, our approach also performs well in the MEGC 2021 spotting task with an overall F1-score of 0.1824 and 0.1357 on CAS(ME)^2 and SAMM Long Videos, respectively. The code is publicly available on GitHub (https://github.com/eddiehe99/pytorch-expression-spotting).

Keywords: Macro- and Micro-Expression Spotting; Image Processing; Computer Vision; Artificial Intelligence; Deep Learning; Swin Transformer; Shifted Patch Tokenization; Locality Self-Attention

1. Introduction

Facial expressions, usually conveyed and perceived by an individual through the movements of facial muscles, are a form of non-verbal communication that provides visual cues of an individual's emotional state. Macro-Expression (MaE) and Micro-Expression (ME) are two categories of facial expressions that vary according to their intensity and duration. MaE, which occurs at higher intensities, involves facial movements that cover a large facial area. It usually lasts from 0.5 s to 4.0 s and can be easily identified from a single frame in a MaE video sequence. Conversely, the Micro-Expression (ME) is subtle and has a shorter duration (usually within 0.5 s [1]) making it more challenging to spot than MaE.

Generally, facial expressions go through three distinct phases: onset, apex, and offset. As described in [2], the onset phase marks the beginning of the facial muscle contraction (the first frame at which an expression starts), the apex phase represents the facial action at its peak intensity, and the offset phase indicates the return of the facial muscles to a neutral state (the last frame at which an expression ends). The concept of expression analysis comprises two aspects: specifically, spotting and recognition [3]. The spotting task is designated to identify whether a given video contains expressions and locate the expression intervals

 doi:10.20944/preprints202306.0079.v2

from the onset to the offset phases if they are found in the video. The task of expression recognition involves categorizing expressions into predetermined emotion types, such as surprise, sadness, happiness, anger, etc. In this paper, our approach is to address the spotting task. It is important to spot expressions not only because expressions provide clues for potential applications such as lie detection and criminal detection, but also because the spotting reduces the labor required to collect expression data [4].

The existing macro- and micro-expression spotting approaches can be roughly divided into traditional approaches and deep learning approaches [5]. Traditional expression spotting approaches use manually crafted features to tell whether a frame is an expression frame or not. The method proposed by Davison et al. [6] involves splitting faces into blocks and calculating the HOG for each frame. Afterwards, this method spotted micro-expressions by using Chi-Squared distance to calculate the dissimilarity between frames at a set interval. Duque et al. [7] proposed the Riesz pyramid-based method. Wang et al. [4] characterized the magnitude of maximal difference in the main direction of optical flow by the Main Directional Maximal Differences (MDMD) method they proposed. Zhang et al. [8] and the baseline of MEGC2020 [9] used the optical flow-based method to spot expressions, which proved that it is functional to extract facial expression movements by describing facial movements using optical flow features. Especially the extraction of micro-expressions which are subtle and shorter in duration [10].

Traditional approaches limit the representation capability of features as they are extracted manually. While these approaches perform well in spotting MaEs, their capability of spotting MEs of which the features present extremely weak differences are much less compelling. With the development of deep learning, some researchers apply deep learning methods in their approaches to overcome the limitations of traditional approaches. Zhang et al. [11] who first introduced deep learning to micro-expression spotting utilized Convolutional Neural Network (CNN) to detect the apex frame and merged nearby detected samples by a feature engineering method they proposed. Pan et al. [12] introduced the usefulness of regions of interest (ROI) selection and adopted the Bilinear Convolutional Neural Network (BCNN) for expression spotting. As the baseline of MEGC2021 [13] and MEGC2022 [14], Yap et al. [15] applied frame skipping and contrast enhancement to their approach which is based on a 3D-CNN network. Furthermore, the approach in which the Histogram of Oriented Optical Flow features of a sliding window is extracted as input features for an RNN composed of LSTM units, was proposed by Verburg et al. [16] who first utilized Recurrent Neural Network (RNN) for expression spotting.

Though Convolutional Neural Networks (CNNs) have dominated in Computer Vision (CV) including expression spotting and recognition, the prevalent architecture today in Natural Language Processing (NLP) is instead the self-attention-based architectures, particularly Transformers [17]. Inspired by the successes of Transformers in NLP, [18] applied a standard Transformer directly to images and attained excellent results on image recognition benchmarks, for instance, ImageNet [19]. Considering the compelling performance Transformers achieved in NLP and CV tasks, some researchers tried combining CNN with Transformer for expression spotting. Pan et al. [20] proposed a Spatio-temporal Convolutional Emotional Attention Network (STCEAN) which extracts spatial features through the convolution neural network and employs the emotional self-attention model to analyze the emotional weights in the temporal dimension for spotting. Based on the bidirectional self-attention mechanism, the BERT network [21] which is a stack of the Transformer's Encoder not only thrives in Natural Language Processing (NLP) tasks, but also behaves outstandingly in extracting spatio-temporal features together with the 3D-CNN in the approach proposed by Zhou et al. [22]. Guo et al. [23] proposed a convolutional transformer network that uses a multi-scale local transformer module to attain the correlation between frames based on the visual features extracted by a 3D convolutional subnetwork. Compared to expression recognition, the usage of Transformers in expression spotting is limited.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 June 2023 doi:10.20944/preprints202306.0079.v2

Proven by Liong et al. [24] and Liong et al. [25], the spotting task can be fashioned as a regression problem that predicts the probability of a frame being within a macro- or micro-expression interval. And the deep learning models can be trained to discover the expression motion information hidden in the optical flow features. Inspired by the above work, we propose a Transformer-based deep learning approach for spotting macro- and micro-expression by analyzing the optical flow features extracted from videos. Our deep learning model, called SL-Swin, which applies Shifted Patch Tokenization (SPT) [26] and Locality Self-Attention (LSA) [26] to the backbone Swin Transformer [27], achieves compelling results by being trained from scratch on small-size expression datasets. In addition, we facilitate the feature learning process by applying the pseudo labeling technique [25] in the training phase and predict the apex frame in each video by employing the peak detection technique [28] after the smooth process. The contributions of this paper are listed below:

- We propose a deep learning approach that uses Swin Transformer as the backbone to generate a score for spotting by analyzing optical flow features.
- We implement SPT which gives a wider receptive field than standard tokenization to embed more spatial optical flow information into visual tokens for the training phase.
- We employ LSA which impels the attention to work locally by forcing each token to concentrate more on tokens with large relation to itself, to enable the network to pay more attention to visual tokens that contain important expression motion information.
- We incorporate both SPT as well as LSA into the backbone Swin Transformer to enable it to be trained from scratch on small-size expression datasets and our study demonstrates the effectiveness of the Transformer-based deep learning approach by outperforming the MEGC 2022 spotting baseline approach and achieving a comparably competent outcome in the MEGC 2021 spotting task.

2. Materials and Methods

In this paper, we spot MaEs and MEs in a given video separately. We use the SL-Swin model to generate an expression score to predict the possibility of a frame within the interval of an expression. The proposed approach is illustrated in Figure 1. Five phases of our approach are outlined: initial feature extraction of optical flow features, pre-processing, optical flow feature learning using the SL-Swin, pseudo labeling, and the expression spotting procedure.

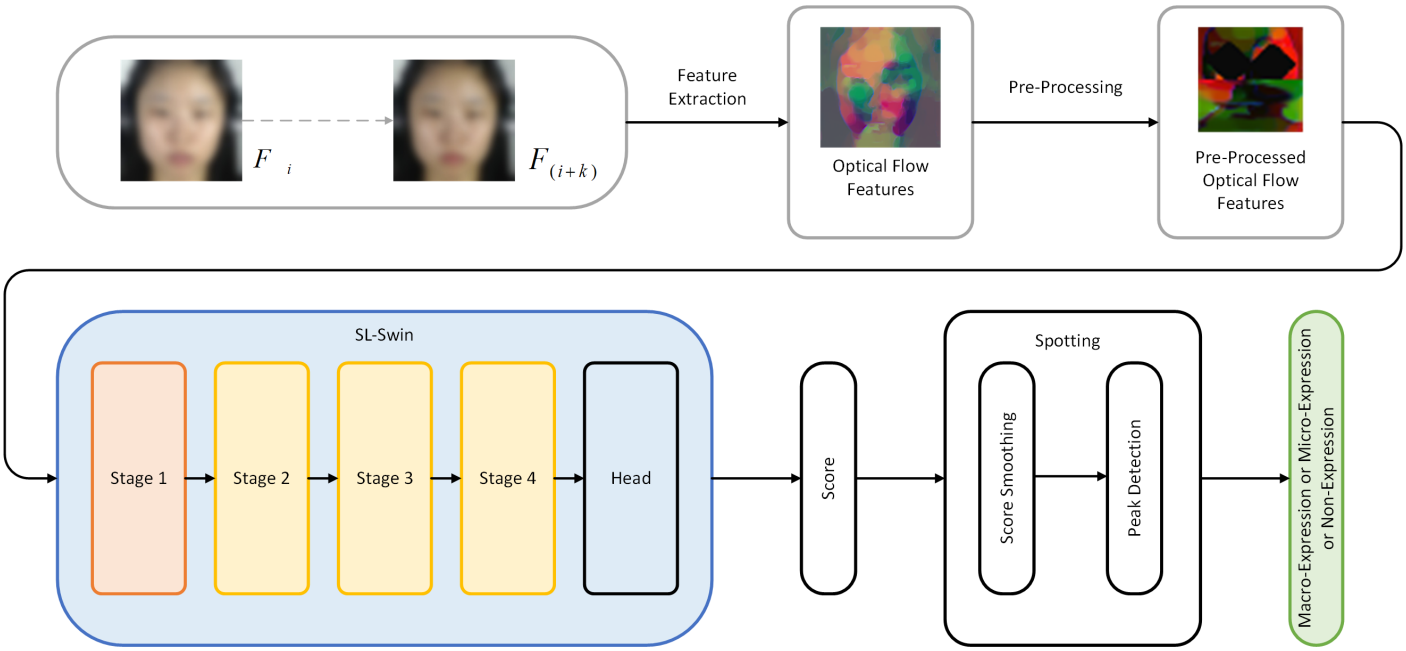


Figure 1. The illustration of the proposed approach.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 June 2023 doi:10.20944/preprints202306.0079.v2

2.1. Feature Extraction

We use optical flow features which carry substantial spatio-temporal motion information [24] [25] as the input for the deep learning model. To begin with, we crop the facial region in each frame and resized it to 128×128 pixels for resolution normalization. The cropping is performed using the OpenCV's DNN Face Detector which is based on a Single-Shot-Multibox detector and uses ResNet-10 Architecture as the backbone and is conducted on every frame of every raw video.

Next, the current frame F_i and frame $F_{(i+k)}$ (the $k-th$ frame from the current frame F_i) are used to compute the optical flow features, where k is half of the average length of an expression interval. Since the TV-L1 optical flow estimation method is the most robust among all optical flow estimation methods tested in [29], we use it in this paper to compute the horizontal component u and vertical component v that consists of the first and second channel of the model input features. In addition, we use them to compute the optical strain ϵ which catches the subtle facial deformation from optical flow components [30]:

$$\epsilon = \begin{bmatrix} \epsilon_{xx} = \frac{\delta u}{\delta x} & \epsilon_{xy} = \frac{1}{2}\left(\frac{\delta u}{\delta y} + \frac{\delta v}{\delta x}\right) \\ \epsilon_{yx} = \frac{1}{2}\left(\frac{\delta v}{\delta x} + \frac{\delta u}{\delta y}\right) & \epsilon_{yy} = \frac{\delta v}{\delta y} \end{bmatrix} \tag{1}$$

Where ϵ_{xy} and ϵ_{yx} are shear strain components while ϵ_{xx} and ϵ_{yy} are normal strain components. The third channel of the features that will be fed into the model is the optical strain magnitude $|\epsilon|$, which can be computed as:

$$|\epsilon| = \sqrt{\epsilon_{xx}^2 + \epsilon_{yy}^2 + \epsilon_{xy}^2 + \epsilon_{yx}^2} \tag{2}$$

To sum up, the input data for pre-processing ahead of the model learning phases is the concatenation of these three components $(u, v, |\epsilon|)$ whose shape is $(128, 128, 3)$.

2.2. Pre-Processing

Before model learning, we pre-process the extracted optical flow features to ensure data consistency and remove noise. Motivated by the work of [8], we subtract the mean feature of the nose region to eliminate the head motion of each frame.

Then, as the eye blinking has a significant disturbance to optical flow features [31], a black polygon-shaped mask is applied to the left and right eye regions with an additional margin of 15 pixels along the height and width. Next, on the basis that eyebrows and the mouth contain significant movements [31], we use three rectangular boxes with 12 pixels as the additional margin to enclose three regions as ROIs. ROI 1 which comes from the region of the left eye and left eyebrow is acquired and resized to 21×21. Meanwhile, ROI 2 which has the same resized size as ROI 1 is from the region of the right eye and right eyebrow. ROI 3 which originates from the region of the mouth is acquired and resized to 21×42.

Eventually, the ultimate pre-processed optical flow features $(u, v, |\epsilon|)$ of which the shape is $(42, 42, 3)$ for the training phase is acquired through the following steps. First, the resized ROI 1 and ROI 2 are horizontally stacked to form the upper portion whose size is 21×42. Afterwards, the resized ROI 3 which composes the lower portion is vertically stacked under the upper portion.

2.3. SL-Swin

For the sake of spotting expression by training from scratch on small-size expression datasets, we propose SL-Swin with these further considerations: (1) The backbone of the network is the Swin Transformer [27] which could effectively consider the local and global features of the expression optical flow features; (2) Shifted Patch Tokenization (SPT) [26] and Locality Self-Attention (LSA) [26] are applied to the backbone to enable the network to be trained from scratch even on small-size expression datasets. (3) A head module is added to predict a score indicating the probability of a frame being within an expression interval.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 June 2023 doi:10.20944/preprints202306.0079.v2

Figure 2 illustrated an overview of the model SL-Swin-T, which is based on the tiny version of the Swin Transformer (Swin-T). SL means both SPT as well as LSA are applied to the model. To start with, the SPT module in "Stage 1" splits the input optical flow features into non-overlapping patches which are treated as "visual tokens". In our approach, the patch size p is 6, and the output is projected to dimension C by a linear embedding layer in SPT. Next, the tokens go through several Transformer blocks with LSA (L Swin Transformer blocks) which keep the resolution of tokens at $(\frac{H}{6} \times \frac{W}{6})$. and together with the SPT are referred to as "Stage 1". The H and W are the width and weight of the pre-processed optical flow features which is put into the model.

The number of tokens is decreased by patch merging layers in the following "Stages" as the network gets deeper because the backbone Swin Transformer is designed to build hierarchical feature maps. The patch merging layer decreases the number of tokens by a multiple of $2 \times 2 = 4$ ($2\times$ downsampling of resolution) by concatenating the tokens of each group of 2×2 adjacent patches. Then, a linear layer within the patch merging layer is applied to project the downsampled $4C$-dimensional concatenated features to the $4C$-dimensional output. Afterwards, feature transformation is conducted by several L Swin Transformer blocks (L means LSA is applied) which maintain the resolution at $\frac{H}{12} \times \frac{W}{12}$. This first combination of the patch merging layer and several L Swin Transformer blocks is denoted as "Stage 2". The procedure is repeated twice, as "Stage 3" and "Stage 4", with output resolutions of $\frac{H}{24} \times \frac{W}{24}$ and $\frac{H}{48} \times \frac{W}{48}$, respectively. In the end, a head that acts as a regression module which consists of the normalization and the MLP is applied to predict a score indicating the probability of a frame being within an expression interval.

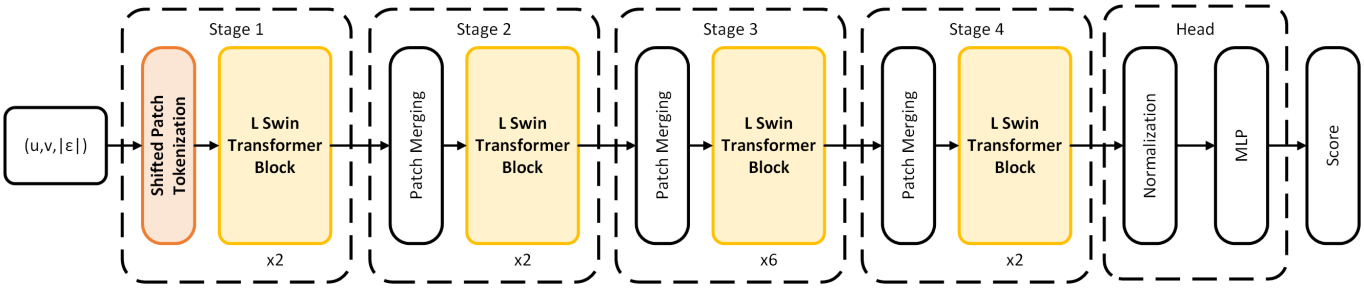


Figure 2. The architecture of the tiny version of the Swin Transformer applied with both SPT and LSA, namely SL-Swin-T.

2.3.1. Swin Transformer

Facial expressions could be divided into individual muscle movement components, known as Action Units (AUs) [5]. As the experiment described in [32], a single macro- or micro-expression may have more than one AU with high intensity. Consequently, to identify whether a macro- or micro-expression appears or not, the model must take the local features, global features, and the relationships among local features from various input portions into consideration.

Inspired by the attention mechanism [17] in the field of Natural Language Processing (NLP) and the ViT [18] which applies attention to the field of Computer Vision (CV), we use a deep learning model called SL-Swin which uses Swin Transformer [27] as the backbone. The Swin Transformer is built in hierarchical architecture and the transformer representation is computed with shifted windows. The standard Transformer architecture conducts global self-attention which leads to quadratic computation complexity in respect of the number of tokens because the relationships between a token and all other tokens are computed. For efficient modeling, the Swin Transformer computes self-attention within windows that evenly partition the tokens into non-overlapping parts. To address the drawback of lacking connections across windows in the window based self-attention module, the shifted window partitioning approach which shifts the window and computes the self-attention within the new windows that cross the boundaries of the non-overlapping windows is proposed in consecutive Swin Transformer blocks. In the computation of every

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 June 2023 doi:10.20944/preprints202306.0079.v2

consecutive Swin Transformer block, the window based multi-head self-attention using regular window partitioning configurations (W-MSA) and window based multi-head self-attention using shifted window partitioning configurations (SW-MSA) are used in pairs. The shifted windowing scheme which includes regular and shifted window partitioning configurations shows efficient modeling power by confining self-attention computation to non-overlapping windows while harnessing cross-window connections. Therefore, the network could effectively take local features, global features, and the relation among local features from different parts of the input optical flow features into consideration.

Whereas the models based on Transformers such as ViT and Swin Transformer require a large amount of training data or require pre-training on a large-size dataset to obtain high performance. The dataset for micro-expression spotting is relatively small, which may limit the performance of the model based on the Transformer. To enable the network to perform well on comparatively small-scale expression datasets, we implement SPT which gives a wider receptive field to the model than standard tokenization by embedding more spatial information in visual tokens. And we employ LSA which enables the network to pay more attention to visual tokens that contain important motion information. The details of SPT and LSA are demonstrated below.

2.3.2. Shifted Patch Tokenization

The Shifted Patch Tokenization (SPT) outputs the tensor with the same shape as the original Patch Embedding Layer of the Swin Transformer or Vision Transformer. Therefore, we use the SPT as a Patch Embedding Layer in our approach. The SPT is the head component of the SL-Swin model, which means the pre-processed optical flow features will be processed by the SPT first when the features are fed into the model. The following describes the overall formulation of the SPT and how we implement the SPT as a Patch Embedding Layer.

Firstly, the shifting strategy is applied to shift each input pre-processed optical flow features by half the patch size in four diagonal directions which are left-up, right-up, left-down, and right-down. Next, the shifted features are concatenated with the original input pre-processed optical flow features after being cropped to the same size as the input. Then, the concatenated features $[x, x_s^1, x_s^2, x_s^3, x_s^4]$ are divided into non-overlapping patches and the patches are flattened to a sequence of vectors, which are formulated as follows:

$$DF\Big(\Big[x, x_s^1, x_s^2, x_s^3, x_s^4\Big]\Big) = \Big[x_p^1; x_p^2; \ldots; x_p^N\Big] \tag{3}$$

Where x is the original pre-processed optical flow features and x_s^1, x_s^2, x_s^3, x_s^4 are the cropped features shifted in left-up, right-up, left-down, and right-down directions. $x_p^i \in \mathbb{R}^{P^2 \times C}$ is the i-th flattened vector. p stands for the patch size and $N = \frac{HW}{p^2}$ stands for the number of patches. DP stands for the dividing and flattening process.

Afterwards, visual tokens (VT) are obtained through layer normalization (LN) and projected by a linear layer (LL). The whole process is formulated as:

$$VT(x) = LL\Big(LN\Big(\Big[x_p^1; x_p^2; \ldots; x_p^N\Big]\Big)\Big) \tag{4}$$

In order to use SPT as a Patch Embedding Layer, we add a positional embedding variable to the output of SPT. The whole process is formulated as:

$$VT_{pe}(x) = VT(x) + E_{pos} \tag{5}$$

Where E_{pos} is the learnable positional embedding variable and $VT_{pe}(x)$ is the ultimate output that will be processed by the rest of the model.

In all, the SPT in our approach could be seen as the combination of the Patch Partition and the Linear Embedding in "Stage 1" of the original Swin Transformer architecture.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 June 2023 doi:10.20944/preprints202306.0079.v2

2.3.3. Locality Self-Attention

The diagonal masking and learnable temperature scaling consist of the core of the Locality Self-Attention Mechanism (LSA). Figure 3(a) demonstrates the difference between the standard self-attention mechanism and the locality self-attention used in the SL-Swin model. Figure 3(b) also shows how LSA is applied in the successive Swin Transformer blocks to form the L Swin Transformer block. The W-MLSA and SW-MLSA in the two successive Swin Transformer blocks of Figure 3(b) denote window based multi-head locality self-attention using regular and shifted window partitioning configurations, respectively. The L indicates that the LSA is applied.

The standard self-attention computation of general ViTs operates as follows. In the beginning, the Query, Key, and Value are obtained by applying a learnable linear projection to each token. Next, calculate the similarity matrix R which represents the relation between tokens is calculated through the dot product operation of Query and Key. The diagonal and the off-diagonal components of R represent self-token and inter-token relations, respectively:

$$R = QK^T \tag{6}$$

Here, Q and K denote learnable linear projections for Query and Key. Afterwards, the diagonal masking forces $-\infty$ on diagonal components of R to emphasize the inter-token relations by basically excluding self-token relations from the following computation. This enforce the model to concentrate more on other tokens rather than its inter-tokens. The diagonal masking is formulated as:

$$R^M = \begin{cases} R_{i,j}(i \neq j) \\ -\infty(i = j) \end{cases} \tag{7}$$

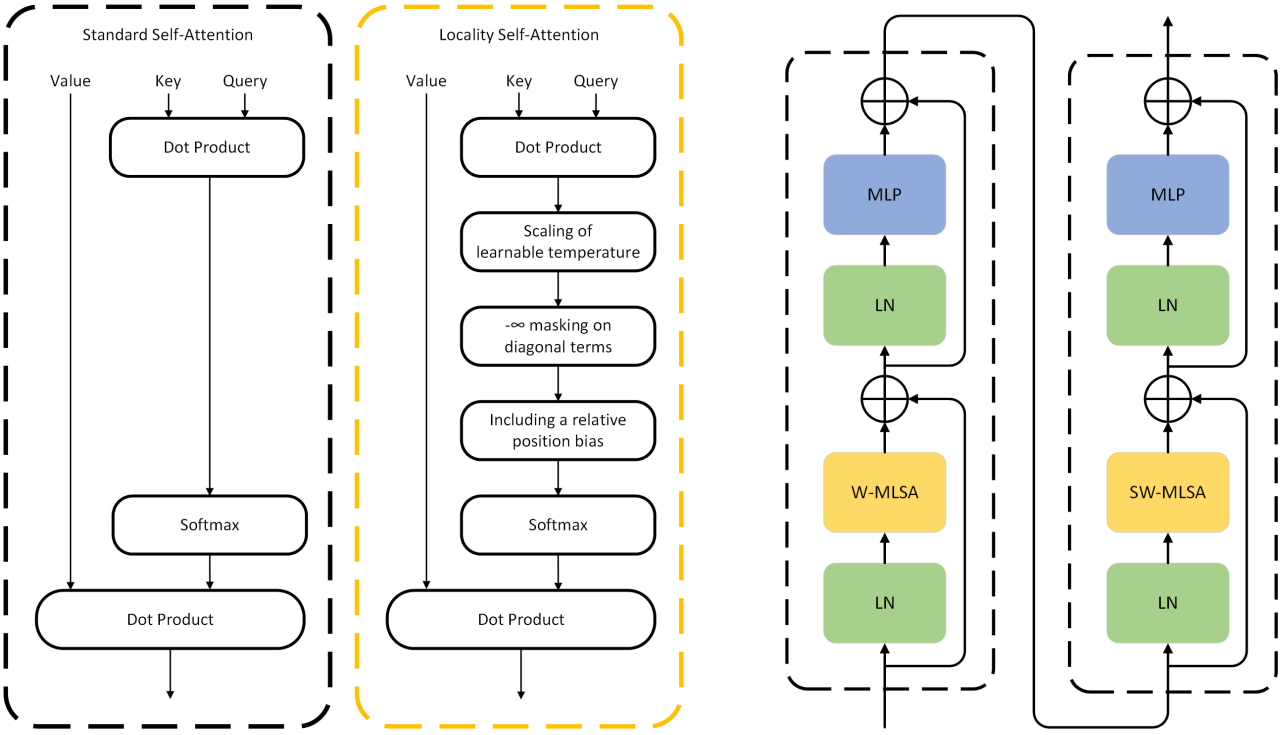
Where R^M represents the masked similarity matrix. i and j indicate the row and column index of the similarity matrix R.

After diagonal masking, the learnable temperature scaling is applied for allowing the model to determine the softmax temperature by itself during the learning phase. As the attention mechanism is used in the SW-MSA and SW-MSA of the backbone Swin Transformer, we also include a relative position bias B for sustaining the backbone architecture. Finally, the attention score matrix is attained through the softmax operation. And the self-attention matrix is acquired by the dot product of the attention score matrix and Value:

$$Attention(Q, K, V) = softmax\left(\frac{R^M}{\tau} + B\right)V \tag{8}$$

Where V is the learnable linear projection of Value, and τ is the learnable temperature.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 June 2023 doi:10.20944/preprints202306.0079.v2



(a) The Standard Self-Attention (left) and LSA (right)

(b) Two Successive L Swin Transformer Blocks

Figure 3. (a) The comparison between the standard self-attention mechanism and the locality self-attention mechanism; (b) two successive L Swin Transformer Blocks. W-MLSA and SW-MLSA are multi-head locality self-attention modules with regular and shifted windowing configurations, respectively.

2.4. Pseudo Labeling

As ground-truth labels (the onset, offset, and apex frame indices) only provide the status label of a given frame, which does not correspond to the optical flow features that carry motion information between frames, we utilize the pseudo labeling presented by Liong et al. [25] in the training phase. Firstly, the sliding window W_i which denotes the interval $[F_i, F_{(i+k)}]$ is scanned across each video. Subsequently, the function g is applied to acquire the pseudo label $\hat{l}$ for each sliding window calculated from the Intersection over Union (IoU) method which compares the sliding window W and the ground-truth interval $W_{grondTruth}$:

$$W_{grondTruth} = \left[F_{onset}, F_{offset}\right] \tag{9}$$

$$IoU = \frac{W \cap W_{grondTruth}}{W \cup W_{grondTruth}} \tag{10}$$

$$g(IoU) = \begin{cases} 0, & IoU \leq 0 \\ 1, & IoU > 0 \end{cases} \tag{11}$$

Finally, the pseudo label sequence $\hat{L} = \{\hat{l}_i,\ for\ i = F_{start}, \ldots, F_{(end-k)}\}$ together with the pre-processed optical flow features are used to train the SL-Swin model. The process of pseudo labeling is shown in Figure 4.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 June 2023 doi:10.20944/preprints202306.0079.v2

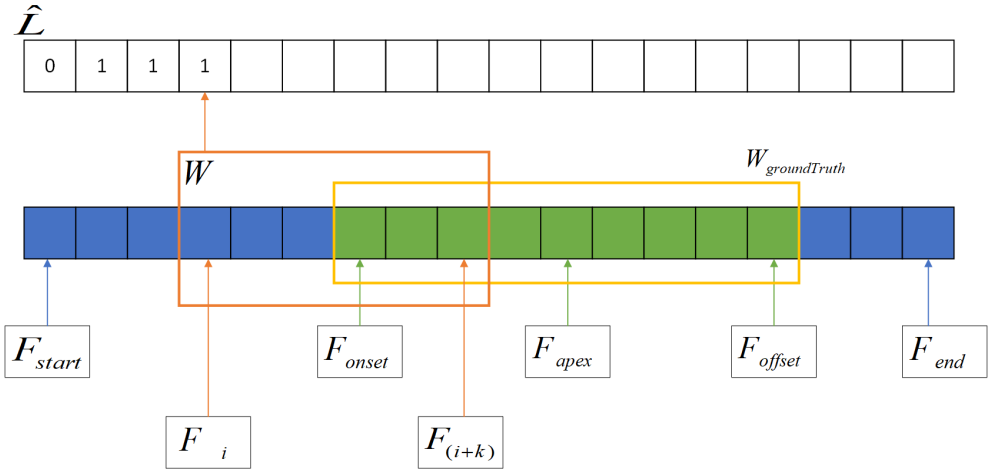


Figure 4. The pseudo labeling in a video.

2.5. Spotting

The predicted scores sequence S of every video is smoothed to get the smoothed scores sequence $\hat{S}$:

$$\hat{s}_i = \frac{1}{2k} \sum_{j=i-k}^{i+k-1} s_j \; for \; i = F_{(start+k)}, \ldots, F_{(end-k+1)} \tag{12}$$

Where s_j and $\hat{s}_i$ indicate the $j-th$ value in the raw predicted scores sequence S and $i-th$ value in the smoothed score sequence $\hat{S}$. In the smoothing scheme, the interval $[F_{(i-k)}, F_{(i+k-1)}]$ of the current frame F_i is averaged. Each smoothed score value $\hat{s}_i$ now represents the probability of the current frame F_i being within an expression interval.

Finally, we also employ the standard threshold and peak detection technique of [28] to spot the peaks in each video where the threshold is defined as:

$$T = \hat{S}_{mean} + t \times \left(\hat{S}_{max} - \hat{S}_{mean}\right) \tag{13}$$

Where $\hat{S}_{mean}$ and $\hat{S}_{max}$ is the average and maximum value of the smoothed scores sequence $\hat{S}$, and t which is a percentage parameter for tuning ranges from 0 to 1. We detect the local maximum (with the minimum distance of k between peaks) to find the peak frame $\hat{F}_p$ with the peak value $\hat{s}_p$. The spotted peak frame $\hat{F}_p$ is considered as the spotted apex frame and the spotted onset-offset interval $[F_{(p-k)}, F_{(p+k)}]$ for evaluation is obtained by extending k frames.

3. Results

This study conducts experiments in both the MEGC 2022 and MEGC 2021 spotting tasks. Note that the model is implemented separately for training and inference of macro- and micro-expressions. The code is available publicly for encouraging community use.

3.1. Evaluation Datasets

3.1.1. MEGC 2022 Datasets

The MEGC 2022 dataset solely provides the unseen test dataset for evaluation, consisting of 10 long videos: five clips cropped from different videos in CAS(ME)^3 [33] and five long videos from SAMM (SAMM Challenge dataset) [34], with frame rates of 30 fps and 200 fps, respectively.

Briefly, CAS(ME)^3 provides around 80 hours of videos with over 8,000,000 frames, including manually labeled 3,490 macro-expressions and 1,109 micro-expressions. This means that the clips from such a large dataset allow effective expression spotting approaches validation without database bias. Additionally, CAS(ME)^3 uses the mock crime

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 June 2023 doi:10.20944/preprints202306.0079.v2

paradigm, along with physiological and voice signals to elicit the micro-expression with high ecological validity, contributing to practical expression analysis.

SAMM, the origin of SAMM Challenge dataset, has the largest amount of different ethnicities, age distribution, and the highest resolutions among all expression datasets currently publicly available. Therefore, the five long videos from this dataset are more representative of a population, and expressions induced from different people could be considered acquired in a non-laboratory environment containing varieties of emotional responses.

Consequently, to a certain extent, the results of these 10 long videos reflect how efficiently the approach spots expressions in the real-world scenario. However, ground-truth labels are not released for these two test datasets, and the evaluation was conducted by the grand challenge system (https://megc2022.grand-challenge.org).

3.1.2. MEGC 2021 Datasets

The MEGC 2021 provides two datasets for training and evaluation, namely CAS(ME)^2 [35] and SAMM Long Videos [32] [36]. Both datasets are fully annotated with onset, apex, and offset by professional coders.

Briefly, CAS(ME)^2, the first dataset to contain both macro-expressions and micro-expressions from the same participants and under the same experimental conditions, includes 98 long videos consisting of 300 macro-expressions and 57 micro-expressions captured from 22 subjects. The resolution and frame rate of this dataset are (640 $\times$ 480) and 30 fps. In addition, researchers used the elicitation procedure that has been proven valid in previous work [37] to induce both macro-expressions and micro-expressions. Also, participants were asked to neutralize their facial expressions while watching emotion-evoking videos. These two procedures make makes all expression samples ecologically valid and dynamic. Besides, the participants were asked to watch the videos of their recorded facial expressions and offer a self-report on each expression, which excludes emotion-irrelevant facial movements and results in pure expression samples. In our experiments conducted on CAS(ME)^2 dataset, frames from the video "0503unnyfarting" of the subject "s23" in the "rawpic" folder have no annotation in the Excel file. Consequently, we exclude this video and concern the other 298 macro-expressions in our experiments.

SAMM Long Videos is an extension of SAMM [34] with 147 long videos (consisting of 343 macro-expressions, and 159 micro-expressions) captured from 32 subjects. Compared to CAS(ME)^2, the SAMM Long Videos dataset whose resolution (2040 $\times$ 1088) and frame rate (200 fps) are higher has more long videos and expressions, particularly the micro-expressions. Additionally, labels of macro-movements and micro-movements are provided in this dataset for indicating not only facial expressions but also other facial movements, such as eye blinks. However, 12 macro-expression samples are omitted in our experiments due to the ambiguous onset annotation.

3.2. Performance Metrics

We use the standard Intersection over Union (IoU) method for evaluating our spotting approach, consistent with the spotting tasks in MEGC 2021 and MEGC 2022. We compare the spotted interval $W_{spotted}$ with the ground-truth interval $W_{grondTruth}$, and consider it a True Positive (TP) when the following condition is met:

$$\frac{W_{spotted} \cap W_{grondTruth}}{W_{spotted} \cup W_{grondTruth}} \geq J \tag{14}$$

Where J is set to 0.5. Otherwise, the spotted interval $W_{spotted}$ is considered a False Positive (FP) result. Besides, the $W_{grondTruth}$ which fails to be spotted is reckoned as a False Negative (FN). Subsequently, we calculate the Precision as well as the Recall. The Precision which is obtained based on Equation 15 measures the approach's accuracy in identifying a spotted interval as an expression interval. The Recall which is calculated from Equation 16

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 June 2023 doi:10.20944/preprints202306.0079.v2

indicates how accurately the approach is able to identify the spotted intervals that actually contain expressions out of all spotted intervals.

$$Precision = \frac{TP}{TP + FP} \tag{15}$$

$$Recall = \frac{TP}{TP + FN} \tag{16}$$

Finally, we use the F1-score to evaluate the performance of the Macro-Expression (MaE), Micro-Expression (ME) spotting approaches, and the overall analysis. Notably, the approaches and evaluations for MaE and ME spotting are conducted separately.

$$F1 - score = \frac{2 \times Precision \times Recall}{Precision + Recall} \tag{17}$$

3.3. Settings

In the experiments of MEGC 2022, the model is trained on CAS(ME)^2 and SAMM Long Videos, respectively, and evaluated on CAS(ME)^3 and SAMM Challenge. The spotted MaE and ME intervals are then submitted to the grand challenge system (https://megc2022.grand-challenge.org) to obtain the result. For MEGC 2021, we employ leave-one-subject-out (LOSO) cross-validation to eliminate subject bias and ensure all samples are evaluated.

Parameter k (half of the average length of an expression interval) is computed to be $\{6, 18\}$ for CAS(ME)^2 as well as CAS(ME)^3, and $\{37, 169\}$ for SAMM Long Videos as well as SAMM Challenge (smaller value for micro-expression, larger value for macro-expression). For peak detection in the spotting procedure, we select $t = 0.60$ for both MEGC 2022 and MEGC 2021.

Note that there are different versions of the backbone model Swin Transformer. We select the tiny version, called Swin-T, as the backbone of our model, which is about $0.25\times$ the model size and computational complexity of the base Swin Transformer (Swin-B). The model we use in all experiments is called SL-Swin-T with the SPT and LSA being applied to the Swin-T. The window size is $M = 7$. The query dimension of each head is set to $d = 32$, and the expansion layer of each MLP is $\alpha = 4$. The other architecture hyper-parameters of the SL-Swin-T are the channel number of the hidden layers in "Stage 1" $C = 96$, layer numbers $= \{2, 2, 6, 2\}$.

In our experiments, the model is trained on an NVIDIA GTX 2080 Ti. The number of epochs is set to 25 and we apply SGD optimizer with a learning rate of 5×10^{-4}. In the training phase, we sample one of every two non-expression frames. To address the small sample size problem during micro-expression training, we also apply data augmentation techniques that include Gaussian blur (with a kernel size of 7×7), and adding random Gaussian noise ($N(0, 1)$), and horizontal flip.

3.4. Permormances

The results of our approach in the MEGC 2022 spotting task are shown in Table 1. And Table 2 compares the results of our approach and other approaches which are categorized into traditional approaches and deep learning approaches. The discussion about the results as well as the detail of the MEGC 2021 spotting task is shown in the Discussion section.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 June 2023 doi:10.20944/preprints202306.0079.v2

3.4.1. MEGC 2022 Spotting Task

Table 1. Performance comparison of our approach in the MEGC 2022 spotting task.

Approaches	CAS(ME)^3 Challenge			SAMM Challenge			Overall		
	Precision	Recall	F1-score	Precision	Recall	F1-score	Precision	Recall	F1-score
Baseline [15]	0.4000	0.1111	0.1739	0.0845	0.1935	0.1176	0.1235	0.1493	0.1351
Swin-T	0.1521	0.1944	0.1707	0.6380	0.0967	0.0769	0.1075	0.1492	0.1250
Ours	0.1944	0.1944	0.1944	0.0689	0.1290	0.0898	0.1170	0.1641	0.1366

3.4.2. MEGC 2021 Spotting Task

Table 2. Performance comparison of our approach against the others in the MEGC 2021 spotting task (F1-score)

Dataset	CAS(ME)^2			SAMM Long Videos		
Approaches	Macro	Micro	Overall	Macro	Micro	Overall
Traditional Approaches						
He et al. [38]	0.1196	0.0082	0.0376	0.0629	0.0364	0.0445
Zhang et al. [8]	0.2131	0.0547	0.1403	0.0725	0.1331	0.0999
He et al. [39]	0.3782	0.1965	0.3436	0.4149	0.2162	0.3638
Deep Learning Approaches						
Baseline [15]	0.2145	0.0714	0.1675	0.1595	0.0466	0.1084
Yand et al. [7]	0.2599	0.0339	0.2118	0.3553	0.1155	0.2736
Yu et al. [40]	0.3800	0.0630	0.3270	0.3360	0.2180	0.2900
Liong et al. [25]	0.2410	0.1173	0.2022	0.2169	0.1520	0.1881
Liong et al. [41]	0.4104	0.0808	0.3250	0.2810	0.1310	0.2380
Ours	0.2236	0.0879	0.1824	0.1675	0.1044	0.1357

4. Discussion

4.1. MEGC 2022 Spotting Task

Table 1 shows the results of our approach in the MEGC 2022 spotting task. We outperform the baseline approach by obtaining an F1-score of 0.1944 on the CAS(ME)^3 dataset and achieving an overall F1-score of 0.1366. By examining the results for the CAS(ME)^3 dataset in detail, our approach achieves a higher recall while having a lower precision. This effect is attributed to the smaller number of false spotted intervals, resulting in a smaller number of False Negatives (FNs). Compared to the approach that only uses the tiny version of the backbone Swin Transformer without SPT and LSA, which is recorded as Swin-T, our approach (SL-Swin-T) presents better results across all indicators, which indicates that the application of SPT and LSA improves the model generalization ability.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 June 2023 doi:10.20944/preprints202306.0079.v2

4.2. MEGC 2021 Spotting Task

Table 3. Detail of the SL-Swin-T in the MEGC 2021 spotting task

Dataset	CAS(ME)^2			SAMM Long Videos			Overall
Expression	MaE	ME	Overall	MaE	ME	Overall	
Total	298	57	355	331	159	490	845
TP	70	12	82	52	33	85	167
FP	258	204	462	238	440	678	1140
FN	228	45	273	279	126	405	678
Precision	0.2134	0.0556	0.1507	0.1793	0.0698	0.1114	0.1278
Recall	0.2349	0.2105	0.2310	0.1571	0.2075	0.1735	0.1976
F1-score	0.2236	0.0879	0.1824	0.1675	0.1044	0.1357	0.1552

For comparison, Table 2 shows the results of our approach against other approaches which are categorized into traditional and deep learning approaches. Overall, our approach outperforms the baseline and proves that the Transformer-based model could perform competitively compared to the model based on Convolution Neural Networks.

Our approach outperforms all traditional approaches except for the state-of-the-art approach proposed by He et al. [39]. Among the deep learning approaches, our approach remains competitive especially on ME spotting for CAS(ME)^2 dataset with the F1-score of 0.0879, only behind the approach proposed by Liong et al. [25] which spots a larger amount of TPs.

4.3. Ablation Studies

We conduct experiments to provide a thorough examination of our approach, focusing on network construction, labeling function, and feature sizes. Experiments are conducted on the CAS(ME)^2 dataset, using similar settings as the MEGC 2021 spotting task.

4.3.1. Network Architecture

To evaluate the effectiveness of the self-attention mechanism, SPT, and LSA, we conducted a similar experiment setting using different combinations of network architecture, SPT, and LSA. S indicates that SPT is applied to the network, L indicates that LSA is applied to the network, and SL indicates that both SPT and LSA are applied. Table 4 displays the experimental results of the various network architectures. According to our findings, SPT and LSA collectively enhance the network's performance on small-size datasets, specifically ME spotting of CAS(ME)^2. It is worth noting that the Swin-T model or the models that are based on it are only approximately 0.25$\times$ the model size and computational complexity of the ViT-B and the SL-ViT-B.

Table 4. Performance comparison of our model (SL-Swin-T) against other Transformer-based models (F1-score)

Network Architecture	CAS(ME)^2		
	MaE	ME	Overall
ViT-B	0.2125	0.0158	0.1415
SL-ViT-B	0.2071	0.0940	0.1738
Swin-T	0.2110	0.0783	0.1663
S-Swin-T	0.2351	0.0749	0.1685
L-Swin-T	0.2378	0.0493	0.1765
SL-Swin-T	0.2236	0.0879	0.1824

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 June 2023 doi:10.20944/preprints202306.0079.v2

4.3.2. Labeling

An ablation study is carried out on the original labeling and pseudo-labeling functions to investigate their impact on spotting when modeling the task as a regression problem. We set the parameter $t = 0.60$ and compare the result on the SL-Swin-T model using original labeling and pseudo labeling separately. The results are shown in 5. We observe that applying pseudo labeling reduces the amount of False Positives (FPs), thus enhancing precision and F1-score, particularly in the overall analysis.

Table 5. Performance comparison of pseudo labeling and original labeling

Dataset		CAS(ME)^2					
Expression	Labeling	TP	FP	FN	Precision	Recall	F1-score
MaE	Original	71	264	227	0.2119	0.2383	0.2243
	Pseudo	70	258	228	0.2134	0.2349	0.2236
ME	Original	14	262	43	0.0507	0.2456	0.0841
	Pseudo	12	204	45	0.0556	0.2105	0.0879
Overall	Original	85	526	270	0.1391	0.2394	0.1760
	Pseudo	82	462	273	0.1507	0.2310	0.1824

4.3.3. Features Size

In our approach, the SL-Swin-T model takes optical flow features $(u, v, |\epsilon|)$ of size $(42, 42, 3)$ as input. Due to the Patch Merging in each stage of the model, the feature map is downsampled by a rate of 2, leading to only $\frac{42}{48} \times \frac{42}{48} = 0.875 \times 0.875$ pixels from the original optical flow features in "Stage 4". To accommodate the windowing configuration, we apply padding when the feature map size is not an integer multiple of the window size $M = 7$. With the hierarchical architecture and self-attention computation within windows, the Swin Transformer has linear computational complexity to image size. This makes the Swin Transformer suitable for processing high-resolution images, in contrast to previous Transformer based architectures which produce feature maps of a single resolution and have quadratic complexity. Hence, we double the hyper-parameters in feature extraction and pre-processing, resulting in the size of $(u, v, |\epsilon|)$ being $(84, 84, 3)$ and allowing $\frac{84}{48} \times \frac{84}{48} = 1.75 \times 1.75$ pixels from the original optical flow features in "Stage 4" of the model. The hyper-parameters for the SL-Swin-T model and training configuration remain unchanged and the spotting parameter t is also set to 0.60 for comparison. Experimental results for ME spotting are presented in Table 6, demonstrating that the model performs better across all indicators, with particularly strong results in terms of the F1-score.

Table 6. Performance comparison of optical flow features of size

Dataset		CAS(ME)^2					
Expression	Features of Size	TP	FP	FN	Precision	Recall	F1-score
ME	(42, 42, 3)	12	204	45	0.0556	0.2105	0.0879
	(84, 84, 3)	13	190	44	0.0640	0.2281	0.1000

4.4. Limitations and Future Work

While our approach demonstrated promising results, we also recognize its limitations and explore potential areas for future research. Firstly, although we built our model upon the tiny version of the Swin Transformer, it is nevertheless considerably large and more complex than CNN-based models, which poses challenges in reproducing results and makes the LOSO experiments time-consuming. Particularly, the ME spotting experiment on the SAMM Long Videos dataset in the MEGC 2021 spotting task, while the MaE spotting experiment requires another week. Moreover, while traditional approachesi could provide detailed explanations for the occurrence of an expression, our twelve-layer model functions

 doi:10.20944/preprints202306.0079.v2

roughly like a black box. Therefore, it is essential to find an effective method for interpreting what the model learns from the training data. This will help to improve the feature extraction and pre-processing.

Secondly, Our model's performance of expression spotting is not as appealing as in other tasks. This may be attributed to its sensitive nature to training configurations such as batch size, epochs, and learning rate. Hence, we assume that our tuning does not fully show the advantage of the application of both SPT and LSA. Consequently, to optimize its performance, we suggest considering fine-tuning techniques such as pre-training the model on other datasets or experimenting different loss and optimization functions specially designed for small-size datasets.

Thirdly, although increasing the input features size to $(84, 84, 3)$ has been proven to enhance the model performance, it is still much smaller than the Swin Transformer's assumed input size of 224×224. Consequently, it is reasonable to anticipate that utilizing a larger input resolution may further improve the results. However, it is also notable that larger input resolutions mean higher computation complexity, requiring advanced hardware and more time for experiments. Simultaneously, employing models with a larger backbone, such as the small version of Swin Transformer (Swin-S) or the base version of Swin Transformer (Swin-B), may also lead to higher performance on high-resolution inputs but with higher computational costs. As such, it is crucial to strike a balance between performance gains and available resources.

5. Conclusions

In this paper, we propose a deep learning approach that uses a Transformer-based model called SL-Swin, which incorporates Shifted Patch Tokenization and Locality Self-Attention into the backbone network Swin Transformer, to predict a score indicating the probability of a frame being within an expression interval by analyzing optical flow features. The results demonstrate that our approach is capable of both MEGC 2022 and MEGC 2021 spotting tasks, which indicates the potential of our approach in accurately identifying expressions on small-size datasets and highlights the practicality of our approach in scenarios where large-scale labeled expression datasets may not be readily available. Our evaluation outcomes surpass the MEGC 2022 spotting baseline result, obtaining an overall F1-score of 0.1366. Additionally, our approach also performs well in the MEGC 2021 spotting task, achieving F1-scores of 0.1824 on CAS(ME)^2 and 0.1357 on SAMM Long Videos. Additionally, our work shows the potential of the Transformer-based model to achieve better performance with increasing data volumes. In the future, researchers could easily deepen the model or increase the size of the model and apply other techniques designed for small-size datasets to enhance the spotting performance. Furthermore, owing to the challenges in the ME annotation process, researchers can consider implementing self-supervised learning to enable the network to learn more meaningful latent representations.

Author Contributions: Conceptualization, Q.C, Q.Z. and E.H; methodology, E.H. and Q.C.; software, E.H.; validation, E.H. and Q.C.; formal analysis, E.H., Q.C. and Q.Z.; investigation, E.H., Q.C. and Q.Z.; resources, E.H. and Q.Z.; data curation, E.H. and Q.C.; writing—original draft preparation, E.H. and Q.C.; writing—review and editing, E.H., Q.C. and Q.Z.; visualization, E.H. and Q.C.; supervision, Q.Z.; project administration, Q.Z.; funding acquisition, Q.Z. All authors have read and agreed to the published version of the manuscript.

Funding: The research was supported by the Special Construction Fund of Faculty of Engineering (no. 46201503).

Data Availability Statement: All research datasets in this article are available. The datasets for MEGC 2022, which are SAMM Challenge and CAS(ME)^3, are available from https://megc2022.github.io/challenge.html. As for the datasets for MEGC 2021, the CAS(ME)^2 dataset is available from http://casme.psych.ac.cn/casme/c3 and the SAMM Long Videos dataset is available from http://www2.docm.mmu.ac.uk/STAFF/M.Yap/dataset.php.

Conflicts of Interest: The authors declare no conflict of interest.

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 June 2023 doi:10.20944/preprints202306.0079.v2

Abbreviations

The following abbreviations are used in this manuscript:

MEGC 2022	Facial Micro-Expression Grand Challenge 2022
MEGC 2021	Facial Micro-Expression Grand Challenge 2021
MaE	Macro-Expression
ME	Micro-Expression
SPT	Shifted Patch Tokenization
LSA	Locality Self-Attention

Appendix A

To encourage reproducibility in the community, the code is made publicly available at https://github.com/eddiehe99/pytorch-expression-spotting and https://github.com/eddiehe99/tensorflow-expression-spotting.

Appendix B

In the cropping process, we tried four detectors, that is the Haar Cascade face detector in OpenCV, the DNN face detector in OpenCV, the HoG face detector in Dlib, and the CNN face detector in Dlib. Among these detectors, we choose the DNN face detector in OpenCV which remains the highest consistency when cropping the facial region in a video with thousands of frames. Figure B1 shows how the facial region is cropped from the raw picture of the CAS(ME)^2 dataset.

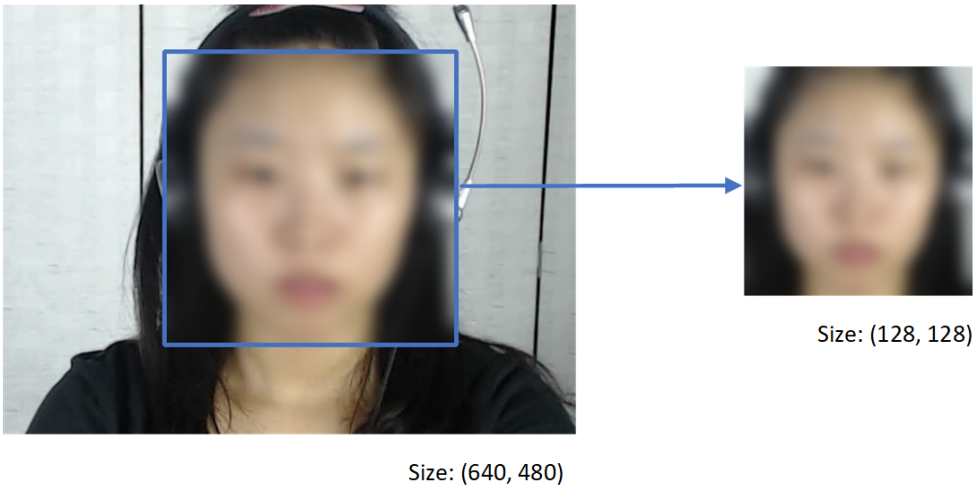


Figure B1. An example of the cropping process.

Appendix C

Figure C1 illustrates an example of ME spotting in the 32_0508funnydunkey video which belongs to the s32 subject of the CAS(ME)^2 dataset, where three predicted apex frames are spotted. For clear display, it only shows the spotted apex frames. As described in 2.5, the spotted interval is obtained by extending k frames to the spotted apex frame and is considered a True Positive (TP) if it satisfied Equation 14. In this case, the second spotted interval extended from the second spotted apex frame is considered as a TP. Conversely, the first and third spotted intervals are considered False Positives (FPs).

Preprints (www.preprints.org) | NOT PEER-REVIEWED | Posted: 8 June 2023 doi:10.20944/preprints202306.0079.v2

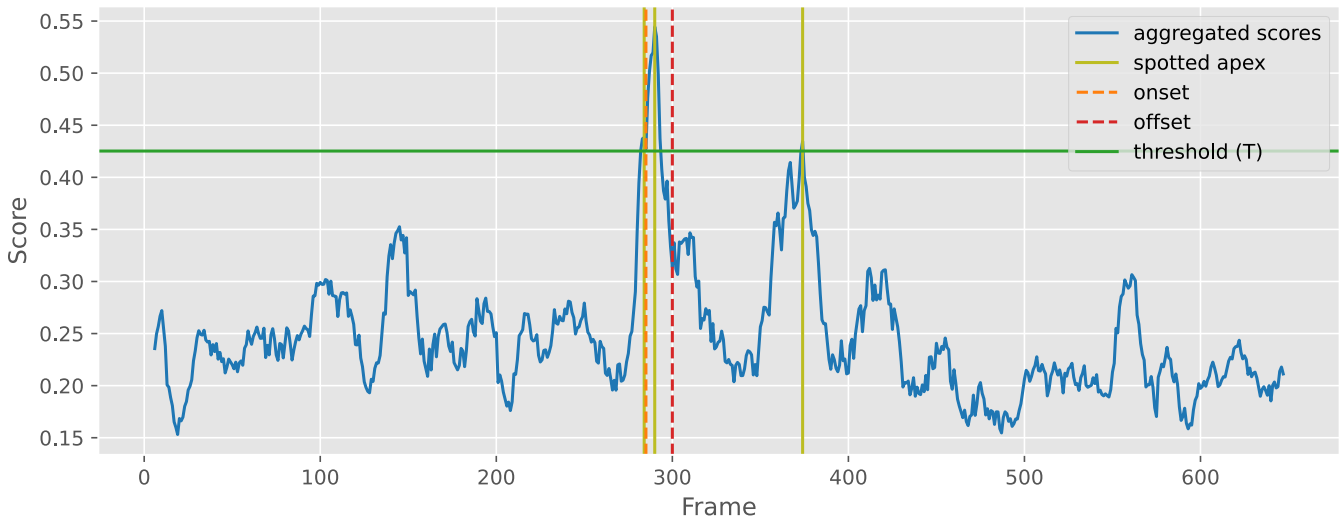


Figure C1. The ME spotting process in the 32_0508funnydunkey video.

References

1. Yan, W.J.; Wu, Q.; Liang, J.; Chen, Y.H.; Fu, X. How Fast are the Leaked Facial Expressions: The Duration of Micro-Expressions. *Journal of Nonverbal Behavior* **2013**, *37*, 217–230. https://doi.org/10.1007/s10919-013-0159-8.
2. Valstar, M.F.; Pantic, M. Fully Automatic Recognition of the Temporal Phases of Facial Actions. *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)* **2012**, *42*, 28–43. https://doi.org/10.1109/TSMCB.2011.2163710.
3. Ben, X.; Ren, Y.; Zhang, J.; Wang, S.; Kpalma, K.; Meng, W.; Liu, Y. Video-Based Facial Micro-Expression Analysis: A Survey of Datasets, Features and Algorithms. *IEEE TRANSACTIONS ON PATTERN ANALYSIS AND MACHINE INTELLIGENCE* **2022**, *44*, 5826–5846. https://doi.org/10.1109/TPAMI.2021.3067464.
4. Wang, S.; Wu, S.; Qian, X.; Li, J.; Fu, X. A main directional maximal difference analysis for spotting facial movements from long-term videos. *NEUROCOMPUTING* **2017**, *230*, 382–389. https://doi.org/10.1016/j.neucom.2016.12.034.
5. Yang, B.; Wu, J.; Zhou, Z.; Komiya, M.; Kishimoto, K.; Xu, J.; Nonaka, K.; Horiuchi, T.; Komorita, S.; Hattori, G.; et al. Facial Action Unit-Based Deep Learning Framework for Spotting Macro- and Micro-Expressions in Long Video Sequences. In Proceedings of the Proceedings of the 29th ACM International Conference on Multimedia; Association for Computing Machinery: New York, NY, USA, 2021; MM '21, pp. 4794–4798. event-place: Virtual Event, China, https://doi.org/10.1145/3474085.3479209.
6. Davison, A.K.; Yap, M.H.; Lansley, C. Micro-Facial Movement Detection Using Individualised Baselines and Histogram-Based Descriptors. In Proceedings of the 2015 IEEE International Conference on Systems, Man, and Cybernetics, 2015, pp. 1864–1869. https://doi.org/10.1109/SMC.2015.326.
7. Duque, C.A.; Alata, O.; Emonet, R.; Legrand, A.C.; Konik, H. Micro-Expression Spotting Using the Riesz Pyramid. In Proceedings of the 2018 IEEE Winter Conference on Applications of Computer Vision (WACV), 2018, pp. 66–74. https://doi.org/10.1109/WACV.2018.00014.
8. Zhang, L.W.; Li, J.; Wang, S.J.; Duan, X.H.; Yan, W.J.; Xie, H.Y.; Huang, S.C. Spatio-temporal fusion for Macro- and Micro-expression Spotting in Long Video Sequences. In Proceedings of the 2020 15th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2020), 2020, pp. 734–741. https://doi.org/10.1109/FG47880.2020.00037.
9. LI, J.; Wang, S.J.; Yap, M.H.; See, J.; Hong, X.; Li, X. MEGC2020 - The Third Facial Micro-Expression Grand Challenge. In Proceedings of the 2020 15th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2020), 2020, pp. 777–780. https://doi.org/10.1109/FG47880.2020.00035.
10. Yu, J.; Cai, Z.; Liu, Z.; Xie, G.; He, P. Facial Expression Spotting Based on Optical Flow Features. In Proceedings of the Proceedings of the 30th ACM International Conference on Multimedia; Association for Computing Machinery: New York, NY, USA, 2022; MM '22, pp. 7205–7209. event-place: Lisboa, Portugal, https://doi.org/10.1145/3503161.3551608.
11. Zhang, Z.; Chen, T.; Meng, H.; Liu, G.; Fu, X. SMEConvNet: A Convolutional Neural Network for Spotting Spontaneous Facial Micro-Expression From Long Videos. *IEEE Access* **2018**, *6*, 71143–71151. https://doi.org/10.1109/ACCESS.2018.2879485.
12. Pan, H.; Xie, L.; Wang, Z. Local Bilinear Convolutional Neural Network for Spotting Macro- and Micro-expression Intervals in Long Video Sequences. In Proceedings of the 2020 15th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2020), 2020, pp. 749–753. https://doi.org/10.1109/FG47880.2020.00052.
13. Li, J.; Yap, M.H.; Cheng, W.H.; See, J.; Hong, X.; Li, X.; Wang, S.J. FME'21: 1st Workshop on Facial Micro-Expression: Advanced Techniques for Facial Expressions Generation and Spotting. In Proceedings of the Proceedings of the 29th ACM International Conference on Multimedia; Association for Computing Machinery: New York, NY, USA, 2021; MM '21, pp. 5700–5701. event-place: Virtual Event, China, https://doi.org/10.1145/3474085.3478579.

14. Li, J.; Yap, M.H.; Cheng, W.H.; See, J.; Hong, X.; Li, X.; Wang, S.J.; Davison, A.K.; Li, Y.; Dong, Z. MEGC2022: ACM Multimedia 2022 Micro-Expression Grand Challenge. In Proceedings of the Proceedings of the 30th ACM International Conference on Multimedia; Association for Computing Machinery: New York, NY, USA, 2022; MM '22, pp. 7170–7174. event-place: Lisboa, Portugal, https://doi.org/10.1145/3503161.3551601.
15. Yap, C.H.; Yap, M.H.; Davison, A.; Kendrick, C.; Li, J.; Wang, S.J.; Cunningham, R. 3D-CNN for Facial Micro- and Macro-Expression Spotting on Long Video Sequences Using Temporal Oriented Reference Frame. In Proceedings of the Proceedings of the 30th ACM International Conference on Multimedia; Association for Computing Machinery: New York, NY, USA, 2022; MM '22, pp. 7016–7020. event-place: Lisboa, Portugal, https://doi.org/10.1145/3503161.3551570.
16. Verburg, M.; Menkovski, V. Micro-expression detection in long videos using optical flow and recurrent neural networks. In Proceedings of the 2019 14th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2019), 2019, pp. 1–6. https://doi.org/10.1109/FG.2019.8756588.
17. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, \.; Polosukhin, I. Attention is All You Need. In Proceedings of the Proceedings of the 31st International Conference on Neural Information Processing Systems; Curran Associates Inc.: Red Hook, NY, USA, 2017; NIPS'17, pp. 6000–6010. event-place: Long Beach, California, USA, https://doi.org/10.48550/arXiv.1706.03762.
18. Dosovitskiy, A.; Beyer, L.; Kolesnikov, A.; Weissenborn, D.; Zhai, X.; Unterthiner, T.; Dehghani, M.; Minderer, M.; Heigold, G.; Gelly, S.; et al. An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale. *arXiv* **2020**, p. 21 pp. Type: Journal Paper, https://doi.org/https://doi.org/10.48550/arXiv.2010.11929.
19. J. Deng.; W. Dong.; R. Socher.; L. -J. Li.; Kai Li.; Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In Proceedings of the 2009 IEEE Conference on Computer Vision and Pattern Recognition, 2009, pp. 248–255. Journal Abbreviation: 2009 IEEE Conference on Computer Vision and Pattern Recognition, https://doi.org/10.1109/CVPR.2009.5206848.
20. Pan, H.; Xie, L.; Wang, Z. Spatio-temporal convolutional emotional attention network for spotting macro- and micro-expression intervals in long video sequences. *Pattern Recognition Letters* **2022**, *162*, 89–96. https://doi.org/10.1016/j.patrec.2022.09.008.
21. Devlin, J.; Chang, M.W.; Lee, K.; Toutanova, K. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. In Proceedings of the Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers); Association for Computational Linguistics: Minneapolis, Minnesota, 2019; pp. 4171–4186. https://doi.org/10.18653/v1/N19-1423.
22. Zhou, Y.; Song, Y.; Chen, L.; Chen, Y.; Ben, X.; Cao, Y. A Novel Micro-Expression Detection Algorithm Based on BERT and 3DCNN. *Image Vision Comput.* **2022**, *119*. Place: USA Publisher: Butterworth-Heinemann, https://doi.org/10.1016/j.imavis.2022.104378.
23. Guo, X.; Zhang, X.; Li, L.; Xia, Z. Micro-expression spotting with multi-scale local transformer in long videos. *Pattern Recognition Letters* **2023**, *168*, 146–152. https://doi.org/10.1016/j.patrec.2023.03.012.
24. Liong, S.T.; Gan, Y.S.; See, J.; Khor, H.Q.; Huang, Y.C. Shallow Triple Stream Three-dimensional CNN (STSTNet) for Micro-expression Recognition. In Proceedings of the 2019 14th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2019), 2019, pp. 1–5. https://doi.org/10.1109/FG.2019.8756567.
25. G. -B. Liong.; J. See.; L. -K. Wong. Shallow Optical Flow Three-Stream CNN For Macro- And Micro-Expression Spotting From Long Videos. In Proceedings of the 2021 IEEE International Conference on Image Processing (ICIP), 2021, pp. 2643–2647. Journal Abbreviation: 2021 IEEE International Conference on Image Processing (ICIP), https://doi.org/10.1109/ICIP42928.2021.9506349.
26. Lee, S.; Lee, S.; Song, B. Improving Vision Transformers to Learn Small-Size Dataset From Scratch. *IEEE ACCESS* **2022**, *10*, 123212–123224. https://doi.org/10.1109/ACCESS.2022.3224044.
27. Z. Liu.; Y. Lin.; Y. Cao.; H. Hu.; Y. Wei.; Z. Zhang.; S. Lin.; B. Guo. Swin Transformer: Hierarchical Vision Transformer using Shifted Windows. In Proceedings of the 2021 IEEE/CVF International Conference on Computer Vision (ICCV), 2021, pp. 9992–10002. Journal Abbreviation: 2021 IEEE/CVF International Conference on Computer Vision (ICCV), https://doi.org/10.1109/ICCV48922.2021.00986.
28. Moilanen, A.; Zhao, G.; Pietikäinen, M. Spotting Rapid Facial Movements from Videos Using Appearance-Based Feature Difference Analysis. In Proceedings of the 2014 22nd International Conference on Pattern Recognition, 2014, pp. 1722–1727. https://doi.org/10.1109/ICPR.2014.303.
29. Zhao, Y.; Tong, X.; Zhu, Z.; Sheng, J.; Dai, L.; Xu, L.; Xia, X.; Jiang, Y.; Li, J. Rethinking Optical Flow Methods for Micro-Expression Spotting. In Proceedings of the Proceedings of the 30th ACM International Conference on Multimedia; Association for Computing Machinery: New York, NY, USA, 2022; MM '22, pp. 7175–7179. event-place: Lisboa, Portugal, https://doi.org/10.1145/3503161.3551602.
30. Shreve, M.; Brizzi, J.; Fefilatyev, S.; Luguev, T.; Goldgof, D.; Sarkar, S. Automatic expression spotting in videos. *Image and Vision Computing* **2014**, *32*, 476–486. https://doi.org/10.1016/j.imavis.2014.04.010.
31. Liong, S.T.; See, J.; Wong, K.; Phan, R.C.W. Automatic Micro-expression Recognition from Long Video Using a Single Spotted Apex. In Proceedings of the Computer Vision – ACCV 2016 Workshops; Chen, C.S.; Lu, J.; Ma, K.K., Eds.; Springer International Publishing: Cham, 2017; pp. 345–360. https://doi.org/10.1007/978-3-319-54427-4_26.
32. Yap, C.H.; Kendrick, C.; Yap, M.H. SAMM Long Videos: A Spontaneous Facial Micro- and Macro-Expressions Dataset. In Proceedings of the 2020 15th IEEE International Conference on Automatic Face and Gesture Recognition (FG 2020), 2020, pp. 771–776. https://doi.org/10.1109/FG47880.2020.00029.

33. Li, J.; Dong, Z.; Lu, S.; Wang, S.J.; Yan, W.J.; Ma, Y.; Liu, Y.; Huang, C.; Fu, X. CAS(ME)3: A Third Generation Facial Spontaneous Micro-Expression Database With Depth Information and High Ecological Validity. *IEEE Transactions on Pattern Analysis and Machine Intelligence* **2023**, *45*, 2782–2800. https://doi.org/10.1109/TPAMI.2022.3174895.
34. Davison, A.K.; Lansley, C.; Costen, N.; Tan, K.; Yap, M.H. SAMM: A Spontaneous Micro-Facial Movement Dataset. *IEEE Transactions on Affective Computing* **2018**, *9*, 116–129. https://doi.org/10.1109/TAFFC.2016.2573832.
35. Qu, F.; Wang, S.J.; Yan, W.J.; Li, H.; Wu, S.; Fu, X. CAS(ME)^2 : A Database for Spontaneous Macro-Expression and Micro-Expression Spotting and Recognition. *IEEE Transactions on Affective Computing* **2018**, *9*, 424–436. https://doi.org/10.1109/TAFFC.2017.2654440.
36. Davison, A.; Merghani, W.; Moi Hoon Yap. Objective classes for micro-facial expression recognition. *Journal of Imaging* **2018**, *4*, 119 (13 pp.)–119 (13 pp.). https://doi.org/10.3390/jimaging4100119.
37. Yan, W.J.; Li, X.; Wang, S.J.; Zhao, G.; Liu, Y.J.; Chen, Y.H.; Fu, X. CASME II: An Improved Spontaneous Micro-Expression Database and the Baseline Evaluation. *PLOS ONE* **2014**, *9*, e86041. Publisher: Public Library of Science, https://doi.org/10.1371/journal.pone.0086041.
38. He, Y.; Wang, S.; Li, J.; Yap, M. Spotting Macro- and Micro-expression Intervals in Long Video Sequences. In Proceedings of the Chinese Academy of Sciences; Struc, V.; GomezFernandez, F., Eds., 2020, pp. 742–748. https://doi.org/10.1109/FG47880.2020.00036.
39. He, Y.; Xu, Z.; Ma, L.; Li, H. Micro-expression spotting based on optical flow features. *Pattern Recognition Letters* **2022**, *163*, 57–64. https://doi.org/10.1016/j.patrec.2022.09.009.
40. Yu, W.W.; Jiang, J.; Li, Y.J. LSSNet: A Two-Stream Convolutional Neural Network for Spotting Macro- and Micro-Expression in Long Videos. In Proceedings of the Proceedings of the 29th ACM International Conference on Multimedia; Association for Computing Machinery: New York, NY, USA, 2021; MM '21, pp. 4745–4749. event-place: Virtual Event, China, https://doi.org/10.1145/3474085.3479215.
41. Liong, G.B.; Liong, S.T.; See, J.; Chan, C.S. MTSN: A Multi-Temporal Stream Network for Spotting Facial Macro- and Micro-Expression with Hard and Soft Pseudo-Labels. In Proceedings of the Proceedings of the 2nd Workshop on Facial Micro-Expression: Advanced Techniques for Multi-Modal Facial Expression Analysis; Association for Computing Machinery: New York, NY, USA, 2022; FME '22, pp. 3–10. event-place: Lisboa, Portugal, https://doi.org/10.1145/3552465.3555040.