applied sciences

MDPI

*Article*

# Late Fusion-Based Video Transformer for Facial Micro-Expression Recognition

**Jiuk Hong, Chaehyeon Lee and Heechul Jung ***

Department of Artificial Intelligence, Kyungpook National University, Daegu 37224, Korea; hong4497@knu.ac.kr (J.H.); 123456ccdd@knu.ac.kr (C.L.)
* Correspondence: heechul@knu.ac.kr; Tel.: +82-53-950-4558

**Abstract:** In this article, we propose a novel model for facial micro-expression (FME) recognition. The proposed model basically comprises a transformer, which is recently used for computer vision and has never been used for FME recognition. A transformer requires a huge amount of data compared to a convolution neural network. Then, we use motion features, such as optical flow and late fusion to complement the lack of FME dataset. The proposed method was verified and evaluated using the SMIC and CASME II datasets. Our approach achieved state-of-the-art (SOTA) performance of 0.7447 and 73.17% in SMIC in terms of unweighted F1 score (UF1) and accuracy (Acc.), respectively, which are 0.31 and 1.8% higher than previous SOTA. Furthermore, UF1 of 0.7106 and Acc. of 70.68% were shown in the CASME II experiment, which are comparable with SOTA.

**Keywords:** deep learning; image processing; facial micro-expression; emotion recognition; vision transformer



**Citation:** Hong, J.; Lee, C.; Jung, H. Late Fusion-Based Video Transformer for Facial Micro-Expression Recognition. *Appl. Sci.* **2022**, *12*, 1169. https://doi.org/10.3390/app12031169

Academic Editors: Wonjoon Kim, Sekyoung Youm and Sungbum Jun

Received: 6 November 2021
Accepted: 21 January 2022
Published: 23 January 2022



## 1. Introduction

Facial micro-expression (FME) often faintly occurs for 0.04–0.2 s when people try hiding their true feelings, unlike macro-expression appearing on the face from 0.75–2 s. Due to these characteristics of FME, it is cost-intensive to build an FME dataset and there are few FME datasets. In addition, several existing datasets, such as SMIC and CASME II [1,2], developed in a strict environment, have a small number of samples.

Because of this nature of FME, most early studies [1,3–6] used handcrafted features such as local binary patterns on three orthogonal planes and optical flow [7]. However, as deep learning began to gain prominence in computer vision, there have been many attempts [8–11] to combine deep neural networks with handcrafted features since a study [12] using convolution neural networks (CNNs) with long short-term memory model (LSTM) in FME recognition was conducted.

Recently, deep-learning methods have achieved the state of the art (SOTA) using a vision transformer model, with composed self-attention layer without CNN rather than using CNN in computer vision. Generally, the vision transformer model outperforms CNN when using transfer learning with pretrained weights using large number of data rather than training from scratch. Interestingly, a recent study [13], which injected CNN-like inductive biases [14], locality and pyramid structure, into transformer models, showed similar performance to CNN with scratch training on the ImageNet dataset.

However, to the best of our knowledge, no studies have applied a vision transformer to FME recognition. We assume the transformer's inductive bias, modeling relations between input patches might seem more suitable for FME recognition than the inductive bias of CNN, since the pattern of FME is subtle and appears only in a part of each frame in the video. Therefore, we propose an FME recognition model using a transformer and optical flow [7], which is a general feature to represent the motion of video, without pretrained weights using a large amount of data. We used optical flow as a motion feature to complement the lack of data [15].

Since FME datasets were captured by a high-speed camera, we thought the influence of the optical flow in FME recognition would be different from general video recognition. Therefore, in ablation we conducted various experiments about that influence and empirically found the proper way to use the optical flow. As a result, our proposed model achieves the SOTA in the SMIC [1] and comparable performance in the CASME II [2] (see Table 2).

## 2. Related Works

### 2.1. Prior Works of FME Recognition

Previous studies on FME were performed using handcrafted features. They can be summarized as follows: Li et al. used local binary pattern histograms from three orthogonal planes (LBP-TOP) to describe the spatiotemporal local textures from cropped face sequences for feature extraction [1], and interpolated video using a temporal interpolation model (TIM) [16]. Liong et al. proposed a feature extraction method using bi-weighted oriented optical flow (Bi-WOOF), variants of optical flow, to encode essential expressiveness of the apex frame and used only two images per video [4]. Wang et al. used the sparse part of Robust PCA to extract the subtle motion information of micro-expression and classified the local texture features of the information extracted by local spatiotemporal directional features [3]. Xiaobai et al. proposed a new unifying framework [5], where motion magnification is employed to counter the low intensity of MEs, for ME spotting and recognition. Yuan et al. designed a hierarchical spatial division scheme for spatiotemporal descriptor extraction to address difficulty with choosing an ideal division grid for different micro-expression samples [6].

However, since deep-learning methods have become de facto, studies in FME recognition have begun to use deep-learning methods: Devangini et al. proposed the first work to explore the possible use of deep learning for micro-expression recognition task. They solved the problem of lack of data using transfer learning from objects and facial expression-based CNN models [12]. Li et al. applied the 3D flow-based CNNs model, which flows consists of gray color information, and horizontal and vertical optical flow [8]. Xia et al. proposed a deep model, which is constituted of several recurrent convolutional layers. They exploited two types to extend the connectivity of convolutional networks across the temporal domain, in which the spatiotemporal deformations are modeled in views of facial appearance and geometry separately [9]. Choi et al. proposed a 2D landmark feature map (LFM) obtained by transforming face landmark information into 2D image information. They also proposed an LFM-based recognition method that is an integrated framework of CNN and LSTM [10]. Xuan et al. developed a multi-task learning (MTL) method to effectively leverage a side task: gender detection. Their method GEME [11] recognized micro-expressions by incorporating unique gender characteristics and subsequently improved the micro-expression recognition accuracy.

### 2.2. Vision Transformer

The transformer is a highly successful model in natural language processing and has recently been applied to computer vision. The best-known vision transformer is ViT [17], which replaces word tokens in a sentence with patch tokens in an image. The biggest difference between the transformer and CNN, which was the mainstream in computer vision, is that it uses self-attention operations, not convolution operations. Self-attention is an operation that allows each token to represent contextual information within the group to which it belongs, rather than representing an individual meaning. For that, self-attention converts the input token into individual query, key, and value, and calculates a scaled dot product [18] between them. Because self-attention models the relationship of patches within the image they belong to, unlike CNN, the vision transformer has a global receptive field. In addition, each query, key, and value depends on the input data, so unlike CNN, the transformer has a property of adaptive weight aggregation, making it more expressive. However, as it has a large capacity, it is necessary to learn with large

amounts of data to achieve good performance. Many studies have been proposed to achieve similar performance to CNN with the same amount of data. Among them, the Swin transformer [13] used in this paper is a study to solve the above problem by borrowing some of the pyramid structure and locality of CNN.

## 3. Proposed Method

Figure 1 depicts the structure of the proposed method. First, we linearly interpolate a different length video $x$ into a fixed length video $x_{\text{fix}}$. Next, we calculate optical flows $x_{\text{opt}}$ from $x_{\text{fix}}$ and then throw away the last frame of $x_{\text{fix}}$ to match the length of $x_{\text{fix}}$ and $x_{\text{opt}}$:

$$x = [i_1, i_2, i_3, \ldots, i_M], \quad \text{image } i \in \mathbb{R}^{H \times W \times C}, \tag{1}$$

$$x_{\text{fix}} = [i_1, i_2, i_3, \ldots, i_N], \quad \text{throw away } i_{N+1} \text{ for matching}, \tag{2}$$

$$x_{\text{opt}} = [o_1, o_2, o_3, \ldots, o_N], \quad \text{optical flow } o \in \mathbb{R}^{H \times W \times C}, \tag{3}$$

where the sequence length $M$ depends on the data sample, $N$ is the desired number of lengths, $(H, W)$ is the resolution of the video, and $C$ is the number of channels. We explain this preprocessing in detail later.

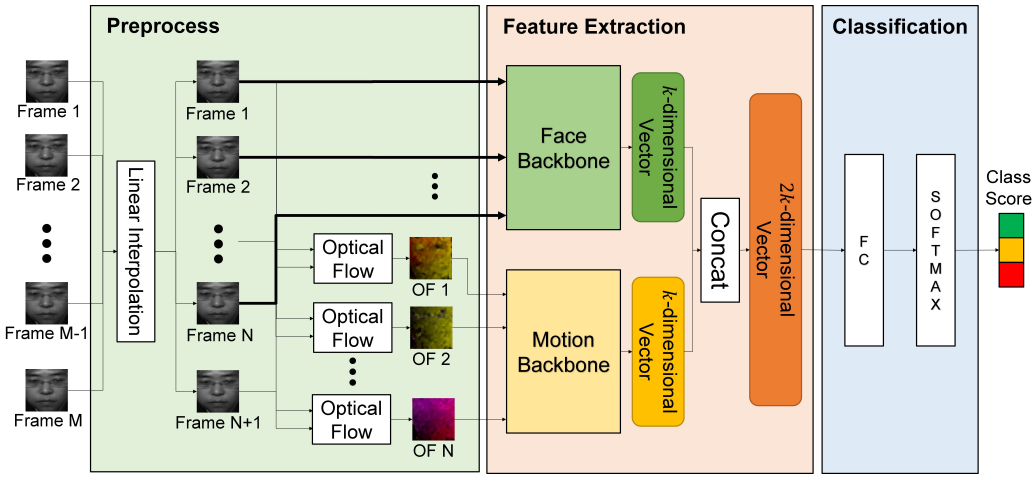


**Figure 1.** Structure of proposed method.

Afterward, grayscale or color images $x_{\text{fix}}$ and optical flows $x_{\text{opt}}$ are independently passed through two transformer backbones, face backbone $f$ and motion backbone $g$. These backbones have the same structure but do not share parameters to extract $k$-dimensional feature vectors, $z_{\text{face}}$ and $z_{\text{motion}}$. These vectors are then transformed to $z_{\text{fusion}}$ via concatenation:

$$z_{\text{face}} = f(x_{\text{fix}}), \quad z_{\text{face}} \in \mathbb{R}^k, \tag{4}$$

$$z_{\text{motion}} = g(x_{\text{opt}}), \quad z_{\text{motion}} \in \mathbb{R}^k, \tag{5}$$

$$z_{\text{fusion}} = \text{concat}([z_{\text{face}}, z_{\text{motion}}]), \quad z_{\text{fusion}} \in \mathbb{R}^{2k}. \tag{6}$$

Finally, we push $z_{\text{fusion}}$ into the classifier $h$ composed of the fully connected layer, followed by the SoftMax layer, to obtain a class score $s$ and use cross entropy loss using $t \in \mathbb{R}^c$ for training:

$$s = h(z_{\text{fusion}}), \quad s \in \mathbb{R}^{\text{c}}, \tag{7}$$

$$\mathcal{L}_{\text{ce}}(s, t) = -\sum_{i=1}^{c} t_i \log(s_i), \quad t \in \mathbb{R}^{\text{c}}, \tag{8}$$

where $c$ is the number of target classes and subscript $i$ means the position of the elements in the vector.

### 3.1. Preprocessing

The number of frames for each video must be the same to use it as a transformer input. A previous study [16] used TIM to equally interpolate the frames of each video. However, TIM's assumption that each frame is linearly independent is fragile when using an FME dataset because the video captured by a high-speed camera (100/200 fps) has a dimmer pattern compared to the normal-speed video (30 fps). In addition, TIM interpolates videos using singular value decomposition that require numerous computations to flatten video vectors.

Due to these limitations, we use linear interpolation. Linear interpolation is a method of curve-fitting using linear polynomials to construct new data points within the range of a discrete set of known data points. Table 1 shows that TIM requires additional computation and does not yield significant performance improvements compared to linear interpolation. We measured the time required for interpolation of 31 frames of video into 8 frames. The interpolation is executed using the CPU, where the RAM capacity is 377 GB, and the CPU is 64-core AMD EPYC 7702.

**Table 1.** TIM vs. linear interpolation using Early Fusion Video Transformer.

| Method | Interpolation Time [a] (ms, millisecond) | UF1 | UAR | Acc. (%) |
|---|---|---|---|---|
| TIM [16] | 602 ± 90.4 | 0.6516 | 0.6565 | 65.85 |
| Linear Interpolation | 22.3 ± 29.7 | 0.6590 | 0.6629 | 65.85 |

In this experiment, we set video length as 8 and do not use Late Fusion. [a] is the average time measured 500 iterations.

Then, we calculated an optical flow feature from the interpolated video. Optical flow is the pattern of apparent motion of objects, surfaces, and edges in a visual scene caused by the relative motion between an observer and a scene. We used dense optical flow obtained using the Farneback algorithm [19], which is a basic method of calculating optical flow. In the proposed method, we set $N$ as a quarter of the average video length on the target dataset.

### 3.2. Transformer Backbone

The Swin video transformer [20] is a model that performs well without a large amount of data using a Swin block composed of window or shifted-window multi-head self-attention, which solves the quadratic complexity problem of the transformer.

In the proposed method, we use Video Swin-B as the backbone. It consists of 4 stages, which have different Swin block numbers $\{2, 2, 18, 2\}$ for each stage. Since we handle a token as $2 \times 4 \times 4$ in the backbone, patch partition first reshapes the input video $N \times C \times H \times W$ into the patch tokens $(\frac{N}{2} \cdot \frac{H}{4} \cdot \frac{W}{4}) \times (2 \cdot 4 \cdot 4 \cdot C)$. Then, a linear embedding layer is applied to project each token to the dimension of 128 and each stage feeds previous tokens to the next stage through Swin blocks.

We expect that the face backbone models the spatial and temporal relationships between patches of all frames for FME recognition, and the motion backbone does same thing as the face backbone in terms of motion information using optical flows.

### 3.3. Late Fusion

In prior work [21], researchers called extracting features of each frame through one shared 2DCNN which could not model the temporal relations, and combining features before classification, Late Fusion (LF). In contrast to LF, they called extracting a combined feature of all frames through one 3DCNN early fusion (EF).

The definitions of LF and EF above are slightly different from our research. However, we used the names because the positions that combine each feature are the same. In

our research, we named LF the process that extracts two features individually from two different inputs, video and optical flows, through two different backbones and combines those features before classification. Additionally, we named EF the process that extracts a combined feature through one backbone from one input in which video and optical flows are concatenated channel-wise, and classifies the feature into emotional classes.

In the proposed method, we used LF even though EF has a low amount of computation due to its shared backbone. This is because we thought that extracting a feature from the concatenated input would degrade performance due to dependency, caused by the optical flows calculated from the video.

## 4. Experimental Results

### 4.1. Dataset and Metrics for the Whole Experiment

We used the SMIC [1] and the CASME II [2] dataset for the evaluation of the proposed method. There are 100 Hz 164 videos classified into positive (51), negative (70), and surprise (43) emotions, built by 16 subjects in SMIC. Each video shows the upper body is different in length, has 33.7 frames on average. The other dataset, CASME II, consists of 200 Hz 247 videos classified into disgust (64), happiness (32), others (99), repression (27), and surprise (25), built by 26 subjects. The average number of frames is 67.2. All video has three color channels: red, green, and blue.

Since the dataset has an imbalance distribution of emotion labels, we used three balanced metrics to reduce the bias: accuracy (Acc), unweighted average recall (UAR) [22], and unweighted F1 score (UF1) [23].

$$Acc. = \frac{\sum_{c=1}^{C} TP_c}{\sum_{c=1}^{C} N_c}, \tag{9}$$

$$UAR = \frac{1}{C} \sum_{c=1}^{C} \frac{TP_c}{N_c}, \tag{10}$$

$$UF1 = \frac{1}{C} \sum_{c=1}^{C} \frac{2TP_c}{2TP_c + FP_c + FN_c}, \tag{11}$$

where $C$ is the number of classes, and $N_c$ is the number of samples for each class. $TP$, $FN$, and $FP$ are the true positive, false negative and false positive, respectively.

### 4.2. Training Scheme

All models were trained on 1 GPU with 1 image per GPU. Specifically, we used a RTX A6000 48 GB. This means that batch size is 1. For backpropagation, we used the AdamW [24] optimizer of which betas are (0.9, 0.999) and weight decay is 0.05. The initial learning rate is $10^{-5}$.

Since each video shows the upper body, we cropped the face part using a face detection model [25] and resized its resolution as (224, 224). In addition, the FME appears in a very faint pattern and can be easily damaged if a strong change is applied. Thus, we used only simple augmentations such as random scaling and rotation in the range of $[0.9, 1.1]$ and $[-10°, 10°]$, and horizontal flip. Furthermore, since the class distribution of each dataset is unbalanced, an imbalanced sampler that matches the class distribution was used before data augmentation to train using the similar amount of data per epoch. We expected the sampler to reduce the bias of the dataset.

We trained the face backbone $f(\cdot)$, the motion backbone $g(\cdot)$, and the classifier $h(\cdot)$ using this dataset with the proposed method. We used same scheme for training models of the ablation study.

### 4.3. Evaluation Protocol

Each person has a different form of expression on their face. Therefore, to avoid person-dependent issues, the performance of the model is evaluated with leave-one-subject-

out (LOSO) cross-validation. LOSO measures the performance of the model using one of the subjects as a validation set and the rest as a training set. Then, we repeat the training and validation process as many times as the number of subjects in the dataset.

### *4.4. Performance of the Proposed Method*

Table 2 shows the performance of the proposed method compared to other studies on SMIC and CASME II. The numerical values of each method are taken from survey [26] or their own paper. We can find that our method achieves the best accuracy and UF1 on average in SMIC and shows comparable performance in CASME II. In the case of SMIC, the proposed method has an improvement of about 0.031 and 1.8% over previous SOTA. In the case of CASME II, the proposed method did not outperform the previous SOTA, but while the previous SOTAs, LFM and GEME, demanded additional complex methods for model learning, our model exhibits comparable performance without such a method.

**Table 2.** Comparison to other methods on SMIC and CASME II.

| Year | Method | SMIC | | CASME II | |
|---|---|---|---|---|---|
| | | UF1 | Acc. (%) | UF1 | Acc. (%) |
| 2013 | LBP-TOP [1] | - | 48.78 | - | - |
| 2014 | DLSTD [3] | - | 68.29 | - | 63.41 |
| 2016 | CNN-LSTM [12] | - | 53.6 | - | 47.3 |
| 2016 | BI-WOOF [4] | 0.6200 | 62.20 | 0.6100 | 57.89 |
| 2018 | HIGO [5] | - | 67.21 | - | 68.29 |
| 2018 | H-STLBP-IP [6] | 0.6126 | 60.78 | 0.6110 | 63.83 |
| 2019 | 3DCNN [8] | - | 55.49 | - | 59.11 |
| 2019 | STRCN [9] | 0.6950 | 72.30 | 0.7470 † | 80.30 † |
| 2020 | LFM [10] | 0.7134 | 71.34 | 0.7165 | 73.98 |
| 2021 | GEME [11] | 0.6158 | 64.63 | **0.7354** | **75.2** |
| 2021 | **Proposed *** | **0.7447** | **73.17** | 0.7106 | 70.68 |

Note: **Bold** show the best performance in each metric. † was measured using only 4 classes in CASME II. * From result of Late Fusion in Table 6.

Figure 2 shows the confusion matrices of the proposed method in the validation phase. In SMIC, the proposed model has difficulty classifying 'negative', especially misclassifying 'negative' as 'positive'. This is interesting because the number of samples labeled 'negative' is the largest, and other studies classify 'negative' relatively well. We believed that these results are due to using the imbalanced sampler. In CASME II, the proposed method is relatively poor at classifying 'others' and 'disgust', which have the most data samples.

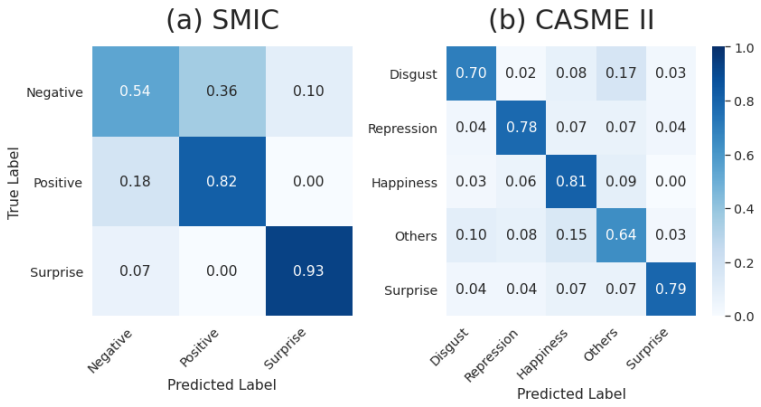


**Figure 2.** Validation confusion matrices of our method. Each matrix is measured using LOSO. (**a**) is the result of SMIC and (**b**) is the result of CASME II.

## 5. Ablation Study

Since we performed FME recognition using a transformer, we need to know the influence of optical flow and distinguish it from the influence of the transformer. Therefore,

we will compare the performance of CNN-based [27], transformer-based [28], and CNN-like transformer-based [20] models. The CNN-based model represents locality inductive bias, the transformer-based model represents inductive bias of global receptive field, and the CNN-like transformer-based model represents intermediate inductive bias between them. Inductive biases, broadly speaking, encourage the learning algorithm to prioritize solutions with certain properties. The comparison with these three models will be a clue to which inductive bias is more suitable for FME recognition. Furthermore, since the FME datasets consist of high-speed videos, 100 Hz or 200 Hz, it is different from regular videos with large changes in each frame. Therefore, we empirically investigate the proper way to use optical flow.

In ablation 5.1 and 5.2, we do not use the LF method because the LF requires too much computation. For the model detail, see Table 3. We represent the amount of computation for each model in multiply–accumulate operations (MACs), which is a common step that computes the product of two numbers and adds that product to an accumulator.

**Table 3.** Backbone information.

| Backbone | Num. of Parameters (million, M) | Num. of MACs [a] (giga, G) |
|---|---|---|
| 3DResNext [27] | 47.5 | 42.45 |
| Video Swin [20] | 86.7 | 0.573 |
| TimeSFormer [28] | 121.3 | 8.55 |

[a] Calculate MACs in the case of video length 16, frame size (224, 224).

### 5.1. Influence of the Optical Flow

We examine whether the motion feature yields significant improvements and analyze the effect of color information, which is considered useless in FME recognition because it is subject-dependent. To minimize the loss of color information, we use the video length of 32 close to the average number of frames in SMIC as a video length (33.7). It makes the effect of optical flow less pronounced.

Table 4 shows that incorporated optical flow had higher performance than those using only grayscale or color information. Interestingly, the 3D-ResNeXt-101, which consists only of CNN, performs best when using motion information, and video transformer models perform better when using image information together, contrary to what is generally known, that color information is meaningless. Therefore, it may be seen that it is effective to use motion information, and in the video transformer, it is appropriate to use image information and motion information together.

**Table 4.** Comparison between using optical flow and not.

| Backbone | Video Length | Preprocess | UF1 | UAR | Acc. (%) |
|---|---|---|---|---|---|
| 3DResNext [27] | 32 | RGB | 0.5945 | 0.6101 | 59.15 |
| | | RGB + OF | 0.6130 | 0.6107 | 60.98 |
| | | GRAY | 0.5583 | 0.5666 | 55.49 |
| | | GRAY + OF | 0.6487 | 0.6620 | 64.63 |
| | | OF | **0.6907** | **0.6983** | **69.51** |
| Video Swin [20] | | RGB | 0.5958 | 0.5858 | **62.20** |
| | | RGB + OF | **0.6203** | **0.6231** | **62.20** |
| | | GRAY | 0.5984 | 0.5928 | 60.37 |
| | | GRAY + OF | 0.6130 | 0.6177 | **62.20** |
| | | OF | 0.5762 | 0.5838 | 57.93 |
| TimeSFormer [28] | | RGB | 0.6046 | 0.6232 | 60.37 |
| | | RGB + OF | **0.6492** | **0.6572** | **64.63** |
| | | GRAY | 0.5061 | 0.5494 | 51.22 |
| | | GRAY + OF | 0.6392 | 0.6465 | 63.41 |
| | | OF | 0.6465 | 0.6476 | **64.63** |

Note: **Bold** shows the best performance of each model in each metric.

### *5.2. Investigation of the Proper Interpolated Length*

Since optical flow represents the motion of objects between two frames, meaningful features may not be extracted for images captured with a high-speed camera, so the proposed method interpolated each sample in half the average number of frames in the dataset to extract meaningful motion information. However, it is unknown whether half the average number is appropriate in most cases. Therefore, it is essential to interpolate with an appropriate number of frames. In this experiment, we compared the average, half of average, and a quarter of average. Based on the above results of comparison using optical flow, we do not consider using only color and grayscale.

In Table 5, the use of video length as 32 generally showed a similar or worse performance to using it as 8 and 16. We think these results are for two reasons. The first reason is that the number of data is too small to use long frames, so the model is overfitted. In general, as the dimension of the input vector used for training increases, the number of data should also increase in proportion to the dimension. However, in the case of FME datasets, the number of samples is small and thus the model is easily overfitted. The second reason is that motion information is more useful than color information, as confirmed in ablation 5.1. Increasing the interpolation length reduces the loss of color information and differences between two frames, which reduces the usefulness of the optical flow and lowers model performance. Thus, it is desirable to use a smaller video length.

**Table 5.** Comparison based on the number of frames.

| Backbone | Video Length | Preprocess | UF1 | UAR | Acc. (%) |
|---|---|---|---|---|---|
| 3DResNext [27] | 8 | RGB + OF | 0.5954 | 0.5938 | 62.80 |
| | | GRAY + OF | 0.6087 | 0.6082 | 64.63 |
| | | OF | 0.5955 | 0.5939 | 62.80 |
| | 16 | RGB + OF | 0.6476 | 0.6540 | 65.24 |
| | | GRAY + OF | 0.6888 | 0.6773 | 71.34 |
| | | OF | **0.6924** | 0.6790 | **71.95** |
| | 32 | RGB + OF | 0.6130 | 0.6107 | 60.98 |
| | | GRAY + OF | 0.6487 | 0.6620 | 64.63 |
| | | OF | 0.6907 | **0.6983** | 69.51 |
| Video Swin [20] | 8 | RGB + OF | 0.6481 | 0.6647 | 64.63 |
| | | GRAY + OF | **0.6977** | **0.6958** | **69.51** |
| | | OF | 0.6678 | 0.6947 | 66.46 |
| | 16 | RGB + OF | 0.6301 | 0.6416 | 62.80 |
| | | GRAY + OF | 0.6226 | 0.6168 | 62.80 |
| | | OF | 0.6528 | 0.6471 | 66.46 |
| | 32 | RGB + OF | 0.6203 | 0.6231 | 62.20 |
| | | GRAY + OF | 0.6130 | 0.6177 | 62.20 |
| | | OF | 0.5762 | 0.5838 | 57.93 |
| TimeSFormer [28] | 8 | RGB + OF | 0.6271 | 0.6440 | 62.20 |
| | | GRAY + OF | 0.6632 | 0.6834 | 65.85 |
| | | OF | 0.6723 | 0.6750 | 67.07 |
| | 16 | RGB + OF | **0.7038** | **0.6953** | **71.34** |
| | | GRAY + OF | 0.6471 | 0.6703 | 64.63 |
| | | OF | **0.7038** | **0.6953** | **71.34** |
| | 32 | RGB + OF | 0.6492 | 0.6572 | 64.63 |
| | | GRAY + OF | 0.6392 | 0.6465 | 63.41 |
| | | OF | 0.6465 | 0.6476 | 64.63 |

Note: **Bold** show the best performance of each model in each metric.

### *5.3. Effect of the Fusion Location*

The proposed model extracts two features individually using grayscale information and optical flow, but it is unknown whether this yields improvement because there is no research on the transformer. Therefore, we compare the performance of EF and LF.

From Table 6, it is difficult to determine which is better. However, when using LF, it needs one more backbone, which requires about twice as much additional operation as EF.

In addition, except for Swin, the highest performance of each model comes from EF, so EF can be considered better. However, since the highest performance among all models comes from LF, it is still difficult to determine superiority and inferiority. As a result, it is still an open problem.

**Table 6.** Comparison of early vs. late fusion.

| Backbone | Video Length | Preprocess | Fusion | UF1 | UAR | Acc (%) |
|---|---|---|---|---|---|---|
| 3DResNext [27] | 8 | RGB + OF | Early | 0.6291 | 0.6403 | 64.02 |
| | | | Late | 0.6759 | 0.6624 | 68.90 |
| | | GRAY + OF | Early | 0.6087 | 0.6082 | 64.63 |
| | | | Late | 0.6142 | 0.6096 | 62.20 |
| | 16 | RGB + OF | Early | 0.6476 | 0.6540 | 65.24 |
| | | | Late | 0.6598 | 0.6464 | 67.07 |
| | | GRAY + OF | Early | **0.6888** | **0.6773** | **71.34** |
| | | | Late | 0.6034 | 0.6151 | 61.59 |
| Video Swin [20] | 8 | RGB + OF | Early | 0.6481 | 0.6648 | 64.63 |
| | | | Late | 0.7027 | 0.7309 | 70.12 |
| | | GRAY + OF | Early | 0.6977 | 0.6958 | 69.51 |
| | | | Late | 0.6638 | 0.7005 | 66.46 |
| | 16 | RGB + OF | Early | 0.6301 | 0.6416 | 62.80 |
| | | | Late | 0.7013 | 0.7208 | 70.12 |
| | | GRAY + OF | Early | 0.6226 | 0.6168 | 62.80 |
| | | | Late | **0.7447** | **0.7377** | **73.17** |
| TimeSFormer [28] | 8 | RGB + OF | Early | 0.6632 | 0.6834 | 65.85 |
| | | | Late | 0.6914 | **0.6983** | 68.90 |
| | | GRAY + OF | Early | 0.6632 | 0.6834 | 65.85 |
| | | | Late | 0.6536 | 0.6499 | 65.24 |
| | 16 | RGB + OF | Early | **0.7038** | 0.6953 | **71.34** |
| | | | Late | 0.6474 | 0.6452 | 64.02 |
| | | GRAY + OF | Early | 0.6471 | 0.6703 | 64.63 |
| | | | Late | 0.6615 | 0.6594 | 66.46 |

Note: **Bold** show the best performance of each model in each metric.

## 6. Conclusions

Recently, various studies have been proposed to solve FME recognition, but no studies have used transformers. In this research, we focused on whether the transformer model can be suitable for FME recognition. Since transformers generally require a large amount of data but there is no sufficient dataset for FME recognition, our main purpose is to train the transformer model successfully. We achieve the purpose using the optical flow, which was mainly used by video processing models, and LF with the transformer. As a result, our model becomes the SOTA in SMIC and achieves comparable performance in CASME II, although we do not use methods specialized for FME recognition as in other studies.

**Author Contributions:** The contribution of the authors for this publication article are as follows: J.H.: methodology, software, conceptualization, writing—original draft, writing—review and editing. C.L.: investigation, writing—review and editing. H.J.: conceptualization, project administration, supervision, writing—review and editing. All authors have read and agreed to the published version of the manuscript.

**Funding:** This work was supported by Institute of Information & Communications Technology Planning & Evaluation (IITP) grant funded by the Korea government(MSIT) (2019-0-00330, Development

of AI Technology for Early Screening of Infant/Child Autism Spectrum Disorders based on Cognition of the Psychological Behavior and Response). Furthermore, this research was also supported in part by a National Research Foundation of Korea (NRF) grant funded by the Korean government (MSIT) (No. 2020R1C1C1007423). In addition, this work was partly supported by the National Research Foundation (NRF), Korea, under project BK21 FOUR.

**Institutional Review Board Statement:** Not applicable.

**Informed Consent Statement:** Not applicable.

**Data Availability Statement:** Data sharing not applicable.

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

## References

1. Li, X.; Pfister, T.; Huang, X.; Zhao, G.; Pietikäinen, M. A spontaneous micro-expression database: Inducement, collection and baseline. In Proceedings of the 2013 10th IEEE International Conference and Workshops on Automatic Face and Gesture Recognition (FG), Shanghai, China, 22–26 April 2013; pp. 1–6.
2. 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.
3. Wang, S.J.; Yan, W.J.; Zhao, G.; Fu, X.; Zhou, C.G. Micro-expression recognition using robust principal component analysis and local spatiotemporal directional features. In *European Conference on Computer Vision*; Springer: Berlin/Heidelberg, Germany, 2014; pp. 325–338.
4. Liong, S.T.; See, J.; Wong, K.; Phan, R.C.W. Less is more: Micro-expression recognition from video using apex frame. *Signal Process. Image Commun.* **2018**, *62*, 82–92.
5. Li, X.; Hong, X.; Moilanen, A.; Huang, X.; Pfister, T.; Zhao, G.; Pietikäinen, M. Towards reading hidden emotions: A comparative study of spontaneous micro-expression spotting and recognition methods. *IEEE Trans. Affect. Comput.* **2017**, *9*, 563–577.
6. Zong, Y.; Huang, X.; Zheng, W.; Cui, Z.; Zhao, G. Learning from hierarchical spatiotemporal descriptors for micro-expression recognition. *IEEE Trans. Multimed.* **2018**, *20*, 3160–3172.
7. Horn, B.K.; Schunck, B.G. Determining optical flow. *Artif. Intell.* **1981**, *17*, 185–203.
8. Li, J.; Wang, Y.; See, J.; Liu, W. Micro-expression recognition based on 3D flow convolutional neural network. *Pattern Anal. Appl.* **2019**, *22*, 1331–1339.
9. Xia, Z.; Hong, X.; Gao, X.; Feng, X.; Zhao, G. Spatiotemporal recurrent convolutional networks for recognizing spontaneous micro-expressions. *IEEE Trans. Multimed.* **2019**, *22*, 626–640.
10. Choi, D.Y.; Song, B.C. Facial Micro-Expression Recognition Using Two-Dimensional Landmark Feature Maps. *IEEE Access* **2020**, *8*, 121549–121563. https://doi.org/10.1109/ACCESS.2020.3006958.
11. Nie, X.; Takalkar, M.A.; Duan, M.; Zhang, H.; Xu, M. GEME: Dual-stream multi-task GEnder-based micro-expression recognition. *Neurocomputing* **2021**, *427*, 13–28.
12. Patel, D.; Hong, X.; Zhao, G. Selective deep features for micro-expression recognition. In Proceedings of the 2016 23rd International Conference on Pattern Recognition (ICPR), Cancun, Mexico, 4–8 December 2016; pp. 2258–2263. https://doi.org/10.1109/ICPR.2016.7899972.
13. Liu, Z.; Lin, Y.; Cao, Y.; Hu, H.; Wei, Y.; Zhang, Z.; Lin, S.; Guo, B. Swin transformer: Hierarchical vision transformer using shifted windows. *arXiv* **2021**, arXiv:2103.14030.
14. Goyal, A.; Bengio, Y. Inductive biases for deep learning of higher-level cognition. *arXiv* **2020**, arXiv:2011.15091.
15. Jung, H.; Lee, S.; Yim, J.; Park, S.; Kim, J. Joint fine-tuning in deep neural networks for facial expression recognition. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 7–13 December 2015; pp. 2983–2991.
16. Zhou, Z.; Zhao, G.; Pietikäinen, M. Towards a practical lipreading system. In Proceedings of the CVPR 2011, Colorado Springs, CO, USA, 20–25 June 2011; pp. 137–144.
17. 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**, arXiv:2010.11929.
18. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, Ł.; Polosukhin, I. Attention is all you need. In *Advances in Neural Information Processing Systems*; The MIT Press: Cambridge, MA, USA, 2017; pp. 5998–6008.
19. Farnebäck, G. Two-frame motion estimation based on polynomial expansion. In *Scandinavian Conference on Image Analysis*; Springer: Berlin/Heidelberg, Germany, 2003; pp. 363–370.
20. Liu, Z.; Ning, J.; Cao, Y.; Wei, Y.; Zhang, Z.; Lin, S.; Hu, H. Video swin transformer. *arXiv* **2021**, arXiv:2106.13230.
21. Peng, M.; Wang, C.; Chen, T.; Liu, G.; Fu, X. Dual temporal scale convolutional neural network for micro-expression recognition. *Front. Psychol.* **2017**, *8*, 1745.
22. Schuller, B.; Vlasenko, B.; Eyben, F.; Wöllmer, M.; Stuhlsatz, A.; Wendemuth, A.; Rigoll, G. Cross-Corpus Acoustic Emotion Recognition: Variances and Strategies. *IEEE Trans. Affect. Comput.* **2010**, *1*, 119–131. https://doi.org/10.1109/T-AFFC.2010.8.
23. Ngo, A.C.L.; Phan, R.C.W.; See, J. Spontaneous Subtle Expression Recognition: Imbalanced Databases and Solutions. In Proceedings of the ACCV, Singapore, 1–5 November 2014.

24. Loshchilov, I.; Frank, H. Decoupled weight decay regularization. In Proceedings of the ICLR 2019, New Orleans, LA, USA, 6–9 May 2019.
25. Chen, C. PyTorch Face Landmark: A Fast and Accurate Facial Landmark Detector. 2021. Available online: https://github.com/cunjian/pytorch_face_landmark (accessed on 20 December 2021).
26. Zhou, L.; Shao, X.; Mao, Q. A survey of micro-expression recognition. *Image Vis. Comput.* **2021**, *105*, 104043.
27. Xie, S.; Girshick, R.; Dollár, P.; Tu, Z.; He, K. Aggregated residual transformations for deep neural networks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 1492–1500.
28. Bertasius, G.; Wang, H.; Torresani, L. Is Space-Time Attention All You Need for Video Understanding? *arXiv* **2021**, arXiv:2102.05095.