Swin-FER: Swin Transformer for Facial Expression Recognition

Mei Bie 1,2, Huan Xu 2, Yan Gao 2, Kai Song 1* and Xiangjiu Che 2,*

1 Institute of Education, Changchun Normal University, Changchun 130032, China; biemei@ccsfu.edu.cn (M.B.); songkai@ccsfu.edu.cn (K.S.)
2 College of Computer Science and Technology, Jilin University, Changchun 130012, China; xuhuan20@mails.jlu.edu.cn (H.X.); yan19860628@163.com (Y.G.)
* Correspondence: chexj@jlu.edu.cn

Abstract: The ability of transformers to capture global context information is highly beneficial for recognizing subtle differences in facial expressions. However, compared to convolutional neural networks, transformers require the computation of dependencies between each element and all other elements, leading to high computational complexity. Additionally, the large number of model parameters need extensive data for training so as to avoid overfitting. In this paper, according to the characteristics of facial expression recognition tasks, we made targeted improvements to the Swin transformer network. The proposed Swin-Fer network adopts the fusion strategy from the middle layer to deeper layers and employs a method of data dimension conversion to make the network perceive more spatial dimension information. Furthermore, we also integrated a mean module, a split module, and a group convolution strategy to effectively control the number of parameters. On the Fer2013 dataset, an in-the-wild dataset, Swin-Fer achieved an accuracy of 71.11%. On the CK+ dataset, an in-the-lab dataset, the accuracy reached 100%.

Keywords: Swin transformer; facial expression recognition; feature fusion; classification

1. Introduction

The transformer is a kind of neural network model which utilizes self-attention mechanisms to establish dependency relationships between sequences, and it has attracted wide attention for its excellent performance in natural-language processing tasks [1]. Many researchers have tried to apply the transformer model to facial expression recognition tasks. This mechanism enables the model to consider all other elements in a sequence while processing each sequence element, which is especially useful for capturing subtle variations in expressions [2]. Compared with convolutional networks, the transformer is not constrained by local receptive fields, allowing it to flexibly focus on any part of an image. Moreover, its self-attention structure supports better parallel processing [3,4]. However, this self-attention mechanism involves computing dependencies between each element and all other elements, resulting in high data dimensions and computational overheads [5]. Additionally, the model includes numerous parameters and usually needs a lot of data for effective training to mitigate overfitting. Consequently, the model is larger, requiring more computational resources for both training and inference [6].

The Swin transformer, a variant of the transformer, has been specifically optimized for visual tasks. Introducing window partitioning mechanisms and hierarchical structures enables the model to learn features of different scales [7]. In this paper, based on the characteristics of facial expression recognition tasks, we made targeted modifications to the original Swin transformer network structure, proposing the Swin-Fer network (Swin Transformer for Facial Expression Recognition), which achieved promising experimental results. The main contributions are as follows:

1. Swin-Fer adopts a fusion strategy from middle to deep layers to capture facial expression features more accurately. This guides the network to learn the relation-
2. Using the data dimension transformation strategy, the whole network model can perceive more spatial dimension information. In addition, in order to improve the generalization ability of the model, the mean module and the split module, as well as group convolution, are introduced. While achieving satisfactory experimental results, the parameter count is kept largely unchanged.

3. The proposed method achieves an accuracy of 71.11% on the Fer2013 dataset under natural conditions and 100% on the CK+ dataset in laboratory environments. The sensitivity and specificity of the model, as indicated by the area under the curve, are also good.

2. Related Works

2.1. Transformer for Facial Expression Recognition

As a model of a self-attention mechanism, the transformer model demonstrates excellent performance and adaptability to various facial expression recognition tasks [8]. Xue et al. [9] applied the transformer method to facial expression recognition tasks and developed the TransFER model by integrating strategies such as Multi-Attention Dropping (MAD) and Multi-Head Self-Attention Dropping. This model highlights important local blocks (patches) while suppressing irrelevant ones, thereby exploring rich relationships between local blocks and addressing to some extent the issue of “large inter-class similarity, small intra-class similarity” in facial expression recognition tasks. Ma et al. [2] proposed the VTFF (Visual Transformer with Feature Fusion) network, which introduced the Attentional Selective Fusion (ASF) method to leverage two feature maps generated by a dual-branch CNN. Through global–local attention mechanisms, multiple features are fused to capture discriminative visual words, and the model achieved excellent performance. In facial expression classification tasks, many researchers input image samples containing various emotional states into transformer models and use a softmax classifier to predict the emotional labels represented by each image. Kim et al. [10] observed that the Vision Transformer (ViT) may have limited ability to capture subtle changes in facial expressions and may potentially lose local features of images. They proposed Squeeze-ViT, which can reduce feature dimensions to lower computational complexity and integrate global and local features to enhance network performance. Zhao et al. [11] proposed the Former-DFER network structure for natural environmental scenes, composed of a CS transformer (a convolutional spatial transformer) and a T transformer (a temporal transformer). This architecture guides the network to learn spatial and contextual facial features, thus improving the accuracy of facial expression classification.

2.2. Overview of the Swin Transformer

The Swin transformer, as described by Liu et al. [7], is a hierarchical vision transformer designed to handle various computer vision tasks efficiently. It uses a shifted-windows approach for self-attention, which allows for better scalability and performance compared to existing transformer-based models. It builds hierarchical feature maps, similar to convolutional neural networks (CNNs), enabling it to capture multi-scale representations effectively.

A key innovation of the Swin transformer is its use of shifted windows for self-attention computation. Self-attention is first computed within local, non-overlapping windows, and then these window positions are shifted between successive layers. This approach facilitates cross-window connections, enhancing the model’s ability to capture long-range dependencies while maintaining computational efficiency. The combination of non-overlapping and shifted windows allows the Swin transformer to balance local context extraction within windows with global context integration across windows, leading to high performance in vision tasks. By focusing on local windows, the computational complexity of the self-attention mechanism is significantly reduced, making the Swin transformer more efficient and scalable for high-resolution images. Liang et al. [12] addresses the challenges...
of facial expression recognition (FER) posed by occlusions and head-pose variations using a convolution-transformer dual branch network (CT-DBN). The CT-DBN leverages the strengths of both convolutional neural networks (CNNs) and the Swin transformer to capture local and global facial information, respectively. Qin et al. [13] integrated a Multi-Level Channel Attention (MLCA) module into each task-specific subnet, enabling adaptive feature selection from optimal levels and channels. This design allows the Swin transformer to efficiently and accurately perform facial expression recognition, achieving strong experimental results and demonstrating superior understanding of facial features.

3. Proposed Method

This paper designed a facial expression recognition method based on the Swin transformer, named Swin-Fer, whose network structure is depicted in Figure 1. Considering the characteristics of facial expression recognition tasks, deeper-level features contain richer semantic information. Swin-Fer employs a fusion strategy from middle to deep layers [14]. After the image is input into the network, the image window segmentation and patch embedding operations are carried out. The input image is divided into blocks of fixed sizes, and each block is embedded to obtain a fixed-size vector representation. The channel number is converted to 96, equivalent to performing a $4 \times 4$ convolution (kernel size = 4) without overlapping regions, with each movement having a stride of 4 (stride = 4). Hence, an input image with a size of $B \times 3 \times 224 \times 224$ is $56 \times 56$ after flattening. The data are then flattened from $B \times C \times H \times W$ (batch size $\times$ channel number $\times$ height $\times$ width) along the height direction, which effectively converts the spatial dimensions into one. Subsequently, a transpose operation is performed to exchange the spatial and channel dimensions.

![Figure 1](image-url)

**Figure 1.** The network architecture of Swin-Fer.

In Figure 1, STB represents the Swin transformer block, where STB1, STB2, STB3, and STB4 include one pair, one pair, three pairs, and one pair of W-MSA and SW-MSA combinations, respectively, and iterates on this basis. The extraction and fusion of the information occurs after the patch merging layer between STB2, STB3, and STB4, resulting in STB2P and STB3P. The letter ‘P’ in STB2P and STB3P denotes the output after the patch merging operation. Specifically, STB2P refers to the output after the patch merging that occurs between STB2 and STB3, while STB3P represents the output after the patch merging that occurs between STB3 and STB4. Layer normalization is a regularization technique in deep learning, which is similar to batch normalization (BN), but it calculates the mean and
variance of each layer rather than each batch of each neuron, making it more suitable for sequence models. After layer normalization of STB4, STB4L is obtained. Along with STB4’s output (STB4O) and the earlier STB2P and STB3P, the information output results from these four levels undergo data dimension transformation, adaptive average pooling, and mean operations before being merged. After passing through the split module, a fusion output is obtained. This result is added to STB4L, and adaptive average pooling is performed again to further reduce the spatial features to obtain the final output.

3.1. Patch Merging

Patch merging is a technique employed in transformer models to enhance the efficiency of image processing. It involves dividing the input image into several sub-images which are then concatenated and fed into the transformer model for processing, as illustrated in Figure 2. This approach reduces the computational load of the model by avoiding processing the entire large-scale image directly, thus improving efficiency when dealing with large images.

![Figure 2. Schematic representation of patch merging.](image)

The concatenated image is fed into the transformer model for processing. During this process, the patch merging compresses the detailed information of the input high-resolution feature map into a low-resolution feature map. While retaining the main information from the original feature map, this operation reduces computational complexity, thereby enhancing the calculation efficiency and generalization capability of the model [7,15].

The patch merging process between two STBs is equivalent to a downsampling operation without convolution. Previous experiments tried to extract features before patch merging, resulting in two issues: excessive parameters and irregular data, leading to sub-optimal feature extraction. Therefore, all experiments in this study executed feature data extraction after patch merging.

3.2. Dimensional Transformation of Data

Within each STB, all odd blocks have a shift size of 0, while all even blocks have a shift size of 3. A shift size of 0 implies no operation, and the spatial features are directly normalized before being unfolded. When the shift size is 3, the dimensions (1 and 2) are shifted, and the shifted content is filled circularly, thus realizing the torch rolling operation. The purpose of torch rolling is to disrupt the internal structure of the data, maintaining the same data in the feature map but altering the position, thus disrupting the dependencies and linear relationships between pixels, facilitating more complex interaction between grids. Regardless of the subsequent operation on odd or even blocks, the output feature dimensions remain the same. The output window features (X_window) need to be merged again, equivalent to recombining spatial features, and with the application of window-based multi-head attention mechanism and shifted-windows multi-head self-attention.
(SW-MSA), interactions within and between windows are achieved, further improving model performance.

In addition, the multi-head self-attention mechanism divides the input data into several heads, and each head generates different query, key, and value vectors and computes the corresponding attention-weighted results. These results are then concatenated to form the final output of the multi-head self-attention mechanism. This output is a three-dimensional tensor that needs to be reshaped to a four-dimensional tensor for better matching with subsequent feature fusion calculations. For example, position encoding needs to add position information to input features, and decoder calculation needs to concatenate the encoder results with target sequences. Therefore, converting the three-dimensional tensor to a four-dimensional tensor facilitates these calculations, improving model efficiency and accuracy.

Assume that the input datum is \( X \in \mathbb{R}^{B \times L \times D} \), where \( B \) represents batch size, \( L \) denotes sequence length, and \( D \) represents input dimensionality. In the multi-head self-attention mechanism, the input datum is divided into \( H \) parts through linear mapping, and after multi-head attention calculation, the output with shape \( Y \in \mathbb{R}^{B \times L \times H} \) is obtained.

Initially, \( H \) is divided into \( \text{num\_heads} \) parts, and the input dimension, \( D \), is also divided into the same number of parts, that is, \( D = \text{num\_heads} \times \text{head\_dim} \), where \( \text{head\_dim} \) represents the dimension size of each head. The output tensor, \( Y \), of the multi-head self-attention mechanism is transformed into a four-dimensional tensor, \( Z \in \mathbb{R}^{B \times L \times \text{num\_heads} \times \text{head\_dim}} \), as follows:

\[
Z = \text{reshape} (Y, (B, L, \text{num\_heads}, \text{head\_dim}))
\] (1)

This operation segments the last dimension of the output tensor, creating a new dimension for each head in the new shape. The transformation from three-dimensional to four-dimensional data effectively opens up spatial features before feature fusion, so that the whole network model can perceive more spatial dimension information, thus enhancing classification performance.

### 3.3. Mean Module

In transformer networks, the mean operation is usually used to reduce the dimensionality of feature tensors extracted through self-attention mechanisms, enabling them to be input into fully connected layers for classification. For instance, in the vision transformer proposed by Dosovitskiy et al. [16], a feature tensor is divided into multiple blocks, followed by self-attention mechanisms, resulting in feature vectors for each block. Subsequently, each feature vector undergoes a mean operation to obtain a fixed-length feature vector for final classifier input.

In the Swin-Fer network, images are segmented into blocks, and features are extracted by self-attention mechanisms; thus, a sequence of features is generated. The mean module allows the obtaining of the average information of the data, achieving dimensionality reduction and reducing computational complexity while retaining the main information.

Assume that the size of the obtained feature tensor is \( H \times W \times n \), where \( H \) represents height, \( W \) represents width, and \( n \) represents the number of channels [17]. The calculation formula for the mean module is as follows:

\[
\text{mean}(X)_{ij} = \frac{1}{n} \sum_{k} X_{i,j,k}
\] (2)

where \( \text{mean}(X)_{ij} \) represents the pixel value of the pooled output feature map at position \((i,j)\), while \( X_{i,j,k} \) represents the pixel value of the input feature map \( X \) at position \((i,j,k)\). The above formula indicates averaging pooling along the third dimension (channel dimension) of the feature tensor to obtain a new feature tensor. Since this operation aggregates the original \( n \) channels into one channel, the number of channels changes from \( n \) to 1. This averaging pooling method concatenates vectors without increasing network parameters.
3.4. Split Module

Swin-Fer focuses on controlling the number of parameters while performing facial expression recognition tasks. With data dimensions already high, it is essential to minimize weight or bias operations as much as possible. As shown in Figure 1, in Branch1 and Branch2, each branch contains a group convolution. Because the data processing at this stage is not very extensive, group convolution can be considered instead of regular convolution to reduce parameters, especially for small datasets. As shown in Figure 3, the group convolution strategy first divides the input feature map into several groups, with an equal number of channels in each group. Then, convolution operations are performed separately on each group, and, finally, the convolution results of each group are concatenated as the final output. Through this channel grouping strategy, different groups can learn different features, enhancing feature diversity. Additionally, group convolution enables better learning of inter-channel relationships, strengthening model representation capability. Specifically:

$$X' = \bigoplus_{g=1}^{G} X^{(g)} \ast K^{(g)} + b$$

(3)

where $X^{(g)}$ represents the g-th group of the input feature map, with a total of G groups. After group convolution, the output feature map is $X'$, $\bigoplus$ denotes the tensor concatenation operation, $K^{(g)}$ represents the convolution kernel of the g-th group, and b is the bias term.

The results of these two branches are concatenated. Subsequently, the tensor dimension transposition operation is performed, where the input is [B, 2, 2, 4, 4] and the output is [B, 2, 2, 4, 4]. Although the second and third dimensions are both 2, they actually switch positions (as shown in Figure 4). The transposed tensor allows concatenation along the batch processing dimension and better utilization of hardware parallelism to speed up calculation. Generally, after tensor dimension transposition, position information is restored. However, since the data volume at this stage of the experiment is already small, the impact on experimental results is negligible, and the restoration operation is not executed.

![Figure 3. Group convolution.](image)

![Figure 4. Transpose operation.](image)
4. Experiment Results and Analysis

4.1. Experimental Datasets

The Cohn–Kanade dataset (CK+) is a laboratory-environment dataset, as depicted in Figure 5, showcasing sample expressions for each category. For the task of facial expression recognition in static images, the last three frames of the expression sequence [18], known as the peak expression states, are usually selected for training and testing. In this paper, 981 images are selected as the experimental dataset, comprising 882 images for training and 99 images for testing. The sample distribution of each category is detailed in Table 1.

![Sample examples from the CK+ dataset.](image)

**Figure 5.** Sample examples from the CK+ dataset.

<table>
<thead>
<tr>
<th>Expression</th>
<th>Surprise</th>
<th>Happy</th>
<th>Disgust</th>
<th>Anger</th>
<th>Fear</th>
<th>Sad</th>
<th>Contempt</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Train</td>
<td>225</td>
<td>186</td>
<td>159</td>
<td>123</td>
<td>66</td>
<td>75</td>
<td>48</td>
<td>882</td>
</tr>
<tr>
<td>Test</td>
<td>24</td>
<td>21</td>
<td>18</td>
<td>12</td>
<td>9</td>
<td>9</td>
<td>6</td>
<td>99</td>
</tr>
<tr>
<td>Total</td>
<td>249</td>
<td>207</td>
<td>177</td>
<td>135</td>
<td>75</td>
<td>84</td>
<td>54</td>
<td>981</td>
</tr>
</tbody>
</table>

Fer2013, a representative dataset for facial expressions in natural environments, included facial occlusions (e.g., hands, hats, and glasses), low pixel resolutions, and facial images captured with arbitrary poses and angles. After preprocessing, the samples in the FER-2013 dataset were scaled to $48 \times 48$ pixels. As a challenging dataset, Figure 6 illustrates sample images for each category, and Table 2 outlines the sample distribution for each classification.

![Sample examples from the Fer2013 dataset.](image)

**Figure 6.** Sample examples from the Fer2013 dataset.

<table>
<thead>
<tr>
<th>Expression</th>
<th>Happy</th>
<th>Neutral</th>
<th>Sad</th>
<th>Fear</th>
<th>Angry</th>
<th>Surprise</th>
<th>Disgust</th>
<th>Total</th>
</tr>
</thead>
<tbody>
<tr>
<td>Train</td>
<td>7215</td>
<td>4965</td>
<td>4830</td>
<td>4097</td>
<td>3995</td>
<td>3171</td>
<td>436</td>
<td>28,709</td>
</tr>
<tr>
<td>Test</td>
<td>1774</td>
<td>1233</td>
<td>1247</td>
<td>1024</td>
<td>958</td>
<td>831</td>
<td>111</td>
<td>7178</td>
</tr>
<tr>
<td>Total</td>
<td>8989</td>
<td>6198</td>
<td>6077</td>
<td>5121</td>
<td>4953</td>
<td>4002</td>
<td>547</td>
<td>35,877</td>
</tr>
</tbody>
</table>

4.2. Experimental Environment

The experimental environment for Swin-Fer is based on Win10, with a 12th Gen Intel(R) Core(TM) i7 3.6GHz CPU and an NVIDIA Geforce RTX 4090 GPU, within a python
development environment. During model training, the adaptive learning rate optimization algorithm, the Adam optimizer, is adopted, dynamically adjusting the learning rate based on gradient information in each iteration, thus achieving faster and more stable convergence. All experiments in the manuscript were conducted using the same environmental settings and hyperparameter configurations. The specific experimental parameters are presented in Table 3.

Table 3. Experimental parameters for Swin-Fer.

<table>
<thead>
<tr>
<th>Parameter Name</th>
<th>Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>Batch size</td>
<td>80</td>
</tr>
<tr>
<td>Learning rate</td>
<td>0.0001</td>
</tr>
<tr>
<td>Optimizer</td>
<td>Adam</td>
</tr>
<tr>
<td>Seed</td>
<td>1024</td>
</tr>
<tr>
<td>Image size</td>
<td>224</td>
</tr>
</tbody>
</table>

4.3. Experimental Results

Applying a transformer to the FER2013 dataset for facial expression recognition is a challenging task due to the small original image size of $48 \times 48$, while most transformer models are suitable for sizes of $224 \times 356$. The Swin transformer requires a large image size for operations such as patch embedding, with a step size of 4. Therefore, to obtain the transformer’s benchmark, the image size of the facial expression pictures had to be enlarged to $224 \times 224$. We tested four lightweight transformer structures, with specific experimental results presented in Table 4. With almost no change in the parameter count, the recognition accuracy of our method reached 71.11% on FER2013—an increase of 0.41% compared to the original Swin transformer—and 100% on CK+—a 3.75% improvement with respect to the original network.

Table 4. Accuracy of transformer methods on the FER2013 and CK+ datasets.

<table>
<thead>
<tr>
<th>Method</th>
<th>Fer2013</th>
<th>CK+</th>
</tr>
</thead>
<tbody>
<tr>
<td>EdgeViT</td>
<td>0.5935</td>
<td>0.725</td>
</tr>
<tr>
<td>EfficientFormer</td>
<td>0.6053</td>
<td>0.7125</td>
</tr>
<tr>
<td>Vision transformer</td>
<td>0.6848</td>
<td>0.85</td>
</tr>
<tr>
<td>Swin transformer</td>
<td>0.7070</td>
<td>0.9625</td>
</tr>
<tr>
<td>Swin-Fer</td>
<td>0.7111</td>
<td>1</td>
</tr>
</tbody>
</table>

To expand the practicality of transformers, there are usually different types of the same transformer architecture, such as tiny, small, base, and large. Increasing the model in transformer methods typically improves the accuracy, but there is also a corresponding exponential increase in parameters between different models. The transformer models selected in Table 4 are all lightweight structures, such as small and tiny-level structures, emphasizing the use of methods like mean, adaptive pooling, and group convolution to control the number of parameters [19]. As shown in Table 5, while applying a fusion strategy to better extract features, the total parameters of the proposed Swin-Fer network increased by only 70 B compared to the original Swin transformer.

Table 5. Comparison of network parameter counts for transformer methods.

<table>
<thead>
<tr>
<th>Network</th>
<th>Number of Parameters</th>
</tr>
</thead>
<tbody>
<tr>
<td>EdgeViT</td>
<td>11.30 M</td>
</tr>
<tr>
<td>EfficientFormer</td>
<td>19.8 M</td>
</tr>
<tr>
<td>Vision transformer</td>
<td>21.42 M</td>
</tr>
<tr>
<td>Swin transformer</td>
<td>27.53 M (27530120 B)</td>
</tr>
<tr>
<td>Swin-Fer</td>
<td>27.53 M (27530190 B)</td>
</tr>
</tbody>
</table>
The experimental results for Swin-Fer and the original Swin transformer on the FER2013 dataset are shown in Figure 7. From Figure 7a, it is evident that due to the use of pre-training methods, both models quickly showed a trend of convergence, stabilizing around 40 iterations. For the facial expression recognition problem, which is indeed a multi-class classification problem, we adopted the One-vs.-Rest strategy to generate the ROC curves. As shown in Figure 7b, the area under the ROC curve for both models is 0.84, with the curves deviating obviously from the 45-degree diagonal line, indicating good sensitivity and specificity. However, as can be seen in Figure 7c, it is apparent that the Swin-Fer method has relatively smaller fluctuations in accuracy, exhibiting more stability. Finally, from the confusion matrices presented in Figure 7d,e, the accuracy changes in various classifications between Swin-Fer and the original Swin transformer are as follows: anger (+0.2%), disgust (+3.6%), fear (+0.7%), happiness (−1.7%), neutral (+1.4%), sadness (+1%), and surprise (+1.9%), with an increase in accuracy for six out of seven classifications, demonstrating the effectiveness of the proposed improvement algorithm.

Figure 7. Experimental results on the FER2013 dataset.
Figure 8 presents the experimental results for Swin-Fer and the original Swin transformer network on the CK+ dataset. Due to the limited sample size of the CK+ dataset, the model may be constrained by insufficient data during training and validation, resulting in fluctuations in accuracy. The proposed Swin-Fer method, based on fusion methods and introducing split modules and data dimension transformation strategies, enhances the model’s ability to capture facial expression details. The experimental results show that compared to the original Swin transformer model, the Swin-Fer method has a relatively smoother loss curve on the CK+ dataset, indicating more stable performance during training. Additionally, the accuracy increased by 3.75%, which shows that the Swin-Fer method also exhibits certain potential and advantages in facial expression recognition tasks in laboratory environments.
4.4. Comparison of Experimental Accuracy

The accuracy of the Swin-Fer network structure proposed in this paper along with the accuracies of other methods introduced in the past five years on the FER2013 and CK+ datasets are compared in Table 6. Some representative research on FER2013 achieved more than 70% accuracy, which shows that facial expression recognition tasks in natural environments are challenging. Adjusting network structures and optimizing hyperparameters can help capture complex features in images, thus further improving model performance. This paper has explored using transformer methods to handle facial expression recognition tasks, focusing on designing the Swin-Fer network model. With the network parameters mostly unchanged, competitive experimental results were achieved, with an accuracy of 71.11%. In addition, the third column of Table 6 compares the accuracies of different network structures on the CK+ dataset, and these accuracies are for the most part very high, with the lowest accuracy reaching 96.25%. The method proposed in this paper, Swin-Fer, achieved 100%. These experimental results enhance confidence in the generalization ability of the model.

Table 6. Comparison of experimental accuracy of different methods on FER2013 and CK+ datasets.

<table>
<thead>
<tr>
<th>Network</th>
<th>Fer2013</th>
<th>CK+</th>
</tr>
</thead>
<tbody>
<tr>
<td>CNN using the Adamax optimizer [20]</td>
<td>66%</td>
<td>-</td>
</tr>
<tr>
<td>VGG16+SE Block [21]</td>
<td>66.8%</td>
<td>99.18%</td>
</tr>
<tr>
<td>DeepEmotion [22]</td>
<td>70.02%</td>
<td>98%</td>
</tr>
<tr>
<td>Swin transformer</td>
<td>70.70%</td>
<td>96.25%</td>
</tr>
<tr>
<td>Improved CNN based on center loss [23]</td>
<td>71.39%</td>
<td>96.64%</td>
</tr>
<tr>
<td>SSER [24]</td>
<td>71.62%</td>
<td>97.59%</td>
</tr>
<tr>
<td>Multi-Channel Attention Residual Network [25]</td>
<td>72.7%</td>
<td>98.8%</td>
</tr>
<tr>
<td>ResNet-50 + pyramid + cascaded attention block + GRU [26]</td>
<td>-</td>
<td>99.23%</td>
</tr>
<tr>
<td>ViT+SE [27]</td>
<td>-</td>
<td>99.8%</td>
</tr>
<tr>
<td>Swin-Fer</td>
<td>71.11%</td>
<td>100%</td>
</tr>
</tbody>
</table>

To further compare the effectiveness of the proposed model, we conducted training on a larger dataset, AffectNet. AffectNet is a large-scale facial expression dataset designed for training and evaluating facial expression recognition models. It contains over one million facial images collected from the Internet, with approximately 45,000 manually annotated images for eight emotion categories. The images in the AffectNet dataset typically have a resolution of \(256 \times 256\) pixels. These images cover a wide range of facial expressions and poses, making the dataset suitable for research on emotion analysis and facial expression recognition. This resolution is appropriate for training and testing deep learning models, particularly those that require high-resolution inputs, such as transformer models. Table 7 shows the comparison of experimental accuracy across different methods on the AffectNet (eight emotions) dataset.

Table 7. Comparison of experimental accuracy of different methods on AffectNet datasets.

<table>
<thead>
<tr>
<th>Network</th>
<th>AffectNet ( Eight Emotions)</th>
</tr>
</thead>
<tbody>
<tr>
<td>EfficientFace [28]</td>
<td>59.89%</td>
</tr>
<tr>
<td>SL + SSL puzzling (B2) [29]</td>
<td>61.32%</td>
</tr>
<tr>
<td>Multi-Task EfficientNet-B0 [30]</td>
<td>61.32%</td>
</tr>
<tr>
<td>DAN [31]</td>
<td>62.09%</td>
</tr>
<tr>
<td>CAGE [32]</td>
<td>62.30%</td>
</tr>
<tr>
<td>Vit-base + MAE [33]</td>
<td>62.42%</td>
</tr>
<tr>
<td>Swin-Fer</td>
<td>63.29%</td>
</tr>
</tbody>
</table>

As indicated in Tables 6 and 7, although the proposed method did not achieve the highest accuracy on the FER2013 dataset, it demonstrated superior performance on the AffectNet (eight emotions) dataset compared to other advanced models. The larger resolution
and color images of AffectNet likely better suit the capabilities of Swin-Fer, highlighting its strength in extracting feature information from high-resolution color images. This suggests that Swin-Fer, based on the Swin transformer for feature extraction, performs well in settings where input images are of larger spatial dimensions, allowing the model to extract more effective feature information.

5. Conclusions

Facial expression recognition is a challenging computer vision task. The key regions of different expressions exhibit diverse distributions, especially in natural conditions, such as head posture, lighting changes, and occlusion, which make feature extraction particularly difficult [34,35]. In order to enhance the model’s generalization, Swin-Fer contains a split module, replaces ordinary convolutions with group convolutions in two branches, and effectively controls the number of parameters through strategies such as averaging and adaptive pooling.

According to the experimental results, the proposed method effectively focuses on the most critical regions of facial expressions and achieves competitive results on datasets from both natural conditions (FER2013) and laboratory environments (CK+). Due to the low sample pixel quality of the Fer2013 dataset, the performance of transformer models is limited, and accuracy needs to be further improved. However, on the AffectNet dataset, which consists of higher pixel quality and colored images, performance was notably better, demonstrating the advantages of Swin-Fer. Enhancing the performance of transformer models on low-resolution facial expression images remains a key focus for future research. Additionally, current research is limited to static-image single-person facial expression recognition. Future research should include a broader range of practical scenarios, such as real-time, multi-person, and dynamic video facial expression recognition, and provide powerful support tools for practical applications.

Author Contributions: Conceptualization, M.B. and X.C.; methodology, H.X. and Y.G.; investigation, H.X. and Y.G.; writing—original draft preparation, M.B., H.X. and K.S. All authors have read and agreed to the published version of the manuscript.

Funding: This work was financially supported by the Special Project on Digitization in Education of the Jilin Educational Scientific Research Leading Group under grant JS2338, key project under grant ZD21100, and Social Science Research of the Education Department of Jilin Province (JJKH20231054SK).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: The data that support the findings of this study are available from the corresponding author upon reasonable request.

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

References

1. Rahali, A.; Akhloufi, M.A. End-to-end transformer-based models in textual-based NLP. *AI* 2023, 4, 54–110. [CrossRef]


10. Kim, S.; Nam, J.; Ko, B.C. Facial Expression Recognition Based on Squeeze Vision Transformer. Sensors 2022, 22, 3729. [CrossRef]


31. Wen, Z.; Lin, W.; Wang, T.; Xu, G. Distract your attention: Multi-head cross attention network for facial expression recognition. Biomimetics 2023, 8, 199. [CrossRef]


**Disclaimer/Publisher’s Note:** The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.