Weighted Spatiotemporal Feature and Multi-task Learning for Masked Facial Expression Recognition

Shiwei He
hsw123@email.swu.edu.cn

Yingjuan Jia
yingjuanjiajyj@163.com

Hanpu Wang
wanghp568@gmail.com

Xinyu Liu
liu1223xy@email.swu.edu.cn

Jianmeng Zhou
f1zhoujm@163.com

Huijie Gu
ghj445926@163.com

Mengyan Li
lmy123099@163.com

Tong Chen
Southwest University
Chongqing, China
c_tong@swu.edu.cn

Abstract

Human facial expressions are a vital form of non-verbal communication that convey significant emotional information. Occasionally, individuals exhibit facial expressions that do not correspond to their genuine emotions, which are termed masked facial expressions (MFEs). Automatic recognition of MFEs can reveal individuals' real feelings. However, the complexity inherent in MFEs, resulting from various factors, poses a significant challenge for extracting discriminative representations from MFE videos. In this study, we innovatively designed a multi-task learning network to decompose the challenging task of mixed expression recognition task into two simpler sub-tasks, thereby alleviating the learning burden on the network. Furthermore, we propose a method called spatiotemporal feature modulation (STFM), which comprises a spatiotemporal feature extractor (STFE) and feature weighting module (FWM), aiming to enable the network to focus on features that are beneficial for classification. Additionally, we developed an adaptive spatial attention module (ASAM) to eliminate redundant data in videos and enhance model efficiency by leveraging the action information embedded in dynamic images. The experimental results demonstrate that our method excels in most challenging 36-class task, achieving an accuracy improvement of 4.75% and 0.77% on the 36-class task and 6E task respectively compared to the current state-of-the-art methods.

Keywords: Masked facial expression, Emotion recognition, Multi-task learning, Dynamic image, Transformer.

1. Introduction

Humans have the remarkable ability to convey a wide range of emotions through complex facial muscle movements, which are commonly known as facial expressions [2, 10, 18]. Studies have indicated that 55% of human emotional information is communicated through facial expressions [15]. Facial expressions provide a direct insight into an individual's internal emotional state. Occasionally, facial expressions are consistent with genuine emotions, but in certain situations, individuals may disguise their real feelings by employing specific facial expressions due to social strategies, cultural habits, or personal defense mechanisms. For instance, during a negotiation, even if someone feels angry at the other party's provocation, they may choose to smile instead of showing anger in order to avoid revealing their vulnerability [4]. These inconsistent expressions, which arise from conscious control of facial muscles that do not reflect genuine emotions, are termed masked facial expressions (MFEs) [8]. Research suggests that MFEs occur quite frequently in human daily life [20]. Even children are capable of distinguishing between real emotions and those deliberately expressed [7]. In MFEs, individuals are unable to fully suppress the leakage of their real emotions [21]. Consequently, by accurately identifying and interpreting individuals' MFEs, it is possible to infer their current emotional state. This technology has a multitude of applications, including medical diagnosis, business negotiation, judicial interrogation, and human-computer interaction.

In recent years, the rapid advancement of computer vision technology has significantly propelled the field of automatic expression recognition. In addition to the traditional automatic expression recognition, novel research areas such as micro-expression recognition have also emerged [26, 19,

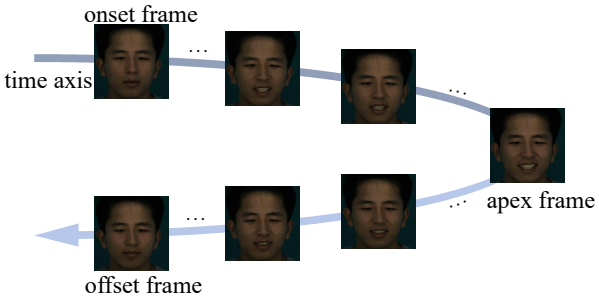

Figure 1. A segment of the "sad to happy" sequence for subject 18

11]. As a new challenge in this field, the progress of research on MFEs has historically been constrained by the lack of dataset resources. It was not until 2021 that the Institute of Psychology of the Chinese Academy of Sciences established the first masked facial expression dataset (MFED) [16], effectively addressing this shortage. In the dataset, participants would first be induced with the experienced emotion through an emotional video. Then they would be asked to immediately complete some required expressions. The required expression may or may not be consistent with the experienced emotion. The combination of 6 experienced emotions and 6 required expressions gives a total of 36 mixed expressions. Among these, when the experienced emotion is consistent with the required expression, they form ordinary macro-expressions, totaling of six. The remaining 30 expressions, where the experienced emotion does not match the required expression, are categorized as masked facial expressions. The MFED uses video clips from the moment participants received an emotional cue (onset) to the moment their expression ended (offset). For each sequence of MFEs, for example, "sad to happy" indicates that the experienced emotion is sad and the required expression is happy. Figure 1 shows a segment of the "sad to happy" sequence for subject 18 from the MFED and the apex frame marking the moment when the required expression reaches its highest intensity.

After the publication of MFED, Zhou et al. [32] firstly conducted a further study and found that the onset frames do not necessarily contain the leakage of the experienced emotion, while the apex frames are more likely to contain the features of the required expressions. Therefore it is difficult to accurately identify MFEs by simply using the onset and apex frames. Subsequently, through statistical testing of LBP-TOP features, Zhou et al. demonstrated that even when the required expression is the same, there are significant differences in the facial action units (AUs) movement patterns when participants use a different expression to mask their experienced emotions. In the task of recognizing masked expressions, the key lies in how to extract discriminative features from expression sequences. This process should make full use of the AU movement pattern characteristics of different masked expressions.

Traditional facial expression recognition tasks typically involve classifying the basic six categories of facial expressions [22], including happiness, sadness, anger, surprise, disgust, and fear. Despite the intricate nature of facial muscle movements, the six expressions exhibit recognizable patterns of facial muscle activity [3]. However, in contrast to typical facial expressions, masked facial expressions (MFEs), as illustrated in Figure 2, are often dominated by the required expression's spatial features. This dominance results in more similar spatial characteristics among mixed expressions when the required expression is the same, thereby making them more challenging to differentiate. Moreover, the required expression and the experienced emotion are deeply coupled. Throughout the sequences of MFEs, the leakage of the experienced emotion can happen at any moment and appear in various forms. For instance, it might only influence the display of AUs related to the required expression, whereas in other cases, it could also show AUs that consistent with the experienced emotion. Although CNN-based methods have achieved good recognition performance in macro-expressions and micro-expressions, they can not effectively capture long-range dependencies and lack the ability to address deep coupling issues.

Figure 2. The onset frames of the six mixed expressions for subject 18 when required expression is happy, where (1)-(6) represent the experienced emotions of anger, disgust, fear, happiness, sadness, and surprise, respectively.

In this study, we propose a transformer-based multi-task spatial-temporal weighting network (MTSTWN) for recognizing MFEs. To address the issue of spatial feature similarity in MFEs under the same required expression, MTSTWN introduces the spatiotemporal feature modulation(STFM) method. In the STFM, the spatiotemporal feature extractor (STFE) explores the spatiotemporal features within the feature maps, enabling the network to fully leverage temporal information. Subsequently, the feature weighting module (FWM) applies spatiotemporal feature weights to the extracted features, aiming to enable the network to focus on learning critical features while ignoring irrelevant ones. To solve the challenge of deep coupling between genuine emotions and posed expressions, MTSTWN applies a multi-task learning framework to MFEs recognition task. This framework simplifies the complex task of 36 mixed expressions classification into two simpler sub-tasks: 6 experienced emotions classification and 6 required expressions

classification. This approach not only significantly reduces the difficulty of classification but also effectively separates the two types of expression features, thereby enhancing the robustness of the features learned by the model. Furthermore, by utilizing the Transformer architecture, MTSTWN can efficiently capture long-range dependencies, enabling more accurate extraction of cues related to the experienced emotion from the overall features. MTSTWN also includes an adaptive spatial attention module (ASAM) based on dynamic images. It generates dynamic images from sequential data and uses a lightweight network to learn a mask, which is applied to the feature map. This ensures that the extracted features focus on the regions where actions occur, reducing redundant information in the video.

The contributions of this paper are summarized as follows:

1.The STFM module is proposed to enhance the model's ability of learning spatiotemporal features while suppressing the impact of inter-class similar features in the spatial domain.

2.A multi-task learning framework is innovatively introduced into the field of masked facial expression recognition. It enables the network to more effectively learn the disentangled features of two types of expressions.

3.We propose the ASAM module, which directs the network to focus on expression-related regions while minimizing the impact of irrelevant areas, thereby further improving the model's attention to key features.

2. Related work

2.1. Automatic recognition of masked facial expression

When emotions arise, people can adopt different strategies to conceal their true feelings. If they choose to maintain a neutral expression, micro-expressions will appear when the suppression of genuine emotions fails. If they use another expression to replace the one that corresponds to their genuine emotions, this is referred to as a masked expression [32]. After the release of MFED, Zhou et al. [32] conducted further research. Using statistical testing methods, it was demonstrated that the AU movement pattern of different masked expressions exhibit distinct differences. Based on this, they proposed a special spatiotemporal handcrafted feature called dynamic AU intensity feature (DAIF). This feature captures the intensity of different AUs in each frame of the masked expression sequence and uses a weighting module to amplify the pattern differences between various expressions. The DAIF is then fed into a network with a visual transformer architecture to perform the classification task, achieving a significant improvement in recognition accuracy compared to the baseline. In addition, there are other works that further contribute to this field. Zhang et al. [29] employ CNNs (VGGNet, GoogLeNet, ResNet, MobileNet) along with data augmentation and regularization techniques, achieving a notable improvement in recognition performance compared to traditional methods. Similarly, Liu et al. [13] introduce a transfer learning approach using pre-trained ResNet18, combined with data augmentation, to enhance the model's generalization ability.

2.2. Multi-task Learning

Multi-task Learning (MTL) is a paradigm in machine learning. Unlike the traditional approach where tasks are executed independently, MTL simultaneously optimizes multiple related tasks, uncovering and leveraging the common features required by these tasks. This encourages the model to learn shared representations that can generalize across tasks, thereby improving the model's learning efficiency and generalization ability [30]. In MTL, the total loss is a combination of the losses from various related tasks. For each task, the other related tasks can be viewed as constraints on that task, which effectively narrows the hypothesis space of the MTL model by handling multiple tasks at once, thereby reducing the risk of over-fitting [27]. Multi-task Learning is also widely applied in expression recognition. For example, Li et al. [12] introduced the poker face vision transformer (PF-ViT) in facial expression recognition, simultaneously performing emotion recognition and generating emotionless faces, integrating multi-task learning to achieve disentanglement between emotion-related and emotion-irrelevant features. Hu et al. [9] designed a new sparse multi-task learning framework that combines handcrafted features and deep features to achieve micro-expression recognition. Zheng et al. [31] proposed DDMTL, which utilizes deep multi-task learning to integrate category label information and sample space distribution information for recognizing facial expressions, demonstrating superior performance even with less training data. Savchenko [23] researched the application of multi-task learning in facial recognition and attribute classification (including age, gender, and ethnicity), achieving excellent emotion classification performance on the AffectNet dataset. Nie et al. [17] introduced the GEME network, which treats gender recognition as an auxiliary task to assist the main task of micro-expression recognition, thereby improving the accuracy of micro-expression recognition.

2.3. Dynamic image

Obtaining precise representations of videos is a core challenge in the field of video understanding. Fernando et al. [6] argue that if a function can accurately order the frames of a video in time based on the video's appearance, then this function can effectively capture the evolution of appearance within the video. By learning such an ordering function, the parameters of these functions can be used as a new representation of the video. Based on this idea,

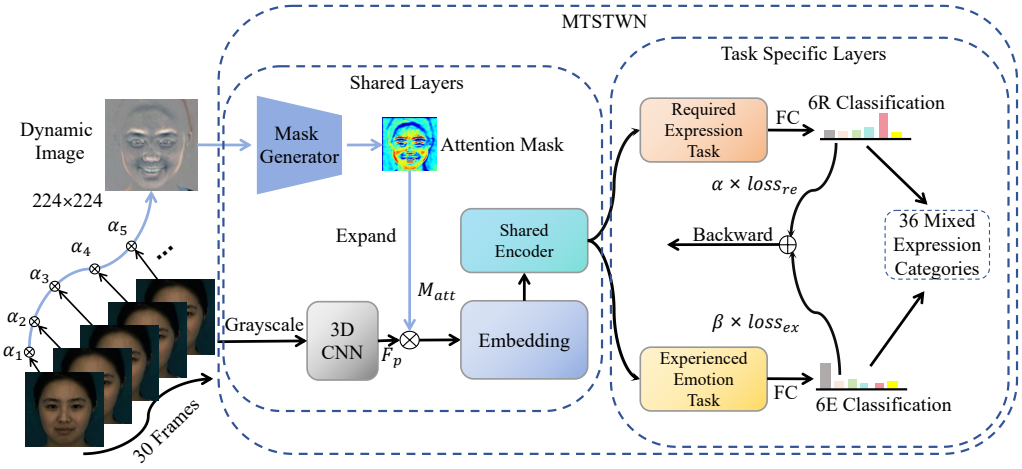

Figure 3. Structure diagram of the multi-task spatial-temporal weighting network (MTSTWN)

Bilen et al. [1] proposed the concept of dynamic image, where the temporal average features of each frame in the video are treated as vectors to be ordered. They learn to find a target vector (i.e. the dynamic image) such that the inner product between this target vector and the average feature vectors at each time point reflects the temporal order of the video frames. Since the target vector has the same dimensions as the frames, it can be represented as a standard RGB image, namely the dynamic image. The dynamic image not only succinctly captures the main visual content of the video but also retains important temporal information. The advantage of this method lies in its ability to compress the spatiotemporal information of the video into a single image, allowing traditional image processing techniques to be directly applied to video analysis tasks. For example, Verma et al. [25, 24] introduced the dynamic image into the field of micro-expression recognition. They employed convolutional neural networks to process the dynamic images generated by micro-expression sequences, extracting multi-scale features that contain both temporal and spatial information, thereby significantly improving the accuracy of micro-expression recognition.

3. Method

3.1. Overview

In the task of masked facial expressions (MFEs) recognition, it is crucial to accurately classify two distinct types of facial expressions. One is the individual's genuine inner emotions, also known as the experienced emotion. The other is the false expressions that individuals deliberately present to conceal their true feelings, referred to as the required expression. The framework of the proposed MTSTWN is shown in Figure 3. Specifically, this study utilizes a multi-task learning framework that combines knowledge learned from two recognition tasks: required expressions and experienced emotions, to classify 36 types of mixed expressions. As shown in Figure 3, after image preprocessing, the video sequences are normalized to 30 frames. Next, a fast dynamic image algorithm [1] is utilized to compute the dynamic image for each sequence (see section 3.3). Subsequently, the dynamic images are input into a lightweight network to generate the attention mask. Meanwhile, the sequence data is converted into grayscale images and sent to the CNN network, where 3D convolution is used to extract primary features. The attention mask is then applied to the obtained feature map, and the results are sent into the multi-task learning framework for further feature extraction and classification. For the multi-task learning framework, this study adopts a hard parameter sharing strategy. The input data first passes through layers with shared parameters to extract features common to both tasks. Within the shared layers, a spatiotemporal feature modulation mechanism is incorporated (see section 3.5). This mechanism is composed of two components. The first is the spatiotemporal feature extractor (STFE), which is capable of comprehensively extracting the spatiotemporal features hidden within the video sequences. The second component is the spatiotemporal feature weighting module (FWM). This module enables the model to focus on the features that are conducive to classification. At the output stage, task-specific methods are applied to extract task-relevant features and complete the classification. The model is optimized by combining the weighted losses from both tasks.

3.2. Preprocessing

Before feeding the video frame data into the network, this study applies a series of preprocessing strategies aimed at ensuring the network focuses on facial information while removing background and other irrelevant features. First, the MTCNN algorithm [28] is used to detect faces in each

video frame and crop them to a uniform size, effectively eliminating background and unrelated content. Next, three landmarks of a standard facial model are selected as references to construct an affine transformation matrix, aligning the faces to mitigate issues caused by inconsistent facial angles or head movements during the experiments. Following that, the temporal interpolation model (TIM) [33] is applied to interpolate the data, standardizing all video sequences to the optimal 30 frames [16]. Finally, the preprocessed data is structured into dimensions of $3 \times 30 \times 224 \times 224$ to serve as the input for the network.

3.3. Adaptive spatial attention module (ASAM)

Unlike image classification tasks, video understanding involves a significant amount of redundant information. In the case of MFEs sequences, preprocessing video frames can help eliminate background clutter, allowing the model to focus more on facial muscle movement features. Even within the facial region, some features, like those around the tip of the nose, may not contribute much to expression classification. Thus, for the task of recognizing MFEs, the network needs to concentrate on areas where significant motion changes occur. To achieve this, this study introduces an adaptive spatial attention module based on dynamic images. This module guides the network in identifying which facial features are most relevant for classifying MFEs, ultimately enhancing recognition performance.

3.3.1 Dyanmic image

Dynamic image, a concept first introduced by Bilen et al. [1] for action recognition, serve as an effective representation of video frame sequences. They encapsulate the features of actions, including the spatial features of the areas where those actions occur. Compared to using onset frames to predict action location information, dynamic images can capture more relevant action information.

(1) anger to anger (2) anger to surprise

Figure 4. Illustration of dynamic images. The experienced emotion and required expression for (1) are both anger, (2) has a experienced emotion of anger and required expression is surprise. For both (1) and (2), the left side shows the sequence onset frame, and the right side displays the dynamic image.

Figure 4 illustrates a comparison between the onset frame of a sequence sample and the dynamic image generated from that sequence. In Figure 4 (1), movement occurs in the eyebrow and mouth regions, which is also reflected in the dynamic image. Conversely, for pixels where no movement occurs, the dynamic image tends to average out the information. Moreover, in Figure 4, when observing the mouth region in (1) and (2), it is evident that the greater and more intense the movement, the more pronounced the relevant areas appear in the dynamic image. Therefore, we can utilize the dynamic image to localize facial action regions.

Given a video data $V = \{I_1, I_2, \ldots, I_n\}$ with n frames, where i is the index of the i-th frame in the video, the dynamic image can be computed using a fast algorithm. The process is as follows:

$$DI(V) = \sum_{t=1}^{n} \alpha_t I_t \tag{1}$$

Where α_t is computed using the following formula:

$$\alpha_t = 2(n-t+1) - (n+1)(H_n - H_{t-1}) \tag{2}$$

Here, $H_t = \sum_{j=1}^{t} 1/j$ is the j-th harmonic number, with $H_0 = 0$. After obtaining the dynamic image, we further attempt to learn the spatial attention mask for each sequence's dynamic image.

3.3.2 Mask Generator

To learn which parts of the facial area deserve more attention, we first extract the dynamic image for each sequence. Then, we use a mask generator to create a mask for the feature map and apply it to the feature map. This study employs a lightweight network design, specifically a 2D convolutional layer, as the mask generator to avoid introducing excessive additional parameters. The specific process is as follows:

$$M_{att} = Expand(2DConv(DI)) \tag{3}$$

Here, M_{att} is the generated attention mask. Applying the attention mask to the feature map F_p can be expressed as:

$$\widehat{F}_p = M_{att} \times F_p \tag{4}$$

Here, $\widehat{F}_p$ is the feature map after applying the attention mask.

3.4. Multi-task learning framework

In MFEs, experienced emotions and required expressions are deeply coupled, and this coupling manifests in various ways. For example, the leakage of genuine expression may occur or not occur in the first frame of the sequence. Furthermore, the extent to which experienced emotions influence required expressions varies throughout the entire frame sequence, making it challenging to directly extract features from the 36 types of mixed expressions. To address this challenge, this study employs a multi-task learning framework with hard parameter sharing, dividing the

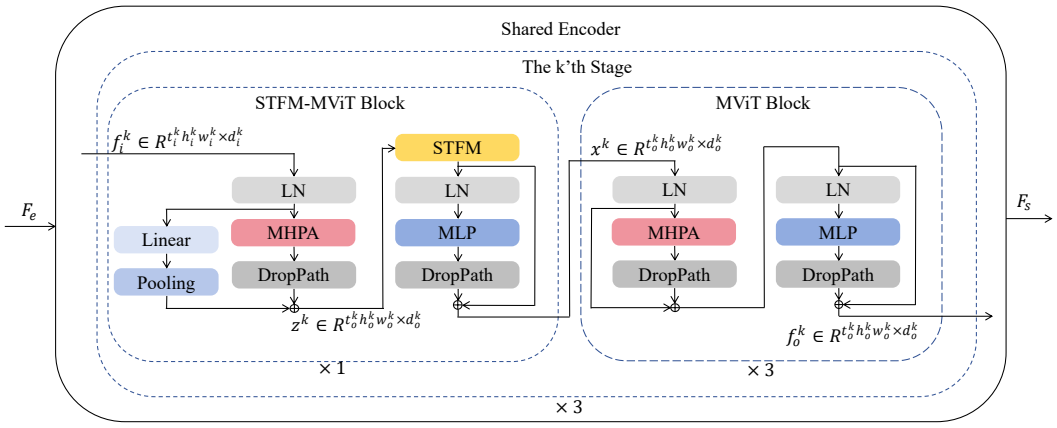

Figure 5. Structure diagram of the shared encoder

network structure into shared layers and task-specific layers. The shared layers extract common emotional features from the frame sequences, while the two task-specific layers focus on mining effective features from the outputs of the shared layers, allowing for a certain level of decoupling between required expressions and experienced emotions. This approach reduces the complexity of classification.

3.4.1 Shared layers

The shared layers are divided into two parts, as shown in Figure 3. The first part consists of the 3D convolutional neural network (CNN) and the mask generator, while the second part is the shared encoder. Given a video sequence, after preprocessing and grayscale conversion we obtain a video $V \in R^{1\times30\times224\times224}$ that is sent to the 3D CNN to extract primary features from the frame V while reducing the scale of the feature map in both spatial and temporal dimensions. The feature map $F_p \in R^{C\times T\times H\times W}$ is derived from the CNN, where C is the number of channels in the feature map after CNN, T is the size in the temporal dimension, and H and W are the height and width of the feature map. After applying the attention mask to the feature map, the results are fed into the patch embedding module to obtain $F_e \in R^{THW\times D}$, where D is the dimension of the embedding vector. Finally, F_e serves as the input to the shared encoder.

The structure of the shared encoder is shown in Figure 5. The shared encoder is responsible for extracting the shared features of experienced emotions and required expressions from the frame sequences while eliminating emotion-irrelevant features from the data. Considering that the characteristics of genuine emotions are embedded within the entire sequence, the network must possess strong long-range modeling capabilities. To balance model performance with computational complexity, this study employs multiscale vision transformers (MViT) [5] as the backbone. The shared encoder consists of three stages, with the first block of each stage being an MViT block that includes the spatiotemporal feature modulation module, called the STFM-MViT block. Each block includes a multi-head pooling attention (MHPA) module, which differs from the traditional multi-head attention module. Before performing the multi-head attention calculations, the MHPA module first pools the query, key, and value. The output length of the MHPA is determined solely by the length of the query. Moreover, if the length of the query decreases, the channel dimension of the output features will increase. The computation process of the self-attention mechanism in the MHPA module of the first block of the k-th stage is summarized as follows:

Firstly, the input is passed through a linear transformation, which can be represented by the following formula:

$$\widehat{Q} = LN(f_i^k)W_Q \tag{5}$$

$$\widehat{K} = LN(f_i^k)W_K \tag{6}$$

$$\widehat{V} = LN(f_i^k)W_V \tag{7}$$

where, As shown in Figure 5, f_i^k represents the input feature of the k-th stage, and LN stands for the layer-normalization, $W_Q, W_K, W_V \in R^{d_i^k\times d_o^k}$ (d_i^k is the input feature dimension of the k-th stage) are all embedding matrices. Then, the input is processed through a pooling layer and performs self-attention calculations, which can be represented by the following formula:

$$Q = P(\widehat{Q};\Theta_Q),\ K = P(\widehat{K};\Theta_K),\ V = P(\widehat{V};\Theta_V) \tag{8}$$

$$Attention(Q,K,V) = Softmax\left(QK^T/\sqrt{d_o^k}\right)V \tag{9}$$

where The operator $P(\cdot;\Theta)$ performs a pooling kernel computation on the input tensor along each of the dimensions. Θ represents the parameters of the pooling operation and

consists of three parts: the pooling kernel k, the stride s, and the padding p. As shown in Figure 5, for the first Block of each stage in MHPA, the pooling stride of the Query is set higher than 1, which reduces the input feature dimensions from $t_i^k h_i^k w_i^k \times d_i^k$ to $t_o^k h_o^k w_o^k \times d_o^k$, To keep the feature sizes consistent in the residual connections, linear and pooling layers are applied to adjust the feature dimensions accordingly. For the last three blocks of each stage, the query's pooling stride is 1, so the input dimensions remain unchanged. Additionally, the first block includes an STFM module to further capture spatiotemporal features and enhance classification performance (see Section 3.5). Overall, as shown in Figure 5, the computation process for the kth stage is as follows:

$$\begin{aligned} z^k =& DropPath(MHPA(LN(f_i^k))) \\ &+ Pooling(Linear(LN(f_i^k))) \end{aligned} \tag{10}$$

$$\begin{aligned} x^k =& DropPath(MLP(LN(STFM(z^k)))) \\ &+ STFM(z^k) \end{aligned} \tag{11}$$

$$f_o^k = MViTBlock^n(x^k),\ n = 3 \tag{12}$$

Here, $DropPath(\cdot)$ is a commonly used regularization technique in transformers, $Linear(\cdot)$ is a linear transformation layer used to change the feature dimension from d_i^k to d_o^k, and $Pooling(\cdot)$ is a pooling layer used to reduce the feature length from $t_i^k h_i^k w_i^k$ to $t_o^k h_o^k w_o^k$. And z^k represents the output of the MHPA module in the first block of the k-th stage, and x^k represents the output of the first block in the k-th stage. $MViTBlock^n(\cdot)$ refers to a sequence of n consecutive MViT blocks. Finally, F_e is processed by the shared encoder to produce F_s which is then passed to the subsequent task-specific layers.

3.4.2 Task-specific layers

As shown in Figure 3, the data passes through the shared layers to generate a shared representation F_s for both the required expression and experienced emotion. In traditional single-task learning, F_s would go through global average pooling and a classification head. However, with 36 types of mixed expressions, identifying discriminative features is challenging because the leakage way of genuine expressions are different across samples. MFEs are already encoded by the human brain as a blend of required expressions and genuine expression, which can be seen as prior knowledge for the network. This helps the network analyze MFEs from both perspectives, making classification easier. Additionally, neural networks are prone to over-fitting, which reduces the test accuracy. Studies show that multi-task learning can effectively reduce the risk of over-fitting [27].

To effectively extract useful features from the shared representation F_s, which comes from a transformer-based network, both task-specific layers utilize the original transformer encoder structure, with parameters that are independent of each other. The computation process can be expressed as follows:

$$F_{ex} = MHA(F_e) \tag{13}$$

$$F_{re} = MHA(F_e) \tag{14}$$

Here, $MHA(\cdot)$ stands for multi-head attention mechanism. F_{ex} and F_{re}represent the features output from the experienced emotion branch and the required expression branch, respectively. Each feature is processed through a global average pooling layer and a fully connected (FC) layer to obtain the classification results p_{ex} and p_{re} for 6 categories. To obtain the classification results for 36 categories, this study defines a matrix $M = \{m_{ij} | 0 \leq i \leq 5, 0 \leq j \leq 5, m_{ij} \in \mathbb{R}\}$, and let $m_{p_{ex}p_{re}} = p_{36}$. meaning that the predicted results p_{ex} and p_{re} are mapped to the corresponding 36-category prediction result.

Both tasks are classification tasks, and this study employs the cross-entropy loss function to calculate the loss for each task individually. The two losses are then combined using a weighted sum to obtain the final loss. The calculation process can be expressed as follows:

First, the cross-entropy loss function is defined as:

$$CELoss = -\frac{1}{N}\sum_{j=1}^{N}\sum_{i=1}^{K} y_{ji} \log\left(\hat{y}_{ji}\right) \tag{15}$$

Here, N is the number of samples, K is the number of expression types, the labels are represented as $Y = \{y_{ji}\}$, and the predicted results are denoted as $\hat{Y} = \{\hat{y}_{ji}\}$, where $j = 1, 2, \cdots, N, i = 1, 2, \cdots, K$.Then, by applying the cross-entropy loss function, we can obtain $Loss_{ex}$ and $Loss_{re}$, and calculate the final loss as follows:

$$Loss = \beta \times Loss_{ex} + \alpha \times Loss_{re} \tag{16}$$

Where β and α are the weights used to adjust the contributions of the experienced emotion task and the required expression task, respectively.

3.5. Spatiotemporal feature modulation (STFM)

Another factor that hinders the improvement of the recognition rate for 36 types of mixed expressions is the inter-class similarity of these expressions under the same required expression. And this similarity is primarily concentrated in the spatial features. For example, in Figure 2, when the required expression is happiness, AU12 is activated across all instances. This similarity makes it challenging for the network to effectively extract distinguishing features for each category. Zhou et al. [32] conducted a

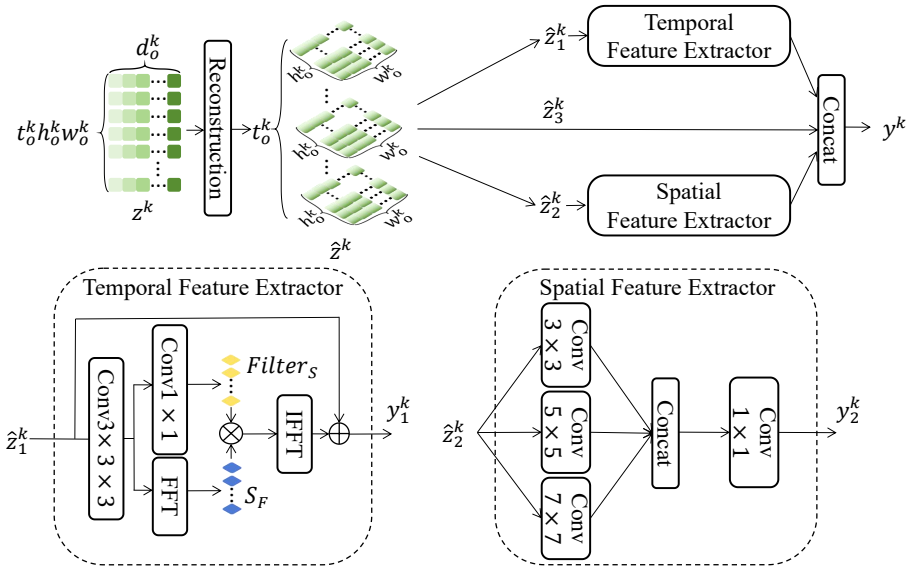

Figure 6. Structure diagram of spatiotemporal feature extractor (STFE)

comparative study by extracting the intensity of action units (AUs) from video frames and found that when the required expression is happy, AU12 is activated, but the degree of activation varies. For instance, when the real emotion is sad, the activation level of AU12 is significantly lower than when the real emotion is happy. To learn these features more effectively, this study proposes a spatiotemporal feature modulation module comprising a spatiotemporal feature extractor (STFE) and a feature weighting module (FWM). The feature extractor is responsible for further extracting spatiotemporal features, while the feature weighting module focuses on identifying which features are most beneficial for recognizing mixed expressions.

3.5.1 Spatiotemporal feature extractor (STFE)

To extract spatiotemporal features more effectively from sequential data, this study introduces a dedicated temporal and spatial feature extraction module. This module splits the input data into three parts along the channel dimension, with one part for extracting temporal features, another for extracting spatial features, and the remaining the left part unchanged. Finally, the three outputs are concatenated in their original order to create the module's final output. Figure 6 illustrates the structure of the STFE.

In the STFM-MViT Block, the output of the first residual structure $z^k \in R^{t_0^k h_0^k w_0^k \times d_0^k}$ (where k indicates the stage in the shared encoder) will serve as the input to the STFE. First, to enable operations along the temporal and spatial dimensions, z^k needs to be reshaped into $\hat{z}^k$, changing the data dimensions from $t_o^k h_o^k w_o^k \times d_o^k$ to $d_o^k \times t_o^k \times h_o^k \times w_o^k$. Next, $\hat{z}^k$ is split into three parts along the channel dimension, a process that can be expressed as:

$$\hat{z}_1^k = S(\hat{z}^k; 0, \frac{1}{3}d_o^k) \tag{17}$$

$$\hat{z}_2^k = S(\hat{z}^k; \frac{1}{3}d_o^k, \frac{2}{3}d_o^k) \tag{18}$$

$$\hat{z}_3^k = S(\hat{z}^k; \frac{2}{3}d_o^k, d_o^k) \tag{19}$$

Where $S(x; start, end)$ denotes slicing x along the channel dimension, starting from $start$ and ending at $end-1$. $\hat{z}_1^k$ is directed to the temporal feature extractor to extract temporal features. $\hat{z}_2^k$ is sent to the spatial feature extractor to extract spatial features. $\hat{z}_3^k$ is retained as a part of output.

The cues for experienced emotions are distributed along the entire sequence, requiring a focus on global temporal features. Although traditional convolutional neural network can extract temporal features, their limited receptive fields may prevent them from fully capturing these global features, impacting the performance. Inspired by DTF [14], this study replaces one-dimensional time domain convolution with frequency domain modulation for temporal feature extraction. This approach is based on the principle that time domain convolution is equivalent to multiplication in the frequency domain. Specifically, the method involves learning the frequency domain filters from the time domain data, multiplying it with the result obtained from applying the fast fourier transform (FFT), and then using the inverse fast fourier transform (IFFT) to return to the time domain. The detailed process for extracting temporal features is as follows:

As shown in Figure 6, in temporal feature extractor, before applying the FFT on the time axis of the feature map,

a 3D convolution with a kernel size of 3 is used to aggregate local spatiotemporal features at each spatial location, enabling the extraction of richer temporal features at every spatial position. Then, the FFT is applied to the time domain data, which can be expressed as follows:

$$S_F = FFT(3DConv(\hat{z}_1^k)),\ S_F \in C^{c \times m \times h_o^k \times w_o^k} \quad (20)$$

Here, S_F is the feature transformed into the frequency domain and $c = d_o^k/3$ represents the number of channels in $\hat{z}_1^k$, and $m = \lfloor t_o^k/2 \rfloor + 1$ denotes the number of frequency domain filters' point number after applying the FFT to the time domain data. Additionally, the time domain data is processed through a convolution layer to learn parameters with dimensions $d_o^k \times m \times h_o^k \times w_o^k \times 2$, which are ultimately combined to form $Filter_S \in C^{d_o^k \times m \times h_o^k \times w_o^k}$. The frequency domain modulation is then performed, followed by the IFFT to obtain the processed time domain data, expressed as follows:

$$y_1^k = IFFT(S_F \times Filter_S) + \hat{z}_1^k \quad (21)$$

where y_1^k is the output of the temporal feature extractor.

In MFEs recognition, spatial features are equally important, requiring the network to capture subtle differences in spatial information. To achieve this, this study employs a multi-scale convolution approach to further extract effective spatial features from the feature maps. Specifically, the input data will merge the time dimension and the batch size dimension before being fed into convolutional layers with kernel sizes of $3 \times 3, 5 \times 5, 7 \times 7$ to extract multi-scale spatial features. The outputs from these layers are then concatenated along the channel dimension. Finally, to maintain consistency between the input and output dimensions, a 1×1 convolution is used for downsampling along the channel dimension. The above process can be summarized as follows:

$$\begin{aligned} y_2^k = &Conv_{1\times1}\{Concat[Conv_{3\times3}(\hat{z}_2^k), \\ &Conv_{5\times5}(\hat{z}_2^k), Conv_{7\times7}(\hat{z}_2^k)]\} \end{aligned} \quad (22)$$

Here, y_2^k represents the output of the spatial feature extractor. Finally, the outputs from the three parts are concatenated to obtain the output of the STFE module, expressed as follows:

$$y^k = Concat(y_1^k, y_2^k, \hat{z}_3^k) \quad (23)$$

Here, y^k represents the output of the STFE.

3.5.2 Feature weighting module (FWM)

After passing through the STFE module, the features produce the output y^k. Next, the FWM module learns which parts of the features are more beneficial for classification. The structure of the FWM is illustrated in Figure 7.

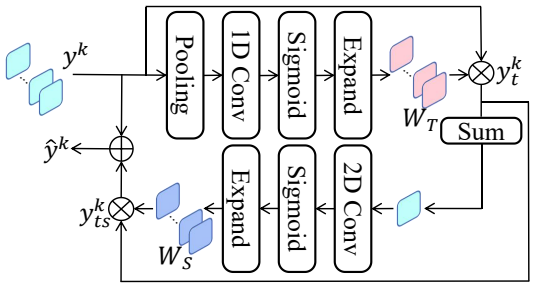

Figure 7. Structure diagram of feature weighting module (FWM)

The FWM consists of two stages. The first stage focuses on learning the weights for each time point in the feature map, while the second stage learns the weights for each spatial location. Specifically, the FWM takes $y^k \in R^{d_o^k \times t_o^k \times h_o^k \times w_o^k}$ as input. It first applies average pooling to compress the spatial information, resulting in a data dimension of $d_o^k \times t_o^k \times 1 \times 1$ This data is then fed into a one-dimensional convolution to learn the weights for each time point, followed by a sigmoid activation function to obtain the final weights. Once the weights are acquired, they are broadcasted across the spatial dimension to yield $W_T \in R^{d_o^k \times t_o^k \times h_o^k \times w_o^k}$. Finally, the weights W_T are applied to y^k, a process that can be expressed as:

$$y_t^k = W_T \times y^k \quad (24)$$

Here, y_t^k is the output of y^k after applying temporal weighting.Similarly, y_t^k undergoes spatial weight learning. First, the temporal dimension information is compressed by summing along the time dimension, resulting in data with dimensions $d_o^k \times h_o^k \times w_o^k$. This data is then passed through a 2D convolution to learn spatial weights, followed by sigmoid function to obtain the spatial weights. After broadcasting, the final weight W_s is obtained, and this weight is applied to y_t^k. This process can be expressed as:

$$y_{ts}^k = W_S \times y_t^k \quad (25)$$

Here, y_{ts}^k is the result of applying both temporal and spatial weighting to y^k. Finally, a residual connection is added to obtain the final output y^k.

4. Experiments

4.1. MFED

The Masked Facial Expression Database (MFED) consists of 778 masked facial expression sequences contributed by 22 participants (including 10 males and 12 females). Each video sequence in this dataset has a resolution of 1280×720 pixels and is recorded at a frame rate of 25 frames per second. There are four classification tasks within this dataset: one for classifying 36 types of mixed expressions,

one for classifying 6 types of experienced emotions (referred to as 6E), one for classifying 6 types of required expressions (referred to as 6R), and a binary classification task to distinguish whether facial expressions are disguised.

4.2. Experimental setup

All experiments in this study were conducted using the leave one-subject-out (LOSO) strategy for validation. The 36-class classification task served as the primary task for module ablation experiments. Model performance was evaluated using accuracy, F1 score, and recall as metrics. During the training phase, random rotation was employed for data augmentation, and the SGD optimizer was chosen.The learning rate of the SGD optimizer is set to 0.0001, the momentum is set to 0.9, and the weight decay is set to 0.0005.

4.3. Ablation experiment

4.3.1 Ablation of MTL

As shown in Table 1, firstly, we conducted baseline experiments using a traditional single-task learning approach based on MViT. The results showed an accuracy of 19.28%, an F1 score of 18.70%, and a recall of 19.18%. These results indicate that relying solely on a single-task expression recognition approach is insufficient to effectively capture the complexity of MFEs. To improve the model's performance, we introduced a multi-task learning (MTL) strategy, dividing the original 36-class mixed expression classification task into two subtasks: requested expression recognition and experienced emotion recognition. With MTL, the model's accuracy increased to 21.08%, the F1 score reached 20.32%, and the recall rose to 21.03%. Compared to single-task learning, multi-task learning improved accuracy, F1, and recall by 1.8%, 1.62%, and 1.85%, respectively. This significant improvement demonstrates that MTL helps the model better understand MFEs from multiple perspectives and enhances its ability to distinguish complex expressions. Building on this, we further incorporated the spatiotemporal feature modulation (STFM) module and the adaptive spatial attention module (ASAM) to improve model performance and conducted an ablation study with MTL. The results showed that, with the integration of both two modules and MTL, the model's accuracy improved to 26.86%, the F1 score increased to 26.25%, and recall rose to 26.85%. Compared to the results without MTL, accuracy, F1, and recall improved by 2.57%, 3.09%, and 2.59%, respectively. These results indicate that our model achieved state-of-the-art (SOTA) performance in the most challenging 36-class classification task.

To directly demonstrate the performance of our method on the 36-classification task, Figure 8 shows the confusion matrix of the experimental results using multi-task learning along with the ASAM and STFM modules, from which it can be intuitively seen that our method performs best in the "sad to happy" category.

Task	Methods	Accuracy	F1	Recall
36 Mixed Expressions	Baseline	19.28%	18.70%	19.18%
	MTL	21.08%	20.32%	21.03%
	ASAM+STFM	24.29%	23.16%	24.26%
	ASAM+STFM+MTL	**26.86%**	**26.25%**	**26.85%**

Table 1. Ablation experiments of Multi-task learning.

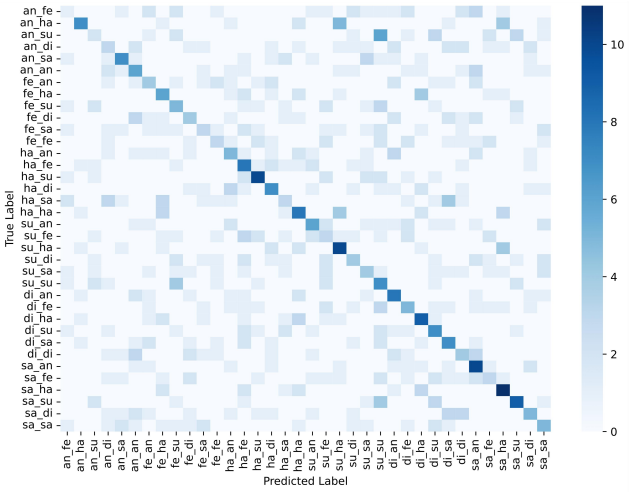

Figure 8. Confusion matrix of the mixed expression recognition

To further validate the effectiveness of multi-task learning, we conducted ablation studies on both task 6E and task 6R using MTL. The experimental results are detailed in Table 2. Notably, when performing task 6E, the loss function weight for the 6R task branch was set to 0.1, similarly, when handling task 6R, the loss function weight for the 6E task branch was also set to 0.1. This design is intended to prevent the auxiliary task from overly influencing the main task.

Task	Methods	Accuracy	F1	Recall
6E	ASAM+STFM	39.08%	38.53%	38.99%
	ASAM+STFM+MTL	**42.93%**	**42.78%**	**42.88%**
6R	ASAM+STFM	58.94%	58.42%	58.94%
	ASAM+STFM+MTL	**62.34%**	**61.88%**	**62.39%**

Table 2. Comparison of single-task learning and multi-task learning for 6E and 6R

In the first group, the experimental results of the network on the 6E task are presented. When using the ASAM and STFM modules with multi-task learning, the accuracy, F1 score, and recall achieved improvements of 3.85%, 4.34%, and 3.89%, respectively. This indicates that for the 6E task, introducing a task branch for required expressions helps the network learn the features of the genuine expressions hidden under the required expressions. In the second group,

the experimental results of the network on the 6R task are presented. The results show that when employing multi-task learning, the accuracy, F1 score, and recall improved by 3.40%, 3.46%, and 3.45%, respectively. This also demonstrates that introducing auxiliary branches with appropriate weights benefits the classification of required expressions. By incorporating multi-task learning, the risk of over-fitting in the neural network can be reduced, which is also an important reason for the improvement in recognition accuracy.

To investigate the contribution of different tasks to the overall recognition rate, we conducted a series of experiments to examine the impact of varying loss weights on the performance of the 36-class classification. The experiments did not employ any additional modules but instead introduced multi-task learning based on MViT. The results are shown in Table 3.

β / α	**Accuracy**	**F1**	**Recall**
0.8/1.2	19.54%	18.87%	19.55%
0.9/1.1	18.38%	17.75%	18.35%
1.0/1.0	**21.08%**	**20.32%**	**21.03%**
1.1/0.9	20.44%	19.93%	20.37%
1.2/0.8	19.67%	19.17%	19.65%

Table 3. Experiment on the values of β and α

when the loss weights of the two tasks are equal, the overall recognition accuracy reached its highest point at 21.08%. Furthermore, the results shows that when the weight of experienced emotions is set to 1.1 and 1.2, the recognition performance is better than when the same weights are applied to required expressions. This suggests that when the weights of experienced emotions and required expressions are not equal, increasing the weight of experienced emotions is more beneficial for improving the classification accuracy of 36 types of mixed expressions, compared to increasing the weight of required expressions. Moreover, to balance the two tasks, setting their loss weights to be equal is a more reasonable choice.

4.3.2 Ablation of STFM

In this section, we explore the impact of the STFM module on the 36-class classification task. First, we conducted a series of ablation experiments to verify the effectiveness of the STFM module. The results are shown in Table 4.

Task	**Methods**	**Accuracy**	**F1**	**Recall**
36 Mixed Expressions	Baseline	19.28%	18.70%	19.18%
	STFM	22.11%	21.35%	22.07%
	ASAM+MTL	22.50%	22.10%	22.55%
	ASAM+STFM+MTL	**26.86%**	**26.25%**	**26.85%**

Table 4. Ablation of the STFM module

As shown in Table 4, The results show that, compared to MViT without any methods, introducing the STFM module improves the accuracy, F1 score, and recall by 2.83%, 2.65%, and 2.89% respectively in the 36-class classification task. When combined with ASAM and multi-task learning (MTL), adding the STFM module further boosts accuracy, F1 score, and recall by 4.36%, 4.15%, and 4.30%, respectively. This demonstrates that using the STFM module for temporal feature extraction significantly enhances the model's performance.

Next, we evaluated the impact of different structure of STFM. The results are shown in Table 5. The results indicate that when only use temporal features yields an accuracy of 21.08%, while relying solely on spatial features results in a slight decline in performance. These findings indicate that a single feature type is insufficient and that temporal features play a more critical role. After implementing the spatiotemporal feature extractor (STFE), the model's accuracy improves to 21.21%. Moreover, compared to using only the FWM or STFE module, simultaneously employing both (STFM) leads to a more significant improvement, with the recognition accuracy reaching 22.11%. This suggests that integrating temporal feature extraction with feature weighting techniques can significantly improve the model's performance. By leveraging the strengths of both approaches, the model can better capture essential patterns and dynamics in the data, leading to more accurate predictions and enhanced overall effectiveness.

Task	**Methods**	**Accuracy**	**F1**	**Recall**
36 Mixed Expressions	Only Temporal Feature	21.08%	19.93%	21.04%
	Only Spatial Feature	20.82%	20.07%	20.93%
	Only STFE	21.21%	20.50%	21.26%
	Only FWM	20.44%	20.06%	20.48%
	STFM	**22.11%**	**21.35%**	**22.07%**

Table 5. Analysis of the different structures of the STFM module

4.3.3 Ablation of ASAM

In this section, we will investigate the impact of the ASAM module on model performance. First, we conducted a series of ablation experiments, and the results are presented in Table 6.

Task	**Methods**	**Accuracy**	**F1**	**Recall**
36 Mixed Expressions	Baseline	19.28%	18.70%	19.18%
	ASAM	20.18%	19.95%	20.13%
	STFM+MTL	25.45%	24.99%	25.41%
	ASAM+STFM+MTL	**26.86%**	**26.25%**	**26.85%**

Table 6. Ablation of ASAM

In the Table 6, After introducing the ASAM module into MViT, the model's accuracy, F1 score, and recall improved by 0.9%, 1.25%, and 0.95%, respectively. When combined with MTL and STFM, the introduction of the ASAM module further enhanced the accuracy, F1 score, and recall, with increases of 1.41%, 1.26%, and 1.44%, respectively. These results indicate that the spatial attention mechanism of the ASAM module enables the network to focus more on the

critical regions where expression actions occur, effectively reducing the negative impact of redundant information in the video on the model's performance.

To visually demonstrate the effect of ASAM, we selected several samples of the spatial attention masks for visualization, and the results are shown in Figure 9. As shown in the picture, The apex frames of different samples under six different required expressions are displayed, along with their corresponding visualized attention masks. Redder colors indicate higher attention levels, while bluer colors signify lower attention from the model in that area. As shown in Figure 9 (1), when the required expression is anger, AU4 is activated in the apex frame, and the corresponding mask indicates that more attention is focused on the eyebrow area of the face. Similarly, as shown in Figure 9 (5), when the required expression is happiness, AU12 is activated in the apex frame, resulting in increased attention on the mouth area of the face in the mask. This indicates that ASAM, through dynamic images and a lightweight network, enables the model to operate more effectively.

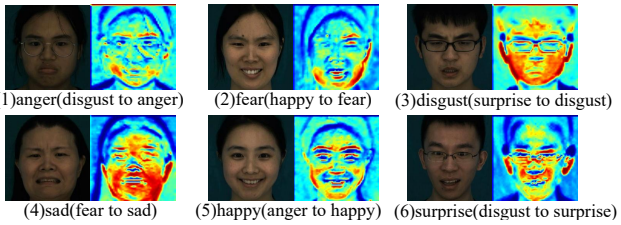

Figure 9. Visualization results of the ASAM module attention mask

4.4. Comparison with other state-of-the-art methods

In this section, we compare our method with other state-of-the-art methods. The results are shown in Table 7

Task	Methods	Accuracy	F1	Recall
36 Mixed Expressions	Ours	**26.86%**	**26.25%**	**26.85%**
	MFED[16]	11.25%	-	-
	ResNet34[29]	20.82%	19.96%	20.82%
	Zhang[29]	22.11%	21.15%	22.11%
	Liu[13]	21.21%	19.83%	21.21%
	3DCNN[32]	10.95%	10.94%	10.94%
	Zhou[32]	21.20%	21.12%	21.03%
6E	Ours	**42.93%**	42.78%	**42.88%**
	MFED[16]	26.83%	-	-
	ResNet34[29]	37.66%	35.38%	37.66%
	Zhang[29]	39.97%	38.93%	39.97%
	Liu[13]	42.16%	41.47%	42.16%
	3DCNN[32]	25.19%	25.38%	25.60%
	Zhou[32]	42.08%	**43.13%**	42.37%

Table 7. Comparison with state-of-the-art methods

In Table 7, we conducted a comparative analysis of the recognition performance in the 36 mixed expression task, as well as the 6 expressions of experienced emotions (6E). The results highlight the performance differences between our method and other existing approaches. To compare the performance differences between image-based methods and video-based methods, we have listed a series of image-based methods. Among them, ResNet34, the methods proposed by Zhang et al. [29] and Liu et al. [13] are image-based methods. Specifically, Zhang et al. achieved the best results by leveraging GoogLeNet in combination with data augmentation techniques. Liu et al., on the other hand, adopted transfer learning methods in an attempt to enhance the model performance. When conducting the 6E task and the 36-mixed expression classification task, the image-based methods use the apex frame at which the expression reaches its maximum intensity as the input to the network. The remaining methods are video-based methods. Among them, 3DCNN is a common method for extracting spatio-temporal features. Zhou et al. [32], in view of the characteristics of MEFs, proposed a dynamic AU intensity feature and achieved very good results.

In the 36 mixed expression task, our method achieved the best performance, with an accuracy of 26.86%, an F1 score of 26.25%, and a recall of 26.85%. Compared to the method of Zhang et al., which was previously the best approach, our method achieves improvements of 4.75%, 5.10%, and 4.74% in accuracy, F1 score, and recall, respectively, for the 36-class classification task. This indicates that the apex frames can hardly fully contain all the information of MEFs. Using video sequences as input and employing multi-task learning in conjunction with the ASAM and STFM modules can more effectively extract discriminative features, significantly enhancing recognition accuracy. In the 6E task, our method also demonstrated exceptional performance, achieving an accuracy of 42.93%, an F1 score of 42.78%, and a recall of 42.88%. Compared to the previous best result, our model achieves an improvement of 0.77% in accuracy.

In summary, across the various tasks compared in the tables, our method consistently performed well in recognizing 36 mixed expressions, 6 experienced emotions. It particularly excelled in mixed emotion classification tasks, highlighting the effectiveness and potential of the models we employed in the field of MFEs recognition.

5. Conclusion

In this study, we propose a multi-task spatiotemporal feature weighting network based on the multiscale vision transformer, designed for masked facial expression recognition. By using a multi-task learning strategy, we simplify the complex 36-class classification task into two more direct tasks: required expression recognition and experienced emotion recognition. This approach allows the network to learn from both perspectives, capturing more robust features. Additionally, the two tasks act as mutual

regularizers, reducing over-fitting. We also introduce a spatiotemporal feature modulation (STFM) module, consisting of a spatiotemporal feature extractor (STFE) and a feature weighting module (FWM). STFE efficiently extracts spatiotemporal features, addressing spatial similarity among mixed expressions with the same required expression, while FWM emphasizes features that aid classification and suppresses irrelevant ones. Moreover, An adaptive spatial attention module (ASAM) enhances the network's focus on regions with significant expression changes, further reducing the impact of irrelevant information. Experimental results demonstrate that our method outperforms existing approaches, improving accuracy by 4.75% on the 36-class task and 0.77% on the 6E task. However, recognizing masked facial expressions remains challenging due to their complexity, suggesting a need for further research. Future research can explore the use of targeted feature extraction methods for different tasks within a multi-task framework, in order to fully leverage the advantages of multi-task learning.

References

[1] H. Bilen, B. Fernando, E. Gavves, A. Vedaldi, and S. Gould. Dynamic image networks for action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 3034–3042, 2016. 4, 5

[2] P. Ekman. Universals and cultural differences in facial expressions of emotion. In *Nebraska symposium on motivation*. University of Nebraska Press, 1971. 1

[3] P. Ekman and W. V. Friesen. Facial action coding system. *Environmental Psychology & Nonverbal Behavior*, 1978. 2

[4] P. Ekman and W. V. Friesen. Felt, false, and miserable smiles. *Journal of nonverbal behavior*, 6(4):238–252, 1982. 1

[5] H. Fan, B. Xiong, K. Mangalam, Y. Li, Z. Yan, J. Malik, and C. Feichtenhofer. Multiscale vision transformers. In *Proceedings of the IEEE/CVF international conference on computer vision*, pages 6824–6835, 2021. 6

[6] B. Fernando, E. Gavves, J. M. Oramas, A. Ghodrati, and T. Tuytelaars. Modeling video evolution for action recognition. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 5378–5387, 2015. 3

[7] P. L. Harris, K. Donnelly, G. R. Guz, and R. Pitt-Watson. Children's understanding of the distinction between real and apparent emotion. *Child development*, pages 895–909, 1986. 1

[8] U. Hess and R. E. Kleck. Differentiating emotion elicited and deliberate emotional facial expressions. *European Journal of Social Psychology*, 20(5):369–385, 1990. 1

[9] C. Hu, D. Jiang, H. Zou, X. Zuo, and Y. Shu. Multi-task micro-expression recognition combining deep and handcrafted features. In *2018 24th International Conference on Pattern Recognition (ICPR)*, pages 946–951. IEEE, 2018. 3

[10] C. E. Izard. Facial expressions and the regulation of emotions. *Journal of personality and social psychology*, 58(3):487, 1990. 1

[11] J. Li, Z. Dong, S. Lu, S.-J. Wang, W.-J. Yan, Y. Ma, Y. Liu, C. Huang, and X. Fu. Cas (me) 3: A third generation facial spontaneous micro-expression database with depth information and high ecological validity. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(3):2782–2800, 2022. 2

[12] J. Li, J. Nie, D. Guo, R. Hong, and M. Wang. Emotion separation and recognition from a facial expression by generating the poker face with vision transformers. *IEEE Transactions on Computational Social Systems*, 2024. 3

[13] Y. Liu, K. Zhao, and X. Fu. Recognition of masked facial expressions based on transfer learning and data augmentation. In *2023 International Annual Conference on Complex Systems and Intelligent Science (CSIS-IAC)*, pages 80–86. IEEE, 2023. 3, 12

[14] F. Long, Z. Qiu, Y. Pan, T. Yao, C.-W. Ngo, and T. Mei. Dynamic temporal filtering in video models. In *European Conference on Computer Vision*, pages 475–492. Springer, 2022. 8

[15] A. Mehrabian. An approach to environmental psychology. *Massachusetts Institute of Technology*, 1974. 1

[16] F. Mo, Z. Zhang, T. Chen, K. Zhao, and X. Fu. Mfed: A database for masked facial expression. *IEEE Access*, 9:96279–96287, 2021. 2, 5, 12

[17] X. Nie, M. A. Takalkar, M. Duan, H. Zhang, and M. Xu. Geme: Dual-stream multi-task gender-based micro-expression recognition. *Neurocomputing*, 427:13–28, 2021. 3

[18] M. Pantic and L. J. M. Rothkrantz. Automatic analysis of facial expressions: The state of the art. *IEEE Transactions on pattern analysis and machine intelligence*, 22(12):1424–1445, 2000. 1

[19] M. Peng, C. Wang, T. Bi, Y. Shi, X. Zhou, and T. Chen. A novel apex-time network for cross-dataset micro-expression recognition. In *2019 8th international conference on affective computing and intelligent interaction (ACII)*, pages 1–6. IEEE, 2019. 2

[20] S. Porter and L. Ten Brinke. Reading between the lies: Identifying concealed and falsified emotions in universal facial expressions. *Psychological science*, 19(5):508–514, 2008. 1

[21] S. Porter, L. Ten Brinke, and B. Wallace. Secrets and lies: Involuntary leakage in deceptive facial expressions as a function of emotional intensity. *Journal of Nonverbal Behavior*, 36:23–37, 2012. 1

[22] J. A. Russell. Core affect and the psychological construction of emotion. *Psychological review*, 110(1):145, 2003. 2

[23] A. V. Savchenko. Facial expression and attributes recognition based on multi-task learning of lightweight neural networks. In *2021 IEEE 19th International Symposium on Intelligent Systems and Informatics (SISY)*, pages 119–124. IEEE, 2021. 3

[24] M. Verma, S. K. Vipparthi, and G. Singh. Affectivenet: Affective-motion feature learning for microexpression recognition. *IEEE MultiMedia*, 28(1):17–27, 2020. 4

[25] M. Verma, S. K. Vipparthi, G. Singh, and S. Murala. Learnet: Dynamic imaging network for micro expression recognition. *IEEE Transactions on Image Processing*, 29:1618–1627, 2019. 4

[26] C. Wang, M. Peng, T. Bi, and T. Chen. Micro-attention for micro-expression recognition. *Neurocomputing*, 410:354–362, 2020. 2

[27] J. Yu, Y. Dai, X. Liu, J. Huang, Y. Shen, K. Zhang, R. Zhou, E. Adhikarla, W. Ye, Y. Liu, et al. Unleashing the power of multi-task learning: A comprehensive survey spanning traditional, deep, and pretrained foundation model eras. *arXiv preprint arXiv:2404.18961*, 2024. 3, 7

[28] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao. Joint face detection and alignment using multitask cascaded convolutional networks. *IEEE signal processing letters*, 23(10):1499–1503, 2016. 4

[29] X. Zhang, Y. Liu, K. Zhao, and X. Fu. Recognition of masked facial expressions based on convolutional neural networks and data augmentation. In *2023 International Annual Conference on Complex Systems and Intelligent Science (CSIS-IAC)*, pages 351–357. IEEE, 2023. 3, 12

[30] Y. Zhang and Q. Yang. A survey on multi-task learning. *IEEE transactions on knowledge and data engineering*, 34(12):5586–5609, 2021. 3

[31] H. Zheng, R. Wang, W. Ji, M. Zong, W. K. Wong, Z. Lai, and H. Lv. Discriminative deep multi-task learning for facial expression recognition. *Information Sciences*, 533:60–71, 2020. 3

[32] J. Zhou, X. Liu, H. Wang, Z. Zhang, T. Chen, X. Fu, and G. Liu. Seeing through the mask: Recognition of genuine emotion through masked facial expression. *IEEE Transactions on Computational Social Systems*, 2024. 2, 3, 7, 12

[33] Z. Zhou, G. Zhao, and M. Pietikäinen. Towards a practical lipreading system. In *CVPR 2011*, pages 137–144. IEEE, 2011. 5