FFNET-M: FEATURE FUSION NETWORK WITH MASKS FOR MULTIMODAL FACIAL EXPRESSION RECOGNITION

Mingzhe Sui, Zhaoqing Zhu, Feng Zhao, and Feng Wu*

University of Science and Technology of China, Hefei 230027, China
{sa20, zhaoqingzhu}@mail.ustc.edu.cn, {fzhao956, fengwu}@ustc.edu.cn

2021 IEEE International Conference on Multimedia and Expo (ICME) | 978-1-6654-3864-3/20/$31.00 ©2021 IEEE | DOI: 10.1109/ICME51207.2021.9428100

ABSTRACT

Compared with 2D facial expression recognition (FER) and 3D FER, 2D+3D FER can handle the effects of illumination changes and pose variations. The combination of 2D texture and 3D attribute information can further improve the performance. However, most existing approaches still face two challenges: the selection of proper networks for extracting multimodal features, and the significance of local features in salient regions for expression classification. To address these challenges, we propose an efficient feature fusion network with masks (FFNet-M) for 2D+3D FER. Each 3D scan is represented by three types of attribute maps (i.e., depth map, normal map, and texture image), which are then fed into FFNet-M with different networks to extract both 2D and 3D features. Moreover, we design two masks to make FFNet-M focus on 2D local features while paying attention to 3D local features in salient regions. Experimental results show that our FFNet-M outperforms state-of-the-art methods on BU-3DFE dataset and also achieves a high accuracy on Bosphorus dataset.

Index Terms— 2D+3D FER, local features, FFNet-M.

1. INTRODUCTION

Automatic facial expression recognition (FER) has a wide range of applications in the fields of emotional robots, medical treatment, fatigue driving detection, and other human-computer interaction areas [1]. Ekman and Friesen [2] classified expressions into six basic categories: anger, disgust, fear, happiness, sadness, and surprise. In general, FER is composed of three parts, including preprocessing, feature extraction, and expression classification. According to the data modality, the FER techniques can be classified into three categories, i.e., 2D FER, 3D FER, and 2D+3D FER. Illumination changes and pose variations are two critical problems in 2D FER [3]. Different radiation angles, illumination source intensities, and poses may cause sharp changes in the grayscale of the generated images, thus reducing the recognition rate significantly. Applying illumination and pose normalization algorithms in preprocessing and using thermal images that are robust to illumination changes are two good ways to alleviate these two problems [1]. Nevertheless, they still have specific limitations such as the loss of subtle face deformations and the lack of texture information [4].

3D face models present better robustness to illumination changes and pose variations, and can also capture subtle deformations caused by expression changes. Note that such deformations are crucial for accurate expression classification. Traditional 3D FER methods can be divided into *model-based* [5] [6] and *feature-based* [7–9] approaches. The former makes use of the training data to build a generic 3D deformable model, which is used to generate the corresponding fitting parameters for each new 3D scan in the testing sets. The latter utilizes conventional feature operators to the depth maps, normal maps, and curvature maps, and then feeds the extracted features into a classifier. However, these techniques need to establish accurate and dense correspondence among face scans and rely heavily on hand-crafted features, limiting the accuracy of expression classification. To solve the problems, deep learning techniques have been introduced into 3D FER and 2D+3D FER in recent years [4] [10–16]. Combining the 2D texture information, 2D+3D FER demonstrates superiority over 3D FER. Li *et al*. [4] first applied convolutional neural network (CNN) to 2D+3D FER, where they fed five types of attribute maps from each 3D scan and one 2D texture image into the deep fusion CNN. The results show that the fused features automatically extracted by deep CNN perform better than hand-crafted features.

Most existing 2D+3D FER methods still face two main challenges. One is the selection of proper networks for extracting multimodal features. Since depth maps are generally smoother than 2D texture images, filters with larger kernel sizes should be employed in CNN to extract 3D features [17]. Nonetheless, most researchers adopted the same network for extracting both 2D and 3D features. The other is the significance of local features in the salient regions for expression classification, which is often ignored in 3D FER and 2D+3D FER but has been proven efficient in 2D FER [18]. Although some recent works exploited different feature extraction networks for 2D texture images and depth maps respectively [10] [15] and other works highlighted the local features in the salient regions through attention mechanisms in

*The corresponding author is Feng Zhao.

978-1-6654-3864-3/21/$31.00 ©2021 IEEE

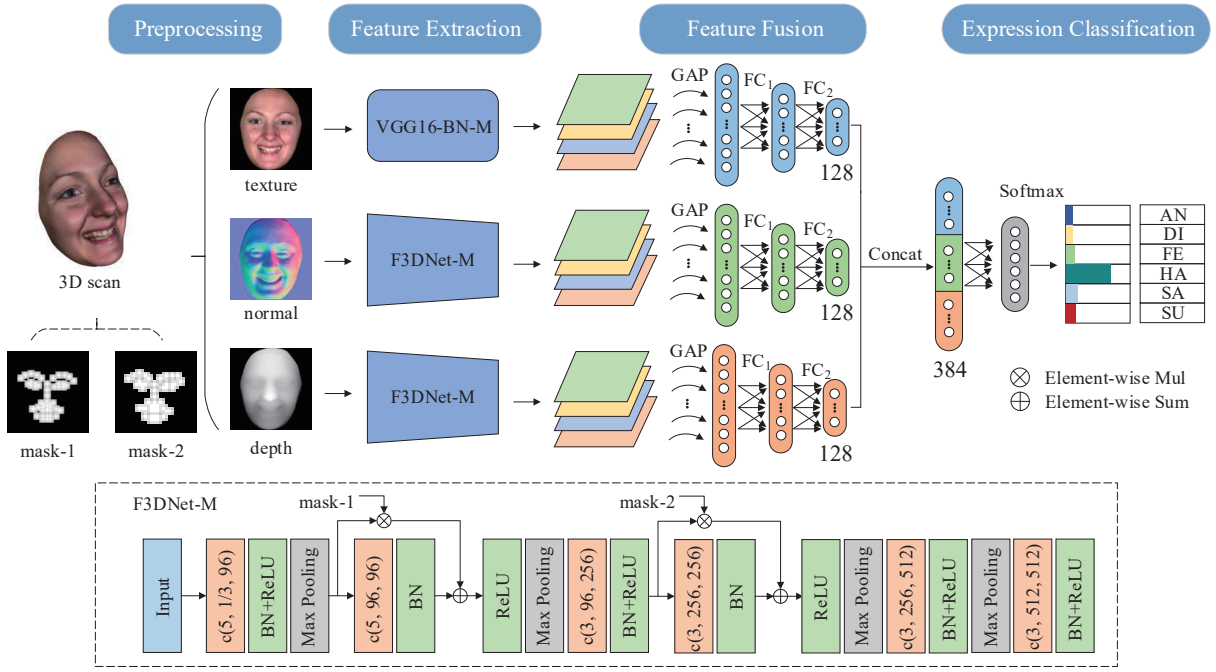


Fig. 1. The framework of the proposed FFNet-M for 2D+3D FER. Multimodal features after global average pooling (GAP) and two fully-connected (FC) layers are concatenated to 384 elements, while the softmax is adopted to predict the expression label.

their networks [13] [14], none of them can deal with the above two challenges simultaneously.

In this paper, we propose a feature fusion network with masks (FFNet-M) for 2D+3D FER to handle both of the two aforementioned challenges. For preprocessing, each 3D scan is represented by one depth map, one normal map, and one 2D texture image. To determine the salient regions in the depth and normal maps, we employ a simple method, which directly makes use of the salient regions in the 2D texture image after the alignment between each pair is done. We then utilize these salient regions to assign the weights and generate two masks with different sizes. Thus, each 3D scan corresponds to five figures. For feature extraction, we exploit different networks to extract both 2D and 3D features, where two F3DNets-M with larger kernel sizes are used for extracting 3D depth and 3D normal features separately, and one VGG16-BN-M is utilized to extract 2D texture features. Furthermore, the incorporation of the two generated masks makes our FFNet-M focus more on both 2D and 3D local features in the salient regions. To reduce the spatial dimensionality, global average pooling is adopted before feature fusion.

Our contributions of this work are listed as follows:

- We propose the FFNet-M that applies two specially designed F3DNets-M with larger kernel sizes and one VGG16-BN-M to extract 3D and 2D features, respectively.
- We generate two aligned masks and incorporate them into the FFNet-M to make it focus more on 2D and 3D local features in the salient regions.
- Our FFNet-M achieves a high accuracy of 87.61% on Bosphorus, and 89.82% on BU-3DFE.

2. METHODOLOGIES

2.1. Overview of FFNet-M

The framework of our FFNet-M is shown in Fig. 1. Given a 3D scan, we first generate one depth map (I_d), one normal map (I_n), and one texture image (I_t), which have been aligned with each other in the first stage of preprocessing. We then utilize the texture image to generate two specific masks in the second stage of preprocessing. In feature extraction, I_d and I_n are fed into F3DNet-M separately, and I_t is fed into VGG16-BN-M. The two generated masks make the FFNet-M focus more on 2D and 3D local features through enhancing the weights of the salient regions. Global average pooling and two fully-connected layers are adopted before feature fusion to reduce the spatial dimensionality.

2.2. Preprocessing

The preprocessing consists of two stages. In the first stage, we apply the *gridfit* algorithm [17] on BU-3DFE and the method in [19] on Bosphorus to generate aligned depth/normal maps with their corresponding 2D texture images. Due to the limitations of 3D face scanning equipment, outliers and holes

(a) Texture (b) Depth (c) Normal (d) Mask-1 (e) Mask-2

Fig. 2. Five figures corresponding to one 3D scan from BU-3DFE (51 landmarks are overlaid on I_t, I_d, and I_n).

often appear in the generated depth maps. To improve their quality, we perform the following steps.

Outlier Removal. Outliers may be caused by the 3D camera or other reasons. To remove them, we calculate the standard deviation σ of the eight neighbors around each pixel. If the depth difference between the center point and its neighbors is greater than 0.6σ, it is considered as an outlier and needs to be removed.

Hole Filling. Some holes may present in the depth maps after projection because of the limited subsampled numbers of points from the original 3D scans. We use bicubic interpolation to fill every hole by calculating the weighted average of its 16 nearest subsampling neighbors. Additionally, the missing points caused by outlier removal are filled as well, which guarantees the integrity of the depth maps.

Noise Removal. To remove the noises brought by the original acquisition settings, we apply Gaussian filtering with a sliding window size of 3×3 and $\sigma = 1$.

Finally, we compute the three normal maps in x, y, and z directions based on each depth map and integrate them into one normal map with three channels. This improves the efficiency since only one F3DNet-M for normal maps is needed.

In the second stage, inspired by [18], we design two masks with weights distributed in the salient regions for the three attribute maps. First, we perform facial landmarks detection by applying the *Dlib* algorithm [20] on the cropped I_t (56×56) to get 68 landmark points. We then select 51 landmarks concentrated in the four salient regions (eyebrows, eyes, nose, and mouth) that are crucial for expression classification, as shown in Fig. 2(a). For I_d and I_n, we directly use the landmarks of I_t as their landmarks because they have been aligned with each other in the first stage of preprocessing, as depicted in Fig. 2(b) and (c). After that, we generate the mask-1 with the same size as the cropped I_t and assign the weights based on:

$$d(P, L) = \max(|P_{\mathrm{x}} - L_{\mathrm{x}}|,\ |P_y - L_y|) \tag{1}$$

$$w(P, L) = \begin{cases} 1 - 0.1 * d_{\mathrm{M}}(P, L) & \text{if } d(P, L) <= 2, \\ 0 & \text{otherwise}, \end{cases} \tag{2}$$

where P is a pixel in the mask and L is a landmark point in I_t. $d_{\mathrm{M}}(P, L)$ is the Manhattan distance between P and L. It can be easily seen that the pixel closer to the landmark has a larger weight in the range of 5×5 centered at each landmark. To deal with overlap, we take the largest $w(P, L)$ when there are multiple landmarks around the same pixel. These 5×5 regions constitute the salient regions, as displayed in Fig. 2(d).

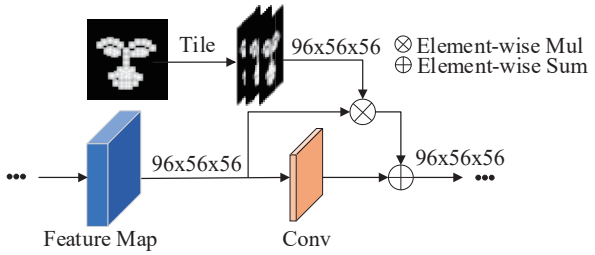


Fig. 3. The connection mode of mask-1 in FFNet-M.

Furthermore, the coordinates of these 51 landmarks in mask-1 are divided by 2 to generate the landmarks of mask-2 with a smaller size of 28×28 approximatively, and accordingly its weights are distributed in the range of 3×3 centered at each landmark, as illustrated in Fig. 2(e).

2.3. Networks

After preprocessing, we apply the FFNet-M to extract 2D and 3D multimodal features, which are composed of two F3DNets-M and one VGG16-BN-M.

F3DNet-M. Since the depth and normal maps are smoother than the 2D texture images, we design two F3DNets-M with larger kernel sizes to extract 3D features, as shown in Fig. 1. It consists of six convolutional layers and four max-pooling layers. The kernel sizes of the first two convolutional layers are both 5×5, and those of the rest are all 3×3. Batch normalization (BN) layer [21] is used following each convolutional layer. Inspired by [18] in 2D FER, we incorporate the two generated masks into each F3DNet-M to focus on 3D local features of depth and normal maps, as shown in Fig. 3. Before the second convolutional layer, we add one residual block and replace the input of the skip connection with the element-wise multiplication of the feature map and mask-1 to enhance the weights in the salient regions. In addition, mask-2 is incorporated into the fourth convolutional layer in the same way. It is worth noting that the sizes of the feature maps before and after the second/fourth convolution layers are exactly the same, thus avoiding down-sampling the input of the skip connection. The only difference between two F3DNets-M for extracting 3D depth features and 3D normal features is the channel number of their first convolutional layer is one or three.

VGG16-BN-M. For 2D texture images, we design the VGG16-BN-M which incorporates the two generated masks at the fourth and sixth convolutional layers in the same way as F3DNet-M based on the backbone of VGG16-BN [21] [22]. The two masks also make the network focus more on 2D local features in the texture images. Considering the limited amount of data, we use the backbone of pre-trained VGG16-BN to initialize the VGG16-BN-M.

Feature Fusion. To effectively combine the extracted 2D and 3D features, we adopt the feature-level fusion strategy. Since direct concatenation of these features could lead to ex-

cessive spatial dimensionality, we first carry out global average pooling, as shown in Fig. 1. The outputs of feature extraction networks are $\mathcal{Y}_t \in \mathbb{R}^{512\times3\times3}$, $\mathcal{Y}_n \in \mathbb{R}^{512\times4\times4}$, and $\mathcal{Y}_d \in \mathbb{R}^{512\times4\times4}$, which represent 2D texture features, 3D normal features, and 3D depth features, respectively. After global average pooling and the first fully-connected layer (FC_1), the dimension of each type of features becomes 256, which is further reduced to 128 by the second fully-connected layer (FC_2). So, the concatenated features after fusion only have 384 elements, which are finally fed into a softmax classifier for classification.

3. EXPERIMENTAL RESULTS

To show the effectiveness of our FFNet-M, extensive experiments are performed on BU-3DFE [23] and Bosphorus [24]. We also conduct a series of ablation experiments to prove the validity of the designed F3DNet-M and VGG16-BN-M.

3.1. Datasets and Evaluation Protocol

BU-3DFE. The BU-3DFE dataset consists of 2500 3D facial expression scans, including 100 subjects (56 females and 44 males) with ages ranging from 18 to 70. Each subject has six basic expressions (i.e., anger, disgust, fear, happiness, sadness, and surprise) with four levels of intensity, and one extra neutral expression with only one level of intensity.

Bosphorus. The Bosphorus dataset contains 4666 3D scans from 105 subjects ranging from 25 to 35 years old. Unlike BU-3DFE, not all files are complete. Only 63 subjects have six basic expressions with one level of intensity.

Evaluation Protocol. To compare with other methods fairly, we follow the protocol used in [4] [13–15]. In this protocol, we randomly select 60 subjects with two highest levels of intensity from 100 subjects for BU-3DFE and 60 subjects from 63 subjects for Bosphorus, which are fixed in all the experiments. Moreover, the protocol uses the average accuracy of 100 times of 10-fold cross-validation to evaluate the performance. So, the selected 60 subjects are randomly divided into ten folds every time. During each round in turn, nine folds with 54 subjects are used for training and the remaining one fold with six subjects is used for testing.

3.2. Implementation Details

The preprocessed depth maps, normal maps, and texture images are resized to $1 \times 112 \times 112$, $3 \times 112 \times 112$, and $3 \times 112 \times 112$, respectively. With a total of seven channels, they are jointly fed into the FFNet-M. In order to save the training time, the two masks of sizes 56×56 and 28×28 corresponding to each 3D scan have been generated before the training stage. The mask sizes of 56 and 28 are empirically selected because the landmarks cannot be detected in the 2D texture images if the sizes are too small.

Table 1. Comparison results on BU-3DFE

Method	Data	Feature	Acc (%)
Soyel *et al*. (2007) [7]	3D	Hand-crafted	61.79
Tang *et al*. (2008) [8]	3D	Hand-crafted	74.51
Gong *et al*. (2009) [6]	3D	Hand-crafted	76.22
Zhen *et al*. (2015) [9]	3D	Hand-crafted	84.50
Yang *et al*. (2015) [25]	3D	Hand-crafted	84.80
Li *et al*. (2015) [26]	2D+3D	Hand-crafted	**86.32**
Li *et al*. (2017) [4]	2D+3D	Deep learning	86.86
Oyedotun *et al*. (2017) [10]	2D+3D	Deep learning	**89.31**
Chen *et al*. (2018) [16]	3D	Deep learning	86.67
Wei *et al*. (2018) [12]	2D+3D	Deep learning	88.03
Zhu *et al*. (2019) [13]	2D+3D	Deep learning	88.35
Jiao *et al*. (2019) [14]	2D+3D	Deep learning	89.11
Lin *et al*. (2020) [15]	2D+3D	Deep learning	89.05
FFNet	2D+3D	Deep learning	89.01
FFNet-M	2D+3D	Deep learning	**89.82**

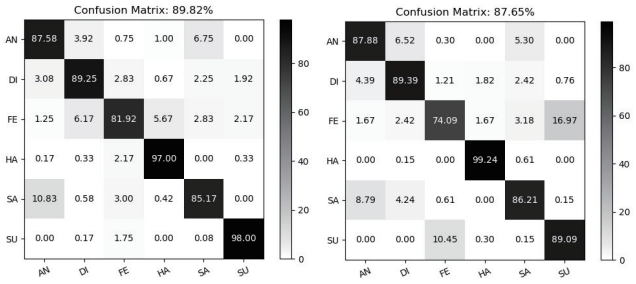
Table 2. Comparison results on Bosphorus

Method	Data	Feature	Acc (%)
Yang *et al*. (2015) [25]	3D	Hand-crafted	77.50
Li *et al*. (2015) [26]	2D+3D	Hand-crafted	**79.72**
Li *et al*. (2017) [4]	2D+3D	Deep learning	80.28
Tian *et al*. (2019) [11]	2D+3D	Deep learning	79.17
Lin *et al*. (2020) [15]	2D+3D	Deep learning	**89.28**
FFNet	2D+3D	Deep learning	86.64
FFNet-M	2D+3D	Deep learning	**87.65**

In the training stage, we apply the backbone of pre-trained VGG16-BN to initialize the VGG16-BN-M, and normal distribution to initialize the F3DNet-M with $\boldsymbol{w} \sim \mathcal{N}(0, 0.02^2)$, $b = 0$ for convolutional layers and $\boldsymbol{w} \sim \mathcal{N}(1, 0.02^2)$, $b = 0$ for BN layers. Random initialization is used for all the fully-connected layers in the FFNet-M, while Adam optimizer with betas $(0.9, 0.999)$ is utilized to optimize the FFNet-M. Besides, we set the learning rate to 0.0001 in 70 epochs for cross-entropy loss function. We train the FFNet-M on NVIDIA Titan 2080-Ti GPU cards using PyTorch.

3.3. Results

Table 1 compares our FFNet-M with state-of-the-art methods on BU-3DFE. It can be seen that the proposed FFNet-M achieves the highest accuracy of 89.82% whether compared with hand-crafted features or deep learning-based features. By removing all the masks in FFNet-M, the classification accuracy of FFNet decreases by 0.81%, which demonstrates the effectiveness of the two masks focusing on local features. Table 2 shows the comparison results on Bosphorus. Similarly, FFNet-M is 1.01% higher than FFNet because of the two generated masks. Our FFNet-M outperforms most methods on Bosphorus as well, and is comparable with



(a) BU-3DFE (b) Bosphorus

Fig. 4. Confusion matrices on BU-3DFE and Bosphorus.

Table 3. Ablation experiments on BU-3DFE

Model	I_d (%)	I_n (%)	I_t (%)	I_{dnt} (%)
F3DNet	82.64	83.86	82.66	86.11
F3DNet-M	**83.61**	**84.71**	82.89	86.95
VGG16-BN	81.94	82.47	83.44	86.67
VGG16-BN-M	82.36	83.17	**84.14**	87.78
FFNet	-	-	-	89.01
FFNet-M	-	-	-	**89.82**

Table 4. Ablation experiments on Bosphorus

Model	I_d (%)	I_n (%)	I_t (%)	I_{dnt} (%)
F3DNet	78.33	80.56	80.83	82.22
F3DNet-M	**79.17**	**81.39**	81.67	83.06
VGG16-BN	77.78	78.33	82.50	83.61
VGG16-BN-M	78.06	79.17	**83.33**	84.44
FFNet	-	-	-	86.64
FFNet-M	-	-	-	**87.65**

OGF^{2}Net [15]. It is worthy to note that FFNet-M is 0.77% higher than OGF^{2}Net [15] on BU-3DFE.

The confusion matrices of FFNet-M on BU-3DFE and Bosphorus are shown in Fig. 4. We can see that our FFNet-M performs well for expressions of happiness and surprise but relatively poor for fear and sadness on both datasets, and the classification accuracy of fear on Bosphorus is just 74.09%. This is consistent with the conclusions of existing methods because the facial features of happiness and surprise are more obvious and discriminative, while fear and sadness have a higher confusion rate.

3.4. Ablation Experiments

To further prove the validity of each module in the FFNet-M, we conduct ablation experiments on BU-3DFE and Bosphorus with six models, as given in the first column of Tables 3 and 4. Note that F3DNet and VGG16-BN mean all the masks are removed.

Evaluation of the masks. We evaluate the masks in F3DNet-M and VGG16-BN-M. Comparing the 3rd, 5th, 7th rows with the 2nd, 4th, 6th rows of Tables 3 and 4, it can be clearly seen that the performance of the networks with masks is better than that of the networks without masks, whether the input is one of I_d, I_n, and I_t or all of them (I_{dnt}). It also proves that enhancing the weights in the four salient regions in I_d, I_n, and I_t to highlight 2D and 3D local features can improve the classification performance in 2D+3D FER.

Table 5. Multimodal experiments

Dataset	Model	Modality	Acc (%)
BU-3DFE	FFNet-M	3D	87.28
BU-3DFE	FFNet-M	2D+3D	**89.82**
Bosphorus	FFNet-M	3D	82.86
Bosphorus	FFNet-M	2D+3D	**87.65**

Evaluation of F3DNet-M and VGG16-BN-M. We use four models to extract features from I_d, I_n, and I_t to prove the effectiveness of selecting different networks for multimodal features. According to the 2nd and 3rd columns of Tables 3 and 4, we can see that F3DNet and F3DNet-M with larger kernel sizes perform better than VGG16-BN and VGG16-BN-M for I_d and I_n, while it is completely opposite for I_t in the 4th column. This can be explained by the fact that VGG16-BN is pre-trained on the natural and color images of ImageNet and it thus performs poorly in extracting depth and normal features. Oyedotun *et al.* [10] applied the pre-trained VGG19 to extract 3D depth features, and the accuracy is only 28.06%, which also proves the above observation. Therefore, our FFNet-M combines F3DNet-M with a higher performance on both I_d and I_n and VGG16-BN-M that performs better on I_t.

Evaluation of multimodal data. We evaluate the effectiveness of multimodal data. From the last column of Tables 3 and 4, we can find that the accuracy of combining all the three attribute maps as the input data is higher than that of using any single attribute map for the first four models. We also compare the performance of FFNet-M in 3D FER (I_d and I_n) and 2D+3D FER (I_d, I_n, and I_t), as described in Table 5. It can be seen that integrating the 2D texture information for 2D+3D FER can further improve the classification performance in comparison with 3D FER on both BU-3DFE and Bosphorus.

4. CONCLUSION

In this paper, we develop an efficient feature fusion network with masks (FFNet-M) for 2D+3D FER. In the preprocessing stage, we generate three aligned attribute maps from 3D scans and design two masks with weights distributed in the four salient regions. For feature extraction, we use two F3DNets-M with larger kernel sizes to extract 3D depth and normal features, and one VGG16-BN-M for extracting 2D texture features. Specifically, the two masks make our FFNet-M focus on 2D and 3D local features by enhancing the weights in the

salient regions. Experimental results show that the proposed FFNet-M outperforms state-of-the-art methods on BU-3DFE and achieves at least comparable accuracy on Bosphorus.

5. ACKNOWLEDGEMENT

This work was supported by the Innovation Funds for NELBITA from Anhui Government, China and the Scientific Research Funds from USTC. We acknowledge the support of GPU cluster built by MCC Lab of Information Science and Technology Institution, USTC.

6. REFERENCES

[1] S. Li and W. Deng, "Deep facial expression recognition: A survey," *IEEE Trans. Affect. Comput.*, pp. 1–20, 2020.

[2] P. Ekman and W.V. Friesen, "Constants across cultures in the face and emotion," *J. Pers. Soc. Psychol.*, vol. 17, no. 2, pp. 124–129, 1971.

[3] M. Jian *et al.*, "Multi-view face hallucination using SVD and a mapping model," *J. Inf. Sci.*, vol. 488, pp. 181–189, 2019.

[4] H. Li, J. Sun, Z. Xu, and L. Chen, "Multimodal 2D+3D facial expression recognition with deep fusion convolutional neural network," *IEEE Trans. Multimedia*, vol. 19, no. 12, pp. 2816–2831, 2017.

[5] I. Mpiperis, S. Malassiotis, and M. G. Strintzis, "Bilinear models for 3-D face and facial expression recognition," *IEEE Trans. Inf. Forensics Security*, vol. 3, no. 3, pp. 498–511, 2008.

[6] B. Gong, Y. Wang, J. Liu, and X. Tang, "Automatic facial expression recognition on a single 3D face by exploring," in *Proc. 17th ACM Int. Conf. Multimedia*, 2009, pp. 569–572.

[7] H. Soyel and H. Demirl, "Facial expression recognition using 3D facial feature distances," in *Image Analysis and Recognition, (Lect. Notes Comput. Sci. 4633)*, 2007, pp. 831–838.

[8] H. Tang and T. S. Huang, "3D facial expression recognition based on properties of line segments connecting facial feature points," in *Proc. 8th IEEE Int. Conf. Automat. Face Gesture Recog.*, 2008, pp. 1–6.

[9] Q. Zhen, D. Huang, Y. Wang, and L. Chen, "Muscular movement model-based automatic 3D/4D facial expression recognition," *IEEE Trans. Multimedia*, vol. 18, no. 7, pp. 1438–1450, 2016.

[10] O. K. Oyedotun *et al.*, "Facial expression recognition via joint deep learning of RGB-depth map latent representations," in *Proc. IEEE Int. Conf. Comput. Vis. Workshops*, 2017, pp. 3161–3168.

[11] K. Tian *et al.*, "3D facial expression recognition using deep feature fusion CNN," in *Proc. Ir. Signals Syst. Conf.*, 2019, pp. 1–6.

[12] X. Wei, H. Li, J. Sun, and L. Chen, "Unsupervised domain adaptation with regularized optimal transport for multimodal 2D+3D facial expression recognition," in *Proc. 13th IEEE Int. Conf. Automat. Face Gesture Recog.*, 2018, pp. 31–37.

[13] K. Zhu *et al.*, "Discriminative attention-based convolutional neural network for 3D facial expression recognition," in *Proc. 14th IEEE Int. Conf. Automat. Face Gesture Recog.*, 2019, pp. 1–8.

[14] Y. Jiao *et al.*, "Facial attention based convolutional neural network for 2D+3D facial expression recognition," in *IEEE Vis. Commun. Image Process.*, 2019, pp. 1–4.

[15] S. Lin *et al.*, "Orthogonalization-guided feature fusion network for multimodal 2D+3D facial expression recognition," *IEEE Trans. on Multimedia*, pp. 1–11, 2020.

[16] Z. Chen, D. Huang, Y. Wang, and L. Chen, "Fast and light manifold CNN based 3D facial expression recognition across pose variations," in *Proc. 26th ACM Int. Conf. Multimedia*, 2018, pp. 229–238.

[17] S. Zulqarnain Gilani and A. Mian, "Learning from millions of 3D scans for large-scale 3D face recognition," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2018, pp. 1896–1905.

[18] B. Hasani and M. H. Mahoor, "Facial expression recognition using enhanced deep 3D convolutional neural networks," in *Proc. IEEE Comput. Soc. Conf. Comput. Vis. Pattern Recogn. Workshops*, 2017, pp. 2278–2288.

[19] G. Mu *et al.*, "Led3D: A lightweight and efficient deep approach to recognizing low-quality 3D faces," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2019, pp. 5766–5775.

[20] "Dlib c++ library," `http://dlib.net/`.

[21] S. Ioffe and C. Szegedy, "Batch normalization: Accelerating deep network training by reducing internal covariate shift," in *Proc. Int. Conf. Mach. Learn.*, 2015.

[22] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," in *Proc. Int. Conf. Learn. Represent.*, 2015.

[23] L. Yin *et al.*, "A 3D facial expression database for facial behavior research," in *Proc. 7th IEEE Int. Conf. Automat. Face Gesture Recog.*, 2006, pp. 211–216.

[24] A. Savran *et al.*, "Bosphorus database for 3D face analysis," in *Proc. European Workshop on Biometrics and Identity Management*, 2008, pp. 47–56.

[25] X. Yang, D. Huang, Y. Wang, and L. Chen, "Automatic 3D facial expression recognition using geometric scattering representation," in *Proc. 11th IEEE Int. Conf. Automat. Face Gesture Recog.*, 2015, vol. 1, pp. 1–6.

[26] H. Li *et al.*, "An efficient multimodal 2D + 3D feature-based approach to automatic facial expression recognition," *Comput. Vis. Image Understand.*, vol. 140, pp. 83–92, 2015.