# Occlusion-Aware Facial Expression Recognition Based Region Re-weight Network

Xinghai Zhang(✉), Xingming Zhang, Jinzhao Zhou, and Yubei Lin

South China University of Technology, Guangzhou, China
cszhangxinghai@mail.scut.edu.cn

**Abstract.** Occlusion is a major obstacle for facial expression recognition (FER) in the wild, which can change facial appearance significantly. Current FER methods, although having achieved much progress in lab-constrained scenarios, suffers from partial occlusion remarkably. In this paper, we propose a novel Region Re-Weight Network (RRWN), to adaptively capture and emphasize the non-occluded areas of the face. RRWN contains two modules: Occlusion-Aware Module (OAM) and Block-Loss Module (BLM). More specifically, OAM works as an adaptive region selector in a convolutional neural network. It selects areas whose features made the best approximation to that of the whole face based on their feature similarity. BLM contains a region biased loss called Block-Loss to emphasize the role of key blocks. We validate our RRWN in four public expression datasets with occlusions: RAF-DB, FERPlus, AffectNet, and SFEW. Experiments show that our RRWN largely improves the performance of FER with occlusion.

**Keywords:** Facial expression recognition · Occlusion · Sparse representation

## 1 Introduction

Facial expression recognition (FER) has been a popular research field for its potential applications in human-computer interaction, driver fatigue monitoring, mental health assessment, and other fields. Despite the high accuracy achieved under a standard environment, spatial occlusion has been the standing challenge to achieving robustness. Occlusions in real-life scenarios encompass a massive number of daily objects and occupy different positions of face images, which greatly affect the robustness of FER algorithms.

Earlier researchers mainly study the influence of occlusion positions on FER. Boucher *et al.* [4] occluded key areas of the face to learn which areas are the most important in human perception. Kotsia *et al.* [15] concluded that mouth occlusion causes a greater decrease in FER than the equivalent eyes occlusion. Then methods based on sparse representation are proposed. Cotter [7] presented the weighted voting method based on sparse representation classifier (SRC) for


D. N. Pham et al. (Eds.): PRICAI 2021, LNAI 13033, pp. 209–222, 2021.
https://doi.org/10.1007/978-3-030-89370-5_16

FER. Zhang *et al.* [31] extracted three typical facial features to evaluate the performance of the SRC method. Subsequently, with the emergence of large-scale datasets and robust novel network architectures, researchers carried out a combination of deep learning and sparse representation. Huang *et al.* [14] exploited the sparse representation and residual statistics to occlusion detection of video sequences. Zhong *et al.* [33] proposed a two-stage multi-task sparse learning framework to find dominant patches and learn specific facial patches for individual expression. Recently attention-based methods are proposed to address occlusions in FER [19,20,27], determined whether the facial block should be emphasized or not based on the importance score.

We are motivated to come up with a new mechanism to provide neural networks with the knowledge of occlusion for recognizing expressions with partial occlusion. When observing face images with occlusions, people will focus on the non-occluded areas and recognize expression based on the information of these non-occluded areas. Inspired by this, we propose a novel Region Re-Weight Network (RRWN) to capture and emphasize the non-occluded areas of the face. RRWN is mainly composed of two modules, Occlusion-Aware Module (OAM) and Block-Loss Module (BLM). OAM learns to pick out the non-occluded facial regions to facilitate recognition, which is compatible with the mainstream convolutional neural network (CNN) architecture. As depicted in Fig. 1, OAM works with a widely-used convolutional architecture, in which the feature maps of the holistic image are decomposed as the combination of feature maps from its local regions. Different from the most widely-used attention-based methods, OAM employs similarity measurements to capture the difference between facial and non-facial areas. After getting the non-occluded regions through OAM, the non-occluded regions will be highlighted in the latter network. In the meantime, we use the Block-Loss to emphasize the role of the key area among these non-occluded regions. Different from other occlusion-aware methods, our method guides the model to separate occlusions from the human face.

The major contributions of this work can be summarized in three aspects: 1) We propose OAM, a novel network structure to avoid facial blocks with occlusion and select non-occluded blocks. 2) A region biased loss (Block-Loss) is proposed to optimize the selection of crucial regions. 3) On four challenging datasets with occlusions, we demonstrate that our methods achieve superior performance.

## 2 Related Work

### 2.1 FER Methods Against Occlusions

Many FER methods consider using prior knowledge to strike a better performance both in lab-constraint and in-the-wild scenarios. Common options to incorporate such knowledge includes manually design refined segmentation based on detected facial landmarks since it is effective to constraint the model's input to only the regions where expression-related actions occur. According to the facial action coding system [10], action units are situated around the eyes, the forehead, and the mouth. Extracting those key areas accordingly reduces noise

from hair, sunglasses, masks, and other occlusions. However, it works only if these key areas are not occluded.

When the location of occlusions is uncertain, dividing the whole facial image into smaller patches while applying some selection or weighting method over the patches is often more robust than the key-area segmentation approaches. Face partitioning methods varies from uniform partitioning [14], landmark-centered partitioning [19], to sampling-oriented [27]. Subsequently, the occluded patches are given smaller importance weights, or simply excluded from the recognition process.

Recent works following this principle prefer to generate an importance score for each block according to its contribution to the classification. For example, Li *et al.* [19] proposed to use a convolution neural network with attention mechanism to compute an adaptive weight from the region itself according to the unobstructedness and importance. Wang *et al.* [27] proposed a novel region attention network using the sigmoid value to represent the attention value and combining the overall and part features to enhance the ability of the network.

The above methods obtain the importance score through a designed deep neural network, and it is considered that the blocks with large importance scores should be focused on by the network. But in fact, the blocks with large importance scores are possible to be the occluded blocks. Different from these works, our method determines whether the block is occluded by the similarity between the facial block and the whole image, rather than simply using the important score. When the face image is partially obscured, its overall characteristics are still close to a face, so the blocks which are close to the face image are non-occluded blocks.

### 2.2 Sparse Representation

Inspired by the success of sparse approximation in the face recognition task [29], researchers proposed adaptations and variations of sparse encoding to the expression recognition task. Methods concerning sparse representation decompose a facial image as a linear combination of images from the same expression category. During the process, four typical facial features, *i.e.*, the raw pixels [31], Gabor wavelets representation [6], local binary patterns [2], and deep features extracted by a deep convolutional network [1] are used as the effective representations for the expression images.

However, the above methods suffer drastically from insufficient training sample size and variations included. To effectively represent an unseen image containing an occluded facial expression, they also require assistance from well-performing decorrelation technique, precise face alignment, and normalization which is far from reaching in many in-the-wild datasets to date. Although we also decompose the whole facial image as a linear combination, our method distinguishes itself from existing sparse representation methods since we measure how much content in each patch is related to the whole image.

# 3 Proposed Method

## 3.1 Overview of Region Re-weight Network

As depicted in Fig. 1, RRWN extends the traditional CNN architecture by the additional OAM and BLM. To begin with, the face image is fed into the first layer of the backbone network to obtain feature maps for the whole face image as well as each local block. Next, OAM selects the non-occluded blocks by measuring the similarity between local and global vectors. Finally, the non-occluded blocks will be highlighted in the latter CNN layers. In addition to OAM, we also introduce BLM which contains a loss function to emphasize the role of critical block, which comes from non-occluded blocks chosen by OAM. As a result, The whole RRWN can be trained in an end-to-end manner.

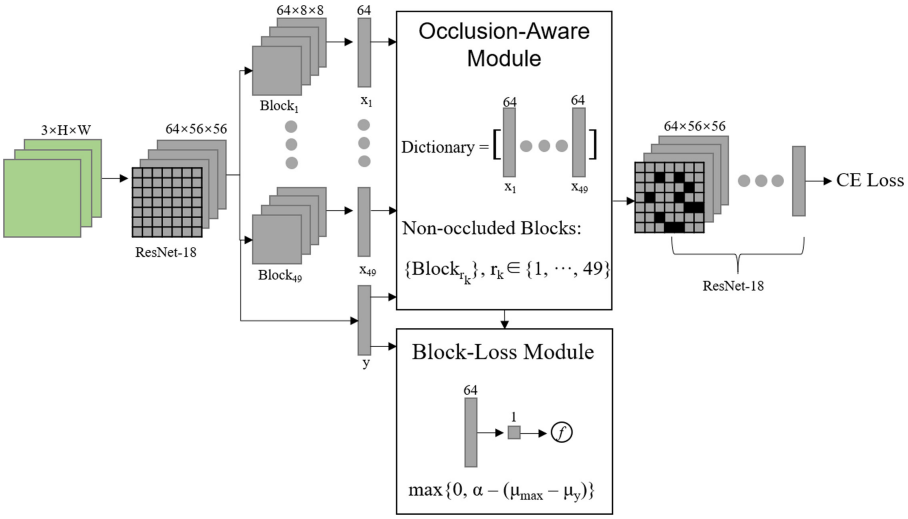



**Fig. 1.** The framework of our RRWN. A face image is fed into Resnet-18 and is represented as the global vector $y$ and local vectors $x_i$. The Occlusion-Aware Module takes $y$ and $x_i$ as input to find the non-occluded $\{Block_{r_k}\}$. Then the $\{Block_{r_k}\}$ will be re-weighted in the latter network (the corresponding black squares). The Block-Loss Modules emphasizes the role of key block among $\{Block_{r_k}\}$ through the Block-Loss function.

## 3.2 Occlusion-Aware Module

We hold the presumption that the overall characteristics of the face image are close to its components rather than the occlusions. In our case, the similarity is used as a mathematical measure to find the clear facial areas similar to the overall face image. In other words, the non-occluded blocks of the face image are

located by the similarity measurement. Inspired by how the orthogonal matching pursuit (OMP) method finds the most similar component of a signal [24], we design OAM to find the non-occluded blocks.

As shown in Fig. 1, after getting feature maps that represent the whole facial image, we partition the feature maps to multiple sub-feature-maps uniformly to obtain diverse blocks of the same size. Next, an adaptive average pooling operation is utilized to encode the feature maps into a vector, i.e., each three-dimensional feature is mapped to a one-dimensional vector. Let $y$ denotes the global vector. We normalize $y$ for convenient calculation so that we have $||y|| = 1$. Similarly, $\chi = \{x_1, x_2, \cdots, x_n\}$ denotes local vectors and $||x_i|| = 1$. According to conventional sparse approximation methods, a dictionary is often created to store atomic vectors before finding the sparse representation of the global vector. In our method, the local vectors are used as the atomic vectors when building the dictionary $D = [x_1, x_2, \cdots, x_n] \in R^{n \times k}$, where $n$ is the number of atomic vectors and $k$ is the dimensionality of the atomic vectors.

After building the dictionary, the inner product of the global vector $y$ and each atomic vector $x_i$ is calculated. Then, the atomic vector with the largest absolute value of the inner product will be selected as the closest match-up to the $y$. This selection iterates until we obtain the maximum number of atomic vectors. In this way, $y$ is decomposed into the vertical projection in the direction of the chosen atomic vectors and the corresponding residual, which can be formulated as,

$$y = \langle y, x_{r_0} \rangle x_{r_0} + R_1, \tag{1}$$

where $\langle ., . \rangle$ is the inner product, $x_{r_0}$ is the closest match atomic vector, $r_0$ is the column index of $D$, $\langle y, x_{r_0} \rangle x_{r_0}$ is the vertical projection in the direction of $x_{r_0}$, and $R_1$ is the residual. Then we decompose the residual $R_1$ in the same way. After k iterations, we can get

$$y = \sum_{k=0}^{K} \langle R_k, x_{r_k} \rangle x_{r_k} + R_{k+1}, \tag{2}$$

where $K$ is a hyper-parameter served as the number of selected atomic vectors, and $R_0 = y$. If $K$ is too small, only a few non-occluded areas can be found. On the other hand, if $K$ is too large, the non-occluded area may also be selected. After several iterations, the linear representation of the target vector can be obtained, which is formulated as follows:

$$\begin{aligned} y &= \sum_{k=0}^{K} c_k x_{r_k} \\ c_k &= \langle R_k, x_{r_k} \rangle \end{aligned} \tag{3}$$

Now that the non-occluded blocks and the corresponding weight are obtained, then we apply a re-weight operation on the original feature maps. The blocks

corresponding to the selected atomic vectors are weighted as Eq. 4 while the unselected blocks remain unchanged, which can be defined as,

$$block_{r_k} = (c_k + c)block_{r_k}, \tag{4}$$

where $block_{r_k}$ denotes the $k^{th}$ selected block. The $c_k$ can be arbitrary in $(0, 1)$. To strengthen the role of the non-occluded area, we increase the weight by $c$ times. If we overemphasize the key blocks and impose great weight on them, it will lead to a decrease of accuracy. We will analyze this in the later ablation studies. After OAM, the new feature maps continue to be input to the rest of ResNet-18.

OAM optimizes the latter network during the training by performing the weighting operation to the original feature maps. OAM can select the atom vector that is closest to the target vector. The weights describe how similar the atom vector is to the target vector. Even if the face is partially occluded, the face is still the dominant object in the image. In this way, OAM can select the non-occluded areas. However, when the occlusion is too large and occupies most of the face image, the overall feature of the image tends to be the occlusion rather than the face, OAM will perform poorly.

### 3.3 Block-Loss Module

After OAM, we find the non-occluded blocks. Among the non-occluded blocks, some blocks contribute to recognizing the expression more significantly than others [4]. To encourage high weights for the most important block among these non-occluded blocks. Inspired by [27], we propose the Block-Loss.

As can be seen in Fig. 1, BLM contains a fully-connected layer and a sigmoid function. After getting the global vector $y$ and the non-occluded local vectors $x_{r_k}$ chosen by OAM, they are fed to BLM. After the fully-connected layer and the sigmoid function, we get their importance value. Block-Loss can be formulated as,

$$\begin{aligned} \mathcal{L}_B &= \max\{0, \alpha - (\mu_{max} - \mu_y)\}, \\ \mu_{max} &= \max\{f(x_{r_k} q)\}, \\ \mu_y &= f(yq), \end{aligned} \tag{5}$$

where $\alpha$ is a hyper-parameter served as a margin, $q$ is the parameter of the fully-connected layer, and $f$ denotes the sigmoid function. In the training process, the Cross-Entropy Loss is jointly optimized with the Block-Loss, which can be defined as,

$$\mathcal{L}_{All} = \mathcal{L}_{CE} + \mathcal{L}_B, \tag{6}$$

where $\mathcal{L}_{CE}$ denotes the Cross-Entropy Loss.

BLM optimizes the former network during the training by the loss function. BLM enforces that one of the important values of non-occluded blocks should be larger than the face image with a margin so that RRWN can focus on the most important block among the non-occluded blocks.

# 4 Experiments

## 4.1 Datasets

**RAF-DB** [17] contains 30,000 facial images annotated with basic or compound expressions by 40 trained human coders. In our experiment, only images with basic emotions(neutral, happiness, surprise, sadness, anger, disgust, fear) are used, including 12,271 images as training data and 3,068 images as test data. **FERPlus** [3] contains 28,709 training images, 3,589 validation images, and 3,589 test images collected by the Google search engine, and all images are resized to $48 \times 48$ pixels. FERPlus supplements a contempt emotion and is annotated by 10 labels. **AffectNet** [23] is the largest FER dataset that contains more than one million images collected by three search engines using expression-related keywords. About 400,000 images are manually annotated with eight discrete facial expressions as FERPlus. It has imbalanced training and test sets as well as a balanced validation set. **SFEW** [8] contains 95 subjects and covers unconstrained facial expressions, a large range of ages, varied head poses, and real-word illumination. We use the newest version of SFEW [9] which has been divided into three sets: training (958 images), validation (436 images), and test (372 images), and all images are annotated with seven discrete facial expressions as RAF-DB.

**Table 1.** Values of hyper-parameters

| Parameter | Value |
|---|---|
| Number of blocks | 49 |
| Number of selected atomic vectors | 10 |
| Weight increment $c$ | 2 |
| Margin $\alpha$ | 0.01 |
| Ratio of the two loss functions | 1:1 |

## 4.2 Implementation Details

The proposed RRWN is implemented on the environment of Python 3.6 and the operating system of Windows 10. Preprocessing methods like image resizing are executed through OpenCV 3.4 for convenience. The proposed network involved in this work is run on Intel(R) Core(TM) i7-6700 3.4 GHz in CPU and NVIDIA RTX 1080 Ti with CUDA 9.0 in GPU. RRWN is implemented using the Pytorch platform and the backbone network is ResNet-18 [12]. By default, the ResNet-18 is pre-trained on MS-Celeb-1M face recognition dataset and we extract the feature maps after the first layer of ResNet-18.

Each face image is first resized to $224 \times 224$. Then the feature maps are partitioned into $7 \times 7$ blocks uniformly as depicted in Fig. 1. After adaptive

average pooling operation, the feature maps are encoded as vectors of 64 dimensions. The number of selected atomic vectors is 10. The margin in Block-Loss is default as 0.01 and the whole network is jointly optimized with Block-Loss and Cross-Entropy Loss in training. The ratio of the two loss functions is empirically set at 1 : 1. Values of hyper-parameters are shown in Table 1. The batch-based stochastic gradient descent optimizer is used to train the model. On all datasets, the batch size is set to 64, the base learning rate was set as 0.01 and was reduced by the polynomial policy with the gamma of 0.1. Finally, the momentum was set as 0.9 and the weight decay was set as 0.0001.

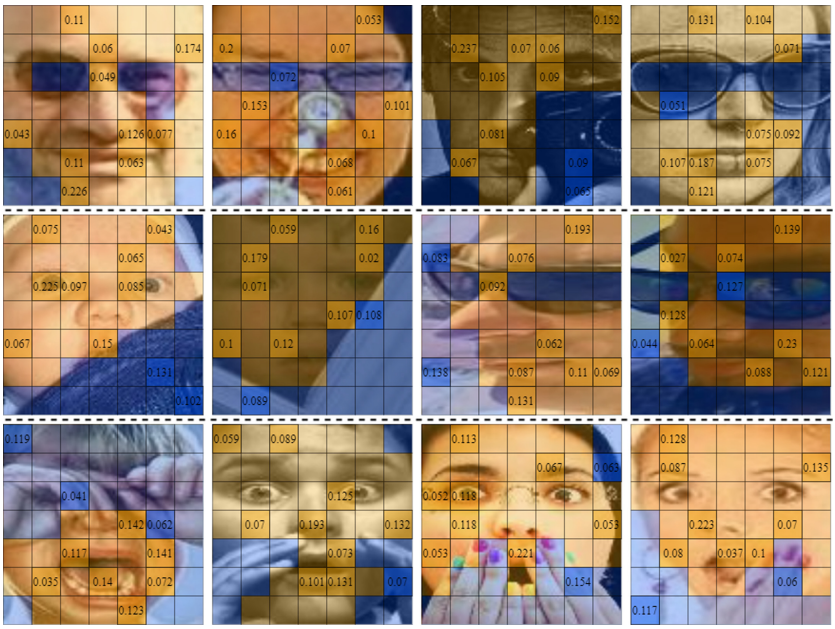


**Fig. 2.** Images with occlusions from RAF-DB. Each image is equally divided into 49 blocks. The orange squares represent the facial non-occluded areas, and the blue squares represent the occluded areas. Dark orange and dark blue squares represent the blocks selected by OAM. The number in the square is the coefficient of the linear combination obtained by OAM. (Color figure online)

### 4.3 Visualization of the Blocks Selected by OAM

OAM should be able to match the non-occluded areas of the face. To demonstrate the effect of OAM, non-occluded blocks selected by OAM are shown in Fig. 2. The occluded areas are covered by blue masks while the clear face areas are covered by orange masks. Areas selected by OAM are further highlighted with a darker color and the corresponding weights. It is clear that most of the selected blocks

which OAM selects are non-occluded blocks. In addition, some non-occluded blocks play an important role in FER because they include key areas such as eyes, mouth, etc.

For the images in the first row, where the occlusion and the face have many differences, OAM can find the key blocks closest to the whole face, making it effective to avoid the blocks with occlusions. In the next row, where the occlusions occupy a relatively larger area of the face image, blocks containing occlusions will be selected because features of the face image in this situation include quite a lot of information of the occlusions. Down to the last row, if the occluded object is a hand, in which the color, texture, and other features are relatively similar to the face, OAM will be possible to select few blocks containing hands.

**Table 2.** Test accuracy(%) on real-world datasets.

| Pretrain | RRWN | RAF-DB | FERPlus | AffectNet |
|---|---|---|---|---|
| ✗ | ✗ | 72.00 | 82.40 | 46.58 |
| ✗ | ✓ | 76.83 | 82.68 | 48.63 |
| ✓ | ✗ | 84.20 | 86.80 | 58.50 |
| ✓ | ✓ | **85.82** | **87.70** | **58.70** |

### 4.4 Ablation Studies Evaluation

**Effectiveness of RRWN:** To evaluate the effectiveness of RRWN compared with the baseline (ResNet-18), we conduct experiments on real-world datasets. Results are shown in Table 2. When training from scratch, our proposed RRWN outperforms the baseline network by a margin of 4.83%, 0.28%, and 2.05% on RAF-DB, FERPlus, and AffectNet respectively. It shows that our method does improve the accuracy of the baseline. In addition, when using ResNet-18 pre-trained on MS-Celeb-1M, our method obtains improvements of 1.62%, 0.9%, 0.2% on these datasets.

**Table 3.** Test accuracy(%) of the two modules on RAF-DB.

| OAM | BLM | ResNet-18 | ResNet-18 (pretrain) |
|---|---|---|---|
| ✗ | ✗ | 72.00 | 84.20 |
| ✗ | ✓ | 73.16 | 84.60 |
| ✓ | ✗ | 75.68 | 85.50 |
| ✓ | ✓ | **76.83** | **85.82** |

Furthermore, to explore the effectiveness of the two modules in improving accuracy, we conduct comparative experiments on RAF-DB. The result is shown

in Table 3. By the way, when only BLM is added, the input vectors of BLM are directly from the vectors after the adaptive average pooling operation. When only adding OAM or BLM, we obtain improvements of 3.68% and 1.16% based on ResNet-18, 1.3% and 0.4% based on ResNet-18 (pretrain). This suggests that both OAM and BLM contribute to improving accuracy. In addition, OAM is the most contributed module for our RRWN.

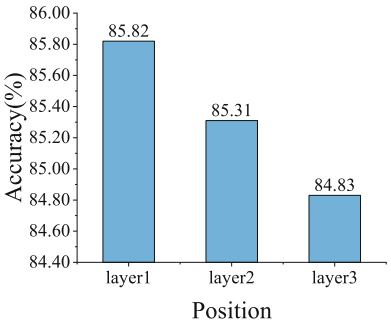


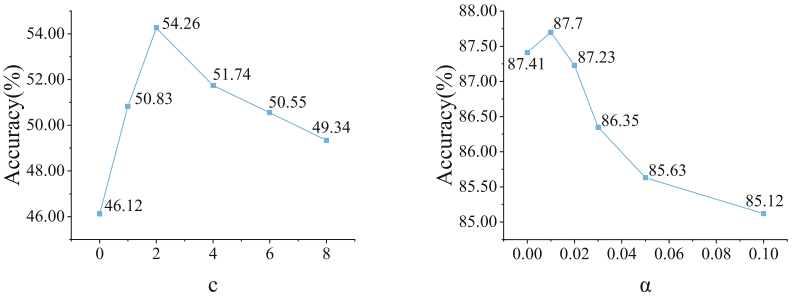

**Fig. 3.** Evaluation of the position on RAF-DB

**Position of OAM:** We study the impact of the different OAM positions. Since the backbone network is ResNet-18, which can be divided into four layers (we represent them as layer1, layer2, layer3, and layer4). Experiments are carried out with OAM being placed after the first, second, and third layers. Result on RAF-DB is shown in Fig. 3. The test result indicates that OAM works best when is placed after the first layer. And the further back it is placed, the worse the effect will be.

We analyze this phenomenon and concluded two major reasons for the declination. First, OAM represents the target vector linearly with a certain number of atomic vectors, so the greater the difference between the blocks, the more accurate OAM is to find the non-occluded blocks. Second, as CNN deepens and constantly carries out convolution, pooling, and other operations, the obtained feature maps become smaller, and the features become more abstract. The features of different blocks are mixed, which leads it more difficult to distinguish different blocks. Therefore, adding OAM after the first layer is appropriate.

**Evaluation of the Weight Increment** $c$: In OAM, we obtain the atomic vectors corresponding to the non-occluded blocks. The blocks corresponding to the selected atomic vectors are re-weighted, and the blocks that are not selected remain unchanged. We study the effect of the amount of weight increase, and the result is shown in Fig. 4(a).

As can be seen from Fig. 4(a), when we just multiply the coefficient $c_k$ to the non-occluded block, *i.e.*, $c = 0$, the result is poor because the coefficient $c_k$ is between 0 and 1, and the non-occluded blocks are weakened when we multiply them directly. On the other hand, the accuracy declined as $c$ increased. Because




(a) Evaluation of $c$ on SFEW (b) Evaluation of $\alpha$ on FERPlus

**Fig. 4.** Parameters evaluation

FER not only focuses on the partial key blocks but also the global features. We should combine local features and global features. As the article [19,27], the combination of global features and local features is more effective. If we focus too much on local features and ignore the global features, the weight increment $c$ is too large and the accuracy will decline.

**Evaluation of the Margin** $\alpha$: From Table 3, we can see that BLM further improves performance on RAF-DB.

The margin $\alpha$ in Block-Loss is set to 0.01 by default. We evaluate the $\alpha$ in FERPlus, the result is shown in Fig. 4(b). Increasing from 0 to 0.01 gradually improves the performance while larger $\alpha$ leads to fast degradation, which indicates the features of the overall face image are also important for FER. It also further confirms that we need to combine local features and global features. We mainly carry out the combination of local features and global features in two aspects. One is to input the global vector into BLM, and the other is to appropriately emphasize the key blocks selected by OAM.

### 4.5 Results and Comparison

We compare our RRWN to several methods on RAF-DB, FERPlus, AffectNet, and SFEW including attention-based methods [19,20,27] and loss-function methods [5,18,21]. The result is shown in Table 4.

pACNN [20] re-weights each patch according to the attention mechanism. gACNN [19] leverages a patch-based attention network and a global network. RAN-ResNet18 [27] captures the importance of facial regions and aggregates region features into a compact representation. These attention-based methods are time-consuming due to the carefully designed deep neural networks. Our RRWN does not increase much computational expense by simply adding two modules to the existing network architecture. DLP-CNN [18] uses a locality-preserving loss for network training. Island Loss [5] proposes the island loss which combines the Center Loss [28] and an inter-class loss. IACNN [21] proposes an identity-sensitive contrastive loss to achieve identity-invariant FER.

These loss-function methods do not emphasize the key block of the face image, whereas our RRWN emphasizes the key block in the non-occluded blocks. Our RRWN outperforms these recent methods with 85.80%, 87.70%, 58.70%, 54.26% on RAF-DB, FERPlus, AffectNet, and SFEW.

**Table 4.** Comparison on datasets with occlusions

| Datasets | Methods | Accuracy(%) |
|---|---|---|
| RAF-DB | FSN [32] | 81.10 |
| | pACNN [20] | 83.27 |
| | DLP-CNN [18] | 84.13 |
| | ALT [11] | 84.50 |
| | gACNN [19] | 85.07 |
| | Our RRWN | **85.82** |
| FERPlus | TFE-JL [16] | 84.30 |
| | PLD [3] | 85.10 |
| | SHCNN [22] | 86.54 |
| | ESR-9 [25] | 87.15 |
| | DTAGN [13] | 87.40 |
| | Our RRWN | **87.70** |
| AffectNet | Up-Sampling [23] | 47.00 |
| | pACNN [20] | 55.33 |
| | IPA2LT [30] | 55.71 |
| | IPFR [26] | 57.40 |
| | Weighted-Loss [23] | 58.00 |
| | Our RRWN | **58.70** |
| SFEW | IACNN [21] | 50.98 |
| | Island Loss [5] | 52.52 |
| | RAN-ResNet18 [27] | 54.19 |
| | Our RRWN | **54.26** |

## 5 Conclusion

In this work, we propose RRWN to address facial expression recognition in the presence of occlusions. Our RRWN uses the Occlusion-Aware module (OAM) to adaptively capture and emphasize the uncovered area of the face. In addition, we design a region biased loss (Block-Loss) function to encourage high weight for the most important region. We evaluate our method on real-world datasets. Experiments show that our proposed method has substantial improvement on RAF-DB, FERPlus, AffectNet, and SFEW compared with other methods.

## References


1. Abavisani, M., Patel, V.M.: Deep sparse representation-based classification. IEEE Sig. Proces. Lett. **26**(6), 948–952 (2019)
2. Ashir, A.M., Eleyan, A.: Facial expression recognition based on image pyramid and single-branch decision tree. Sig. Image Video Process. **11**(6), 1017–1024 (2017). https://doi.org/10.1007/s11760-016-1052-9
3. Barsoum, E., Zhang, C., Ferrer, C.C., Zhang, Z.: Training deep networks for facial expression recognition with crowd-sourced label distribution. In: Proceedings of the 18th ACM International Conference on Multimodal Interaction, pp. 279–283 (2016)
4. Boucher, J.D., Ekman, P.: Facial areas and emotional information. J. Commun. **25**, 21–29 (1975)
5. Cai, J., Meng, Z., Khan, A.S., Li, Z., O'Reilly, J., Tong, Y.: Island loss for learning discriminative features in facial expression recognition. In: 2018 13th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2018), pp. 302–309. IEEE (2018)
6. Cotter, S.F.: Sparse representation for accurate classification of corrupted and occluded facial expressions. In: 2010 IEEE International Conference on Acoustics, Speech and Signal Processing, pp. 838–841. IEEE (2010)
7. Cotter, S.F.: Weighted voting of sparse representation classifiers for facial expression recognition. In: 2010 18th European Signal Processing Conference, pp. 1164–1168. IEEE (2010)
8. Dhall, A., Goecke, R., Lucey, S., Gedeon, T.: Static facial expression analysis in tough conditions: data, evaluation protocol and benchmark. In: 2011 IEEE International Conference on Computer Vision Workshops (ICCV Workshops), pp. 2106–2112. IEEE (2011)
9. Dhall, A., Ramana Murthy, O., Goecke, R., Joshi, J., Gedeon, T.: Video and image based emotion recognition challenges in the wild: Emotiw 2015. In: Proceedings of the 2015 ACM on International Conference on Multimodal Interaction, pp. 423–426 (2015)
10. Ekman, R.: What the Face Reveals: Basic and Applied Studies of Spontaneous Expression Using the Facial Action Coding System (FACS). Oxford University Press, Oxford (1997)
11. Florea, C., Florea, L., Badea, M.S., Vertan, C., Racoviteanu, A.: Annealed label transfer for face expression recognition. In: BMVC, p. 104 (2019)
12. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 770–778 (2016)
13. Huang, C.: Combining convolutional neural networks for emotion recognition. In: 2017 IEEE MIT Undergraduate Research Technology Conference (URTC), pp. 1–4. IEEE (2017)
14. Huang, X., Zhao, G., Zheng, W., Pietikäinen, M.: Towards a dynamic expression recognition system under facial occlusion. Patt. Recogn. Lett. **33**(16), 2181–2191 (2012)
15. Kotsia, I., Buciu, I., Pitas, I.: An analysis of facial expression recognition under partial facial image occlusion. Image Vis. Comput. **26**(7), 1052–1067 (2008)
16. Li, M., Xu, H., Huang, X., Song, Z., Liu, X., Li, X.: Facial expression recognition with identity and emotion joint learning. IEEE Trans. Affect. Comput. **12**, 544–550 (2018)

17. Li, S., Deng, W.: Reliable crowdsourcing and deep locality-preserving learning for unconstrained facial expression recognition. IEEE Trans. Image Process. **28**(1), 356–370 (2018)
18. Li, S., Deng, W., Du, J.: Reliable crowdsourcing and deep locality-preserving learning for expression recognition in the wild. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp. 2852–2861 (2017)
19. Li, Y., Zeng, J., Shan, S., Chen, X.: Occlusion aware facial expression recognition using CNN with attention mechanism. IEEE Trans. Image Proces. **28**(5), 2439–2450 (2018)
20. Li, Y., Zeng, J., Shan, S., Chen, X.: Patch-gated CNN for occlusion-aware facial expression recognition. In: 2018 24th International Conference on Pattern Recognition (ICPR), pp. 2209–2214. IEEE (2018)
21. Meng, Z., Liu, P., Cai, J., Han, S., Tong, Y.: Identity-aware convolutional neural network for facial expression recognition. In: 2017 12th IEEE International Conference on Automatic Face & Gesture Recognition (FG 2017), pp. 558–565. IEEE (2017)
22. Miao, S., Xu, H., Han, Z., Zhu, Y.: Recognizing facial expressions using a shallow convolutional neural network. IEEE Access **7**, 78000–78011 (2019)
23. Mollahosseini, A., Hasani, B., Mahoor, M.H.: Affectnet: a database for facial expression, valence, and arousal computing in the wild. IEEE Trans. Affect. Comput. **10**(1), 18–31 (2017)
24. Pati, Y.C., Rezaiifar, R., Krishnaprasad, P.S.: Orthogonal matching pursuit: recursive function approximation with applications to wavelet decomposition. In: Proceedings of 27th Asilomar Conference on Signals, Systems and Computers, pp. 40–44. IEEE (1993)
25. Siqueira, H., Magg, S., Wermter, S.: Efficient facial feature learning with wide ensemble-based convolutional neural networks. In: Proceedings of the AAAI Conference on Artificial Intelligence, vol. 34, pp. 5800–5809 (2020)
26. Wang, C., Wang, S., Liang, G.: Identity-and pose-robust facial expression recognition through adversarial feature learning. In: Proceedings of the 27th ACM International Conference on Multimedia, pp. 238–246 (2019)
27. Wang, K., Peng, X., Yang, J., Meng, D., Qiao, Y.: Region attention networks for pose and occlusion robust facial expression recognition. IEEE Trans. Image Process. **29**, 4057–4069 (2020)
28. Wen, Y., Zhang, K., Li, Z., Qiao, Y.: A Discriminative feature learning approach for deep face recognition. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) ECCV 2016. LNCS, vol. 9911, pp. 499–515. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-46478-7_31
29. Wright, J., Yang, A.Y., Ganesh, A., Sastry, S.S., Ma, Y.: Robust face recognition via sparse representation. IEEE Trans. Patt. Anal. Mach. Intell. **31**(2), 210–227 (2008)
30. Zeng, J., Shan, S., Chen, X.: Facial expression recognition with inconsistently annotated datasets. In: Proceedings of the European Conference on Computer Vision (ECCV), pp. 222–237 (2018)
31. Zhang, S., Zhao, X., Lei, B.: Robust facial expression recognition via compressive sensing. Sensors **12**(3), 3747–3761 (2012)
32. Zhao, S., Cai, H., Liu, H., Zhang, J., Chen, S.: Feature selection mechanism in CNNs for facial expression recognition. In: BMVC, p. 317 (2018)
33. Zhong, L., Liu, Q., Yang, P., Liu, B., Huang, J., Metaxas, D.N.: Learning active facial patches for expression analysis. In: 2012 IEEE Conference on Computer Vision and Pattern Recognition, pp. 2562–2569. IEEE (2012)