Occluded Thermal Face Recognition Using Bag of CNN ($BoCNN$)

Sumit Kumar, *Student Member, IEEE*, and Satish Kumar Singh, *Senior Member, IEEE*

***Abstract*—In this letter, we are proposing $BoCNN$ architecture framework for occluded thermal face recognition. We analyzed the performance of Pretrained models using transfer learning and they exhibit promising results for thermal faces without occlusion. The performance degrades with occlusion. We have used different decision level fusion strategies post transfer learning for the performance enhancement of the pretrained models. All the fusion strategies used in the proposed work give better results compared to any of the single CNN architecture.**

***Index Terms*—Mean score, maximum score, majority voting, mid wave infra red (MWIR), near infra red (NIR), graph laplace (GL).**

I. Introduction

FACE biometric has always been one of the most important research domain because of its vivid application in the field of authentication, security, and intelligence. There are lots of challenges in real-world surveillance such as illumination variations, poor lighting conditions, low resolution, and occlusions. A thermal face heat signature is sometimes more relevant [1]–[3] compared to visible images in all these challenging circumstances except occlusions. Different methods such as Thermal Minutia Points, local interest points, traditional handcrafted descriptors [4]–[7] and Ada boosting post segmentation-based fractal texture analysis [8] has been used in past to generate a unique template from thermal signature. CNN architectures and neural networks proposed in recent times [9], [10] also perform reasonably well in MWIR/NIR face recognition framework. For security and surveillance, partial occlusion is a challenging problem whether be it thermal or visible range images. A number of traditional methods such as Gabor feature based representation [11], [12], GL based methods [13] have earlier been proposed for occlusion detection considering the availability of ground truth samples in abundance. The work proposed by Li *et al.* [14] uses CNN with attention mechanism (ACNN) which combines multi-representations of ROI to detect occluded region. The work proposed by gao *et al.* [15] tries to generate the global structure of data representation and error images by exploiting the low-rankness of both of them. Dictionary-based learning proposed by Ou *et al.* [16] tries to distribute variable weights to each pixel based on reconstruction errors with an aim to assign minimal weights to regions affected by occlusion.

Manuscript received February 1, 2020; revised May 9, 2020; accepted May 9, 2020. Date of publication May 21, 2020; date of current version June 25, 2020. The associate editor coordinating the review of this manuscript and approving it for publication was Dr. Kai Liu. *(Corresponding author: Sumit Kumar.)*

Sumit Kumar is with the Indian Institute of Information Technology Allahabad, Allahabad 211015, India (e-mail: phc2014001@iiita.ac.in).

Satish Kumar Singh is with the Indian Institute of Information Technology Allahabad, Allahabad 211012, India (e-mail: sk.singh@iiita.ac.in).

Digital Object Identifier 10.1109/LSP.2020.2996429

TABLE I
Comparative Accuracy Measure

Architecture	Accuracy without occlusion			Accuracy with occlusion		
	P1	P2	P3	P1	P2	P3
Resnet-50	97.42	97.50	98.37	48.25	51.21	59.62
InceptionV3	98.06	97.81	97.88	45.02	49.09	57.41
Vgg-19	97.48	97.74	97.98	42.28	45.75	50.00
Inc.ResnetV2	95.67	96.96	96.54	36.56	44.84	50.06
Resnet-101	97.42	97.27	97.79	51.74	50.60	58.88

Again, a Hierarchical sparse and low-rank model which is a combination of sparse representation on dictionary learning and low-rank representation on the error [17] proposed in 2018 takes multi-order gradient direction domain component of the query and reconstruct it using the dictionary(images without occlusion). Though a lot of research has been done on occlusion removal, but is limited only to visible or NIR images with a prerequisite of ground truth in abundance. There is very little analysis on the performance of thermal face recognition once occlusion comes into play.

II. Methodology

A. Motivation and Major Contribution

Different CNN pre-trained models [18]–[20], [34] have earlier been proposed for face recognition. These models perform really well for visible facial images. We analyzed the performance of these pre-trained models over thermal images [7] (on CVBL-IIITA dataset) using transfer learning over the partitions ratio of $P1, P2, P3$ ie. 40–60, 50–50, 60–40 of the train and test dataset respectively(as shown in Table I). The performance of thermal face without occlusion is recognizable on all single CNN architectures used in this work. But once occlusion comes into play, the performance of these models degrade, which is quite evident from the Table I.

The reconstruction of the occluded part of thermal faces is very difficult due to very few neighboring features which tend to generate additional noise, thus making recognition more difficult. The other alternative is to maximize the exploitation of the few features present in the image. We have tried to improve the performance using multiple pre-trained CNN series architectures and Residual DAG architectures of different depths. The performance saturates, and there is very little or no enhancement using all these variations because of limited features and additional noise. The other way is the combination of features of these pre-trained models using different fusion strategies with a motive that low confidence classification data which is sometimes architecture-specific and whose probability of misclassification is high can be correctly classified using the

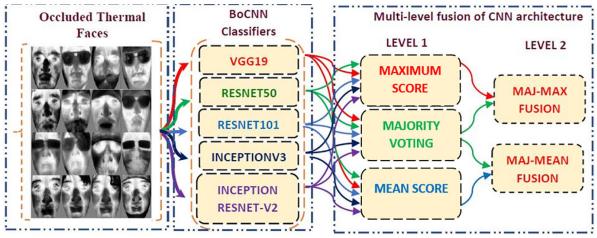

Fig. 1. Proposed $BoCNN$ model.

mutual information taken across all the networks. A number of different Fusion strategies at different levels namely data, feature and decision [21], [22] [23]–[26] have been employed efficiently for performance enhancement in the past.

The proposed $BoCNN$ model is the combination of multiple pre-trained DAG and serial CNN models, namely VGG-19, Resnet-50, Resnet-101, Inception-V3, and Inception-ResnetV2. The choice of these models is very much dependent on uniqueness in terms of the design and depth of the respective networks. VGG-19 model has been proposed with an intuition that more depth in the networks with small filter size leads to more discriminating features. The other proposed networks used are variants of residual networks that use skip connections with lesser depth, thus resolving the issue of diminishing gradients. The inception models used in work rely on the parallel generation of different features using filters of different sizes in parallel, generating a variance in features on specific layers. A combination of residual and inception models (Inception-ResNet-V2) has also been taken into consideration for fusion. Due to the variability in the architectural design and depth of each CNN model, the features generated are different. It has been analyzed empirically too that the misclassification of each network is not mutually inclusive. We have used these pre-trained models keeping the weights of the initial ten layers freeze so that generalized information can be captured, which tends to be very similar across the dataset. The other reason to use the weights of the initial layers is due to the size of the dataset (not more than seven images per class for training).

B. Fusion Strategies

The combination of different trained network post-training is done by different fusion strategies namely majority voting, maximum score, mean score at level 1 and the majority-mean, majority-max at 2 respectively. The majority-voting has been used for generating the most frequent class based on consensus. The maximum score fusion selects the class-label with the highest probabilistic score across all classifiers, whereas the mean score takes the wholesome belongingness of each sample for each class across different classifiers fusion as represented in Fig. 1. As majority voting sometimes suffers from ambiguity, we have used the fusion of mean and maximum score with majority voting post thresholding at level 2 to resolve the issue.

1) Majority-Voting: Different class matrices $CM_{T_{[1,2,3\ldots L]}}$ are generated for L classifiers based on the maximum score of each sample corresponding to a specific class, as shown in Fig. 2. Different class labels may be assigned to the same sample by different classifiers. The class label with the highest frequency corresponding to a sample is chosen as the true class for that specific data sample.

Suppose, we have m samples from S_1 to S_m to be classified among n different classes ranging from C_1 to C_n, using L classifiers ranging from T_1 to T_L then the probabilistic score of sample S_i for belongingness to class C_j by classifier T_k can be $P_{T_k}(i,j)$. The class matrix CM_{T_k} is represented as

$$CM_{T_k}(i) = C_{\text{label}}(R_{\max}(P_{T_k}(i,j))) \quad \forall i = 1 \text{ to } m \text{ and } j = 1 \text{ to } n \tag{1}$$

The RHS in the equation (1) gives the class label(C_{label}) of sample i based on the maximum probabilistic score across n different classes($R_{\max}$) for classifier T_k. The returned value is stored in class matrix $CM_{T_k}(i)$.

Applying majority-voting across L different classifiers we generate a class frequency matrix $M_f(i,j)$ based on how frequently the data sample i is associated with the class label j across L classifiers.

$$M_f(i,j) = Norm(Count(CM_{T_k}(i))) \tag{2}$$

The RHS in the equation (2) gives the normalized associativity of each sample to specific class post fusion of k different classifiers using majority-voting. After getting the matrix $M_f(i,j)$ from equation (2) the class label of i^{th} sample can be generated as in equation (3)

$$P_{Class}[i] = C_{\text{label}}(R_{\max}(M_f(i,j))) \tag{3}$$

The same can be realised from Fig. 2 where sample S_1, S_1 and S_2 are assigned classes C_1, C_2 and C_1 respectively based on the maximum frequency(row-wise).

2) Maximum Score: After getting the class-wise probabilistic score matrix of each sample using L different classifiers, the highest probabilistic score of the selected sample corresponding to each class is taken in consideration for final probabilistic class matrix generation as shown in Fig. 2.

Suppose, we have m samples from S_1 to S_m to be classified among n different classes ranging from C_1 to C_n, using L classifiers ranging from T_1 to T_L then the probabilistic score of sample S_i for belongingness to class C_j by classifier T_k can be $P_{T_k}(i,j)$. Applying Maximum Score fusion across L different classifiers

$$P_{\max}(i,j) = Max(P_{T_k}(i,j)) \forall\, i = 1 \text{ to } m,\ j = 1 \text{ to } n \tag{4}$$

The maximum score matrix obtained in equation(4) is normalized as

$$Norm_{P_{\max}}(i,j) = \frac{p_{\max}(i,j)}{\sum_{j=1}^{n} p_{\max}(i,j)} \forall\, i = 1 \text{ to } m \tag{5}$$

The normalized score matrix $Norm_{P_{\max}}(i,j)$ obtained in equation (5) can be treated as final probabilistic score matrix. After generating the final probabilistic matrix post fusion of all the classifier's scores, the class label can be generated based on the highest probabilistic score as

$$P_{Class}[i] = C_{\text{label}}(R_{\max}(Norm_{P_{\max}}(i,j)) \quad \forall\, i = 1 \text{ to } m \text{ and } j = 1 \text{ to } n \tag{6}$$

In the section "Max-score Fusion" of Fig. 2 it can be visualized that the row wise normalization operation is done over matrix $P_{\max}(i,j)$ according to equation (5) to generate the normalised matrix $Norm_{P_{\max}}(i,j)$ which ensures that summation of score in each row equals 1. The same matrix is used for generating the class label of ith data sample based on maximum probabilistic score value as shown in equation (6).

3) Mean Score: After getting the class-wise probabilistic score of each sample across k different classifiers, we take the mean of probabilistic scores. The final class corresponding to each sample is chosen based on the highest normalized score after row-wise normalization.

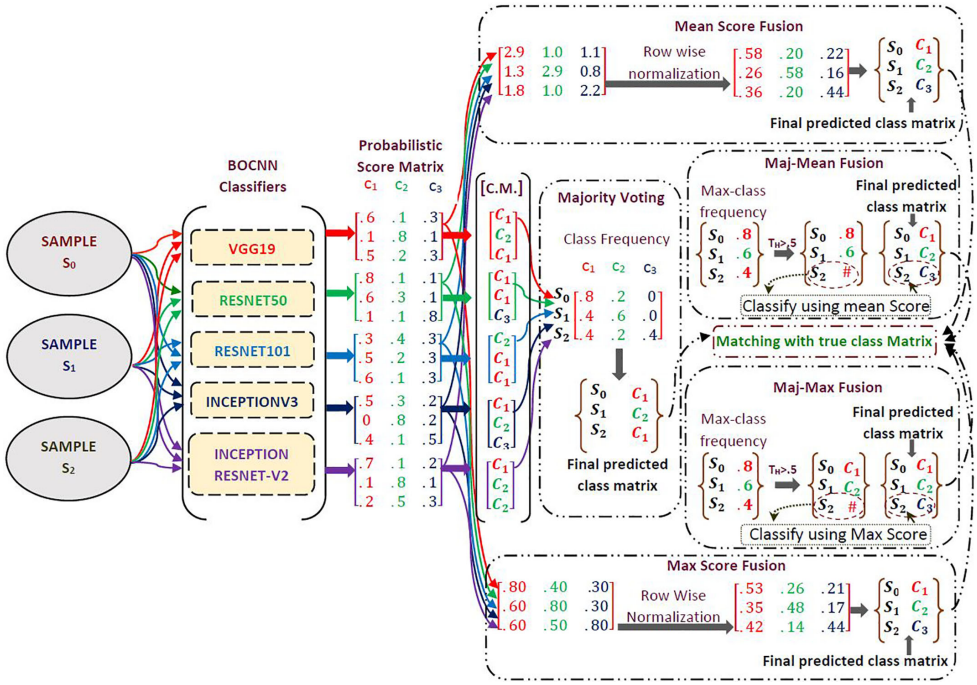

Fig. 2. An elaborate representation of complete fusion architecture.

Suppose, we have m samples from S_1 to S_m to be classified among n different classes ranging from C_1 to C_n, using L classifiers ranging from T_1 to T_L then the probabilistic score of sample S_i for belongingness to class C_j by classifier T_k can be $P_{T_k}(i,j)$.

Applying Mean Score fusion across L different classifiers

$$Norm_{P_{\text{Mean}}}(i,j) = \frac{\sum_{k=1}^{L} P_{T_k}(i,j)}{\sum_{j=1}^{n}\sum_{k=1}^{L} P_{T_k}(i,j)} \tag{7}$$

In the RHS of equation (7), the nominator part gives the summation of score of i_{th} sample for j_{th} class across L different classifiers which can be realised from the section "Mean score fusion" of Fig. 2. The score normalization is done by denominator of equation (7) i.e. diving by summed up score of i^{th} sample across all n classes taking L different classifiers in consideration.

4) Majority-Maximum Fusion: It is a combination of both the fusion strategies i.e. majority-voting and maximum score. We have set a threshold based on which the sample is classified either by majority-voting or maximum score fusion. Suppose $M_f(i,j)$ is the class frequency matrix as calculated in equation (2) and $Norm_{P_{\max}}(i,j)$ is the normalized max-score matrix as calculated in equation (5) then the predicted class of sample i is

$$P_{Class}[i] = \begin{cases} C_{\text{label}}(R_{\max}(M_f(i,j))) & \text{if} \rho \geq 0.5 \\ C_{\text{label}}(R_{\max}(Norm_{P_{\max}}(i,j))) & \text{otherwise} \end{cases} \tag{8}$$

According to equation (8) if the normalized class frequency of assigned class C_j to sample S_1 using majority-voting is greater than equal to 0.5 then that specific sample is assigned the same class, else the sample is classified using max-score fusion. The same can be realised from the section "majority-max Fusion" in Fig. 2 where the normalized class frequency of sample S_2 is less than 0.5 thus it is classified using Max score fusion whereas the remaining samples are classified using majority-voting.

5) Majority-Mean Fusion: It is a combination of both the fusion strategies i.e. majority-voting and mean score. We have set a threshold based on which the sample is classified either by majority-voting or mean score fusion. Suppose $M_f(i,j)$ is the class frequency matrix as calculated in equation (2) and $Norm_{P_{\text{Mean}}}$ is the normalized mean-score matrix as calculated in equation (7) then the predicted class of sample i is

$$P_{Class}[i] = \begin{cases} C_{\text{label}}(R_{\max}(M_f(i,j))) & \text{if} \rho \geq 0.5 \\ C_{\text{label}}(R_{\max}(Norm_{P_{\text{Mean}}}(i,j))) & \text{otherwise} \end{cases} \tag{9}$$

According to equation (9), if the normalized class frequency of assigned class C_j to sample S_1 using majority-voting is greater than equal to 0.5 then that specific sample is assigned the same class, else the sample is classified using mean-score fusion. The same can be realised from the section "Maj-Max Fusion" in Fig. 2 where the normalized class frequency of sample S_2 is less than 0.5 thus it is classified using Mean score fusion whereas the remaining samples are classified using majority-voting.

C. Complexity

The complexity post probabilistic score generation is $\Theta(n^2)$ of all fusion architecture(due to the access of class wise probabilistic score of each sample two times) compared to $\Theta(n)$ of single CNN models keeping the number of classes and classifiers constant. Average time using fusion architecture is 11.212 milliseconds per sample whereas normal CNN architecture takes 4.21 milliseconds post probabilistic score generation.

III. Experiments

A. Dataset

We have used IIIT Delhi occluded thermal face dataset [27], [28] for performance analysis. The dataset consists of 75 different subjects, each having approximately 7 to 10 samples with varied realistic occlusion. Multiple occlusion methods and smaller sample sizes per class result in very less intraclass common features, thus making the dataset very challenging.

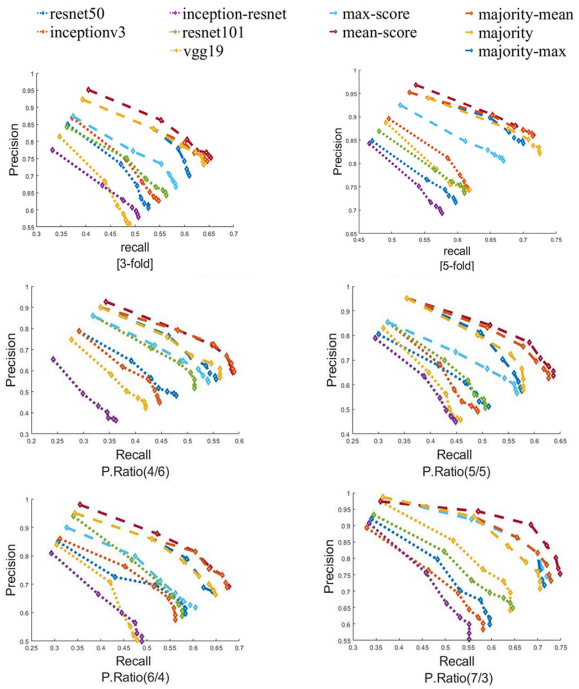

Fig. 3. Precision Vs Recall using non-exhaustive cross validation namely k-fold and hold out.

The other dataset used for comparing accuracy without occlusion(Table I) is the thermal dataset of CVB Lab IIITA [7] collected in KV-IIITA. The device used is the MWIR-Flir-E-40 series camera. It contains 125 classes of different subjects, each having 20 samples with variations in pose and expressions. The dataset is collected in the day time with ambient lighting conditions and a uniform background for all subjects.

B. Performance Analysis

Hold out (partition ratio of 80–20, 70–30, 60–40, 50–50 and 60–40 of the train and test data) and k-fold (with k = 3 and k = 5) non-exhaustive cross-validation methods have been used with 100 epochs having a mini-batch size of 10 for each case. The number of images taken into consideration is higher for initial ranks and decreases exponentially as the rank increases so that the misclassification of lower-ranked images (challenging images) can be analyzed clearly. Fig. 3 shows the APR Vs ARR using multiple CNN architectures being employed using transfer learning. From the figure, it can be visualized that fusion architecture(specifically mean score and majority mean) always creates a significant gap with respect to all the single CNN architectures. Fig. 4 shows the test accuracy of single and fused architectures over different partitions. Comparing the best among both the cases for the partition of 40–60, the majority-mean fusion shows a relative increment of 14% to resnet101. For the partition of 50–50, Mean-score shows a relative increment of 24.2% compared to Resnet50. For the partition of 60–40, Mean-score shows a relative increment of 16.1% compared to resnet50. For the partition of 70–30, Mean-score again shows a relative increment of 16.14% compared to Resnet-101. A very similar trend can be seen for the partition ratio 80–20 where Mean-score and Majority-mean fusion again give a relative increment of 15.05% compared to Resnet-101. A similar gap

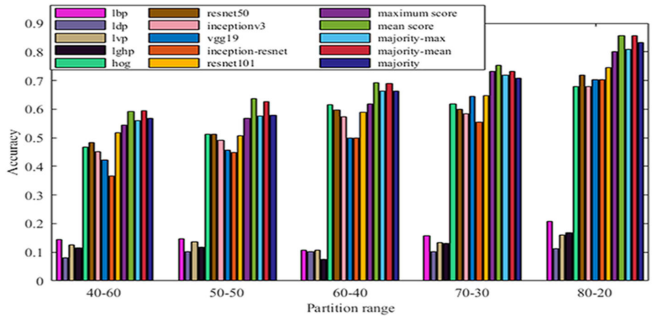

Fig. 4. Accuracy over different partitions.

in terms of accuracy can also be visualized with descriptors used for comparison namely LBP [29], LDP [30], LVP [31], LGHP [32] and HOG [33]. Among descriptors, HOG performs best, but fusion strategy always outperforms all descriptors used for comparison with a gap of 32%,23%,13%,22%, and 25% over partition ratio of 80–20, 70–30, 60–40, 50–50 and 60–40 respectively among best of both the cases as shown in Fig. 4. The performance comparison with 3-fold and 5-fold cross-validation is also shown using Average precision of retrieval Vs. Average recall rate. Here also similar trend can be observed and the same can be visualized from Fig. 3. Although the performance of all single CNN architecture is very much significant on the unoccluded thermal dataset, we still applied fusion strategy and found that the performance of all fusion strategies better than the single CNN models over each partition range. Considering the ratio of 50–50 the recognition accuracy is 99.1%,98.92%,98.87%, 99.2% and 98.64% respectively for the majority, majority-mean, majority-maximum, mean score and maximum score fusion strategies respectively.

IV. Conclusion

The fusion strategy is preferable over single CNN architecture when there are less discriminating features and a high occurrence of noisy information, as the same is proven empirically. In the case of higher occurrence of noise, cumulative features taken from multiple architectures always enhance the performance. Among all the fusion strategies, mean score and majority-mean perform unanimously better than remaining fusion methods. In spite of variations in train and test dataset size, the recognition accuracy after fusion is always better with a substantial margin than single CNN architectures employed using transfer learning.

References

[1] D. A. Socolinsky and A. Selinger, "A comparative analysis of face recognition performance with visible and thermal infrared imagery," in *Proc. IEEE Object Recognit. Supported User Interact. Service Robots*, 2002, pp. 217–222.

[2] D. A. Socolinsky, A. Selinger, and J. D. Neuheisel, "Face recognition with visible and thermal infrared imagery," *Comput. Vision Image Understanding*, vol. 91, nos. 1/2, pp. 72–114, 2003.

[3] D. A. Socolinsky and A. Selinger, "Thermal face recognition in an operational scenario," in *Proc. IEEE Comput. Soc. Conf. Comput. Vision Pattern Recognit.*, 2004, vol. 2, pp. II-1012–II-1019.

[4] P. Buddharaju, I. T. Pavlidis, P. Tsiamyrtzis, and M. Bazakos, "Physiology-based face recognition in the thermal infrared spectrum," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 29, no. 4, pp. 613–626, Apr. 2007.

[5] M. Akhloufi and A. Bendada, "Thermal faceprint: A new thermal face signature extraction for infrared face recognition," in *Proc. IEEE Can. Conf. Comput. Robot Vision*, 2008, pp. 269–272.

[6] G. Hermosilla, P. Loncomilla, and J. Ruiz-del Solar, "Thermal face recognition using local interest points and descriptors for HRI applications," in *Proc. Robot Soccer World Cup*, 2010, pp. 25–35.
[7] S. Kumar and S. K. Singh, "A comparative analysis on the performance of different handcrafted descriptors over thermal and low resolution visible image dataset," in *Proc. 5th IEEE Uttar Pradesh Section Int. Conf. Elect., Electron. Comput. Eng.*, 2018, pp. 1–6.
[8] A. Ibrahim, A. Tharwat, T. Gaber, and A. E. Hassanien, "Optimized superpixel and adaboost classifier for human thermal face recognition," *Signal, Image Video Process.*, vol. 12, no. 4, pp. 711–719, 2018.
[9] M. Peng, C. Wang, T. Chen, and G. Liu, "Nirfacenet: A convolutional neural network for near-infrared face identification," *Inform.*, vol. 7, no. 4, p. 61, 2016.
[10] Z. Wu, M. Peng, and T. Chen, "Thermal face recognition using convolutional neural network," in *Proc. IEEE Int. Conf. Optoelectron. Image Process.*, 2016, pp. 6–9.
[11] M. Yang and L. Zhang, "Gabor feature based sparse representation for face recognition with Gabor occlusion dictionary," in *Proc. Eur. Conf. Comput. Vision*, 2010, pp. 448–461.
[12] W. Zhang, S. Shan, X. Chen, and W. Gao, "Local gabor binary patterns based on Kullback–Leibler divergence for partially occluded face recognition," *IEEE Signal Process. Lett.*, vol. 14, no. 11, pp. 875–878, Nov. 2007.
[13] Y. Deng, Q. Dai, and Z. Zhang, "Graph laplace for occluded face completion and recognition," *IEEE Trans. Image Process.*, vol. 20, no. 8, pp. 2329–2338, Aug. 2011.
[14] Y. Li, J. Zeng, S. Shan, and X. Chen, "Occlusion aware facial expression recognition using CNN with attention mechanism," *IEEE Trans. Image Process.*, vol. 28, no. 5, pp. 2439–2450, May 2018.
[15] G. Gao, J. Yang, X.-Y. Jing, F. Shen, W. Yang, and D. Yue, "Learning robust and discriminative low-rank representations for face recognition with occlusion," *Pattern Recognit.*, vol. 66, pp. 129–143, 2017.
[16] W. Ou *et al.*, "Robust discriminative nonnegative dictionary learning for occluded face recognition," *Pattern Recognit. Lett.*, vol. 107, pp. 41–49, 2018.
[17] C. Y. Wu and J. J. Ding, "Occluded face recognition using low-rank regression with generalized gradient direction," *Pattern Recognit.*, vol. 80, pp. 256–268, 2018.
[18] K. Simonyan and A. Zisserman, "Very deep convolutional networks for large-scale image recognition," 2014, *arXiv:1409.1556*.
[19] K. He, X. Zhang, S. Ren, and J. Sun, "Deep residual learning for image recognition," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2016, pp. 770–778.
[20] C. Szegedy, S. Ioffe, V. Vanhoucke, and A. A. Alemi, "Inception-v4, inception-resnet and the impact of residual connections on learning," in *Proc. 31st AAAI Conf. Artif. Intell.*, 2017.
[21] E. E. Hansley, M. P. Segundo, and S. Sarkar, "Employing fusion of learned and handcrafted features for unconstrained ear recognition," *IET Biometrics*, vol. 7, no. 3, pp. 215–223, 2018.
[22] K. Zhang, Z. Zhang, Z. Li, and Y. Qiao, "Joint face detection and alignment using multitask cascaded convolutional networks," *IEEE Signal Process. Lett.*, vol. 23, no. 10, pp. 1499–1503, Oct. 2016.
[23] S. Dodge, J. Mounsef, and L. Karam, "Unconstrained ear recognition using deep neural networks," *IET Biometrics*, vol. 7, no. 3, pp. 207–214, 2018.
[24] D. T. Nguyen, T. D. Pham, N. R. Baek, and K. R. Park, "Combining deep and handcrafted image features for presentation attack detection in face recognition systems using visible-light camera sensors," *Sensors*, vol. 18, no. 3, p. 699, 2018.
[25] L. Nanni, S. Ghidoni, and S. Brahnam, "Ensemble of convolutional neural networks for bioimage classification," *Appl. Comput. Informat.*, 2018.
[26] Y. Zhang, Z. Mu, L. Yuan, and C. Yu, "Ear verification under uncontrolled conditions with convolutional neural networks," *IET Biometrics*, vol. 7, no. 3, pp. 185–198, 2018.
[27] T. I. Dhamecha, A. Nigam, R. Singh, and M. Vatsa, "Disguise detection and face recognition in visible and thermal spectrums," in *Proc. IEEE Int. Conf. Biometrics*, 2013, pp. 1–8.
[28] T. I. Dhamecha, R. Singh, M. Vatsa, and A. Kumar, "Recognizing disguised faces: Human and machine evaluation," *PLOS One*, vol. 9, no. 7, 2014.
[29] T. Ahonen, A. Hadid, and M. Pietikainen, "Face description with local binary patterns: Application to face recognition," *IEEE Trans. Pattern Anal. Mach. Intell.*, vol. 28, no. 12, pp. 2037–2041, Dec. 2006.
[30] B. Zhang, Y. Gao, S. Zhao, and J. Liu, "Local derivative pattern versus local binary pattern: Face recognition with high-order local pattern descriptor," *IEEE Trans. Image Process.*, vol. 19, no. 2, pp. 533–544, Feb. 2010.
[31] K.-C. Fan and T.-Y. Hung, "A novel local pattern descriptor-local vector pattern in high-order derivative space for face recognition," *IEEE Trans. Image Process.*, vol. 23, no. 7, pp. 2877–2891, Jul. 2014.
[32] S. Chakraborty, S. K. Singh, and P. Chakraborty, "Local gradient hexa pattern: A descriptor for face recognition and retrieval," *IEEE Trans. Circuits Syst. Video Technol.*, vol. 28, no. 1, pp. 171–180, Jan. 2018.
[33] N. Dalal and B. Triggs, "Histograms of oriented gradients for human detection," in *Proc. IEEE Comput. Soc. Conf. Comput. Vision Pattern Recognit.*, 2005, vol. 1, pp. 886–893.
[34] S. Christian, V. Vincent, I. Sergey, S. Jonathon, and W. Zbigniew, "Rethinking the inception architecture for computer vision," in *Proc. IEEE Conf. Comput. Vision Pattern Recognit.*, 2016, pp. 2818–2826.