Bidirectional Residual Declarative Network: A Deep Learning Framework for Robust Facial Expression Recognition

Ruikai Cui

Research School of Computer Science, Australian National University, Canberra Australia
u6919043@anu.edu.au

Abstract. Automatic facial expression recognition is of great importance for the use of human-computer interaction (HCI) in various applications. Due to the large variance in terms of head position, age range, and so on, detecting and recognizing human facial expressions in realistic environments remains a challenging task. In recent years, deep neural networks have started being used in this task and demonstrate state-of-the-art performance. Here we propose a reliable framework for robust facial expression recognition. The basic architecture for our framework is ResNet-18, in combination with a declarative L_p sphere/ball projection layer and a bidirectional fully connected (FC) layer. The proposed framework also contains data augmentation, voting mechanism, and a YOLO based face detection module. The performance of our proposed framework is evaluated on a semi-natural static facial expression dataset *Static Facial Expressions in the Wild* (SFEW), which contains over 800 images extracted from movies. Results show excellent performance with a validation accuracy of 53.90% and a test accuracy of 57.24%, which indicates the considerable potential of our framework.

Keywords: Facial Expression Recognition · Deep Declarative Network · Bidirectional Neural Network · Deep Learning · Data Augmentation.

1 Introduction

Facial expression detection and recognition, i.e. the task of automatically perceiving and recognizing human facial expressions based on vision inputs or other bio signals, is of great interest in the HCI research field. Although facial expression recognition under lab-controlled environments has been well addressed [14, 19, 18, 16, 3], it remains a challenging problem to recognize unconstrained expressions captured in realistic environments.

The difficulty of this task is mainly caused by three factors. First, human facial expressions are dynamic in nature [7], which makes the effectiveness of recognizing expressions via static vision data inherently worse than video-based methods [2]. Second, the real-world environment, such as illumination, camera lens, would pose an influence on recognition performance. Third, human appearance various from each other. The within-class variability is usually large due to age, gender, etc. Some studies have shown the influence of these factors [4].

In recent years, researchers have focused more on recognize facial expressions in realistic environments. The Static Facial Expressions in the Wild (SFEW) dataset [7] was proposed, aiming to simulate the complex real-world environment. Over 800 images were extracted from 37 movies and it contains a total of 95 subjects. This real-world environment really poses a significant challenge to existing approaches. A number of methods which performs great on lad-controlled datasets like JAFFE [17], PIE [25], and MMI [22], yield significant lower performance on this dataset [7].

To achieve good recognition performance, a method that robust to the influence factors is needed. Here, we introduce a novel neural network architecture for facial expression recognition, the Bidirectional Residual Declarative Network, which is constructed based on ResNet [10] then improved by ideas distilled from deep declarative networks [8] and bidirectional neural networks [20], and it demonstrates competitive performance. In addition, we describe a modeling framework consisting of data augmentation method, voting mechanism, and a YOLO based face detection module, which behaves well when the dataset doesn't contain enough images. The performance of our framework was tested on the SFEW dataset.

This article is organized as follows. In section 2, we provide details of our proposed framework and discuss technique details on implementation. In section 3, we give an experimental analysis of our framework. Section 5 presents the performance evaluation of our framework as well as comparison with methods proposed by other researchers on the SFEW dataset. Finally, we would discuss the limitations of our work and future work, and then give a conclusion about our contribution.

2 Framework

In this section, we introduce the architecture of the Bidirectional Residual Declarative Network in full detail (Section 3.1) as well as the data augmentation methodology (Section 3.2), the face detection module (Section 3.3), and voting mechanism (Section 3.4).

2.1 Architecture

ResNet We use the ResNet-18 as the backbone architecture. The ResNet is first proposed in [10], aiming to ease the training process of deep neural networks by introducing residual connections. The original architecture

as well as its variations have been well examined on visual recognition tasks like the ImageNet Classification challenge [24], CIFAR-10 dataset, and achieved better performance compared with previous architectures like AlexNet [11, 26, 15]. Therefore, we build our network upon ResNet-18. The structure of our network is shown in Fig. 1.

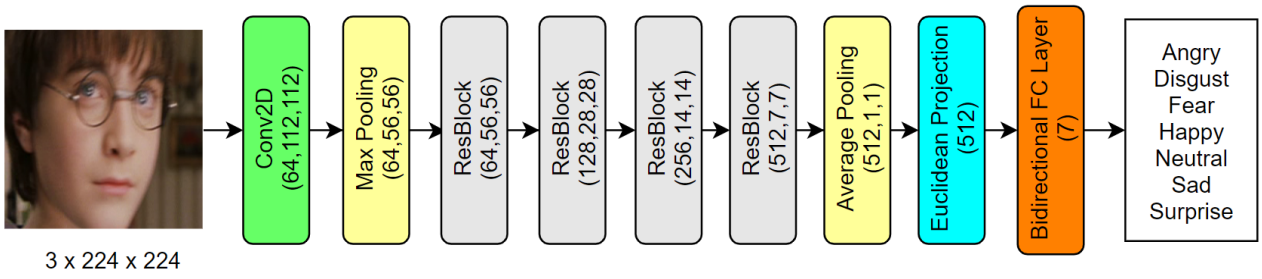


Fig. 1. Structure of Bidirectional Residual Declarative Network. The second line of each block denotes the output dimension.

As shown in Fig. 1, for an image input, it would first be downsampled by first a convolution layer and then a max pooling layer. The output of the pooling layer would then be sent into the ResBlocks. We build the four ResBlocks according to [10]. The dimension of the last ResBlock' s output is $(512, 7, 7)$. So, we first adopt average pooling and then flatten the output to change it into a vector of length 512. This vector would then be normalized by applying L_p sphere/ball projection, and we changed the original fully connected layer into a bidirectional layer to further improve the robustness and generalization ability. The bidirectional layer would finally yield the prediction.

After each convolution and before activation, we adopt batch normalization (BN) [12]. Besides, BN would be used before the projection layer. Stochastic gradient descent (SGD) with a mini-batch size of 256 is our choice for optimization. The learning rate starts from 0.1 and would be adjusted by dividing it by 10 every 30 epochs. We also apply a 0.0001 weight decay rate and 0.9 momentum rate.

With regard to the model complexity, our proposed network has a total of 11,180,103 learnable parameters and the parameter size is about 42 MB.

Projection Layer Proper regularization is critical for speeding up training and improving generalization performance [21]. Rather than use some conventional regularization methods, we take advantage of the new development in the deep learning field, namely deep declarative networks [8], and introduce L_p sphere/ball projection into our network.

Conventional approaches toward regularization would do the operation that for $x \in \mathbb{R}^n \mapsto y \in \mathbb{R}^n$, we have

$$y = \frac{1}{\|x\|_p} x \tag{1}$$

where $\|\cdot\|_p$ denotes the L_p-norm.

Instead of doing regularization in this way, we apply the declarative approach, which can be generalized as

$$y_p \in argmin_{u \in \mathbb{R}^n} \frac{1}{2}\|u - x\|_2^2, \ subject\ to\ \|u\|_p = r \tag{2}$$

$$y_p \in argmin_{u \in \mathbb{R}^n} \frac{1}{2}\|u - x\|_2^2, \ subject\ to\ \|u\|_p < r \tag{3}$$

for L_p-sphere projection and L_p-ball projection respectively. r denote the radius of the sphere or ball. We set it to 1 in our network, which could be interpreted as projecting with a unit sphere or ball constrain.

In terms of the projection layer implementation, we use the *ddn* library [9], and then integrate it into our network by adding it before the FC layer. For the choice of projection type, we tested L_1, L_2, L_∞-sphere/ball projection, and decided to apply the L_2-sphere projection in our final network to achieve the best performance.

Bidirectional FC Layer The bidirectional fully connected layer is derived from the idea of bidirectional neural networks (BDNN) [20], which is proposed to enable neural networks to remember input patterns as well as output vectors. It is demonstrated in the original work that the BDNNs perform well on tasks such as finding cluster centers and class prototypes. Due to its ability of learning input patterns from outputs, we could use it to reduce the network generalization error and thus improve the robustness of our model.

Fig. 2 shows the topology of conventional neural networks (NN) and BDNNs. The difference between these two topologies is that the training process in BDNNs is bidirectional, i.e. the error would not only backpropagate from right to left but also left to right. To implement such a network, we construct a symmetric network for the input-to-output network, which means the number of output and input neural is exchanged. The weights of

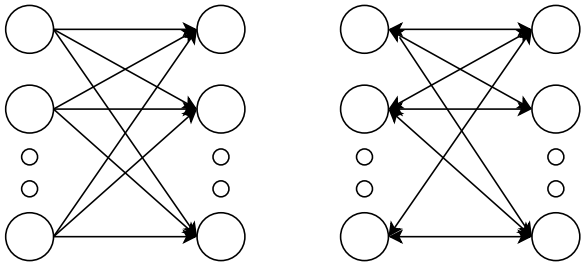


Fig. 2. Topology of NNs (Left) and BDNNs (Right)

these two networks are associated by shared memory. Before training the left-to-right network, we would first use the output to predict the input and backpropagate the error, and then update the shared weights.

For our task, we set the input and output neural size as 512 and 7 respectively, and thus the symmetric would use a 7-D vector to predict the pattern of a 512-D vector. No hidden layers are added to make it consistent with the ResNet backbone.

2.2 Data Augmentation

Machine learning algorithms often suffer from the overfitting problem, and this is more significant for deep learning especially when the dataset is small, and the network is deep. To address this problem, we rely on data augmentation so that, in each iteration, the algorithm never sees the exact same set of images

The face detector would crop a 256×256 region in a source image corresponding to the face region. We first horizontally flip it to create a mirror image and then expand these two images' width and height by 2 times respectively. By applying this augmentation strategy, we expand the original dataset by 6 times. This process is to mimic viewing subjects from different angles. As for training, the required input size is 224×224. We would randomly crop such a region from the preprocessed images with the per-pixel mean subtracted . This allows our model to learn from not only the whole face but also a partial region.

2.3 Face Detection

To learning meaningful representations of facial expressions, locating faces is the first step. We use the library *faced* as our face detection module. When setting the recognition threshold to 0.6, it yields 66 images that do not contain faces, which is 7.56% of the whole cropped images. Therefore, we can rely on these detected faces to train our network. We can observe a significant improvement in recognition accuracy as the network trained by detected faces is over 45% while the same network but trained with raw images only has an accuracy of around 20%.

The faced library would do face detection in two stages. At stage one, a custom fully convolutional neural network (FCNN) implemented based on YOLO[23] would take a 288×288 RGB image and outputs a 9×9 grid where each cell can predict bounding boxes and the probability of one face. [13] At stage two, a convolutional neural network (CNN) would be used to take the face-containing rectangle and predict the face bounding box. This module is trained on the WIDER FACE dataset [27]. In the end, we can get bounding boxes of a face and the probability of how likely it is really a face.

2.4 Voting Mechanism

There are many scenarios for the output of the face detect module, i.e. faces detected, partial face detected, non-face detected, and their combinations. Fig. 3 illustrates the situations where multiple possible faces are detected. Although they are different images, each pair is cropped from the same image. This is due to the complexity of the real-world environment and the limit of our face detection module. To address this issue and take advantage of face probability, we introduce the voting mechanism to our framework.

Fig. 3. Faces detected. (a) partial face (left) (b)non face (middle) (c) desired face (right)

When detecting faces, the detection module would also give the probability of one face while our network would yield a 7-D vector as the probability of each expression. When we have more than one detected faces in an image, instead of taking the image with the highest probability as the expression prediction evidence, we would average the 7-D vector weighted by the face probability.

3 Experimental Analysis

In this section, we provide an experimental analysis of the Bidirectional Residual Declarative Network architecture as well as the voting mechanism.

3.1 Evaluation Method

To evaluate the proposed architecture, we constructed the confusion matrix for prediction results and apply several accuracy metrics such as accuracy, precision, and recall:

$$accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{4}$$

$$precision = \frac{TP}{TP + FP} \tag{5}$$

$$recall = \frac{TP}{TP + FN} \tag{6}$$

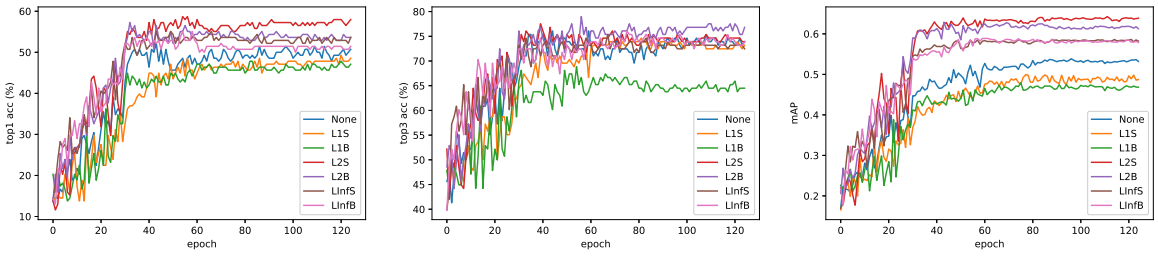
where TP, TN, FP, FP are the true positive, true negative, false positive, and false negative classifications, respectively. Among them, accuracy would be our principal performance indicator since its simplicity, and we would also consider the top 3 classification accuracy as some expressions are similar and even hard for humans to correctly identify.

We didn't adopt cross-validation in our evaluation as this is expensive considering training such a deep neural network would take over 1 hour on an RTX2070 platform. Instead, we split the dataset into train, test, and validation sets. The performance results in section 3.2 are evaluated on the test set, and we would use the validation set as the final evaluation of our model's performance.

3.2 Ablation Study

In this section, we design two experiments to evaluate the performance of the projection layer and the bidirectional FC layer. We use the SFEW dataset and this is also the dataset that our framework would finally be evaluated on. In the first experiment, we exam the performance improvement that the projection layer gives us as well as the effect of different projection types. In the second experiment, we would test the bidirectional FC layer's performance and its influence to different projection types. We didn't apply voting in both experiments. Therefore, we manual deleted these non-face images in our test set.

Projection Layer Performance We first test the projection layers, and therefore, we use a conventional NN as the FC layer instead of a bidirectional FC layer. As discussed in Section 3.1, we adopted six different projections, which are L_1-sphere (L1S), L_1-ball (L1B), L_2-sphere (L2S), L_2-ball (L2B),L_∞-sphere (LInfS),L_∞-ball(LInfB), respectively, and combined with a network without projection, we trained 7 networks in total, aiming to compare their effect in terms of classification performance. The result is shown in Fig. 4



(a) top-1 accuracy (b) top-3 accuracy (c) mean Average Precision (mAP)

Fig. 4. Model Performance by Epoch

From Fig. 4, we can observe that the network with L2S projection yields the best performance in both top-1 accuracy and mAP, and L2B projection is the best choice if we regard the top-3 accuracy as the most important metric. The difference between networks with the projection layer and without the projection layer is significant. We can observe a nearly 10% top-1 accuracy improvement after adding the L2S projection. However, some projection types, such as L1S and L1B, would give a similar performance with plain ResNet-18, or even worse performance.

The difference in convergence performance is also shown in this figure. Networks with projection would generally have a faster convergence speed. However, this difference is not obverse. In addition, declarative layers (the projection layer) are more computationally expensive compared with conventional layers, and some of them even do have closed-form solutions, such as (LInfS) [8]. Therefore, the convergence advantage is negligible.

Bidirectional FC Layer Performance Our second ablation study aims to explore the effect of the bidirectional FC layer. Therefore, we conduct experiments to test the performance of our architecture with and without the bidirectional FC layer. Table. 1 shows the network performance on the test set after training for 125 epoch.

Table. 1 Performance of Models with and without Bidirectional FC Layer.

	Top-1		Top-3		mAP	
	FC	BDFC	FC	BDFC	FC	BDFC
None	50.72	47.82	73.91	71.73	53.18	44.70
L1S	48.55	43.47	73.18	71.01	48.71	43.90
L1B	47.10	45.65	64.49	66.66	46.83	52.70
L2S	57.97	57.24	73.91	77.53	63.85	64.61
L2B	53.62	57.24	76.81	75.36	61.25	64.89
LInfS	53.62	52.89	72.46	73.91	58.21	57.68
LInfB	51.44	46.37	73.91	71.73	57.90	53.61

The results show that different projection layers behave differently with the bidirectional FC layer. The bidirectional FC layer significantly reduced the performance for networks with L_1-sphere/ball projection layers and the network without projection. It yields a similar performance for networks with L_∞-sphere/ball projection except for the top-1 accuracy for the model with L_∞-ball projection. However, we can observe a performance improvement for networks with L_2-sphere/ball projection. For the model with L_2-sphere projection, the Top-3 accuracy and mAP improved a lot despite a slight decrease in terms of Top-1 accuracy.

This is consistent with the theory that BDNNs can learn for both input and output and thus have a better generalization ability.

3.3 Voting Mechanism

We tested the performance of the voting mechanism on the validation set. The prediction accuracy for the network without voting is 52.78% and it is 53.90% for the network with voting. The network contains the L2S projection and the bidirectional FC layer. We can observe that both accuracies decreased significantly on the validation. However, this is normal as we did a lot of hyperparameter tuning during training. Although it is not a huge improvement, the voting mechanism indeed improved the performance. The reason is that voting would only happen when the face detector finds multiple faces in one image. However, for the majority, the detector would only detect and crop one face region. Moreover, voting relies on the face probability given by the detector, which further increases the uncertainty. Therefore, we cannot hope the voting mechanism brings significant improvement in our framework.

4 Result & Discussion

We present and discuss our model's performance on the validation set of SFEW dataset. We would also compare it with approaches proposed by other researcher to determine whether it is a competitive framework for the robust facial expression recognition task.

4.1 Performance

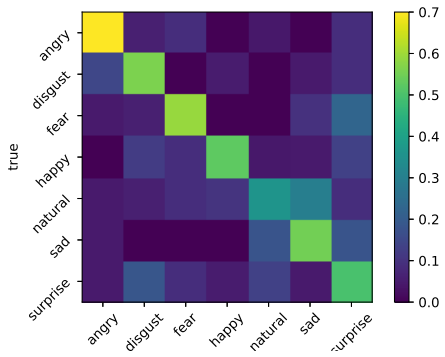


Fig. 5. Confusion Matrix of our Framework on Validation Set.

We present the confusion matrix in Fig. 5. The color of each block indicates the recall of that class. According to this figure, the recall of Angry facial expression is the highest. The model misunderstands a number of natural

and sad expression. However, for some images of these two classes, it is even hard for humans to identify the correct emotion.

4.2 Comparison

We compare our proposed framework with other researchers' approaches. The performance of the first approach is the SFEW baseline provided at [1]. They detect faces using Mixture of Pictorial Structures and then compute the Pyramid of Histogram of Gradients and Local Phase Quantisation features for the aligned faces. A support vector machine (SVM) was trained on the vector computed from feature fusion. The second approach is proposed in [6]. They proposed a new feature descriptor, namely Histogram of Oriented Gradients from Three Orthogonal Planes (HOG_TOP), and they adopt Multiple Kernel Learning (MKL) to find an optimal feature fusion. The classifier for their approach is also a SVM but with multiple kernels.

Table. 2 Performance Comparison with Other Approaches.

	SFEW Baseline	MKL[6]	Our Method
Test	39.33%	45.21%	57.24%
Val	36.08%	40.21%	53.90%

The results of all the approaches are shown in Table. 2. We can observe that our approach has a significantly better performance compared with the SFEW baseline, and surpass the MKL approach by 13% in terms of the validation performance. This indicates that our framework has a great potential on the real-world facial expression recognition task.

5 Limitation & Future Work

We aim to simulate the real-world environment and exam the performance of our proposed framework. However, images in this dataset are still not in the real world. For example, movies may use the illumination to reflect a character's emotion, which means a person in a dark environment is more likely to have a negative emotion and a bright environment may indicate the person is happy. Therefore, future work can improve our work by using a dataset that more close to the real world.

Future work could also extend our work by generalizing the voting mechanism to other components. For example, the face detector can be improved by introducing other framework based detectors to the detection process. The detected faces could either be an average of these detectors or from the detector with the highest confidence.

Fine-tuning on a pre-trained model is another approach that can possibly improve our model's performance. Although we applied data augmentation, the dataset is still not sufficient for a deep neural network. Fine-tuning a model pre-trained on a bigger dataset like the FER dataset [5] would be a good method to avoid overfitting and increase the generalization ability.

6 Conclusion

In this paper, we present a new neural network architecture for facial expression recognition, which is based on ResNet-18 and combined with a declarative projection layer and a bidirectional FC layer. The proposed framework also includes a data augmentation method, a face detection module, and a voting mechanism. We examined the projection layer as well as the bidirectional FC layer by an ablation study, and proved that they indeed improved the backbone network. With a combination of these techniques, we achieved a competitive performance with 57.24% test accuracy and 53.90% validation accuracy, which demonstrates the proposed framework is reliable and robust for facial expression recognition in a real-world environment.

7 Acknowledgment

Our original dataset is the depression dataset, and we switch to the SFEW dataset with permission from Prof. Gedeon.

References

1. The third emotion recognition in the wild challenge 2015, `https://cs.anu.edu.au/few/ChallengeDetails2015.html` Accessed June 1, 2020
2. Ambadar, Z., Schooler, J.W., Cohn, J.F.: Deciphering the enigmatic face: The importance of facial dynamics in interpreting subtle facial expressions. Psychological science **16**(5), 403–410 (2005)
3. Caifeng Shan, Shaogang Gong, McOwan, P.W.: Robust facial expression recognition using local binary patterns. In: IEEE International Conference on Image Processing 2005. vol. 2, pp. II–370 (2005)

4. Calder, A.J., Keane, J., Manly, T., Sprengelmeyer, R., Scott, S., Nimmo-Smith, I., Young, A.W.: Facial expression recognition across the adult life span. Neuropsychologia **41**(2), 195 – 202 (2003), the cognitive neuroscience of social behavior
5. Carrier, P.L., Courville, A., Goodfellow, I.J., Mirza, M., Bengio, Y.: Fer-2013 face database. Universit de Montral (2013)
6. Chen, J., Chen, Z., Chi, Z., Fu, H.: Emotion recognition in the wild with feature fusion and multiple kernel learning. In: Proceedings of the 16th International Conference on Multimodal Interaction. p. 508–513. ICMI '14, Association for Computing Machinery, New York, NY, USA (2014)
7. 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)
8. Gould, S., Hartley, R., Campbell, D.: Deep declarative networks: A new hope. arXiv preprint arXiv:1909.04866 (2019)
9. Gould, S., Hartley, R., Campbell, D.: Deep declarative networks: A new hope. Tech. rep., Australian National University (arXiv:1909.04866) (Sep 2019)
10. 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)
11. He, K., Zhang, X., Ren, S., Sun, J.: Identity mappings in deep residual networks. In: European conference on computer vision. pp. 630–645. Springer (2016)
12. Ioffe, S., Szegedy, C.: Batch normalization: Accelerating deep network training by reducing internal covariate shift. arXiv preprint arXiv:1502.03167 (2015)
13. Itzcovich, I.: faced. `https://github.com/iitzco/faced` (2018)
14. Kotsia, I., Pitas, I.: Facial expression recognition in image sequences using geometric deformation features and support vector machines. IEEE Transactions on Image Processing **16**(1), 172–187 (Jan 2007). https://doi.org/10.1109/TIP.2006.884954
15. Krizhevsky, A., Sutskever, I., Hinton, G.E.: Imagenet classification with deep convolutional neural networks. In: Advances in neural information processing systems. pp. 1097–1105 (2012)
16. Liu, P., Han, S., Meng, Z., Tong, Y.: Facial expression recognition via a boosted deep belief network. In: The IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (June 2014)
17. Lyons, M., Akamatsu, S., Kamachi, M., Gyoba, J.: Coding facial expressions with gabor wavelets. In: Proceedings Third IEEE international conference on automatic face and gesture recognition. pp. 200–205. IEEE (1998)
18. Ma, L., Khorasani, K.: Facial expression recognition using constructive feedforward neural networks. IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics) **34**(3), 1588–1595 (2004)
19. Moore, S., Bowden, R.: Local binary patterns for multi-view facial expression recognition. Computer vision and image understanding **115**(4), 541–558 (2011)
20. Nejad, A., Gedeon, T.: Bidirectional neural networks and class prototypes. In: Proceedings of ICNN'95-International Conference on Neural Networks. vol. 3, pp. 1322–1327. IEEE (1995)
21. Oymak, S.: Learning compact neural networks with regularization. arXiv preprint arXiv:1802.01223 (2018)
22. Pantic, M., Valstar, M., Rademaker, R., Maat, L.: Web-based database for facial expression analysis. In: 2005 IEEE international conference on multimedia and Expo. pp. 5–pp. IEEE (2005)
23. Redmon, J., Divvala, S., Girshick, R., Farhadi, A.: You only look once: Unified, real-time object detection. In: Proceedings of the IEEE conference on computer vision and pattern recognition. pp. 779–788 (2016)
24. Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., Huang, Z., Karpathy, A., Khosla, A., Bernstein, M., et al.: Imagenet large scale visual recognition challenge. International journal of computer vision **115**(3), 211–252 (2015)
25. Sim, T., Baker, S., Bsat, M.: Short papers-face database-the cmu pose, illumination, and expression database. IEEE Transactions on Pattern Analysis and Machine Intelligence **25**(12), 1615–1618 (2003)
26. Szegedy, C., Ioffe, S., Vanhoucke, V., Alemi, A.A.: Inception-v4, inception-resnet and the impact of residual connections on learning. In: Thirty-first AAAI conference on artificial intelligence (2017)
27. Yang, S., Luo, P., Loy, C.C., Tang, X.: Wider face: A face detection benchmark. In: IEEE Conference on Computer Vision and Pattern Recognition (CVPR) (2016)