Dean&Francis
ISSN 2959-6157

Lightweight and accuracy upgrade: Surface partial Key point detection model based on deep reinforcement learning

Muhan Jia

Beijing University of Posts and Telecommunications, Beijing, China
*Corresponding author: Jmuhanbupt@outlook.com

Abstract:

This paper introduces a novel lightweight and high-accuracy model for facial key point detection, incorporating advancements in deep learning to optimize performance under limited computational resources. The proposed architecture, Inception-MLP, integrates a pre-trained Inception module for multi-scale feature extraction and replaces traditional fully connected layers with a single-hidden-layer multilayer perceptron (MLP), reducing complexity while maintaining strong representational power. Unlike conventional CNN-based models that often suffer from parameter redundancy and prolonged training times, this method strikes an effective balance among accuracy, model size, and training efficiency. Comparative evaluations against Basic CNN, VGG, and EfficientNet-Lite3 demonstrate that Inception-MLP offers superior performance across multiple metrics, making it suitable for real-world applications, especially on mobile or embedded devices. The model's ability to achieve precise localization of facial landmarks with fewer parameters highlights its deployment potential in scenarios requiring both lightweight inference and high reliability. This work contributes a feasible direction toward efficient deep learning-based facial analysis, providing a valuable reference for future model design in similar tasks.

Keywords: Facial key point detection; Inception module; Lightweight deep learning.

1. Introduction

As computer vision progresses rapidly, an increasing number of research efforts and industrial applications have focused on facial key point detection. These key points not only provide fundamental support for the geometric alignment of facial images but also significantly contribute to subsequent tasks such as face recognition, expression recognition, and 3D modeling [1]. Traditional methods rely on handcrafted features and shallow models, which are difficult to handle the variability of facial appearances and com-

plex backgrounds. Deep learning models autonomously extract hierarchical features via several layers of nonlinear transformations, capturing details from low-level elements such as edges and textures to high-level attributes like facial structure and pose, without requiring manual feature design [2]. As a result, deep learning is now widely recognized as the primary method for facial key point detection. Convolutional Neural Network (CNN) is a typical deep learning algorithm applied to facial key point detection [3]. They have deep structures that transform raw image pixels into multi-level feature representations, which contain semantic abstraction features required for various applications. The structure of CNNs is relatively simple, but to achieve high accuracy, they often require stacking a large number of convolutional layers [4]. Compared with CNN, Visual Geometry Group (VGG) contains more convolutional layers and fully connected layers. This increases the complexity of the network structure and also leads to a significant increase in the number of trainable parameters and training time [5]. EfficientNet-Lite3 is a relatively popular lightweight solution for facial key point recognition nowadays. It mainly adopts Mobile Inverted Bottleneck Convolution (MBConv) instead of the traditional convolutional layer to reduce the amount of calculation[6]. The most common real-world applications of facial keypoint detection models include facial unlocking in mobile apps and access control systems. These applications all require completion within a few seconds. Therefore, the deployability of deep learning models largely depends on their inference speed and their performance on mobile and embedded devices with limited computational capacity. All of this requires deep learning models to achieve lightweight deployment while improving accuracy. Inference time and the number of trainable parameters are important criteria for measuring the lightweight nature of a model.

In this paper, Basic CNN is used as the baseline. On the basis of Basic CNN, the Inception module is introduced for feature extraction, and fully connected hidden layers are used to replace traditional fully connected layers, thereby improving the convolutional neural network structure and forming Inception-MLP [7]. This enables the model to improve accuracy while reducing training time, with only a slight increase in the number of trainable parameters. Compared with Basic CNN, EfficientNet-Lite3 and VGG, this model achieves significant improvements and successfully balances accuracy and lightweight deployment. Inception-MLP provides a feasible solution for real-world scenarios that are sensitive to computing resources and have higher requirements for accuracy.

2. Principal of proposed method

2.1 Basic CNN

The network design of the Basic CNN model incorporates three convolutional layers, three max-pooling layers, and a single fully connected layer. The input is a single-channel grayscale image. A max pooling layer is introduced after every convolutional layer to downsample the feature maps. The max pooling layers perform down sampling to integrate the distributed spatial features extracted by the preceding convolutional and pooling layers into global semantic features, which are then mapped to the final output space. The fully connected layer contains 500 neurons, which is an empirically tuned configuration. This setup provides sufficient representational capacity without introducing excessive parameters that could lead to overfitting. All convolutional layers utilize a filter size of 3×3, with channel counts of 32, 64, and 128 in sequence. The lower-level convolutional layer (Conv1) focuses on basic textures, while the higher-level convolutional layer (Conv3) captures semantic features. ReLU activation is applied after all convolutional layers to introduce non-linearity and accelerate convergence [8]. A 2×2 pooling window is applied during each subsampling operation, using a stride of 2. The output layer adopts a linear activation function, which is suitable for regression tasks. In this paper, Basic CNN is used as the baseline.In this paper, subsequent improvements are made based on the baseline model to form the the introduced approach. The results are presented in Figure 1. The gray section represents the traditional convolutional neural network, while the blue section illustrates the proposed method in this paper.

ISSN 2959-6157

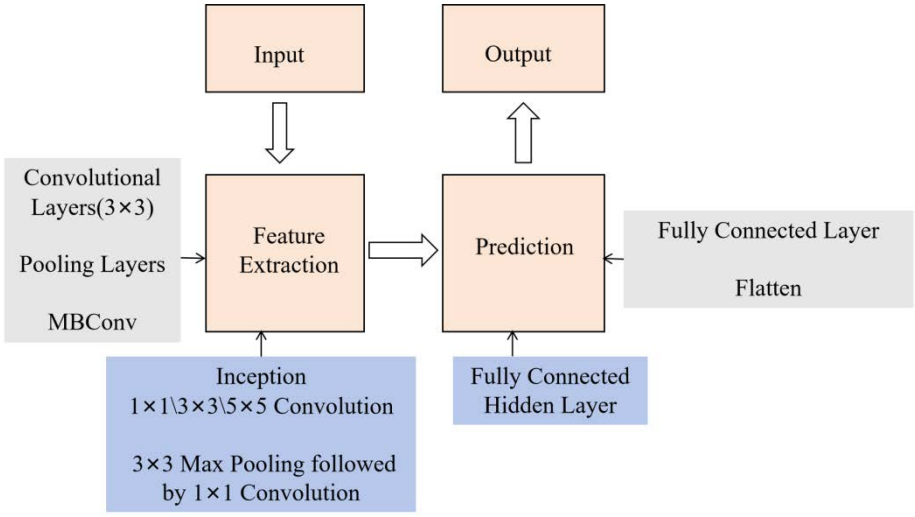

Fig. 1 The flowchart of Deep learning.

The total number of parameters in Basic CNN is 6,508,202, among which the fully connected layer accounts for approximately 98%. Although it can accomplish the task of facial key point detection, it suffers from drawbacks such as a large number of parameters and low efficiency. Its network structure is shown in Table 1.

Table 1. The network structure of Basic CNN.

Layer	Output Size	Trainable Parameters
Input	96×96×1	0
Conv1	94×94×32	320
Pool1	47×47×32	0
Conv2	45×45×64	18,496
Pool2	22×22×64	0
Conv3	20×20×128	73,856
Pool3	10×10×128	0
FC1	500	6,400,500
Output	30	1,5030

2.2 Inception-MLP

According to the above theoretical analysis of the Basic CNN structure, it can be concluded that the primary challenge in model compression stems from the fully connected layer. To address this concern, this paper introduces the Inception module to improve the framework of the CNN model. The Inception model is incorporated to overcome the limitations of traditional CNN structures in terms of computational efficiency and feature extraction capability. The model is not trained from scratch; instead, a pre-trained Inception module is used for feature extraction, combined with a fully connected network to perform key point coordinate regression.

The Inception-v3 module adopted in this paper is pre-trained and includes convolutional layer weights, batch normalization parameters, and fully connected layer weights [9]. The choice of using a pre-trained Inception model for transfer learning is based on the following considerations: First, Inception has demonstrated excellent feature extraction capabilities in the ImageNet classification task. Its multi-scale convolution structure can capture hierarchical features ranging from edge textures to semantic objects. Second, ImageNet contains 1.4 million images across 1,000 categories, which enables the model to learn highly generalized low-level visual features. Studies have shown that such low-level features are transferable across different computer vision tasks. Given that the training dataset used in this paper consists of only 7,049 images, training a model from scratch is prone to overfitting and poor performance on the test set. A pre-trained model can effectively mitigate this issue. Third, using a pre-trained

module significantly reduces the number of parameters that need to be trained, thereby achieving the goal of lightweighting as much as possible.

The Inception model is primarily designed to address the parameter redundancy and low computational efficiency of Basic CNN. Through its sparse connection architecture, the model achieves a "globally sparse, locally dense" computational pattern. Multi-scale convolutional parallel processing enables simultaneous capture of features at different granularities [10]. In this paper, a dimensionality reduction mechanism is applied. When extracting features from input images using the Inception network, each module is composed of four concurrently operating branches: a 1×1 convolution, two convolutions (3×3 and 5×5) each with a preceding 1×1 bottleneck for dimensionality reduction, and a 3×3 max pooling operation succeeded by a 1×1 convolution for feature projection.

The 1×1 convolution is responsible for capturing fine-grained local information, the 3×3 convolution is designed to extract features over a moderate spatial range, the 5×5 convolution captures broader contextual information, and the pooling operation enhances feature invariance. This structure is particularly well-suited for facial key point detection tasks, as different key points (such as eye corners and the nose tip) require feature support at different scales. In addition, the 1×1 convolution used for dimensionality reduction significantly reduces the computational cost of the 3×3 and 5×5 convolutions, thereby improving model efficiency while maintaining depth. For example, when the input depth is 128, the 5×5 convolutional path first performs a 1×1 convolution to reduce the dimensionality to 16 channels, followed by a 5×5 convolution with 32 channels. Finally, the outputs of all four paths are concatenated into a 256-channel feature map. The total number of parameters in the model is 9,977,258.

According to the above calculations, the higher dimensionality of features extracted by the Inception module leads to an increase in the input size of the fully connected layer. As a result, the parameter count increases. For the purpose of reducing model complexity and accelerating training, this paper introduces a single-hidden-layer Multilayer Perceptron (MLP) to replace the original fully connected layer, based on the Inception-CNN architecture [11]. Figure 2 illustrates the network structure of Inception-MLP.The network takes a 96×96×1 grayscale image as input. Features are extracted through the Inception module, producing a high-dimensional feature map of size 17×17×768. This is then reduced to 7×7×768 via adaptive average pooling and flattened into a 37,632-dimensional vector. A dense hidden layer with 200 neurons receives the input vector, after which an output layer generates the 30 corresponding facial landmark coordinates. The architecture is lightweight and possesses strong feature extraction capability, making it suitable for facial key point detection tasks.

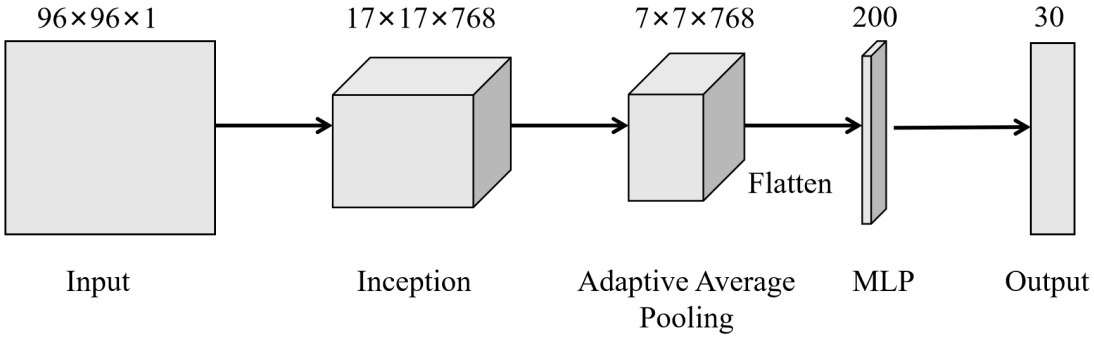

Fig. 2 Inception-MLP.

MLP compresses the multi-dimensional features extracted by the Inception module into a 200-dimensional hidden space to integrate the features. It learns complex functional relationships through activation by the ReLU function and maps the features to the coordinate space using a linear output layer [12]. A total of 7,532,630 parameters are included in the Inception-MLP, as shown in Table 2.

Table 2. The network structure of Inception-MLP.

Layer	Output Size	Trainable Parameters
Input	96×96×1	0
Inception Feature Extraction	17×17×768	0
Adaptive Average Pooling	7×7×768	0

ISSN 2959-6157

Flatten	37,632	0
Fully Connected Hidden Layer	200	7,526,600
Output	30	6,030

2.3 EfficientNet-Lite3

The primary distinction between EfficientNet-Lite3 and CNN can be found in its use of a lightweight module called MBConv. In MBConv, a 1×1 convolution is first applied for dimensional expansion. This is followed by a depth-separated convolution, which applies spatial convolution separately to each channel. Then, a pointwise convolution (1×1) is used for dimensionality reduction and to reintegrate information across channels. Finally, the output is produced through residual connections. The total number of parameters in EfficientNet-Lite3 is 79,136,642, and its architecture is shown in Table 3.

Table 3. The network structure of EfficientNet-Lite3.

Layer	Output Size	Trainable Parameters
Input	96×96×1	0
StemConv	48×48×32	288
MBConv1	48×48×16	9,824
MBConv2	48×48×24	280,096
MBConv3	24×24×40	1,056,544
MBConv4	12×12×80	1,644,960
MBConv5	6×6×112	17,562,912
MBConv6	6×6×192	45,987,840
MBConv7	3×3×320	12,539,008
HeadFC	500	39,600
Output	30	15,390

2.4 VGG

VGG, as a classical convolutional neural network architecture, consists of eight convolutional layers (all using 3×3 kernels) and four fully connected layers. It captures more complex features through multiple layers of nonlinear transformations [13]. Each 2×2 max pooling step reduces the spatial resolution of the feature maps by 50% and simultaneously doubles the channel count, resulting in a consistent block pattern. Its network architecture is shown in Table 4, with a total number of parameters amounting to 9,811,422.

Table 4. The network structure of VGG.

Layer	Output Size	Trainable Parameters
Input	96×96×1	0
Conv1	64×94×94	640
Conv2	64×92×92	36,928
Maxpool1	64×46×46	0
Conv3	128×44×44	73,856
Conv4	128×42×42	147,584
Maxpool2	128×21×21	0
Conv5	256×19×19	295,168
Conv6	256×17×17	590,080
Maxpool3	256×8×8	0

FC1	512	8,389,632
FC2	512	262,656
FC3	512	262,656
FC4	512	262,656
Output	30	15,390

3. Results

3.1 Dataset

The Facial Keypoints Detection dataset provided by Kaggle was utilized for training the models. Each face image contains 15 facial key point positions, such as the inner corner of the left eyebrow; inner, center, and outer points of both eyes; the central point of the nose; the lateral endpoints of the lips; and the central top and bottom points of the lips. Each detected key point is characterized by a real-valued (x, y) coordinate pair defined in image pixel space.

The Facial Key points Detection dataset contains a total of 8,832 grayscale images with a resolution of 96×96 pixels. The dataset is divided into separate subsets for training and testing purposes, where 7,049 images constitute the training subset and 1,783 images are employed for model testing.

3.2 Different Model Results

Basic CNN, Inception-MLP, EfficientNet-Lite3 and VGG were trained for facial key point prediction. Mean squared error (MSE) was employed by all models to quantify loss, while PCK@0.1 served as the metric for accuracy evaluation. The Adam optimizer was applied in all cases, which adjusts the learning rate based on squared gradients and computes moving averages of the gradients [14]. All models were trained for a total of 600 epochs. Their training loss processes are shown in Figure 3. Table 5 presents a comprehensive comparison of training time, accuracy, MSE, and trainable parameter counts for the three models.

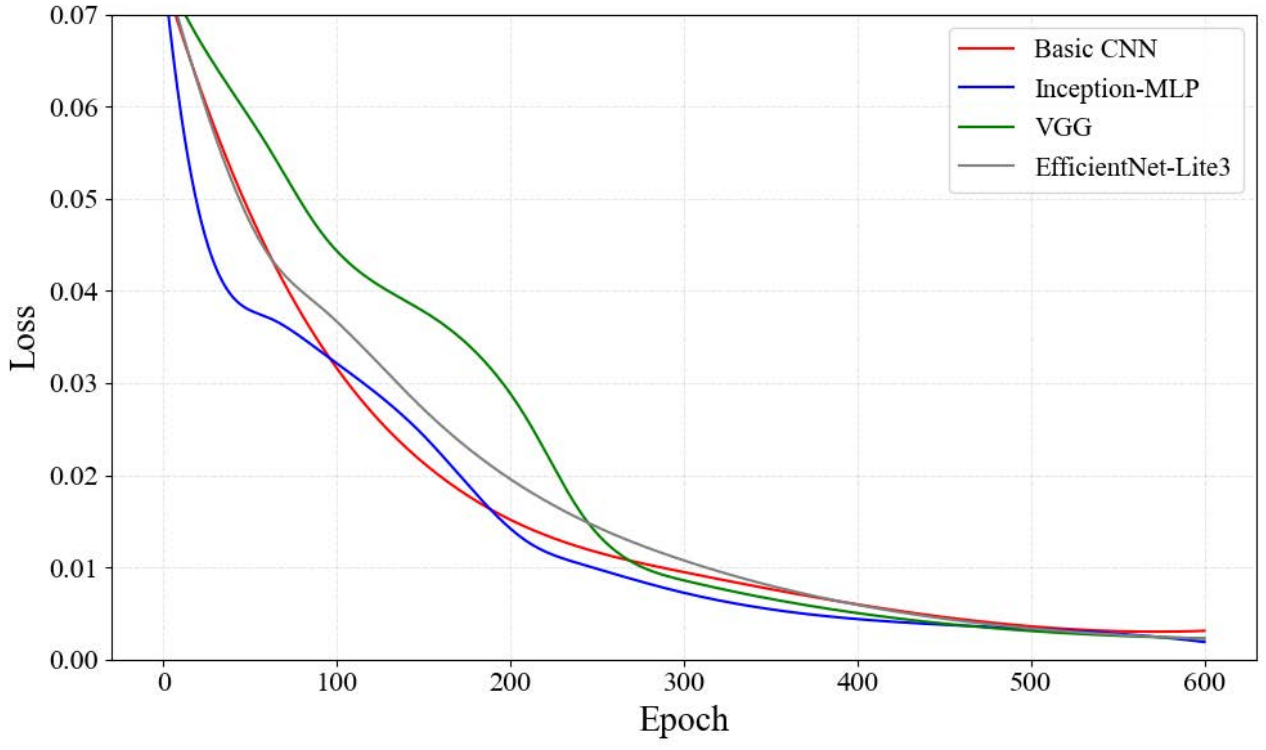

Fig. 3 Basic CNN, Inception-MLP, VGG, EfficientNet-Lite3 Loss.

ISSN 2959-6157

Table 5. Models comparison.

Model	Accuracy	Loss Value	Training Time/Epoch	Trainable Parameters (Million)
Basic CNN	81.2%	0.00305	152s	6.5
Inception-MLP	83.3%	0.00227	121s	7.5
VGG	82.8%	0.00238	256s	9.8
EfficientNet-Lite3	83.1%	0.00230	133s	7.9

From the data, it can be observed that compared to Basic CNN, Inception-MLP improves accuracy by 2.1%, reduces training time by 31 seconds, and increases the number of parameters by approximately 15%. Although the number of parameters increases slightly, its superior performance in both accuracy and training time makes it more suitable for lightweight deployment than the baseline model.Compared to VGG and EfficientNet-Lite3, two well-established and widely used convolutional neural network models, Inception-MLP improves accuracy by 0.5% and 0.2%, reduces training time by 135 seconds and 12 seconds, and decreases the number of parameters by 2.3 million and 0.4 million, respectively.

Overall, compared with the three mainstream deep learning models(Basic CNN, VGG, and EfficientNet-Lite3), Inception-MLP achieves higher accuracy, shorter training time, and a moderate number of parameters. These metrics suggest that Inception-MLP can serve as a superior lightweight deployment solution in real-world applications, potentially replacing the other three models.

4. Conclusion

This paper mainly investigates the problem of facial key point prediction given raw facial images. Specifically, for a given 96×96 image, we predict 15 pairs of (x, y) coordinates corresponding to facial key points. The baseline for comparison is provided by Basic CNN, a standard convolutional neural network. Based on it, this paper improves the network structure by further exploring the sparse connectivity of the Inception module and replacing the fully connected layer with an MLP structure, aiming to improve accuracy while keeping computational complexity as low as possible to meet the requirements of facial key point detection. Experiments conducted on the Kaggle dataset demonstrate the effectiveness of the deep structures, especially Inception-MLP, compared to Basic CNN, EfficientNet-Lite3 and Simple VGG regarding accuracy, training efficiency, and computational cost. This helps verify the feasibility of lightweight deployment for facial key point prediction in practical scenarios.

Future research directions based on this paper include two aspects:

a. This paper has demonstrated the effectiveness of the Inception model as a pre-trained model; however, training from scratch may have potential to further improve performance.

b. As shown by the results in this paper, compared to other model-based methods, using the Inception module and MLP structure may increase time complexity and parameter count, which are unfavorable for lightweight deployment. Nonetheless, the results have shown significant improvements. Future work will focus on designing deep architecture specifically tailored for this task to further enhance performance.

References

[1] U. Özaydın, T. Georgiou, M. Lew. A Comparison of CNN and Classic Features for Image Retrieval. 2019 International Conference on Content-Based Multimedia Indexing(CBMI), 2019: 1-4.

[2] Zhang Zhanpeng, Luo Ping, C. C. Loy, et al. Facial landmark detection by deep multi-task learning. Computer Vision-ECCV 2014, 2014, 8694: 94-108.

[3] S. Verma, D. Ganesh Gopal, P. Kumar Sharma. An Optimized CNN and Transfer Learning Approach for Pneumonia Detection. 2023 3rd International Conference on Technological Advancements in Computational Sciences (ICTACS), 2023: 577-581.

[4] S. Colaco, D. S. Han. Facial Keypoint Detection with Convolutional Neural Networks. 2020 International Conference on Artificial Intelligence in Information and Communication (ICAIIC), 2020: 671-674.

[5] K. Simonyan, A. Zisserman. Very deep convolutional networks for large-scale image recognition. arXiv preprint, 2014: arXiv; 1409.1556.

[6] Tan Mingxing, Le Quoc. EfficientNet: Rethinking model scaling for convolutional neural networks. Proceedings of the 36th International Conference on Machine Learning, 2019, 97: 6105-6114.

[7] Xue Mingliang, Huang Ming, Yang Jingjing, et al. MICNet: A Hybrid Model Based on Inception Network and CNN for Automatic Modulation Classification. 2021 International Conference on Wireless Communications and Smart Grid

(ICWCSG), 2021: 331-335.
[8] Sun Yi, Wang Xiaogang, Tang Xiaoou. Deep convolutional network cascade for facial point detection. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2013: 3476-3483.
[9] C. Szegedy, V. Vanhoucke, S. Ioffe, et al. Rethinking the Inception Architecture for Computer Vision. Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR), 2016: 2818-2826.
[10] N. Joshi, D. Kumar, V. Kukreja, et al. Automated Tea Leaves Recognition: Multi-Classification Using YOLOv5 and Inception V3 Model. 2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT), 2023: 1-6.
[11] Wan Zhiqiang, Jin Wei. Design and Application of Scientific Research Project Management System based on Convolutional Neural Network and Stochastic Gradient Descent with Multilayer Perceptron. 2024 International Conference on Distributed Systems, Computer Networks and Cybersecurity (ICDSCNC), 2024: 1-5.
[12] Li Huijun, Ji Gang, Ma Zengliang. A Nonlinear Predictive Model Based on Multilayer Perceptron Network. 2007 IEEE International Conference on Automation and Logistics, 2007: 2686-2690.
[13] S. Colaco, D. S. Han. Facial Keypoint Detection with Convolutional Neural Networks. 2020 International Conference on Artificial Intelligence in Information and Communication (ICAIIC), 2020: 671-674.
[14] Diederik P. Kingma, Jimmy Ba. Adam: A method for stochastic optimization. arXiv preprint, 2014: arXiv: 1412.6980.