Information-Synthesis Network for Facial Landmarks Estimation

Bo Pu

Leiden University
Leiden, The Netherlands

Abstract. Most of the prior research approaches for facial landmarks estimation improve their accuracy by sacrificing the running performance. However, this paper presents a new approach for localizing the positions of facial landmarks precisely, which strikes a balance between the rate of accuracy and the efficiency of network. Since the commonly used Max-Pooling layer actually causes the information loss, our model concatenates the feature extracted on each convolutional layer toward reducing the loss of information in the network. The experimental results demonstrate that the accuracy of this network increases while the performance maintains computational efficiency since the loss of information is integrated in the concatenate layer. Furthermore, the weight of each convolutional layer is optimized to filter the information overlap of different layers while minimizing redundant computation.

Keywords: Convolutional Neural Network · Facial Landmarks · Information Overlap · Weight Optimization

1 Introduction

Facial landmarks extraction is a popular topic related to facial recognition and vision studies. Actually this problem is a classic multi-label regression problem being studied for years. With the development of availability of ever more comprehensive datasets and especially convolutional neural network recently, this task becomes more fascinated as the accuracy rate increasingly rises based on the framework of deep learning [19]. Existing approaches includes the three levels of coarse-to-fine network which cascades for the whole structure to predict landmarks [1] and the typical pipeline which localizes the landmarks before detecting the position of face [2]. The researches on related vision topics [17], such as face identification, benefit from the structure of deep convolutional network and other deep models. However, current researches focus on coarse-to-fine idea, which causes unnecessary computation to improve accuracy. Meanwhile few studies are made to maintain running performance while enhance accuracy. It is worthwhile to conduct related researches on the strategy of balancing the accuracy and efficiency.

A common characteristic of convolutional neural network is that they make use of pooling layers to progressively reduce the computation. The pooling layer is essential to the network as it brings about invariance of translations, and moreover provides good outcomes and performances. However the loss of image information continues growing as the depth of network rises, and there is few study to impede the decrease

adfa, p. 1, 2011.

of accuracy and benefit from the increase of efficiency of network caused by pooling layer. Generally the invariance and efficiency achieved by pooling layers comes at the price of limiting estimation accuracy. As such, by concatenating the extracted feature produced from previous convolutional layers together before inter-product layer, for landmark task a trade-off is made between accuracy and network performance.

In this paper we propose an Information-Synthesis architecture for precise localization of facial landmarks in RGB images without significant computational overhead. This model allows us to utilize as many as pooling layers for computational efficiency, while maintaining high localization precision. More importantly it presents a strategy to balance the accuracy and efficiency of convolutional neural network.

2 Related Work

Deep models, like Convolutional Neural Networks (CNN), Restricted Boltzmann Machines (RBMs), now play an essential role to advance the progress of computer vision, such as precisely landmarks localization [9], image classification [10,17] and multi-object detection [11]. When it comes to the big datasets and deep network structures, previous researches have no advantage to extract the complicated relationship between spatial information and facial landmarks.

Facial landmarks estimation has been highly developed with deep models in recent years. Unlike previous texture-based approaches [12] that find each facial landmark independently which ignores learning facial shape and hand-crafted features which have poor generalization performance, Most of the research works are based on deep regression model, which take the whole image as input to predict all landmarks and extract features progressively [13].

In [2], the authors propose a three-level structure network. Networks at the second and third levels adjust the initial prediction at the first level to achieve high accuracy. Another paper [7] presented similar four level convolutional network cascaded. Each network level refine a subset of facial landmarks generated in previous network levels. Furthermore, Dong [8] proposes an Adaptive Cascade Deep Convolutional Neural Networks (ACDCNN), which refines each landmark after initializing the shape by CNN model. Rather than in the well segmented face images, Zuijin Liang [6] investigates a novel Backbone-Braches Fully-Convolutional Neural Network (BB-FCN) in unconstrained environment, which roughly detects the locations of facial landmarks and the branches further refine the localizations.

Most of works mentioned emphasize the coarse-to-fine idea while other design much deeper network [4] to increase accuracy. However, without balancing the accuracy and performance, these works achieves higher accuracy at the sacrifice of performance, which costs amounts of time to do computation on the deep network. Therefore, Jonathan Thompson [5] takes the characteristic of different layers in CNNs into consideration, which made a trade-off between generalization performance and accuracy on object localization by overcoming the limitations of pooling to improve the precision. Eigen [14] predicts depth by using a cascade of CNN models in his paper which discusses the limitation of different layers and suggests joint training in shared-feature architecture can improve the generalization performance. Few study synthesizes the information of upper layers to overcome the limitation of pooling layer.

3 The Proposed Information-Synthesis Network

3.1 Task Description

The traditional task is localizing all the facial landmarks on given pictures precisely. While our model aims at minimizing the distance between ground truth and output by balancing the running performance and accuracy.

3.2 Framework

The Information-Synthesis network we designed is quite different from traditional CNNs [1]. Figure 1 shows the overview of the model using Caffe [3].

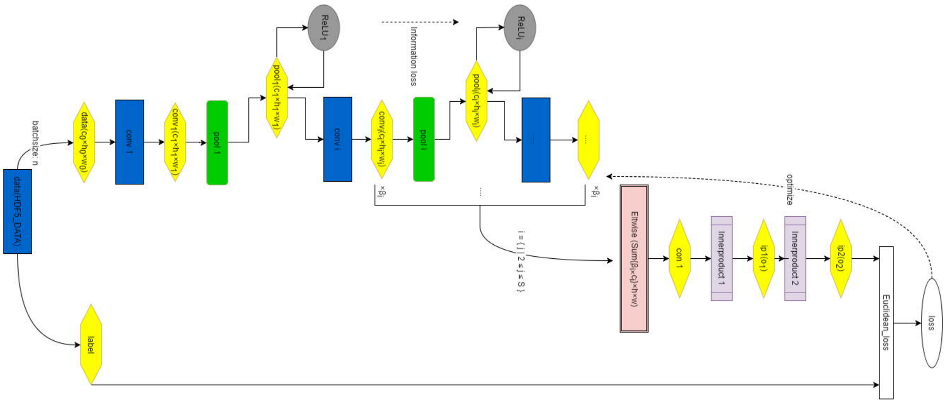
data(HDF5_DATA)
batchsize: n
data($c_0 \times h_0 \times w_0$)
conv 1
conv1($c_1 \times h_1 \times w_1$)
pool 1
pool1($c_1 \times h_1 \times w_1$)
ReLU1
conv i
convi($c_i \times h_i \times w_i$)
pool i
Information loss
pooli($c_i \times h_i \times w_i$)
ReLUi
....
$\times\beta_i$
$\times\beta_i$
$i = \{ j \mid 2 \le j \le S \}$
Eltwise (Sum{$\beta_i \times c_i$}×h×w)
con 1
Innerproduct 1
ip1(o1)
Innerproduct 2
ip2(o2)
optimize
Euclidean_loss
loss
label

Fig. 1. Information-Synthesis Network Framework

The building blocks of this network are consisted of a stack of different layers: convolutional, pooling and ReLU layer. It traditionally can duplicate to increase the depth of network. However the information loss can rise as the depth grows. The essence of the Information-Synthesis network is the concatenate layer which synthesizes the information from each building block. Considering the information can be overlapping across the blocks. The feature map c_i produced by convolutional layer is assigned with weight β_i, where the range is $0 \le \beta_i \le 1$ and $2 \le i \le S$, to adjust the percentage of features we utilize. Noted that the number of building blocks is S. Hence the feature map in concatenate layer is defined as follows:

$$c = \sum_{i=2}^{S} (\beta_i \times c_i) \tag{1}$$

3.3 Weights Optimization

Table 1. The presudocode of weight optimization in details

Algorithm: Weight Optimization

Require: A set of experiments which the weights are initialized randomly are conducted to collect data set $\mathrm{D}_n = ((\hat{\beta}, \hat{y})_1, ..., (\hat{\beta}, \hat{y})_i, ..., (\hat{\beta}, \hat{y})_n)$ as input.

```
$\varphi \leftarrow \varphi_r$
for $p = 1, 2, 3, ...$ do
    while $\hat{f}_{\min} - f_{\min} \geq \varphi$ do
        $\hat{f}(\vec{k}, \vec{\beta}) \leftarrow \sum_0^p \sum_2^S k \times \beta^p$
        for $t = 1, 2, 3, ...$ do
            $\vec{k}_t \leftarrow V_{\text{regression}}$
            for each $(\hat{\beta}, \hat{y})$ in $\mathrm{D}_n$ do
                minimize error $\sum e^2$ where $e = \{\hat{f}(k_t, \hat{\beta}) - \hat{y}\}$
            end for
        end for
        $\hat{f}_{\min} = \min f$ when $\beta_p \leftarrow \hat{\beta}_{\min}$ under constraints $\mathrm{R}(\beta)$
        $f_{\min} \leftarrow loss(\beta_p)$ where $loss$ is the IS network
    end while
end for
```

Assigning weights manually is impossible as we cannot distinguish which part of features is replicated in the previous building blocks. Unlike the existed approaches that concatenate the different building blocks all together, the efficient method is to optimize the assigned weights to minimize the computation of redundant information.

$$\vec{\beta} = \arg\min_{\vec{\beta}}\{loss(\vec{\beta}, \vec{c})\} \tag{2}$$

Where weights $\vec{\beta} = (\beta_2, \beta_3, ..., \beta_i, ..., \beta_S)$ subject to the constraint $\sum_2^S \beta \leq 1$, which means the time for updating parameters is controlled to guarantee its efficiency.

Because of the complexity of information overlapping across different layers, determining the weights to minimize the loss of information is a tough problem. However, regression analysis [15] can address the issue succinctly by forecasting the value of weights when the output of network is minimum. The pipeline of optimizing weights is defined as pseudo code in Table. 1.

Obviously the algorithm assists us on measurement of information overlap of different convolutional layers.

4 Implementation

Caffe is utilized to implement the details of convolutional neural networks since its better operability. The experiment is set on GPU with CUDA installed in an Ubuntu server. We will release the detail of our model after acceptance.

5 Experiment and Comparison

5.1 Facial Landmarks Dataset

The training dataset contains 5,590 LFW [16] images and 7,876 other images downloaded from the web. Meanwhile the test dataset contains 1,521 BioID images, 781 LFPW training images. These dataset and validation set are defined in the files and we use the face detector [2] to detect the face bounding box for pre-processing.

Since the size of training set is small, it would fall into a local optimum after 10000 iterations. The data augmentation is one of strategies for the network avoiding over-fitting and improving generalization [19]. The strategies of data augmentation include image rotation, translation, flip horizontally and changing the contrast and brightness of the images.

5.2 Evaluation Metric

The evaluation metric is the sum of squares of differences of its two inputs, which is defined in Caffe as the EuclideanLoss layer.

$$\frac{1}{2n}\sum_{i=1}^{n} \quad x_i^1 - x_i^2 \quad {}^{2} \tag{3}$$

5.3 Results

5.3.1 Weight Optimization

The relationship between weights and output is nonlinear. The best policy is to fit the function based on the data obtained from experiements. The form of function $f(\beta)$ is determined: multivariate second degree polynomial function.

Table 2. Output of network after 10000 interations

param experiment	$\hat{\beta}_2$	$\hat{\beta}_3$	$\hat{\beta}_4$	$\hat{f}(\beta)$
1	0.1	0.4	0.9	28.3134
2	0.0	0.0	1.0	30.1515
3	0.0	1.0	1.0	27.4737
4	1.0	1.0	1.0	24.3263
5	0.0	1.0	0.0	31.4732
6	1.0	0.0	0.0	36.0575
...	...	...	...	...

15	0.1	0.3	0.6	28.3707

After that the parameters of function $\vec{k}$ can be fitted through regression algorithm.

$$f = 36.6191 - 15.9662\beta_2 - 1.6210\beta_3 - 13.9947\beta_4 + 14.6877\beta_2^2 - 2.9474\beta_3^2 + 8.2006\beta_4^2 \quad (4)$$

The minimum of this function theoretically is calculated: $f_{\min} = 27.1380$, when the value of $\vec{\beta}$ is (0.4014, 0, 0.5986). Subsequently the value is brought into the network and the output is 27.6405, which put an end to the algorithm as $\left|\hat{f}_{\min} - f_{\min}\right| < \varphi_r$.

5.3.2 Comparison

We compare the results of two networks: the architecture we design with the weights optimized and the general convolutional neural network of Deepface [1]. In order to illustrate the effect of the concatenate layer which synthesis the image information to the accuracy of network. We set the same parameters on each layer and learning algorithm. The outputs of the two networks list in Table. 2.

Table 3. The results of output after 80000 interations

	Our Model	General CNN [1]
Train phase	4.6524	6.5457
Test phase	5.6340	7.0647

The values of loss both in training phase and test phase are showed in Fig. 2.

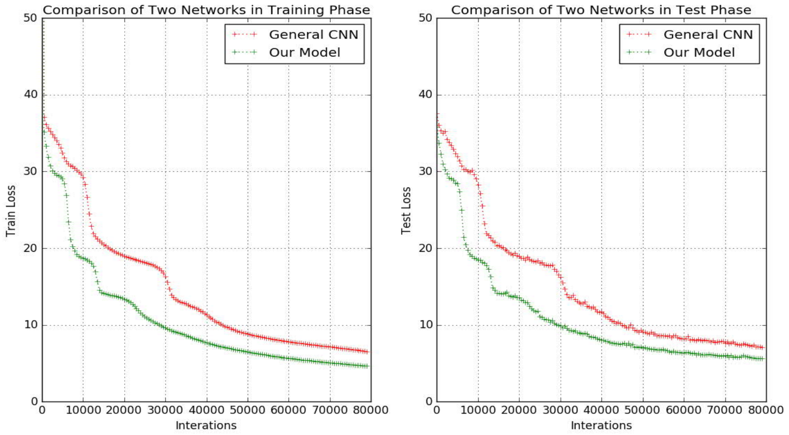

Fig. 2. The curves of two networks in training phase and test phase. The general one [1] has difficulty in fast convergence and not as accurate as our model when the running time is nearly same. Our model achieves higher accuracy while maintaining good performance.

The loss of information caused by pooling layer actually leads to inaccurate features which influence the network producing satisfied results. Amounts of running time are wasted without obtaining higher accuracy. The image information on different layers can cause overlapping. Nevertheless, it actually maximizes the rate of information utilization by searching the optimized weights to avoid information overlap. As the results are showed in Fig.3, both networks have similar running performance while the former possesses higher accuracy. The appropriate information extracted from previous convolutional layers can indeed assist the network locate landmarks precisely.

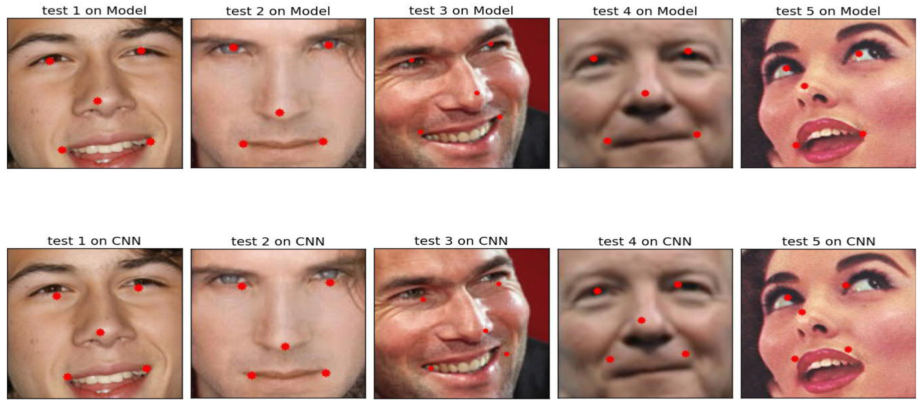

Fig. 3. Results comparison between our model and general CNN model.

6 Conclusion and Future Work

In this work, we proposed a model for synthesizing image information from different convolutional layers. And we have successfully applied our model on the face landmarks estimation and compared it with Deepface [1]. It turns out that our model achieves higher accuracy and maintains the same running performance in the experiments. Therefore, we have shown that the precision lost due to pooling layers can be recovered efficiently from different layers without a loss in computational performance. In future work, we will investigate different methods for determining the weights of each convolutional layer.

References

1. Taigman, Yaniv, et al. "Deepface: Closing the gap to human-level performance in face verification." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2014.
2. Sun, Yi, Xiaogang Wang, and Xiaoou Tang. "Deep convolutional network cascade for facial point detection." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2013.
3. De Bie, Tijl, et al. "CAFE: a computational tool for the study of gene family evolution." *Bioinformatics* 22.10 (2006): 1269-1271.
4. Szegedy, Christian, et al. "Going deeper with convolutions." Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2015.
5. Tompson, Jonathan, et al. "Efficient object localization using convolutional networks." *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*. 2015.

6. Liang, Zhujin, Shengyong Ding, and Liang Lin. "Unconstrained Facial Landmark Localization with Backbone-Branches Fully-Convolutional Networks." *arXiv preprint arXiv:1507.03409* (2015).
7. Zhou, Erjin, et al. "Extensive facial landmark localization with coarse-to-fine convolutional network cascade." *Proceedings of the IEEE International Conference on Computer Vision Workshops*. 2013.
8. Dong, Yuan, and Yue Wu. "Adaptive Cascade Deep Convolutional Neural Networks for face alignment." *Computer Standards & Interfaces* 42 (2015): 105-112.
9. Ranjan, Rajeev, Vishal M. Patel, and Rama Chellappa. "HyperFace: A Deep Multi-task Learning Framework for Face Detection, Landmark Localization, Pose Estimation, and Gender Recognition." *arXiv preprint arXiv:1603.01249*(2016).
10. Krizhevsky, Alex, and G. Hinton. "Convolutional deep belief networks on cifar-10." *Unpublished manuscript* 40 (2010).
11. Girshick, Ross, et al. "Rich feature hierarchies for accurate object detection and semantic segmentation." *Proceedings of the IEEE conference on computer vision and pattern recognition*. 2014.
12. Dabov, Kostadin, et al. "Image denoising with block-matching and 3D filtering." *Electronic Imaging 2006*. International Society for Optics and Photonics, 2006.
13. Wu, Yue, and Qiang Ji. "Discriminative deep face shape model for facial point detection." *International Journal of Computer Vision* 113.1 (2015): 37-53.
14. Eigen, David, Christian Puhrsch, and Rob Fergus. "Depth map prediction from a single image using a multi-scale deep network." *Advances in neural information processing systems*. 2014.
15. Opfermann, Johannes. "Kinetic analysis using multivariate non-linear regression. I. Basic concepts." *Journal of thermal analysis and calorimetry*60.2 (2000): 641-658.
16. Zhu, Xiangxin, and Deva Ramanan. "Face detection, pose estimation, and landmark localization in the wild." *Computer Vision and Pattern Recognition (CVPR), 2012 IEEE Conference on*. IEEE, 2012.
17. Huiskes, Mark J., and Michael S. Lew. "Performance evaluation of relevance feedback methods." In Proceedings of the 2008 international conference on Content-based image and video retrieval, pp. 239-248. ACM, 2008.
18. Zhang, Zhanpeng, et al. "Facial landmark detection by deep multi-task learning." *Computer Vision–ECCV 2014*. Springer International Publishing, 2014. 94-108.
19. Cui, Xiaodong, Vaibhava Goel, and Brian Kingsbury. "Data augmentation for deep neural network acoustic modeling." *IEEE/ACM Transactions on Audio, Speech and Language Processing (TASLP)* 23.9 (2015): 1469-1477.