FACIAL POINT DETECTION BASED ON A CONVOLUTIONAL NEURAL NETWORK WITH OPTIMAL MINI-BATCH PROCEDURE

*Masatoshi Kimura Takayoshi Yamashita Yuji Yamauchi Hironobu Fujiyoshi**

Chubu University
1200, Matsumoto-cho, Kasugai, AICHI

ABSTRACT

We propose a Convolutional Neural Network (CNN)-based method to ensure both robustness to variations in facial pose and real-time processing. Although the robustness of CNNs has attracted attention in various fields, the training process suffers from difficulties in parameter setting and the manner in which training samples are provided. We demonstrate a manner of providing samples that results in a better network. We consider four methods: 1) subset with augmentation, 2) random selection, 3) fixed-person subset, and 4) the conventional approach. Experimental results indicate that the subset with augmentation technique has sufficient variations and quantity to obtain the best performance. Our CNN-based method is robust under facial pose variations, and achieves better performance. In addition, since our networks structure is simple, processing takes approximately 10ms for one face on a standard CPU.

Index Terms— convolutional neural network, mini-batch, facial points, random selection

1. INTRODUCTION

Facial point detection is an active area of research in computer vision, and is an essential preprocessing step for applications such as face recognition and facial expression estimation. These applications require accurate detection of facial feature points, even if facial images have been taken with various poses, lighting conditions, expressions, and occlusions.

Researchers have tackled various facial detection tasks under these difficult conditions. The main approaches can be divided into two categories: classification methods [2][9][20] and direct prediction methods [3][5][6][11][15][17]. Classification methods extract candidate regions using local sliding windows. The optimal points are then estimated from these candidates using shape constraints [2][11][20]. Prediction methods employ a regressor to detect facial points from the whole face region without scanning. The positions of facial points are iteratively updated until convergence is achieved. Recently, classification and prediction methods have been combined in a coarse-to-fine framework to improve accuracy [15][19][14]. In this framework, the initial positions of facial points are first predicted, and the fine positions of facial points are then estimated. Many approaches must determine what type of feature representation to employ. Together with shape information, appearance information is important in detecting facial parts, and yet this feature type is underspecified.

In this work, we present a convolutional neural network (CNN)-based approach to overcome the above issues in the detection of facial points. The robustness of CNNs has attracted attention in various fields. We investigate the robustness of CNNs against shape and appearance changes such as facial pose variations. In addition, we demonstrate the best gradient descent training strategy with a subset called the mini-batch. A CNN is constructed by a huge number of parameters that are trained with backpropagation and then gradually updated. During the update process, the CNN receives small subsets of training samples from a large dataset. When the parameters of CNN have been updated using all samples, they are updated again with the first subset of samples. The manner in which the subsets are provided is critical to the training process. We demonstrate the best manner of providing subsets to obtain efficient parameters. In the next section, we introduce some related work and describe the proposed method. We then evaluate the manner of providing subsets of data for the training process, and compare the performance of the proposed method with that of the conventional method using a public dataset.

2. RELATED WORK

Facial point detection is an important preprocessing step for face recognition and facial analysis. Active shape models and the active appearance model, which simulate the holistic appearance or shape, are representative methods applied in early studies [5][4]. Taking a classification approach, Vukadinovic et al. trained the detectors of each facial point independently using Gentle Boost and Gabor filters [18], and Amberg et al. employed a branch-and-bound algorithm to find optimal configurations from a large number of candidates given by component detectors from the whole image [1]. These methods are limited in finding facial part regions when there are large variations in appearance. Uricar et al. proposed a method based on the deformable part model and structure-output sup-

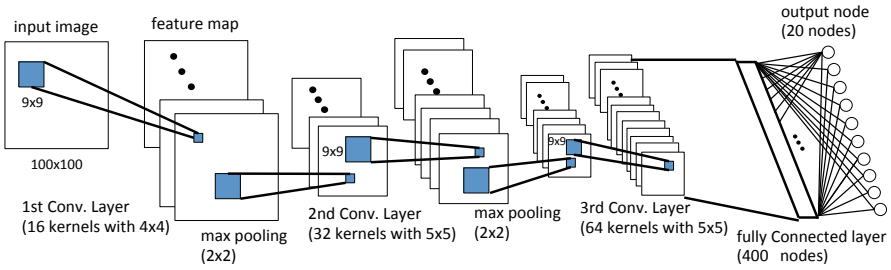

Fig. 1. Structure of Convolutional Neural Networks

port vector machines [13], whereas Belhumerur et al. developed a Bayesian model combining the outputs of local detectors with a consensus-based non-parametric global model for part locations [2]. This model provides high accuracy, but requires high-resolution images.

Regression-based methods are at the forefront of research. Valstar et al. employed support vector regression and a conditional Markov random field to obtain global consistency [17]. Cao et al. proposed a method based on the regression of random ferns that receive the whole face region as input [3]. The Conditional Regression Forest (CRF) proposed by Dantone et al. detects facial points using regression forests for each face pose [6]. CRF consists of two stages: the first estimates the facial pose, and the second regresses the facial points using regression forests. The performance of facial point detection depends on the regression forests for facial pose estimation. These methods either lack flexibility in pose variation, or incorrectly detect the frames of eyeglasses.

Deep CNNs exhibit a performance level similar to that of human experts [10]. Krizhevsky applied deep CNNs to an object recognition benchmark to classify 1000 different classes, and achieved good performance [7]. The advantage of ConvNets is that it is able to extract complex and suitable features for the task. This reduces the burden of designing features, because the entire system is trained from raw pixels. Sun et al. proposed a method based on CNNs that cascade from the whole facial region to local regions [16]. Although they achieved state-of-the art performance, their method has a complex structure and is prone to incorrect detections in the presence of accessories such as eyeglasses.

Many conventional methods are unable to achieve both robustness for face pose variation and real-time processing. In this paper, we propose a CNN-based method to tackle the issue of face pose variation, and employ a simple structure to enable real-time processing.

3. PROPOSED METHOD

We propose a CNN-based method to overcome the problem of robustness to face pose variation and allow real-time processing. In addition, we demonstrate a gradient descent training strategy using image subsets. During the update process, CNN receives small subsets of the large training dataset. We consider which mini-batch method provides these subsets in the most efficient way.

3.1. Convolutional Neural Network

As shown in Fig. 1, CNNs consist of an alternate succession of convolutional layers and subsampling layers. There are several types of layers, including input layers, convolutional layers, pooling layers, and classification layers. Besides the raw data, each input layer also includes edge and normalized data as its input. The convolutional layer has M kernels of size $Kx \times Ky$, and these are filtered in order to input data. The filtered responses from all the input data are then subsampled in the pooling layer. Max pooling can output the maximum value in certain regions, such as an area of 2×2 pixels. The convolutional layer and pooling layer are laid alternately to create the deep network architecture. Finally, the output feature vectors from the last pooling layer are used in the regression layer. Whereas conventional CNNs employ a classification layer to output the probability of each class, our method employs a regression layer that outputs coordinates. That is, the output node corresponds to the (x, y) coordinates of each facial point.

CNNs require a supervised learning process in which the filters are randomly initialized and updated through backpropagation [12]. Backpropagation uses the function shown in Eq. (1) to estimate the connected weights that minimize E using the gradient descent method in Eq. (2).

$$E = \frac{1}{2}\sum_{p=1}^{P} E_p \tag{1}$$

$$w_{ji}^{(l)} \leftarrow w_{ji}^{(l)} + \Delta w_{ji}^{(l)} = w_{ji}^{(l)} - \lambda \frac{\partial E_p}{\partial w_{ji}^{(l)}} \tag{2}$$

Note that $\{p|1, ..., P\}$ is the training sample, $\boldsymbol{o_p}$ is the corresponding value of training sample p in the output layer, and $\boldsymbol{t_p}$ is the label data of p. λ is the training ratio, and $w_{ji}^{(l)}$ is the weight that connects node i in layer l to node j in the next layer. The error for each training sample E_p is the sum of the differences between the output value and the label. $\Delta w_{ji}^{(l)}$ is represented as:

$$\Delta w_{ji}^{(l)} = -\lambda \delta_k^{(l)} y_j^{(l-1)} \tag{3}$$

Fig. 2. Process of mini-batch

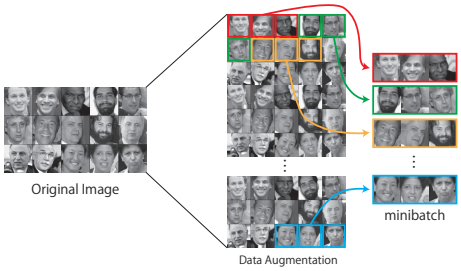

Fig. 3. The process of augmented mini-batch

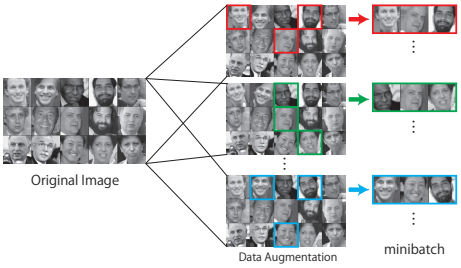

Fig. 4. Process of random mini-batch

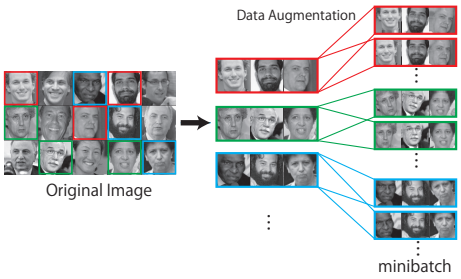

Fig. 5. The process of fixed-person mini-batch

$$\delta_k^{(l)} = e_k \phi(V_k^{(l)}) \tag{4}$$

$$V_k^{(l)} = \sum_j w_{kj}^{(l)} * y_j^{(l-1)} \tag{5}$$

$y_j^{(l-1)}$ is the output of node j in the $(l-1)$th layer and e_k is the error of node k. $V_k^{(l-1)}$ is the accumulated value connected to node k from all nodes in the $(l-1)$th layer. The local gradient descent is given by Eq. (4). The activation function ϕ can take various forms, such as sigmoid, ReLU[7] and Maxout[8]. The connected weights in the entire network are updated concurrently for a predetermined number of iterations, or until some convergence condition is satisfied.

When using backpropagation, there are many ways to calculate the error E, including *full-batch*, *online*, and *mini-batch*. *Full-batch* provides all of the training data at once. As such, it requires few iterations, but has poor convergence because of the increasing gradient descent. *Online* gives the training data iteratively. Hence, it obtains optimized results from its small gradient descent, but requires a considerable processing time to complete the multiple iterations. *Mini-batch* is a commonly used middle approach that updates the weights with small subsets of training data. It is able to effectively update connected weights with a huge amount of training data within a reasonable length of time.

3.2. Mini-batch process

To train the parameters of CNN, the samples are divided into small subsets and input to the CNN. This process is called mini-batch. Different subsets are provided on each iteration. When all samples have been used to update the parameters, the first subset is given to CNN again to complete an epoch. In the training process, divided subsets are used repeatedly until a specified performance is attained. Because the training process is based on gradient descent, the manner in which subsets are provided to the CNN is important to obtain better parameters. However, there is no knowledge of how the mini-batches are prepared. To investigate the influence of the preparation process, we demonstrate an effective means of obtaining better parameters.

First, we consider the number of samples using data augmentation (called aug. mini-batch). Data augmentation is a common strategy to increase the number of samples in a small dataset. This process uses shifting, rotation, and scaling to produce datasets with millions of samples from a few thousand samples. As shown in Fig. 3, aug. mini-batch chooses augmented images at random to divide the subsets. All subsets are utilized repeatedly.

Second, we consider the repetition procedure based on random selection (called random mini-batch). In random mini-batch (Fig. 4), subsets are randomly formed from samples in the dataset. Whereas training with random mini-batch uses the samples repeatedly during the training process, it does not use the same subset.

Third, we consider the selection of content for the mini-batch (called fixed-person mini-batch). Although aug. mini-batch contains different augmented samples in the subset, the same subset can be used repeatedly in each epoch. The fixed-person mini-batch consists of different original samples in the subset. As shown in Fig. 5, when a subset is used to train the CNN, the samples in the subset are augmented. As a result, although the original samples in the subset are the same, the content of the training samples is different in each epoch.

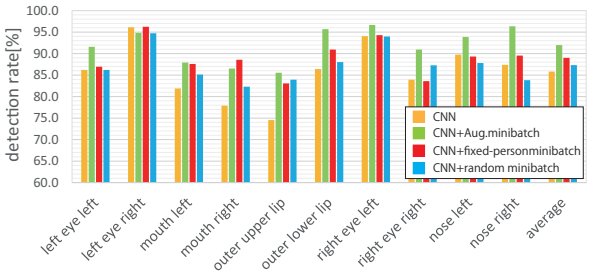

Fig. 6. Comparison between different mini-batch processes

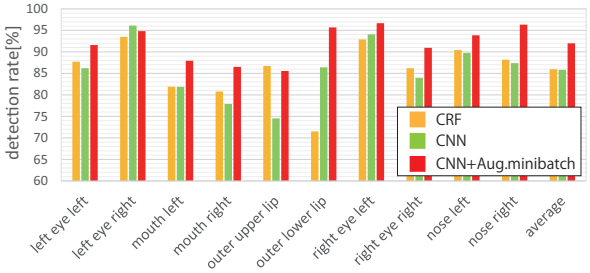

Fig. 7. Performance comparison between CNN and CRF

4. EXPERIMENTS

4.1. Performance comparison

We demonstrate the effectiveness of mini-batch and the robustness of the proposed method to pose variations. We use the Labeled Faces in the Wild (LFW) dataset, and attempt to detect 10 facial feature points—the left and right corners of both eyes, nose, and mouth, and the upper and lower points of the lip. LFW is a commonly used dataset representing an uncontrolled environment formed of images collected from the internet. Dantone annotated 10 facial points and face poses. The dataset contains 1500 samples for training and 927 samples for evaluation. We increase the number of training sample to 20000 by data augmentation. The data augmentation ranges are ± 10 pixels for translation and $\pm 15^\circ$ for rotation. The input images are 100×100 grayscale. As shown in Fig. 1, the CNN consists of five layers, with three convolutional layers, one fully connected layer, and one regression layer. The filter size is 9×9 pixels, and the convolutional layers have 16, 32, and 64 filters, respectively. We employ maxout as the activation function. In the fully connected layer, there are 400 nodes that have been trained using the dropout technique. The regression layer has 20 nodes, corresponding to the coordinates of facial points. The learning coefficient is 0.1, the batch size is 10, and the number of iterations is 300000.

As in previous researches[6], we evaluate the localization error as a fraction of the inter-ocular distance. This is invariant to the actual size of the images in terms of the inter-ocular distance. We declare that a point has been correctly detected if the pixel error is less than 10% of the inter-ocular distance.

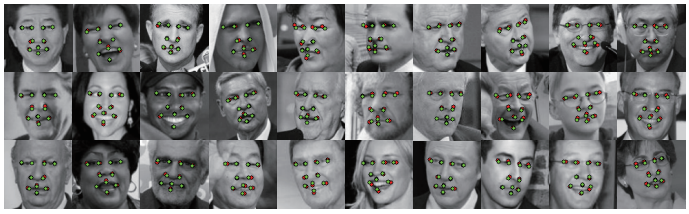

Fig. 8. Facial point detection results

4.2. Performance of different mini-batch processes

The evaluation results for each mini-batch process are shown in Fig. 6. Compared with the conventional CNN (without data augmentation), all mini-batch strategies produce improved detection performance. Although random mini-batch uses large subsets that contain differently augmented samples and combinations, it is comparatively weak compared with aug. mini-batch and fixed-person mini-batch. This indicates that the CNN requires an appropriate amount of samples, rather than simply a large quantity.

Aug. mini-batch gives the best performance of the mini-batch methods. It improves the detection accuracy by 4% over random mini-batch and 2% over fixed-person mini-batch. As aug. mini-batch chooses samples from the augmented dataset, the combination of people in each subset is different. In contrast, the combination of people in the fixed-person mini-batch is the same. Accordingly, there is a slight difference between the various mini-batch procedures. This suggests that the CNN requires abundant variation in the subset contents. This in turn implies that aug. mini-batch has sufficient variation and quantity.

A comparison between CRF and CNN is shown in Fig. 7. Whereas the performance of conventional CNN without augmentation is slightly better than that of CRF, CNN with augmentation achieves an average 6% improvement. Furthermore, the detection rate of the lower mouth point reaches 96%, compared to just 72% with CRF. Some examples of the detection results are shown in Fig. 8. Our method detects facial feature points (green points) that are close to the ground truth (red points). Our approach achieves high accuracy, even with different facial poses. In addition, the processing speed of our method on C++ implementation takes 10ms to one image on 3.4GHz CPU.

5. CONCLUSION

We have proposed a CNN-based facial feature point detection method that is robust to variations in facial pose. We have also considered various mini-batch procedures to train efficient CNN parameters. Our experimental results suggest that aug. mini-batch is the best way to obtain both sufficient variation and quantity in the training samples. The proposed method achieved better performance than a CRF-based method, and was shown to be robust to face pose variations.

6. REFERENCES

[1] B. Amberg and T. Vetter. Optimal landmark detection using shape models and branch and bound. In ICCV, 2011.

[2] P. N. Belhumeur, D. W. Jacobs, D. J. Kriegman, and N. Kumar. Localizing parts of faces using an consensus of exemplars. In CVPR, 2011.

[3] X. Cao, Y. Wei, F. Wen, and J. Sun. Face alignment by explicit shape regression. In CVPR, 2012.

[4] T. F. Cootes, C. J. Taylor, D. H. Cooper, and J. Graham. Active shape models -their training and application. Computer vision and image understanding, 1995.

[5] T. F. Cootes, G. J. Edwards, and C. J. Taylor. Active appearance models. In ECCV, 1998.

[6] M. Dantone, J. Gall, G. Fanelli, and L. V. Gool. Real-time facial feature detection using conditional regression forests. In CVPR, 2012.

[7] A. Krizhevsky, I. Sutskever, and G. Hinton. magenet classification with deep convolutional neural networks. In NIPS, 2012.

[8] I. Goodfellow, D. Warde-Farley, M. Mirza, A. Courville, and Y. Bengio. Maxout networks. arXiv preprint arXiv:1302.4389, 2013.

[9] L. Liang, R. Xiao, F. Wen, and J. Sun. Face alignment via component based discriminative search. In ECCVV, 2008.

[10] Y. LeCun, L. Bottou, Y. Bengio, and P.Haffner. Gradient-based learning applied to document recognition. In IEEE, 1998.

[11] X. Liu. Generic face alignment using boosted appearance model. In CVPR, 2007.

[12] D. E. Rumelhart, G. E. Hinton, and R. J. Williams. Learning internal representations by error propagation. In Proc. Explorations in the Microstructures of Cognition, 1986.

[13] M, Uricar, V. Faranc, and V. Hlavac. Detector of facial landmarks learned by the structure output sum. In VISAPP, 2012.

[14] S, Ren, X. Cao, Y. Wei, J. Sun. Face Alignment at 30000 FPS via Regressing Local Binary Features. In CVPR, 2014.

[15] P. Sauer, T. Cootes, and C. Taylor. Accurate regression procedures for active appearance models. In BMVC, 2011.

[16] Y. Sun, X. Wang, and X. Tang. Deep Convolutional Network Cascade for Facial Point Detection. In CVPR, 2013.

[17] M. Valstar, B. Martinez, X. Binefa, and M. Pantic. Facial point detection using regression and graph models. In CVPR, 2010.

[18] D. Vukadinovic, and M. Panic. Fully automatic facial feature point detection using gabor feature based boosted classifiers. In ICSMC, 2005.

[19] X. Yu, J. Huang, S. Zhang, W. Yan, and D. Metaxas. Pose-free Facial Landmark Fitting via Optimized Parts Mixture and Cascaded Deformable Shape Model. In ICCV, 2013.

[20] X. Zhu, and D. Ramanan. Face detection, pose estimation, and landmark localization in the wild. In CVPR, 2012.