applied sciences

MDPI

Article

Augmented EMTCNN: A Fast and Accurate Facial Landmark Detection Network †

Hyeon-Woo Kim [1], Hyung-Joon Kim [1], Seungmin Rho [2] and Eenjun Hwang [1,*]

[1] School of Electrical Engineering, Korea University, 145 Anam-ro, Seongbuk-gu, Seoul 02841, Korea; guihon12@korea.ac.kr (H.-W.K.); hyungjun89@korea.ac.kr (H.-J.K.)
[2] Department of Software, Sejong University, Seoul 05006, Korea; smrho@sejong.edu
* Correspondence: ehwang04@korea.ac.kr; Tel.: +82-2-3290-3256
† This paper is an extended version of our paper published in Proceedings of the 2019 IEEE International Conference on Big Data and Smart Computing (BigComp), Kyoto, Japan, 27 February–2 March 2019.

Received: 20 February 2020; Accepted: 24 March 2020; Published: 26 March 2020

Abstract: Facial landmarks represent prominent feature points on the face that can be used as anchor points in many face-related tasks. So far, a lot of research has been done with the aim of achieving efficient extraction of landmarks from facial images. Employing a large number of feature points for landmark detection and tracking usually requires excessive processing time. On the contrary, relying on too few feature points cannot accurately represent diverse landmark properties, such as shape. To extract the 68 most popular facial landmark points efficiently, in our previous study, we proposed a model called EMTCNN that extended the multi-task cascaded convolutional neural network for real-time face landmark detection. To improve the detection accuracy, in this study, we augment the EMTCNN model by using two convolution techniques—dilated convolution and CoordConv. The former makes it possible to increase the filter size without a significant increase in computation time. The latter enables the spatial coordinate information of landmarks to be reflected in the model. We demonstrate that our model can improve the detection accuracy while maintaining the processing speed.

Keywords: facial landmark extraction; convolutional neural networks; cascaded structure; face detection

1. Introduction

Facial landmarks such as eyes, nose, and mouth are prominent feature points on the face, and diverse tasks such as face recognition, gaze detection, person tracking, emotion recognition, and virtual makeup have been performed based on facial landmarks [1,2]. In particular, to meet the diverse demands of tasks such as real-time processing or rendering on mobile devices, fast and accurate face landmark extraction is essential [3]. So far, plenty of research has been done on extracting facial landmarks. In recent years, as convolutional neural networks (CNNs) have shown overwhelmingly strong performance in the field of image classification [4] and object detection [5], they have been investigated for applications in facial landmark extraction. Facial landmarks are popularly represented by 68 points, which cover facial contours, eyes, eyebrows, nose, and mouth [6–8]. In an effort to detect such facial landmark points accurately, adding more convolution layers has been attempted, as in Visual Geometry Group Network (VGGNet) [9,10]. Even though this produces better results, it requires more computational resources and is therefore not appropriate for real-time processing.

Various deep learning models have also been proposed for real-time extraction of facial landmarks [11–13]. Such models usually use fewer than 10 points for real-time facial landmark extraction. However, such few facial landmark points fail to represent landmark properties such as shape accurately. In the previous work, to extract a sufficient number of facial landmark points in real

time, we proposed an EMTCNN model by extending the original multi-task cascaded convolutional neural network (MTCNN) model [12] which extracts five facial landmark points in real time. By making the CNN layers of the MTCNN model deeper, the EMTCNN model could extract 68 facial landmark points in real time. Even though the accuracy of landmark detection can be improved further by making the output network (O-Net) of the EMTCNN model even more in-depth, the processing time would increase sharply owing to the increased number of parameters. Hence, in this study, we augment the EMTCNN model using two convolution techniques—dilated convolution [14–16] and CoordConv [17]—to improve the detection accuracy while maintaining the processing speed. The former makes it possible to extend the receptive field without increasing the number of parameters. The latter allows the feature map to reflect the spatial coordinate information of facial landmarks. In addition, to make our model more robust, we construct a face image dataset based on open face data and other face data collected manually and augment it by using image operations such as flipping and illumination to retain diversity of the subject race, age, and gender, face posture, and image shooting environment.

Through various experiments, we demonstrate that our proposed model can improve detection accuracy at similar speeds. This paper is organized as follows: Section 2 introduces some related works and Section 3 describes how to extract the facial landmark points in real time. Experimental results for the proposed method are described in Section 4, and Section 5 concludes the paper.

2. Related Works

Traditional real-time extraction methods in the fields of human images have been proposed in many research studies [18–21]. However, these conventional approaches have the drawback of using hand-crafted features. Recently, CNNs were demonstrated to be superior in the diverse fields of computer vision, energy forecasting [22,23], and biomedical science [24–28]. This is because it performs overwhelmingly well in feature map extraction from the convolution layers. Various studies on the extraction of high-quality feature maps based on the CNN architecture are underway [14–17]. The quality of facial landmark extraction has also been improved remarkably by using CNNs. However, in the case of high-resolution images, performing facial landmark extraction on the entire image requires a large amount of computation, and the results could be inaccurate. To alleviate such problems, the landmark extraction process can be divided into two stages: detecting facial regions and extracting facial landmarks in those detected regions.

A facial region in an image can be detected by using an object recognition technique, and many object detection algorithms have been proposed to localize objects within the image. For instance, Region CNN (R-CNN) [29], which is the first CNN-based object recognition model, uses the selective search algorithm [30] and creates bounding boxes by combining pixels that have similar feature patterns, then classifies images using CNN and support vector machine (SVM) [31]. However, this method requires a considerable amount of processing time because all the bounding boxes are fed to the CNN as input. To solve this problem, Ren [32] proposed the fast R-CNN based on the concept of region of interest (ROI) pooling. Instead of feeding all the bounding boxes to the CNN, they pooled the regions corresponding to ROIs from the final feature map of the CNN. Both R-CNN and fast R-CNN use the selective search algorithm to detect bounding box candidates. However, as selective search is a slow and time-consuming process, it affects the performance of the network. To rectify this, Ren et al. [33] proposed faster R-CNN to model ROI proposals of the region proposal network (RPN). They used a sliding window to compute the coordinates and scores of the bounding boxes. On the other hand, Redmon [34] proposed You Only Look Once (YOLO), which divides the image into N × N grids and computes the class probability and offset value along with the bounding boxes on the grid. Then, they selected the bounding boxes that had a higher class probability than some threshold value. Even though YOLO could detect object regions very quickly compared to other models, it had the problem of low accuracy. To improve the detection accuracy, Redmon et al. [35] added a batch normalization process to all the convolution layers, replaced the fully connected layer with a

convolution layer, and extended the final feature map size from 7×7 to 13×13. Through these attempts, they detected 9000 object classes while further improving the speed and performance, referred to as YOLO9000. In addition, they recently reported a new model that improves the performance further by introducing a new network [36].

As mentioned earlier, facial landmark extraction can be done in conjunction with face detection to reduce computation time and improve extraction accuracy. One popular approach for facial landmark detection is to use an open library such as Dlib [37]. This library has shown reasonable performance in face landmark recognition using the ensemble approach of regression trees proposed by Kazemi and Sullivan [38]. Another approach for facial landmark detection is to use CNN-based algorithms. Sun et al. [39] first suggested a method of using cascaded CNNs to extract facial landmarks. They obtained facial landmark points from the first network and then refined the results through shallow networks. Since then, many researchers have studied how to use CNNs to extract facial landmarks. Similar to the method proposed by Sun et al., MTCNN [12] uses relatively light CNNs to detect facial regions and extract five facial landmark points in real time. Ranjan et al. [40] proposed a multi-task CNN structure using combined features from multiple convolutional layers. In this way, face recognition, landmark localization, pose estimation, and gender recognition are jointly performed. Recently, recurrent neural networks (RNNs) have been widely used for refining the output of the main CNN to improve facial landmark extraction performance [41,42]. On the other hand, semantic segmentation can also be used for representing facial landmarks. For instance, Kim et al. [10] proposed a pixel-based facial landmark extraction scheme using SegNet [43,44]. SegNet consists of encoder and decoder networks. The encoder network performs convolution to down-sample the input image to the feature map. The decoder network performs deconvolution to up-sample the feature map to its original size. The final feature map is classified by a facial landmark class through the softmax function.

Generally, the quality of a feature map depends on the depth of the CNN model. The deeper the model, the higher quality of feature map can be obtained. However, as the depth of the model increases, the number of parameters increases exponentially, and the processing time increases accordingly. There has been plenty of research on how to extract features effectively without increasing the number of parameters. Chen [14–16] used dilated convolution instead of the existing convolution method for image semantic segmentation. This increases the convolution filter size based on a constant rate and fills the empty spaces between zeros. They effectively enlarged the field of view of the filters to incorporate the broader context without increasing the number of parameters or the amount of computation. Liu [17] proposed a convolution layer called CoordConv that has a channel containing coordinate information. The row and column information of the channel is normalized to the same size as the feature of each layer and concatenated to the input of the layer. Thus, the coordinate transformation problem was solved by the CoordConv layer with perfect generalization; it was 150 times faster with 10–100 times fewer parameters than convolution.

3. Materials and Methods

In this section, we describe how our model accurately detects 68 facial landmark points in real time. We constructed the EMTCNN model by extending the original MTCNN model in our previous work. To improve the detection accuracy further while maintaining the processing speed, we augment the EMTCNN model using two convolution techniques—dilated convolution and CoordConv.

3.1. Network Architecture

3.1.1. EMTCNN Augmentation

MTCNN is a cascaded structure composed of relatively light CNNs including a proposal network (P-Net), refinement network (R-Net), and output network (O-Net). Since MTCNN has a pyramid structure, it extracts face candidate regions from images at various scales and uses them as input images. This allows the model to learn different image scales effectively. The facial landmark extraction

process is divided into three steps; each step is performed on a different network. In the first step, the facial region candidates are extracted through the P-Net. Then, non-maximum suppression (NMS) [45] is used to remove highly overlapped candidates. In the second step, the filtered candidates are refined through the R-Net and NMS. In the last step, one facial region and its five facial landmark points are produced through the O-Net.

To extract the 68 most popular facial landmark points, instead of 5, we extend the O-Net of the MTCNN model by using two 3 × 3 convolution layers and increasing the number of convolution filters. This allows the second 3 × 3 convolution layer to see 5 x 5 regions with fewer parameters than using the 5 × 5 convolution layer [46]. Nevertheless, as the network expands, the number of parameters naturally increases, so there is a limit to expanding the network. To avoid this, we reduce the number of facial region candidates generated by the P-Net by ignoring small facial regions. Even though there is a trade-off between processing time and detection accuracy when manipulating the O-Net, there is still a limit to the extraction performance that can be achieved just by changing the parameter numbers while maintaining the real-time processing property. Hence, to improve the accuracy of real-time facial landmark point extraction without significantly increasing the number of parameters, we augment the EMTCNN model with two convolution techniques—dilated convolution and CoordConv. Figure 1 shows the structure of our augmented EMTCNN.

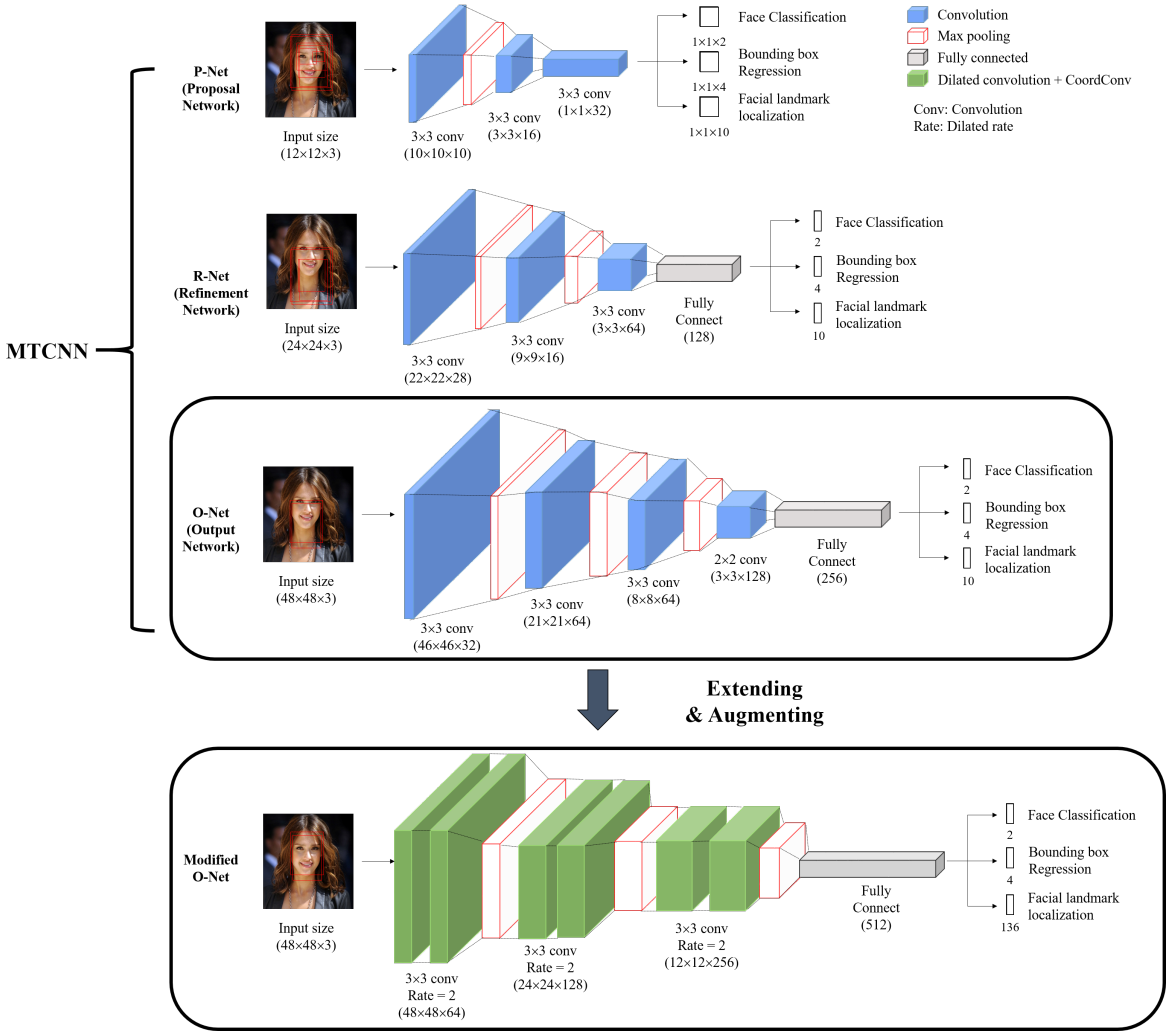


Figure 1. Augmented EMTCNN with modified O-Net.

Table 1 shows the number of parameters of major CNN models used for facial landmark extraction [40–42]. Although the numbers of parameters are increased a lot by expanding and augmenting the existing MTCNN, they are still small compared to those for other networks.

Table 1. Comparison of the number of parameters.

Model	Input Size	Number of Parameters
VGG-19 [47]	224 × 224	143,667,240
Hyper Face [40]	227 × 227	29,677,932
ResNet-18 [48]	224 × 224	11,689,512
MTCNN [12]	-	481,336
EMTCNN [2]	-	6,029,938
Augmented EMTCNN	-	6,083,186

VGG-19: Visual Geometry Group Network (VGGNet) with 19-layer. ResNet-18: Residual Neural Network with 18-layer.

3.1.2. Dilated Convolution

We expanded the convolution filter size of the O-Net to make the network consider larger image regions. However, there was a limit to the improvement in performance that could be achieved by expanding the convolution filter because the amount of computation that had to be carried out owing to the expansion increased dramatically. Dilated convolution, which is also known as Atrous convolution, makes it possible to expand the receptive field without increasing the size of the convolution filter. The receptive field represents the number of pixels in the original image that contain one pixel of the feature map. In other words, the larger the receptive field is, the more information the feature map contains. Figure 2a,b shows a general convolution filter and dilated convolution filter with rate equal to 2. In the dilated convolution filter, blanks between the weights are filled with zeroes. In this way, we can increase the filter size without any significant increase in the amount of computation required. As a result, even with the same number of parameters, we can achieve better accuracy by using dilated convolution. Hence, we use a 3 × 3 dilated convolution filter with rate equal to 2 instead of a general 3 × 3 convolution filter to extract facial landmarks more accurately without a significant increase in the number of parameters.

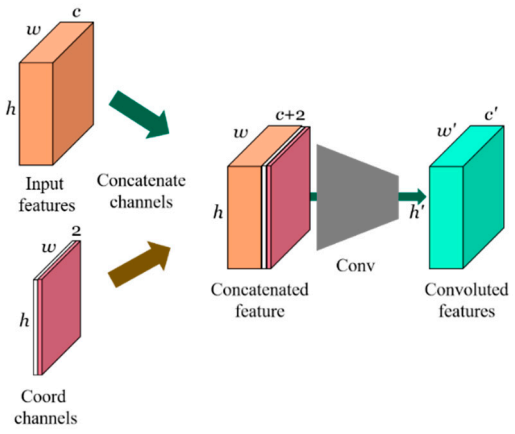
(**a**) General convolution (kernel 3 × 3/rate = 1 (**b**) Dilated convolution (kernel 3 × 3/rate = 2)

Figure 2. Dilated convolution.

3.1.3. CoordConv Layer

In general, the CNN consists of several convolution layers and a pooling layer used to find significant features in the input images. So far, it has exhibited unprecedented performance in object

detection and image classification. Nevertheless, this structure has some limitations. For instance, convolution filters can detect the noticeable features in an image, but they do not consider positional relationships among the feature maps. Such relationships can be utilized if they are common to all face images. For instance, the position of the human eye is above the mouth. Even though positional information is significant, the EMTCNN model does not consider it. Such spatial coordinate information can be incorporated into the model by augmenting the CoordConv layer. In the CoordConv layer, two additional channels are added to each feature map as shown in Figure 3a. Figure 3b shows sample Coord channels with height and width equal to 8. As shown in the figure, each channel represents a nominalized coordinate value between -1 and 1 and has the same size as the feature map. One channel represents coordinate information on the horizontal axis, and the other channel represents coordinate information on the vertical axis.



(**a**) CoordConv layer

-1	-0.7	-0.4	-0.1	0.1	0.4	0.7	1
-1	-0.7	-0.4	-0.1	0.1	0.4	0.7	1
-1	-0.7	-1	-1	-1	-1	-1	-1	-1	-1
-1	-0.7	-0.7	-0.7	-0.7	-0.7	-0.7	-0.7	-0.7	-0.7
-1	-0.7	-0.4	-0.4	-0.4	-0.4	-0.4	-0.4	-0.4	-0.4
-1	-0.7	-0.1	-0.1	-0.1	-0.1	-0.1	-0.1	-0.1	-0.1
-1	-0.7	0.1	0.1	0.1	0.1	0.1	0.1	0.1	0.1
-1	-0.7	0.4	0.4	0.4	0.4	0.4	0.4	0.4	0.4
		0.7	0.7	0.7	0.7	0.7	0.7	0.7	0.7
		1	1	1	1	1	1	1	1

(**b**) Coord channels

Figure 3. CoordConv layer.

3.2. Metric (Loss Function)

As the MTCNN uses the Euclidean distance as its loss function, it can be used to evaluate the accuracy of the extracted facial landmark points. An example of the 68 most popular landmark coordinates extracted by Dlib is shown in Figure 4a. As the coordinates are symmetric about the y-axis, their x-axis coordinates are often not learned properly compared to the y-axis coordinates, which produces an incorrect result. Figure 4b shows an example in which the coordinates are gathered near their average. To solve this problem, instead of considering only the distance between two points, we consider the x-axis and y-axis coordinates separately. For this reason, we use the Manhattan distance, which can consider the positions of the x- and y-axes individually and give more weight to the x-axis coordinates rather than the y-axis coordinates to evaluate the loss. Equation (1) represents our loss function.

$$N\_Loss(p, \hat{p}) = \frac{1}{68} \sum_{i=1}^{68} \frac{\alpha \left| (x_i - \hat{x}_i) \right|}{W} + \frac{\beta \left| (y_i - \hat{y}_i) \right|}{H} \tag{1}$$

Here, $N\_Loss(p, \hat{p})$ represents the normalized loss of two sets, p and $\hat{p}$, of 68 facial landmark points. p is generated by the EMTCNN and $\hat{p}$ is the ground truth. x and y ($\hat{x}$ and $\hat{y}$) are the coordinates of a point in p ($\hat{p}$). α and β are the weights of the x- and y-axes, respectively, and the sum of α and β is 1. W and H represent the width and height of the image, respectively.

(**a**) 68 facial landmark points by Dlib (**b**) Incorrect landmark points extraction

Figure 4. Facial symmetry.

3.3. Dataset

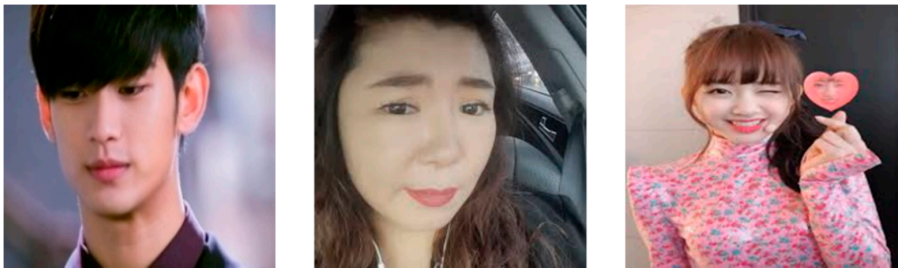
To train models for facial region detection, we use the WIDER FACE dataset [49], which consists of 32,203 images and 393,703 labeled faces with a high degree of variability in scale, pose, and occlusion. Incidentally, the face sizes provided in the WIDER FACE dataset are too small to display the 68 landmark points. Therefore, to train the O-Net, we used another face dataset, 300 videos in the wild (300-VW) [50], which contains approximately 110 video files and 68 facial landmark point data for each frame. As we used only 200 frames from each video file, our dataset contained a total of 22,000 images from 111 video files. However, 300-VW is more likely to cause overfitting for a particular person owing to the small number of people appearing in the images. Therefore, we used 20,000 additional facial images [51] to consider various cases in the models. As there was no ground truth for the added images, we used Dlib to create 68 facial landmark points for each facial image. Figure 5a shows the original facial images and Figure 5b shows the erroneous facial landmark point detections by Dlib for the images. The erroneous detections can be classified into three types: (i) landmarks are not detected at all; (ii) some of the facial landmarks are incorrectly detected and their coordinates are slightly off the correct position; (iii) non-facial regions are detected as facial landmarks.

To solve this problem, we used both Dlib and SegNet [10]. We first executed SegNet for face segmentation and then overlaid the result onto the original facial image to highlight the facial landmarks. By applying Dlib to the overlaid region, we obtained accurate landmark coordinates. Figure 5c shows the detection results obtained by combining SegNet and Dlib, which are much better than those obtained by Dlib alone.

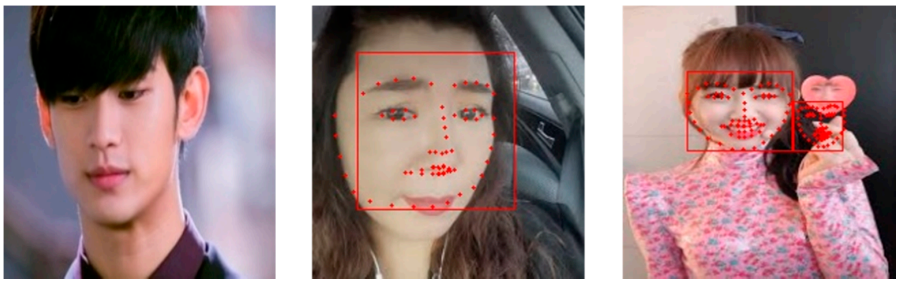
We collected approximately 42,000 images from 300-VW and other sources. In addition, we performed several image augmentation operations on the images in the dataset to represent still more diverse situations. For instance, facial images with diverse angles can be generated by flipping the images, and different illuminations and noise environments can be produced by adjusting the brightness and salt and pepper noise. Further, some pixels can be masked to reflect the effect of wearing various accessories such as glasses and masks. Using various image augmentation operations, we obtained a total of 1,680,000 images, as calculated in Table 2. Figure 6 shows images produced using the five image augmentation operations we considered in this work.

Table 2. Image dataset augmentation.

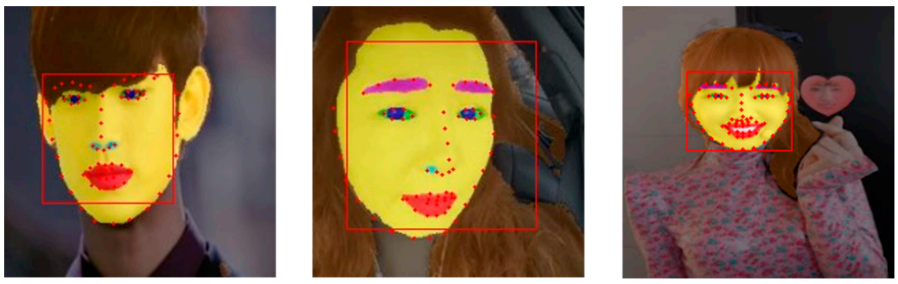
Images	Augmentation Operations			
	Flip (Left/Right)	Brightness Adjustment	Salt & Pepper Noise	Region of Interest Filling
42,000	×2	×2	×2	×5
Total	42,000 × 40 = 1,680,000			



(**a**) Original images



(**b**) Erroneous landmark point extraction by Dlib
(no detection, inaccurate detection, false detection)



(**c**) Landmark point extraction by SegNet + Dlib

Figure 5. Results of facial landmark point extraction.

(**a**) Original (**b**) Flip (**c**) ROI filling (**d**) Brightness adjustment (**e**) Salt & pepper noise

Figure 6. Image augmentation results.

4. Experiment

This section describes the experiments we performed to evaluate our scheme. First of all, we measure the speed of facial landmark point extraction by our augmented EMTCNN model and compare it with that by other methods. Next, we compare the accuracy of each model in facial landmark point extraction. Finally, we evaluate the extraction accuracy of our model depending on the learning weights of the x- and y-axes. In the experiments, we used an Intel (R) Core (TM) i7-8700 CPU (Santa Clara, California, USA), Samsung 32G DDR4 memory (Suwon, Korea), NVIDIA Geforce GTX 1080ti (Santa Clara, California, USA). The operating system is Windows 10 version and all the experiments are all implemented by Python 3.5 environment. The mini-batch size was 384, and we had a total of 480,000 iterations.

4.1. Training

As mentioned before, all the CNN-based models we consider in this paper consist of three networks. As each network has different size inputs and roles, it is necessary to preprocess the dataset for training each network properly. The role of P-Net is to select facial region candidates from among randomly cropped regions of an image. To do this, the regions are first cropped randomly from the image and evaluated by calculating the Intersect over Union (IoU) value, which represents the number of pixels contained in the bounding box of the facial region. Based on the IoU values, all the cropped regions are divided into three categories—positive, negative, and part—using the following criteria: positive > = 0.65, 0.4 < part < 0.65, negative < = 0.3. After all, we need a training dataset that contains face images and the ground truth of the facial region to train the P-Net properly to find positive regions from face images accurately. The R-Net refines the facial region candidates produced by the P-Net further by resizing and classifying them once more using the same criteria as the P-Net. The R-Net is trained using the resized regions and their category information. Finally, as the O-Net extracts the landmark points of the facial region, the dataset should contain such facial landmark point information. The O-Net uses the facial region candidates produced by the R-Net as input and produces a facial region and its facial landmark points as its output.

4.2. Accuracy of Landmark Point Extraction

In this experiment, we compare the landmark point extraction accuracy of three models—augmented EMTCNN, EMTCNN, and Dlib—using the Helen dataset, which was not present in the training set. These images have 68 predefined facial landmark points as the ground truth. Hence, the accuracy can be compared quantitatively using the total distance between the ground truth points and landmark points extracted by each model. The mean normalized distance (MND) can be computed using Equation (2):

$$MND = \left(\frac{1}{68} \sum_{i=1}^{68} \frac{\sqrt{(x_i - \hat{x}_i)^2}}{W} + \frac{\sqrt{(y_i - \hat{y}_i)^2}}{H} \right) \times 100. \quad (2)$$

Table 3 compares the mean normalized distances of various methods used to extract the 68 facial landmark points, including EMTCNN and Augmented EMTCNN. Since Dlib was trained with the -300-W dataset, it was excluded from the comparison for 300-W. As we can see from the table, the mean normalized distance for Augmented EMTCNN was improved compared to that for EMTCNN.

Table 3. Comparison of mean normalized distances.

Method	Mean Normalized Distance	
	Helen	**300-W**
DRMF (Discriminative Response Map Fitting) [52]	6.70	9.22
RCPR (Robust Cascaded Pose Regression) [53]	5.93	8.35
ESR (Explicit Shape Regression) [54]	5.70	7.58
CFAN (Coarse-to-Fine Auto-encoder Networks) [55]	5.53	7.69
SDM (Supervised Descent Method) [56]	5.50	7.50
CFSS (Coarse-to-Fine Shape Searching) [57]	4.63	5.76
TCDCN (Tasks-Constrained Deep Convolutional Network) [58]	4.60	**5.54**
Dlib [37]	**4.47**	-
EMTCNN	5.66	6.63
Augmented EMTCNN	4.65	5.59

For the ground truths shown in Figure 7a, the actual facial landmark points extracted using the Dlib, EMTCNN, and Augmented EMTCNN models are shown in Figure 7b–d, respectively. From the figures, we can see that each model performs slightly differently depending on the input image, even though their overall detection results are acceptable. Figure 8 shows more examples of facial landmark detection performed by our Augmented EMTCNN model.

(**a**) Ground truth (**b**) Dlib

(**c**) EMTCNN (**d**) Augmented EMTCNN

Figure 7. Comparison of facial landmark extraction results.

Figure 8. Facial landmark points extracted by our Augmented EMTCNN model.

4.3. Landmark Point Extraction Speed

In this experiment, we compare the speed of the five models Augmented EMTCNN, EMTCNN, MTCNN, Tasks-Constrained Deep Convolutional Network (TCDCN), and Dlib. Augmented EMTCNN, TCDCN, and Dlib showed very good landmark extraction accuracy. We also considered MTCNN for reference even though the number of landmark points is five. To compare the extraction speed, we measured the number of frames processed per second (fps) for the 200×200 input images. Table 4 shows the number of extracted facial landmark points and the speed in fps of each model. It is clear that the processing speed of the original MTCNN is the highest, as it finds just five facial landmark points. However, such few landmark points are insufficient to represent facial landmark features properly. On the other hand, Augmented EMTCNN and EMTCNN extracted 68 landmark points at speeds of 68 fps and 70 fps, respectively. Although Augmented EMTCNN is a little slower than EMTCNN, its extraction accuracy is almost twice that of EMTCNN, and an extraction speed of 68 fps is sufficient for real-time processing. In the case of Dlib, the processing speed is not enough to support the real-time processing of video, even though it can detect the same number of landmark points.

Table 4. Comparison of extraction speeds.

	Augmented EMTCNN	**EMTCNN**	**MTCNN**	**TCDCN**	**Dlib**
No. of landmark points	68	68	5	68	68
Speed (fps)	68	70	99	23	15

4.4. Effects of Weights on Accuracy

In Figure 4, we mentioned that the y-axis of the facial landmark points has symmetry and, owing to this, the x-axis coordinates of the facial landmark points were not well learned compared to the y-axis coordinates. To reflect this property of the x-axis coordinates, we trained the model by using different weights for the x- and y-axes. Table 5 indicates the variation of the mean normalized distance according to the ratio of the x- and y-axis learning weights. In the case of the EMTCNN model, the mean normalized distance was improved until the weight ratio of the x- and y-axes became 8:2. On the other hand, in the case of the Augmented EMTCNN model, the ratio 6:4 produced the best mean normalized distance. Hence, we trained each model using these weight ratios.

Table 5. Comparison of extraction accuracy depending on weights.

		(x-axis Learning Weight/y-axis Learning Weight)				
		5:5	**6:4**	**7:3**	**8:2**	**9:1**
Mean normalized distance	EMTCNN	6.94	6.72	6.36	**5.66**	6.83
	Augmented EMTCNN	5.42	**4.65**	5.22	5.43	6.07

5. Conclusions

In this paper, we proposed a new method for extracting the 68 most popular feature points to represent facial landmarks in real time. More specifically, we first extended the original MTCNN model to increase the number of facial landmark points from 5 to 68. Then, to improve the accuracy of facial landmark extraction, we augmented the EMTCNN model by using two state-of-the-art convolution techniques—dilated convolution and CoordConv. In our experiments, we compared the number of extracted facial landmark points, processing speed, and extraction accuracy of four methods—Dlib, MTCNN, EMTCNN, and Augmented EMTCNN. The Augmented EMTCNN model extracted the 68 most popular feature points at a sufficient speed for real-time processing, and its accuracy was almost the same as the best accuracy achieved by Dlib. Our scheme can be applied to

various applications that require real-time object recognition, such as face recognition in payment services and pedestrian detection in autonomous driving. In the near future, we will investigate the face segmentation model and apply its segmentation results to input images for training to further improve the extraction accuracy.

Author Contributions: Conceptualization, H.-W.K.; methodology, H.-W.K. and H.-J.K.; validation, H.-W.K., H.-J.K. and S.R.; formal analysis, S.R. and E.H.; data curation, H.-W.K.; writing—original draft preparation, H.-W.K. and H.-J.K.; writing—review and editing, S.R. and E.H.; visualization, H.-J.K.; supervision, E.H.; project administration, S.R. and E.H.; funding acquisition, S.R. All authors have read and agreed to the published version of the manuscript.

Funding: This work was supported by the National Research Foundation of Korea (NRF) grant funded by the Korea government (MSIT) (NRF-2019R1F1A1060668).

Acknowledgments: We greatly appreciate the anonymous reviewers for their comments and suggestions.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Kim, H.; Park, J.; Kim, H.; Hwang, E. Facial landmark extraction scheme based on semantic segmentation. In Proceedings of the 2018 International Conference on Platform Technology and Service (PlatCon), Jeju, Korea, 29–31 January 2018; pp. 1–6.
2. Kim, H.; Kim, H.; Hwang, E. Real-Time Facial Feature Extraction Scheme Using Cascaded Networks. In Proceedings of the 2019 IEEE International Conference on Big Data and Smart Computing (BigComp), Kyoto, Japan, 27 February–2 March 2019; pp. 1–7.
3. Kim, H.; Kim, H.; Hwang, E. Real-time shape tracking of facial landmarks. *Multimedia Tools Appl.* **2018**, in press. [CrossRef]
4. Krizhevsky, A.; Sutskever, I.; Hinton, G.E. ImageNet classification with deep convolutional neural networks. *Commun. ACM* **2017**, *60*, 84–90. [CrossRef]
5. Jung, S.; Kim, Y.; Hwang, E. Real-time car tracking system based on surveillance videos. *EURASIP J. Image Video Process.* **2018**, *2018*, 133. [CrossRef]
6. Fan, H.; Zhou, E. Approaching human level facial landmark localization by deep learning. *Image Vis. Comput.* **2016**, *47*, 27–35. [CrossRef]
7. Ramanan, D.; Zhu, X. Face detection, pose estimation, and landmark localization in the wild. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Providence, RI, USA, 16–21 June 2012; pp. 2879–2886.
8. Hou, Q.; Wang, J.; Cheng, L.; Gong, Y. Facial landmark detection via cascade multi-channel convolutional neural network. In Proceedings of the IEEE International Conference on Image Processing (ICIP), Quebec City, QC, Canada, 27–30 September 2015; pp. 1800–1804.
9. Feng, Z.H.; Kittler, J.; Awais, M.; Huber, P.; Wu, X.J. Face detection, bounding box aggregation and pose estimation for robust facial landmark localisation in the wild. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 2106–2115.
10. Kim, H.; Park, J.; Kim, H.; Hwang, E.; Rho, S.; Rho, S. Robust facial landmark extraction scheme using multiple convolutional neural networks. *Multimedia Tools Appl.* **2018**, *78*, 3221–3238. [CrossRef]
11. Zhang, Z.; Luo, P.; Loy, C.C.; Tang, X. Facial landmark detection by deep multi-task learning. In Proceedings of the European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014; pp. 94–108.
12. Zhang, K.; Zhang, Z.; Li, Z.; Qiao, Y. Joint Face Detection and Alignment Using Multitask Cascaded Convolutional Networks. *IEEE Signal Process. Lett.* **2016**, *23*, 1499–1503. [CrossRef]
13. Deng, Z.; Li, K.; Zhao, Q.; Chen, H. Face landmark localization using a single deep network. In Proceedings of the Chinese Conference on Biometric Recognition, Chengdu, China, 14–16 October 2016; pp. 68–76.
14. Chen, L.C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. Semantic image segmentation with deep convolutional nets and fully connected CRFs. *arXiv* **2014**, arXiv:1412.7062.
15. Chen, L.C.; Papandreou, G.; Kokkinos, I.; Murphy, K.; Yuille, A.L. DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. *IEEE Trans. Pattern Anal. Mach. Intell.* **2018**, *40*, 834–848. [CrossRef]

16. Chen, L.C.; Papandreou, G.; Schroff, F.; Adam, H. Rethinking Atrous Convolution for Semantic Image Segmentation. *arXiv* **2017**, arXiv:1706.05587.
17. Liu, R.; Lehman, J.; Molino, P.; Such, F.P.; Frank, E.; Sergeev, A.; Yosinski, J. An Intriguing Failing of Convolutional Neural Networks and the CoordConv Solution. *Adv. Neural Inf. Process. Syst.* **2018**, *31*, 9605–9616.
18. Rew, J.; Choi, Y.H.; Kim, D.; Rho, S.; Hwang, E. Evaluating skin hereditary traits based on daily activities. *Front. Innov. Future Comput. Commun.* **2014**, *301*, 261–270. [CrossRef]
19. Kim, H.; Kim, W.; Rew, J.; Rho, S.; Hwang, E. Evaluation of hair and scalp condition based on microscopy image analysis. In Proceedings of the 2017 International Conference on Platform Technology and Service (PlatCon), Busan, Korea, 13–15 February 2017; pp. 1–4.
20. Rew, J.; Choi, Y.H.; Rho, S.; Hwang, E. Monitoring skin condition using life activities on the SNS user documents. *Multimed. Tools Appl.* **2018**, *77*, 9827–9847. [CrossRef]
21. Rew, J.; Choi, Y.H.; Kim, H.; Hwang, E. Skin Aging Estimation Scheme Based on Lifestyle and Dermoscopy Image Analysis. *Appl. Sci.* **2019**, *9*, 1228. [CrossRef]
22. Kim, J.; Moon, J.; Hwang, E.; Kang, P. Recurrent inception convolution neural network for multi short-term load forecasting. *Energy Build.* **2019**, *194*, 328–341. [CrossRef]
23. Le, T.; Vo, M.; Vo, B.; Hwang, E.; Rho, S.; Baik, S. Improving Electric Energy Consumption Prediction Using CNN and Bi-LSTM. *Appl. Sci.* **2019**, *9*, 4237. [CrossRef]
24. Le, N.Q.K.; Ho, Q.T.; Ou, Y.Y. Incorporating deep learning with convolutional neural networks and position specific scoring matrices for identifying electron transport proteins. *J. Comput. Chem.* **2017**, *38*, 2000–2006. [CrossRef]
25. Le, N.Q.K.; Yapp, E.K.Y.; Ou, Y.Y.; Yeh, H.Y. iMotor-CNN: Identifying molecular functions of cytoskeleton motor proteins using 2D convolutional neural network via Chou's 5-step rule. *Anal. Biochem.* **2019**, *575*, 17–26. [CrossRef]
26. Le, N.Q.K.; Nguyen, V.N. SNARE-CNN: A 2D convolutional neural network architecture to identify SNARE proteins from high-throughput sequencing data. *PeerJ Comput. Sci.* **2019**, *5*, e177. [CrossRef]
27. Le, N.Q.K.; Huynh, T.T.; Yapp, E.K.Y.; Yeh, H.Y. Identification of clathrin proteins by incorporating hyperparameter optimization in deep learning and PSSM profiles. *Comput. Methods Programs Biomed.* **2019**, *177*, 81–88. [CrossRef]
28. Le, N.Q.K.; Ho, Q.T.; Yapp, E.K.Y.; Ou, Y.Y.; Yeh, H.Y. DeepETC: A deep convolutional neural network architecture for investigating and classifying electron transport chain's complexes. *Neurocomputing* **2020**, *375*, 71–79. [CrossRef]
29. Girshick, R.; Donahue, J.; Darrell, T.; Malik, J. Rich feature hierarchies for accurate object detection and semantic segmentation. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 580–587.
30. Uijlings, J.R.R.; Van De Sande, K.E.A.; Gevers, T.; Smeulders, A.W.M. Selective Search for Object Recognition. *Int. J. Comput. Vis.* **2013**, *104*, 154–171. [CrossRef]
31. Cortes, C.; Vapnik, V. Support-vector networks. *Mach. Learn.* **1995**, *20*, 273–297. [CrossRef]
32. Girshick, R. Fast r-cnn. In Proceedings of the IEEE International Conference on Computer Vision, Santiago, Chile, 13–16 December 2015; pp. 1440–1448.
33. Ren, S.; He, K.; Girshick, R.; Sun, J. Faster r-cnn: Towards real-time object detection with region proposal networks. *Adv. Neural Inf. Process. Syst.* **2015**, *28*, 91–99. [CrossRef] [PubMed]
34. 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, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 779–788.
35. Redmon, J.; Farhadi, A. YOLO9000: Better, faster, stronger. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Honolulu, HI, USA, 21–26 July 2017; pp. 7263–7271.
36. Redmon, J.; Farhadi, A. YOLOv3: An Incremental Improvement. *arXiv* **2018**, arXiv:1804.02767.
37. King, D.E. Dlib-ml: A machine learning toolkit. *J. Mach. Learn. Res.* **2009**, *10*, 1755–1758.
38. Kazemi, V.; Sullivan, J. One millisecond face alignment with an ensemble of regression trees. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Columbus, OH, USA, 23–28 June 2014; pp. 1867–1874.
39. Sun, Y.; Wang, X.; Tang, X. Deep convolutional network cascade for facial point detection. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 8–10 June 2015; pp. 3476–3483.

40. Ranjan, R.; Patel, V.M.; Chellappa, R. HyperFace: A Deep Multi-Task Learning Framework for Face Detection, Landmark Localization, Pose Estimation, and Gender Recognition. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *41*, 121–135. [CrossRef]
41. Xiao, S.; Feng, J.; Liu, L.; Nie, X.; Wang, W.; Yan, S.; Kassim, A. Recurrent 3d-2d dual learning for large-pose facial landmark detection. In Proceedings of the IEEE International Conference on Computer Vision, Venice, Italy, 22–29 October 2017; pp. 1633–1642.
42. Lai, H.; Xiao, S.; Pan, Y.; Cui, Z.; Feng, J.; Xu, C.; Yin, J.; Yan, S. Deep Recurrent Regression for Facial Landmark Detection. *IEEE Trans. Circuits Syst. Video Technol.* **2016**, *28*, 1144–1157. [CrossRef]
43. Badrinarayanan, V.; Handa, A.; Cipolla, R. Segnet: A deep convolutional encoder-decoder architecture for robust semantic pixel-wise labelling. *arXiv* **2015**, arXiv:1505.07293.
44. Badrinarayanan, V.; Badrinarayanan, V.; Cipolla, R. SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation. *IEEE Trans. Pattern Anal. Mach. Intell.* **2017**, *39*, 2481–2495. [CrossRef]
45. Rothe, R.; Guillaumin, M.; Van Gool, L. Non-maximum suppression for object detection by passing messages between windows. In Proceedings of the Asian Conference on Computer Vision, Singapore, 1–5 November 2014; pp. 290–306.
46. Szegedy, C.; Vanhoucke, V.; Ioffe, S.; Shlens, J.; Wojna, Z. Rethinking the inception architecture for computer vision. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 2818–2826.
47. Simonyan, K.; Zisserman, A. Very deep convolutional networks for large-scale image recognition. *arXiv* **2014**, arXiv:1409.1556.
48. 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, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 770–778.
49. Yang, S.; Luo, P.; Loy, C.C.; Tang, X. Wider face: A face detection benchmark. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Las Vegas, NV, USA, 26 June–1 July 2016; pp. 5525–5533.
50. Shen, J.; Zafeiriou, S.; Chrysos, G.G.; Kossaifi, J.; Tzimiropoulos, G.; Pantic, M. The first facial landmark tracking in-the-wild challenge: Benchmark and results. In Proceedings of the IEEE International Conference on Computer Vision Workshops, Santiago, Chile, 7–13 December 2015; pp. 50–58.
51. Le, V.; Brandt, J.; Lin, Z.; Bourdev, L.; Huang, T.S. Interactive facial feature localization. In Proceedings of the European Conference on Computer Vision, Rome, Italy, 8–11 October 2012; pp. 679–692.
52. Asthana, A.; Zafeiriou, S.; Cheng, S.; Pantic, M. Robust discriminative response map fitting with constrained local models. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013; pp. 3444–3451.
53. Burgos-Artizzu, X.P.; Perona, P.; Dollár, P. Robust face landmark estimation under occlusion. In Proceedings of the IEEE International Conference on Computer Vision, Sydney, Australia, 1–8 December 2013; pp. 1513–1520.
54. Cao, X.; Wei, Y.; Wen, F.; Sun, J. Face Alignment by Explicit Shape Regression. *Int. J. Comput. Vis.* **2013**, *107*, 177–190. [CrossRef]
55. Zhang, J.; Shan, S.; Kan, M.; Chen, X. Coarse-to-fine auto-encoder networks (cfan) for real-time face alignment. In Proceedings of the European Conference on Computer Vision, Zurich, Switzerland, 6–12 September 2014; pp. 1–16.
56. Xiong, X.; De la Torre, F. Supervised descent method and its applications to face alignment. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013; pp. 532–539.
57. Zhu, S.; Li, C.; Change Loy, C.; Tang, X. Face alignment by coarse-to-fine shape searching. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 4998–5006.
58. Zhang, Z.; Luo, P.; Loy, C.C.; Tang, X. Learning deep representation for face alignment with auxiliary attributes. *IEEE Trans. Pattern Anal. Mach. Intell.* **2015**, *38*, 918–930. [CrossRef] [PubMed]