electronics

MDPI

Article

Real-Time Hair Segmentation Using Mobile-Unet

Ho-Sub Yoon [1,2,*], Seong-Woo Park [2] and Jang-Hee Yoo [1,2]

[1] HRI Section, Artificial Intelligence Research Laboratory, Electronics and Telecommunications Research Institute (ETRI), 218 Gajung-ro, Yueseong-gu, Daejeon 34129, Korea; jhy@etri.re.kr
[2] ICT Department, ETRI School, University of Science & Technology, Daejeon 34129, Korea; starspder@naver.com
* Correspondence: yoonhs@etri.re.kr; Tel.: +82-42-860-5233

Abstract: We described a real-time hair segmentation method based on a fully convolutional network with the basic structure of an encoder–decoder. In one of the traditional computer vision techniques for hair segmentation, the mean shift and watershed methodologies suffer from inaccuracy and slow execution due to multi-step, complex image processing. It is also difficult to execute the process in real-time unless an optimization technique is applied to the partition. To solve this problem, we exploited Mobile-Unet using the U-Net segmentation model, which incorporates the optimization techniques of MobileNetV2. In experiments, hair segmentation accuracy was evaluated by different genders and races, and the average accuracy was 89.9%. By comparing the accuracy and execution speed of our model with those of other models in related studies, we confirmed that the proposed model achieved the same or better performance. As such, the results of hair segmentation can obtain hair information (style, color, length), which has a significant impact on human-robot interaction with people.

Keywords: computer vision; hair segmentation; FCN; deep learning; Mobile-Unet; HRI

Citation: Yoon, H.-S.; Park, S.-W.; Yoo, J.-H. Real-Time Hair Segmentation Using Mobile-Unet. *Electronics* **2021**, *10*, 99. https://doi.org/10.3390/electronics10020099

Received: 1 December 2020
Accepted: 29 December 2020
Published: 6 January 2021

1. Introduction

A person's hair information is important in assessing their appearance. Hair varies widely in color, length, pattern, and texture information depending on gender, age, fashion, culture, and personal taste. Appearance information, whether positive or negative, acts as an important factor in human-to-human interactions as well as in human interactions with robots [1]. For robots to have social relationships with people they need to mimic the way people have social relationships [2]. However, a strand of hair is very flexible and thin, which can be transformed into a variety of shapes, similar to skin color, or heavily influenced by external lighting, making it difficult to segment skin, hair, and background areas. Additionally, hair segmentation information can not only complement social robot applications and face recognition results, but it can also be used for a variety of applications, such as make-up changes and character photo editing [3]. Thus, a variety of studies have been conducted recently on hair partitioning and recognition of style information [3–6], but the study is challenging because hair areas vary widely depending on complex backgrounds, different poses, reflected light, race, hair color, and dyeing.

Earlier hair split and color automatic recognition were proposed by Yacoob [4], and hair areas were extracted using the ratio of face and color information and area grinding methods for the front of the face. Wang [5] suggested a pre-learned hair segmentation classifier to automatically expand into the entire area if the hair seeded area was manually specified. However, this method has the weaknesses of having to manually designate hair areas and of having low division accuracy if the background and hair color are similar. Wang [6] proposed the data-driven isomeric manifold inference method. This method manually enters the hair area from the labeled hair segmentation training image and produces a probability map that enables the generation of the seed area. This method also

addresses the difficulties of manually designating the initial hair area, but the difficulties of the hair area not being split from the complex background remain.

Recently, there has been much success with deep neural networks (DNNs) and in many tasks, including semantic segmentation, DNN-based hair segmentation methods have been introduced. Guo and Aarabi [7] presented a method for binary classification using neural networks that perform training and classification on the same data using the help of a pre-training heuristic classifier. They used a heuristic method to mine positive and negative hair patches from each image with high confidence and trained a separate DNN for each image, which was then used to classify the remaining pixels. Long [8] demonstrated convolutional neural networks (CNNs) that first trained end-to-end and pixel-to-pixel for object segmentation. Fully convolutional networks (FCNs) predict dense outputs from free-sized inputs. Both training and inference are performed on the whole image at one time by back-propagation and dense feed-forward computation. Network up-sampling layers enable pixel-wise prediction and learning in nets with subsampled pooling. However, though this method has obtained good segmentation results, it has not been proposed for the purpose of hair segmentation but has only been used for general object segmentation. Chai [3] attempted to create the first, fully automatic method for three-dimensional (3D) hair segmentation from a single input image, with no parameter tuning or user interaction. Moreover, Qin [9] introduced the use of a fully connected conditional random field (CRF) and FCN to perform pixel-wise semantic segmentation on hair, skin, and background.

However, with the application of DNN in hair segmentation, the accuracy remarkably improves, but the applications on mobile devices or embedded platforms without a GPU (Graphics Processing Unit) are not easy to apply in real-time due to its unique, large number of parallel computations with fully connected heavy-weight network architectures. Therefore, adaptive importance learning [10], knowledge distillation [11], and MobileNet2 [12] algorithms can be utilized to train a light-weight network for speed-up. Adaptive importance learning proposes a learning strategy to maximize the pixel-wise fitting capacity of a given light-weight network architecture. Knowledge distillation technology is utilized when a large (teacher) pre-trained network is used to train a smaller (student) network. These two methods are suitable for DNN models of CNN and ResNet architectures with general pipeline structures for detection or classification purposes, and a MobileNet2-based approach is more efficient for image segmentation based on U-Net with structures that segment through steps to compress and restore images.

In this paper, we propose a new approach to a hair segmentation method with a light-weighted model based on Mobile-Unet for fast and accurate results. The proposed method includes the optimization techniques of depth-wise separable convolution and an inverted residual block. Furthermore, we have used the proposed generated and augmented datasets for training the deep neural networks model. This paper is organized as follows: Section 2 presents the pre-processing steps, landmark detection, size normalization, data augmentation, and hair recoloring; Section 3 describes the proposed method in detail, followed by the experimental results including the training datasets demonstrated in Section 4; finally, Section 5 concludes this paper.

2. Pre-Processing Steps

To achieve DNN-based hair segmentation, several pre-processing methods are required. Here, we describe landmark detection, size normalization, and data augmentation of face images, as well as hair recoloring of a detected hair region.

2.1. Landmark Detection and Size Normalization

As an initial experiment, the size of the input image was simply normalized to a size of 224 × 224, which was used as training data without other pre-processing. In this case, the real-time test did not obtain good results according to changes in the environment, such as distance or lighting. To solve this problem, we sought a method to normalize

images regardless of size, location, or distance by immobilizing them in a specific reference position rather than simply resizing them and configuring the dataset. Here, we used 68 face landmarks detectors [13] to normalize the input images. The face detector used a single-shot multi-box detector [14] that had relatively fast detection and was highly accurate. We normalized the input image to the central position of the image, as shown in Figure 1, after the detection of landmarks.

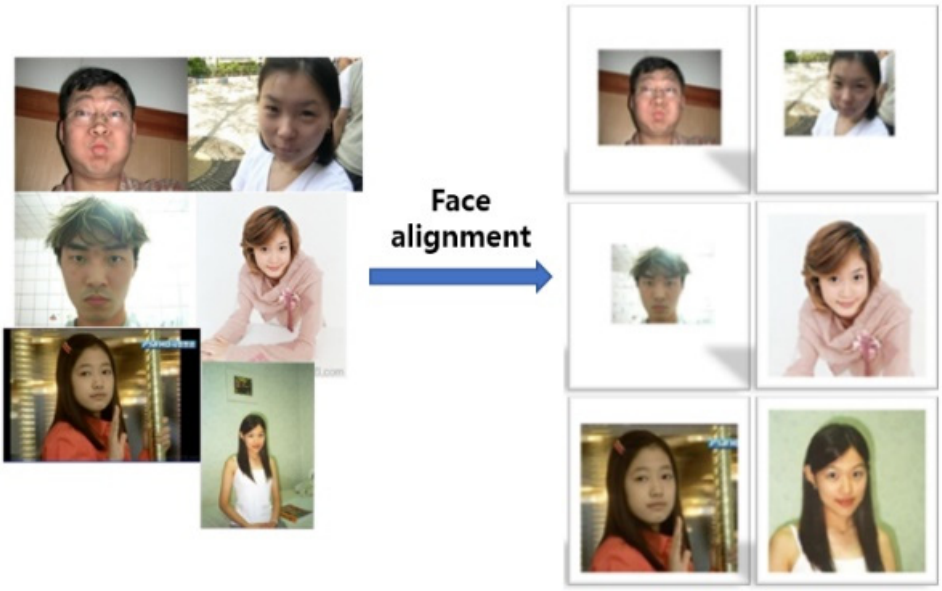

Figure 1. Images normalized by face alignment.

As shown in Figure 1, we normalized images of any size, by converting them to a size of 224 × 224, so that the nose was at the central position of the image. If the converted area was larger than the original, we set the pixels in the padded background area to 255. The reason the border of the image was not padded to zero was mainly that the border of normal images, or real-time images, was bright instead of dark.

2.2. Data Augmentation

If a video is taken in real-time, the input image can be characterized according to changes in distance, pose, camera angle, and light intensity. For training in DNN models, if sufficient training datasets that are the same as a variable input environment cannot be collected, it is common to increase the training data by artificially generating such environmental changes. Consequently, a total of 95,200 items of training data were obtained from 6800 items of training data by using flip and rotation (± 5, ± 15, ± 30) to acquire data, which enabled 14 additional images to be obtained from one image. Figure 2 shows the results of the data augmentation.

2.3. Basic Hair Segmentation Model

For hair segmentation, the use of a segmentation model with the basic structure of an encoder–decoder [15] is more appropriate than AlexNet or VGGNet, which are composed of CNN and ResNet architectures with general pipeline structures for detection or classification purposes. Object segmentation, for which location information—such as hair segmentation—is important, requires spatial information for images entering the input layer of the model to be matched with spatial information for the resulting image going out to the output layer. To satisfy this process, the segmentation model takes the form of down-sampling and up-sampling steps, as shown in Figure 3.

Figure 2. Example of data augmentation.

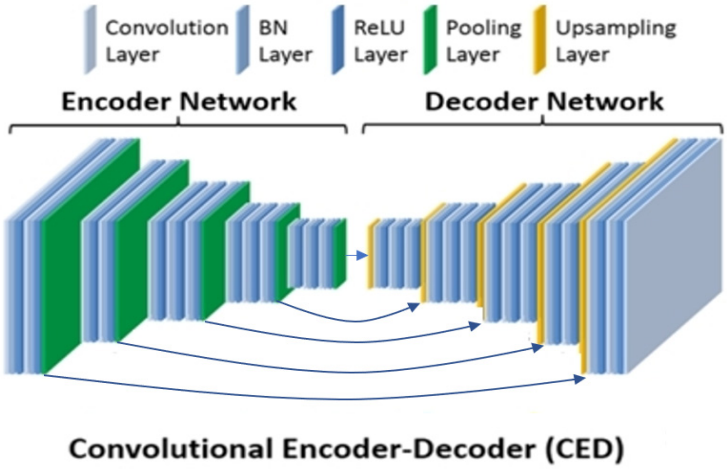

Figure 3. The basic structure of the encoder–decoder.

A model that initiates the image segmentation and introduces an encoder–decoder-style restoration structure as the core of the model is the FCN model for semantic segmentation [8]. However, the FCN does not fully recover the segmentation because it uses an asymmetric method that goes from a very small channel to a large channel instead of a format in which dimensions are recovered at a certain rate during up-sampling. The model that addresses the asymmetry of the skip-connection and up-sampling is the U-Net model. Therefore, we used the U-Net model as a basis for the hair segmentation model.

2.4. Hair Recoloring Processing

When the hair is correctly segmented, it is possible to recolor the hair various colors that differ from the input hair color. Hair recoloring has the advantage of allowing users to experience what colors would match when they change their hair color. To automatically convert hair colors, we used a method to convert input images into intensity images using color maps to select desired colors and to match them in the lookup table (LUT) from brightness values [16]. Figure 4 shows an example of the hair recoloring process.

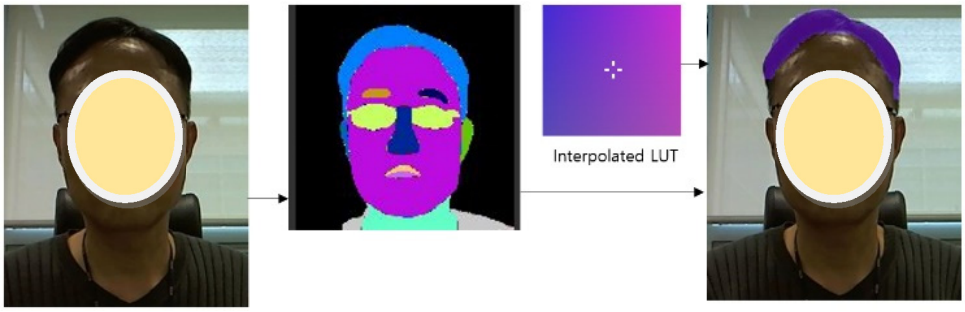

Figure 4. Hair recoloring process.

3. Proposed Method

In human–robot interaction, where real-time interaction is essential, the hair segmentation model using the traditional U-Net model has difficulty in real-time processing. To solve this, we propose the use of the two optimization algorithms (depth-wise separable convolution and inverted residual block) used by MobileNetV2 [12] in the traditional U-Net.

3.1. Depth-Wise Separable Convolution

The depth-wise separable convolution operation is an optimization technique that reduces the computational capacity by performing 1×1 component operations on each channel instead of the normal convolution operation. Figure 5a presents general convolution and Figure 5b presents depth-wise separable convolution. The U-Net in real-time processing must reduce the convolution operations that require the most operations. Depth-wise separable convolution can be seen as a factorization of this common convolution.

$$Cost = D_K \times D_K \times M \times N \times D_F \times D_F \tag{1}$$

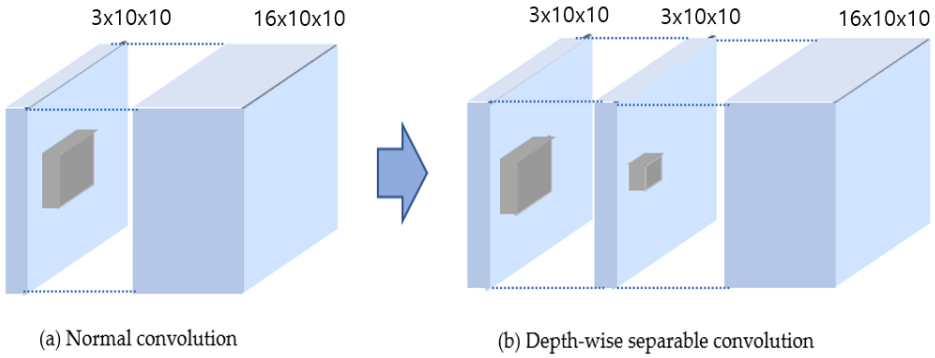

Figure 5. Comparison of (**a**) normal convolution and (**b**) separable convolution [12].

In Equation (1), D_K represents the kernel size, D_F represents the size of the input image, M represents the input channel, and N represents the output channel. *Cost* represents the number of computations.

$$Cost = D_K \times D_K \times M \times D_F \times D_F + M \times N \times D_F \times D_F \tag{2}$$

Equation (2) shows the total computational capacity of the depth-wise separable convolution operation. The difference between the arithmetic of (1) and (2) is shown in (3).

$$Diff\_Ratio = (1/N) + (1/(D_K{}^2)) \tag{3}$$

The *Diff_Ratio* indicates the difference in the ratio between Equations (1) and (2). This is about 8 to 9 times the difference, proving that the depth-wise separation operation is much faster than the normal convolution operation.

3.2. Inverted Residual Block

The inverted residual block structure is a method that reduces the amount of calculation by replacing a common structure that is connected between feature maps with a structure with large channels between 1×1 component bottlenecks. Residual blocks indicate the beginning and end of a convolutional block with a skip connection. By adding these two states, the network has the ability to access earlier activations that were not modified in the convolutional block. This method was revealed to be essential to building large depth networks. When looking more closely at the skip connection, it becomes clear that an original residual block follows a wide to narrow to wide approach, concerning the number of channels, if the input channel has a large number of channels that are compressed with an inexpensive 1×1 convolution. In that case, the following 3×3 convolution has far fewer parameters. To add input and output at the end, the number of channels is increased again using another 1×1 convolution.

On the other hand, an inverted residual block follows a narrow to wide to narrow approach. The first step widens the network using a 1×1 convolution because the following 3×3 depth-wise convolution has already greatly reduced the number of parameters. Afterward, another 1×1 convolution "squeezes" the network in order to match the initial number of channels. The inverted residual block method is very effective in terms of efficiency when we use GPUs. A GPU has internal and external memories, so the size of the lift and the size of the drop are the most important. Considering the memory swap, the low number of channels in the input layer and the last output layer indicates that it is efficient. This is an important attribute for mobile robots with limited memory. Figure 6 shows the configuration of the inverted residual block.

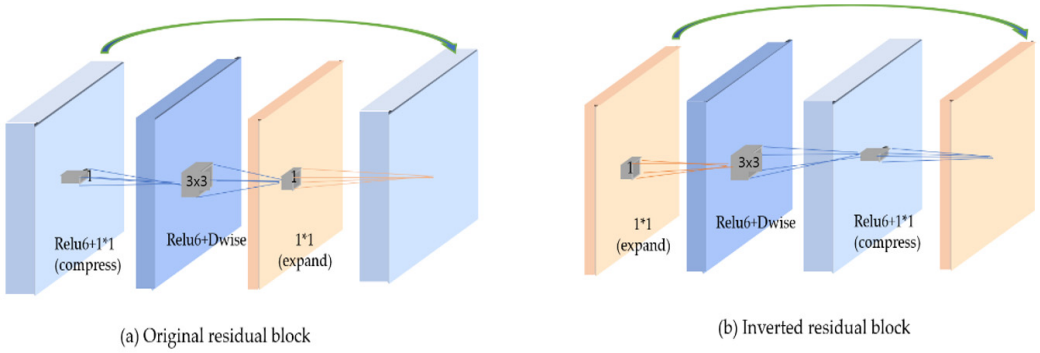

Figure 6. Configuration of the inverted residual block [12].

3.3. Proposed Mobile-Unet

U-Net combined with the optimization techniques described in the previous section is called Mobile-Unet. Figure 7 shows the brief architecture of the proposed Mobile-Unet model.

The input size started at 224×224. The image size was reduced by each step to an image size of 7×7 after a total of five steps of contraction. In the next stage, we went through an expansion phase of the five steps symmetrically. Because the expansion phase is a process of restoring information, the layer structure was built to be the same as that of the contraction phase. Table 1 shows the Mobile-Unet contraction process, where T is the channel extension factor and C is the final number of channels after the last convolution operation. N indicates the number of times the operation is repeated for the inverted bottleneck and S indicates the stride. The best empirical result was yielded when the expansion factor T was set to 6.

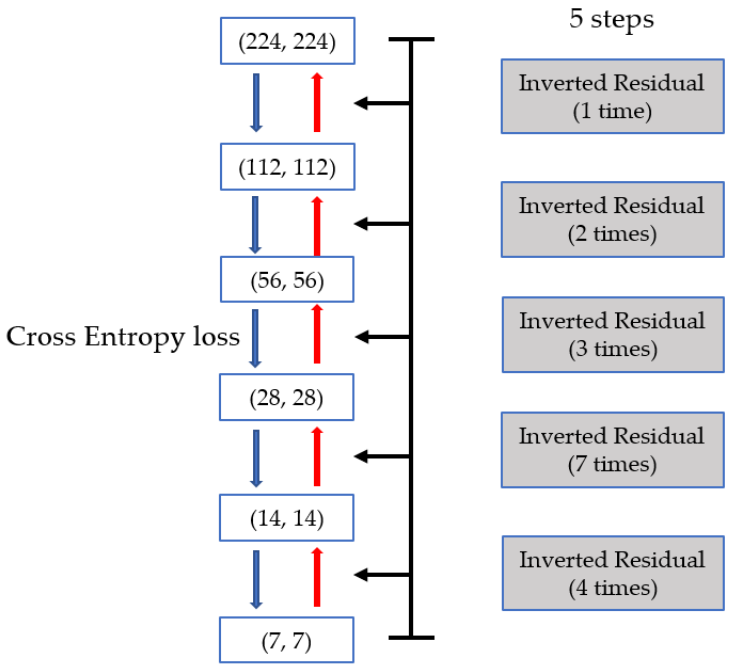

Figure 7. Proposed Mobile-Unet architecture.

Table 1. Contraction processing of our Mobile-Unet.

Input Size	Convulsions	T	C	N	S
224 × 224 × 3	Conv2D 3 × 3	-	32	1	2
112 × 112 × 32	Bottleneck1	1	16	1	1
112 × 112 × 16	Bottleneck2	6	24	2	2
56 × 54 × 24	Bottleneck3	6	32	3	2
28 × 28 × 32	Bottleneck4	6	64	4	2
14 × 14 × 16	Bottleneck5	6	96	3	1
14 × 14 × 96	Bottleneck6	6	160	3	2
7 × 7 × 160	Bottleneck7	6	320	1	1
7 × 7 × 320	Conv2D 1 × 1	6	1280	1	1
7 × 7 × 1280	-	-	-	-	-

Table 2 shows the process of the Mobile-Unet expansion in detail. The number of repetitions was set to 1, and the image size was doubled for each inverted residual operation. We set the expansion ratio to 6. The final step calculates the score from 224 × 224 × 16 to 224 × 224 × 3 and then to 224 × 224 × 1.

Table 2. Expansion processing.

Input Size	Convulsions	T	C	N	S
7 × 7 × 1280	D_Bottleneck1	6	96	1	2
14 × 14 × 96	D_Bottleneck2	6	32	1	2
28 × 28 × 32	D_Bottleneck3	6	24	1	2
56 × 56 × 24	D_Bottleneck4	6	16	1	2
112 × 112 × 16	D_Conv2D 4 × 4, pad = 1	-	16	1	2
224 × 224 × 16	Conv2D 1 × 1	-	3	1	-
224 × 224 × 3	Conv2D 1 × 1	-	1	1	-
224 × 224 × 1	-	-	-	-	-

4. Experimental Results

Here, the real-time hair segmentation using the Mobile-Unet method proposed in this study was evaluated. In our experiments, the hair segmentation datasets, including ETRIHair data, the experimental environment and results, and performance comparison are described.

4.1. Hair Segmentation Datasets

Public hair segmentation datasets that include hair-partitioned masks are relatively small, which makes it difficult to collect enough data to train with DNNs. The effort and time required to build hair segmentation datasets are greater than when building another training dataset for the face or human body because it is particularly difficult to divide thin hair areas into different hair segmentation areas to create training target masks. The currently released hair segmentation datasets [17,18] employ some of the existing large face datasets and manually partition hair partitions to build the training dataset. Table 3 shows the sets of data we used for training to develop the proposed algorithm.

Table 3. Our training datasets for hair segmentation.

Datasets	# of Images	From	Align	Permission
Parts Labels [17]	3.0 K	LFW	None	Public
CelebBHair [18]	3.5 K	CelebA	Automatic Align	Public
ETRIHair	1.3 K	Googling	None	Private

The first dataset was collected by extracting some data from the Labeled Faces in the Wild (LFW) dataset and labeling a hair region. This dataset, called the Parts Labels dataset [17], has a total of three labeled areas—hair, background, and face—and consists of about 2900 images that are not pre-processed and are publicly available. The pixel value of the hair areas in Parts Labels is set to 255 and the pixels of the other mask image for the rest of the areas are set to 0 to handle the binary division problem. The second dataset, called the CelebBHair dataset [18], was collected by applying a hair mask to some images selected from the CelebA face datasets. This dataset consists of about 3500 images and has the same background, face, and hair area divisions as Parts Labels. The same pre-processing used on the Part Labels datasets was also carried out on these images for use in this study.

The third ETRIHair dataset collected about 1200 face data of Asian people and the hair segmentation was marked manually. The reason for also collecting Asian hair data is that the majority of the two previously released datasets consists of European hair data. After the initial training, we tested the hair segmentation DNN model using only the first two datasets. The accuracy for Asians was measured to be about 30% lower than that for Europeans based on an interaction of union (IOU) evaluation. If the dataset is developed using the proposed pre-processing method on these two datasets, the data will consist of approximately 6000 images. Many data items were excluded as there was a large number of noise images, which are those that do not include the hair region or occluded the hair region or images with faces that could not be detected by the face detector [14], such as those where most of the face was covered by hair or the back of the head. To address this issue, the data could have been corrected using a hair segmentation model that was already developed [3,9], but there was no guarantee that the results would be consistently well-segmented in various environments. Therefore, the hair segmentation was carried out using manual image segmentation software. Despite these tasks, the final set of data used for training the DNN model, excluding non-calibrating data, consisted of 6800 images with 1000 images for verification.

4.2. Experimental Environment and Results

To develop the DNN model, we used the PyTorch deep learning framework in Python. The GPUs we used for training employed parallel training with two NVidia GTX TITAN-Xs.

The input images went through the normalization process after switching to Tensor, the batch size was set to 128, and for the loss, we used BCELoss. We used the Stochastic Gradient Descent (SGD) optimizer (learning rate = 0.001, weight decay = 1×10^{-8}, momentum = 0.9 and nesterov = "True"). Of the 95,200 images obtained for hair segmentation, we randomly selected 90% for training and 10% for testing. Table 4 shows the processing time for each module. As shown in Table 4, the proposed Mobile-Unet module had almost the same speed as face detection, indicating that real-time responses are possible. Our hardware specification for the inference test is a notebook with an Intel I7-8750H CPU, 16 G of memory, and a GTX1060 graphics card.

Table 4. Processing time of each module.

Module	Time (ms)
Face detection	13
68 landmark detection	6
Hair segmentation	13
Total	32

The accuracy of hair segmentation can be evaluated by various criteria. Here, we made measurements based on the Dice coefficient and IOU evaluation criteria. Table 5 shows the results of racial hair segmentation. As described in Section 4.1, our study obtained segmentation results that were similar to those for Europeans, as we learned by adding an Asian database to the evaluation. However, Asian hair is still less accurate in segmentation than European hair. The reason for this is that additional Asian training data from ETRIHair were used, but the overall training data rate was smaller than that of Europeans. Moreover, Asians have a lot of short hairstyles for both men and women, because the area of hair is relatively small compared to the entire face, so even untrained a slightly different hairstyle increased the IOU error.

Table 5. Accuracy of racial hair segmentation.

Criterion	European (500)	Asian (500)
Dice coefficient	94.7%	86.2%
IOU	89.9%	81.1%
Average	92.3%	83.7%

Table 6 shows that the accuracy for men's hair is generally high, at 93.8% on average, because men's hair is relatively less diverse. In contrast, women's hair has a relatively wide variety of hairstyles. Additionally, women's hair has a wide variety of hair colors, and if similar hair colors do not exist in the training dataset hair segmentation does not work well. The difference in accuracy between men and women based on the IOU criterion was 11.6%.

Table 6. Accuracy of gender hair segmentation.

Criterion	Man (500)	Women (500)
Dice coefficient	97.1%	85.4%
IOU	90.4%	78.8%
Average	93.8%	82.1%

4.3. Performance Comparison

Figure 8 compares the performance of the recently proposed hair segmentation methods and the proposed methods. We used the IOU performance criterion. In most cases, the results show good performance based on the CELEbA test dataset. Compared with Unet [15], the accuracy is similar to our method, but the inference time is over ten times

slower. The highest method in terms of accuracy was DeepLab V3 [19], which achieved an accuracy of 91.2% based on the CELEbA test dataset, but they had two stages (hair segmentation and a hair detection stage). If they do not use a hair detection stage, the accuracy rate is 89.9%, which is the same score as our proposed method. DeepLab V3's execution time was 55 ms, so it was not fast when compared with our models in Figure 8. The best method in terms of execution time was the Tkachenka [20] method, which had the fastest real-time speed of 6 ms. However, the accuracy was only 80.2%; the trade-off between speed and accuracy was obvious. The Mobile-Unet proposed in this study achieved an accuracy of 89.9% on the CELEbA test dataset. The execution time was 13 ms, which is the second-fastest among the methods shown on the graph. Thus, considering the proposed Mobile-Unet method's accuracy and execution time together, we can see that it is more efficient than the other methods.

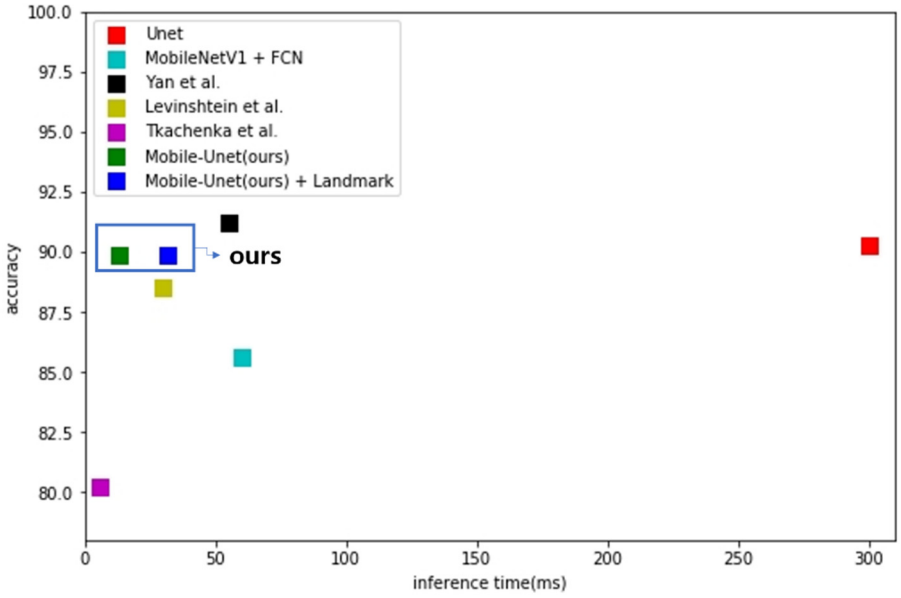

Figure 8. Comparison of the proposed Mobile-Unet method's performance.

Table 7 shows the comparison of the computational and memory complexity between DeepLab V3, the original Unet, and the proposed method. The proposed method measures approximately 4.2 times faster than DeepLab V3 in inference time and the number of parameters that determine memory complexity use approximately 4.2 percent only that of DeepLab V3.

Table 7. Comparison of the computational and memory complexity.

Models	Inference Time (ms)	# of Parameters (MB)
DeepLab V3 + ResNet18	55	58.62
Unet	307	31.03
Proposed Method	13	2.46

Figure 9 visually depicts the results of hair segmentation on test datasets obtained using the proposed method. For baldness, other algorithms often show the wrong results, but in our method, the results are output without hair segmentation results.

Figure 9. Good results of the proposed method.

Figure 10 shows the poor results obtained by our method, which occurred because of noise regions such as a hand, hairbrush, or bounded input images.

Figure 10. Poor results of the proposed method.

5. Conclusions

We have described a real-time hair segmentation method based on the Mobile-Unet model with optimization techniques of depth-wise separable convolution and inverted residual block technology. Optimization is an important task for real-time processing in mobile devices and embedded edge computing. In general, increasing the number of layers in deep neural networks can improve recognition performance, but it also increases processing time. Therefore, a balance between performance and speed is required. The proposed algorithm has balance in terms of performance and speed. In experiments, the results produced an average hair segmentation accuracy of 89.9% with a 32 ms processing time for different genders and races. By comparing the accuracy and execution speed with other models in related studies, we confirmed that the proposed model achieved equal performance and better speed. Additionally, the proposed method showed similar performance on Asian hair segmentation to European hair segmentation using the proposed ETRIHair dataset and augmentation approach. In addition, there was a performance rate decrease of more than 20% in the experiment if the proposed Asian datasets were not used. Furthermore, we may study new models for more accurate and detailed hair segmentation by adding new optimization algorithm such as knowledge distillation.

Author Contributions: Conceptualization: H.-S.Y.; Data curation: S.-W.P.; Funding acquisition: H.-S.Y. and J.-H.Y.; Investigation: S.-W.P. and H.-S.Y., Project administration: H.-S.Y. and J.-H.Y.; Writing—original draft, S.-W.P. and H.-S.Y.; Writing—review & editing, H.-S.Y. and J.-H.Y. All authors have read and agreed to the published version of the manuscript.

Funding: This research was supported by an Institute of Information and Communications Technology Planning and Evaluation (IITP) grant funded by the Korean government (MSIT) (2019-0-00330, Development of AI Technology for Early Screening of Infant/Child Autism Spectrum Disorders based on Cognition of the Psychological Behavior and Response), and a Korea Evaluation Institute of Industrial Technology (KEIT) grant funded by the Korean government (MOTIE) (0077553, Development of Social Robot Intelligence for Social Human–Robot Interaction of Service Robot).

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Data available on request due to privacy. The data presented in this study are available on request from the corresponding author.

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

References

1. Onyeulo, E.B.; Gandhi, V. What Makes a Social Robot Good at Interacting with Humans? *Information* **2020**, *11*, 43. [CrossRef]
2. Breazeal, C.; Scassellati, B. Robots that imitate humans. *Trends Cogn. Sci.* **2002**, *6*, 481–487. [CrossRef]
3. Chai, M.; Shao, T.; Wu, H.; Weng, Y.; Zhou, K. Autohair: Fully automatic hair modeling from a single image. *ACM Trans. Graph.* **2016**, *35*. [CrossRef]
4. Yacoob, Y.; Davis, L. Detection, analysis and matching of hair. In Proceedings of the Tenth IEEE International Conference on Computer Vision (ICCV), Beijing, China, 17–21 October 2005; pp. 741–748.
5. Wang, D.; Chai, X.; Zhang, H.; Chang, H.; Zeng, W.; Shan, S. A novel coarse-to-fine hair segmentation method. *J. Softw.* **2011**, *24*, 233–238.
6. Wang, D.; Shan, S.; Zhang, H.; Zeng, W.; Chen, X. Data driven hair segmentation with isomorphic manifold inference. *Image Vis. Comput.* **2014**, *32*, 739–750. [CrossRef]
7. Guo, W.; Aarabi, P. Hair segmentation using heuristically-trained neural networks. *IEEE Trans. Neural Netw. Learn. Syst.* **2016**, *29*, 25–36. [CrossRef] [PubMed]
8. Long, J.; Shelhamer, E.; Darrell, T. Fully convolutional networks for semantic segmentation. In Proceedings of the 2015 IEEE Conference on Computer Vision and Pattern Recognition, Boston, MA, USA, 7–12 June 2015; pp. 3431–3440.
9. Qin, S.; Kim, S.; Manduchi, R. Automatic skin and hair masking using fully convolutional networks. In Proceedings of the 2017 IEEE International Conference on Multimedia and Expo (ICME), Hong Kong, China, 10–14 July 2017; pp. 103–108.
10. Zhang, L.; Wang, P.; Shen, C.; Liu, L.; Wei, W.; Zhang, Y. Anton van den Hengel Adaptive importance learning for improving lightweight image super-resolution network. *Int. J. Comput. Vis.* **2020**, *128*, 479–499. [CrossRef]
11. Seyed-Iman, M.; Mehrdad, F.; Ang, L.; Nir, L.; Akihiro, M.; Hassan, G. Improved Knowledge Distillation via Teacher Assistant. *arXiv* **2019**, arXiv:1902.03393.
12. Sandler, M.; Howard, A.; Zhu, M.; Zhmoginov, A.; Chen, L. MobileNetV2: Inverted Residuals and Linear Bottlenecks. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Salt Lake City, UT, USA, 18–23 June 2018; pp. 4510–4520.
13. Jang, J.; Jeon, S.H.; Kim, J.; Yoon, H. Robust Deep Age Estimation Method Using Artificially Generated Image Set. *ETRI J.* **2017**, *39*, 643–651. [CrossRef]
14. Wei, L.; Dragomir, A.; Dumitru, E.; Christian, S.; Scott, R.; Fu, C.-Y.; Berg, A.C. SSD: Single Shot MultiBox Detector. *arXiv* **2016**, arXiv:1512.02325v5.
15. Ronneberger, O.; Fischer, P.; Brox, T. U-net: Convolutional networks for biomedical image segmentation. In Proceedings of the 18th International Conference on Medical Image Computing and Computer-Assisted Intervention, Munich, Germany, 5–9 October 2015; pp. 234–241.
16. Levinshtein, A.; Chang, C.; Phung, E.; Kezele, I.; Guo, W.; Aarabi, P. Real-time deep hair matting on mobile devices. *arXiv* **2018**, arXiv:1712.07168v2.
17. Kae, A.; Sohn, K.; Lee, H.; Learned-Miller, E. Augmenting CRFs with Boltzmann Machine Shape Priors for Image Labeling. In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, Portland, OR, USA, 23–28 June 2013; pp. 2019–2026.
18. Ileni, T.; Borza, D.; Darabant, A.S. A Deep Learning Approach to Hair Segmentation and Color Extraction from Facial Images. In Proceedings of the ACIVS 2018: Advanced Concepts for Intelligent Vision Systems, Poitiers, France, 24–27 September 2018; pp. 438–449.
19. Yan, Y.; Berthelier, A.; Duffner, S.; Naturel, X.; Garcia, C.; Chateau, T. Human Hair Segmentation in the Wild Using Deep Shape Prior. In Proceedings of the Third Workshop on Computer Vision for AR/VR, Long Beach, CA, USA, 17 June 2019; pp. 1–4.
20. Tkachenka, A.; Karpiak, G.; Vakunov, A.; Kartynnik, Y.; Ablavatski, A.; Bazarevsky, V.; Pisarchyk, S. Real-time Hair Segmentation and Recoloring on Mobile GPUs. In Proceedings of the CVPR Workshop on Computer Vision for Augmented and Virtual Reality, Long Beach, CA, USA, 17 June 2019; pp. 1–3.