Dokuz Eylül Üniversitesi Mühendislik Fakültesi Fen ve Mühendislik Dergisi
Dokuz Eylul University Faculty of Engineering Journal of Science and Engineering

Basılı/Printed ISSN: 1302-9304. Elektronik/Online ISSN: 2547-958X

Segmentation of Portrait Images Using A Deep Residual Network Architecture

Derin Kalıntı Ağ Mimarisi Kullanarak Portre Görüntülerinin Bölütlenmesi

Taner Danışman [1*]

[1] Akdeniz University, Faculty of Engineering, Computer Engineering Department, Antalya, TURKEY
Sorumlu Yazar / Corresponding Author *: tdanisman@akdeniz.edu.tr

Geliş Tarihi / Received: 12.11.2019 *Araştırma Makalesi/Research Article*
Kabul Tarihi / Accepted: 04.02.2020 *DOI: 10.21205/deufmd.2020226523*
Atıf şekli/ How to cite: DANIŞMAN, T.(2020) Segmentation of Portrait Images Using a Deep Residual Network Architecture. DEUFMD, 22(65), 569-580.

Abstract

Segmenting portrait images into semantic areas is an important step towards scene understanding and image analysis. Although segmentation is a very active field of study, there are few studies in the field of portrait segmentation. One of the most crucial steps in portrait segmentation is the precise segmentation process where semantically related pixels grouped together including hair, face, body, and background. However, this is a challenging problem due to the extreme variations in hair shape, color, and background. In order to handle such variations, we proposed a deep residual network based on ERFNet architecture. We used geometrically normalized faces as an input for the network. Experimental studies on EG1800 dataset (two-classes) and LFW Part Labels Dataset (three-classes) showed that the proposed method provides state of the art mIoU (mean intersection over union) and pixel-based accuracy. We obtained 96.37% mIoU and 98.17% pixel based accuracy for EG1800 dataset and 90.1% mIoU and 97.14% accuracy for the LFW dataset.

Keywords: *Portrait Segmentation, Deep Learning, Deep Residual Networks, Geometric Normalization, Encoder Decoder Networks*

Öz

Portre görüntülerini anlamsal alanlara bölütlemek, sahne anlama ve görüntü analizinde önemli bir adımdır. Bölütleme çok aktif bir çalışma alanı olmakla birlikte, portre bölümlendirme alanında az sayıda çalışma bulunmaktadır. Portre bölütlemesindeki en önemli adımlardan biri, saç, yüz, gövde ve arka plan gibi anlamsal olarak ilişkili piksellerin birlikte gruplandığı, detaylı bölütleme işlemidir. Ancak, saç şekli, rengi ve arka planındaki aşırı farklılıklar nedeniyle bu zor bir problemdir. Çalışmamızda, bu çeşitliliklerin üstesinden gelmek için ERFNet mimarisine dayanan derin bir kalıntı ağı önerdik. Geometrik olarak normalleştirilmiş yüzleri ağ için bir girdi olarak kullandık. İki sınıflı EG1800 veri kümesi ve üç sınıflı LFW Parts Labels Veri Seti üzerinde yapılan deneysel çalışmalar, önerilen yöntemin yüksek doğrulukta ortalama kesişim değeri (mIoU) verdiğini ve piksel tabanlı doğruluğu sağladığını göstermiştir. EG1800 veri kümesi için %96,37 mIoU ve % 98,17 piksel tabanlı doğruluk ve LFW veri kümesi için %90,1 mIoU ve %97,14 doğruluk elde ettik.

Anahtar Kelimeler: *Portre Bölütleme, Derin Öğrenme, Derin Kalıntı Ağlar, Geometrik Normalleştirme, Kodlayıcı Kod Çözücü Ağlar*

1. Introduction

In recent years, selfie and self-portrait images become more popular among mobile users. According to Google Inc. 24 billion selfie images uploaded to their servers in 2015. Semantic segmentation of different parts of photographs can be used to develop existing methods based on face analysis as well as tasks such as portrait editing and manipulation. The precision of the segmentation is also an important feature for different domains including the defense industry, remote sensing, intelligent vehicle technologies, and microscopic images. However, basic approaches to the segmentation problem such as threshold method are not enough to solve the visual challenges. Therefore, researchers use Artificial Intelligence methods as a machine learning approach, which achieve high success rates, especially in image segmentation problem.

Artificial Intelligence aims to teach human behaviors to machines by using machine learning methods. Convolutional Neural Networks (CNN) is one of the most successful machine learning method used for image segmentation task. They have been applied to image segmentation problems since the 1980s and recently get more interest due to their high success rates. Although Deep CNN's are successful, they must be well designed for the given problem. There are many factors that affect the performance of the network such as the depth of the network, size and number of convolution filters, augmentation (artificially creating the training images through rotations, shifts, flips) and regularization (modifications on the learning algorithm to reduce the generalization error [1]). Theoretically, more layers provide better representation of the input data. For this reason, it is expected that the multi layered network with more layers can solve the more complex problems. Therefore, the depth of the network has great prominence. However, more layers will also bring the over-fitting problem and increases the learning duration. It is also likely that the learning error rate is going to increase when more layers added [2]. A solution to this problem is the use of early stop, dropout techniques and Deep Residual Networks (DRN) [3]. Early-stop and dropout techniques can be applied to any CNN. However, DRNs need architectural changes to allow the transfer of data to deeper layers. Compared to the traditional CNN's, the DRNs are easy to optimize, and they provide better segmentation results with deeper nets.

There are different taxonomies exist in the object segmentation problem. Mainly the segmentation methods are classified into Layer based and Block based segmentation methods [4]. Layered methods focus on object detectors and depth ordering. Block based methods use image intensity values, edge/color discontinuities and regional features for the segmentation. It can be further divided into intensity based, region based, edge and boundary based methods. A list of unsupervised image segmentation studies can be found in [5]. In this study, we did not cover segmentation based on hand crafted features (e.g. SIFT, HOG).

Threshold based methods aim to segment the image over pixel densities using the best threshold value. It is a simple method, but it performs poorly in semantic areas with different textures. The most known threshold method in the literature is the Otsu method [6]. This method assumes that there are two classes (foreground and background) in the image. It aims to find the most suitable threshold value with the lowest variance in the class and the largest variance between the classes. However, the success of the Otsu method is limited in cases where one of the classes of pixels in the image is relatively less than the other. If the average gray density difference between classes is small, then this method may also fail. In addition, it provides only two class segmentation and it does not consider the semantic integrity of the regions. [7] used histogram peaks distribution (HPD) and quantum evolution algorithm for portrait segmentation problem. However, they segment portraits from a blue background only.

Conditional Random Fields (CRF) based models are also well studied methods for the object segmentation problem [8–12]. CRFs use Conditional Probability Distributions of label sequences, and thus they are similar to the Logistic Regression. A hierarchical CRF model for object segmentation problem is proposed by [13]. They integrate features derived from different quantization levels, which improve quality of segmentation. Graph Cut method is proposed in [14] where the image is treated as a graph. Max-flow/min-cut algorithms determine the pixel labels. Like other graph based methods, this method requires manual background and foreground segmentation. FaceCut method [15], focuses on accurate facial feature segmentation using Modified Active Shape Models (MASM). However, ASM is a deformable template based approach and can even be affected by the user's facial expression. In another study, [16] used deep learning along with one class SVM to detect the human facial regions. They used region refinement and MASM to refine the segmented regions on LFW dataset. These methods focus only on the inner face area and do not address the challenging hair segmentation problem.

The aforementioned methods do not take into account the semantic integrity of the segmented regions. Therefore, an AI based solution is needed. Deep convolutional networks automatically detect key features in the input image through convolutions. Deep CNN's and Deep Residual Networks (DRNs) can achieve higher success rates than traditional methods especially for multi class image segmentation problems. The major deep architectures used in the literature for the image segmentation problem is as shown in Table 1.

Table 1. Major deep learning architectures.

Network Architecture	*# of Layers*	*Year*
LeNet5 [17]	7	1998
AlexNet [18]	8	2012
VGGNet [19]	19	2014
GoogLeNet [20]	22	2015
ResNet [3]	152	2016
ResNeXt [21]	101	2016
SegNet [22]	37	2017
ERFNet [23]	23	2018

In general, a CNN designed for the object segmentation problem consists of a set of layers for extracting, learning and classifying pixels. It is composed of Convolutional, ReLU (REctified Linear Unit), Pooling, Deconvolution Up Sampling) and SoftMax layers respectively.

The convolution and pooling layers are repeated several times with respect to the size of the input image to extract the important features. The ReLU layer applies the function in (1) in all elements on an input x, without changing its spatial or depth information. Another said all positive elements in x remain unchanged while the negatives become zero. The ReLU function is advantageous in terms of computational needs. Therefore, the majority of the deep learning methods use it as an activation function.

$$f(x) = \max(0, x) \qquad (1)$$

A deep residual encoder decoder network composed of three sections. These are factorized residual network modules with dilations, down sampling, and up sampling. Residual network with dilations increases the resolution of output feature maps without reducing the receptive field of individual neurons [24]. Down sampling in the encoder supports multiple branching and creates two branches where max pooling and convolution are applied to the input image. The results then fused to obtain the output. Up sampling provides inverse convolutions in the decoder to obtain the original size of the input image.

Although there are numerous studies exist for general semantic image segmentation especially on Pascal VOC tasks [25] there a few studies focusing on portrait segmentation problem. [26] proposed PortraitFCN+ which is a fully automatic segmentation for portrait images. They extend the FCN-8s framework [27] to use the domain knowledge. The input image is segmented into foreground and background classes. In order to solve different geometric positions of the portrait faces, they introduced x and y channels representing the positions of the pixels relative to the face. They also report that the use of a shape channel improves the segmentation results. By using augmentation techniques, they generate 19,000 training samples from 1,500 Flickr photos. They obtained 95.91% of mIoU on their EG1800 dataset.

[28] use Portrait Mode in their AI supported cameras on Android mobile devices (Google Pixel series) to segment foreground and background objects to provide optical depth of field effect. Part of their work that provides foreground and background segmentation is available in DeepLab v3+ API. They used 3 stacked U Nets, on 4 channel input images (RGB and encoded face mask) for person segmentation. They obtained 97.01% mIoU on EG1800 test using the EG1800 training set (1.5K images). When they use their own training set

(465K images), they obtained 97.7% mIoU on EG1800 test set.

[29] used convolutional networks to extract features from regions based on SLIC super pixels. In order to preserve spatial context, they used the zoom out technique by defining local, proximal, distant and scene levels. A more detailed review on deep learning techniques applied to semantic segmentation problem can be found in [30].

In this study, we proposed a deep residual network inspired by the ERFNet [23] architecture to segment portrait images into semantic regions. Original ERFNet architecture focuses on visual semantic segmentation for intelligent vehicles. Since we have fewer classes in portrait segmentation problem, we simplified the architecture to make it more suitable for boundary-based object segmentation. Existing methods like Conditional Random Fields (CRF) models cannot perform long range connections and among different parts of the object which results in excessive smoothing of object boundaries [31]. Similar problems exist in the deconvolution layers of the Deep CNN's such as dilated convolutions produce gridding artifacts. Therefore, we designed a simplified architecture to improve the coarse segmentation results. The main contribution of our study is three fold:

- We proposed a modified ERFNet architecture for the portrait image segmentation task.
- We showed that combined use of the IPD (Interpupillary Distance) based normalization and augmentation techniques improve the segmentation results for the portrait segmentation problem.
- We obtained 96.37% mIoU score and 98.16% pixel based accuracy on the EG1800 Segmentation dataset.

Section 2 presents the details of the proposed study and deep residual architecture. The datasets, our preprocessing methodology, and the environment also presented in Section 2. Section 3 covers experimental results obtained on LFW Part Labels [32] and EG1800 [26] dataset. Section 4 concludes the study with a discussion and conclusion.

2. Material and Method

In this study, we formulate a Deep Residual Network as illustrated in Figure 1 for portrait segmentation problem whose core architecture is inspired by recent ERFNet architecture proposed in [23].

In the encoder decoder type of networks, there are two steps in the training process. Training of the encoder for extracting information and training of the decoder using the encoder model for segmenting the image. The encoder computes a coarse segmentation of the input data and extracts valuable information from the input image through a variety of convolutional filters.

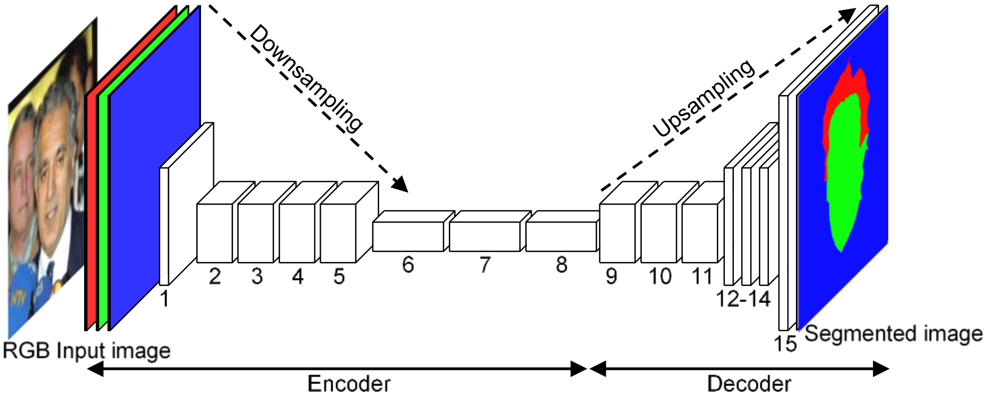

Figure 1. The encoder decoder deep residual network architecture used in the study.

Table 2. Details of the deep layers used in experiments.

Network		Type	Input Feature	Output Feature	Output Resolution for LFW/EG1800	
Encoder		Input	3	3	256×256	152×200
	1	Downsampler	3	16	128×128	76×100
	2	Downsampler	16	64	64×64	38×50
	3-5	3 × Residual Block	64	64	64×64	38×50
	6	Downsampler	64	128	32×32	19×25
	7-8	3 × Residual Block	128	128	32×32	19×25
Decoder	9	Upsampler	128	64	64×64	38×50
	10-11	2 × Residual Block	64	64	64×64	38×50
	12	Upsampler	64	16	128×128	76×100
	13-14	2 × Residual Block	16	16	128×128	76×100
	15	Upsampler	16	3	256×256	152×200

Decoder network performs up scaling of the image obtained from the encoder. Table 2 presents the details of the deep layers used in LFW and EG1800 dataset experiments. The encoder utilizes 3×3 filters over a 256×256 and 152×200 input image for the LFW and EG1800 dataset respectively. The three downsampler blocks in the encoder reduce the image size to one eighth of the initial size of the image. Therefore, the output of the encoder for an image with a resolution of 256×256 is 32×32 (19×25 for EG1800). Because of this low resolution, we need to upscale it properly to obtain fine segmented results using the decoder. We have reduced the EG1800 sample size from 600×800 to 152×200 to reduce complex calculations.

The main job of the decoder is then to learn a way to map the low resolution output of the encoder to pixel based predictions at higher resolution space using upsampler blocks, original data, and the encoder. The upsampling is a nonlinear process for low resolution input feature maps. In order to provide object coherency (e.g. elimination of holes in segmented areas), the decoder network equipped with a sequence of dilation layers. Finally, it provides the output as the same size as the input image.

The objective of the residual blocks is to ensure that the information contained in the previous layer is transferred to the subsequent layers. It is aimed that the general information is not lost in the feature extraction layers. Residual blocks allow us to create deeper networks while transferring the same input to the next layer with the encoded information. The input is transmitted to the next layer without any convolution so that the added new layer can learn the problem being solved. In this way, the new layer can learn different knowledge than the previous layers learned. The transfer creates a deeper and still learnable architecture. Each residual block (Layer 3-5, 7-8, 10-11 and 13-14) utilizes 3×1, 1×3, 3×1 and 1×3 filters over the input. These 1D factorized convolutions both provide information transfer and reduce computational complexity. Original ERFNet architecture contains a total of 23 layers (encoder: 16 layers, decoder: 7 layers) for twenty class segmentation task. Since we focus on two class (foreground and background in EG1800 dataset) and three class (background, hair, and skin in LFW dataset) segmentation problem, our modified network composed of 15 layers (encoder: 8 layers, decoder: 7 layers).

The learning architecture is finalized by the Softmax layer. It computes the likelihood that the input image belongs to a particular class by using the values generated by the network in multi class classification problems. Besides that, a dropout value of 0.02 is applied to the hidden layers. It randomly removes nodes and their incoming and outgoing connections from the network. As a result, the model becomes robust and insensitive to the weights of the other nodes thus it can generate the more generalized model.

2.1. Dataset

We used the original LFW Part Labels [32] and EG1800 dataset [26] in our experiments. The LFW dataset contains a total of 2,927 face images of size 250×250 pixels and corresponding background, skin, and hair labels. Manual annotations are based on super pixels performed on funneled images. We used 2,587 images for training and 100 images for testing. The train set is composed of 1,798 male photos and 609 Female photos. The test set is equally split in terms of gender.

The original EG1800 dataset is a URL based dataset that contains a total of 1,800 images of size 600×800 pixels where 1,500 images are reserved for training and the remaining 300 images are reserved for testing. These images were manually annotated using Photoshop quick selection tool. Because of this, there are annotation errors in the segmentation data as shown in Figure 2. Besides that, some image URLs are not available anymore. Therefore, in this study, we used 1,589 of them.

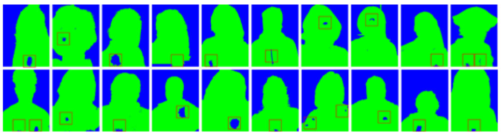

Figure 2. Random annotation errors from EG1800 dataset. Small square regions show the location of the annotation errors.

After the preprocessing step we obtained a total of 1,502 images. 1,352 of them are used for the training and 150 of them are used for testing.

2.2. Preprocessing

The first step is the normalization of the input image. Although the deep learning based methods skip positional and geometric normalizations, [28] showed that face masks and additional information increase the segmentation accuracy. We performed several geometric normalizations to normalize the approximate distance of the faces to the camera. Viola Jones face detector [33] is used for face detection. For eye detection, we used a neural network based eye detector [34] available in STASM library [35]. Note that we did not directly use the funneled images and the annotations. Instead, we performed our own geometric normalization method on both the funneled images and the annotations based on eye centers as shown in Figure 3.

The first row in Figure 4 shows the funneled images from LFW Part Labels dataset and the second row shows our normalized samples using the proposed method.

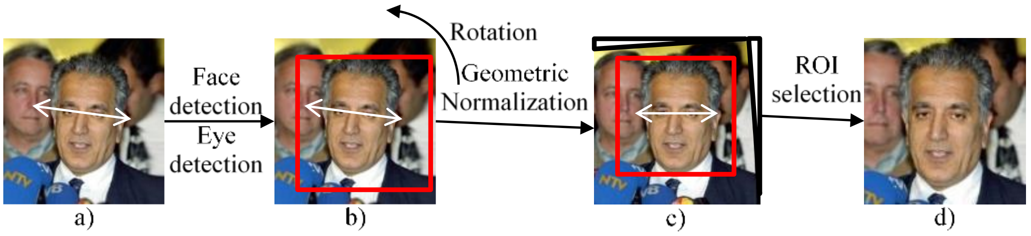

Figure 3. Geometric normalization steps in preprocessing. a) Face and eye detection. b) Region of Interest (ROI) selection with respect to interpupillary distance (IPD). c) Geometric normalization and ROI selection. d) Final result.

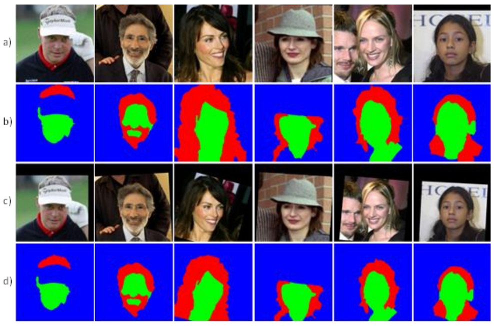

Figure 4. a) Input images from LFW. b) Ground truth segmentation data. c) Geometric normalization results obtained from input images. In these images, left and right eye locations are automatically fixed to known positions using the eye detection algorithm. d) Normalized ground truth images.

Since we modified the funneled images with our geometric normalization method, we also performed the same geometric normalization steps for the ground truth data including rotation and cropping.

2.3. Environment

We performed our experiments using Torch Deep Learning Framework Torch-7 [36] on an Ubuntu OS (18.04) with CUDA 9.2 (Nvidia GTX 1060 6GB RAM). OpenCV implementation of Viola Jones face detector and neural network based eye detector is used in the normalization step.

2.4. Evaluation metrics

We used commonly accepted mIoU (Mean Intersection over Union) and total pixel based accuracy metrics to measure the quality of the segmentation process. mIoU metric also known as the Jaccard similarity coefficient which is the size of the intersected pixels divided by the size of the union of the pixels as shown in (2) where Fg and Bg denote predicted foreground and background pixels respectively.

$$IoU(Fg, Bg) = \frac{\mathrm{Fg} \cap \mathrm{Bg}}{\mathrm{Fg} \cup \mathrm{Bg}} \qquad (2)$$

We compute the IoU score for foreground and background classes separately and then averaged over all classes to create the mIoU score. It takes into account both the false alarms and the missed values for each class.

In order to compute the pixel based accuracy, we consider the correctly classified pixels for each class divided by the total number of pixels as shown in (3)

$$Accuracy = \frac{\mathrm{TP} + \mathrm{TN}}{\mathrm{TP} + \mathrm{TN} + \mathrm{FP} + \mathrm{FN}} \qquad (3)$$

3. Results

We performed our experiments on LFW and EG1800 datasets. For both of the dataset, we follow the same protocol except the SoftMax layers (two-classes vs three-classes).

3.1. Experiments on LFW dataset

In the encoder, the input image size is set to be 256×256 pixels and maximum epoch is set to 100 epochs. Note that the original image size in LFW dataset is 250×250 pixels. Since our downsampling layers shrink the initial image to the one eighth of the initial size, the objective here is to select a dimension that can be divisible by eight to get the maximum benefit from the convolutions in the encoding step. The learning rate is initially set to 5×10^{-4} and it is divided by 2 into every 50 epochs. Dropout is set to 0.2. We performed 100 epochs in the encoder training and 200 epochs for the decoder training. Table 3 presents detailed results.

According to Table 3, our method provides higher segmentation accuracy and mIoU value for all segmentation classes. Since we use the same deep residual architecture and configuration for both of the methods, the main reason for the difference is the geometric normalization which provides better alignment of the patterns. Figure 5 shows the results obtained during the training and testing of encoder and decoder. It shows that the use of the normalized images provides higher segmentation accuracy than funneled images.

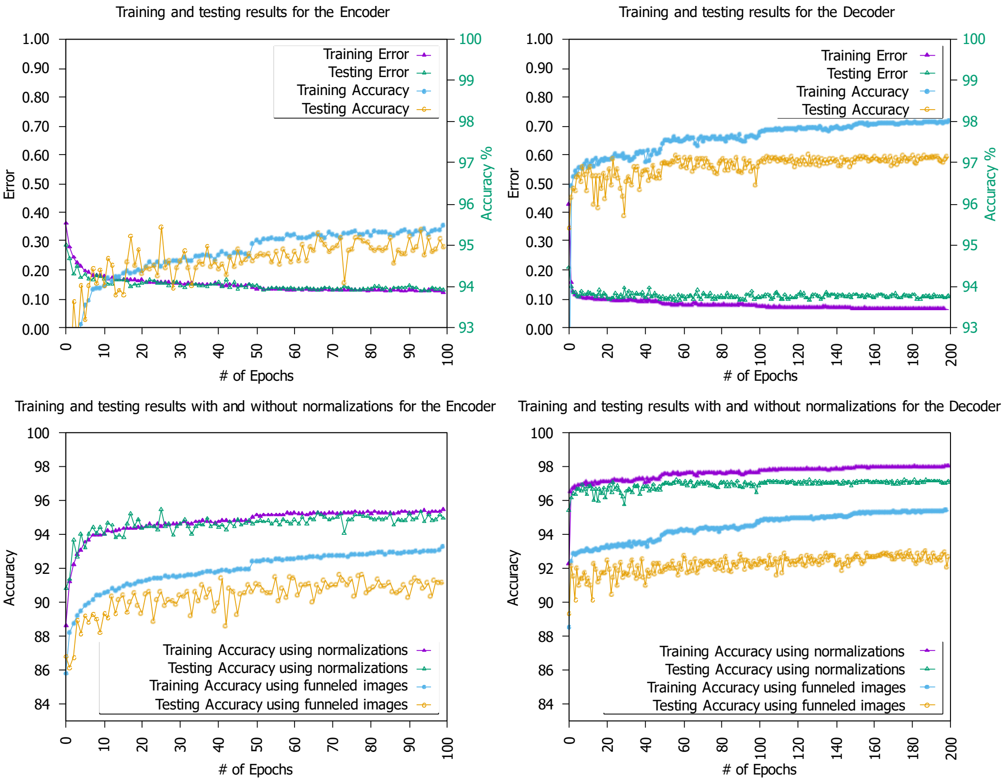

Figure 5. Training error, testing error and the accuracy for the encoder and decoder on the LFW Dataset. Use of the normalized images boosts up the training accuracy for both the encoder and the decoder.

We obtained 95.48% training accuracy and 94.94% testing accuracy for the encoder. The highest IoU values obtained from the encoder in the training and testing are 84.62% and 83.74% respectively. For the decoder, we obtained 97.99% training accuracy and 97.14% testing accuracy. The highest mIoU values for the training and testing of the decoder is 92.39% and 90.10%. In order to find the effect of our IPD (Interpupillary Distance) based normalization to the portrait segmentation problem, we also performed the same tests on the funneled LFW Part Labels dataset [37] without our normalization step. Note that funneled dataset is composed of aligned images with respect to image funnel. .

Table 3. Details of the segmentation results used in this study.

		Encoder					Decoder				
		mIoU	Skin	Hair	Bg.	Acc.	mIoU	Skin	Hair	Bg.	Acc.
Our Method	Train	84.62	94.53	89.12	96.16	95.48	92.39	97.09	95.53	98.27	97.99
	Test	**83.74**	**94.16**	**90.61**	**96.01**	**94.94**	**90.10**	**95.79**	**93.98**	**97.96**	**97.14**
Funneled	Train	80.99	92.39	85.19	93.97	93.26	86.48	95.12	93.16	95.71	92.42
	Test	77.46	90.58	82.91	93.55	91.13	80.93	91.61	87.19	94.60	92.66

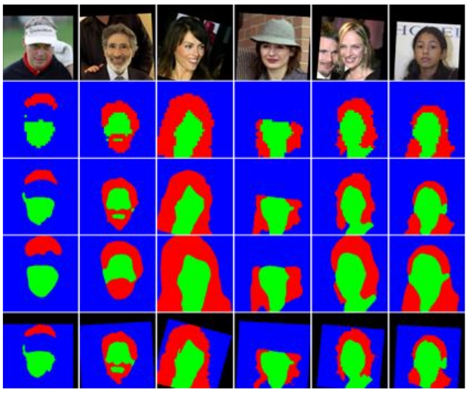

Figure 6. The first row shows normalized images. The second row shows the result of the encoder at 32 × 32 pixels resolution. The third row shows the result of the decoder at 250 × 250 resolution. The fourth row shows the result of the decoder trained with the original funneled images. The last row shows the ground truth data. Note that black pixels are also background pixels.

Figure 6 presents the low resolution output of the encoder and high resolution output of the decoder network as well as the ground truth data. The effect of using geometric normalization is clearly visible when we compare our results in the third row and funneled results in the fourth row. With the normalization of the images, we obtained more precise segmentation.

3.2. Experiments on EG1800 dataset

We used augmentation technique (e.g. flip, random x and y translations) to increase the number of samples. In order to prevent noisy updates, we also utilized mini batch hyper parameter as 40.

Figure 7 shows training and testing results for the encoder and decoder. We obtained 96.37% mIoU and 98.17% pixel based accuracy on the EG1800 dataset. Considering Figure 7, it is clear that the use of the normalization improves the accuracy for both the encoder and decoder. In the tests, we achieved a pixel accuracy of 98.17% using geometric normalization and 97.64% without using it. Similarly, we obtained higher mIoU score when using the normalizations than original data (96.37% vs 95.16%). According to the experiments, our mIoU score 96.37% is the second best score for the EG1800 dataset. Table 4 summarizes state of the art mIoU results of different studies.

Table 4. mIoU results for the EG1800 dataset. Since the EG1800 is a URL based dataset, some of the URLs are not available anymore. Therefore, the total training size is not the same size for different studies. *: Google research team artificially increased the training set size to 465,000 in their study.

Deep Model	Train Set Size	mIoU %
PortraitFCN [26]	1,500	94.20
PortraitFCN+ [26]	1,500	95.91
Google research [28]	1,500	97.01
Google research [28]	465,000*	97.70
Our Model without normalizations+ augmentation	1,439	95.16
Our Model with normalizations	1,439	96.12
Our Model with normalizations+ augmentation	1,439	96.37
PortraitFCN [26] (our tests)	1,439	93.41

We also tested our deep model on the unseen Selfie dataset [38]. The Selfie dataset composed of 46,836 selfie images of Instagram users. However, it does not contain any segmentation data. Therefore, we provide overlaid segmentations (presented in red color) as shown in Figure 8.

3.3. Computation time

Computation time depends on many factors such as number of layers in encoder/decoder, optimizer algorithm, learning rate, momentum, batch size, data augmentation and input/output resolution. It is a natural result that more layers in deep learning bring additional complexity in computations. Our 15-layer network utilizes residual layers to decrease the computation time. Data augmentation is also computationally

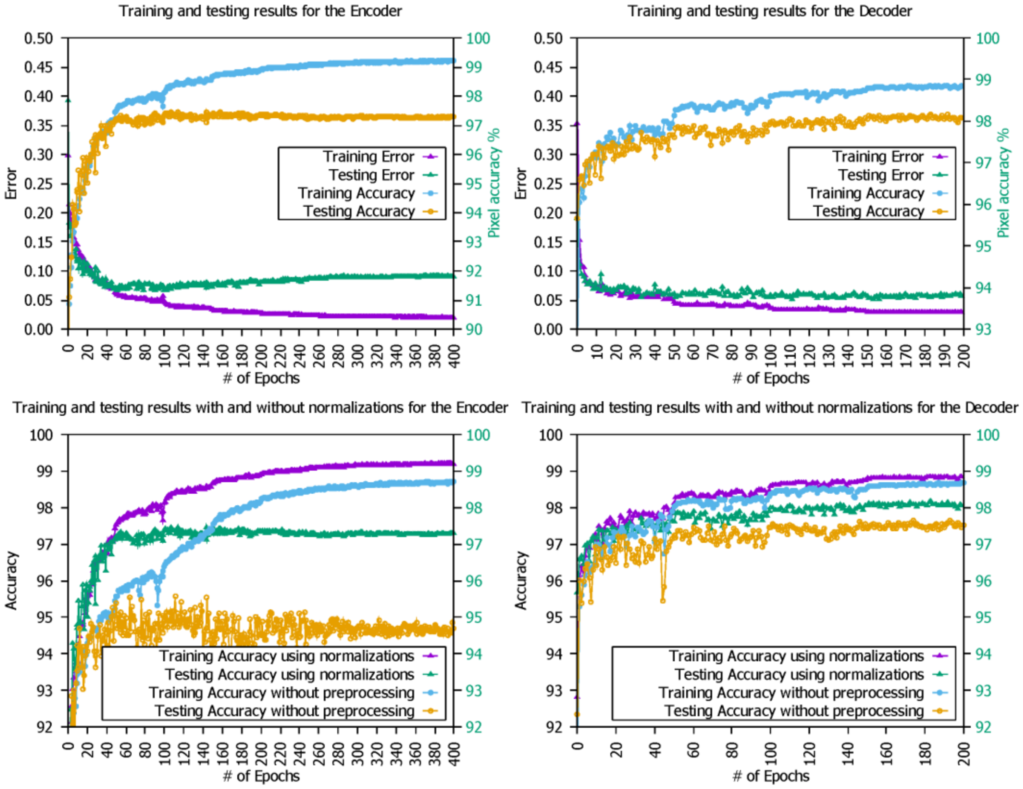

Figure 7. Training error, testing error and the accuracy for the encoder and decoder on the EG1800 Dataset.

complex process but provides more generalized models. For training, our network uses random data augmentation methods, which may affect the overall total computation time in each run. We used the augmentation techniques for training only. In this study, we did not consider test time augmentation (TTA) but planned it as future work.

According to our experiments, at 152×200 resolution, time to test one sample on Nvidia GTX 1060 GPU is 23ms (43 FPS) which is an acceptable rate for a real life scenario. At 600×800 initial resolution, it takes 163ms (6 FPS).

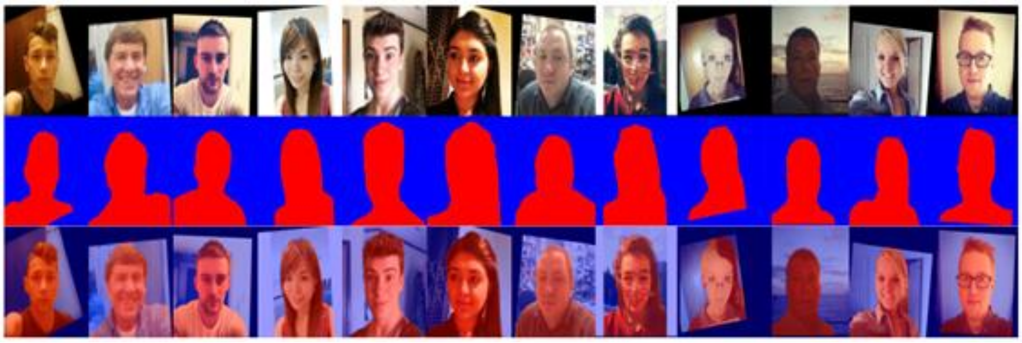

Figure 8. Segmentation examples from Selfie Dataset. First row: Normalized input images. Second row: Output of the decoder. Third row: Segmentation result overlaid on the input image.

4. Discussion and Conclusion

In this study, we proposed a modified Deep Residual Network inspired by the ERFNet architecture for the portrait segmentation problem. One of the most difficult tasks in portrait segmentation problem is the hair segmentation due to the challenging variations in color and shape. In order to target this problem, we employed Deep CNN residual learning units. We further support our model by augmenting the training data using flip and translations. According to the experiments on LFW and EG1800 datasets, we showed the IPD based geometric normalization boost the segmentation accuracy for all segmentation classes. It also provides higher mIoU values in all tests. For the skin, hair and background classes in LFW dataset we obtained segmentation accuracy of 95.79%, 93.98% and 97.96% respectively. Similarly, the use of geometric normalization increases both the pixel based accuracy and mIoU score on the EG1800 dataset where we obtained 96.37% mIoU score.

We showed that use of the normalized images with the proposed architecture provides state-of-the-art segmentation results. Since we use portrait images, we assumed that there is only one face in the image. As future work, we are planning to study on non-portrait and selfie images typically having multiple people using instance segmentation technique. We also plan to perform snapshot ensembling and test time augmentation techniques to improve the overall mIoU performance of proposed method.

Acknowledgment This work was supported by the Scientific Research Projects Coordination Unit of Akdeniz University Project Number: TTU-2018-3295.

References

[1] Goodfellow, I., Bengio, Y., Courville, A. 2016. Deep Learning, MIT Press

[2] He, K., Sun, J. 2014. Convolutional Neural Networks at Constrained Time Cost, CoRR, Vol. abs/1412.1

[3] He, K., Zhang, X., Ren, S., Sun, J. 2015. Deep Residual Learning for Image Recognition, CoRR, Vol. abs/1512.0

[4] Zaitoun, N. M., Aqel, M. J. 2015. Survey on Image Segmentation Techniques, Procedia Computer Science, Vol. 65, p. 797–806. DOI: https://doi.org/10.1016/j.procs.2015.09.027

[5] Zhang, H., Fritts, J. E., Goldman, S. A. 2008. Image Segmentation Evaluation: A Survey of Unsupervised Methods, Computer Vision and Image Understanding, Vol. 110, No. 2, p. 260–280. DOI: https://doi.org/10.1016/j.cviu.2007.08.003

[6] Otsu, N. 1979. A Threshold Selection Method from Gray-Level Histograms, IEEE Transactions on Cybernetics, Vol. 9, No. 1, p. 62–66. DOI: 10.1109/TSMC.1979.4310076

[7] Liu, H., Yan, J., Li, Z., Zhang, H. 2007. Portrait Beautification: A Fast and Robust Approach, Image and Vision Computing, Vol. 25, No. 9, p. 1404–1413. DOI: https://doi.org/10.1016/j.imavis.2006.12.010

[8] Lafferty, J. D., McCallum, A., Pereira, F. C. N. 2001. Conditional Random Fields: Probabilistic Models for Segmenting and Labeling Sequence Data, Proceedings of the Eighteenth International Conference on Machine Learning, Morgan Kaufmann Publishers Inc., San Francisco, CA, USA, p. 282–289

[9] Toyoda, T., Hasegawa, O. 2008. Random Field Model for Integration of Local Information and Global Information, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 30, No. 8, p. 1483–1489. DOI: 10.1109/TPAMI.2008.105

[10] Shotton, J., Winn, J., Rother, C., Criminisi, A. 2009. TextonBoost for Image Understanding: Multi-Class Object Recognition and Segmentation by Jointly Modeling Texture, Layout, and Context, International Journal of Computer Vision

[11] Boix, X., Gonfaus, J. M., Weijer, J., Bagdanov, A. D., Serrat, J., Gonzàlez, J. 2012. Harmony Potentials, International Journal of Computer Vision, Vol. 96, No. 1, p. 83–102. DOI: 10.1007/s11263-011-0449-8

[12] Lin, G., Shen, C., Reid, I. D., van den Hengel, A. 2015. Efficient Piecewise Training of Deep Structured Models for Semantic Segmentation, CoRR, Vol. abs/1504.0

[13] Ladický, L., Russell, C., Kohli, P., Torr, P. H. S. 2009. Associative Hierarchical CRFs for Object Class Image Segmentation, 2009 IEEE 12th International Conference on Computer Vision, p. 739–746. DOI: 10.1109/ICCV.2009.5459248

[14] Boykov, Y. Y., Jolly, M. P. 2001. Interactive Graph Cuts for Optimal Boundary Amp; Region Segmentation of Objects in N-D Images, Proceedings Eighth IEEE International Conference on Computer Vision. ICCV 2001 (Vol. 1), p. 105–112 vol.1. DOI: 10.1109/ICCV.2001.937505

[15] Luu, K., Le, T. H. N., Seshadri, K., Savvides, M. 2012. Facecut - a Robust Approach for Facial Feature Segmentation, 2012 19th IEEE International Conference on Image Processing, p. 1841–1844. DOI: 10.1109/ICIP.2012.6467241

[16] Luu, K., Zhu, C., Bhagavatula, C., Le, T. H. N., Savvides, M. 2016. A Deep Learning Approach to Joint Face Detection and Segmentation, M. Kawulok; M. E. Celebi; B. Smolka (Eds.), Advances in Face Detection and Facial Image Analysis, Springer International Publishing, Cham, p. 1–12. DOI: 10.1007/978-3-319-25958-1_1

[17] Lecun, Y., Bottou, L., Bengio, Y., Haffner, P. 1998. Gradient-Based Learning Applied to Document Recognition, Proceedings of the IEEE, Vol. 86, No. 11, p. 2278–2324. DOI: 10.1109/5.726791

[18] Krizhevsky, A., Sutskever, I., Hinton, G. E. 2012. ImageNet Classification with Deep Convolutional Neural Networks, Proceedings of Neural Information Processing Systems (NIPS), p. 1106–1114

[19] Simonyan, K., Zisserman, A. 2014. Very Deep Convolutional Networks for Large-Scale Image Recognition, CoRR, Vol. abs/1409.1

[20] Szegedy, C., Liu, W., Jia, Y., Sermanet, P., Reed, S., Anguelov, D., Erhan, D., Vanhoucke, V., Rabinovich, A. 2015. Going Deeper with Convolutions, 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), p. 1–9. DOI: 10.1109/CVPR.2015.7298594

[21] Xie, S., Girshick, R. B., Dollár, P., Tu, Z., He, K. 2016. Aggregated Residual Transformations for Deep Neural Networks, CoRR, Vol. abs/1611.0

[22] Badrinarayanan, V., Kendall, A., Cipolla, R. 2017. SegNet: A Deep Convolutional Encoder-Decoder Architecture for Image Segmentation, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 39, No. 12, p. 2481–2495. DOI: 10.1109/TPAMI.2016.2644615

[23] Romera, E., Álvarez, J. M., Bergasa, L. M., Arroyo, R. 2018. ERFNet: Efficient Residual Factorized ConvNet for Real-Time Semantic Segmentation, IEEE T INTELL TRANSP, Vol. 19, No. 1, p. 263–272. DOI: 10.1109/TITS.2017.2750080

[24] Yu, F., Koltun, V., Funkhouser, T. A. 2017. Dilated Residual Networks, CoRR, Vol. abs/1705.0

[25] Everingham, M., Van Gool, L., Williams, C. K. I., Winn, J., Zisserman, A. 2010. The Pascal Visual Object Classes (VOC) Challenge, International Journal of Computer Vision, Vol. 88, No. 2, p. 303–338. DOI: 10.1007/s11263-009-0275-4

[26] Shen, X., Hertzmann, A., Jia, J., Paris, S., Price, B., Shechtman, E., Sachs, I. 2016. Automatic Portrait Segmentation for Image Stylization, Computer Graphics Forum, Vol. 35, No. 2, p. 93–102. DOI: 10.1111/cgf.12814

[27] Long, J., Shelhamer, E., Darrell, T. 2015. Fully Convolutional Networks for Semantic Segmentation, 2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR), p. 3431–3440. DOI: 10.1109/CVPR.2015.7298965

[28] Wadhwa, N., Garg, R., Jacobs, D. E., Feldman, B. E., Kanazawa, N., Carroll, R., Movshovitz-Attias, Y., Barron, J. T., Pritch, Y., Levoy, M. 2018. Synthetic Depth-of-Field with a Single-Camera Mobile Phone, ACM Transactions on Graphics, Vol. 37, No. 4, p. 64:1--64:13. DOI: 10.1145/3197517.3201329

[29] Mostajabi, M., Yadollahpour, P., Shakhnarovich, G. 2014. Feedforward Semantic Segmentation with Zoom-out Features, CoRR, Vol. abs/1412.0

[30] Garcia-Garcia, A., Orts-Escolano, S., Oprea, S., Villena-Martinez, V., Rodr\'\iguez, J. G. 2017. A Review on Deep Learning Techniques Applied to Semantic Segmentation, CoRR, Vol. abs/1704.0

[31] Krähenbühl, P., Koltun, V. 2011. Efficient Inference in Fully Connected CRFs with Gaussian Edge Potentials, Proceedings of the 24th International Conference on Neural Information Processing Systems, Curran Associates Inc., USA, p. 109–117

[32] Kae, A., Sohn, K., Lee, H., Learned-Miller, E. 2013. Augmenting CRFs with Boltzmann Machine Shape Priors for Image Labeling, 2013 IEEE Conference on Computer Vision and Pattern Recognition, p. 2019–2026. DOI: 10.1109/CVPR.2013.263

[33] Viola, P., Jones, M. 2001. Rapid Object Detection Using a Boosted Cascade of Simple Features, Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001 (Vol. 1), p. I–I. DOI: 10.1109/CVPR.2001.990517

[34] Rowley, H. A., Baluja, S., Kanade, T. 1998. Neural Network-Based Face Detection, IEEE Transactions on Pattern Analysis and Machine Intelligence, Vol. 20, No. 1, p. 23–38. DOI: 10.1109/34.655647

[35] Milborrow, S., Nicolls, F. 2008. Locating Facial Features with an Extended Active Shape Model, D. Forsyth; P. Torr; A. Zisserman (Eds.), Computer Vision -- ECCV 2008, Springer Berlin Heidelberg, Berlin, Heidelberg, p. 504–513

[36] Collobert, R., Kavukcuoglu, K., Farabet, C. 2011. Torch7: A Matlab-like Environment for Machine Learning, BigLearn, NIPS Workshop

[37] Huang, G. B., Jain, V., Learned-Miller, E. G. 2007. Unsupervised Joint Alignment of Complex Images, 2007 IEEE 11th International Conference on Computer Vision, p. 1–8

[38] Kalayeh, M. M., Seifu, M., LaLanne, W., Shah, M. 2015. How to Take a Good Selfie?, Proceedings of the 23rd ACM International Conference on Multimedia, ACM, New York, NY, USA, p. 923–926. DOI: 10.1145/2733373.2806365