Background Image Matting

Jade Li, Yirui Miao, Daniel Hu, Zhi Cao, Andy Jin

1. Introduction

In recent years, there has been a growing demand for background replacement features. This demand is driven by factors such as remote work and study, where users may want to conceal their home environments during video calls, or by a desire to use a personal photo as a virtual background. In either case, an easy-to-use background replacement feature is essential. The key aspect of background replacement is the effective separation of the foreground (portrait) from the background.

The problem statement is how to implement a background replacement feature that works seamlessly and accurately in various environments. Our research focuses on training a model capable of distinguishing between the human being (foreground) and their surroundings (background). We take inspiration from previous solutions, aiming to model that process complex environments and adapt a different background. This segmentation not only facilitates background alteration or blurring, but also ensures a professional appearance during video calls. [11]

This research addresses the dual needs of privacy and professionalism, making it an essential tool for digital communication. The background replacement functionality allows users to conceal their surroundings, preventing sensitive or personal information from being visible. This helps privacy, protecting user data. Additionally, the feature provides an accurate and efficient segmentation of the user's figure, allowing for rapid and seamless background replacement, even when the user's physical environment is not ideal. For any professional, this feature is key. Moreover, the background replacement feature enables users to conduct meetings from anywhere, without concern for their surroundings. The convenience of this feature is enhanced by its ability to work effectively with a variety of backgrounds, including complex ones such as natural scenes, home settings, or restaurants. This versatility makes it easier for users to participate in virtual meetings without disrupting their daily routines.

2. Backgrounds

For our data, we relied on public datasets [1, 3, 4] that contain segmented individuals and background. Then, we followed papers discussing various model architectures that vary between relying on pre-trained and non pre-trained architectures. [5–8, 10–12]. Finally to transport the individual to another background we used either a naive copy-paste method or a more complex poisson image blending technique [9]. We focus on the specific technologies from the discussed papers below.

Olaf's work [10] presents the U-Net architecture for biomedical image segmentation, highlighting its use of data augmentation for efficient use of annotated samples. The U-Net's architecture, incorporating a contracting path for context capture and an expanding path for precise localization, has proven effective in segmentation challenges for neuronal structures and cell tracking. The network's excessive data augmentation also enables it to learn invariance to deformations, addressing the needs of biomedical segmentation tasks with limited training data.

The paper "Background Matting: The World is Your Green Screen" [11] presents a method using a deep network with an adversarial loss to recover the alpha matte and foreground color for compositing onto a novel background. This approach involves a self-supervised scheme that bridges the domain gap, utilizing an adversarial network with a discriminator to improve matting quality.

Ning Xu' s research [12] result presents a deep learning algorithm for image matting, utilizing a two-part model: a convolutional encoder-decoder network and a smaller network for refining alpha matte predictions. The paper introduces a large-scale dataset for training and testing, achieving superior results on benchmarks like alphamatting.com, and demonstrates its effectiveness on real-world images with improved matte predictions and sharper edges.

Liang-Chieh Chen's paper [6] introduces DeepLab, a system for semantic image segmentation, achieving state-of-the-art results. The system incorporates Atrous Spatial Pyramid Pooling (ASPP), which segments objects at multiple scales through parallel branches with different atrous rates, capturing varying object sizes and image context. ASPP, along with multiscale inputs and data augmentation, enhances DeepLab's performance, especially with a ResNet-101 backbone.

"Poisson Image Editing" [9] introduces tools for seamless image editing, utilizing a generic framework based on the Poisson equation to import and modify image regions. This approach enables editing of arbitrary patches, guided

interpolation techniques for seamless cloning, and the use of discrete Poisson solvers for variety problems in image editing.

3. Methodologies

In the pursuit of refining the segmentation of human figures from complex backgrounds, we have tailored a U-Net architecture [10] renowned for its proficiency in image segmentation tasks. Our study encapsulates the utilization of dual U-Net configurations—the elementary single-layer model delineated in Section 3.1, and the multifaceted standard U-Net model expounded in Section 3.2. These models were harnessed to forecast segmented imagery of human portraits.

To bolster the models' precision and versatility, we initially pre-trained them on a diverse dataset and subsequently fine-tuned them on the specialized Portrait Processing Module (PPM) dataset [4]. This methodological approach significantly improved the accuracy and adaptability of our models, thereby enhancing the background replacement feature in video conferencing tools.

The inherent structure of the U-Net, with its contracting paths for contextual capture and expanding paths for image reconstruction, along with skip connections for detail preservation, provided a solid foundation for our experiments. These features are paramount in generating precise segmentation masks—a key attribute that makes the U-Net an exemplary candidate for optimizing background segmentation in video communication applications.

As detailed in Section 3.3, our model's pretraining was performed on distinct datasets before undergoing fine-tuning on the PPM dataset. We experimented with freezing certain layers from the pretrained network to gauge their impact on performance compared to a completely unfrozen model. Additionally, we conducted an exhaustive search for optimal hyperparameters that would yield the highest accuracy in segmentation. In Section 3.4, we benchmarked our improved U-Net model against a contemporary state-of-the-art model to validate our results. Further, in section 3.5, we employed poisson image editing techniques to seamlessly integrate segmented figures into alternative backgrounds. The subsequent sections will delve deeper into these techniques and acknowledge the contributions of foundational research that informed our approach.

3.1. Simplified U-Net Model

Our Simplified U-Net Model represents the most basic iteration of the U-Net architecture, tailored for this project with only one depth level. This model is the earliest version we tested, serving as a baseline to understand the minimal architectural requirements needed for image segmentation tasks. Its simplistic design is intentionally chosen to explore the foundational capabilities of U-Net architectures under constrained settings.

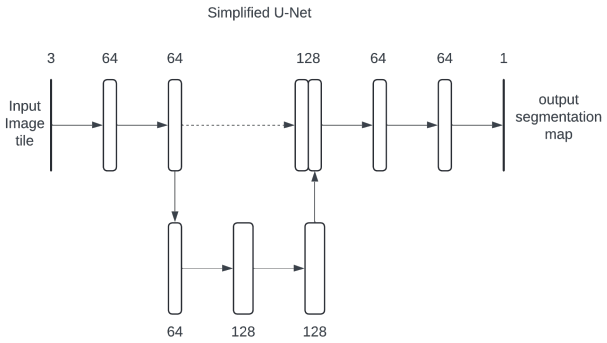

Figure 1. Simplified U-Net Model

3.1.1 Architecture

The model is constructed using TensorFlow's Keras API, featuring a streamlined U-shaped architecture that facilitates both the contraction of the feature space and the subsequent expansion for precise localization. The input layer accepts images of size 256×256 with three color channels.

In the contracting path, the first convolutional block consists of two 3x3 convolutional layers with 64 filters each, using ReLU activation and He normal initializer for weight initialization. This is followed by a dropout of 10% to prevent overfitting and a max pooling layer with a 2x2 window to reduce dimensionality.

Transitioning to the expansive path, an up-sampling layer increases the dimensionality of the feature maps, which is then concatenated with the corresponding feature map from the first convolutional block to help the model localize features more accurately. This is followed by two more convolutional layers with 64 filters each, another dropout of 20%, and a final 1x1 convolutional layer that outputs the segmentation mask using a sigmoid activation function to ensure the output values are between 0 and 1.

3.1.2 Training

The training process was carried out on a notably small dataset of just 100 images from the PPM-100 dataset. This limitation was deliberate, to stress-test the model's learning efficacy under sparse data conditions. We utilized the Adam optimizer with a learning rate of 0.001, and incorporated early stopping to mitigate overfitting, a significant risk given the small data set. Despite these measures, the model's performance was suboptimal, highlighting the challenges of deep learning with limited training examples.

3.1.3 Relevance to Project

The primary value of deploying this rudimentary version of the U-Net lies in its educational potential and its utility as a benchmarking tool. It serves to illustrate the limitations of minimalistic neural network designs in handling complex tasks like image segmentation. The insights gained from the performance of this model help in guiding the subsequent iterations and enhancements necessary for achieving practical and reliable segmentation results. This foundational understanding is crucial for projects where data availability is constrained and computational efficiency is a priority.

3.2. Standard U-Net Model

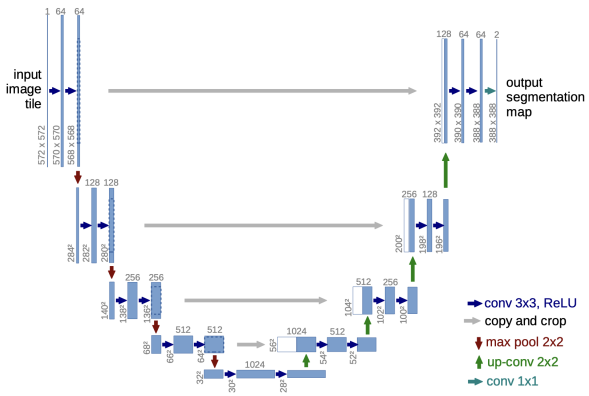

Figure 2. Simplified U-Net Model

3.2.1 Motivation

The unsatisfactory performance of the first model was not surprising, since this model is very shallow. Therefore, it might not be suitable for human portrait segmentation due to its lack of complexity. To address this issue, we decided to build a U-Net model with more layers and resolve the memory issues by other means. This decision is also supported by [12] since this paper provided a relatively more modern architecture, which is also using the encoder-decoder model. In this paper, the basic encoder-decoder model with similar structure to that in [10] shows the ability to perform human portrait segmentation tasks, implying that the idea of using U-Net model is reasonable. In this step, we borrowed the entire encoder-decoder architecture from [12] and reimplemented it using Tensorflow. To resolve the memory issue, we reshaped both the image and matte to 256*256 pixels to reduce memory consumption. The data preprocessing is conducted with OpenCV.

3.2.2 Data Preprocessing

In this part, we took the easy way to reduce the size of the image by using OpenCV to directly resize the image to 256*256 pixels. This caused some images to look a bit disproportional, and could have potentially caused some pixels at the edge of the map to but it saved the pain of manually cropping the image while maintaining the complete human portrait. Our image preprocess overcame the memory consumption problem, but could become a bottleneck in improving the accuracy.

3.2.3 Model Architecture

As shown in the image, the encoder has 4 levels, each consisting of 3 convolutional layers with ReLU that increase the channels of the feature matrix. A 2*2 max pooling is applied by the end of the level, downsizing the height and width of the feature matrix by 1/2. The last feature matrix output before the pooling step is saved for utilization in the decoder. After the four levels of encoding, two more layers of convolutions are applied, then upscaled by applying a 2*2 unpooling, marking the beginning of the decoder. The decoder also has 4 layers. At the beginning of each layer, the current feature matrix is concatenated with the output of the corresponding layer of the encoder as a new matrix. Then two convolutional layers and ReLU are applied to halve the number of channels, and one unpooling layer is applied to double the height and width of the matrix. At the very end, we take the 64-channel output, and apply a 1*1 convolution to turn it into a single channel output matrix. The main differences between our model and the model the image shows are that 1) the original input has 3 channels instead of 1, and our final output has 1 channel instead of 2. 2) our model used padding for each convolution so the height and width of the feature matrix only change when pooling or unpooling is applied.

3.2.4 Training

In this part, we used the same dataset as we did in 3.1.4. As mentioned before, this choice was deliberate, with the main goal being to verify the efficacy of the model before conducting training on large datasets. The training results are shown and discussed in 4.1.

3.3. Pretrained on Labeled Real Data

In the initial phase of pretraining, we utilized the Pascal VOC 2012 dataset [3], which encompasses 7,282 images featuring 19,694 labeled objects across 21 distinct classes. However, the domain discrepancy between this dataset and the specific category of human facial images we sought to segment proved to be significant, resulting in suboptimal

outcomes when applying conventional neural network models. To mitigate this domain gap, we incorporated the Human Matting dataset [1] into our training regimen. This dataset is composed of authentic human face images, each paired with a corresponding mask, which serves to refine our model's segmentation capabilities.

Despite these adjustments, we observed the persistence of artifacts within the segmentation masks, particularly when they were applied to synthesize images against novel backgrounds. To address this, we executed an additional fine-tuning stage on the PPM dataset [4], employing binary cross-entropy loss as our optimization criterion. This fine-tuning significantly enhanced the matting quality, as evidenced by the improved clarity and reduced artifacts in the segmentation masks. The intricate details of this process, as well as the quantitative improvements it yielded, are discussed further in the upcoming sections.

3.4. MobilenetV2/DeepLab

DeepLab is a deep learning system designed for semantic image segmentation, incorporating several key features. [2] Atrous Convolution, also known as dilated convolution, increases the receptive field of convolutional layers without reducing the spatial dimensions of the input, allowing the network to capture finer details and more contextual information. Atrous Spatial Pyramid Pooling (ASPP) applies filters with different sampling rates and fields of view, enabling robust segmentation at multiple scales and helping the network recognize and segment objects of varying sizes. The ASPP module is further augmented with an image-level feature and batch normalization, enhancing its ability to capture global context and stabilizing the training process. Finally, the decoder module refines the segmentation output by recovering object boundaries, enhancing the quality of the final segmentation mask, particularly at object edges. [8] DeepLab performs better in semantic segmentation tasks due to its combination of multi-scale awareness, global context capture, and edge refinement. The Atrous Spatial Pyramid Pooling (ASPP) module plays a key role in this performance, as it applies filters with different sampling rates and effective fields of view, making DeepLab more robust at segmenting objects of varying sizes and capturing image context at multiple scales. Additionally, the incorporation of image-level features and batch normalization enhances global context capture, aiding in distinguishing between different objects and reducing segmentation errors. The decoder module then refines the segmentation output by recovering object boundaries, leading to sharper and more accurate segmentations, especially at the edges. In summary, DeepLab's architecture combines these elements to create a powerful and reliable tool for semantic image segmentation.

3.5. Image Matting

Using poisson image blending techniques [9] we're able to better represent the cut out person from the original image into a different background. This allows for a smoother transition at the edges of the cut out person, creating an overall seamless photo. There presents a major disadvantage, however, in the destination image affecting the color tone of the pasted person (this can be shown in Figure 3). As evaluated from a human perspective, the lady presents teal colored undertone as a consequence of the underwater background.

Figure 3. Poisson blended image example

4. Results

4.1. Quantitative Results

4.1.1 Transfer Learning

In our study, we implemented transfer learning on a U-Net model that had been pretrained on the Human Matting dataset. Our analysis, as outlined in Table 1, examines the impact of freezing different numbers of layers during the fine-tuning process on binary cross entropy loss and accuracy.

Table 1 reveals that freezing all layers except for the fully connected (FC) layer resulted in a binary cross entropy loss of 0.3613 and an accuracy of 0.8214. When only the first two layers were frozen, the model achieved a loss of 0.3202 and an accuracy of 0.8527. Freezing just the first layer provided further improvement, reducing the loss to 0.3013 and increasing the accuracy to 0.8814. The most notable enhancement was observed when no layers were frozen; the model achieved the lowest loss of 0.2710 and the highest accuracy of 0.9123.

These results indicate that while transfer learning enables the model to start with a more informed understanding of the task, allowing some layers to adjust to the new data can significantly improve performance. The unfrozen layers can learn features that are more specific to the target dataset, leading to more accurate predictions. Conversely, freezing too many layers can hinder this fine-tuning process, causing the model to retain features that are less optimal for the specific task, as reflected by the higher loss and lower accuracy.

Thus, our study demonstrates the importance of selectively freezing layers during transfer learning to balance the retention of learned features with the flexibility to adapt to new data, especially for complex tasks such as image matting where domain-specific nuances are crucial.

	Binary Cross Entropy Loss	Accuracy
Freeze All Layers	0.3613	0.8214
Freeze First Two Layers	0.3202	0.8527
Freeze First Layer	0.3013	0.8814
Freeze No Layers	0.2710	0.9123

Table 1. Transfer Learning: Binary Cross Entropy Loss and Accuracy for Each Number of Layers Frozen

4.1.2 Results on Real Data

As demonstrated in Table 2, we present a comparative analysis of various models, highlighting the impact of pretraining on distinct datasets. We test our model on real human face data from PPM dataset. Our evaluation encompasses the Simplified U-Net Model, a standard U-Net Model, and the same U-Net Model pretrained on two separate datasets: Pascal VOC 2012 [3] and Human Matting [1]. Additionally, we include the results of the MobileNetV2 architecture for comparison.

From the binary cross-entropy loss and accuracy metrics, it is evident that pretraining plays a critical role in model performance. The Simplified U-Net Model, without any pretraining, serves as a baseline with a binary cross-entropy loss of 0.4602 and an accuracy of 0.7227. When pretrained on Pascal VOC 2012 [3], the model exhibits a marked improvement, reducing the loss to 0.3213 and increasing accuracy to 0.8814, indicating that even when there is a domain gap, pretraining on a diverse set of images contributes to learning generalized features.

Pretraining on the Human Matting dataset [1], which is more domain-specific, resulted in further performance enhancements. The binary cross-entropy loss decreased to 0.2710, and the accuracy ascended to 0.9123, underscoring the significance of domain relevance in pretraining datasets. Moreover, the MobileNetV2/DeepLab architecture outperformed all U-Net based models with a binary cross-entropy loss of just 0.1183 and the highest recorded accuracy of 0.9565. This suggests that the MobileNetV2/DeepLab architecture, with its lightweight and efficient ASPP structure, may be better suited for tasks that demand both high accuracy and computational efficiency.

Model	Binary Cross Entropy Loss	Accuracy
Simplified U-Net Model	0.5419	0.7227
U-Net Model	0.3602	0.8328
Pretrained on Pascal VOC 2012	0.3213	0.8814
Pretrained on Human Matting	0.2710	0.9123
MobileNetV2	0.1183	0.9565

4.2. Qualitative Results

In Figure 4, from left-to-right we have the baseline, no pretrained, and pretrained models respectively. The baseline model, which we assume has been optimized for this task, displays a clear and accurate segmentation of the human figure with minimal artifacts around the edges. The non-pretrained model, while able to capture the broad out-

line of the figure, introduces notable errors particularly around areas of complex background textures and finer details such as hair, indicating a lack of feature discrimination that is likely due to insufficient training on relevant data.

Our pretrained model, on the other hand, exhibits a significantly improved mask, approaching the accuracy of the baseline. The nuances of the figure, including the silhouette and the majority of the hair, are well-captured. However, it's important to note that some artifacts remain—though they are greatly reduced compared to the non-pretrained model. This result suggests that the model has learned more sophisticated features from the Human Matting dataset, enabling better generalization when applied to new images.

Upon applying the poisson blending technique to the segmented output, the edges and transitions between the figure and the new background become smoother, rendering any remaining segmentation imperfections less noticeable. This post-processing step is particularly useful in harmonizing the segmented figure with a range of backgrounds, a quality desirable in applications such as virtual meetings or augmented reality.

Figure 4. Comparison of different model results with poisson blending techniques

4.3. Discussion

Both qualitative results and quantitative results are generally within expectation. As we can see, a more complex model works better than a very simple model, and pretraining on another similar dataset and applying transfer learning also helps to improve performance. One surprising observation is that when using a complex enough model (e.g. Standard U-Net Model) with the right hyperparameters, only a small amount of iterations can yield surprisingly good results.

During our training process, we observed that Human Matting dataset performs better in our project. This is not unexpected, since our goal was to conduct human portrait segmentation.

By observing the qualitative data, we also learn that a good matting process is also very important for generating high quality images, as it can reduce the visual significance of error generated by non-precise ML model.

Due to the limited time and computational resources, our model is still underfitting. This can be observed in both qualitative and quantitative results. Compared with the models designed in our referred papers, our model is significantly simpler. We made an attempt to train a more complex model, but failed due to the lack of computational resources. Therefore, we believe improving the model's complexity and training on augmented bigger dataset would be beneficial.

5. Conclusion

Our research focuses on training a model capable of accurately distinguishing between the human figure (foreground) and its surroundings (background). We employ a U-Net architecture, refined to suit segmentation tasks, and rely on datasets like Pascal VOC 2012 and Human Matting for training. This training approach, combined with techniques such as Poisson image editing, allows for effective segmentation and integration of the human figure into a new background, leading to smooth, visually acceptable results.

Comparing that to a state-of-the-art pre-trained baseline, we did not manage to exceed existing performance but saw similar results. While the result is somewhat good enough for us to display the human figure, not perfectly, with the new background. It does not perfectly solve our problem statement, but we indeed get some good progress for this goal.

If we had more time, we would explore and implement more complex models and projects like Background Matting V2, leveraging advanced architectures to further enhance segmentation accuracy and performance. Additionally, we would incorporate extensive data augmentation techniques to create a more diverse training set, reducing overfitting and improving generalization. This, coupled with additional training, would refine the models further, ensuring precise segmentation and seamless background replacement functionality.

References

[1] Achieve dataset. https://www.kaggle.com/datasets/ntquyen11/matting/code. 1, 4, 5

[2] Deeplabv3. https://github.com/McDo/Modanet-DeeplabV3-MobilenetV2-Tensorflow. 4

[3] Pascalvoc. http://host.robots.ox.ac.uk/pascal/VOC/voc2012/segexamples/index.html. 1, 3, 5

[4] Ppm-100. https://paperswithcode.com/dataset/phm-100. 1, 2, 4

[5] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. Semantic image segmentation with deep convolutional nets and fully connected crfs, 2016. 1

[6] Liang-Chieh Chen, George Papandreou, Iasonas Kokkinos, Kevin Murphy, and Alan L. Yuille. Deeplab: Semantic image segmentation with deep convolutional nets, atrous convolution, and fully connected crfs, 2017. 1

[7] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation, 2017. 1

[8] Liang-Chieh Chen, Yukun Zhu, George Papandreou, Florian Schroff, and Hartwig Adam. Encoder-decoder with atrous separable convolution for semantic image segmentation, 2018. 1, 4

[9] Patrick Perez, Michel Gangnet, and Andrew Blake. Poisson image editing, 2003. Microsoft Research UK. 1, 4

[10] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation, 2015. 1, 2, 3

[11] Soumyadip Sengupta, Vivek Jayaram, Brian Curless, Steve Seitz, and Ira Kemelmacher-Shlizerman. Background matting: The world is your green screen, 2020. 1

[12] Ning Xu, Brian Price, Scott Cohen, and Thomas Huang. Deep image matting, 2017. 1, 3

A. Appendix

A.1. Source Code

The source code for the project can be found at the following GitHub repository:

https://github.com/erayyym/supermatting