A MODULAR APPROACH TO IMAGE MATTING

by

Mir Afgan H. Talpur

A thesis submitted to the
School of Graduate and Postdoctoral Studies
in partial fulfillment of the requirements for the degree of

Master of Science in **Computer Science**

Faculty of Science
Ontario Tech University
Oshawa, Ontario, Canada
November 2022

Thesis Examination Information

Submitted by: **Mir Afgan Talpur**

Master of Science in **Computer Science**

Thesis Title:

A Modular Approach to Image Matting

An oral defense of this thesis took place on October 6, 2022 in front of the following examining committee:

Examining Committee:

Chair of Examining Committee	**Dr. Gabby Resch**
Research Supervisor	**Dr. Faisal Qureshi**
Examining Committee Member	**Dr. Bill Kapralos**
Thesis Examiner	**Dr. Alvaro Quevedo**, Ontario Tech

The above committee determined that the thesis is acceptable in form and content and that a satisfactory knowledge of the field covered by the thesis was demonstrated by the candidate during an oral examination. A signed copy of the Certificate of Approval is available from the School of Graduate and Postdoctoral Studies.

Abstract

Image matting is the art of creating an accurate alpha matte for the purpose of foreground separation in an image or video. Although there have been many methods which only require an input image, the best-performing image matting models continue to rely on additional inputs — mainly the trimap — for more accurate alpha matte estimations. We propose a modular image matting architecture which leverages advancements in semantic segmentation and our trimap generation network to allow for a trimap-free approach to some of the most popular trimap-based image matting methods. Our design delivers promising results, allowing users to take advantage of powerful trimap-based methods, without having to worry about additional inputs, all while granting them the freedom to swap different networks in and out for the different stages of the modular architecture.

Keywords: Image Matting; Foreground Estimation; Trimap Generation; Neural Networks; Computational Photography

Author's Declaration

I hereby declare that this thesis consists of original work of which I have authored. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners. I authorize the Ontario Tech University to lend this thesis to other institutions or individuals for the purpose of scholarly research. I further authorize the Ontario Tech University to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research. I understand that my thesis will be made electronically available to the public.

Mir Afgan H. Talpur

Statement of Contributions

I hereby certify that I am the sole author of this thesis and that no part of this thesis has been published. I have used standard referencing practices to acknowledge ideas, research techniques, or other materials that belong to others. Furthermore, I hereby certify that I am the sole source of the creative works and/or inventive knowledge described in this thesis.

Acknowledgements

Firstly, I would like to thank my supervisor, Dr. Faisal Qureshi, for inviting me to join his lab for both my undergraduate and graduate studies. His mentorship and guidance helped me overcome some of the most difficult hurdles in my academic career. He always encouraged me to chase after the problems I was most interested in, based on my personal interests and goals.

Secondly, I would also like to thank my former and current colleagues at the VCLab. My friends at the lab were always tremendously helpful with both career and life advice. These past 4 years were filled with memories of excitement, innovation, determination, lots of coffee, and even more laughter. These are memories that I will always cherish.

And, of course, I would like to thank my friends and family for their never ending support throughout the past few years. Everyone — from my fam in Whitby to my bhais back in London to my kuyas in Oshawa — have had a monumental impact on me, both as a student and as a person. Their friendship means the world to me.

Thanks to my siblings: Aahan, Rida, and Rawish. It has been a treat watching them grow into the people they are today. Thank you to my Nanu and Dadu Amma. Their support and prayers throughout my academic career have gotten me through the toughest of times. Thank you for continuing to watch over our family.

Most of all, I would like to thank my parents, Mir Liaqat Talpur and Mahjabeen Talpur. They sacrificed everything back home for a chance of a better life here for my siblings and me. Every day, I am grateful for their decision. They did not hesitate to support my decision to return to school. And even with all the ups and downs over the past few years, they always made sure that our family got back up stronger than ever.

"حرکت میں برکت"

- Urdu Proverb

Contents

List of Tables

List of Figures

Glossary

DIM

Deep Image Matting (trimap-based work) [1].

FBA

F, B, α Matting (trimap-based work) [2].

GFM

Glance and Focus Matting (trimap-free work) [3].

HAtt

Hierarchical Attention Matting (trimap-free work) [4].

Human Matting

Image matting with a human subject.

Image Matting

The technique of estimating or extracting the foreground and background of an image.

IQR

Interquartile Range.

LF

A Late Fusion CNN for Digital Matting (trimap-free work) [5].

MAD

Mean of Absolute Differences (metric).

MSE

Mean-Squared Errors (metric).

Portrait Matting

Close up human matting, usually performed on a portrait (shoulders and up).

Real-World Image

A natural, everyday image, usually with a complex background, taken in an unconstrained scene.

SAD

Sum of Absolute Differences (metric).

Semantic Segmentation

A pixel-level approach to image classification, which seeks to label each pixel in an image as belonging to a certain class.

SHM

Semantic Human Matting (trimap-free work) [6].

Trimap

A partition of an image, which separates pixels into three regions: foreground (white), background (black), and unknown (grey).

Chapter 1

Introduction

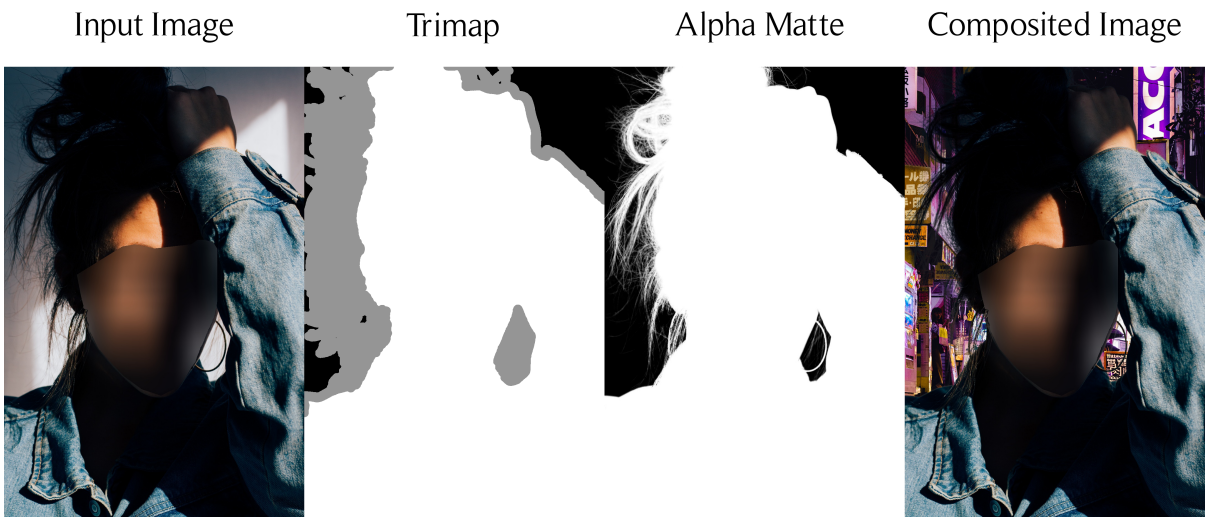

Figure 1.1: An example of image matting.

In the world of photography, one of the most prevalent mantras is the idea of transforming a photo from "ordinary to extraordinary". One way photographers achieve this goal is by editing and modifying their photos. Among these modifications, foreground extraction remains one of the most powerful photography editing tools. Although this tool is widely used by professional photographers, editors, and visual effects artists, the applications of foreground extraction are no longer solely aimed at professionals. This versatile idea can be leveraged for many purposes, including art, entertainment, and accentua-

tion of a foreground subject. Among the seemingly endless applications of foreground separation are green screen in movies, portrait mode in phone cameras, and background replacement for social media posts. The versatility of — and urgency for advancements in — foreground extraction techniques have only been further emphasized in the past few years with the growing need for reliable background replacement, blurring, and removal in video conferencing applications.

This is where one of the most exciting areas of computational photography comes in: image matting. Image matting is the art of accurately separating the foreground and background in an image or video. This is achieved by meticulously constructing an alpha matte for a precise, defined foreground in an image or video of interest. These alpha mattes act as masks which allow us to preserve fine details such as hair and fabric in our foregrounds. Thankfully, the aforementioned applications of image matting have been made more available in recent years with the advent of deep learning in computer vision. Nevertheless, even with these advances, image matting remains a very difficult task.

Inherently, image matting is an underconstrained problem. This can be seen in the matting equation:

$$I_i = \alpha_i F_i + (1 - \alpha_i) B_i, \qquad \alpha_i \in [0, 1]. \tag{1.1}$$

Here, I_i represents a known RGB value at pixel i, F_i is the foreground value, B_i is the background value, and α_i is the foreground transparency value [1]. Consequently, we can see that there are 7 unknown and 3 known values per pixel in a 3-channel, RGB image.

Traditionally, image matting was performed using sampling [7, 8] and propagation-based [9, 10, 11] methods. These works usually tried to solve the matting equation and were heavily dependent on colour and spatial information to do so. Unfortunately, this was also their biggest weakness. These methods would often fail with complex backgrounds or similar colours in the foreground and background. These methods frequently also relied on an additional user-defined input called a trimap. A trimap is a partitioned

image, consisting of three regions: an explicit foreground region (white), an explicit background region (black), and an unknown region (grey), where the alpha matte is calculated. An example of a trimap — and its accompanying image — can be seen in Figure 1.1. Although much less labour-intensive than creating an alpha matte, sketching out a trimap from scratch remains a laborious task for most users. Until the requirement for supplementary inputs is eliminated, image matting will remain inaccessible to many end users and cannot be adopted by the general masses.

Another major hurdle was the lack of large, diverse datasets in image matting. Annotating ground truth alpha mattes is a painstaking and lengthy task and is often performed by experts in order to achieve high-quality mattes. The idea of automating this process is one of the main reasons why image matting is such an exciting field. Traditional methods generally relied on a few dozen images taken in front of simple, static backgrounds and often did not have human subjects in the foreground [12].

Fortunately, the past decade has ushered in an era of advancements of deep learning in computer vision. Along with the influx of deep learning works in image matting, came various large, high-quality datasets [1, 13, 14, 15]. These ranged from a few hundred foreground images composited onto a variety of backgrounds to thousands of natural, real-world images coupled with high-quality alpha matte annotations. Many of these large datasets focused solely on humans in a variety of poses, allowing for human matting. Specifically, many of these were aimed at portrait matting: image matting on a close-up shot of a human subject, with a reduced field of view. This type of matting naturally emphasises the fine details in hair and clothing. It also is the type of matting that is best-suited for applications such as social media posts and background effects in video conferencing programs.

As mentioned earlier, deep learning did not only help advance image matting, it also did wonders for other areas of computer vision. These developments, particularly in semantic segmentation — a pixel-wise form of image classification, which labels each

pixel as belonging to a particular class — have allowed for the emergence of a new approach to image matting: trimap-free matting. As its name would suggest, this image matting technique does not require a trimap as an additional input. Seeing how trimap-free methods do not refer to trimaps for prior knowledge, they often rely on some form of semantic segmentation as the first step to matting. This first step provides a rough representation of the foreground subject, which is transformed, adapted, and prepared in the appropriate manner and passed through to the matting step.

Generally, the semantic segmentation stage is automated, which may lead to errors in estimating the foreground subject. Additionally, because defining a foreground without any priors is a challenging task, trimap-free methods are often specialized and trained on certain subjects, such as humans, pets, or cars. For these reasons, although trimap-free methods can be much more convenient, they are typically not as accurate as trimap-based methods, as shown in Figure 1.2. This is due to the fact that trimap-based approaches involve human input in the matting process. However, creating a trimap is an impractical for many users. End users must decide whether they can sacrifice accuracy for convenience when choosing their image matting approach.

Consequently, we formulate our research goal: to develop a method that bridges the gap between the performance and interactivity of trimap-based networks and the convenience of trimap-free networks. With this in mind, we design and explore the viability of a modular image matting architecture. This design provides an alternate, convenient approach to portrait matting that avoids the need for any additional inputs. We do so with the intuition that decomposing image matting from a complicated task to a sequence of simpler, straightforward tasks is how a human would approach this problem, coupled with the insight that deep learning networks excel at performing such tasks. Our modular design leverages advances in semantic segmentation, a trimap generation network, and a pretrained trimap-based image matting network, as a potential alternative to current trimap-free image matting methods.

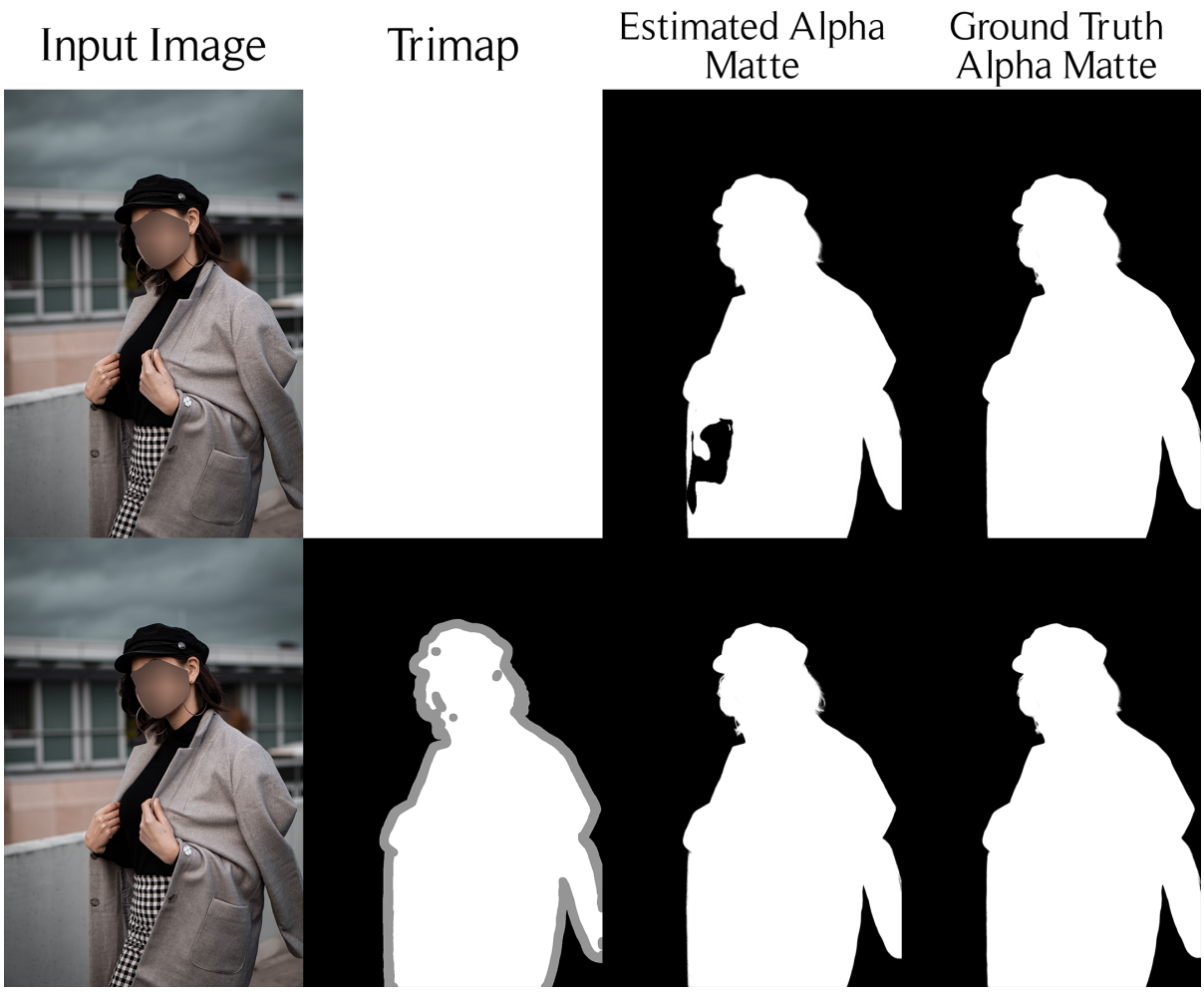

Figure 1.2: An example comparing the inputs and outputs of both trimap-free and trimap based matting. Top row: result from P3M-Net, a trimap-free method [15]. Bottom row: result from FBA Matting, a trimap-based method [2].

The two main contributions of our work are:

1. A modular network architecture for image matting. One that allows for a trimap-free approach to some of the most popular trimap-based image matting works. A versatile design that grants the user the freedom to replace all three stages, given they share the correct inputs and outputs. All while achieving competitive results when compared to current state-of-the-art trimap-free methods and a minimal decrease in accuracy when compared to its trimap-based counterparts.

2. A trimap generation network. When used in conjunction with a semantic segmen-

tation network, this trimap generation network creates reliable trimaps, while only requiring a single input image. These trimaps can then be used as additional inputs to other image matting networks.

1.0.1 Thesis Breakdown

This thesis explores our research and contributions to the field of image matting. We start with a literature review in Chapter 2. Following our exploration of relevant works, we give a detailed overview of our modular image matting design in Chapter 3. We dive into the details of our experiments in Chapter 4, including a description of the dataset in Chapter 4.1, sharing our results in Chapter 4.3, and analyzing our findings in Chapter 4.4. Finally, we summarize and discuss future directions, including potential applications of our research, in Chapter 5.

Chapter 2

Related Works

In this chapter, we discuss the background and history of image matting works, from traditional approaches to deep learning-based methods. These sections are further broken down into more specific types of approaches to give a clearer view of the ever-changing landscape of research in image matting. Within each subsection, we describe the various techniques researchers have proposed, along with the datasets they introduced.

2.1 Traditional Methods

Prior to the growth of deep learning within computer vision works, traditional methods were used to estimate alpha mattes. These works generally fit into two categories: sampling-based [7, 8] and propagation-based [9, 10, 11] image matting.

2.1.1 Sampling-Based Works

One of the earliest sampling-based techniques was the patent filed by Berman *et al.* in 1998 [7]. Their approach estimated the alpha matte within the unknown region of a user-generated trimap, sampling the surrounding known foreground and background regions. The unknown pixels were then calculated using a combination of the two known regions.

This method was based on their intuition that the unknown region was a mixture of the subject and background colour, determined by the distance of the unknown pixel's colour to the background colour. This approach struggled with fine structures as it relied on low-level features which were easily misconstrued due to factors such as background spill and complex backgrounds.

Chuang *et al.* improve on the aforementioned technique by proposing a Bayesian approach to matting [8]. With this approach, they analyzed the unknown pixels by building local distributions, rather than forming a global distribution using sampling. Using a trimap, this approach built foreground and background probability distributions by sliding a window from the foreground and background regions into the unknown region. Combining known foreground, background, and alpha values with calculated ones, it created oriented Gaussian distributions. They modeled the parameters using a Bayesian framework and estimated optimal opacity, foreground, and background using a maximum-likelihood criterion.

2.1.2 Propagation-Based Works

Sun *et al.* introduced Poisson Matting, in which they used a two-step approach to matting [9]. First, they approximated a matte gradient field using the input image. Then, they solved Poisson equations to generate an alpha matte. This method solved these equations using the unknown regions of a trimap, specifically information related to boundaries. They did so by defining boundary conditions in their equations based on the given trimap. While this approach improved on previous methods, it required user interaction in certain cases and failed with intertwined foreground subjects, similar foregrounds and backgrounds, and complex backgrounds.

Levin *et al.* debuted a popular image matting technique called Closed-Form Matting [10]. Here, they assume that there is a small area around each unknown pixel, in which the foreground and background were constant. They are able to derive a cost function,

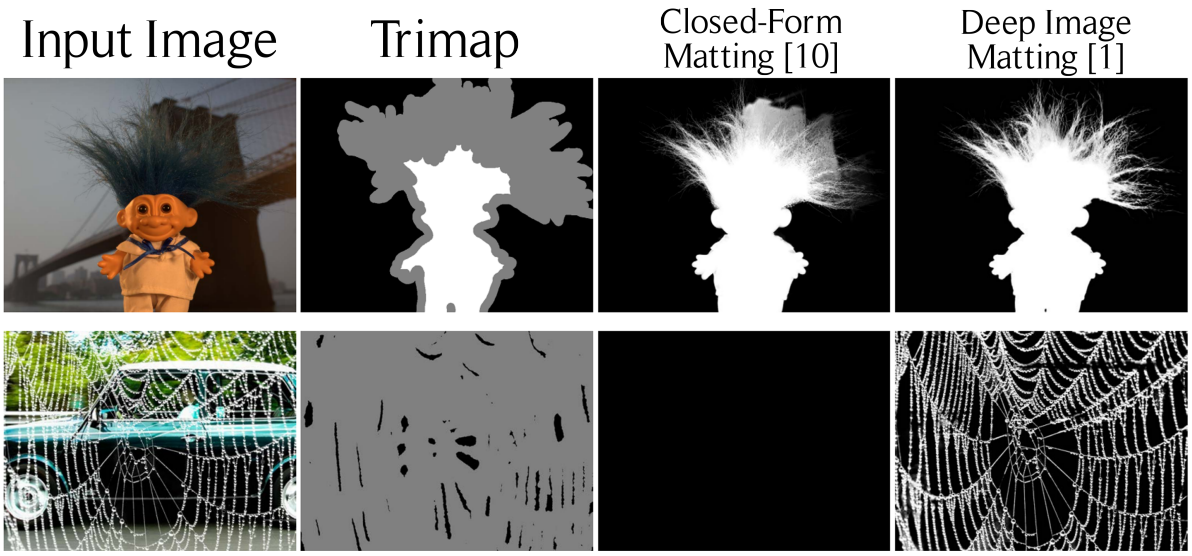

Figure 2.1: Comparing the estimated alpha mattes from Closed-Form Matting [10], a traditional, propagation-based work, and Deep Image Matting [1], a deep learning-based approach. Figure originally from [1].

and using these assumptions, they analytically eliminate the foreground and background colours. This results in a quadratic cost function in the alpha values. This cost function is solved through a system of linear equations.

2.2 Deep Learning-Based Methods

While traditional methods showed some potential, their dependence on low-level features, such as colour and limited spatial context, led them to fail in complex scenes. An example of this can be seen in Figure 2.1, comparing the estimated alpha mattes from [10] and [1]. Until the emergence of deep learning in image matting, this would remain the case with numerous matting methods [16, 17, 18].

2.2.1 Trimap-Based Works

Although there were a few attempts to leverage deep learning for image matting, it was not widely adopted prior to the seminal Deep Image Matting (DIM) paper by Xu *et al.*

in 2017 [1]. The authors leveraged an encoder-decoder matting network and subsequent small refinement network to achieve state-of-the-art results on the alphamatting.com benchmark [12] and their Composition-1K benchmark, which would go on to become a popular image matting evaluation dataset [1]. They delivered these results by training the model on a large, custom, composited dataset, the Adobe Deep Matting dataset.

Prior to this paper, deep learning-based image matting was restricted due to hardware limitations and a lack of data. The Adobe Deep Matting dataset featured 431 images and corresponding ground truth alpha mattes. Using these alpha mattes, Xu *et al.* extracted and composited these foregrounds onto various backgrounds, allowing for thousands of composite images in the dataset. In order to avoid repetition and allow for better generalization, they randomly augmented the image-trimap pairs in various ways, including random crops, flipping, and trimap dilation. Both this training dataset and training technique were used extensively in following deep learning-based matting works [2, 19, 20, 21].

Cai *et al.* improved upon DIM by breaking the matting pipeline down into two subtasks: trimap adaptation and alpha matte estimation [22]. Using a single encoder and two decoders, one for each sub-task, this pipeline first infers the global structural semantics on the input image and modifies the trimap to better suit the true unknown regions of the image. This was based on the insight that matting networks work better on more accurate, finer trimaps rather than coarse ones. Following the trimap adaptation step, the alpha matte is generated by the second decoder and passed to the propagation unit. This propagation stage, composed of 3 successive units, was created with propagation-based matting in mind, and utilized ResBlocks [23] in conjunction with convolutional long short term memory (LSTM) cells. The former extracts input features while the latter preserves memory as the outputs are passed to the next propagation unit. This disentangled approach achieved state-of-the-art results, hinting at the value of breaking down the task of matting.

Lu *et al.* proposed IndexNet, claiming indices-guided unpooling in the decoder as a superior alternative to upsampling [19]. IndexNet is introduced as a flexible network module which plays a large role in their index-guided encoder-decoder framework. They found that this approach was able to retain boundary details more effectively than the traditional upsampling-based approach. This approach was able to beat DIM by a considerable amount while using a much lighter MobileNetV2 [24] backbone.

Finally, Forte and Pitié introduced a low-cost modification to the DIM method in F, B, α (FBA) Matting [2]. This modification allowed matting networks to predict foreground and background colours in addition to the alpha matte. These colours are estimated directly from the same encoder-decoder as the matte, allowing for a single network to perform the entire task. Some minor alterations were made to the ResNet-50 [23] encoder and the output is modified to allow for 7 channels. This method achieved state-of-the-art results while remaining efficient, both in terms of computational and memory cost.

2.2.2 Trimap-Free Works

While deep learning-based approaches that required a trimap made substantial progress in image matting, the requirement of an auxiliary input — particularly, one as challenging to create as a trimap — left image matting inaccessible to casual users. With this in mind, many researchers pivoted to trimap-free approaches, which sought to eliminate the need for a trimap as an additional input. A few of these works have attempted some form of trimap generation [6, 25, 26], as discussed in Chapter 2.2.2.

Chen *et al.* debuted Semantic Human Matting (SHM), an exciting portrait matting method, which used a semantic segmentation network, T-Net, to generate a trimap, and a matting network, M-Net, as the matting stage [6]. Together, these two networks, along with a fusion module, learned both coarse semantics and fine details to estimate alpha mattes accurately.

Zhang *et al.* proposed A Late Fusion CNN for Digital Matting (LF), a method that uses a single encoder with two decoder branches for foreground and background classification [5]. The outputs from the decoders are then fused and blended to estimate the final alpha matte. This approach was pursued with the goal of achieving more degrees of freedom with a second decoder branch, allowing for better alpha matte estimations. LF managed to achieve state-of-the-art results on their custom benchmark, composed of human subjects from the internet and from the Composition-1k dataset from the DIM paper [1].

Liu *et al.* use a three-stage design, which involves a mask prediction network, a quality unification network, and a matting refinement network, to perform trimap-free matting [27]. The mask prediction network, trained on both coarse and fine data, predicts a coarse semantic mask which is fed into the quality unification network. Here, the predicted mask is adjusted depending on its quality — the quality will be improved for coarse masks, while the quality for fine masks is lowered — and is passed on to the matting refinement network. By correcting the quality of the mask in the previous stage, the matting refinement stage receives a consistent quality of input and is able to create a final alpha matte estimation.

A different approach, Background Matting, was first introduced by Sengupta *et al.* in which an image of the background, without the foreground subject, is used as an additional input instead of a trimap [28]. Although this design achieved state-of-the-art results on an altered version of Composition-1K, its biggest limitation was the requirement of a background image. The user would need to have the foresight to take a background image, which would have to be nearly identical to the original composition — with regard to framing and lighting — for a successful matte. The method also fails with dynamic backgrounds such as moving water or a car driving by in the background.

Qiao *et al.* tried a different approach to trimap-free matting when they created Hierarchical Attention Matting (HAtt) [4]. This method leveraged spatial and channel-

wise attention as a novel approach to trimap-free matting. This design achieved state-of-the-art performance on Composition-1K.

Another exciting work was the MODNet paper by Ke *et al.* [14]. The authors proposed a trimap-free approach which did not require any supplementary inputs, including trimaps and background images. MODNet decomposes the matting problem into three sub-objectives: semantic estimation, detail prediction, and semantic-detail fusion. These sub-objectives are optimized simultaneously through a single, light-weight network. Their justification for this approach was that neural networks are more effective at learning a set of simple objectives as opposed to learning a single, complex objective. With this design, MODNet achieved real-time inference, outperforming other trimap-free methods in terms of both speed and accuracy [14]. The method achieved state-of-the-art results, with respect to trimap-free approaches, and delivered remarkable visual results. The study showed that their model performed well in predicting hollow structures and hair details, but occasionally failed with some poses and clothing.

Lin *et al.* improved upon the Background Matting paper [28] by putting forward a two-network architecture: a low-resolution base network and a high-resolution refinement network which operates on selective patches [13]. Background Matting V2's improved design resulted in a real-time background matting solution that improved upon the original state-of-the-art approach by Sengupta *et al.* in both the speed of inference and the resolution at which it operates [13, 28].

As mentioned earlier, when it comes to deep learning-based image matting, datasets play a crucial role. Large-scale datasets help matting networks generalize to real-world images. Another major contribution from this study was the creation of two datasets: VideoMatte240K and PhotoMatte13K/85. Aptly named, the datasets contain 240,000 frames from high-resolution videos and 13,000 high-quality images, respectively. Quality alpha matte annotations require skilled human artists and a painstakingly tedious labelling process; the reason why creating image matting datasets is quite burden-

some. Their included ablation study demonstrated improved results when using the high-resolution datasets, solidifying the case for these datasets.

The primary advantage of a trimap-free approach is that the user does not need to supply additional inputs to the input image. However, like the original Background Matting paper by Sengupta *et al.* [28], Background Matting V2 requires an image of the background at the time of capture of the input image [13]. This additional, inconvenient step renders the model ineffective should the user forget to capture the background image at the time the original picture was taken.

Li *et al.* introduced Glance and Focus Matting (GFM), in which they used a shared encoder and two decoders to solve two sub-objectives: segmentation and matting [3]. The glance decoder executes the former sub-objective, while the focus decoder performs the latter. They also debuted a large, high-quality background dataset, BG-20K, with contained no salient objects, as well as a diverse benchmark with real-world images to help judge a model's generalization ability. This method achieved state-of-the-art results on this benchmark, further supporting the argument for solving sub-objectives.

Finally, Li *et al.* proposed P3M-Net — the work that introduced the P3M-10K dataset that we use — which utilizes a multi-task framework to tackle the image matting problem, all while introducing Privacy-Preserving Training (PPT) to image matting [15]. Within portrait matting, PPT refers to training models with anonymized images, created by blurring the identifiable areas of the face, with the goal of preserving the privacy of individuals in the data. The authors achieved this by introducing the P3M-10K dataset, which contains 10,000 high-quality image and ground truth alpha matte pairs. Additionally, they introduced the P3M-500-P and P3M-500-NP evaluation benchmarks that we use as well. The former benchmark contains blurred faces like the training dataset, whereas the latter contains normal faces, in order to analyze the generalization capability of the model.

As for the network architecture of P3M-Net, like GFM [3], they combined a single

encoder for learning basic visual features, a segmentation decoder, and a matting decoder. Additionally, they use three modules: a tripartite-feature integration module, a deep bipartite-feature integration module, and a shallow bipartite-feature integration module to model interactions between the encoder and two decoders, the encoder and segmentation decoder, and the encoder and matting decoder, respectively [15]. This highly-interconnected structure was formulated with the goal of emphasizing the interactions between each branch and the encoder.

Li *et al.* reimplemented some of the top trimap-free methods and trained them on P3M-10K in order to compare them with P3M-Net [15]. Their design achieved state-of-the-art results on both P3M-500-P and P3M-500-NP; the latter result suggesting their model generalized well to normal faces, despite being trained on blurred faces. At the time of writing, P3M-Net remains the best-performing model on these two benchmarks.

Trimap Generation

In many previous image matting works, it was assumed that a trimap was readily available as input. Unfortunately, this can be a tedious and often labour-intensive task, particularly for casual users. In recent years, different trimap-free works have attempted to tackle the image matting problem with various tools such as alternate supplementary inputs [13, 28, 29], while a few works focused on solving matting sub-objectives in order to circumvent the historical need for a trimap. Others decided to overcome this obstacle by introducing various methods to generate a trimap from the input image, allowing these methods to remain trimap-free from the user's perspective.

Trimap generation was not necessarily a new innovation for deep learning-based, trimap-free methods. In fact, there had been a few attempts at automatic trimap generation using traditional computer vision techniques [30, 31]. However, these generated trimaps still had ample room for improvement as they struggled with similar foreground and background colours.

When the interest in deep learning-based trimap-free methods began increasing, researchers turned to deep learning for this task. Shen *et al.* used a trimap generation step in their network structure [25]. They modelled this step as a pixel classification problem and use a slightly modified semantic segmentation network to classify each pixel as belonging to one of three classes: foreground, unknown, and background.

Chen *et al.* [6] utilized a trimap generation network with a similar function as [25]. This network also worked at the pixel level by using semantic segmentation to label pixels as either foreground, unknown, or background. The output 3-channel trimap is concatenated with the 3-channel image, resulting in a 6-channel input for the subsequent matting stage — many networks, such as DIM, use a 4-channel input: a 3-channel image concatenated with a 1-channel trimap [1, 6].

Zhou *et al.* introduced a semantic-guided trimap generation network, based on a modified DeepLabv3 [32] encoder, requiring both an RGB image and a soft segmentation as inputs into their network [33]. The requirement of a soft segmentation for this network does not address the need of eliminating all additional inputs but rather replaces a trimap with a segmentation as an additional requirement.

2.3 Summary of Related Works

All of the aforementioned works, collated and shared below in Table 2.1, have contributed to and shaped the landscape of image matting research. These works offer many insights into both the advantages and disadvantages of the various techniques used to tackle the matting problem. We took these into account when designing our approach and decided to pursue a modular design to matting.

While some recent works opted to break matting down to sub-objectives which could be solved using a single network [14, 15], our modular approach would require breaking down the matting pipeline into a series of smaller tasks, each solved with separate

networks. While both Background Matting and Background Matting V2 eliminate the need for a trimap, they remain reliant on a background image as another input [13, 28]. Our design, based on SHM [6], only requires an input image. However, rather than treating trimap generation as a multi-class segmentation problem, as demonstrated in SHM [6] and Boosting SHM [27], we opt to further break the trimap generation task into two sub-tasks: binary semantic segmentation and trimap generation. Furthermore, we choose to create the more commonly used 1-channel trimaps, rather than the 3-channel trimaps created in these works. Lastly, we concatenate the input images with these generated trimaps and pass them into various trimap-based works [1, 2, 19]. By creating and subsequently passing generated trimaps into proven trimap-based networks, we hope to bridge the gap between the convenience of trimap-free works and the performance of these trimap-based works.

Table 2.1: Breakdown of Related Works and Required Inputs for Each Method

Type	Method	Year	Trimap-Free	Inputs
Traditional	Matting Patent	1998	✗	Image, Trimap
	Bayesian Matting	2001	✗	Image, Trimap
	Poisson Matting	2004	✗	Image, Trimap
	Closed-Form Matting	2008	✗	Image, Trimap
Deep Learning	Deep Image Matting	2017	✗	Image, Trimap
	Disentangled Image Matting	2019	✗	Image, Trimap
	IndexNet	2020	✗	Image, Trimap
	F, B, α Matting	2020	✗	Image, Trimap
	Semantic Human Matting	2018	✓	Image
	Late Fusion Matting	2019	✓	Image
	Boosting Semantic Human Matting	2020	✓	Image
	Background Matting	2020	✓	Image, Background
	Hierarchical Attention Matting	2020	✓	Image
	MODNet	2020	✓	Image
	Background Matting V2	2020	✓	Image, Background
	Glance and Focus Matting	2021	✓	Image
	P3M-Net	2021	✓	Image
	Semantic-Guided Automatic Matting	2021	✓	Image, Segmentation

Chapter 3

A Modular Design for Portrait Matting

Our approach to designing a modular image matting network stems from one main intuition: neural networks may perform better on a series of simpler steps as opposed to one large, complex task. Some works have demonstrated success in trimap-free matting by breaking down the process using multiple networks [6, 26, 27], others have attempted it by solving sub-objectives within a single network [3, 14, 15]. Building on the former approaches, a modular design may allow us to bridge the gap between trimap-based and trimap-free matting, while also introducing the possibility of interactivity in trimap-free matting, discussed further in Chapter 5.1. We choose to explore the feasibility of modular image matting by generating viable trimaps to use with proven trimap-based networks. Our three-step trimap-free matting process starts with the semantic segmentation stage.

3.1 Semantic Segmentation

As with other trimap-free methods, the reasoning for semantic segmentation as the preliminary network in our design is to generate the rough outline of the foreground subject in the image. This first stage can be depicted visually, as shown in Figure 3.1, and

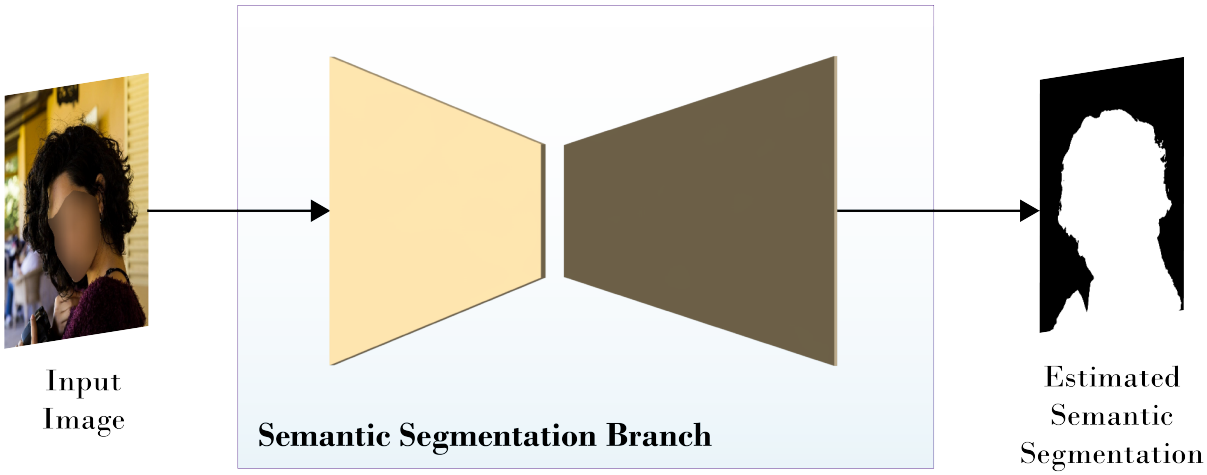

Figure 3.1: The design of the semantic segmentation stage. This network takes an image as input and returns a segmentation as output, which is passed on to the next stage of our modular design.

mathematically as:

$$\mathcal{S}(\mathbf{x};\theta_s) \tag{3.1}$$

Here, $\mathcal{S}$ is the semantic segmentation network and $\mathbf{x}$ is the input image. As shown in both depictions, the segmentation network requires solely an image as input and it outputs a semantic segmentation.

This stage may be the most important in our modular design as it acts as the foundation upon which the rest of our matting process is built upon; any major successes and failures will likely come down to the performance of the segmentation stage. Without this vital step, our matting stage would essentially be approaching the problem blind. Matting networks need a defined region to operate in. This region can be provided through a user-defined trimap or through an automatically generated segmentation with some form of subsequent processing. Otherwise, they end up failing.

For our experiments, we alternate between two semantic segmentation network structures: UNet [34] and UNet++ [35]. Both models use an EfficientNet-B4 [36] backbone, pretrained on ImageNet weights [37]. These models were obtained through the Segmen-

tation Models library [38] and were fine-tuned on the P3M-10k dataset with a starting learning rate of 1×10^{-3}, using inferred ground truth segmentations. This ground truth creation process is described in detail in Chapter 4.1.2. Both networks were tuned using Dice Loss,

$$DL = 1 - \frac{2y\hat{y} + 1}{y + \hat{y} + 1} \quad [39]$$

Here, y is the ground truth segmentation mask and $\hat{y}$ is the predicted segmentation mask. Both semantic segmentation models were trained in isolation and any case in which they were further tuned to the entire modular architecture design will be indicated. This process is detailed later, in Chapter 4.5.2.

Our motivation for using two models in this study is to demonstrate one of the main advantages of a modular design: flexibility. Our design allows users to replace and choose their network of choice for each stage, provided it is compatible with respect to its inputs and outputs. Users may have different preferences based on inference time, accuracy, and computational restrictions, and the freedom to choose networks based on these preferences is the main reason for pursuing a modular design. We take a look at the impact on our results of swapping between the two segmentation networks in Chapter 4.5.3.

3.2 Trimap Generation

Without converting the output segmentation from the preceding network into a trimap, the matting stage would only be able to operate within the bounds of the segmentation. By converting it to a trimap, we are able to provide the matting network with defined foreground, unknown, and background regions. The matting network can then operate solely within the unknown region.

In order to perform trimap generation, we first take advantage of the semantic segmentation stage, as explained above. The segmentation created by the preceding stage

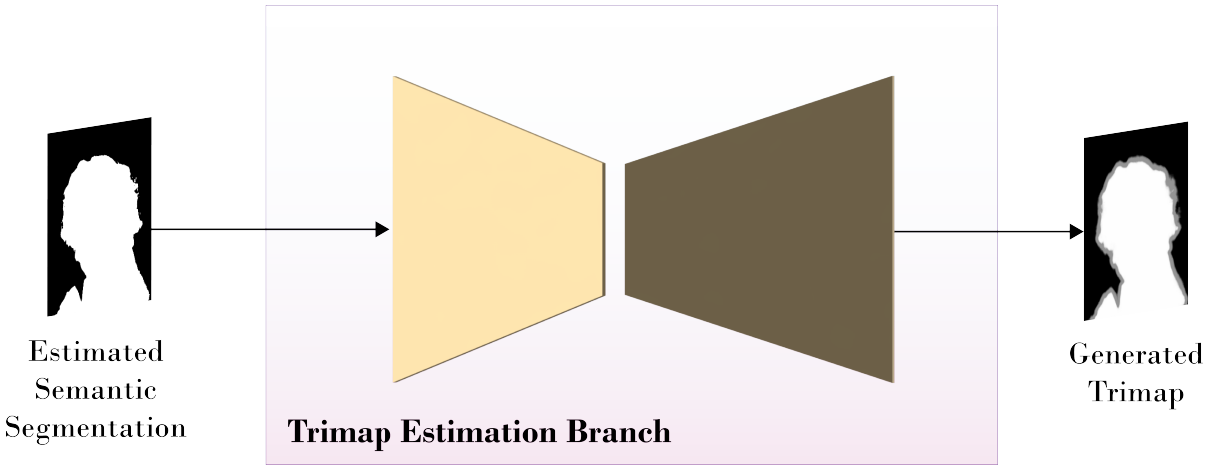

Figure 3.2: The general design of the trimap generation stage. This network takes the outputted segmentation from the previous semantic segmentation stage as input and returns a trimap as output. This trimap is then passed on to the next stage of our modular design.

gives us a rough outline of where the foreground subject is in the image, without having to worry about fine structures such as hair and fabric. This is followed by the trimap generation stage, which is illustrated in Figure 3.2 and represented mathematically as:

$$\mathcal{T}\left(\mathcal{S}(\mathbf{x};\theta_s);\theta_t\right) \tag{3.2}$$

Here, $\mathcal{T}$ is the trimap generation network, $\mathcal{S}$ is once again the semantic segmentation network, and $\mathbf{x}$ is the input image. As shown in both the network illustration and the mathematical representation, the trimap generation network uses the segmentation created by the preceding semantic segmentation network as the input and returns an estimated trimap as output.

Our trimap generation network is trained on ground truth trimaps, created by eroding and dilating ground truth alpha mattes. This process is further detailed in 4.1.2. The objective of our trimap generation network is to erode and dilate the edges of the aforementioned semantic segmentation estimation. These modifications to the segmentation give us an unknown region, in which the matting stage can operate. By both dilating and eroding, we give the matting network a bit more room to operate in both

the estimated foreground region and background region, allowing for small mistakes by the segmentation stage to be remedied.

Our trimap generation network follows a simple encoder-decoder structure, as shown in Figure 3.4. This autoencoder was trained using Mean Squared Error (MSE) Loss,

$$MSE = \frac{1}{N}\sum_{i=0}^{N}(y_i - \hat{y}_i)^2 \quad [40]$$

Here, $\hat{y}_i$ is the predicted value at pixel i, whereas y_i is the ground truth value at that pixel. This network architecture and training setup allowed us to generate fairly accurate trimaps. At this stage, with the preceding semantic segmentation network and the current trimap generation network working in tandem, we are able to produce reasonable trimaps while only requiring an image as input.

While similar results can be achieved by applying erosion and dilation through a tool like OpenCV, we choose to create a network because it allows our entire matting architecture, from segmentation through to matting stages, to be differentiable. Thereby allowing us to train the entire design together, should we choose. We dive deeper into this idea in Chapter 4.5.2.

3.3 Image Matting

The final stage of our modular design is the image matting step. This stage, shown in Figure 3.3, takes the original image concatenated with the generated trimap from the previous stage as input and returns the final alpha matte as output. This is depicted mathematically as:

$$\mathcal{M}\left(\mathbf{x} \oplus \mathcal{T}\left(\mathcal{S}(\mathbf{x};\theta_s);\theta_t\right);\theta_m\right) \tag{3.3}$$

Here, $\mathcal{M}$ is the matting network, $\mathcal{T}$ and $\mathcal{S}$ are the trimap generation and semantic

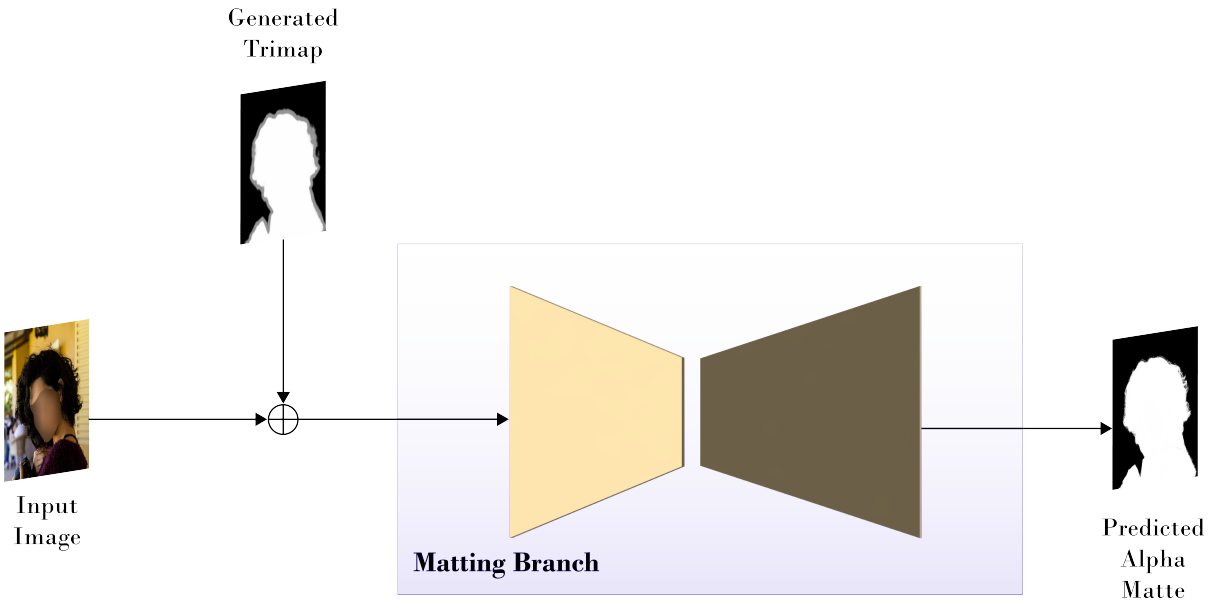

Figure 3.3: The design of the image matting stage. As input, this network uses the original image concatenated with the previously generated trimap. It produces and returns the final alpha matte estimation as the output.

segmentation networks, respectively, and $\mathbf{x}$ is the input image.

For the matting stage of our modular design, we opted to use pretrained models from some of the most popular trimap-based image matting works. These are Deep Image Matting [1], IndexNet [19], and FBA Matting [2]. These are image matting works which have been shown to provide excellent results, albeit relying on a trimap as additional input. Again, one of the main benefits of our modular design is that we are granted the freedom to swap out and replace the networks at each stage.

In addition to comparing our architecture results to leading trimap-free methods, we also compare the results of utilizing these pretrained models within our networks with generated trimaps versus the results of passing in ground truth trimaps. This way, we can determine whether or not our modular design is a viable alternative to the original trimap-based approach to these works. This process is further detailed in our ablative study, discussed in Chapter 4.5.1.

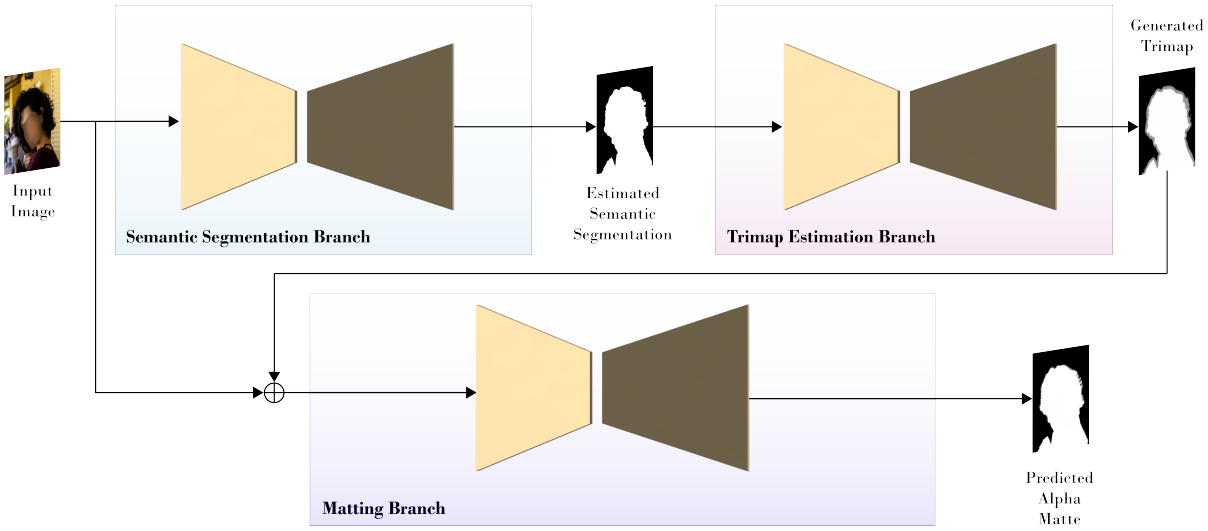

Figure 3.4: The entire modular image matting architecture. Each stage is depicted as a general encoder-decoder network but can be replaced with compatible networks depending on the user's preferences.

3.4 Network Architecture

Our main experiment explores the viability of a modular image matting network architecture for the purposes of trimap-free image matting. As mentioned above, we use a semantic segmentation network, $\mathcal{S}$, for the first stage of our network, a trimap generation network, $\mathcal{T}$, for the second stage, and finally an image matting network, $\mathcal{M}$, for the third and final stage. This is depicted mathematically as:

$$\mathbf{y} = \mathcal{M}\left(\mathbf{x} \oplus \mathcal{T}\left(\mathcal{S}(\mathbf{x}; \theta_s); \theta_t\right); \theta_m\right) \tag{3.4}$$

Here, $\mathbf{x}$ is the input image and $\mathbf{y}$ is the resulting estimated alpha matte. In its entirety, the general design of our network can be seen in Figure 3.4. In summary, the first stage, the semantic segmentation network, produces a semantic segmentation using the original image as input. This segmentation is passed onto the trimap generation stage, which produces a trimap from the segmentation. This generated trimap is concatenated with the original image and passed into the final matting stage, which produces and returns

the estimated alpha matte as output. As part of our modular design, the networks at each stage can be swapped out and replaced with other compatible networks, as desired by the user.

An example network design for one of our variants, DIM (UNet) can be seen in Figure 3.5. This variant utilizes a fine-tuned UNet for our semantic segmentation stage, our new trimap generation network for the second stage, and a pretrained Deep Image Matting model [1] for the image matting stage. Like all of our other test variants, we use a single image as input for the first network, the semantic segmentation stage. This stage outputs a semantic segmentation, which is then fed into the second network, the trimap generation stage. This network outputs a trimap based on the inputted semantic segmentation. The generated trimap is then concatenated with the original input image and fed into the final image matting stage, which produces the final output alpha matte.

As mentioned earlier, one of the main advantages of a modular design is flexibility. The end user can decide which networks work best for their needs and constraints at each stage. We perform our experiments on different variants of our design to demonstrate how swapping networks can affect the estimated alpha mattes. These variants are listed below, with specified semantic segmentation networks in parentheses:

- DIM (UNet);
- DIM (UNet++);
- DIM (UNet++, Fine-Tuned);
- IndexNet (UNet);
- IndexNet (UNet++);
- FBA Matting (UNet); and
- FBA Matting (UNet++).

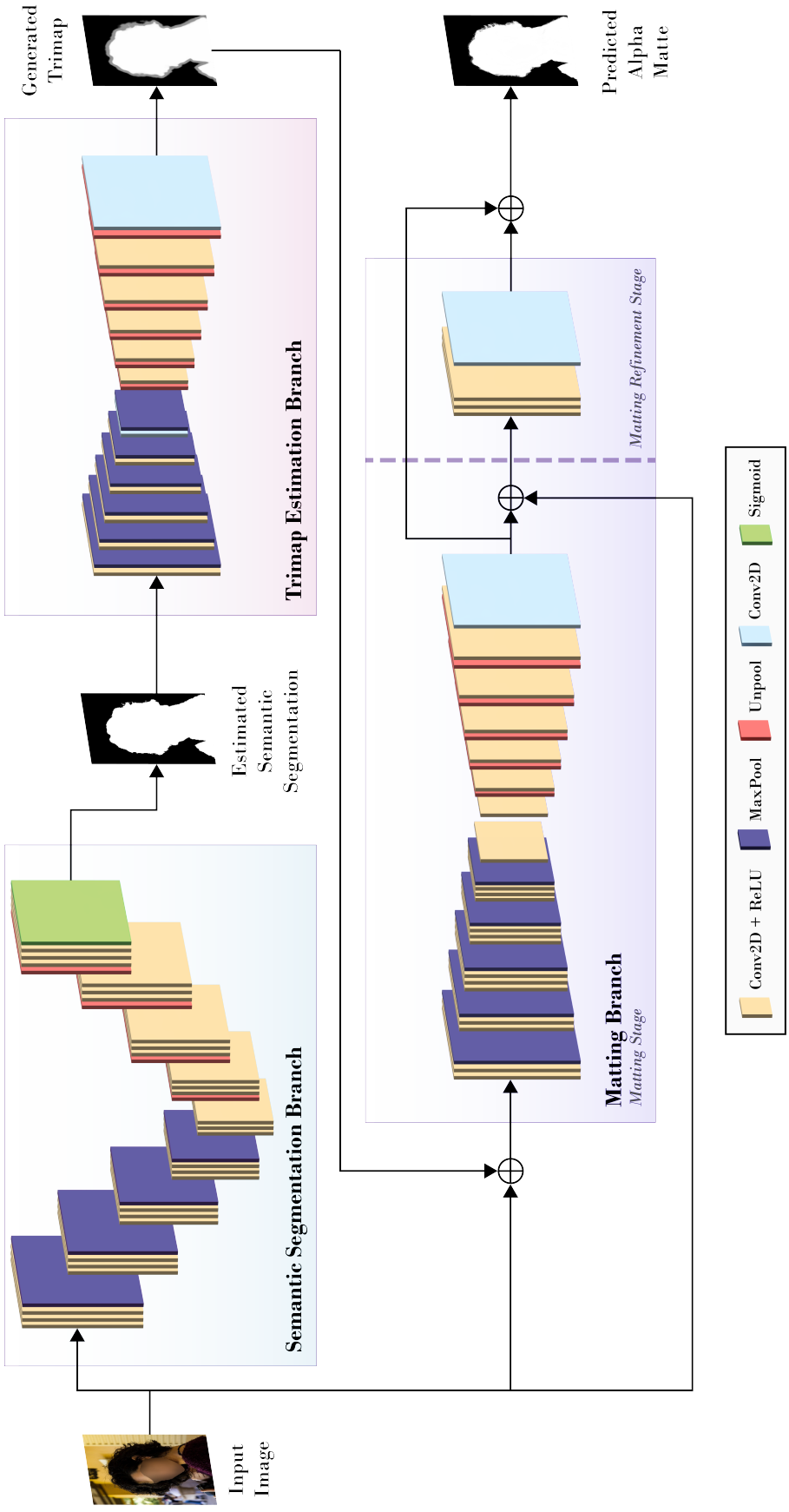

Figure 3.5: The DIM (UNet) Variant of our design. The segmentation branch is UNet and the matting branch is from [1].

Chapter 4

Experiments

4.1 Datasets

The rise of deep learning-based approaches in image matting can partially be attributed to large-scale datasets like the Adobe Deep Matting dataset introduced by Xu *et al.* in 2017 [1]. Deep learning-based approaches require massive amounts of data in order for patterns to be recognized. Consequently, the datasets required for these deep learning-based approaches required upwards of tens of thousands of images, each with a high-quality corresponding mask. Unfortunately, high-quality ground truth alpha mattes are difficult to annotate and the laborious process often requires domain experts. In order to save time and money, earlier deep learning-based image matting approaches relied on composited datasets. Here, a single image and its corresponding high-quality alpha matte annotation were used to extract a foreground. This foreground was then composited onto a number of random backgrounds, thereby allowing multiple variations of the input images, while only requiring the single, original alpha matte. In order to allow generalization to real-world, natural images, training techniques such as augmentations were utilized [1]. However, natural training images remain the preferable option for allowing more accurate alpha matte estimations on natural test images.

Thankfully, a few datasets have been introduced over the past few years which are composed of high-quality alpha matte annotations on natural images [13, 14]. One of those datasets is the Privacy Preserving Portrait Matting (P3M-10K) dataset [15], which we chose to utilize for our portrait matting research. Aside from being composed of 10,000 natural images and high-quality alpha mattes, this dataset also has another unique property which piqued our interest: it blurred out the faces of all foreground subjects. An example of a blurred image can be seen in Figure 4.1.

4.1.1 Privacy in Deep Learning

Recently, there has been a lot of debate and controversy revolving around privacy in technology, mainly in deep learning. The P3M-10K dataset was introduced to address the problem of privacy in deep learning training data [15]. As mentioned earlier, annotating ground truth alpha mattes is a painstaking task, making it difficult to obtain high-quality ground truths. Fortunately, P3M-10K contains thousands of real-world images, with corresponding high-quality alpha mattes, and a variety of foreground subjects in various poses in front of complex backgrounds.

However, one concern with this type of approach was how well networks trained on blurred facial data would translate to normal, everyday images of human subjects. Fortunately, this dataset was shown to generalize to clear images of faces, as well as partially obstructed faces due to face masks [15]. With all these factors being considered, the P3M-10K dataset was ideal for our research purposes.

4.1.2 Ground Truth Segmentations and Trimaps

Although the P3M-10K dataset provides both high-quality images and corresponding ground truth alpha matte annotations, our network architecture requires additional ground truths for training our network, including segmentation masks for the semantic segmentation branch and trimaps for the trimap generation branch. Accordingly,

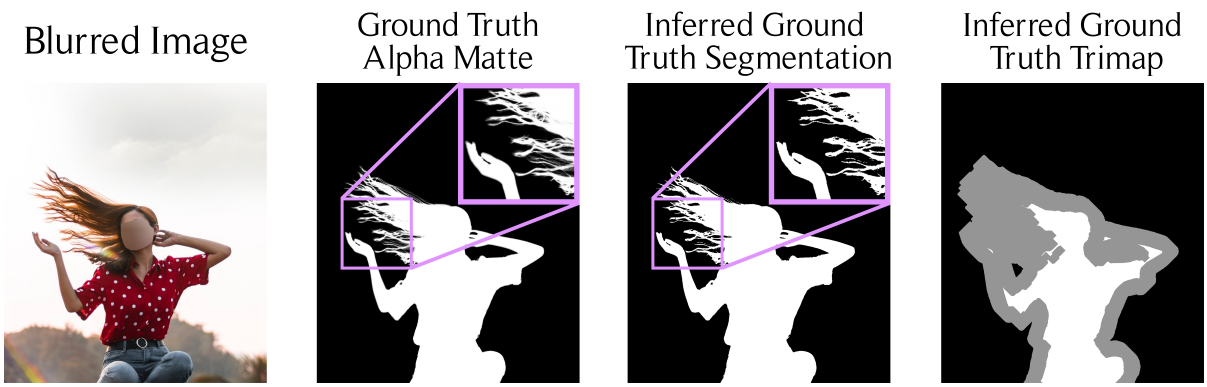

Figure 4.1: Generating the additional ground truths from the alpha matte.

these labels were able to be inferred from the ground truth alpha mattes, as shown in Figure 4.1.

The segmentation masks were created by applying OpenCV's binary thresholding to the alpha matte, allowing for binary semantic segmentation [41]. In the case of binary semantic segmentation, each pixel will be classified as belonging to the foreground or to the background, either belonging to class 1 or class 0. On the other hand, in the case of image matting, each pixel is given an alpha value of 1.0 if it is in the foreground, 0.0 if it belongs to the background, or any value in between, depending on its transparency. This results in more detail being preserved in the alpha matte as compared to the semantic segmentation, as shown in the aforementioned figure.

As for the ground truth trimaps, they were created by eroding and dilating the corresponding ground truth alpha mattes, as described by [1]. We used OpenCV here as well [41]. Eroding and dilating the edges of the alpha matte allow for a reasonable estimation of the unknown region that lies between the foreground and background on boundary regions of the alpha matte, as demonstrated in Figure 4.1. These inferred ground truths attest to the versatility granted by an accurate alpha matte and, by extension, the power of image matting.

4.2 Metrics

We measure our experiments with three metrics: Sum of Absolute Differences (SAD), Mean-Squared Errors (MSE), and Mean of Absolute Differences (MAD). These three metrics are among the most common metrics used in image matting works and were used in [15]. We use the same values as them in our calculations.

For the trimap-based approaches, these values are calculated within the unknown regions of the ground truth trimaps. For the trimap-free approaches, these values are calculated across the whole image, in order to account for any potential errors in the segmentation stage, as well as the unknown regions of the ground truth trimaps. For each metric, the differences are summed over every pixel in the image for whole image metrics or across every pixel in the foreground for the trimap unknown region metrics. For MSE and MAD, the summed value is then divided by the number of pixels.
The SAD equation is:

$$SAD = \frac{1}{1000} \sum_{i=0}^{N} |y_i - \hat{y}_i|$$

Next, the equation for MSE is:

$$MSE = \frac{1}{N} \sum_{i=0}^{N} (y_i - \hat{y}_i)^2$$

Lastly, for the equation for MAD is:

$$MAD = \frac{1}{N} \sum_{i=0}^{N} |y_i - \hat{y}_i|$$

For each metric, $\hat{y}_i$ is the predicted value at pixel i and y_i is the ground truth value at that pixel.

4.3 Results

In each of the tables showing our results below, our networks are composed of a specified segmentation stage, our trimap generation network, and the stated matting network. More details on the setup of each stage can be found in Chapter 3. All three networks were trained independently, unless otherwise specified. The semantic segmentation network will either be a UNet or UNet++ model, both pretrained with an EfficientNet-B4 backbone using ImageNet weights, and further trained on the P3M-10K inferred ground truth segmentations. The trimap generation stage for these tests is the encoder-decoder network we used trained on the P3M-10K ground truth trimaps we generated.

Finally, for the matting stage, we use DIM, FBA Matting, and IndexNet models, which were all pretrained on the Adobe Deep Matting dataset [1, 2, 19]. FBA Matting, however, made a few modifications to the dataset to allow for more accurate augmentations. The DIM model labelled as fine-tuned was tuned end-to-end across the entire design with the UNet++ semantic segmentation network, our trimap generation network, and the DIM pretrained network. We describe this version of our design in more detail in Chapter 4.5.2. In both Tables 4.1 and 4.2, we compare the quantitative results from our design variants to other trimap-free methods. The values for the Late Fusion (LF), Hierarchical Attention Matting (HAtt), Semantic Human Matting (SHM), and Glance and Focus Matting (GFM) models were retrieved from [15]. Li *et al.* trained each model on the P3M-10K dataset and evaluated them on the P3M-500-P and P3M-500-NP benchmarks. It should be noted that the MODNet model is the pretrained model from [14] and was trained on their image matting dataset, not P3M-10K like the other models.

Starting with the quantitative results on the trimap unknown region of the P3M-500-P benchmark in Table 4.1, we see that our best-performing variant, FBA Matting with the UNet++ semantic segmentation network, performs favourably in the SAD metric compared to most trimap-free methods. The only exception being the state-of-the-art P3M-Net. However, our design is edged out by GFM and SHM in the MSE metric and

Table 4.1: Performance on P3M-500-P and P3M-500-NP Benchmark with Trimap-Free Matting Networks (Trimap Unknown Region). * Denotes Values from [15].

Network	P3M-500-P			P3M-500-NP		
	SAD ↓	MSE ↓	MAD ↓	SAD ↓	MSE ↓	MAD ↓
MODNet (2020)	17.10	0.078	0.117	17.64	0.063	0.103
LF* (2019)	12.43	0.042	0.082	14.53	0.042	0.083
HAtt* (2020)	11.03	0.038	0.075	13.48	0.040	0.080
SHM* (2018)	9.14	0.026	0.055	9.14	0.026	0.055
GFM* (2021)	8.84	0.027	0.062	10.16	0.027	0.062
P3M-Net (2021)	**6.78**	**0.019**	**0.047**	**7.54**	**0.017**	**0.046**
DIM (Ours, UNet)	12.30	0.047	0.086	13.98	0.045	0.082
DIM (Ours, UNet++)	9.92	0.036	0.068	11.98	0.035	0.071
DIM (Ours, UNet++, Fine-Tuned)	9.58	0.035	0.065	11.25	0.035	0.065
IndexNet (Ours, UNet)	11.53	0.045	0.079	13.57	0.044	0.078
IndexNet (Ours, UNet++)	9.44	0.032	0.065	11.18	0.033	0.065
FBA Matting (Ours, UNet)	10.96	0.044	0.074	13.10	0.044	0.074
FBA Matting (Ours, UNet++)	8.55	0.029	0.058	10.31	0.031	0.059

Table 4.2: Performance on P3M-500-P and P3M-500-NP Benchmark with Trimap-Free Matting Networks (Whole Image). * Denotes Values from [15].

Network	P3M-500-P			P3M-500-NP		
	SAD ↓	MSE ↓	MAD ↓	SAD ↓	MSE ↓	MAD ↓
MODNet (2020)	137.98	0.073	0.080	110.03	0.055	0.063
LF* (2019)	42.95	0.019	0.025	32.59	0.013	0.019
HAtt* (2020)	25.99	0.005	0.015	30.53	0.007	0.018
SHM* (2018)	21.56	0.010	0.013	20.77	0.009	0.012
GFM* (2021)	13.20	0.005	0.008	15.50	0.006	0.009
P3M-Net (2021)	**8.49**	**0.003**	**0.005**	**10.90**	**0.003**	**0.006**
DIM (Ours, UNet)	27.54	0.012	0.016	31.11	0.013	0.018
DIM (Ours, UNet++)	22.9	0.010	0.013	25.53	0.015	0.011
DIM (Ours, UNet++, Fine-Tuned)	21.89	0.010	0.013	25.28	0.015	0.011
IndexNet (Ours, UNet)	26.35	0.012	0.015	30.69	0.014	0.018
IndexNet (Ours, UNet++)	22.41	0.009	0.013	25.24	0.011	0.015
FBA Matting (Ours, UNet)	25.57	0.012	0.015	30.03	0.014	0.017
FBA Matting (Ours, UNet++)	21.35	0.009	0.012	24.28	0.011	0.014

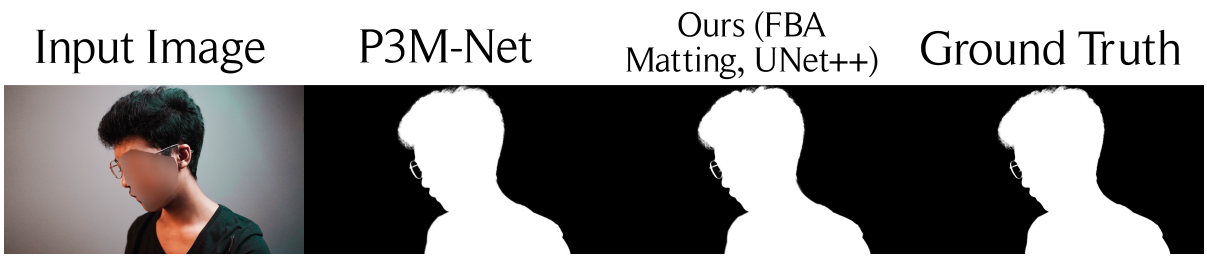

Figure 4.2: Example from P3M-500-P benchmark.

again by SHM in MAD. This carries over to the P3M-500-NP benchmark, where our top-performing method is beat out by SHM, GFM, and P3M-Net in SAD and MSE. It is also bested in MAD by SHM and P3M-Net. Our method does, however, score better a SAD, MSE, and MSE than MODNet, LF, and HAtt on both the P3M-500-P and P3M-500-NP benchmarks.

Moving over to the results on the whole image on the P3M-500-P evaluation benchmark in Table 4.2, our best-performing method outscores most trimap-free models, with the exception of GFM and P3M-Net, on all three metrics. Our fine-tuned DIM variant also manages to beat out MODNet, LF, and HAtt on all three metrics. It ties SHM in MSE and MAD, but scores slightly worse than in SAD.

As for the P3M-500-NP evaluation on the whole image, our top variant beats out MODNet, LF, and HAtt in SAD once again. HAtt does manage to score better MSE and MAD results. SHM, GFM, and P3M-Net outscore our model in all three metrics again. For both the P3M-500-P and P3M-500-NP benchmarks, there is a larger gap between the top-performing models on SAD compared to the lower scoring ones.

4.3.1 Visual Analysis

Seeing as how image matting is a visual task, the perceivable results play just as important of a role in the big picture as quantitative results. First, we examine the predicted alpha mattes between our design variants to see how modifying different stages of our modular design can impact the final alpha matte estimation. Furthermore, to measure the visual

performance of our design, we also compare our design's alpha matte estimations to both ground truth alpha mattes and the mattes generated from the current state-of-the-art model, P3M-Net.

Starting with some of the images from the P3M-500-P evaluation benchmark, shown in Figures 4.2 and 4.3, we see that in many cases, our design and P3M-Net perform similarly. More examples can be found in Figures C.1, C.2, C.3, C.4, C.5, and C.6, in the appendix. In Figure 4.3, our design preserves stray hairs and the outer boundary noticeably better, but P3M-Net manages to capture the hollow areas between the body and hair. This was a common theme amongst a large part of the benchmark.

One interesting takeaway was the fact that in quite a few images, such as the examples in Figure 4.4, our method managed to retain hair and other fine structures *better than the ground truth annotations*. Oddly, this means that our design was actually punished in these cases in the quantitative results above. This emphasizes the value of high-quality, and more importantly, accurate ground truth annotations. Granted, annotating ground truth alpha mattes is a very laborious undertaking, requiring meticulous attention to detail.

Figure 4.3: More examples from P3M-500-P benchmark.

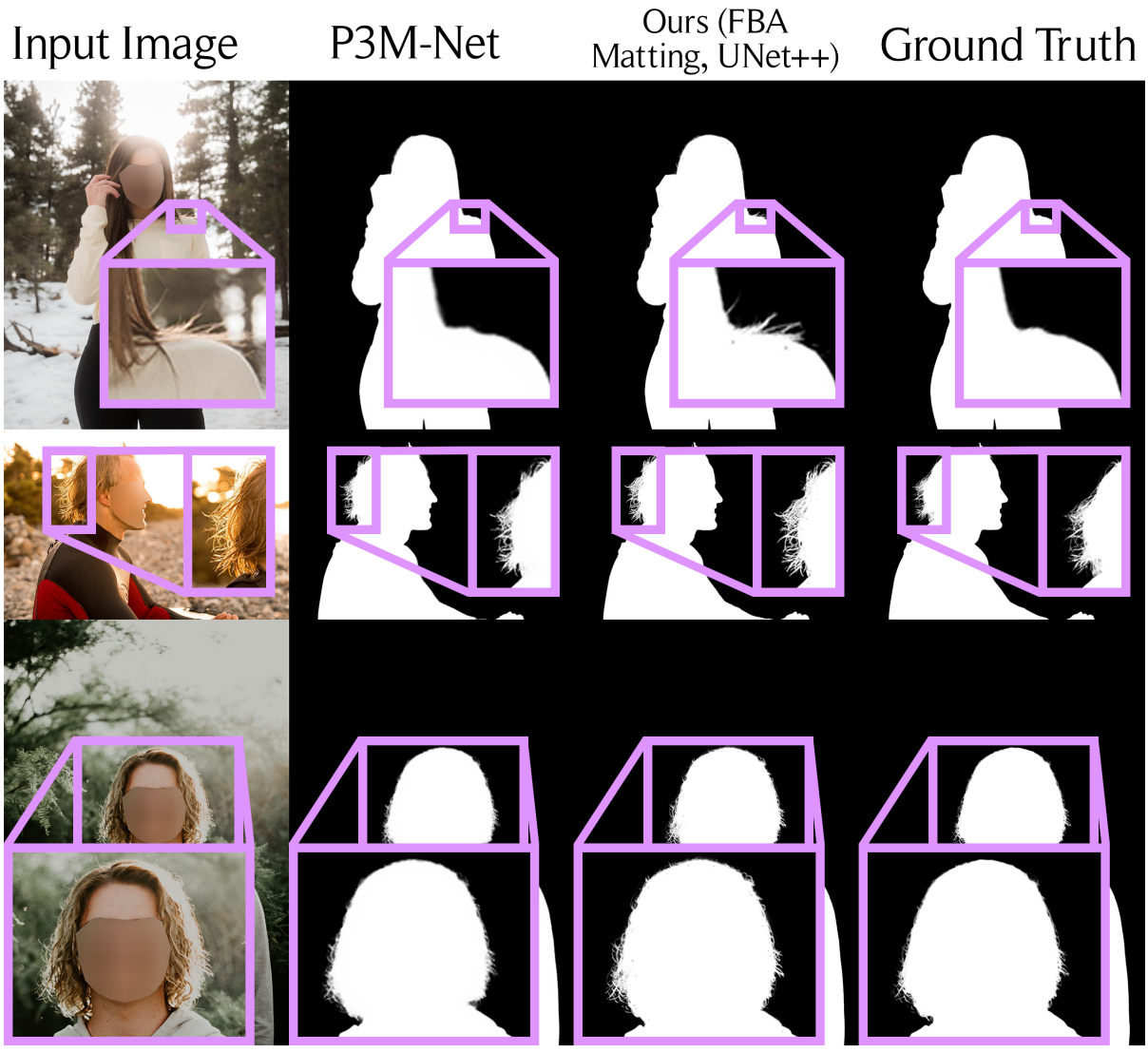

Figure 4.4: Fine detail retention in examples from P3M-500-P benchmark.

4.3.2 Visual Performance on Videos

We ran a few royalty free stock videos found on Pexels.com [42] through both networks — P3M-Net and FBA (Ours, UNet++) — and analyzed the resulting alpha matte predictions and foreground extractions. We include a few frames from the two videos and their corresponding alpha mattes in the figures below. Additional frames can be seen in Figures D.1 and D.2 in the appendix. For context, we chose to include the results from these videos because they exhibit different cinematographic properties.

The first video features a medium shot of a woman standing in a lake. The shot closes in to a medium close up and zooms back out to a medium shot. The background is of a medium complexity, with a blurred shore line in the distance and some clouds in an otherwise clear sky. The subject has very curly hair and the reflection of the sunlight off of the lake causes some interesting lighting effects on her skin and clothing.

The second video is quite a bit more complex as it involves a medium shot of a woman with long, straight hair jogging. The camera moves with the subject and maintains her position in frame. Because the camera is moving and so is the subject, we see that there is a lot of shaking throughout the video. With every step she takes, her hair changes positions considerably every few frames. Of course, as both her and the camera move, the background changes too. Collectively, this is quite a complex video in the perspective of image matting.

With that being said, we start off with Figure 4.5. Here, we see that both networks do a great job in the semantic segmentation estimation; the subject's entire body is preserved with no large artifacts. The differences start at the boundary regions of the alpha matte estimations. On the left, P3M-Net has slightly blurrier boundaries throughout, compared to our design on the right. The most noticeable difference is in the hair. On the right, we see clear outlines of individual strands of hair as well as curls being clearly defined. On the left, this the hair is much less defined. Although both are not quite perfect, the right seems to have done a better job capturing the fine details. This hints at the efficacy

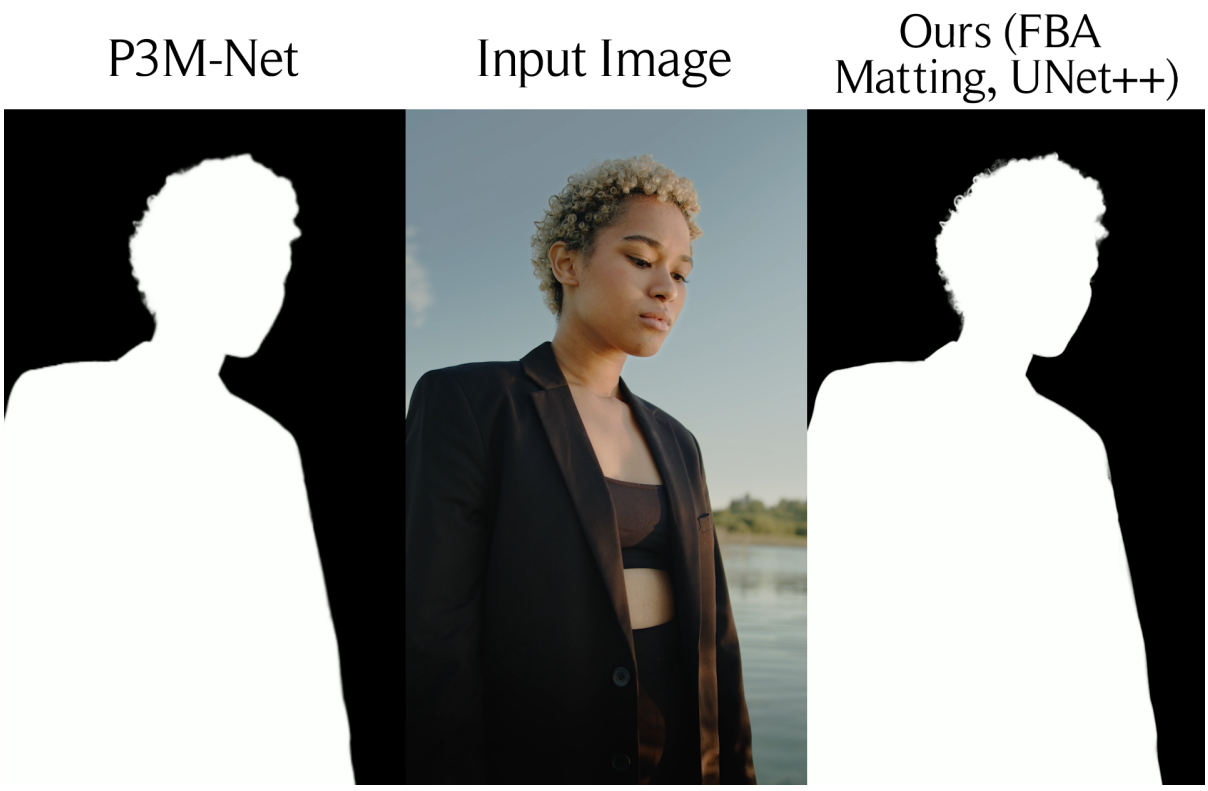

Figure 4.5: Visual performance on video of woman in water: medium shot.

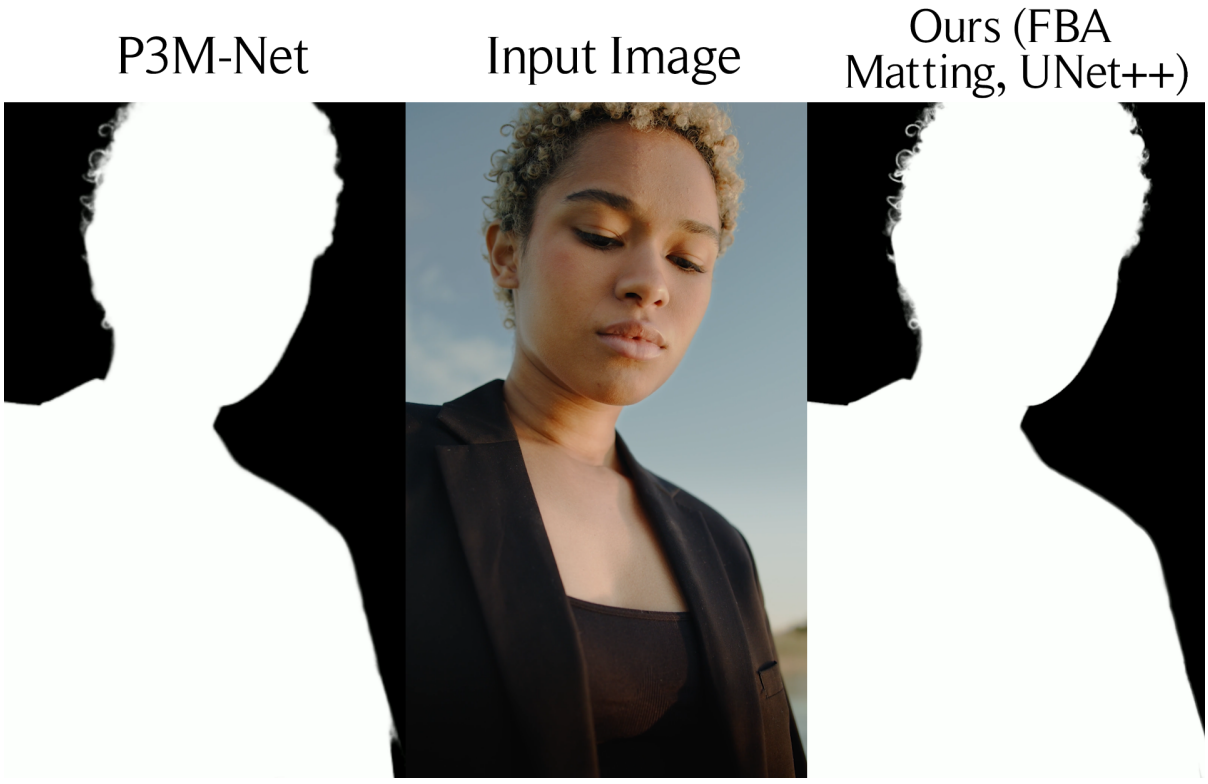

Figure 4.6: Visual performance on video of woman in water: medium close-up shot.

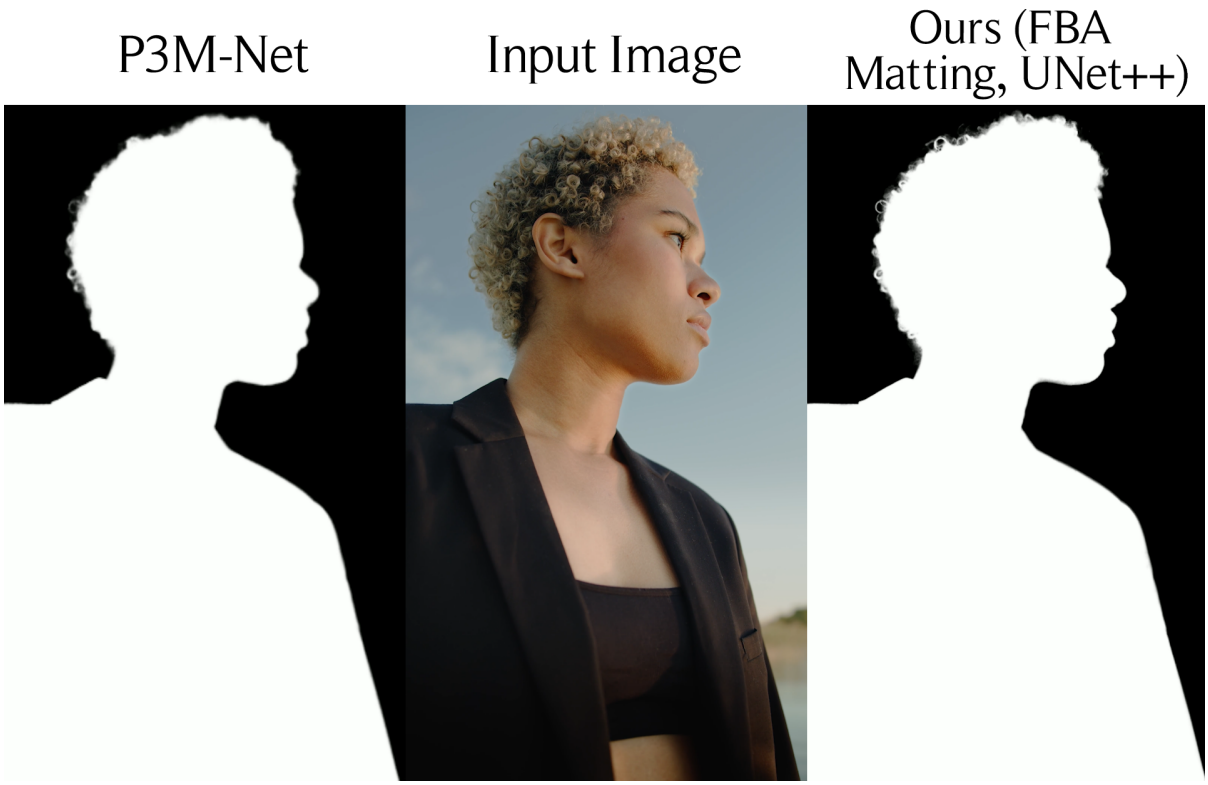

Figure 4.7: Visual performance on video of woman in water: medium close-up shot, from the side.

of using a proven trimap-based network for the matting stage, such as FBA Matting.

Next, we use a frame with a cropped-in shot in Figure 4.6. Here, we see that although there is a big improvement in the clarity of the fine structures on the left, the difference between P3M-Net and ours, on the right, is much more pronounced. Much more detail is preserved in the hair once again, with smaller structures absent in P3M-Net's estimated alpha matte.

The last frame from this video that we analyze shows the subject from her side, shown in Figure 4.7. Here, we see more of the same from the two networks. An excellent job with the initial corresponding semantic segmentation steps, but more fine structures and detail preserved with our design. Once again, this is most noticeable in the hair. While our design does a better job of identifying and singling out individual strands and curls, both designs leave some of the background in the final matte estimation.

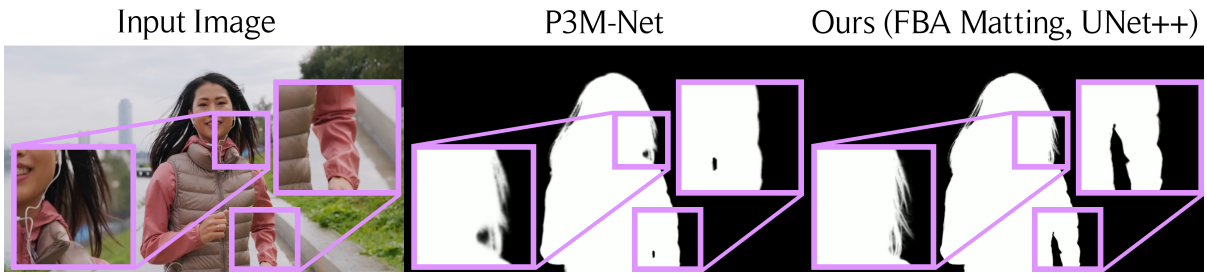

Figure 4.8: Visual performance on video of woman jogging: medium shot, frame 1.

Switching over to the second video, we see both models' predicted alpha mattes in Figure 4.8. Again, this video is much more complicated, in an image matting sense, so we expect to see more errors in these alpha matte estimations. Starting with the initial semantic segmentation estimation, both perform well on the subject's body, but P3M-Net in the middle does struggle with the gap in between her left arm and her torso. Going to her hair, we see that both networks struggle with defining the hollow areas in her hair. P3M-Net does define one area above her left shoulder that our model misses out on. In terms of the fine structure estimation, our design does a much better job of separating out individual strands of hair compared to P3M-Net. Again, this can be credited to a successful trimap generation and subsequent matting stage. These fine details being preserved demonstrates how a well-defined trimap coupled with a powerful trimap-based matting network can be used to create detailed alpha mattes.

The results on a frame later in the video, shown in Figure 4.9, show a case where the state-of-the-art P3M-Net, in the middle of the figure, fails to capture most fine structures and also has large semantic estimation mistakes. Once again, it fails to capture the space between the subject's arm and torso. Moreover, the most noticeable mistake is the large portion of her hair that is missing. Because this hair is blowing in the wind and is therefore slightly transparent and a slightly different colour, it appears that P3M-Net has failed to capture it as part of the final alpha matte estimation. Our design, on the right, does not handle this perfectly, but it retains most of the hair and many of the gaps in between. These results illustrate the potential of our design and offer a preview of how

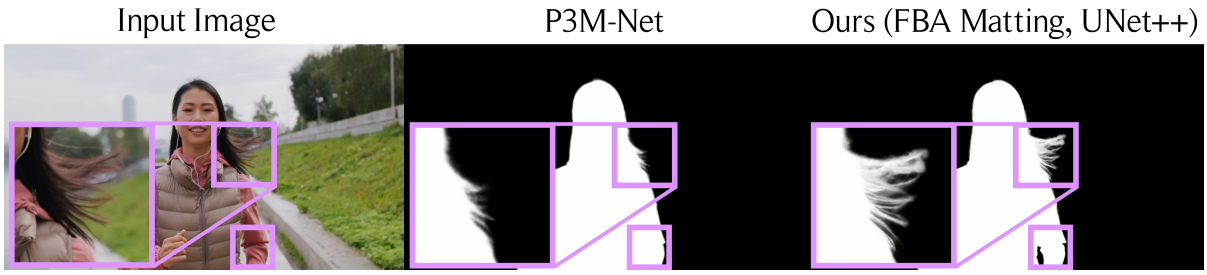

Figure 4.9: Visual performance on video of woman jogging: medium shot, frame 2.

well our design adapts to real-world images and videos as compared to the state-of-the-art P3M-Net, which outperformed ours in numerous quantitative results.

Lastly, we take a look at a foreground estimation from each network in Figure 4.10. Starting with ours on the right, we see a realistic semantic segmentation of the subject's body but, as expected, some issues with hollow areas between the hair. On the right side of her face, some of the hollow areas have been retained but some of the big ones on the other side remain. This would entail seemingly random streaks of white and a blotch of blue being preserved should this foreground extraction be used as is. Going to the P3M-Net foreground extraction on the left, we see the issue of losing the hollow area between the left and arm again. In the hair, it performed similarly to ours, but left out a few individual strands of hair around the outside. Additionally, none of the hollow areas in between her hair are maintained in this alpha matte estimation. Again, in the context of image matting, this video is very complicated and although there were some issues, overall, both methods perform admirably.

Although not perfect, the results from both networks show us the promise of trimap-free matting. Creating a cohesive, consistent string of trimaps for these videos — particularly, the second video — by hand would be an immensely painstaking task. These methods both showcase the true potential of trimap-free matting. Moreover, these alpha matte estimations and foreground extractions are a testament to how far trimap-free methods have come over the past few years.

Figure 4.10: Visual performance on video of woman jogging: medium shot, foreground extracted.

4.4 Discussion

As an exploratory work into this modular approach to image matting, both quantitatively and visually, the alpha matte estimations from our design shows the potential of trimap-free methods. Although there is always an expected decrease in accuracy compared to trimap-based matting, our design shows great promise in taking the next step towards a viable, convenient approach to previous trimap-based image matting works.

Once again, it should be noted that while most of the models shown were trained on P3M-10K, the MODNet pretrained model we used was trained on a different dataset. This may explain the poor results it produced across our tests. In addition, some of our results were retrieved from [15] as we did not have access to the models to train them on P3M-10K. As such, the results for SHM on P3M-500-NP on the trimap region may be inaccurate. The results across all metrics are identical to the performance on the

P3M-500-P benchmark, which is highly unlikely. However, we chose to include them as is considering this was how they were presented in the original paper [15].

We opted to use the FBA Matting variant of our network architecture with the UNet++ semantic segmentation network, and our trimap generation network in our detailed comparison with the state-of-the-art P3M-Net network. This FBA Matting variant had the best performance from all of our networks, as shown in Tables 4.2 and 4.1.

Starting with the trimap region results, our method achieves competitive performance with the top trimap-free methods on both benchmarks. Only P3M-Net beats our top-performing variant in SAD on the P3M-500-P benchmark. Granted, our design is slightly outperformed on some of the other metrics compared to GFM and SHM as well. That being said, our method was consistently performing near the top for both benchmarks on the trimap regions. This implies that our design worked fairly well, but there may have been some errors during the semantic segmentation stage. We came to this insight as our matting stage uses tried-and-true trimap-based networks. These networks work well with precise trimaps, suggesting that deviations from the baseline, explained in Chapter 4.5.1, are due to mistakes in our generated trimaps, which use our estimated semantic segmentations as a foundation. An additional insight relates to the variation in our design's results based on which networks we use; an insight we discuss in further detail later on.

In terms of the results across the whole image, all of our methods score similarly in all three metrics within both benchmarks. The relative performance across the different matting methods and design changes remain consistent — the UNet++ versions of each model outperform the UNet versions and the FBA-based variants beat out the IndexNet variants, which, in turn, beat the DIM-based variants. This implies that the matting stage works as expected within each design version. However, as we noted in the trimap region analysis, this also indicates that the main issue in our design lies with the earlier

segmentation stage. As we predicted, any large error from the semantic segmentation network will be carried over into the subsequent trimap generation and matting networks. This can be seen in the example outlier images in Figure 4.12.

The preservation of these errors in subsequent networks has been a valid concern with decomposed image matting, in which the semantic segmentation stage and matting stage are handled by two completely different networks. One of the main limitations of a decomposed approach to image matting is that mistakes in the earlier stages carry on to the later stages. As such, the outliers from our results were all due to this inherent drawback. In cases where our segmentation network had small mistakes, the trimap generation step could usually mitigate those errors by eroding and dilating the boundary regions. In the scenarios where the segmentation network completely fails, those mistakes essentially become irreparable. Some examples of these cases can be seen in Figure 4.12, while a larger selection can be seen in Figures B.1, B.2, and B.3 in the appendix. This was only the case in a minority of our results. A few estimated alpha mattes with very large SAD values, some magnitudes of order greater, may have caused these potentially misleading results. In Chapter 4.5.4, we detail the process of removing potential outliers for both our top-performing design and the state-of-the-art P3M-Net method. Removing the outliers resulted in much closer metrics between the two models, as seen in Tables 4.5 and 4.6. After removing the outliers for both models, the results become more representative of the similar visual performance of the two methods.

The discrepancy between the trimap-based networks with ground truth trimaps as compared to our design in Table 4.3 is due to the gap that remains between ground truth trimaps and our generated ones. Our long-term goal is to find ways to bridge this gap in order to bring trimap-based matting accuracy to trimap-free methods. It appears that at this stage in the ever-changing landscape of image matting, particularly with respect to a modular design, that tightening this gap will rely on further advancements in semantic segmentation. Until these improvements arrive, we can sidestep this road-

block by introducing innovative applications, such as the ideas presented in Chapter 5.1. While not ideal trimap-free alternatives, these applications can be valid means to achieve trimap-based performance, without having to create a trimap from scratch.

Based on these results, our design remained competitive with the top trimap-free methods in both benchmarks. Although it never surpassed the state-of-the-art P3M-Net, it remained close throughout, even with a few large outliers. Our visual analysis showed how well our design performed on real-world images and videos; outperforming P3M-Net many times too. This may be indicative of the domain gap that remains between benchmark datasets and real-world data.

One of the most fascinating insights from these results is the potential for innovation in matting due to the flexibility of our modular design. Within the trimap unknown region across both benchmarks, there is a sizeable gap between the SAD of the lowest-performing variant and the top-performing variant of our design. This gap is even greater when analyzing the whole image results for both networks. Replacing networks at the various stages of our architecture resulted in noticeable differences, both in the quantitative results and our visual analysis.

Undoubtedly, the most exciting part about these findings is the notion of how modularity will impact the future of image matting. The idea of further innovations in the matting process being swapped into current models can be realized with modular designs. While the top-performing trimap-free networks today may become obsolete tomorrow, our modular design will allow for compatible, improved networks to substitute the current models at each stage of our design.

Ultimately, these experiments have demonstrated the potential and feasibility of using a modular approach to trimap-free image matting. With ample room for exploration and improvement, modular architectures seem to have a compelling future in image matting.

4.5 Ablative Studies

4.5.1 Comparing Our Design with Baseline Matting Methods

When analyzing the performance of our networks, we wanted to compare our results to those of other trimap-free methods, but also, with those of a baseline. In order to assess the viability of this work with respect to the larger picture of image matting, these baselines would have to consist of popular trimap-based image matting techniques. In this case, the baseline was the performance of each of our matting network variants with ground truth trimaps as the auxiliary inputs, as opposed to our generated trimaps. We passed these ground truth trimaps to the standalone, pretrained DIM, IndexNet, and FBA Matting networks. The results on these trimap-based networks can be seen in Table 4.3.

Next, we take these baseline trimap-based results with ground truth trimaps and compare them to our corresponding trimap-free designs with generated trimaps, as shown in Table 4.4. As expected, the original, proven trimap-based versions beat out our corresponding network variants as they rely on ground truth trimap values, which were inferred from ground truth matte values. On the other hand, our models generated their own trimaps using our architecture described in Chapter 3. Despite the fact that our model was outperformed by the trimap-based networks, these results are quite exciting. We have shown that even though we remove a crucial secondary input from the image matting pipeline, we are able to achieve results that are in the same neighbourhood as the original trimap-based networks. Although there is ample room for improvement, these results indicate that a modular design may be a more feasible alternative path to using trimap-based networks. An approach that delivers reasonable alpha mattes without having to input a trimap.

Neither the goal nor the expectation was to beat the trimap-based scores, but rather, to minimize the difference between them. The only difference between the trimap-based

Table 4.3: Performance of Baseline Trimap-Based Matting Networks on P3M-500-P Benchmark

Network	SAD ↓	MAD ↓	MSE ↓
DIM with Ground Truth Trimaps	8.45	0.060	0.022
IndexNet with Ground Truth Trimaps	7.92	0.056	0.019
FBA Matting with Ground Truth Trimaps	**5.76**	**0.041**	**0.012**

Table 4.4: Comparing Our Best-Performing Design Variants with Their Corresponding Trimap-Based Networks on P3M-500-P (Trimap Unknown Region)

Network	SAD ↓	MAD ↓	MSE ↓
DIM (Ours, UNet++, Fine-Tuned)	9.58	0.065	0.035
DIM with Ground Truth Trimaps	8.45	0.060	0.022
IndexNet (Ours, UNet++)	9.44	0.065	0.032
IndexNet with Ground Truth Trimaps	7.92	0.056	0.019
FBA Matting (Ours, UNet++)	8.55	0.058	0.029
FBA Matting with Ground Truth Trimaps	**5.76**	**0.041**	**0.012**

methods and ours was that theirs used ground truth trimaps and ours generated them using our architecture. Accordingly, the more accurate our generated trimaps become, the closer we are to the performance of the original trimap-based networks.

4.5.2 Impact of Fine-Tuning Entire Network Architecture

One of the clear advantages of using a trimap generation network, as opposed to simply using a computer vision library such as OpenCV, to generate our trimaps is that our network is differentiable. This property of neural networks allows us to not only train the trimap generation network individually but also to fine-tune our entire modular design — from semantic segmentation network to trimap generation network to matting network — end-to-end.

Considering the fact that the pretrained, trimap-based networks we use have been proven to work effectively, as seen in their own papers [1, 2, 19] and in our trimap-based evaluations in Chapter 4.5.1, we opt for freezing the matting stage, in this case, the pretrained DIM network. Our focus is on the two prior stages, so we fine-tune them but use the entire network architecture to do so. We lower the learning rate to 1×10^{-6} and use the P3M-10K dataset.

For this ablative study, we compare the result of this fine-tuned variant of the network to the network without it. As shown in Tables 4.1 and 4.2, we see how this affects the results on the trimap region and the whole image, respectively. We see improvements across most of the metrics in the trimap region and drops in the SAD metric for the whole image. These improvements hint at the benefits of using a fully differentiable design. We would expect similar improvements by performing this type of fine-tuning with our other variants as well. In a future study, it would be intriguing to assess how changing factors such as the learning rate or chosen dataset for this step would impact the final performance and results of these variants.

4.5.3 Choice of Semantic Segmentation Network

We wanted to see how the choice of a different network at the segmentation stage can affect the entire matting process and to see if a larger, more powerful segmentation network would lead to better alpha matte estimations at the end of our pipeline. Because many of the current trimap-based networks perform so well already, this increases the importance of an effective semantic segmentation network, and subsequently, the trimap generation network. We have seen that the matting stage works very well on ground truth trimaps, so we hope to leverage advancements in semantic segmentation to create precise trimap estimations.

We start off with a UNet model with a EfficientNet-B4 backbone pretrained on ImageNet weights. We replace this model with a UNet++ variant with the same pretrained

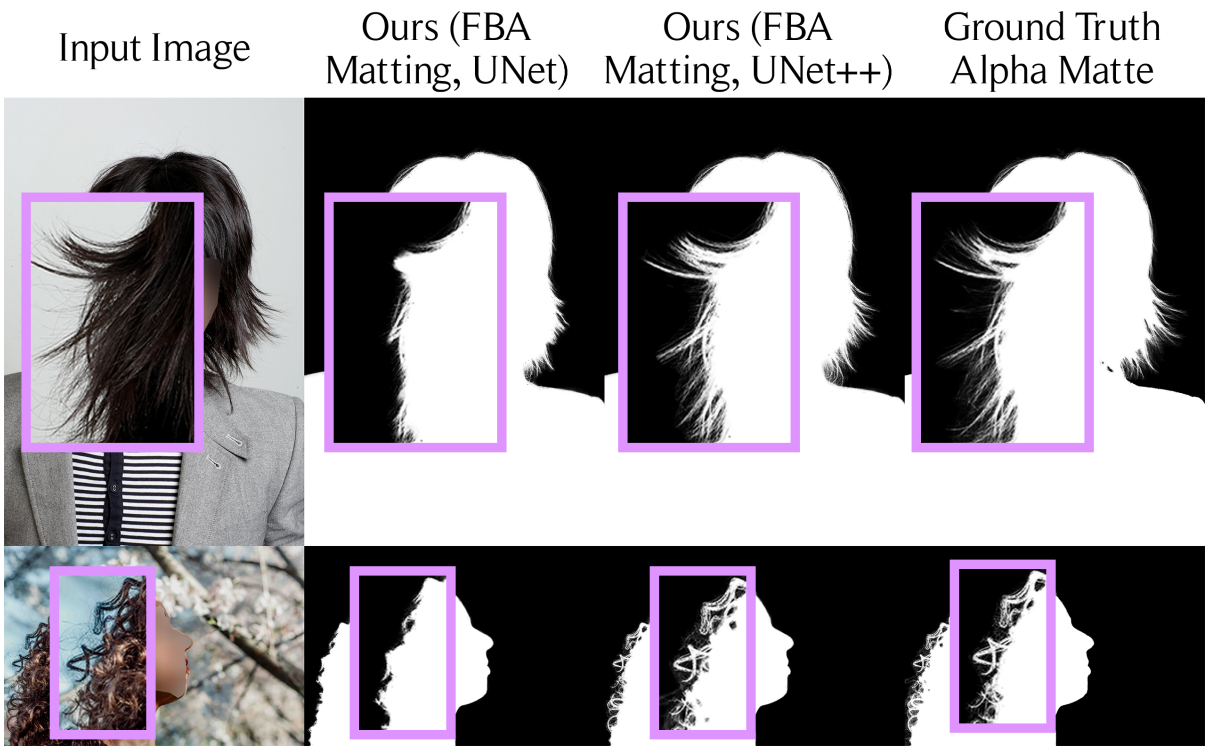

Figure 4.11: Comparing the impact of the semantic segmentation network on the estimated alpha mattes from our design.

backbone and alternate between the two models for our evaluation. We did so with the intent of assessing the importance in semantic segmentation network choice in our model. After all, the freedom to choose and replace the network at each stage is the main draw to a modular approach. When opting for the UNet++ variant, we see improvements across the board in both trimap regions and whole images for every matting network, as shown in Tables 4.1 and 4.2.

Visually too, there were noticeable improvements between the estimated alpha mattes using the UNet++ variants compared to the identically configured UNet ones, as shown in Figure 4.11. The most drastic improvements were in video, where the inconsistency between frames with the UNet variants was jarring compared to the natural consistency and smoothness in the case of the UNet++ variants.

In the case of this exploratory work, our motivation was to improve the quantitative and visual performance of our design, so the UNet++ variant was the most alluring for

Table 4.5: Effect of Outliers on Whole Image SAD

Benchmark	Network	SAD ↓ (With Outliers)	SAD Threshold	Number of Outliers	SAD ↓ (No Outliers)
P3M-500-P	FBA (Ours)	21.35	28.18	70 (14%)	7.78
	P3M-Net	8.49	15.53	40 (8%)	6.02
P3M-500-NP	FBA (Ours)	24.28	33.61	74 (14.8%)	8.82
	P3M-Net	10.90	20.19	37 (7.4%)	7.18

us. On the other hand, some users may have efficiency or portability in mind. In that case, perhaps a more efficient, lighter network might serve them better. With that being said, although encouraging, the improved results on the UNet++ variants is not the most exciting aspect of this ablative study. But rather, the idea that this modular design allows users to swap out and replace networks at each stage to best suit their needs.

4.5.4 Outlier Removal

While the trimap region results were comparable across both benchmark datasets between our design and the state-of-the-art P3M-Net, there was a large discrepancy between the whole image results. After delving into each method's whole image results, we found that for most test images, the results were quite similar. There were, however, a few large SAD values which may have been skewing the results, which can be seen in Figure 4.13 and Figure 4.14 for the P3M-500-P and P3M-500-NP benchmark datasets, respectively. This was also the case in the trimap region SAD across both methods, though not as drastic as the whole image SAD, as shown in Figure 4.15 and Figure 4.16 for the P3M-500-P and P3M-500-NP benchmark datasets, accordingly.

There was a common issue with these large SAD images: failures in clearly defining the foreground subject, as shown in Figure 4.12. In the examples shown in this figure, we see large chunks of the foreground subjects not included or areas in the background included in the estimated alpha matte. This implies that these failures stem from the semantic segmentation stage. This is discussed in further detail below.

To further analyze the spread of the evaluation data, we calculated the interquartile

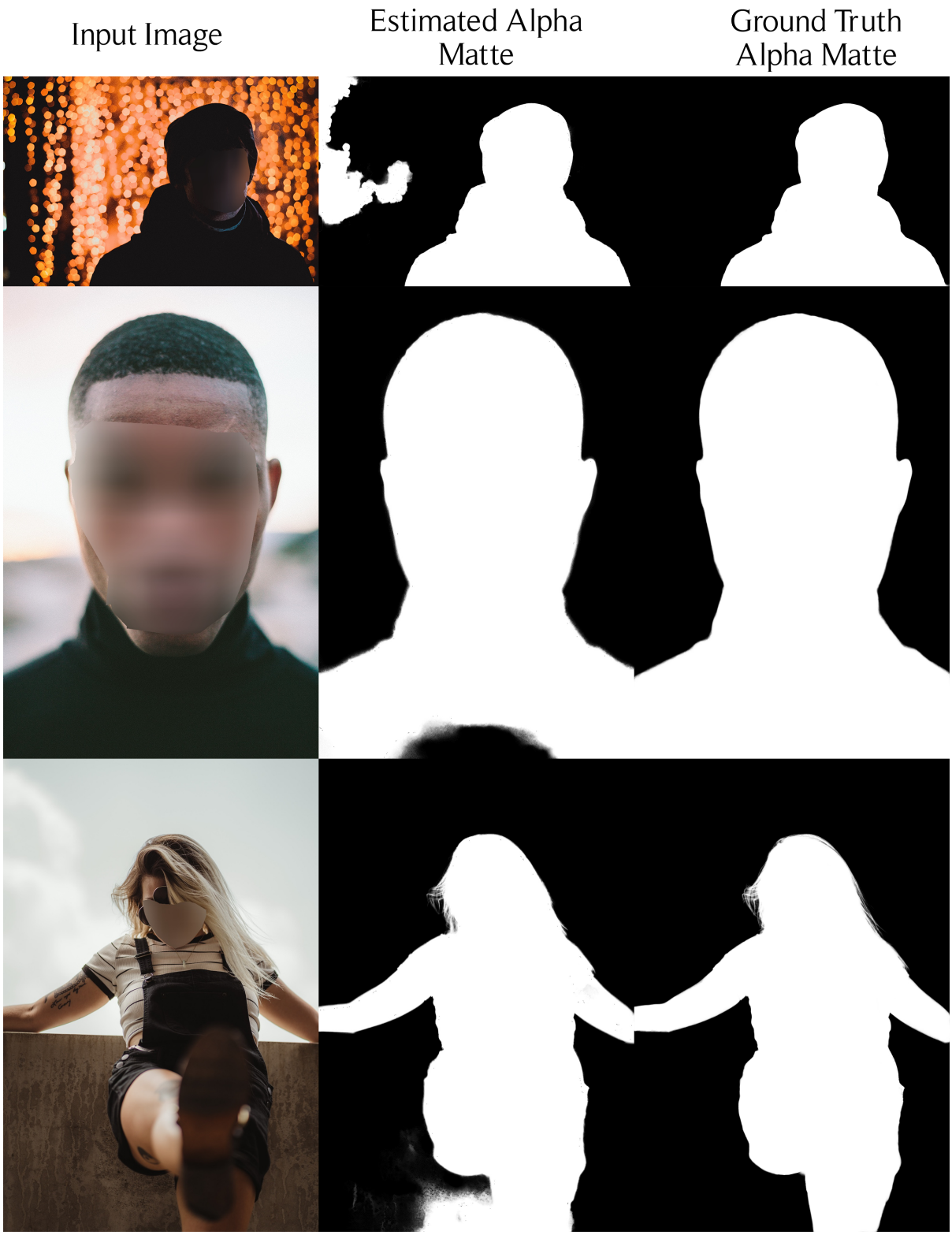

Figure 4.12: A few example images that were considered outliers based on IQR. Left: input image, centre: our estimated alpha matte, right: ground truth alpha matte.

Table 4.6: Effect of Outliers on Trimap Unknown Region SAD

Benchmark	Network	SAD ↓ (With Outliers)	SAD Threshold	Number of Outliers	SAD ↓ (No Outliers)
P3M-500-P	FBA (Ours)	8.55	19.73	27 (5.4%)	7.05
	P3M-Net	6.78	14.33	28 (5.6%)	5.78
P3M-500-NP	FBA (Ours)	10.31	25.42	34 (6.8%)	8.07
	P3M-Net	7.54	16.90	26 (5.2%)	6.60

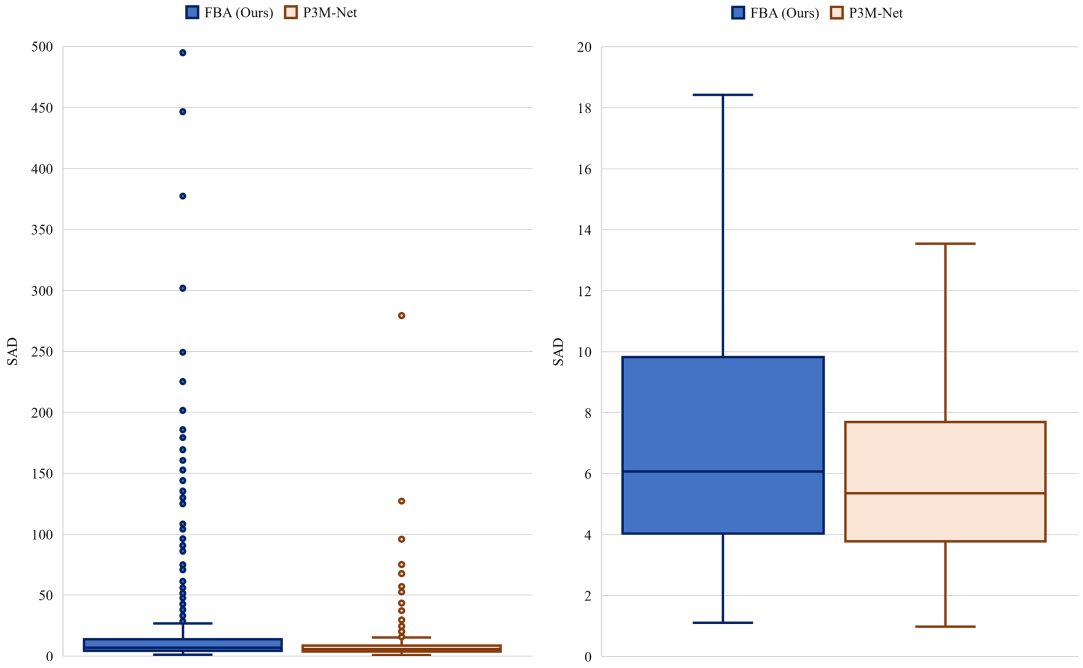

Figure 4.13: Box and whisker plot of SAD (whole image) on P3M-500-P benchmark, depicting the spread of the SAD before (left) and after (right) outlier removal.

range across the whole image and trimap region results on both datasets. The upper threshold for outliers equation is $Upper\ Outlier\ Threshold = Q_3 + 1.5 \times (Q_3 - Q_1)$. Here, Q_1 is the 1st quartile — the lowest 25% of data lies below this point — and Q_3 is the 3rd quartile — the lowest 75% of data lies below here. Based on the spread of the data, the upper outlier threshold marks the upper cutoff for outliers; any data point lying above is considered an outlier. Table 4.5 shows that for whole image SAD, approximately 14-15% of the images fall into this category in our design, whereas 7-8% do so for P3M-Net. For trimap region SAD, this applies to approximately 5-7% of the estimated alpha mattes from our method compared to around 5-6% for P3M-Net, as seen in Table 4.6.

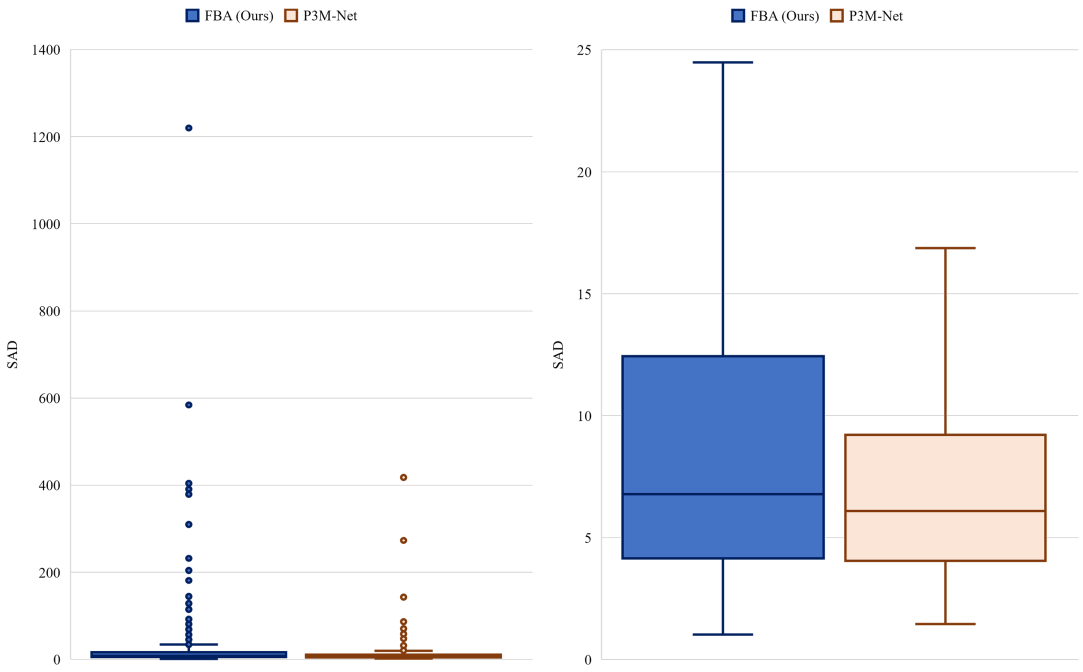

Figure 4.14: Box and whisker plot of SAD (whole image) on P3M-500-NP benchmark, depicting the spread of the SAD before (left) and after (right) outlier removal.

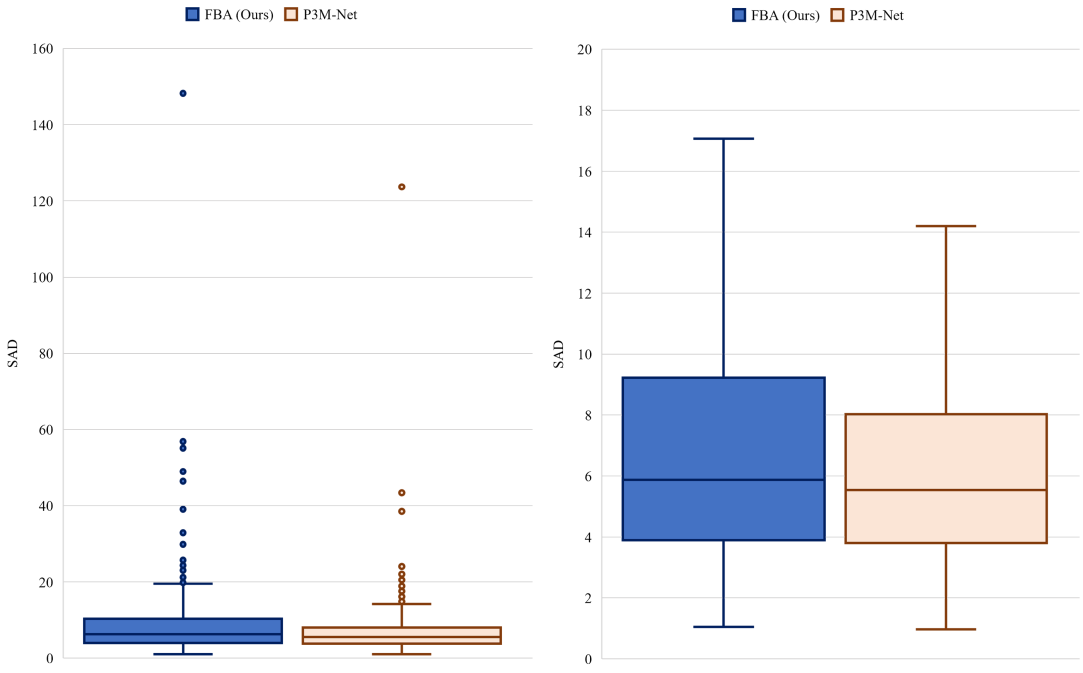

Figure 4.15: Box and whisker plot of SAD (trimap unknown region) on P3M-500-P benchmark, depicting the spread of the SAD before (left) and after (right) outlier removal.

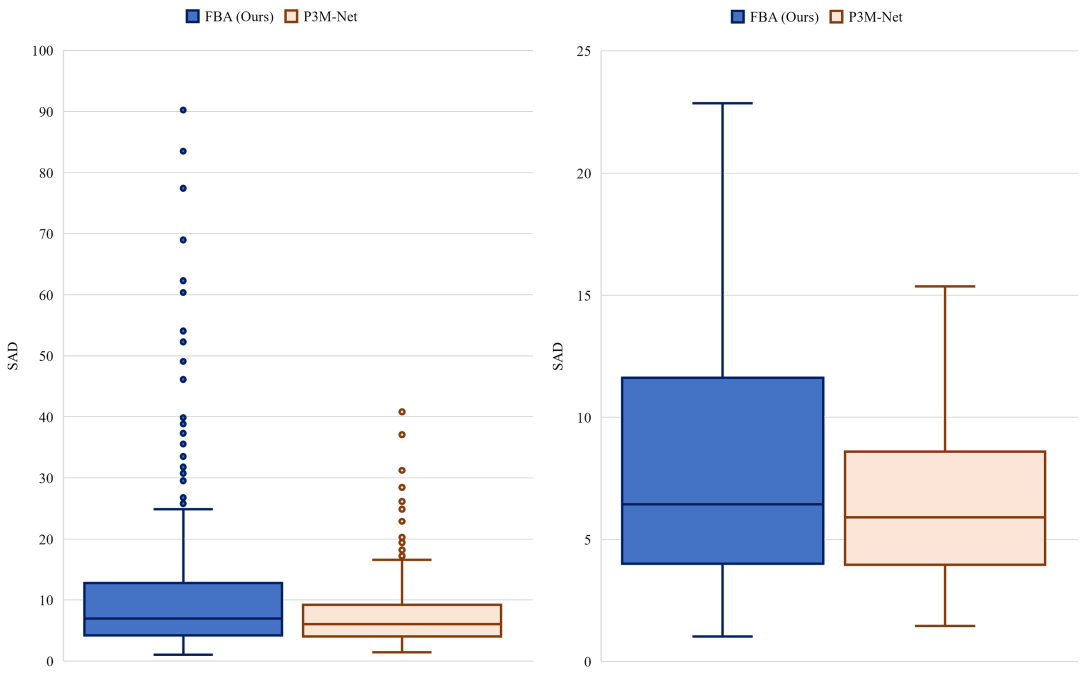

Figure 4.16: Box and whisker plot of SAD (trimap unknown region) on P3M-500-NP benchmark, depicting the spread of the SAD before (left) and after (right) outlier removal.

Upon removal of these outliers, the whole image SAD metric drops considerably for both P3M-500-P and P3M-500-NP results on our design, while there is a small drop for P3M-Net, as seen in Table 4.5. There are also much smaller drops in the SAD metric across the trimap region, when accounting for outlier removal based on trimap region SAD, shown in Table 4.6. The large whole image SAD data points are likely due to errors in the segmentation step which were carried over by the trimap step, and subsequently, into the matting step. Unfortunately, these occasional large errors are commonplace in trimap-free methods as foreground regions are estimated without additional human input. There are even some large outliers in the state-of-the-art P3M-Net results, albeit not as drastic as some of the large errors in our method's results. This is depicted visually in the Figures mentioned earlier, Figures 4.13 and 4.14 for the whole image SAD and Figures 4.15 and 4.16 for the trimap region SAD.

Updated results following outlier removal for all metrics can be found in Table A.1

and Table A.2 in the appendix for outliers in the whole image and in the trimap region, respectively. A few example images that were considered outliers can be seen in Figure 4.12 and more examples can be seen in Figures B.1, B.2, and B.3 in the appendix.

After the outliers were removed for both models, the results become much closer and show a similar spread in the evaluation data. Based on the number and percentage of outliers for each evaluation, our method provides desirable results approximately 85% of the time with regards to whole image SAD, whereas P3M-Net does so around 92% of the time. While we do not discard these data points completely, as they provide valuable insight into the shortcomings of our current design, delving into the spread of the data and the presence of these outliers has shown that in most cases, our design performs very similarly to the state-of-the-art P3M-Net on these evaluation benchmarks.

Chapter 5

Conclusions and Future Directions

5.1 Applications

As previously discussed, there are numerous applications of image matting, from daily tasks such as background blur in a video call to more complicated endeavours such as green screen in a movie. While the same applications apply to our work too, there are a few specific use cases where our design can truly shine.

5.1.1 Trimap Refinement Application

The need for trimap-free approaches in image matting is clear: while trimap-based methods showcase the full potential of image matting, end users are more likely to adopt this technology if the only required input was a single image. This would include avoiding any additional inputs, such as trimaps and background images. Perhaps allowing users to refine a generated trimap before sending it to the final matting stage of the network would be a promising alternative as well.

As we showed in Chapter 4.5.4, our design may produce occasional failure cases. These are mostly due to failures in the semantic segmentation output and, subsequently, trimap generation failures. This is where we can leverage one benefits of modular design

— an output at each stage — to eliminate, or at least mitigate, some of these failure cases.

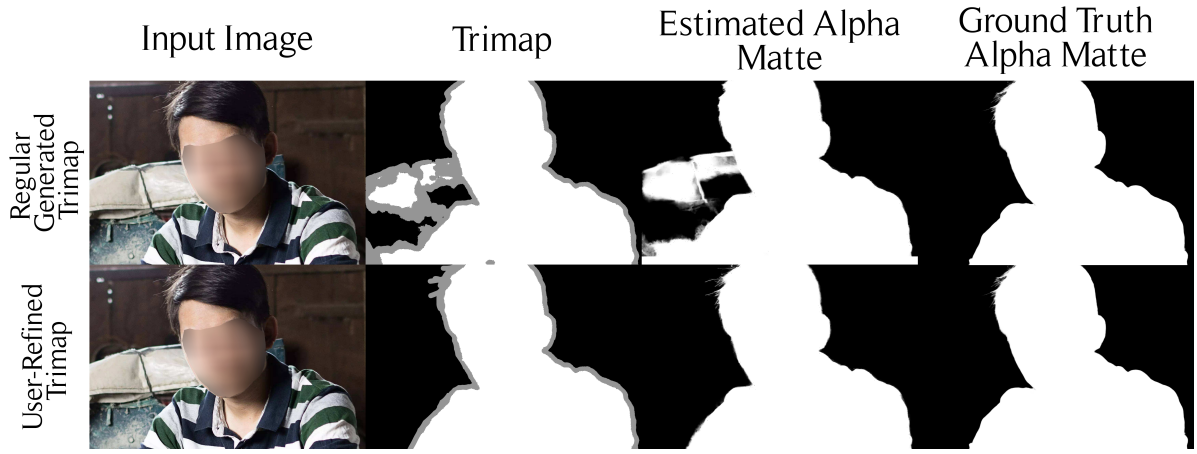

Figure 5.1: An example of a trimap refinement application. Left: input image. Top row, left: failure case trimap from our design. Top row, right: output alpha matte estimation using failure case trimap. Bottom row, left: user-refined failure case trimap (refined in Adobe Photoshop [43]). Bottom row, right: output alpha matte estimation using modified trimap. Right: ground truth alpha matte.

Two major use cases for this modular trimap-free design would be a social media application which allows you to modify or replace the background in an image and a keying tool for video editing, which does not require a green screen. In both scenarios, an intermediate stage which allows for user input to modify and refine the trimap can help create precise alpha matte estimations. Imagine the input image from Figure 5.1 is a picture we wish to upload on social media or a frame from a video we are editing. If we use our current design, we are stuck with the erroneous trimap and subsequent output alpha matte. However, using our modular design, we can create a theoretical application which shows the user the initial generated trimap and ask for permission to continue with matting or allow them to make tweaks to it.

In the failure case depicted in the figure, we altered the trimap to adjust for known foreground and background regions, and modified the unknown area to the areas where we wanted the matting network to operate. This resulted in a much better alpha matte, which is actually more precise than the ground truth alpha matte in the hair regions.

Ironically, this output would be penalized in quantitative results for retaining every strand of hair as they are not included in the ground truth matte. Nevertheless, a mobile or web application, or even a plugin for professional software, where users can use our design and modify the generated trimaps can have tremendous implications for both personal and professional use.

Treating our trimap generation pipeline — semantic segmentation and trimap generation — as a preliminary trimap estimation can lead to better alpha matte estimations. Throughout this thesis, we have argued that creating a trimap is a daunting task for those unfamiliar with the task. Although this may be the case, modifying an estimation may be a much more appealing alternative than making one from scratch. We believe that general users would prefer having a starting point over having to create a trimap from the ground up, as it is a much less daunting and less tedious task.

We also believe that this would be the preferred option for professionals as it is a much easier substitute than masking each frame individually and worrying about changes in fine structures such as hair. In this case, they ensure the trimap fits in each frame, make the necessary adjustments, and allow the matting network to handle the fine details.

It should be noted that while P3M-Net may outperform our network on certain images, this interactive style of application can only be utilized with a modular design. Until there are further improvements in human semantic segmentation, this may be the most alluring application of our design and the most feasible path to bridging the gap between trimap-free and trimap-based image matting.

5.1.2 Image Matting Library

The most simple application of our approach to image matting would be one where a user can select from a list of networks for each stage in order to create an ideal image matting pipeline. They would be able to select an input image or video, followed by a semantic segmentation network of choice, their preferred trimap generation network, and

finally, a matting network of their choosing.

This application would have numerous use cases, including for professional editors, social media use, and perhaps most interestingly, for researchers. In the earlier two examples, it could be presented as a web or mobile application with an easy-to-follow user interface.

In the case of research, this may be presented as an image matting library for a popular deep learning framework. Image matting researchers would be able to pick and choose from preloaded, pretrained models and tinker with the settings until they find a solution that best fits their needs, similar to Segmentation Models [38] for semantic segmentation. Our modular design would allow this image matting playground to reach its full potential; allowing researchers to add, remove, and tune networks as they see fit.

5.2 Conclusion

This thesis serves as a detailed, exploratory study into modular image matting with the goal of assessing its viability as a trimap-free approach to proven trimap-based methods. We demonstrated the importance of an accurate semantic segmentation stage, the freedom a reliable trimap generation network brings, and the power of a proven trimap-based matting network. Our modular design and its proposed applications demonstrate the potential of our approach in bridging the gap between trimap-based and trimap-free works. Our proposed trimap refinement application offers insight into how the interactive element of trimap-based approaches can be introduced to trimap-free works.

While the current implementation is far from perfect at this time, this research provides numerous insights to help move modular matting forward. This may include further refinements to the design of our architecture, utilizing further breakthroughs in segmentation, trimap generation, and matting, or creative applications that find new ways to leverage trimap-free matting.

These aforementioned applications offer a small glimpse into the seemingly limitless use cases of image matting. Some applications are specific to our design, others to trimap-free methods or portrait matting in general. We believe that these applications, particularly the trimap refinement application, offer an immediate solution to convenient, straightforward matting. Allowing general users to achieve excellent results while avoiding the hassle of designing a trimap from scratch. While a large gap remains between trimap-free and trimap-based matting, applications like this can help bridge that gap until further improvements in semantic segmentation, and even in deep learning, arrive.

Other trimap-free approaches have shown promising results in the past. The major drawback in comparison to a modular architecture is that these individual methods may potentially become stagnant. Whereas, when it comes to a modular architecture, continued improvement in both segmentation and trimap networks can progressively eliminate the difference between generated and ground truth trimaps. Likewise, there may be new, state-of-the-art trimap-based approaches that are introduced in the future. As these advancements and developments keep rolling in, a modular architecture allows us to substitute and exchange the different stages with compatible alternatives until we have found an arrangement that best suits us; be it lightweight models for faster inference or larger, more robust models when accuracy is of the utmost importance. Although there is ample room for improvement, with all things considered, modular designs have established themselves as an exciting path forward. But for now, the user must determine whether or not the ease and convenience of a trimap-free method outweighs the reliability and accuracy of a trimap-based approach.

At the onset of this research, we sought out to explore a method to bridge the gap between the performance and interactivity of trimap-based networks and the ease of use of trimap-free methods. After a comprehensive quantitative and visual analysis, we believe that our proposed modular matting architecture is a positive step in that direction.

5.3 Current Challenges and Future Directions

Because this work acts as an early exploration into the space of automated trimap generation and modular image matting, there remains much room for improvement. Firstly, there are some failure cases due to occasional poor segmentations in the first stage. Errors in these segmentations are carried forward into the trimap generation stage, and subsequently, to the matting stage. Secondly, because we have used three networks in this initial design, this current implementation is large and computationally expensive.

One of the main benefits of this modular design is that individual stages can be easily swapped in and out of the pipeline to best suit the user. This design can be leveraged to tackle both of these limitations. We have shown that improving the segmentation stage results in better trimaps, and consequently, better alpha matte estimations. A user can choose to replace the current segmentation network in the first stage with one that better suits their foreground subjects or replace it with a more efficient segmentation network, potentially at the cost of accuracy. The same idea applies to the trimap generation network. If a user decided that another trimap generation network suits their needs — perhaps one that is more efficient or a model that focuses on coarse trimap generation and trimap refinement as sub-objectives that can be solved in one network — they have the freedom to make that change. Again, the choice to swap networks in and out depends on the needs and constraints of the end user.

Even with the aforementioned limitations of this research in its current state, this work demonstrates the promise and potential of this approach to image matting. As we mentioned earlier, the errors of the earlier stages of this architecture are compounded by subsequent stages; a poor segmentation leads to a poor trimap and a poor trimap leads to a poor alpha matte estimation. Conversely, improvements in both the semantic segmentation stage and trimap generation stage can lead to further improvements in the alpha matte estimations. There is still untapped potential in existing trimap-based models with regards to being used in the last stage of this architecture. More accurate

and efficient semantic segmentation and trimap generation stages can lead to better alpha matte estimations; alpha mattes more representative of the ground truth. Mattes which may eventually be created by these trimap-based networks.

Continuing with the idea of efficiency, there is much room for improvement in that aspect with this design. Naturally, a three-network design is computationally expensive and resource intensive. Utilizing efficient networks in each stage can improve the efficiency of the entire architecture, but may come at the cost of accuracy. If the user determines that an efficient network is worth any potential any trade-off in accuracy, the modular design grants the user the freedom to make that change.

Bibliography

[1] Ning Xu, Brian Price, Scott Cohen, and Thomas Huang. Deep Image Matting. In *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 311–320, Honolulu, HI, July 2017. IEEE.

[2] Marco Forte and François Pitié. F, B, Alpha Matting. *arXiv:2003.07711 [cs]*, March 2020. arXiv: 2003.07711 version: 1.

[3] Jizhizi Li, Jing Zhang, Stephen J. Maybank, and Dacheng Tao. Bridging Composite and Real: Towards End-to-end Deep Image Matting, October 2021. arXiv:2010.16188 [cs, eess].

[4] Yu Qiao, Yuhao Liu, Xin Yang, Dongsheng Zhou, Mingliang Xu, Qiang Zhang, and Xiaopeng Wei. Attention-Guided Hierarchical Structure Aggregation for Image Matting. In *2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 13673–13682, Seattle, WA, USA, June 2020. IEEE.

[5] Yunke Zhang, Lixue Gong, Lubin Fan, Peiran Ren, Qixing Huang, Hujun Bao, and Weiwei Xu. A Late Fusion CNN for Digital Matting. In *2019 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 7461–7470, Long Beach, CA, USA, June 2019. IEEE.

[6] Quan Chen, Tiezheng Ge, Yanyu Xu, Zhiqiang Zhang, Xinxin Yang, and Kun Gai. Semantic Human Matting. *arXiv:1809.01354 [cs, stat]*, September 2018. arXiv: 1809.01354 version: 2.

[7] Arie Berman, Arpag Dadourian, and Paul Vlahos. Method for Removing from an Image the Background Surrounding a Selected Object, October 2000.

[8] Yung-Yu Chuang, B. Curless, D.H. Salesin, and R. Szeliski. A Bayesian approach to digital matting. In *Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001*, volume 2, pages II–264–II–271, Kauai, HI, USA, 2001. IEEE Comput. Soc.

[9] Jian Sun, Jiaya Jia, Chi-Keung Tang, and Heung-Yeung Shum. Poisson matting. *ACM Transactions on Graphics*, 23(3):315–321, August 2004.

[10] A. Levin, D. Lischinski, and Y. Weiss. A Closed-Form Solution to Natural Image Matting. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 30(2):228–242, February 2008.

[11] Yağız Aksoy, Tunç Ozan Aydın, and Marc Pollefeys. Information-Flow Matting. *arXiv:1707.05055 [cs]*, April 2019. arXiv: 1707.05055 version: 2.

[12] Christoph Rhemann, Carsten Rother, Jue Wang, Margrit Gelautz, Pushmeet Kohli, and Pamela Rott. Alpha matting evaluation website, 2009.

[13] Shanchuan Lin, Andrey Ryabtsev, Soumyadip Sengupta, Brian Curless, Steve Seitz, and Ira Kemelmacher-Shlizerman. Real-Time High-Resolution Background Matting. *arXiv:2012.07810 [cs]*, December 2020. arXiv: 2012.07810 version: 1.

[14] Zhanghan Ke, Kaican Li, Yurou Zhou, Qiuhua Wu, Xiangyu Mao, Qiong Yan, and Rynson W. H. Lau. Is a Green Screen Really Necessary for Real-Time Portrait Matting? *arXiv:2011.11961 [cs]*, November 2020. arXiv: 2011.11961 version: 2.

[15] Jizhizi Li, Sihan Ma, Jing Zhang, and Dacheng Tao. Privacy-Preserving Portrait Matting. *arXiv:2104.14222 [cs]*, July 2021. arXiv: 2104.14222.

[16] Jue Wang and Michael F. Cohen. Optimized color sampling for robust matting. In *2007 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8, 2007.

[17] Eduardo S. L. Gastal and Manuel M. Oliveira. Shared sampling for real-time alpha matting. *Computer Graphics Forum*, 29(2):575–584, May 2010. Proceedings of Eurographics.

[18] Kaiming He, Christoph Rhemann, Carsten Rother, Xiaoou Tang, and Jian Sun. A global sampling method for alpha matting. In *CVPR 2011*, pages 2049–2056, 2011.

[19] Hao Lu, Yutong Dai, Chunhua Shen, and Songcen Xu. Index Network. *arXiv:1908.09895 [cs]*, April 2020. arXiv: 1908.09895 version: 2.

[20] GyuTae Park, SungJoon Son, JaeYoung Yoo, SeHo Kim, and Nojun Kwak. MatteFormer: Transformer-Based Image Matting via Prior-Tokens. Technical Report arXiv:2203.15662, arXiv, March 2022. arXiv:2203.15662 [cs] type: article.

[21] Sebastian Lutz, Konstantinos Amplianitis, and Aljosa Smolic. AlphaGAN: Generative adversarial networks for natural image matting. *arXiv:1807.10088 [cs]*, July 2018. arXiv: 1807.10088.

[22] Shaofan Cai, Xiaoshuai Zhang, Haoqiang Fan, Haibin Huang, Jiangyu Liu, Jiaming Liu, Jiaying Liu, Jue Wang, and Jian Sun. Disentangled Image Matting. *arXiv:1909.04686 [cs]*, September 2019. arXiv: 1909.04686.

[23] Kaiming He, Xiangyu Zhang, Shaoqing Ren, and Jian Sun. Deep Residual Learning for Image Recognition, December 2015. arXiv:1512.03385 [cs].

[24] Mark Sandler, Andrew Howard, Menglong Zhu, Andrey Zhmoginov, and Liang-Chieh Chen. MobileNetV2: Inverted Residuals and Linear Bottlenecks. In *2018*

IEEE/CVF Conference on Computer Vision and Pattern Recognition, pages 4510–4520, Salt Lake City, UT, June 2018. IEEE.

[25] Xiaoyong Shen, Xin Tao, Hongyun Gao, Chao Zhou, and Jiaya Jia. Deep Automatic Portrait Matting. In Bastian Leibe, Jiri Matas, Nicu Sebe, and Max Welling, editors, *Computer Vision – ECCV 2016*, volume 9905, pages 92–107. Springer International Publishing, Cham, 2016.

[26] Rahul Deora, Rishab Sharma, and Dinesh Samuel Sathia Raj. Salient Image Matting. *arXiv:2103.12337 [cs]*, March 2021. arXiv: 2103.12337.

[27] Jinlin Liu, Yuan Yao, Wendi Hou, Miaomiao Cui, Xuansong Xie, Changshui Zhang, and Xian-sheng Hua. Boosting Semantic Human Matting with Coarse Annotations. *arXiv:2004.04955 [cs, eess]*, April 2020. arXiv: 2004.04955.

[28] Soumyadip Sengupta, Vivek Jayaram, Brian Curless, Steve Seitz, and Ira Kemelmacher-Shlizerman. Background Matting: The World is Your Green Screen. *arXiv:2004.00626 [cs]*, April 2020. arXiv: 2004.00626.

[29] Hang Cheng, Shugong Xu, Xiufeng Jiang, and Rongrong Wang. Deep Image Matting with Flexible Guidance Input. *arXiv:2110.10898 [cs]*, October 2021. arXiv: 2110.10898.

[30] Chang-Lin Hsieh and Ming-Sui Lee. Automatic trimap generation for digital image matting. In *2013 Asia-Pacific Signal and Information Processing Association Annual Summit and Conference*, pages 1–5, Kaohsiung, Taiwan, October 2013. IEEE.

[31] Vikas Gupta and Shanmuganathan Raman. Automatic Trimap Generation for Image Matting. Technical Report arXiv:1707.00333, arXiv, July 2017. arXiv:1707.00333 [cs] type: article.

[32] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking Atrous Convolution for Semantic Image Segmentation, December 2017. arXiv:1706.05587 [cs].

[33] Yuhongze Zhou, Liguang Zhou, Tin Lun Lam, and Yangsheng Xu. Semantic-guided Automatic Natural Image Matting with Trimap Generation Network and Light-weight Non-local Attention. Technical Report arXiv:2103.17020, arXiv, September 2021. arXiv:2103.17020 [cs] type: article.

[34] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-Net: Convolutional Networks for Biomedical Image Segmentation, May 2015. arXiv:1505.04597 [cs].

[35] Zongwei Zhou, Md Mahfuzur Rahman Siddiquee, Nima Tajbakhsh, and Jianming Liang. UNet++: A Nested U-Net Architecture for Medical Image Segmentation, July 2018. arXiv:1807.10165 [cs, eess, stat].

[36] Mingxing Tan and Quoc V. Le. EfficientNet: Rethinking Model Scaling for Convolutional Neural Networks, September 2020. arXiv:1905.11946 [cs, stat].

[37] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. ImageNet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, Miami, FL, June 2009. IEEE.

[38] Pavel Iakubovskii. Segmentation Models PyTorch, 2019.

[39] Shruti Jadon. A survey of loss functions for semantic segmentation. In *2020 IEEE Conference on Computational Intelligence in Bioinformatics and Computational Biology (CIBCB)*, pages 1–7, October 2020. arXiv:2006.14822 [cs, eess].

[40] Adam Paszke, Sam Gross, Francisco Massa, Adam Lerer, James Bradbury, Gregory Chanan, Trevor Killeen, Zeming Lin, Natalia Gimelshein, Luca Antiga, Alban Desmaison, Andreas Kopf, Edward Yang, Zachary DeVito, Martin Raison, Alykhan

Tejani, Sasank Chilamkurthy, Benoit Steiner, Lu Fang, Junjie Bai, and Soumith Chintala. Pytorch: An imperative style, high-performance deep learning library. In H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett, editors, *Advances in Neural Information Processing Systems 32*, pages 8024–8035. Curran Associates, Inc., 2019.

[41] G. Bradski. The OpenCV Library. *Dr. Dobb's Journal of Software Tools*, 2000.

[42] Pexels.com.

[43] Adobe Inc. Adobe Photoshop.

Appendix A

Outlier Removal: Full Metrics

The following tables show the impact of outliers on both the whole image and trimap unknown region metrics across the P3M-500-P and P3M-500-NP benchmarks for both our design and P3M-Net.

Table A.1: Effect of Outliers on Whole Image Metrics

Benchmark	Network	With Outliers			Outliers Removed		
		SAD ↓	MSE ↓	MAD ↓	SAD ↓	MSE ↓	MAD ↓
P3M-500-P	FBA (Ours)	21.35	0.009	0.012	7.78	0.002	0.005
	P3M-Net	**8.49**	**0.003**	**0.005**	**6.02**	**0.001**	**0.004**
P3M-500-NP	FBA (Ours)	24.28	0.011	0.014	8.82	0.003	0.005
	P3M-Net	**10.90**	**0.003**	**0.006**	**7.18**	**0.002**	**0.004**

Table A.2: Effect of Outliers on Trimap Unknown Region Metrics

Benchmark	Network	With Outliers			Outliers Removed		
		SAD ↓	MSE ↓	MAD ↓	SAD ↓	MSE ↓	MAD ↓
P3M-500-P	FBA (Ours)	8.55	0.029	0.058	7.05	0.026	0.053
	P3M-Net	**6.78**	**0.019**	**0.047**	**5.78**	**0.017**	**0.044**
P3M-500-NP	FBA (Ours)	10.31	0.031	0.059	8.07	0.025	0.053
	P3M-Net	**7.54**	**0.017**	**0.046**	**6.60**	**0.015**	**0.043**

Appendix B

Outlier Removal: Example Images

Figure B.1: More images that were considered outliers based on IQR — Part 1.

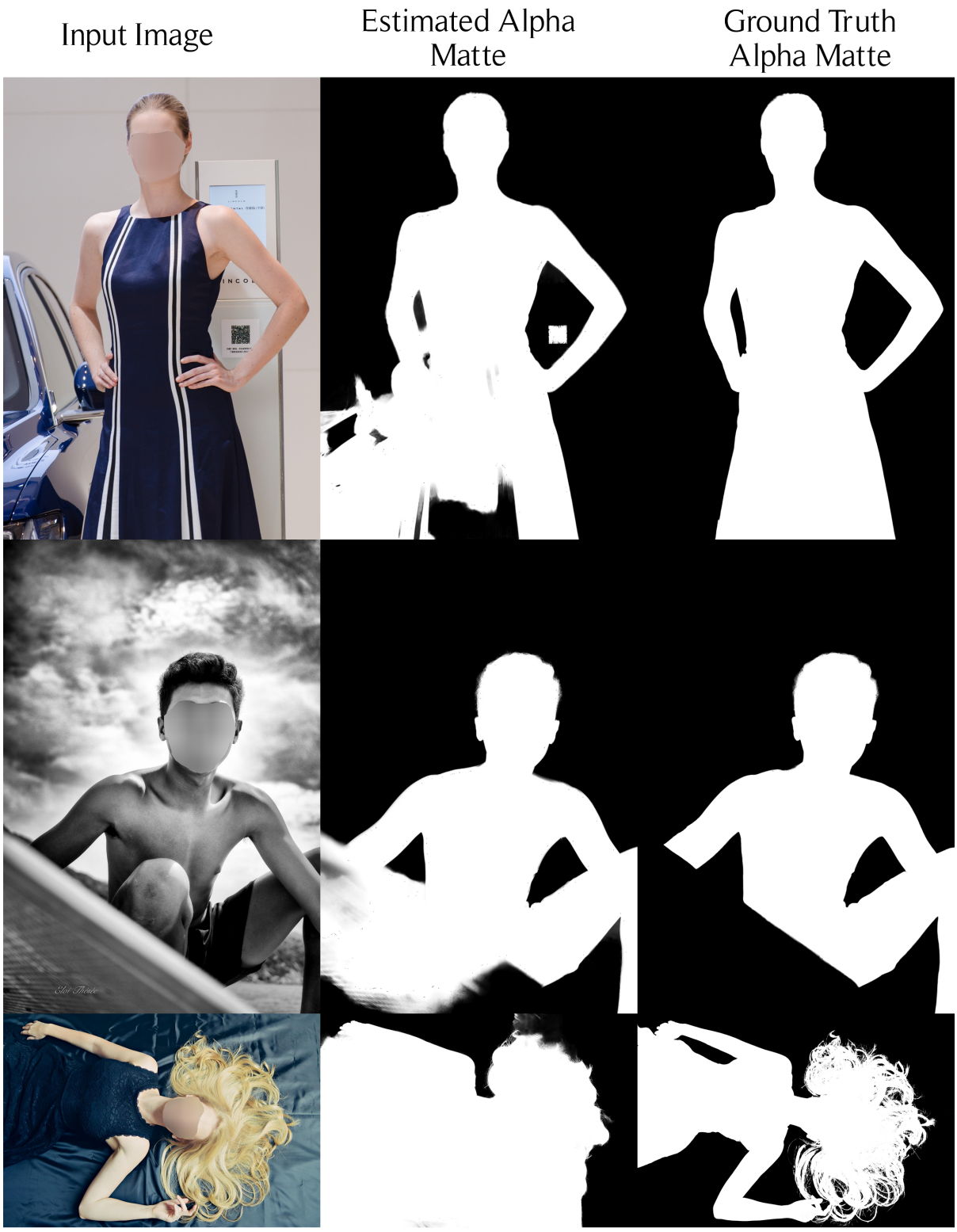

Figure B.2: More images that were considered outliers based on IQR — Part 2.

Figure B.3: More images that were considered outliers based on IQR — Part 3.

Appendix C

Visual Performance on Benchmark: Example Images

The following images contain more examples from the P3M-500-P benchmark to demonstrate the performance of our design and how it compares to P3M-Net.

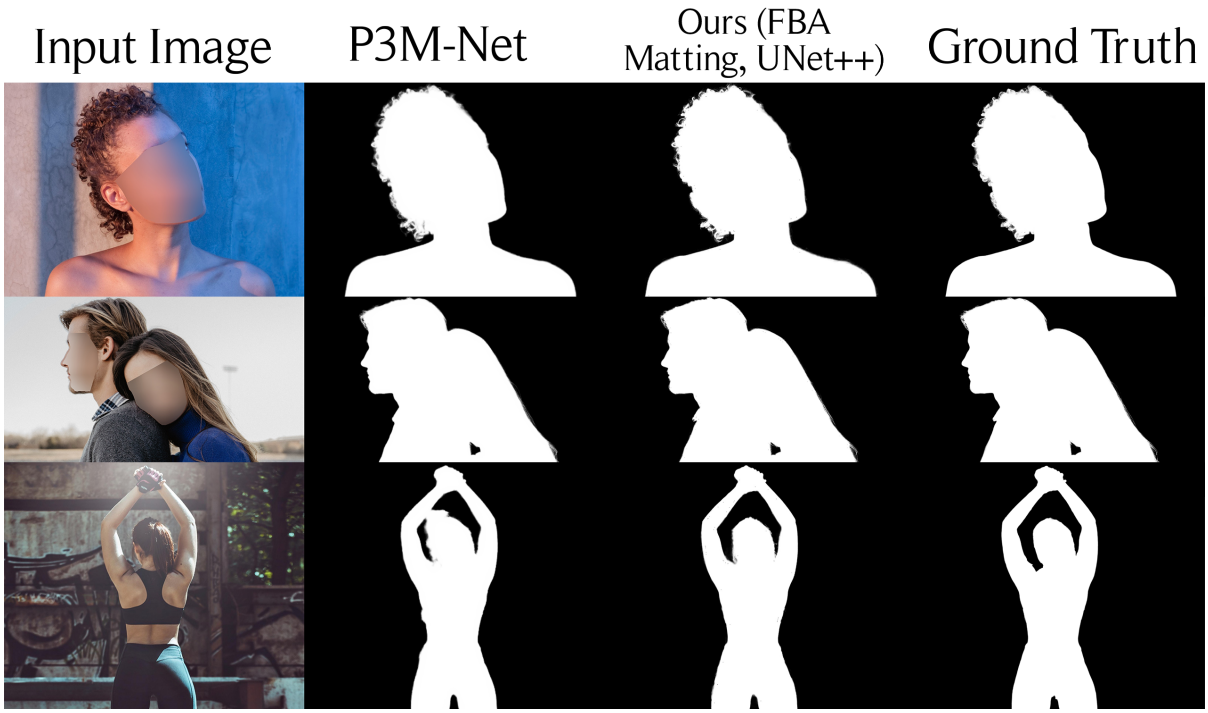

Figure C.1: More alpha mattes from P3M-500-P benchmark — part 1.

Figure C.2: More alpha mattes from P3M-500-P benchmark — part 2.

Figure C.3: More alpha mattes from P3M-500-P benchmark — part 3.

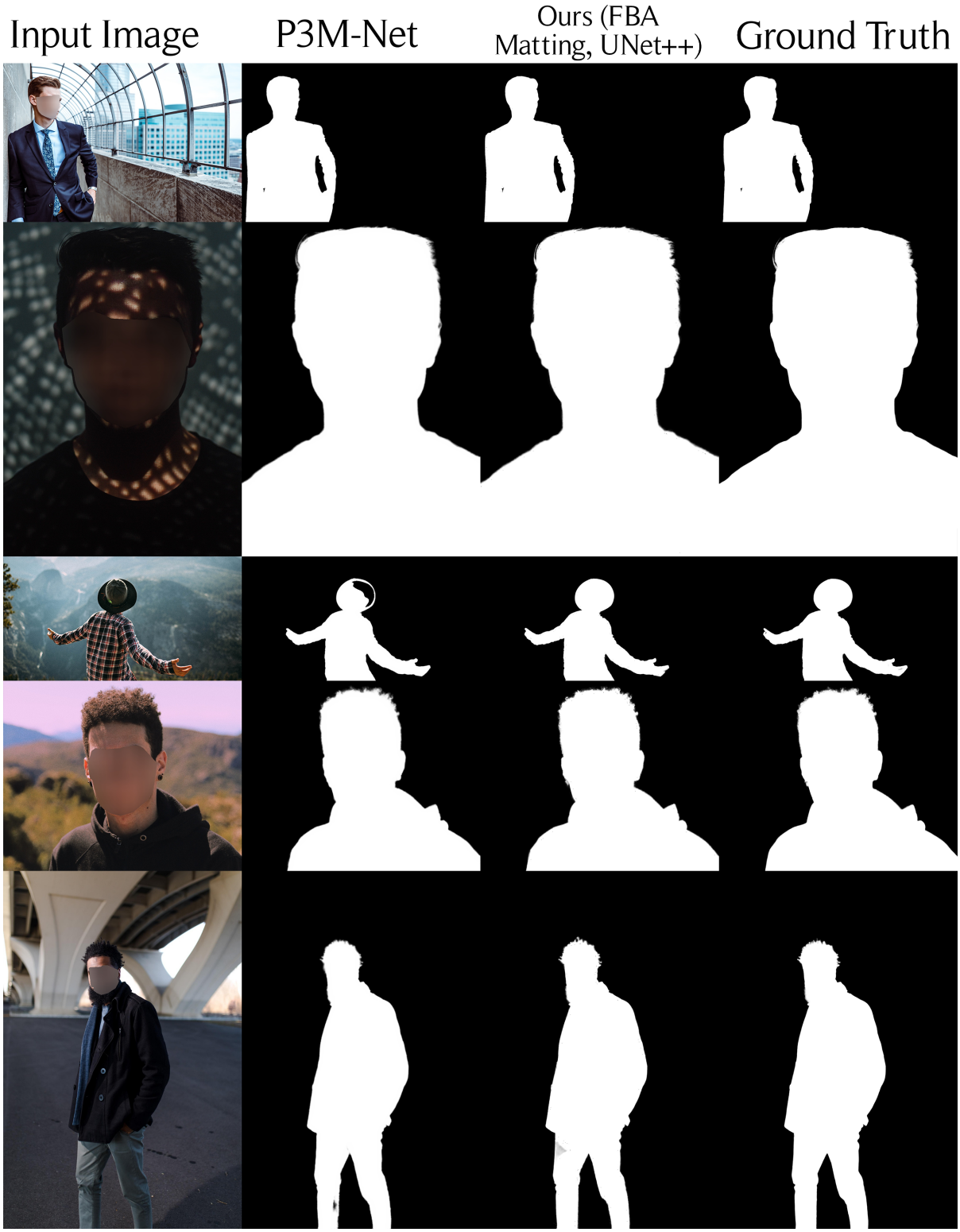

Figure C.4: More alpha mattes from P3M-500-P benchmark — part 4.

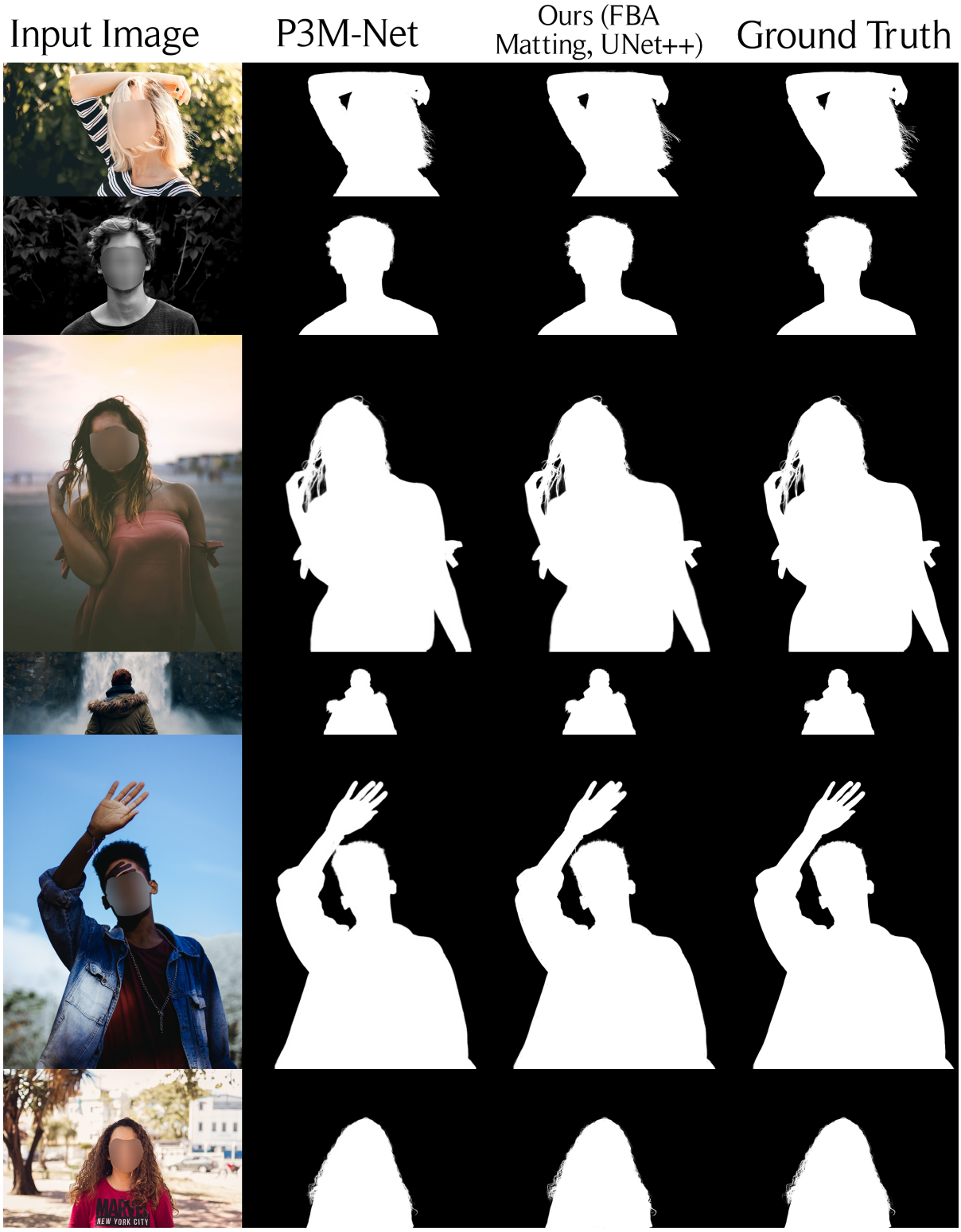

Figure C.5: More alpha mattes from P3M-500-P benchmark — part 5.

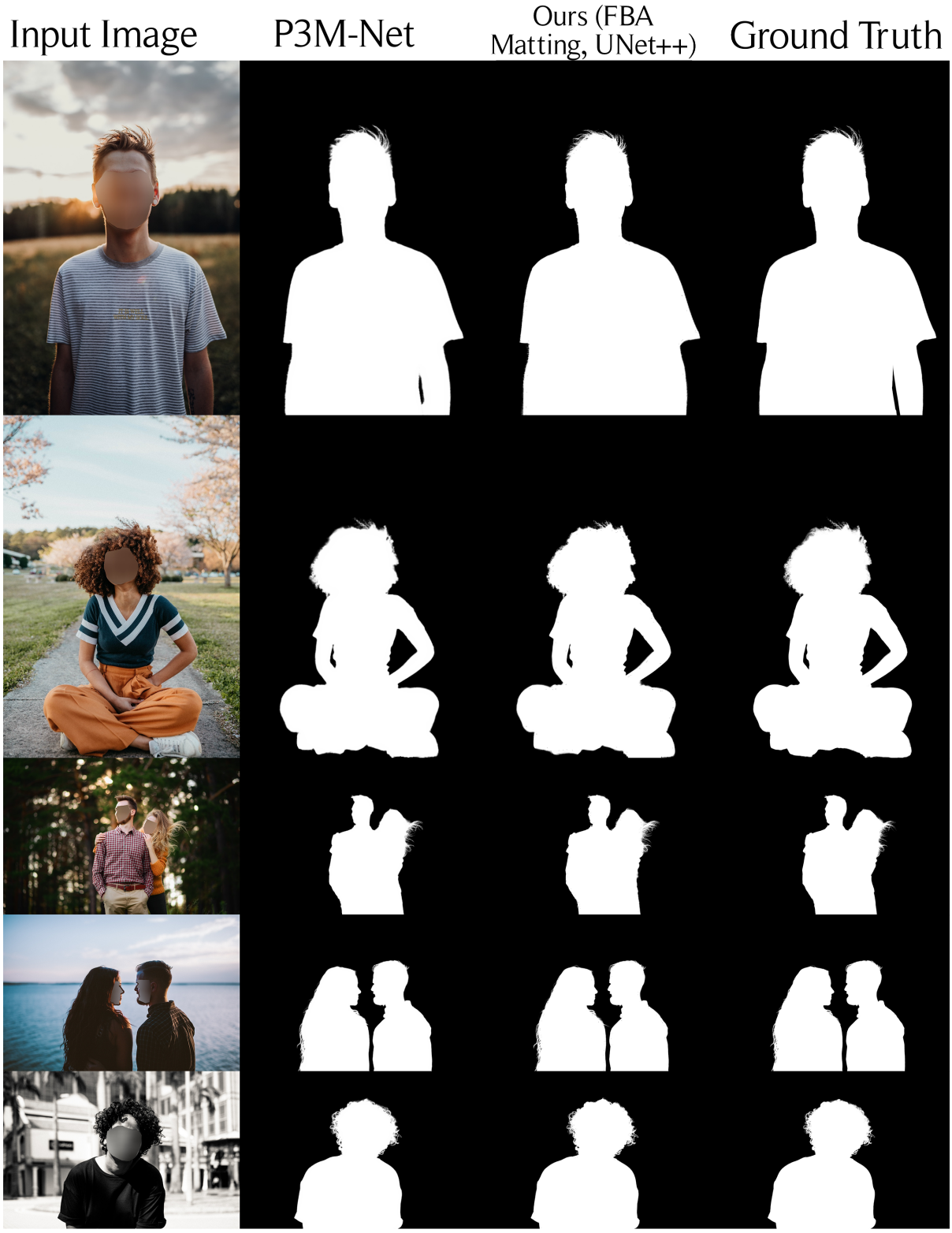

Figure C.6: More alpha mattes from P3M-500-P benchmark — part 6.

Appendix D

Visual Performance on Video: Example Images

The following images are stills from stock videos to demonstrate the performance of our design and how it compares to P3M-Net.

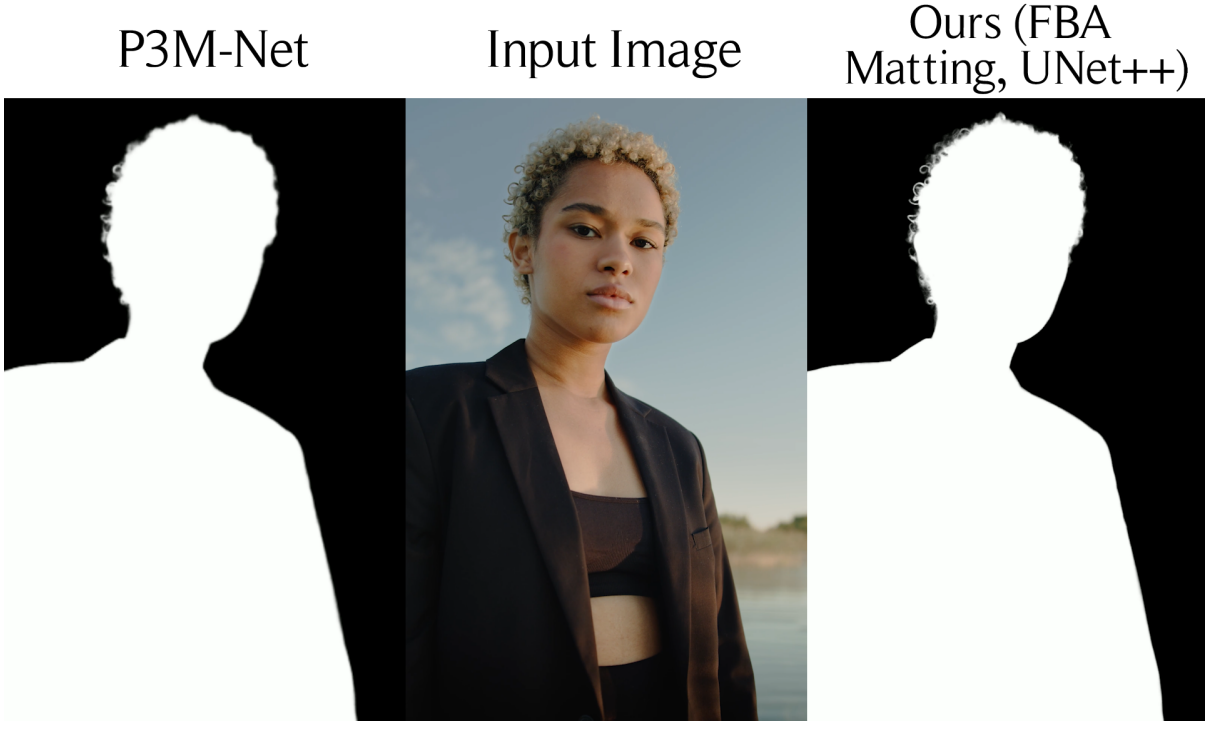

Figure D.1: Visual performance on video of woman in water: medium shot, part 2.

Figure D.2: Visual performance on video of woman jogging: medium shot, foreground extracted, part 2.