Robust Human Matting via Semantic Guidance

Xiangguang Chen*[1], Ye Zhu*[2], Yu Li**[3], Bingtao Fu[1], Lei Sun[1], Ying Shan[2], and Shan Liu[1]

[1] Platform Technologies, Tencent Online Video
[2] ARC Lab, Tencent PCG
{seanxgchen,samuelzhu}@tencent.com
[3] International Digital Economy Academy (IDEA)
liyu@idea.edu.cn

Abstract. Automatic human matting is highly desired for many real applications. We investigate recent human matting methods and show that common bad cases happen when semantic human segmentation fails. This indicates that semantic understanding is crucial for robust human matting. From this, we develop a fast yet accurate human matting framework, named Semantic Guided Human Matting (**SGHM**). It builds on a semantic human segmentation network and introduces a light-weight matting module with only marginal computational cost. Unlike previous works, our framework is data efficient, which requires a small amount of matting ground-truth to learn to estimate high quality object mattes. Our experiments show that trained with merely 200 matting images, our method can generalize well to real-world datasets, and outperform recent methods on multiple benchmarks, while remaining efficient. Considering the unbearable labeling cost of matting data and widely available segmentation data, our method becomes a practical and effective solution for the task of human matting. Source code is available at https://github.com/cxgincsu/SemanticGuidedHumanMatting.

1 Introduction

Human matting aims to predict an alpha matte to extract human foreground from an input image or video, which has many important applications in visual processing. To achieve that, a green screen is often required for studio solutions. However, a green screen is not always available in many real scenarios, such as daily video conferencing and background replacement effects shot with mobile devices. Therefore, human matting methods without a green screen are highly desired. Many previous works use an additional trimap for matting, which indicates three kinds of regions in an image, namely foreground, background, and unknown. However, it requires careful manual annotation to obtain a trimap. Background matting approaches [1, 2] are recently proposed which use a pre-recorded background image as a prior. Though decent results are obtained, it only can handle cases with a static background and a fixed camera pose.

* These authors contributed equally to this work.
** Corresponding Author.

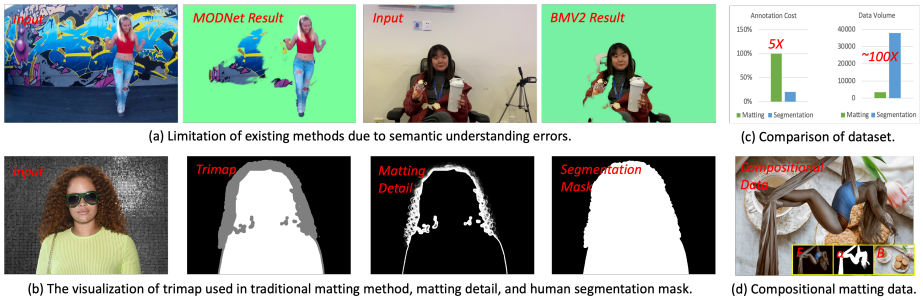

Fig. 1. Limitation of existing method and motivations of this work. (a) Common failure cases of latest works [1, 3] happen when semantic understanding fails. (b) Traditional matting methods rely on the input of a trimap. Since matting details are located around the human mask boundaries [4], the coarse segmentation mask can also be leveraged as a prior in matting. (c) Currently, segmentation data is much easier to annotate, and the amount of publicly available data is much larger than that of matting data. (d) Compositing foreground with different backgrounds can enlarge the matting dataset size but it has a domain gap as it looks unreal [5].

Many recent works focus on developing methods towards automatic human matting. Some early attempts [6, 7] try to generate pseudo trimap as a first step and predict a matte from the trimap. Due to limited training data, these methods cannot generalize well to real-world examples [2]. Another drawback of these methods is that they cannot run in real-time which is required for many applications, such as background replacement in live video conferencing. The recent work MODNet [3] proposes a fast and fully automatic portrait matting method. RVM [8] is another recent work which leverages temporal information in a video to improve robustness and stability.

In this work, we aim to develop a robust, accurate, and fast method for automatic human matting, which shares the same goal as MODNet [3] and RVM [8]. We investigate the failure cases of existing automatic methods on real-world examples and observe that these failure cases are often due to inaccurate semantic understanding. As shown in Fig. 1 (a), parts of the background are wrongly predicted as foreground or part of human body are wrongly segmented. This indicates a weak semantic understanding ability of these state-of-the-art (SOTA) methods. In order to enhance their ability of semantic understanding while keeping fine-grained details of matting, we seek to utilize semantic segmentation task to guide matting process. There are three reasons behind this motivation. 1) Segmentation mask determines the overall accuracy of foreground and background predictions, and fine-grained structures only appear around the mask. This indicates that a semantic human mask can replace a trimap (Fig. 1 (b)) and be used as a prior condition for matting [4]. 2) The labeling of high-quality matting requires skillful annotators and is very time-consuming. For that, the amount of available training data for matting is quite limited (at the order of hundreds and thousands) compare to segmentation task, which require only simple line draw-

ings around boundaries. As a matter of fact, there are many human segmentation datasets at a scale that is two or more magnitude larger (Fig. 1 (c)). A larger amount of data is of great significance to the generalization ability on real-world images. 3) Synthetic datasets created by compositing images (Fig. 1 (d)) are also used in training matting models, but they have a clear limitation due to the drastic domain gap between synthetic and real-world images. This prevents the trained models from generalizing to real-world examples. The work [5] analyzes the domain gap issue systematically. Our approach does not suffer from this issue by using less of such data.

Based on the above analysis, We propose a multi-stage framework to predict semantic segmentation mask and matting alpha successively. A segmentation sub-network is first employed for the task of segmentation, and then it is reused to guide the matting process to focus on the surrounding area of the segmentation mask. To achieve real-time efficiency as well as better performance, we let the two tasks share the encoder part of the model, which has been proved superior to separated encoders in [5]. By this design, our matting module successfully handled many challenging cases. In summary, our network consists of a shared encoder, a segmentation decoder and a matting decoder, and the segmentation decoder feeds useful intermediate information to the matting decoder. In training, a two-stage pipeline is proposed. Firstly, the encoder and the segmentation decoder are trained with publicly available segmentation datasets. With these data, our segmentation sub-network is trained to predict robust human masks. Secondly, 269 matting images are employed to train the matting decoder. To comprehensively evaluate the performance of matting methods, we adopt 5 benchmarks to carry out qualitative and quantitative comparison. One of them is our self-collected dataset from complex scenarios, such as diverse background, multiple human, body accessories, and low light. Our method outperforms all other methods across all benchmarks.

We summarize our contribution as follows:

1. We develop a robust, accurate and efficient human matting framework, which utilizes shared encoder for both segmentation and matting. It gives our method the ability to use powerful semantic understanding to guide matting process meanwhile help to reduce computation.
2. The proposed framework can make fully use of coarse mask training data and reduce matting reliance on high-quality and large number of annotations. With only about 200 matting images, our method is able to produce high quality alpha details.
3. Extensive experiments show our method achieves the state-of-the-art results on multiple benchmarks.

2 Related Work

In this section, we review matting with auxiliary input and automatic matting, which are related to our work. We also review segmentation as segmentation provides the rough mask of human region.

Matting with auxiliary input. Early methods are mostly optimization or filter based which require an additional trimap as input [9–19]. Deep learning is introduced in trimap-based matting methods in [20–22] that use a deep network for trimap-based matting. These trimap-based methods are often general to different matting target objects but it requires the user to provide trimap annotations. Background mattings [2, 1] are recently proposed to replace the trimap input with a pre-recorded background image as a prior condition. Although background matting can generate decent results on static background, it cannot be applied to camera moving circumstances. Recently proposed mask-guided method [4] achieves SOTA results once a coarse is provided. In their work, mask is generated from manual annotation or segmentation output, which greatly limits the convenience of use. Our goal is to incorporate the mask generation into the matting process, so as to realize fully automatic matting and still keeping real-time running.

Automatic matting. Fully automatic matting without any additional input has been pursued [23–25]. Methods in [26, 27] studies class agnostic matting but cannot generalize well. Some methods like [6, 7, 28–30] dedicate to human matting. In this direction, the latest MODNet [3] aims at fast portrait matting and RVM [8] is towards robust human matting using temporal information. For MODNet, it performs well in the portrait image, but easily fails in full body image. Recent work P3M-Net [31] proposes a dual decoder to do human matting, which is similar to us. But there are several significant differences: 1) P3M-Net use segmentation decoder to generate a pseudo trimap while our segmentation predicts real mask. P3M-Net predicts alpha details only on trimap unknown region. This setting tends to output false matting results when trimap is wrongly predicted. Our matting decoder treats mask as guidance and regresses alpha at the whole image. Under this setting, the matting decoder is given an opportunity to correct semantic errors. 2) Our segmentation decoder and matting decoder are trained at two separate stages. At the segmentation training stage, the segmentation decoder is strongly supervised by a large dataset. As a result, the segmentation decoder predicts more robust results than the weakly supervised result in P3M-Net. 3) Another advantage of our model is it is data-efficient in that we only use a very small amount of high-precision data to train the matting decoder.

Segmentation. Semantic segmentation assigns a semantic class label to every pixel in the scene. Its difference with matting is that it predicts a hard binary mask that belongs to either foreground or background and cannot generate fine details and transparent value as in matte. So directly applying segmentation mask to image and video composition will generate hard boundary at the foreground object, leaving noticeable artifacts when replacing the backgrounds. However, segmentation can provide strong semantic cues of the object location which facilitate our matting task. Many deep learning-based semantic segmentation are fully convolutional and some effective modules like Atrous Spatial Pyramid Pooling (ASPP) [32] are proposed. We follow them in our segmentation network design.

3 Method

Given a color image I, the matting task can be formulated as follows:

$$I = \alpha F + (1 - \alpha)B, \ \alpha \ \in \ [0, 1], \tag{1}$$

where F, B are foreground and background, and α is the alpha matte denoting where is foreground part located. For image matting problem, we should predict the alpha matte from the input color image, which is a hard and ill-posed task. As mentioned earlier, existing methods rely on additional auxiliary inputs like trimap or pre-captured background. Automatic method like RVM is not robust against semantic error. Based on this, we try to design a framework to better leverage the semantic prior from segmentation, but produce fine detail and transparent matte values. A straightforward way is to rely on a semantic segmentation mask and generate the matting results using a new matting network. This setup is developed and demonstrated in mask-guided (MG) matting [4]. The two-step setup treats segmentation and matting as two separate tasks and has a few drawbacks. First of all, the matting network only uses the predicted segmentation map and ignores the rich semantic features. Second, using a separate matting network will extract features again from image and introduce additional computation, which slows down the speed noticeably on high resolution.

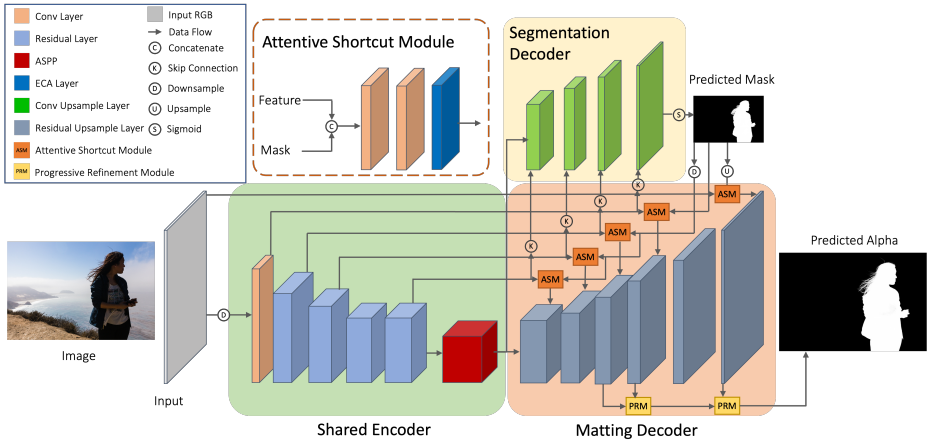

Fig. 2. The network structure of our SGHM. High-resolution image is first downsampled for the shared encoder, then the segmentation decoder is used to generate a coarse semantic mask prediction. We propose an Attentive Shortcut Module(ASM) to adaptively fuse shared features and masks. Finally, the matting decoder refines the unknown area of human margin and predicts the alpha matte.

Based on above analysis, we propose a new human matting method named Semantic Guided Human Matting (SGHM), which uses a segmentation network

to guide human matting. Specifically, we share the encoder between segmentation and matting task. Thus, matting task can learn accurate semantic understanding from reusing the rich semantic features in encoder and focus on predicting alpha details in matting decoder.

As shown in Fig. 2, our SGHM consists of a shared encoder to extract image features, a segmentation decoder to predict image segmentation mask, and a matting decoder with Progressive Refinement Module (PRM) [4] to predict a high-resolution matting result. We propose to use an Attentive Shortcut Module (ASM) to combine the features from encoder and mask from segmentation decoder for matting decoder.

3.1 Shared Encoder

As mentioned above, we propose to improve matting results by using semantic human segmentation features. So we make the segmentation and matting tasks share an encoder. More specifically, we first train the encoder and segmentation decoder as segmentation model, and then fix the parameters of the encoder and train the matting decoder with segmentation features extracted from encoder. We adopt ResNet50 [33] as feature extraction backbone followed by a ASPP module [32] for shared encoder, which extracted features at $\frac{1}{4}$, $\frac{1}{8}$, $\frac{1}{16}$, $\frac{1}{32}$, $\frac{1}{64}$ scale for two decoders with an input image at $\frac{1}{4}$ scale, which can be denoted as F_0, F_1, F_2, F_3, F_4.

3.2 Segmentation Decoder

Our segmentation decoder is a light-weight and efficient module, which contains 4 convolution layers and 4 up-sample layers. For each convolution layer, it can be defined as:

$$X_i = \text{Conv}(\text{Concat}(\text{Upsample}(X_{i+1}), F_i), i = 3, 2, 1, 0, \tag{2}$$

where F_i is the feature from shared encoder and X_i is the output feature of convolution layer. In particular, $X_4 = F_4$ is the direct input of segmentation decoder. Following each convolution layer, a batch normalization layer and a ReLU layer are attached except the last one. Finally, we obtain the output segmentation mask S. We denote our segmentation branch as SGHM-S in reporting the results later.

3.3 Matting Decoder

Our matting decoder inputs the segmentation features and segmentation mask of different scales, outputs the matting results at $1, \frac{1}{4}, \frac{1}{8}$ scale. Firstly, we use ASM module to combine the features from encoder and segmentation mask. Then we sequentially process the features of different scales by several upsample blocks. We predict matting results at $1, \frac{1}{4}, \frac{1}{8}$ scale by output modules. Finally, we adopt

PRM module to produce the final high-resolution matting result based on the matting results at three output scales.

Attentive Shortcut Module. Our model proposed to use semantic segmentation to improve human matting by sharing encoder of segmentation and matting. In addition to features from shared encoder , we also feed the segmentation mask of different scales as input of matting decoder. For matting decoder, how to fuse the features and mask from segmentation is of vital importance. One direct way is to concatenate these two inputs for further processing. We propose to use ASM to fuse these two inputs. With the help of ASM, we can get more adaptive features for matting decoder. Specifically, the ASM contains two convolution layers, two SpectralNorm layers [34] and an efficient channel attention layer [35]. Channel attention can produce an adaptive feature by calculating a channel-wise weight vector corresponding to input feature.

Upsample Block. Upsample block process input features sequentially from $\frac{1}{64}$ scale to the original scale. First, it element-wisely adds the feature of the current scale and the feature of the previous scale upsampled by residual blocks from $\frac{1}{64}$ scale to $\frac{1}{2}$ scale. Then, for $\frac{1}{2}$ scale and 1 scale, we replace the residual blocks with a single transposed convolution layer with batch normalization and ReLU for efficiency.

Output Block. We predict matting result at $1, \frac{1}{4}, \frac{1}{8}$ scale. For each output scale, we attach a matting result prediction block after the upsample block. Each prediction block contains a convolution layer, batch normalization layer, ReLU and convolution layer sequentially.

Progressive Refinement Module. We adopt Progressive Refinement Module (PRM) [4] to further refine the output matting alphas from output blocks. PRM can selectively fuse the matting alphas from the previous scale and the current scale with a self-guidance mask, which can preserve the confident regions from the previous scale and focus on refining uncertain regions at the current scale. Specifically, the self-guidance mask of the current scale is generated from matting alpha obtained at the previous scale as follows:

$$g_l = \begin{cases} 0 & 0 < \alpha_{l-1} < 1, \\ 1 & \text{otherwise}, \end{cases} \tag{3}$$

where α_{l-1} is the matting alpha of previous scale.The α_{l-1} is upsampled to match the size of the raw matting output α'_l of the current scale. With the self-guidance mask g_l, the refined matting alpha of current scale can be calculated as following:

$$\alpha_l = \alpha'_l g_l + \alpha_{l-1}(1 - g_l). \tag{4}$$

Note that features in confident region predicted from the previous scale are preserved and the current scale only focuses on refining the uncertain region.

4 Training

We train our SGHM model in two stages. First, we train the segmentation network using widely available segmentation datasets. In this stage, the parameters

of shared encoder and segmentation decoder are updated simultaneously. After segmentation net is trained, the shared encoder can extract powerful semantic features to provide information for the matting task. Next, we fix the shared encoder and segmentation decoder, and train the matting decoder only. The coarse mask output from segmentation net is also used as the input at this stage. During inference, the two decoders are executed successively. Segmentation mask is first predicted and fed to matting decoder to produce matting result.

4.1 Segmentation Training

We train the segmentation sub-network with about $47.2k$ paired images, which are from SPD[36] (about $2.5k$), Portrait Matting [7] (about $1.7k$), dataset released in [37](about $5.2k$), human parsing dataset [38] (about $4.7k$), Privacy-Preserving dataset [31] (about $9.4k$) and green screen dataset from BMV2 [1] (about $23.7k$). We treat green screen data as segmentation mask since it provides more body posture diversity than alpha details. Note that we drop some image pairs by annotation checking, and collect about $35k$ background images from internet for random background composition.

We adopt Binary Cross Entropy (BCE) loss to train segmentation model. For data augmentation, we adopt random affine transformation, random horizontal flipping, random noise, random color jitters, random composite, and random crop to 320×320. We train our segmentation model on 8 NVIDIA Tesla A100 GPUs with a batch size of 10 for each GPU. We use Adam as optimizer, and the learning rate is initialized to $5e^{-4}$. The model is totally trained for 100 epochs with a cosine learning rate decay scheduler.

4.2 Matting Training

We train matting model on the foreground images of AIM [20] dataset except transparent object images. The total foreground images are 269, and we use MS COCO dataset as background images.

Following MG [4], we adopt l_1 regression loss, composition loss [20], Laplacian loss [39] for training matting model. We denote the ground-truth alpha with $\hat{\alpha}$ and the prediction alpha with α. Then the combined loss function can be formulated as:

$$L(\hat{\alpha}, \alpha) = L_{l_1}(\hat{\alpha}, \alpha) + L_{comp}(\hat{\alpha}, \alpha) + L_{lap}(\hat{\alpha}, \alpha). \tag{5}$$

We apply this combined loss on all output matting alphas at $1, \frac{1}{4}, \frac{1}{8}$ scale with adaptive weights g_l calculated in Eq. 3 to force the training to more focused on the unknown region at each scale. Moreover, we set different weights for different scales to form the final loss function as follows:

$$L_{tot} = \sum_{l} \omega_l L(\hat{\alpha}_l \cdot g_l, \alpha_l \cdot g_l), \tag{6}$$

where ω_l is the loss weight of different scales. We set $\omega_{\frac{1}{8}} : \omega_{\frac{1}{4}} : \omega_1 = 1 : 2 : 3$ in our experiments.

We train our matting model with $100,000$ iterations on 4 NVIDIA Tesla A100 GPUs with a batch size of 8 for each GPU. We use Adam as optimizer, and the learning rate is initialized to $1e^{-3}$. We adopt the same data augmentation with training of segmentation, with a random crop of 1280×1280. We also adopt mask perturbation for augmentation. Note that we fix the parameters of segmentation model during training matting module, which can force the matting decoder to focus more on features to predict alpha details. If not fixed matting performance will drop as it will overfit to the small set of matting data.

5 Experiment

5.1 Benchmarks

To verify the effectiveness of the proposed method, we evaluate the performance on the following 5 benchmarks, including three real-world datasets and two composition datasets.

AIM [20]. We select 12 human images from AIM dataset for testing. Each foreground human image is composited to 20 backgrounds which are selected from top-240 of BG-20K[40] test set.

D646 [26]. Similar to AIM, 11 foreground images are composited with the last 220 backgrounds from BG-20K test set.

PPM-100 [3]. This dataset provides 100 finely annotated portrait images with various backgrounds. Images from PPM-100 are more realistic and natural than composition images.

P3M-500-NP [31]. We use the face kept images rather than face masked from P3M. The purpose is to avoid the unknown impact of face blur on evaluation. This benchmark has a great diversity of body postures.

RWCSM-289. To further verify our model generalization, we build a real-world complex scene matting dataset, denoted as RWCSM-289. It contains a variety of complex living and working scenarios. Its sources are hand-hold captured videos, online video meetings, TV shows, live videos, and Vlogs. Many of them come from youtube and are used by RVM [8]. It is worth noting that this dataset include motion and multi-person scenes, which is helpful to evaluate model robustness. The ground truth alpha is annotated by PhotoShop.

5.2 Quantitative Comparison

We compare our approach with the state-of-the-art automatic matting methods, including LFM [27], SHM [6], HATT [26], BSHM [41], MODNet [3], P3MNet [31], video matting method RVM [8] and mask-guided method MG [4]. We use inference size 512 for MODNet since it provides the best results on PPM-100. For RVM, We generate 10 frames video by repeating 10 times for every single image and take last frame result as evaluation target. For P3MNet the recommended testing resize strategy is used. For MG, we feed our segmentation result to its network as mask guidance. Both MG and our method keep the short size of

Table 1. Quantitative results on real-world benchmarks. '↓' : lower values are better.

Dataset	Method	MAD↓	MSE↓	Grad↓	Conn↓
PPM-100	LFM	15.80	9.40	-	-
	SHM	15.20	7.20	-	-
	HATT	13.70	6.70	-	-
	BSHM	11.40	6.30	-	-
	P3MNet	15.61	12.86	56.37	130.42
	MODNet	8.60	4.40	64.26	80.82
	RVM	10.95	6.53	63.13	105.19
	SGHM (ours)	**5.97**	**2.58**	**48.20**	**51.17**
P3M-500-NP	LFM	18.80	13.10	31.93	19.50
	SHM	12.20	9.30	20.30	17.09
	HATT	17.60	7.20	19.99	27.42
	P3MNet	6.50	3.50	**10.35**	12.51
	MODNet	12.82	7.41	16.02	20.23
	RVM	11.10	7.06	15.30	19.17
	SGHM (ours)	**6.49**	**3.11**	11.39	**10.16**
RWCSM-289	P3MNet	32.92	31.09	28.42	77.37
	MODNet	18.95	15.76	19.65	46.18
	RVM	14.36	11.25	15.68	28.52
	SGHM (ours)	**9.23**	**6.57**	**13.52**	**18.68**

images to 1280 when testing. We use mean absolute difference (MAD), mean squared error (MSE), spatial gradient (Grad) [42], and connectivity (Conn) [42] as alpha matting quality metrics. Note that MAD and MSE values are scaled by 10^3 and all metrics are calculated over the whole image.

Table 1 and Table 2 show the results of different matting methods evaluated on real-world and composition datasets. It shows that our method outperforms other methods across all real-world datasets in all metrics. Specifically, our method is ahead of compared method on PPM-100. On P3M-500-NP, we achieve the results (MAD 6.49, MSE 3.11) that are on par with the P3MNet (MAD 6.50, MSE 3.50) by only introducing face-masked P3M data into the segmentation stage. For complex scene data RWCSM-289 which covers more diversity of background, number of humans, body accessories, illumination, and image resolution, we significantly outperform P3MNet and MODNet, and are better than video-based approach RVM. On the composition datasets, SGHM still achieves the best results, showing consistently excellent performance of the proposed method.

5.3 Qualitative Comparison

This section shows qualitative comparisons on real-world benchmarks. We reveal alpha details in Fig. 3 and model robustness in Fig. 4. In Fig. 3 rows 1 to 4, we

Fig. 3. Visual comparison of different methods on alpha details. SGHM-S denotes the segmentation results of our method. SGHM denotes the final matting results. SGHM-S+MG denotes using SGHM-S as extra input for MG. Our proposed method produces superior results from coarse to fine. Best viewed on monitor with zooming in for detail.

compare hair details and find ours predict fine-grained hair details comparable to mask-based method MG, which are more accurate than P3MNet, MODNet and RVM. Multiple body postures are displayed in rows 5 to 8. Other methods tend to get semantic errors (can be found in MODNet at row of 6, MG at row of 7, P3MNet at row 5) while SGHM produces more accurate alpha matte. It is worth noting that our method has the ability to correct semantic errors in the coarse masks (see row 1, 6 and 8 from SGHM-S to SGHM).

In Fig. 4, we select two SOTA methods MODNet and RVM for robustness comparison from four categories of videos. The extracted foreground is composited with a green background for visualization. Our method predicts much fewer semantic errors and demonstrates better robustness against semantic understanding errors than the other two methods.

Table 2. Quantitative results on composition benchmarks. '↓' : lower values are better.

Dataset	Method	MAD↓	MSE↓	Grad↓	Conn↓
AIM	P3MNet	44.78	37.70	43.02	100.80
	MODNet	33.18	23.58	29.08	74.47
	RVM	27.07	17.54	28.84	60.73
	SGHM (ours)	**14.34**	**7.18**	**19.29**	**29.40**
D646	P3MNet	20.25	15.27	36.93	54.74
	MODNet	10.52	4.72	32.62	28.61
	RVM	10.50	4.94	35.24	28.60
	SGHM (ours)	**6.59**	**2.19**	**19.07**	**17.02**

Table 3. Size and Speed Comparison. The matting metrics are evaluated on PPM-100 dataset and the speed is evaluated on HD size on an NVIDIA A100 GPU. SGHM-S is the segmentation branch of our method. It runs at over 100 FPS as it is on 1/4 of full image resolution.

Method	#Parameters (M)	FPS	MAD	MSE
MODNet	6.49	20.76	8.60	4.40
RVM	3.75	71.81	10.95	6.53
SGHM-S	40.22	106.14	11.84	5.72
SGHM-S+MG	69.92	18.14	8.83	4.18
SGHM	43.94	34.76	5.97	2.58

5.4 Size and Speed Comparison

As mentioned in Section 3, MG uses a segmentation mask as extra input to its matting network. Unlike MG, we incorporate the mask generation stage into matting framework. Since MG uses an independent matting network, it introduces more parameters and its total parameter number is the combination of segmentation and matting networks. The speed is thus slowed down. Our SGHM shares the encoder with the segmentation, which causes marginal extra parameters. SGHM also runs faster than MG on the same setting and can achieve 34 FPS on HD image (1920×1080) on NVIDIA A100. For the matting quality, our method achieves better performance. This shows we have both speed and accuracy advantages over MG. MODNet and RVM are also compared. Although they have fewer parameters, they both have limitation. MODNet runs slower on HD inference size (20.76 FPS) than 512 (81.01 FPS). RVM predicts unsatisfactory fine-grained alpha results across all benchmarks.

5.5 Ablation Studies

Role of Segmentation Task. We propose to introduce segmentation task to improve the performance and generalization of alpha matting in two ways. One

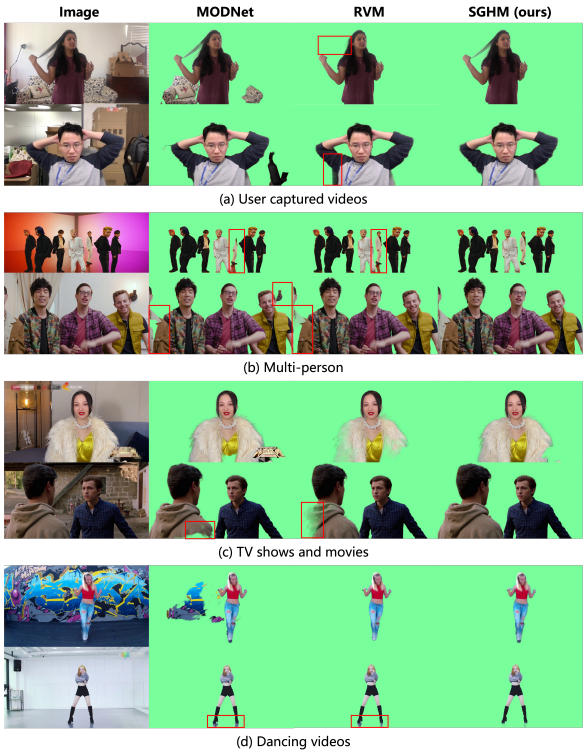

Fig. 4. Visual comparison of different methods on four categories of videos. Our method is more robust to semantic errors.

is sharing encoder features and other is coarse mask guidance. Table 4 shows our ablation study results on PPM-100. The results lead to two conclusions: (1) Sharing semantic features is very beneficial to matting task, which helps to reduce MAD from 7.83 to 5.97. (2) Mask guidance plays an indispensable role in guiding matting process, as matting performance drops dramatically when mask guidance is removed. Since our matting model is trained on only hundreds of images, robust semantic features and good mask guidance are both helpful for improving model generalization.

Role of ASM. We propose ASM to combine semantic features and segmentation mask for matting decoder. As listed in the fourth and fifth rows of Table 4, model gets worse results without ASM. SpectralNorm and ECA layer are the two key components in ASM. In-depth analysis reveals that MAD drops from 5.97 to 7.11 when ECA layer is removed, while MAD is 6.33 when SpectralNorm is removed. This indicates ECA layer contributes more as it channel-wisely re-weight the features to adapt them for matting. Note that in the first row, we remove the mask input to only use the features from encoder and keep the same Conv layers with proposed ASM.

Table 4. Ablation study on different settings, tested on PPM-100.

ASM	Mask guidance	Sharing encoder weights	MAD	MSE
√			17.23	8.54
		√	10.46	5.07
√	√		7.83	4.04
	√	√	7.50	3.52
√	√	√	**5.97**	**2.58**

Table 5. Results of different training datasets sizes, tested on PPM-100. The LargeSeg dataset consists of 140k human masks which are collected from multiple publicly available datasets. The size of D646 is 362 which is selected from the Distinctions-646 training set.

	Segmentation Dataset Size	Matting Dataset Size	MAD	MSE
Baseline	40k	200+	5.97	2.58
+LargeSeg	170k	200+	**5.16**	**2.04**
+D646	40k	600+	5.71	2.45

Role of Dataset Size. We further conduct an experiment to verify the data efficiency of our method. As can be seen in Table 5, a larger segmentation dataset improves matting results significantly, while increasing the matting dataset size improves slightly. Note that it is easy to collect these human segmentation masks from publicly available datasets. But labeling fine-grained matting requires a much higher annotation skill level and it is time and money costing. This is an important and practical finding that we can efficiently improve matting performance by collecting more coarse human masks in an easy and fast way rather than paying for the high cost fine-detailed alpha annotating.

6 Conclusion

In this work, we investigate the major challenge in robust human matting and reveal that it is from the semantic understanding. Based on this, we propose a semantic guided human matting method. We introduce an additional matting decoder to the semantic segmentation network. By reusing the features from semantic segmentation encoder, the matting decoder is aware of global semantic information and also can generate fine matting details. With very small number of matting data, we can train a robust, accurate and real-time matting model which achieves top performance on multiple benchmark datasets. We believe that our proposed framework is a practical pipeline for matting application which does not rely on large number of high annotation cost matting data.

References

1. Lin, S., Ryabtsev, A., Sengupta, S., Curless, B., Seitz, S., Kemelmacher-Shlizerman, I.: Real-time high-resolution background matting. In: CVPR (2021)
2. Sengupta, S., Jayaram, V., Curless, B., Seitz, S., Kemelmacher-Shlizerman, I.: Background matting: The world is your green screen. In: CVPR (2020)
3. Ke, Z., Sun, J., Li, K., Yan, Q., Lau, R.W.: Modnet: Real-time trimap-free portrait matting via objective decomposition. In: AAAI (2022)
4. Yu, Q., Zhang, J., Zhang, H., Wang, Y., Lin, Z., Xu, N., Bai, Y., Yuille, A.: Mask guided matting via progressive refinement network. In: CVPR (2021)
5. Li, J., Zhang, J., Maybank, S.J., Tao, D.: Bridging composite and real: towards end-to-end deep image matting. International Journal of Computer Vision pp. 1–21 (2022)
6. Chen, Q., Ge, T., Xu, Y., Zhang, Z., Yang, X., Gai, K.: Semantic human matting. In: ACM MM (2018)
7. Shen, X., Tao, X., Gao, H., Zhou, C., Jia, J.: Deep automatic portrait matting. In: ECCV (2016)
8. Lin, S., Yang, L., Saleemi, I., Sengupta, S.: Robust high-resolution video matting with temporal guidance. In: WACV (2022)
9. Aksoy, Y., Ozan Aydin, T., Pollefeys, M.: Designing effective inter-pixel information flow for natural image matting. In: CVPR (2017)
10. Chen, Q., Li, D., Tang, C.K.: Knn matting. TPAMI **35**(9), 2175–2188 (2013)
11. Chuang, Y.Y., Curless, B., Salesin, D.H., Szeliski, R.: A bayesian approach to digital matting. In: CVPR (2001)
12. Gastal, E.S., Oliveira, M.M.: Shared sampling for real-time alpha matting. In: Computer Graphics Forum. vol. 29, pp. 575–584 (2010)
13. Levin, A., Lischinski, D., Weiss, Y.: A closed-form solution to natural image matting. TPAMI **30**(2), 228–242 (2007)
14. Levin, A., Rav-Acha, A., Lischinski, D.: Spectral matting. TPAMI **30**(10), 1699–1712 (2008)
15. Sun, J., Jia, J., Tang, C.K., Shum, H.Y.: Poisson matting. In: ToG. vol. 23, pp. 315–321 (2004)
16. Chen, T., Wang, Y., Schillings, V., Meinel, C.: Grayscale image matting and colorization. In: ACCV (2004)
17. Pham, V.Q., Takahashi, K., Naemura, T.: Real-time video matting based on bilayer segmentation. In: ACCV (2009)
18. Park, Y., Yoo, S.I.: A convex image segmentation: Extending graph cuts and closed-form matting. In: ACCV (2010)
19. Sindeev, M., Konushin, A., Rother, C.: Alpha-flow for video matting. In: ACCV (2012)
20. Xu, N., Price, B., Cohen, S., Huang, T.: Deep image matting. In: CVPR (2017)
21. Forte, M., Pitié, F.: F, b, alpha matting. CoRR **abs/2003.07711** (2020)
22. Liu, Y., Xie, J., Shi, X., Qiao, Y., Huang, Y., Tang, Y., Yang, X.: Tripartite information mining and integration for image matting. In: ICCV (2021)
23. Yang, S., Wang, B., Li, W., Lin, Y., He, C., et al.: Unified interactive image matting. arXiv preprint arXiv:2205.08324 (2022)
24. Dai, Y., Price, B., Zhang, H., Shen, C.: Boosting robustness of image matting with context assembling and strong data augmentation. In: CVPR. pp. 11707–11716 (2022)

25. Chen, G., Liu, Y., Wang, J., Peng, J., Hao, Y., Chu, L., Tang, S., Wu, Z., Chen, Z., Yu, Z., et al.: Pp-matting: High-accuracy natural image matting. arXiv preprint arXiv:2204.09433 (2022)
26. Qiao, Y., Liu, Y., Yang, X., Zhou, D., Xu, M., Zhang, Q., Wei, X.: Attention-guided hierarchical structure aggregation for image matting. In: CVPR (2020)
27. Zhang, Y., Gong, L., Fan, L., Ren, P., Huang, Q., Bao, H., Xu, W.: A late fusion cnn for digital matting. In: CVPR (2019)
28. Zhu, B., Chen, Y., Wang, J., Liu, S., Zhang, B., Tang, M.: Fast deep matting for portrait animation on mobile phone. In: ACM MM (2017)
29. Sun, Y., Tang, C.K., Tai, Y.W.: Human instance matting via mutual guidance and multi-instance refinement. In: CVPR (2022)
30. Xing, Y., Li, Y., Wang, X., Zhu, Y., Chen, Q.: Composite photograph harmonization with complete background cues. In: ACM MM (2022)
31. Li, J., Ma, S., Zhang, J., Tao, D.: Privacy-preserving portrait matting. arXiv (2021)
32. Chen, L.C., Papandreou, G., Schroff, F., Adam, H.: Rethinking atrous convolution for semantic image segmentation. ArXiv (2017)
33. He, K., Zhang, X., Ren, S., Sun, J.: Deep residual learning for image recognition. CVPR (2016)
34. Miyato, T., Kataoka, T., Koyama, M., Yoshida, Y.: Spectral normalization for generative adversarial networks. arXiv (2018)
35. Wang, Q., Wu, B., Zhu, P., Li, P., Zuo, W., Hu, Q.: Eca-net: efficient channel attention for deep convolutional neural networks, 2020 ieee. In: CVPR (2020)
36. supervise.ly: Supervisely person dataset. supervise.ly (2018)
37. Wu, Z., Huang, Y., Yu, Y., Wang, L., Tan, T.: Early hierarchical contexts learned by convolutional networks for image segmentation. In: ICPR. IEEE (2014)
38. Gong, K., Liang, X., Li, Y., Chen, Y., Yang, M., Lin, L.: Instance-level human parsing via part grouping network. In: ECCV (2018)
39. Hou, Q., Liu, F.: Context-aware image matting for simultaneous foreground and alpha estimation. In: ICCV (2019)
40. Li, J., Zhang, J., Maybank, S.J., Tao, D.: End-to-end animal image matting. arXiv (2020)
41. Liu, J., Yao, Y., Hou, W., Cui, M., Xie, X., Zhang, C., Hua, X.s.: Boosting semantic human matting with coarse annotations. In: CVPR (2020)
42. Rhemann, C., Rother, C., Wang, J., Gelautz, M., Kohli, P., Rott, P.: A perceptually motivated online benchmark for image matting. In: CVPR. IEEE (2009)