# Boosting Robustness of Image Matting with Context Assembling and Strong Data Augmentation

Yutong Dai[1], Brian Price[2], He Zhang[2], Chunhua Shen[3]
[1]The University of Adelaide [2]Adobe Inc. [3]Zhejiang University

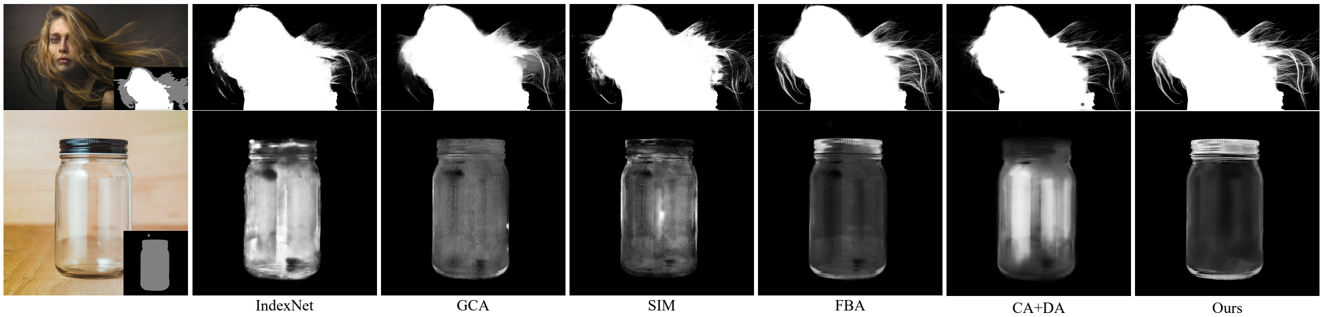

**Figure 1.** Matting results on real-world images. From the second column to right are results of IndexNet [23], GCA [18], SIM [29], FBA [11], CA+data augmentation ($\mathcal{DA}$) [14] and our method, respectively. Note that, all the methods are trained with the DIM [35] dataset (except SIM is trained with the SIMD [29] dataset). They are comparable on benchmark images, while present varying results on real-world images. Our method shows better generalization ability.

## Abstract

*Deep image matting methods have achieved increasingly better results on benchmarks (e.g., Composition-1k/*`alphamatting.com`*). However, the robustness, including robustness to trimaps and generalization to images from different domains, is still under-explored. Although some works propose to either refine the trimaps or adapt the algorithms to real-world images via extra data augmentation, none of them has taken both into consideration, not to mention the significant performance deterioration on benchmarks while using those data augmentation. To fill this gap, we propose an image matting method which achieves higher robustness (RMat) via multilevel context assembling and strong data augmentation targeting matting. Specifically, we first build a strong matting framework by modeling ample global information with transformer blocks in the encoder, and focusing on details in combination with convolution layers as well as a low-level feature assembling attention block in the decoder. Then, based on this strong baseline, we analyze current data augmentation and explore simple but effective strong data augmentation to boost the baseline model and contribute a more generalizable matting method. Compared with previous methods, the proposed method not only achieves state-of-the-art results on the Composition-1k benchmark (11% improvement on SAD and 27% improvement on Grad) with smaller model size, but also shows more robust generalization results on other benchmarks, on real-world images, and also on varying coarse-to-fine trimaps with our extensive experiments.*[1]

## 1. Introduction

Image matting, as a fundamental computer vision task, aims to obtain the high-quality alpha matte of the foreground object given an input image. Mathematically, image matting is formulated as:

$$I_i = \alpha_i F_i + (1 - \alpha_i) B_i \, , \tag{1}$$

where $I_i$, $\alpha_i$, $F_i$ and $B_i$ are observed colour value, alpha value, foreground value and background value of pixel $i$, respectively. It is an ill-posed problem because there are 7 unknowns given 3 equations. With recent success of deep learning, deep matting methods [4, 11, 14, 18, 23, 29, 35] achieve promising results on benchmarks such as Composition-1k [35] and `alphamatting.com` [26]. While

[1] This work was in part done when YD was an intern at Adobe and CS was with The University of Adelaide. CS is the corresponding author. Project page: https://dongdong93.github.io/RMat/

increasingly higher accuracy have been promised on benchmarks, due to the limited training/test data, robustness of these methods is still under explored.

First, robustness to the trimap, the commonly used prior input, is important for a matting algorithm. In real applications, trimaps are labeled by users, with unpredictable precision of unknown regions. However, as shown in Fig. 2, existing matting methods [11, 29] are sensitive to the shape/size of the given trimap so that it requires users' more time to accurately brush the trimap. A main reason why existing methods are sensitive to the precision of trimap is they focus more on detailed cues, where robustness to trimap with varing precision, which relies more on context information, is less cared about. One possible solution is to optimize the trimap to be a more detailed one. This was proposed in [1], where an extra branch was used to generate a more precise trimap. Though multi-task learning is leveraged in this method to adapt the trimap, its context modeling is still limited, which restricts its robustness in applications. Therefore, we wonder *whether it is possible to enhance the context modeling ability (robustness) of a matting algorithm with a simpler and more effective approach.*

Meanwhile, it has been known that deep matting models trained on synthetic data undertake the risk of poor generation to real-world domains [14, 29, 39] (Fig. 1). However, due to the difficulty of obtaining ground-truth annotations for real-world images, only synthetic datasets are available to train the matting algorithms, so some works attempted to narrow the domain gap. For example, [14, 39] leverage extra data augmentation to adapt the models to real-world images, while significant performance degradation on the synthetic benchmark happens at the same time. Although better prediction on real-world images is appreciated, it is desirable that the model can be generalized to broader scenes without sacrificing too much performance on images from one domain such as the benchmark data, because it is hard to confirm which domain a test image comes from, and not to mention that the real-world test images in [14, 39] can only cover a tiny part of real scenes. Therefore, *a model showing better domain generalization ability is in demand.*

Motivated by these demands, we present a more robust matting method (RMat), which achieves higher robustness to diverse trimap precision and better generalization to various domains. In detail, two steps are designed. The first step is to build a strong baseline model with multilevel context assembling. It is implemented by combining transformer blocks with convolution layers, where global context is learned via self-attention modules and local context is emphasized by convolution layers. Considering the uniqueness of matting that needs local context information and original test resolutions to capture details, we explore designs and implementations aiming at this task to build an efficient model. Further, founded on this strong baseline model, we investigate strong data augmentation for matting. We analyze the problems behind current augmentation and propose strong augmentation strategies specifically for matting. Finally, to verify robustness of the model, a series of experiments and visualizations are carried out in comparison with state-of-the-art methods.

In summary, our main contributions are: **1)** A strong matting framework with multilevel context assembling; **2)** Strong augmentation strategies targeting matting; **3)** Designs of experiments and visualizations to verify generalization capability of matting models; **4)** State-of-the-art results on benchmarks (w/ and w/o fitting the training sets), higher robustness to varying trimap precision, and better generalization to real-world images.

## 2. Related Work

**Deep Image Matting.** Before the success of deep learning, conventional matting methods [2, 3, 13, 15, 27, 28, 31] dominated this field by solving Equation (1) using different assumptions such as the local smoothness assumption in [15]. Due to the nature of relying on low-level color cues, their assumptions are easily violated in complex images. To overcome this dilemma, deep matting methods [1, 4, 11, 14, 18, 19, 23, 29, 30, 35, 38] appeared with the development of deep learning.

Among recent state-of-the-art deep matting methods [4, 14, 18, 23], context and dynamic networks are two vital and correlated components. The context includes both global context and local context. Global context intuitively benefits better recognition of the foreground object. It motivates studies on extra context learning modules [14, 18, 21]. It is also one of the reasons behind using ASPP or PPM module in recent methods [4, 11, 23, 29]. Local context instead promotes detail capture by caring about correlations within a local region. The convolution operations or dynamic kernels learned from local regions [4, 23] model local context into the network. On another side, dynamic networks were introduced to matting [4, 18, 23] to enlarge the model capacity. They also benefit the network in combination with the context assembling [23].

Since we aim for a more robust matting method, which needs multilevel context information as well as ample model capacity, the first step is taking both context and dynamic networks into consideration efficiently. We show that it is achievable by combining transformer blocks and convolution layers. We investigate various designs and also provide our insights into them. Considering the limited training data and a relative large capacity of our model, we study strong data augmentation strategies to prevent overfitting the training data and also generalize the model better.

**Domain Generalization.** Domain generalization aims at learning better representations that can be transferred to unseen domains. There are many potential solutions, such

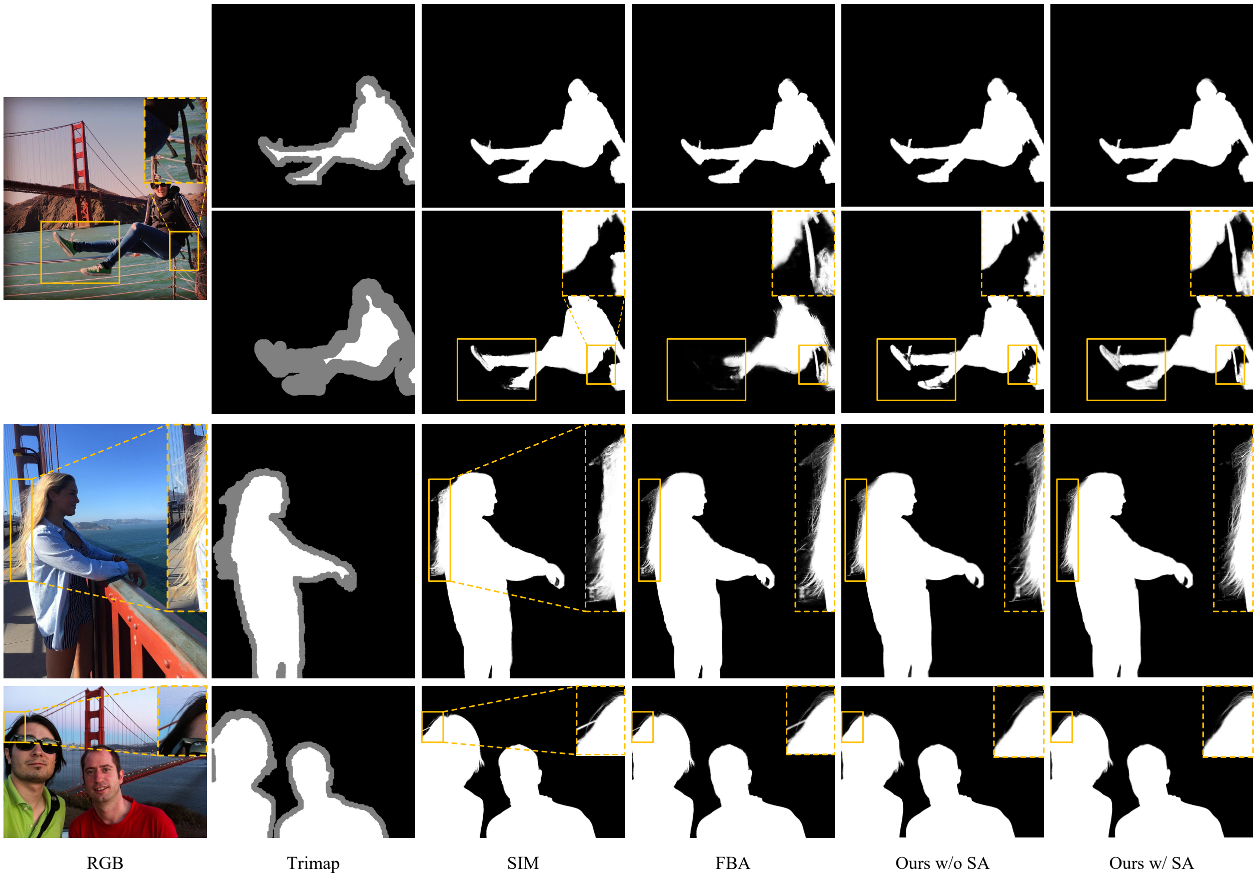

**Figure 2.** Visual results on real-world images showing robustness of methods. The methods in comparison are SIM [29], FBA [11], and our method w/o and w/ Strong Augmentation ($\mathcal{SA}$), respectively. The first two rows are results of the same RGB image with different trimaps. Our methods are more robust to various trimap precision. In the third row, our model using $\mathcal{SA}$ captures better details. The last row presents the benefit of modeling global context: the bridge component on the left top side is apart from the main bridge body, which is recognized as foreground in SIM and FBA. Our method, however, distinguishes it from the foreground human clearly thanks to the context assembling. Dashed lines represent zooming in.

as data augmentation [36, 40, 42], meta learning [10, 16], and adversarial training [12, 24]. In deep matting, since only synthetic training data is available, the trained models usually suffer from poor generalization. They may work well on specific domains, such as the synthetic ones similar to the training set, but show obvious decreasing performance when applied to another domain, as the examples in Fig. 1. Extra data augmentation [14, 39] has been applied to adapt models to real-world images, but they consider limited cases only, such as the resolution gap between the foreground object and background, which may bias the models to those images. We may observe it from Table 5.

Therefore, we move a step further by rethinking strong data augmentation for matting. We first analyze why current extra augmentation deteriorates the benchmark performance, then propose strong augmentation strategies targeting matting. Our goal is to prevent the model overfitting the synthetic training data and help them generalize better to real-world images.

## 3. A Strong Matting Framework with Context Assembling

As noted in conventional sampling-based matting [27, 31] and propagation-based matting [2, 15], both nearby and long-distance pixels contribute to alpha prediction depending on their correlations. In deep models, the correlations are related to context. Existing deep matting methods attempt to model contextual attention [18] or extra context information [14] in the network, while the global context is still under explored. This may limit their performance on complex images such as Fig. 2. In order to assemble multilevel context information, including global context, we build a baseline combining transformer blocks and convolution layers. Designs of the framework are detailed below:

**Encoder Design.** As shown in Fig. 3, the encoder has two branches: a transformer-based branch modeling global context and a convolution-based branch supplementing low-level information for details. Driven by recent vision transformers [8, 32, 34, 41], we use a 32-stride pyramid vision transformer backbone to obtain hierarchical features. Since matting models do inference with various original input resolutions, fixed position embedding is unsuitable for the application. We therefore take advantages of [34], where fixed position embedding is replaced with overlapped convolutions. Due to the large capacity of the transformer blocks, only 2-stride convolution layers are used in the convolution-based branch to form 8-stride. We use two small backbones in [34] (mit-b1 and mit-b2) because of the limited training data for matting. Finally, two encoder architectures with different capacities (E1, E2) are built. BiseNetV2 [37] also uses multiple branches in the encoder for segmentation. Different from our purpose on recovering missing details, [37] aims to combine high-level and low-level information in the encoding stage, to balance accuracy and efficiency.

**Decoder Design.** Various decoder designs [1,4,5,18,23, 39] have been studied in matting models. As the bridge to recover resolutions and capture details, the decoder matters for matting. For instance, previous methods applied feature skip [23], attention-guided refinement [18] or dynamic upsampling [4] to build functional matting decoders aiming at richer details. As the first matting method applying transformers, and considering the importance of the decoder, we investigate an efficient decoder design for our framework.

In general, options for a decoder, in order of decreasing receptive field size, include transformer layers, convolution layers, and MLP layers. Since the transformer branch in the encoder promises a large capacity and global reception field, and to reduce computation as well, we only consider using MLP layers and convolution layers in the basic decoder. These also work well to combining multilevel context information. As a result, several baseline models with different decoders are investigated as listed in Table 2.

**Feature Skip Design.** Skip information from encoder to decoder has been widely adopted in deep matting methods [11, 18, 23, 29]. We categorize the skip information into two sources: **1)** the transformer branch of the encoder (TSkip), where feature maps with different resolutions are skipped to the decoder after MLP/convolution layers. These feature maps transport abundant global information while recovering the resolution. Since the transformer branch starts from $\frac{1}{4}$ resolution, some details may be missing at the initial downsampling stage, so we use **2)** another source of skip information learned in the convolution branch (LSkip).

**Low-Level Feature Assembling Attention Block (LFA) Design.** Inspired by [4, 18], where low-level feature maps assist on refining decoder features, we explore efficient low-level feature assembling using a transformer block. It can be easily extended from the transformer block in the encoder . Let $Attn(Q, K, V)$ denotes the self-attention operation in the transformer block, the feature fusion attention then can be represent by $Attn(f_{low}, f_{low}, f_d)$, where $f_{low}$ is the skipped feature from the encoder and $f_d$ is the feature in the decoder to be refined. Only one LFA block is added after the $\frac{1}{4}$ resolution decoder layer as noted in Fig. 3 in our experiments to restrict the computation. We observe introducing this block further improves the accuracy.

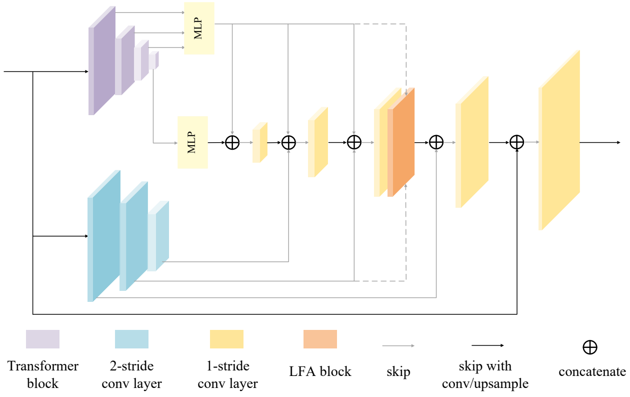

**Figure 3.** Model architecture of our framework.

## 4. Domain Generalization and Data Augmentation for Image Matting

As training images for matting are generated by composition, it inevitably results in generalization problems on real-world images. Also, it is noticed that large-capacity transformer-based models may encounter the overfitting problem [6, 33], especially when the dataset is small. Matting datasets [25, 29, 35], unfortunately, have limited sizes. They usually use only hundreds of foreground images to generate tens of thousands of synthetic training data, so overfitting is a potential problem. To handle this issue, we study strong augmentation ($\mathcal{SA}$) for better generalization.

Targeting the domain gap between synthetic data and real-world images, extra data augmentation ($\mathcal{DA}$) was proposed [14, 39]. It mainly includes Re-JEPG and Gaussian blur. Experiments in [14, 39] show $\mathcal{DA}$ improves results on real-world images, but deteriorates the performance on the benchmark significantly, as shown in Table 4 (CA vs. CA+$\mathcal{DA}$). Therefore, $\mathcal{SA}$ firstly needs to overcome the performance degradation on the benchmark.

### 4.1. Rethinking Domain Generalization and Gaps

**Why Current Extra Data Augmentation ($\mathcal{DA}$) Deteriorates Performance on the Benchmark.** An example of using Re-JPEG and Gaussian blur is shown in Fig. 4a. As observed, some background pixels mix values with foreground pixels after augmentation. Their alpha values therefore change from 0 to a value in range $(0, 1)$. This kind of alpha value blending also exists in transparent regions and in some foreground pixels close to the background. In previous works [14,39], however, the same alpha ground truths

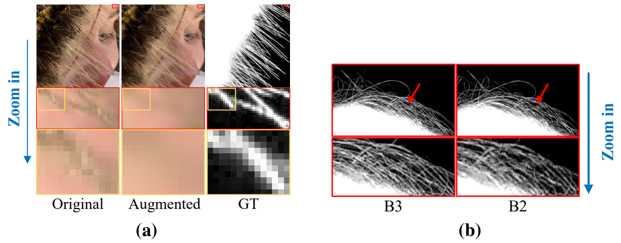

**Figure 4.** (a) A comparison between images before and after $\mathcal{DA}$ . The augmented image loses its structure and does not match the ground truth. (b) A comparison between B3 and B2 (Table 2). The hairs in B2 are blurred.

are used after $\mathcal{DA}$, which violates the matting equation. The image after augmentation loses the structure and does not match the alpha ground truth any more. Using these image-alpha pairs for training could mislead the network to wrong predictions. Hence, we argue it is at least one of the main reasons behind the performance deterioration. To verify this assumption, we carry out a toy-level experiment on DIM. By using Gaussian blur and Re-JPEG with the possibility 0.25 for each as $\mathcal{DA}$, two models are trained: 1) a model trained with $\mathcal{DA}$ using original alpha ground truths; 2) a model trained with $\mathcal{DA}$ using modified alpha ground truths generated by applying the same augmentation as was applied to the RGB image. To ease the difficulty, only L1 alpha prediction loss is applied and fewer training iterations are used. Other training details match the main experiments, as detailed in Section. 5. As present in Table 1, $\mathcal{DA}$ makes the errors higher, and adjusting the ground truth slightly relieves the problem. Hence, it is at least reasonable to claim that modification of ground truth matters for using $\mathcal{DA}$. The correct ground truth, however, is hard to obtain.

**What are the Domain Gaps for Matting.** During the data loading stage, we assume the composition process satisfies the Equation (1). Hence, no matter how the foreground and the background are processed individually, the composited image satisfies this linear equation. Under this assumption, what are the main domain gaps for matting?

**1)** *Complexity of Surrounding Context*. For example, the third example in Fig. 2 is challenging because it is rarely found in the synthetic datasets. **2)** *Source of Images*. In real-world images, foreground and background are from the same source, while this condition is not met in synthetic data. There could be many differences between them: brightness, saturation, sharpness, noise level, etc. **3)** *Manual Operations During Photography and Modifications Made to the Images*. For instance, unfocused boundaries, blurred regions, mosaic generated by image compression, etc. They rarely exist in synthetic datasets.

In this work, we rely on the network to deal with the context domain gap by assembling multilevel context information. As for remaining feature-level gaps, we investigate simple but efficient strong augmentation strategies to generalize the algorithm to real-world images better.

| $\mathcal{DA}$ | Modified $\alpha$ | SAD | Grad |
|---|---|---|---|
| | | 32.65 | 18.13 |
| ✓ | | 35.79 | 20.16 |
| ✓ | ✓ | 34.00 | 20.12 |

**Table 1.** $\mathcal{DA}$ using different ground truths. This toy experiment is trained with a batch size of 32, 45k iterations.

## 4.2. Strong Data Augmentation for Matting

Driven by above analysis, we study $\mathcal{SA}$ for matting. The augmentations are divided into three categories:

1) ***Linear Pixel-Wise Augmentation.*** By pixel-wise, we mean no interpolation happens on the image. Linear denotes the operations that can be linearly represented. It includes linear contrast, brightness adjustment, noise, etc. Only pixel-level changes happen without any information exchange among different pixels and even different channels. If we look at one channel of location $i$ on image $I$, it can be formulated by:

$$\begin{aligned} I'_i &= aI_i + b \\ &= a\left[\alpha F_i + (1-\alpha)\,B_i\right] + \left[\alpha b + (1-\alpha)\,b\right], \\ &= \alpha\,(aF_i + b) + (1-\alpha)\,(aB_i + b) \end{aligned} \tag{2}$$

where $a$ and $b$ are constant parameters for the linear transformation. According to this equation, linear pixel-wise augmentation obeys Equation (1) no matter it happens on $I_i$, $F_i$ or $B_i$. Augmentation on an image can also be viewed as processing the foreground and background individually. It is natural to extend this equation by:

$$I'_i = \alpha\,(aF_i + b) + (1-\alpha)\,(mB_i + n)\ , \tag{3}$$

where different linear transformations happen on the foreground and the background.

2) ***Nonlinear Pixel-Wise Augmentation.*** In the opposite to linear operations, there are also non-linear augmentations, such as gamma correction, hue/saturation adjustment, etc. Due to their nonlinear nature, Equation (1) is violated if the augmentations happen on $I$.

3) ***Region-Wise Augmentation.*** Region-Wise augmentation means operations applied using multiple pixels. For instance, blur, jpeg compression, etc. After interpolations on $I$, Equation (1) is violated, which needs alpha ground truth to be modified accordingly.

Based on this categorization, we propose strong data augmentation strategies:

i) **Augment the Foreground Alone (AF).** Motivated by the random jitter in [18], augmenting foreground alone is effective and obeys the composition equation. The ground truth does not need to be modified no matter which augmentation is taken because it happens before composition.

ii) **Augment the Foreground and the Background Individually (AFB).** It is an extended version of option i) and inspired by Equation (3). Through augmenting foreground

| No. | Encoder | Decoder | TSkip | LSkip | #Params | SAD(↓) | MSE(↓) | Grad(↓) | Conn(↓) |
|---|---|---|---|---|---|---|---|---|---|
| B1 | E1 | MLP | MLP |  | 13.6M | 39.55 | 0.0102 | 24.22 | 37.35 |
| B2 | E1 | Conv | MLP |  | 15.4M | 29.85 | 0.0063 | 13.04 | 25.61 |
| B3 | E1 | Conv | MLP | ✓ | 15.8M | 28.94 | 0.0055 | 12.75 | 24.66 |
| B4 | E1 | Conv | MLPConv | ✓ | 18.2M | 29.99 | 0.0064 | 15.98 | 25.86 |
| B5 | E1 | MLPDW | MLP | ✓ | 14.0M | 29.79 | 0.0062 | 14.18 | 25.78 |
| B6 | E1 | MLPDW | MLPConv | ✓ | 16.1M | 31.45 | 0.0065 | 15.59 | 27.65 |
| B7 | E2 | Conv | MLP | ✓ | 26.8M | 26.11 | 0.0048 | 10.59 | 21.38 |
| B8 | E2 | Conv | MLPConv | ✓ | 29.2M | 25.66 | 0.0045 | 10.40 | 20.90 |
| B9 | E2 | MLPDW | MLP | ✓ | 25.1M | 28.42 | 0.0055 | 12.98 | 24.15 |
| B10 | E2 | MLPDW | MLPConv | ✓ | 27.2M | 30.66 | 0.0064 | 14.26 | 26.88 |

**Table 2.** Ablation study on decoder, feature skip designs on the Composition-1k test set. 'MLPDW' denotes 'MLP+DepthWise Conv'.

| No. | LFA | $l_{lap}$ | $l_g$ | $l_{gp}$ | #Params | SAD | Grad |
|---|---|---|---|---|---|---|---|
| - |  |  |  |  | 15.8M | 28.94 | 12.75 |
| - |  | ✓ |  |  | 15.8M | 27.64 | 10.68 |
| - |  |  | ✓ |  | 15.8M | 27.71 | 10.23 |
| - |  |  |  | ✓ | 15.8M | 27.00 | 9.50 |
| N3 |  | ✓ |  | ✓ | 15.8M | 25.86 | 9.69 |
| - | ✓ |  |  |  | 16.8M | 27.67 | 12.44 |
| M3 | ✓ | ✓ |  | ✓ | 16.8M | 25.70 | 9.50 |
| - |  |  |  |  | 26.8M | 26.11 | 10.59 |
| M7 | ✓ | ✓ |  | ✓ | 27.9M | 25.00 | 9.02 |

**Table 3.** Ablation study on the LFA module and loss functions on the Composition-1k. The upper part(containing **N3/M3**) and the lower part(containing **M7**) are based on **B3** and **B7**, respectively.

and background individually before composition, the linear composition equation is still satisfied, the ground truth alpha matte hence does not need to be modified no matter what augmentation is taken.

iii) **Augment the Composited Image (AC).** This strategy can be further divided into two sub types. If linear pixel-wise augmentation is applied, the composition equation is satisfied as Equation (2). Using other strategies instead violates the equation, where a new ground truth is needed. Due to the expense of obtaining the real ground truth, we propose to generate pseudo label to facilitate the training. The strategy is to predict the pseudo label using the parameters from the last training iteration by rotating or channel-shuffling the input to generate a new training sample. We anticipate this operation promotes the network to learn features of the augmented images without sacrificing accuracy.

# 5. Experiments and Discussions

## 5.1. Implementation Details

Our models are trained on the deep image matting (DIM) dataset [35] only. It contains 431 foreground images in the training set and 50 foreground images in the test set. We generate the training samples using background images randomly selected from MS COCO [20], and use the same rules as [35] to produce test images using background images selected from Pascal VOC [9]. The evaluation metrics are commonly-used Sum of Absolute Differences (SAD), Mean Squared Error (MSE), Gradient (Grad) error, and Connectivity (Conn) error. Implementation of [35] is used.

**Training Details.** Our baseline models follow the dataloader pipeline in [18]. To be specific, the 4-channel input concatenates the RGB image and the trimap. The RGB image is generated on-the-fly through the following basic augmentation: foreground random affine, foreground random combination, random resize, random crop, foreground random jitter, and composition. More details are explained in [18]. $512 \times 512$ patches are finally generated for training. We initialize the weights of the mit backbones using the pre-trained weights on ImageNet-1K [7] from [34] for the transformer branch. Other parameters are initialized with Xavier. The training stage is optimized by AdamW [22] optimizer using initial learning rate $6 \times 10^{-4}$ with cosine decay. The warm up stage takes 1000 iterations. Without specially clarifying, we update parameters for $90k$ iterations with a batch size of 32. Batch size 64 and $120k$ iterations are used for final benchmark results, as detailed in Table 4 and 5.

**Loss Functions.** Our baseline models only use L1 alpha prediction loss and composition loss as [23]. Since other loss functions, such as laplacian loss ($l_{lap}$) and gradient loss ($l_g$), are applied in previous pure convolution-based methods [11,14], here we validate their effects in our framework. Besides using the usual $l_{lap}$ and $l_g$, we define a new gradient loss with gradient penalty($l_{gp}$) for local smoothness:

$$\begin{aligned} l_{gp} = &\left\|\nabla\alpha_x - \nabla\hat{\alpha}_x\right\|_1 + \left\|\nabla\alpha_y - \nabla\hat{\alpha}_y\right\|_1 \\ &+ \lambda\left(\left\|\nabla\alpha_x\right\|_1 + \left\|\nabla\alpha_y\right\|_1\right) \end{aligned}, \tag{4}$$

where $\lambda$ is set as 0.01 in our experiments.

## 5.2. Results on the Deep Image Matting Dataset

**Ablation Study on Model Architecture.** Based on the two-branch encoder, here we investigate designs of the decoder, the skip layer and the additional attention module.

According to the results in Table 2 and Table 3, we draw the following observations: **1)** Compared with the MLP layer and the MLPDW layer, the Conv layer suits the decoder of matting better (B1 vs. B2, B5 vs.B3, B6 vs. B4, B9

| Method | SAD | MSE | Grad | Conn | # Params |
|---|---|---|---|---|---|
| CF [15] | 168.1 | 0.091 | 126.9 | 167.9 | - |
| KNN [2] | 175.4 | 0.103 | 124.1 | 176.4 | - |
| DIM [35] | 50.4 | 0.014 | 31.0 | 50.8 | $>$ 130.55M |
| IndexNet [23] | 45.8 | 0.013 | 25.9 | 43.7 | 8.15M |
| CA [14] | 35.8 | 0.0082 | 17.3 | 33.2 | 107.5M |
| CA+$\mathcal{DA}$ [14] | 71.3 | 0.0236 | 38.8 | 72.0 | 107.5M |
| GCA [18] | 35.28 | 0.0091 | 16.9 | 32.5 | 25.27M |
| A$^2$U [4] | 32.15 | 0.0082 | 16.39 | 29.25 | 8.09M |
| SIM [29] | 28.0 | 0.0058 | 10.8 | 24.8 | 70.16M |
| FBA [11] | 26.4 | 0.0054 | 10.6 | 21.5 | 34.69M |
| FBA+TTA [11] | 25.8 | 0.0052 | 10.6 | 20.8 | 34.69M |
| M3$^{\ddagger}$ | 23.98 | 0.0042 | 8.54 | 18.88 | 16.8M |
| M7$^{\ddagger}$ | **22.87** | **0.0039** | **7.74** | **17.84** | 27.9M |

**Table 4.** Benchmark results on the Composition-1k test set. The best performance is in boldface. ‡ denotes training with a batch size of 64, 120k iterations using our $\mathcal{SA}$.

vs.B7, B10 vs. B8); **2)** Skipped information from the transformer branch to the decoder can be efficiently achieved by a simple MLP layer (B3 vs. B4, B5 vs. B6, B9 vs. B10); **3)** Low-level skip fusion is important for recovering details (B2 vs. B3, also see Fig. 4b); **4)** Additional low-level feature assembling attention module further improves the results (Table 3); **5)** The advantage of larger backbone is gradually weakened with improvement on the architecture and loss functions (B3→M3 vs. B7→M7 in Table 3).

**Ablation Study on Loss Functions.** Here we justify effectiveness of laplacian loss ($l_{lap}$), gradient loss ($l_g$) and the proposed gradient loss with gradient penalty ($l_{gp}$) in our framework. Results are reported in Table 3. Compared with using only basic loss functions, $l_{lap}$, $l_g$, and $l_{gp}$ all reduce the errors, and our $l_{gp}$ works better than normal $l_g$. Combining $l_{lap}$ and $l_{gp}$ together builds the best prediction. We use these two losses in the following experiments.

**Comparison with State of the Art.** Benchmark results on the Composition-1k are list in Table 4. Our models achieve significantly better results on all the metrics. Compared with a currently top-performing FBA+TTA [11] model, our method (M7$^{\ddagger}$) gains 11% improvement on SAD and 27% improvement on Grad without any augmentation on the test images. Moreover, our method is more robust to trimap precision, as shown in Fig. 2 and 5, and the detailed evaluations in the supplement.

### 5.3. Generalization on Various Benchmarks

To verify the generalization ability of matting methods to unseen domains, comparison experiments are carried out in Table 5. Specifically, we test models trained merely with the DIM dataset [35] on several different benchmarks without fitting on their training set (except that SIM [29] is trained on SIMD [29], which has 763 foregrounds, 332 more foregrounds than DIM). The test benchmarks include Distinction-646 [25], SIMD [29], and AIM-500 [17]. Distinction-646 and SIMD are synthetic benchmarks, and AIM-500 is a real-world one but with simple scenes, so none of them alone is perfect for measuring generalization ability of a model. However, since they contain images from different sources and various domains, it is at least reasonable to combine them together to see how algorithms perform quantitatively on all of them, and the overall results should reflect how well a model can adapt to diverse images to some extent. Note that, since SIMD has only provided alpha ground truths and foregrounds until the submission, we generate the test set following the rule of [35] and name it as SIMD$_{our}$. To ensure all the methods can be test on a normal modern graphic card, we restrict the maximum length of the test images in SIMD$_{our}$ by 2000.

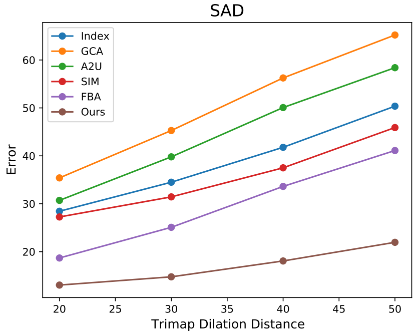

**Figure 5.** Robustness to trimap precision on AIM-500

**Ablation Study on Strong Data Augmentation.** Here we investigate the $\mathcal{SA}$ strategies. We either use AF, AFB alone, or combine them with AC. Specifically, the linear pixel-wise augmentations include: *linear contrast, brightness adjustment, channel inversion/shuffling, gaussian/poisson noise, random dropout, cloud, snow, multiply, salt and pepper*; the nonlinear pixel-wise augmentations include: *gamma contrast, hue and saturation add on, histogram equalization*; and region-wise augmentations consist of *gaussian blur and jpeg compression*. If AF or AFB is applied alone, we set the possibility as 0.5 and keep the ground truths unmodified; if they are combined, possibility of each is changed to 0.25; further, if AC is added on, we set its possibility as 0.1 when AF and AFB do not happen, and generate pseudo labels for the augmented samples as explained in Section 4 when needed. More details are in supplement. As shown in Table 5, both AF or AFB improve the AIM-500 results, especially AFB, but they also make errors on Distinction-646, SIMD$_{our}$, and Composition-1k (supplement) slightly higher. AF is more stable on synthetic benchmarks compared with AFB. Hence, we carry out 'AF+AFB'. It averages the effects of AF and AFB. Based on 'AF+AFB', AC further improves results on AIM-500 and keeps results on other synthetic benchmarks comparable, so we use 'AF+AFB+AC' as the final $\mathcal{SA}$.

**Comparison with State of the Art.** Compared with other methods, our M3$^{\ddagger}$ ans M7$^{\ddagger}$ models achieve best per-

| Method | | | | Distinction-646 [25] | | | | SIMD$_{our}$ [29] | | | | AIM-500 [17] | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | | | SAD | MSE | Grad | Conn | SAD | MSE | Grad | Conn | SAD | MSE | Grad | Conn |
| IndexNet$^*$ [23] | | | | 42.64 | 0.0256 | 40.17 | 42.76 | 92.45 | 0.0388 | 45.85 | 93.14 | 28.49 | 0.0288 | 18.15 | 27.95 |
| CA$^*$ [14] | | | | 49.07 | 0.0557 | 114.77 | 48.27 | 79.46 | 0.0291 | 51.03 | 77.88 | 26.33 | 0.0266 | 18.89 | 25.05 |
| CA+$\mathcal{DA}^*$ [14] | | | | 46.03 | 0.0356 | 55.45 | 46.18 | 102.97 | 0.0469 | 74.39 | 103.52 | 32.15 | 0.0388 | 30.25 | 31.00 |
| GCA$^*$ [18] | | | | 31.00 | 0.0171 | 21.19 | 29.62 | 75.81 | 0.0271 | 40.57 | 74.45 | 35.10 | 0.0389 | 25.67 | 35.48 |
| A$^2$U$^*$ [4] | | | | 28.74 | 0.0143 | 17.42 | 27.62 | 68.70 | 0.0268 | 39.00 | 66.76 | 30.38 | 0.0307 | 22.60 | 30.69 |
| SIM$^*$ [29] | | | | **22.68** | 0.0137 | 20.11 | 21.03 | **37.07** | 0.0099 | 22.29 | 33.30 | 27.05 | 0.0311 | 23.68 | 27.08 |
| FBA$^*$ [11] | | | | 30.70 | 0.0150 | 18.89 | 29.65 | 41.55 | 0.0109 | 23.21 | 35.07 | 19.05 | 0.0162 | 11.42 | 18.30 |
| | AF | AFB | AC | | | | | | | | | | | | |
| N3 | | | | 25.86 | 0.0105 | 13.25 | 24.11 | 41.83 | 0.0099 | 22.19 | 34.98 | 16.08 | 0.0122 | 11.69 | 15.55 |
| N3 | ✓ | | | 25.18 | 0.0108 | 13.45 | 23.35 | 44.14 | 0.0122 | 23.46 | 37.96 | 16.38 | 0.0112 | 10.75 | 16.00 |
| N3 | | ✓ | | 27.02 | 0.0123 | 14.44 | 25.34 | 44.46 | 0.0113 | 23.93 | 38.30 | **13.46** | 0.0097 | 9.98 | **12.47** |
| N3 | ✓ | ✓ | | 26.46 | 0.0117 | 14.18 | 24.87 | 42.89 | 0.0110 | 23.37 | 36.05 | 14.63 | 0.0108 | 10.46 | 13.75 |
| N3 | ✓ | ✓ | ✓ | 26.32 | 0.0119 | 14.40 | 24.56 | 43.49 | 0.0109 | 23.96 | 36.84 | 14.18 | 0.0093 | 9.36 | 13.61 |
| M3$^‡$ | ✓ | ✓ | ✓ | 23.67 | 0.0100 | 11.30 | 21.37 | 42.68 | 0.0121 | 21.20 | 35.84 | 13.68 | **0.0091** | 9.36 | 13.06 |
| M7$^‡$ | ✓ | ✓ | ✓ | 23.25 | **0.0097** | **11.09** | **21.00** | 37.31 | **0.0090** | **20.00** | **30.10** | 13.97 | 0.0094 | **8.89** | 13.21 |

**Table 5.** Generalization results on the Distinction-646, SIMD$_{our}$ and AIM benchmarks. The best performance is in boldface. The second is underlined. **All the models are merely trained with the DIM training set** (431 **foregrounds), except that SIM is trained with the SIMD training set** (736 **foregrounds), so we set SIM's results on SIMD as blue color to represent SIM is trained with this dataset.** $^*$ means using the officially provided model. ‡ denotes training with a batch size of 64, 120k iterations using our $\mathcal{SA}$ (AF+AFB+AC).

formance on all three benchmarks in Table 5, especially with MSE and Grad metrics. Note that, $\mathcal{DA}$ in [14] degrades its performance on Composition-1k (Table 4), SIMD$_{our}$ and AIM-500 significantly, even though AIM-500 is a real-world benchmark. Our $\mathcal{SA}$ instead promises comparable results on the synthetic benchmarks and much better results on the real-world benchmark. The advantages of $\mathcal{SA}$ against $\mathcal{DA}$ can also be noticed from visual examples in Fig. 1 and 6. Moreover, when longer training time is taken, stable improvements are observed. The examples in Fig. 2 and 6 further verify the effectiveness of our model and $\mathcal{SA}$. More visual results on real-world images and benchmarks are shown in the supplement.

### 5.4. Results on the `alphamatting.com`

We show the results of M7$^‡$ on the `alphamatting.com` [26] online benchmark in Table. 6. Note that, SIM is trained with the SIMD training set, which has 736 foregrounds in the training set, while DIM only has 431 foregrounds in the training set; GCA and A$^2$U retrain their models with the whole DIM dataset (including both training set and test set). Our result is directly reported from M7$^‡$ without using extra data or fine-tuning the model, but it still achieves top-performing ranks, especially on MSE and Grad. See the full table in the supplement.

| Method | MSE | | | | Grad | | | |
|---|---|---|---|---|---|---|---|---|
| | overall | S | L | U | overall | S | L | U |
| Ours-M7$^‡$ | **6.8** | **5.9** | **5.5** | 9.1 | **4.7** | **4.8** | **3.8** | **5.5** |
| SIM [29] | 7 | 8.1 | **5.5** | **7.4** | 6.9 | 8.5 | 5.9 | 6.5 |
| A$^2$U [4] | 15.5 | 13 | 12.6 | 20.8 | 12.3 | 11.3 | 9.4 | 16.1 |
| GCA [18] | 15.3 | 15.1 | 14.5 | 16.4 | 13.7 | 13.6 | 12.5 | 15 |
| CA [14] | 17.6 | 20.9 | 18.6 | 13.3 | 14.6 | 15.8 | 15.5 | 12.6 |
| IndexNet [23] | 22.9 | 25.3 | 21.5 | 22 | 18.6 | 17.3 | 17.3 | 21.4 |

**Table 6.** Results on the `alphamatting.com` online benchmark.

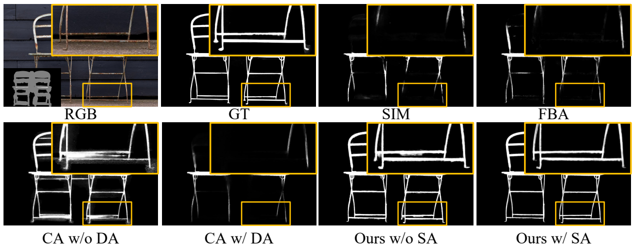

**Figure 6.** Visual results on the AIM-500 benchmark. The methods in comparison are SIM [29], FBA [11], CA w/o $\mathcal{DA}$ [14], CA w/ $\mathcal{DA}$ [14], Ours w/o $\mathcal{SA}$, Ours w/ $\mathcal{SA}$.

## 6. Conclusion

We propose RMat, a matting method showing higher robustness to various trimap precision and images from different domains. The efforts behind this include a new matting framework and strong augmentation strategies specifically designed for matting. We first build the strong baseline by assembling multilevel context information, then analyse the problems behind current data augmentation and design strong augmentation strategies. To verify generalization capability of the model, we not only show visual results on real-world images, but also design a series of evaluation experiments on several benchmarks without fitting their training sets. Our method achieves state-of-the-art results on all the benchmarks. We hope our work opens up more possibilities for future works on deep matting.

**Limitations** There are still many challenging cases, such as strong light in the background, cannot be handled by our method. We show failure cases in the supplement. To tackle those cases, we may need to better learn the structure of the foreground objects. We leave it as future work.

## References

[1] Shaofan Cai, Xiaoshuai Zhang, Haoqiang Fan, Haibin Huang, Jiangyu Liu, Jiaming Liu, Jiaying Liu, Jue Wang, and Jian Sun. Disentangled image matting. In *Int. Conf. Comput. Vis.*, pages 8819–8828, 2019. 2, 4

[2] Qifeng Chen, Dingzeyu Li, and Chi-Keung Tang. Knn matting. *IEEE Trans. Pattern Anal. Mach. Intell.*, 35(9):2175–2188, 2013. 2, 3, 7

[3] Yung-Yu Chuang, Brian Curless, David H Salesin, and Richard Szeliski. A bayesian approach to digital matting. In *IEEE Conf. Comput. Vis. Pattern Recog.*, volume 2, pages II–II. IEEE, 2001. 2

[4] Yutong Dai, Hao Lu, and Chunhua Shen. Learning affinity-aware upsampling for deep image matting. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 6841–6850, 2021. 1, 2, 4, 7, 8

[5] Yutong Dai, Hao Lu, and Chunhua Shen. Towards lightweight portrait matting via parameter sharing. In *Comput. Graph. Forum*, volume 40, pages 151–164. Wiley Online Library, 2021. 4

[6] Stéphane d'Ascoli, Hugo Touvron, Matthew Leavitt, Ari Morcos, Giulio Biroli, and Levent Sagun. Convit: Improving vision transformers with soft convolutional inductive biases. *arXiv preprint arXiv:2103.10697*, 2021. 4

[7] Jia Deng, Wei Dong, Richard Socher, Li-Jia Li, Kai Li, and Li Fei-Fei. Imagenet: A large-scale hierarchical image database. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 248–255. Ieee, 2009. 6

[8] Alexey Dosovitskiy, Lucas Beyer, Alexander Kolesnikov, Dirk Weissenborn, Xiaohua Zhai, Thomas Unterthiner, Mostafa Dehghani, Matthias Minderer, Georg Heigold, Sylvain Gelly, et al. An image is worth 16x16 words: Transformers for image recognition at scale. *Int. Conf. Learn. Represent.*, 2020. 4

[9] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *Int. J. Comput. Vis.*, 88(2):303–338, 2010. 6

[10] Chelsea Finn, Pieter Abbeel, and Sergey Levine. Model-agnostic meta-learning for fast adaptation of deep networks. In *Int. Conf. Mach. Learn.*, pages 1126–1135. PMLR, 2017. 3

[11] Marco Forte and François Pitié. $f$, $b$, alpha matting. *CoRR*, 2020. 1, 2, 3, 4, 6, 7, 8

[12] Yaroslav Ganin, Evgeniya Ustinova, Hana Ajakan, Pascal Germain, Hugo Larochelle, François Laviolette, Mario Marchand, and Victor Lempitsky. Domain-adversarial training of neural networks. *J. Mach. Learn. Research.*, 17(1):2096–2030, 2016. 3

[13] Kaiming He, Christoph Rhemann, Carsten Rother, Xiaoou Tang, and Jian Sun. A global sampling method for alpha matting. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 2049–2056. IEEE, 2011. 2

[14] Qiqi Hou and Feng Liu. Context-aware image matting for simultaneous foreground and alpha estimation. In *Int. Conf. Comput. Vis.*, pages 4130–4139, 2019. 1, 2, 3, 4, 6, 7, 8

[15] Anat Levin, Dani Lischinski, and Yair Weiss. A closed-form solution to natural image matting. *IEEE Trans. Pattern Anal. Mach. Intell.*, 30(2):228–242, 2007. 2, 3, 7

[16] Da Li, Yongxin Yang, Yi-Zhe Song, and Timothy M Hospedales. Learning to generalize: Meta-learning for domain generalization. In *Proc. AAAI Conf. Artificial Intell.*, 2018. 3

[17] Jizhizi Li, Jing Zhang, and Dacheng Tao. Deep automatic natural image matting. *Int. Joint Conf. Artificial Intell.*, 2021. 7, 8

[18] Yaoyi Li and Hongtao Lu. Natural image matting via guided contextual attention. In *Proc. AAAI Conf. Artificial Intell.*, volume 34, pages 11450–11457, 2020. 1, 2, 3, 4, 5, 6, 7, 8

[19] Shanchuan Lin, Andrey Ryabtsev, Soumyadip Sengupta, Brian L Curless, Steven M Seitz, and Ira Kemelmacher-Shlizerman. Real-time high-resolution background matting. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 8762–8771, 2021. 2

[20] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *Eur. Conf. Comput. Vis.*, pages 740–755. Springer, 2014. 6

[21] Yuhao Liu, Jiake Xie, Xiao Shi, Yu Qiao, Yujie Huang, Yong Tang, and Xin Yang. Tripartite information mining and integration for image matting. In *Int. Conf. Comput. Vis.*, pages 7555–7564, 2021. 2

[22] Ilya Loshchilov and Frank Hutter. Decoupled weight decay regularization. *arXiv preprint arXiv:1711.05101*, 2017. 6

[23] Hao Lu, Yutong Dai, Chunhua Shen, and Songcen Xu. Indices matter: Learning to index for deep image matting. In *Int. Conf. Comput. Vis.*, pages 3266–3275, 2019. 1, 2, 4, 6, 7, 8

[24] Yawei Luo, Liang Zheng, Tao Guan, Junqing Yu, and Yi Yang. Taking a closer look at domain shift: Category-level adversaries for semantics consistent domain adaptation. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 2507–2516, 2019. 3

[25] Yu Qiao, Yuhao Liu, Xin Yang, Dongsheng Zhou, Mingliang Xu, Qiang Zhang, and Xiaopeng Wei. Attention-guided hierarchical structure aggregation for image matting. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 13676–13685, 2020. 4, 7, 8

[26] Christoph Rhemann, Carsten Rother, Jue Wang, Margrit Gelautz, Pushmeet Kohli, and Pamela Rott. A perceptually motivated online benchmark for image matting. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 1826–1833. IEEE, 2009. 1, 8

[27] Ehsan Shahrian, Deepu Rajan, Brian Price, and Scott Cohen. Improving image matting using comprehensive sampling sets. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 636–643, 2013. 2, 3

[28] Jian Sun, Jiaya Jia, Chi-Keung Tang, and Heung-Yeung Shum. Poisson matting. In *ACM SIGGRAPH*, pages 315–321. 2004. 2

[29] Yanan Sun, Chi-Keung Tang, and Yu-Wing Tai. Semantic image matting. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 11120–11129, 2021. 1, 2, 3, 4, 7, 8

[30] Jingwei Tang, Yagiz Aksoy, Cengiz Oztireli, Markus Gross, and Tunc Ozan Aydin. Learning-based sampling for natural image matting. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 3055–3063, 2019. 2

[31] Jue Wang and Michael F Cohen. Optimized color sampling for robust matting. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 1–8. IEEE, 2007. 2, 3

[32] Wenhai Wang, Enze Xie, Xiang Li, Deng-Ping Fan, Kaitao Song, Ding Liang, Tong Lu, Ping Luo, and Ling Shao. Pyramid vision transformer: A versatile backbone for dense prediction without convolutions. *Int. Conf. Comput. Vis.*, 2021. 4

[33] Chuhan Wu, Fangzhao Wu, Tao Qi, and Yongfeng Huang. Fastformer: Additive attention can be all you need. *arXiv preprint arXiv:2108.09084*, 2021. 4

[34] Enze Xie, Wenhai Wang, Zhiding Yu, Anima Anandkumar, Jose M Alvarez, and Ping Luo. Segformer: Simple and efficient design for semantic segmentation with transformers. *Adv. Neural Inform. Process. Syst.*, 2021. 4, 6

[35] Ning Xu, Brian Price, Scott Cohen, and Thomas Huang. Deep image matting. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 2970–2979, 2017. 1, 2, 4, 6, 7

[36] Zhenlin Xu, Deyi Liu, Junlin Yang, Colin Raffel, and Marc Niethammer. Robust and generalizable visual representation learning via random convolutions. *Int. Conf. Learn. Represent.*, 2020. 3

[37] Changqian Yu, Changxin Gao, Jingbo Wang, Gang Yu, Chunhua Shen, and Nong Sang. Bisenet v2: Bilateral network with guided aggregation for real-time semantic segmentation. *Int. J. Comput. Vis.*, 129(11):3051–3068, 2021. 4

[38] Haichao Yu, Ning Xu, Zilong Huang, Yuqian Zhou, and Humphrey Shi. High-resolution deep image matting. *Proc. AAAI Conf. Artificial Intell.*, 2020. 2

[39] Qihang Yu, Jianming Zhang, He Zhang, Yilin Wang, Zhe Lin, Ning Xu, Yutong Bai, and Alan Yuille. Mask guided matting via progressive refinement network. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 1154–1163, 2021. 2, 3, 4

[40] Hongyi Zhang, Moustapha Cisse, Yann N Dauphin, and David Lopez-Paz. mixup: Beyond empirical risk minimization. *Int. Conf. Learn. Represent.*, 2017. 3

[41] Sixiao Zheng, Jiachen Lu, Hengshuang Zhao, Xiatian Zhu, Zekun Luo, Yabiao Wang, Yanwei Fu, Jianfeng Feng, Tao Xiang, Philip HS Torr, et al. Rethinking semantic segmentation from a sequence-to-sequence perspective with transformers. In *IEEE Conf. Comput. Vis. Pattern Recog.*, pages 6881–6890, 2021. 4

[42] Zhun Zhong, Liang Zheng, Guoliang Kang, Shaozi Li, and Yi Yang. Random erasing data augmentation. In *Proc. AAAI Conf. Artificial Intell.*, volume 34, pages 13001–13008, 2020. 3