Unsupervised Deep Background Matting Using Deep Matte Prior

Yong Xu, Baoling Liu, Yuhui Quan* and Hui Ji

***Abstract*—Background matting is a recently developed image matting approach, with applications to image and video editing. It refers to estimating both the alpha matte and foreground from a pair of images with and without foreground objects. Recent work has applied deep learning to background matting, with very promising performance achieved. However, existing deep models are supervised which require a large dataset with ground truth alpha mattes for training. To avoid the cost of data collection and possible bias in training data, this paper proposes a dataset-free unsupervised deep learning-based approach for background matting. Observing that the local smoothness of alpha matte can be well characterized by the untrained network prior called deep matte prior, we model the foreground and alpha matte using the priors encoded by two generative convolutional neural networks. To avoid possible overfitting during unsupervised learning, a two-stage learning scheme is developed which contains projection-based training and Bayesian post refinement. An alpha-matte-driven initialization scheme is also developed for performance boost. Even without calling external training data, the proposed approach provides very competitive performance to the supervised learning-based methods in the experiments.**

***Index Terms*—Background Matting; Deep Prior; Image Matting; Unsupervised Learning**

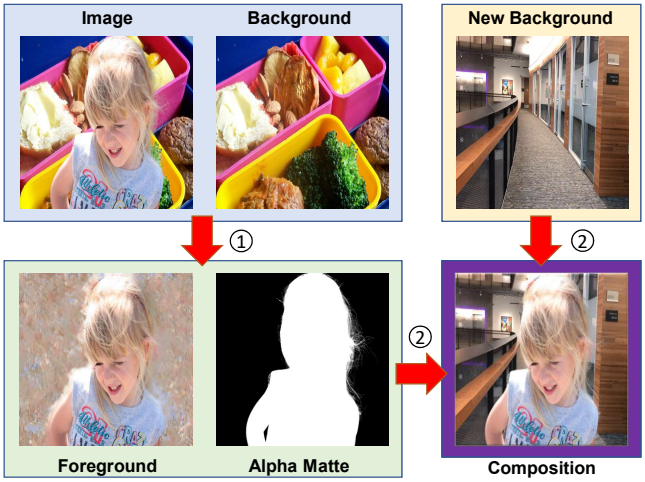


Fig. 1: Given a pair of images with and without foreground, background matting predicts both the alpha matte and foreground which are used to composite a new image with another background. The prediction results in the figure are obtained by the proposed unsupervised learning method which has no requisite on training data.

I. Introduction

IMAGE matting is an important technique in image editing and film making. It also has applications to other image processing tasks such as color correction [1] and all-in-focus synthetic aperture imaging [2]. In image matting, an image $\bar{\boldsymbol{I}}$ is modeled by the composite of a foreground layer $\boldsymbol{F}$ and a background layer $\boldsymbol{B}$ as follows:

$$\bar{\boldsymbol{I}} = \boldsymbol{\alpha} \odot \boldsymbol{F} + (1 - \boldsymbol{\alpha}) \odot \boldsymbol{B}, \quad \boldsymbol{\alpha}(j) \in [0, 1], \ \forall j, \qquad (1)$$

where $\boldsymbol{\alpha}$ is the so-called *alpha matte* that represents the opacity of the foreground color for each pixel, $\odot$ denotes the element-wise multiplication operator, and linear indexing is applied to $\bar{\boldsymbol{I}}, \boldsymbol{F}, \boldsymbol{\alpha}$. The task of image matting is to extract the foreground layer as well as the alpha matte from a given image. The extracted alpha matte can be used to combine the extracted foreground on a different background to produce a new plausible image.

Notice that for each image pixel, there are three unknowns in the matting problem (1). Therefore, it is a highly ill-posed problem, and the main concern in image matting is about how to resolve the solution ambiguity. Most existing methods introduce additional external information (*e.g.* a trimap input by the user) or priors (*e.g.* a green-screen environment) for reducing the solution ambiguity. The trimap-based methods require manual annotations from users, which can be labor-intensive in the batch processing of many images. The methods assuming a green-screen environment has quite limited applicability, as such an assumption does not hold true for many scenarios, especially in urban areas.

In recent years, there has been an increasing interest on how to automate the task of image matting and make it applicable to most scenarios. One such work is the so-called *background matting* [3], which is trimap-free. To achieve it, the background matting approach takes a pair of images as the input: one with foreground objects and the other without foreground objects. In such a configuration, the problem is then formulated as

$$\bar{\boldsymbol{I}} = \boldsymbol{\alpha} \odot \boldsymbol{F} + (1 - \boldsymbol{\alpha}) \odot \bar{\boldsymbol{B}}, \quad \boldsymbol{\alpha}(j) \in [0, 1], \ \forall j, \qquad (2)$$

where $\bar{\boldsymbol{I}}$ denotes the image with foreground objects and $\bar{\boldsymbol{B}}$ denotes its counterpart without foreground objects. The aim

Yong Xu and Baoling Liu are with School of Computer Science and Engineering at South China University of Technology, China, and with Peng Cheng Laboratory, Shenzhen, China. (email: yxu@scut.edu.cn,csblliu@foxmail.com)

Yuhui Quan is with School of Computer Science and Engineering at South China University of Technology, China, as well as with Guangdong Provincial Key Laboratory of Computational Intelligence and Cyberspace Information, China. (email: csyhquan@scut.edu.cn)

Hui Ji is with Department of Mathematics at National University of Singapore, Singapore 119076. (email: matjh@nus.edu.sg)

*Corresponding author: Yuhui Quan (email: csyhquan@scut.edu.cn).

This work was supported in part by National Natural Science Foundation of China under Grants 61872151 and 62072188, in part by Science and Technology Program of Guangdong Province under Grant 2019A050510010, in part by CCF-Tencent Open Fund 2020, and in part by Singapore MOE Academic Research Funds R-146-000-315-114 and MOE2017-T2-2-156.

of the problem is then about estimating two unknowns: the foreground $\boldsymbol{F}$ and the alpha matte $\boldsymbol{\alpha}$.

In comparison to trimap-based image matting, background matting only requires the user to take one more photo without the subject of interest, which takes much less effort than manually creating a trimap. Such a usage convenience is much more appealing for batch image processing, as well as for processing multiple frames with a fixed background but moving/different objects. In addition, the applicability of background matting is much wider than that of green-screen environment-based matting. See Fig. 1 for a demonstration of background matting.

A. Motivations

Background matting is also a challenging ill-posed problem, as there are two unknowns for each image pixel. In [3], [4], supervised deep learning methods are developed for background matting. These studies collected large datasets with image triplets: images with foreground objects, images without foreground objects, and manually-crafted alpha mattes. Then deep networks are trained over such datasets, with state-of-the-art results achieved.

While such supervised learning-based methods provided very promising performance, the prerequisite on a large number of image triplets makes them very costly to implement in practice. Recall that the alpha mattes of the training samples require manual annotations, which can be very time-consuming to achieve high precision. Also, if the training samples do not sufficiently cover the variations of foregrounds and alpha mattes, the trained model can be biased and does not generalize well on those images not very related to the training data. This is likely to occur for non-portrait image matting, as the foreground objects can be arbitrary in terms of types and appearance. In addition, the training samples for matting are often generated and augmented by synthesizing images from different background images and pairs of foreground and alpha matte, whereas this may generate images with unreal scenarios (*e.g.* people on a lake), bringing undesired patterns and misleading semantics to the training data.

Motivated by the cost and possible bias introduced by the prerequisite on training datasets of a supervised deep learning-based method, this paper aims at developing an unsupervised deep learning-based approach for background matting which does not require any external training sample. In other words, no ground truth alpha matte will be called for training, and the proposed approach only takes an image pair $(\bar{\boldsymbol{I}}, \bar{\boldsymbol{B}})$ as the input and directly learns to estimate the corresponding $\boldsymbol{F}$ and $\boldsymbol{\alpha}$. Such a training-data-free approach has its great benefits in practice. In addition, it can also be used for generating foreground objects and alpha mattes that are close to the ground truths for boosting the supervised training.

B. Main Ideas

As the network is not exposed to any ground truth alpha matte, the development of an unsupervised deep learning-based approach for background matting is technically more challenging than developing the supervised ones. In this paper, we tackle challenge by leveraging the generative prior encoded by an untrained convolutional neural network (CNN). It is inspired by the deep image prior [5] which empirically showed that the regular image structures of an natural image can be more efficiently approximated by a CNN architecture than the random patterns. We observed that this also applies to alpha matte, *i.e.*, the generative CNN for approximating the alpha matte tends to output a structured alpha matte with local smoothness. This forms a very useful prior for image matting, as local smoothness is one important property of alpha mattes, and we refer to such a prior as the *deep matte prior*.

Based on the deep image prior and deep matte prior, we adopt two generative CNNs to model the foreground and alpha matte respectively, and the predictions from these two CNNs will then be used to reconstruct the background given by the input image pair. While the deep image/matte priors partially addressed the possible overfitting arising from the solution ambiguity, a straightforward training by the standard procedure will still suffer from the overfitting to undesired solutions, *i.e.* the learned CNNs may output foreground and alpha matte with not very high quality. Therefore, we develop a two-stage scheme for effectively training such a double generative CNN architecture. In the 1^{st} stage, the CNNs are trained with a projection strategy to reduce possible overfitting. In the 2^{nd} stage, a Bayesian post-refinement is introduced to address the issue of training asynchronization, which leads to better performance. In addition, since the optimization over two deep CNNs is highly sensitive to network initialization, an initialization scheme driven by the estimated alpha matte is developed for improvement.

C. Contributions and Significance

To summarize, this work proposes a training-data-free unsupervised deep learning approach for background matting, with the contributions listed below:

- To the best of our knowledge, this work is the first available unsupervised deep learning approach for background matting. While requiring no external training data, it can outperform state-of-the-art supervised learning-based methods (*e.g.* [3]). Such a dataset-free approach can certainly see its great value in the practice, in terms of both matting performance and usage convenience.
- The proposed approach leverages the deep image prior and deep matte prior for the implicit regularizations of the foreground and alpha matte. Moreover, the possible overfitting is effectively handled by a two-stage training scheme, which addresses the solution ambiguity and leads to good performance. These techniques have potentials in other layer separation and image decomposition tasks.

While the proposed approach is free from external training data, we emphasize that it is not against dataset-based learning methods. Indeed, it provides a complement that addresses the case where external training data is biased, insufficient or unavailable. In such cases, the proposed approach can still learn a good prior on the foreground and alpha matte from the test image itself for matting. Further, similar to that deep image prior [5] has invoked many studies on unsupervised learning

for image recovery, our work can also inspire the development of other unsupervised learning methods for image matting.

D. Organization and Notations

The rest of this paper is organized as follows. Section II is devoted to literature review. Section III presents the details of the proposed approach. Section IV conducts the experimental evaluation. Section V concludes the paper.

Through the paper, unless specified, linear indexing is used for images, image patches and maps. Calligraphic letters are used for operators or distributions, boldfaced letters for vectors or matrices, and normal letters for scalars.

II. Related Work

While there have been many studies on image matting, background matting is a new topic in its infancy stage, with only a little work on it. This section first has a brief review on existing image matting methods, followed by a detailed review on background matting. In addition, since our proposed approach is based on untrained network priors, we also have a brief review on their related work in image processing.

A. Traditional Image Matting

Most existing image matting methods handle solution ambiguity when solving (1) via introducing additional information or constraints, such as a trimap (*e.g.* [6], [7], [8], [9], [10], [11], [12], [13], [14]), scribbles (*e.g.* [15], [16]), and a constrained environment (*e.g.* [17]). In addition to the often-used image prior (*e.g.* local smoothness) on foreground and background images, traditional non-learning methods also introduce other priors to estimate alpha matte on the unknown regions of the trimap, *e.g.*, patch-wise color smoothness for $\boldsymbol{F}$ and $\boldsymbol{B}$ at the boundary of $\boldsymbol{\alpha}$ [7], local linear model encoded by matting Laplacian [15], [13], non-local smoothness combined with local smoothness [8], and proportionality of the gradient between $\boldsymbol{\alpha}$ and $\bar{\boldsymbol{I}}$ which is also called Poisson matting [6]. These priors are designed for the regions with partially-known trimap entries. Thus, they often fail or are not applicable to large unknown areas, eg subjects containing large semi-transparent areas. Also, these handcrafted priors are not adaptive to input image. There are also some methods (*e.g.* [18], [19]) which consider the sparsity priors via dictionary learning. In comparison to these methods, ours is a background-based approach and it employs the deep generative priors encoded by untrained neural networks. In comparison to existing handcrafted priors, the proposed method leverage the modeling power of deep learning to have a solution with better performance.

B. Deep Learning for Image Matting

In last few years, deep learning has emerged as a powerful approach for image matting, with noticeable improvement over traditional methods. Xu *et al.* [20] proposed a large-scale data set for image matting as well as an end-to-end CNN to solve the problem. Wang *et al.* [21] proposed to use the high-level features from a deep CNN to calculate the matting Laplacian. Inspired by the traditional sampling-based methods, Tang *et al.* [22] proposed a deep model which consists of both background and foreground sampling networks and a subsequent matting network. Viewing upsampling operators as index functions, Lu *et al.* [23] proposed an index-guided encoder-decoder framework for matting, where self-learned indices are used to guide the pooling and upsampling operators. Hou *et al.* [24] employed two encoder networks to extract local and global information respectively for matting. Cai *et al.* [25] disentangled image matting into two tasks: trimap adaptation and alpha estimation, which are respectively implemented with an individual network. Qiao *et al.* [26] introduced the spatial and channel-wise attention to the network for better processing edges and rough shapes. Zhou *et al.* [27] proposed an attention transfer module to reduce possible artificial content in the result, which is composed of a feature attention block and a scale transfer block. Recently, rather than focus on performance boosting, there are some studies making efforts on setting image matting free from trimaps. For instance, Liu *et al.* [28] showed that is possible to perform trimap-free matting on portrait images, but not general foreground objects. They utilized coarse annotations from segmentation datasets to enlarge the training dataset for matting and train the network in a semi-supervised setting. All methods discussed above are based on supervised learning and not for background matting. In comparison, ours is an unsupervised learning-based approach for background matting.

C. Background Matting

The background matting studied in [3], [4] solves (2) so as to avoid the issues arising from using trimaps. Compared to traditional image matting, the studies on background matting are still at the infancy stage. Sengupta *et al.* [3] trained a deep network with an adversarial loss to predict the alpha matte using a large set of portrait images with ground truths. Concretely, they first trained a matting network with a supervised loss on annotated data. To bridge the domain gap to unannotated real images, they trained another matting network guided by the first one and by a discriminator that quantifies the composition quality. Lin *et al.* [4] proposed a coarse-to-fine network which first predicts at the coarse scale, the foreground, the alpha matte and an error map. Then, these predictions are up-sampled to the fine scale, followed by the refinement on the hard regions defined as those having relatively-high error rates in the error map. These two methods showed better performance over existing non-learning and learning-based trimap-based methods. However, since these models require a portrait segmentation map as input [3] or are trained on portrait images [3], [4], they cannot generalize well to the images containing other kinds of subjects. In comparison, our proposed approach has no bias to training data and can be applied to various types of images, while having competitive performance as seen in the experiments.

D. Untrained Network Priors for Image Processing

There have been recent interests in using untrained neural networks as natural image priors. Deep image prior [5] and its variants such as deep decoder [29] are capable of solving many

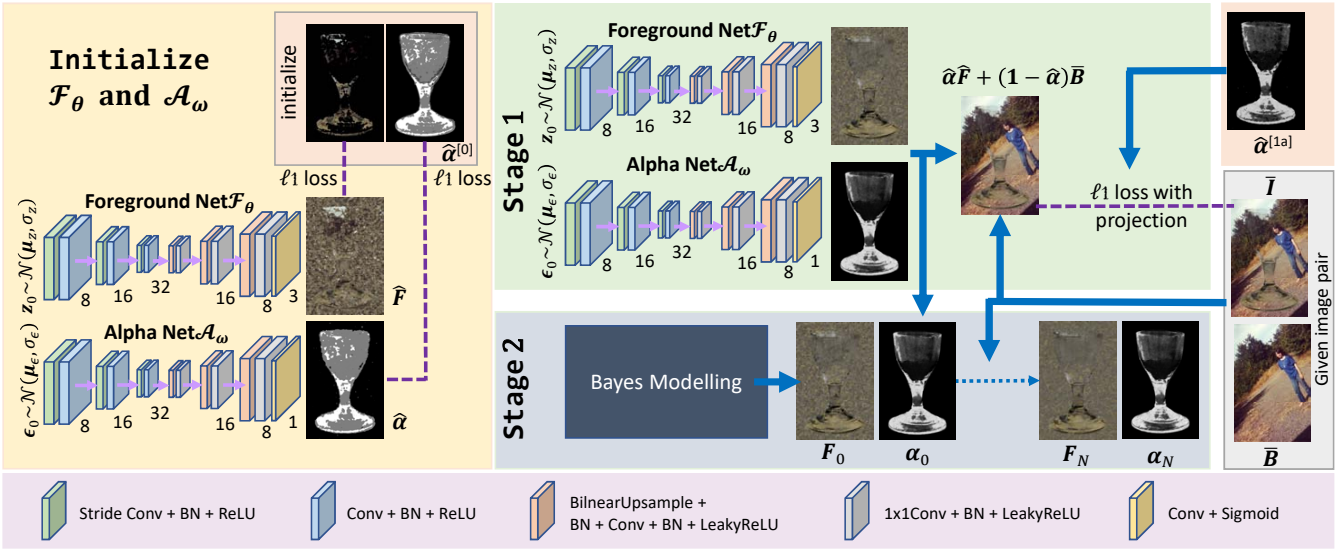


Fig. 2: Overview of the proposed approach for background matting. The number under each layer denotes the output channel number.

image recovery problems without training data. Gandelsman *et al.* [30] proposed to train multiple deep networks with deep image prior for image decomposition and image segmentation. Currently, there is no published work on applying untrained network priors to image matting. Our work is the first one and its proposed learning scheme is specialized for the problem which also differs from existing work. In addition, alpha matte is generally not a natural image, and our results demonstrate that an untrained deep CNN can work as a deep matte prior that leads to good estimation on the alpha matte.

III. Proposed Method

A. Overview

In the proposed approach, we model the foreground layer $\boldsymbol{F}$ and the alpha matte $\boldsymbol{\alpha}$ as follows:

$$\boldsymbol{F} \to \hat{\boldsymbol{F}} = \mathcal{F}_{\boldsymbol{\theta}}(\boldsymbol{z}_0), \quad \boldsymbol{\alpha} \to \hat{\boldsymbol{\alpha}} = \mathcal{A}_{\boldsymbol{\omega}}(\boldsymbol{\epsilon}_0), \tag{3}$$

where $\mathcal{F}_{\boldsymbol{\theta}}, \mathcal{A}_{\boldsymbol{\omega}}$ are two generative CNNs parameterized by $\boldsymbol{\theta}, \boldsymbol{\omega}$ respectively, and

$$\boldsymbol{z}_0 \sim \mathcal{N}(\mu_{\boldsymbol{z}}, \sigma_{\boldsymbol{z}}), \quad \boldsymbol{\epsilon}_0 \sim \mathcal{N}(\mu_{\boldsymbol{\epsilon}}, \sigma_{\boldsymbol{\epsilon}}), \tag{4}$$

are two input random seeds. The predictions from these two CNNs are related by

$$\bar{\boldsymbol{I}} = \mathcal{A}_{\boldsymbol{\omega}}(\boldsymbol{\epsilon}_0) \odot \mathcal{F}_{\boldsymbol{\theta}}(\boldsymbol{z}_0) + (1 - \mathcal{A}_{\boldsymbol{\omega}}(\boldsymbol{\epsilon}_0)) \odot \bar{\boldsymbol{B}}. \tag{5}$$

Then, these two CNNs are trained to approximately satisfy (5).

The two CNNs act as natural image and alpha matte models that incorporate the priors on their intermediate network layers, *i.e.*, there are regularizations on predictions implicitly imposed by the CNN architecture. Such deep image prior and deep matte prior encoded by the CNNs can make the prediction results satisfy certain image statistics (*e.g.* local smoothness) that facilitate background matting.

While deep image and matte priors partially regularize the CNN's predictions towards reasonable image statistics, there is ambiguity in their predictions that may cause likely overfitting of the CNNs. Thus, more treatments are needed for improving the prediction accuracy. Towards this end, we develop a two-stage scheme for the network learning: *(i)* training with a projection strategy; and *(ii)* applying Bayesian post-refinement for synchronizing the predictions from the two CNNs.

Since the problem (5) is highly non-linear and non-convex with ill-posedness, the accuracy of the prediction using (5) is largely dependent on the initialization of network parameters. Thus, we also develop an alpha map initialization scheme for initializing the CNNs effectively. In addition, we run the two-stage training twice. In the first run, we train the CNNs using an initialized alpha matte. After that, we have a much refined prediction of the alpha matte, and then we use it to initialize the CNNs and run the two-stage training again. See Fig. 2 for an illustration of the proposed approach and see also Algorithm 1 for the details.

It is hard to design a CNN with specific structure to model the foreground layer or alpha matte perfectly. For simplicity, both the CNNs in our method are set to have the same encoder-decoder architecture. The encoder contains three blocks, and each block contains two layers of $\text{Conv}^{\downarrow_2}_{3\times3}$-BN-LReLU. The decoder also has three blocks, and each block has one $\uparrow_2$-BN-$\text{Conv}_{3\times3}$-BN-LReLU layer and one $\text{Conv}_{1\times1}$-BN-LReLU layer. A sigmoid layer is applied to scaling the final result to the range of $[0, 1]$. Since alpha matte is assumed consistent across color channels, the final convolutional layer of $\mathcal{A}$ has only one output channel and its output is duplicated three times. In above, Conv is for convolutional layer, BN for batch normalization, LReLU for leaky rectified linear unit, and $\downarrow_2/\uparrow_2$ for downsampling/upsampling with factor of 2.

B. Network Initialization with Alpha Matte Initialization

Given the image pair $(\bar{\boldsymbol{I}}, \bar{\boldsymbol{B}})$, it is straightforward to estimate the pure background area defined by the index set

$$\Omega_{\text{B}} = \{j : \bar{\boldsymbol{I}}(j) = \bar{\boldsymbol{B}}(j)\}. \tag{6}$$

Algorithm 1 Unsupervised background matting

Input: An image pair $(\bar{\boldsymbol{I}}, \bar{\boldsymbol{B}})$
Output: Foreground $\boldsymbol{F}$, Alpha Matte $\boldsymbol{\alpha}$
1. Initialize $\boldsymbol{\alpha}^{[0]}$ via (8).
2. Initialize $\boldsymbol{\theta}$, $\boldsymbol{\omega}$ via (11) and (9) respectively.
3. **[Stage 1a]** Jointly update $\boldsymbol{\theta}$, $\boldsymbol{\omega}$ via (12).
4. **[Stage 1b]** Jointly update $\boldsymbol{\theta}$, $\boldsymbol{\omega}$ via (12) and (13).
5. **[Stage 2]** Calculate $\boldsymbol{F}_N$, $\boldsymbol{\alpha}_N$ via (18).
6. Update $\boldsymbol{\alpha}^{[0]}$ via (19), repeat 2∼5 one more time.
7. **return** $\boldsymbol{F} := \boldsymbol{F}_N$, $\boldsymbol{\alpha} := \boldsymbol{\alpha}_N$.

Inside the pure background area, the corresponding entries in alpha matte should be zeros. Let $\boldsymbol{P}_j^{\mathrm{I}}, \boldsymbol{P}_j^{\mathrm{B}}$ denote the $J \times J$ patch centered at the jth pixel at the Y channel in the YIQ color space of $\bar{\boldsymbol{I}}$ and $\bar{\boldsymbol{B}}$ respectively, where J is set to 5 in our practice. Before the first run of learning, we first initialize the alpha value on other pixels based on the normalized correlation map defined by

$$\boldsymbol{S}(j) = \frac{(\boldsymbol{P}_j^{\mathrm{I}} - \frac{1}{J^2}\sum_{k=1}^{J^2} \boldsymbol{P}_j^{\mathrm{I}}(k))^\top (\boldsymbol{P}_j^{\mathrm{B}} - \frac{1}{J^2}\sum_{k=1}^{J^2} \boldsymbol{P}_j^{\mathrm{B}}(k))}{\|\boldsymbol{P}_j^{\mathrm{I}} - \frac{1}{J^2}\sum_{k=1}^{J^2} \boldsymbol{P}_j^{\mathrm{I}}(k)\|_2 \|\boldsymbol{P}_j^{\mathrm{B}} - \frac{1}{J^2}\sum_{k=1}^{J^2} \boldsymbol{P}_j^{\mathrm{B}}(k)\|_2}. \tag{7}$$

This map indicates the similarity between an observed image patch and the corresponding background patch. If an observed image patch is highly similar to its background correspondence, it is probably that the center pixel is not a pure foreground pixel, and we set the alpha value of this pixel to 0.5 which means totally uncertain. In other words, we calculate the initial alpha matte $\hat{\boldsymbol{\alpha}}^{[0]}$ by

$$\hat{\boldsymbol{\alpha}}^{[0]}(j) = \begin{cases} 0, & \text{if } j \in \Omega_{\mathrm{B}}, \\ 0.5, & \text{if } j \notin \Omega_{\mathrm{B}}, \boldsymbol{S}(j) > \tau, \\ 1, & \text{otherwise}, \end{cases} \tag{8}$$

where τ is simply set to 0.9 in our practice. See Fig. 3 for three demos of our initialization scheme.

Based on $\hat{\boldsymbol{\alpha}}^{[0]}$, the two CNNs are separately initialized as follows. The CNN $\mathcal{A}$ for alpha matte is directly trained to fit the initial alpha matte:

$$\min_{\boldsymbol{\omega}} \ \|\mathcal{A}_{\boldsymbol{\omega}}(\boldsymbol{\epsilon}_0) - \hat{\boldsymbol{\alpha}}^{[0]}\|_1. \tag{9}$$

Let the pure foreground area defined by the index set

$$\Omega_{\mathrm{F}} = \{j : \hat{\boldsymbol{\alpha}}^{[0]}(j) = 1\}. \tag{10}$$

The CNN $\mathcal{F}$ for foreground is trained to fit the image pixels in the pure foreground area:

$$\min_{\boldsymbol{\theta}, \hat{\boldsymbol{F}} = \mathcal{F}_{\boldsymbol{\theta}}(\boldsymbol{z}_0)} \sum_{j \in \Omega_{\mathrm{F}}} |\hat{\boldsymbol{F}}(j) - \bar{\boldsymbol{I}}(j)|. \tag{11}$$

This can guide the CNN to capture the foreground correctly and help to alleviate introducing background edges on the predicted foreground in the subsequent stages.

C. Two-stage Scheme for Network Learning

1) Training with projection: In the 1st stage, the two CNNs are jointly learned via solving

$$\min_{\boldsymbol{\theta}, \boldsymbol{\omega}} \|(\mathcal{A}_{\boldsymbol{\omega}}(\boldsymbol{\epsilon}_0) \odot \mathcal{F}_{\boldsymbol{\theta}}(\boldsymbol{z}_0) + (1 - \mathcal{A}_{\boldsymbol{\omega}}(\boldsymbol{\epsilon}_0)) \odot \bar{\boldsymbol{B}} - \bar{\boldsymbol{I}}\|_1. \tag{12}$$

Fig. 3: Visual inspection of $\hat{\boldsymbol{\alpha}}^{[0]}$. Upper row: Input image. Bottom row: $\hat{\boldsymbol{\alpha}}^{[0]}$.

Note that the ℓ_1 norm is used to achieve the robustness to outliers. We observed that when a foreground pixel is similar to its background correspondence, the CNNs trained using (12) with sufficient iterations tend to be lazy, a kind of overfitting that simply sets the alpha value to zero and the foreground pixel to arbitrary value. This is probably because zero-valued alpha allows the corresponding foreground to be arbitrary and makes the foreground CNN learning easier.

To address the issue, we split the first stage into two parts. In the first part, we directly train the CNNs using (12) with a number of iterations, so as to have an improved alpha matte estimation from $\hat{\boldsymbol{\alpha}}^{[0]}$, denoted by $\hat{\boldsymbol{\alpha}}^{[1a]}$. Then in the second part, we use additional 1000 runs of (12) with projection at each iteration to address the overfitting and make the CNNs focus more on boundary prediction. The projection is given as follows:

$$\hat{\boldsymbol{F}}^{[1b]}(j) = \bar{\boldsymbol{I}}(j), \quad \text{if } \hat{\boldsymbol{\alpha}}^{[1a]}(i) = 1. \tag{13}$$

The rational of the projection comes from that, the pixels with consistency between their background regions and foreground regions often have their alpha values set to 1 by the initialization and kept 1 after the training in the first part. During this period, the initial learning rate on $\mathcal{A}$ is set to five times of the original one to jump out of the local minimum, and then gradually decreased to the original one.

2) Bayesian Post Refinement: After the 1st stage, we have the estimations denoted by $\boldsymbol{F}^* = \hat{\boldsymbol{F}}^{[1b]}, \boldsymbol{\alpha}^* = \hat{\boldsymbol{\alpha}}^{[1b]}$. However, the two CNNs may not "synchronize" during and after the 1st-stage training. That is, the alpha matte CNN $\mathcal{A}$ may be faster overfitting than the foreground CNN $\mathcal{F}$, as the alpha matte generally has simpler structures than the foreground layer but their corresponding CNNs are set to the same architecture with the similar training processes. As a result, the predicted alpha matte may be over-smooth. For improvement, we adapt the Bayesian post-processing [31] to our case so as to refine the predicted foreground and alpha matte. It models the variations among foreground layer and alpha matte with a posterior probability model and maximizes the posterior probability for refinement. Concretely, the refinement is done via solving

$$\max_{\boldsymbol{F}, \boldsymbol{\alpha}} p(\boldsymbol{\alpha}, \boldsymbol{F} | \boldsymbol{\alpha}^*, \boldsymbol{F}^*, \bar{\boldsymbol{B}}) \propto p(\boldsymbol{\alpha} | \boldsymbol{\alpha}^*) p(\boldsymbol{F} | \boldsymbol{F}^*) p(\boldsymbol{\alpha}, \boldsymbol{F}, \bar{\boldsymbol{B}}), \tag{14}$$

where the probability components are modeled with Gaussian distributions as follows:

$$p(\boldsymbol{\alpha}|\boldsymbol{\alpha}^*) \propto \exp(-\frac{\|\boldsymbol{\alpha}-\boldsymbol{\alpha}^*\|^2}{2\sigma_\alpha^2}), \tag{15}$$

$$p(\boldsymbol{\alpha},\boldsymbol{F},\bar{\boldsymbol{B}}) \propto \exp(-\frac{\|\bar{\boldsymbol{I}}-\boldsymbol{\alpha}\odot\boldsymbol{F}-(1-\boldsymbol{\alpha})\odot\bar{\boldsymbol{B}}\|^2}{2\sigma_{\mathrm{I}}^2}), \tag{16}$$

$$p(\boldsymbol{F}|\boldsymbol{F}^*) \propto \exp(-\frac{\|\boldsymbol{F}-\boldsymbol{F}^*\|^2}{2\sigma_{\mathrm{F}}^2}). \tag{17}$$

The parameters σ_α^2, σ_{F}^2, σ_{I}^2 are set to 10, 1, 1 respectively. Note that σ_α^2 is set much larger than σ_{F}^2 as the overfitting on alpha matte learning is empirically more severe than that on foreground layer learning in our framework. Applying an iterative block solver to (14) yields

$$\begin{cases} \boldsymbol{F}_{n+1} = \dfrac{\sigma_{\mathrm{I}}^2\boldsymbol{F}_n + \sigma_{\mathrm{F}}^2(\boldsymbol{\alpha}_n\odot(\bar{\boldsymbol{I}}-(1-\boldsymbol{\alpha}_n)\odot\bar{\boldsymbol{B}}))}{\sigma_{\mathrm{I}}^2+\sigma_{\mathrm{F}}^2(\boldsymbol{\alpha}_n\odot\boldsymbol{\alpha}_n)}, \\ \boldsymbol{\alpha}_{n+1} = \dfrac{\sigma_{\mathrm{I}}^2\boldsymbol{\alpha}_n + \sigma_\alpha^2((\boldsymbol{F}_n-\bar{\boldsymbol{B}})\odot(\bar{\boldsymbol{I}}-\bar{\boldsymbol{B}}))}{\sigma_{\mathrm{I}}^2+\sigma_\alpha^2((\boldsymbol{F}_n-\bar{\boldsymbol{B}})\odot(\boldsymbol{F}_n-\bar{\boldsymbol{B}}))}, \end{cases} \tag{18}$$

for $n=1,\cdots,N$, where $\boldsymbol{F}_0=\boldsymbol{F}^*$ and $\boldsymbol{\alpha}_0=\boldsymbol{\alpha}^*$.

D. A Second Run of Network Learning for Improvement

The initial alpha matte fed to the CNNs during the training plays an important role towards the performance. For the improvement on the initial alpha matte, we run the training scheme proposed in Section III-C one more time. After the first run, we have a much refined estimation of alpha matte denoted by $\hat{\boldsymbol{\alpha}}^{[2]}$. Then we update $\hat{\boldsymbol{\alpha}}^{[0]}$ by

$$\hat{\boldsymbol{\alpha}}_{\text{new}}^{[0]}(j) = \begin{cases} 1, & \text{if } (\hat{\boldsymbol{\alpha}}^{[0]}(j)+\hat{\boldsymbol{\alpha}}^{[2]}(j))/2 > 3/4, \\ 0, & \text{if } \hat{\boldsymbol{\alpha}}^{[0]}(j)=0, \\ 0.5, & \text{otherwise.} \end{cases} \tag{19}$$

The threshold value of $3/4$ is used, as only the pixels which are considered as pure foreground pixels in the original initialization (*i.e.* $\hat{\boldsymbol{\alpha}}^{[0]}(j)=1$) and whose alpha values after the first run are larger than 0.5 (*i.e.* $\hat{\boldsymbol{\alpha}}^{[2]}(j)>0.5$), are viewed as pure foreground pixels in the initialization of the new round. Afterwards, we initialize the CNNs using (11) and (9) with $\hat{\boldsymbol{\alpha}}_{\text{new}}^{[0]}$ and call the two-stage learning scheme again. Then the output of two trained CNNs are used as the final results.

IV. Experiments

A. Experimental Settings and Implementation Details

1) Datasets and metrics: The experimental evaluation is conducted on the SC (synthetic-composite) Adobe dataset [20] which provides a set of foreground images, alpha mattes and background images to synthesize images for evaluating image matting. Following [3], we first use 11 portraits provided by the dataset to composite 220 portrait images (resized to 512×512) with 20 randomly-selected backgrounds for test. We also evaluate on 780 non-portrait images (also resized to 512×512) composited by all 39 different foreground objects in the test set and 20 backgrounds. The SAD (Sum of Absolute Differences) and MSE (Mean of Squared Error) between the whole estimated and ground truth alpha mattes are used as the quantitative metrics. In addition, we collected a small dataset of real images without ground truths for the evaluation, on which the results are evaluated by visual inspection as well as by user study.

2) Implementation details: In all the experiments, both the CNNs for foreground and alpha matte are trained using the Adam optimizer with the learning rate of 0.001. The number of iterations in the joint training stage is set to 5000. The number of iterations in the Bayesianpost-refinement stage is set to 20. On the synthesized images (512×512 pixels), our PyTorch implementation takes around 4 minutes on average to process an image using a single RTX 3090 GPU. The code will be released upon the paper's acceptance.

B. Evaluation on Composite Portrait Images

Background matting is a very recent topic in image matting with few works available. In the experiments, two latest methods on this topic are included for comparison, *i.e.*, BGM [3] and BGMv2 [4]. Both of them are based on supervised deep learning. There are multiple published models for these two methods, and the best ones are used for performance evaluation. We also inlcude DoubleDIP [30] in the comparison, an untrained network-based method for general image decomposition. The DoubleDIP is adapted to background matting with its network on background prediction disabled.

Moreover, several trimap-based supervised deep models are included for a more comprehensive comparison: ATN [27], IM [23] and CAM [24]. Following [3], their input trimaps are generated by thresholding the ground truth alpha mattes, *i.e.*, setting all values in $(0,1)$ to 0.5, followed by the dilation with 10 or 20 steps. Their results are quoted from existing literature whenever possible, or re-produced by the codes published online. Furthermore, considering their similarity and applicability to matting, we also adopt two recent still-image salient object detection methods for comparison: GateNet [32] and SSOD [33]. We tried several ways to adapt their models to the matting task, including (a) retraining or fine-tuning on the matting dataset; (b) using $\ell1$, $\ell2$ or their original cross-entropy loss. As fine-tuning with the cross-entropy loss shows better performance, it is used for reporting the results on matting. For discussion convenience, the proposed approach is named as DMP (Deep Matte Prior) in the remaining discussion.

Method	Additional Input	SAD↓	MSE ($\times10^{-3}$)↓
ATN [27]	Trimap-10	3.55	4.3
	Trimap-20	4.56	5.7
CAM [24]	Trimap-10	3.04	4.0
	Trimap-20	3.12	4.0
IM [23]	Trimap-10	2.05	1.9
	Trimap-20	2.25	2.2
GateNet [32]	n/a	29.05	86.7
SSOD [33]	n/a	39.14	136.5
DoubleDIP [30]	Background	12.61	18.5
BGM [3]	Background	1.64	1.3
BGMv2 [4]	Background	1.56	1.2
DMP [Ours]	Background	1.54	1.7

TABLE I: Evaluation on portrait images of SC Adobe dataset. Trimap-K: Trimaps generated by ground truth thresholding and subsequent dilation by K steps.

Fig. 4: Comparison of alpha mattes predicted by different methods on portrait images from the SC Adobe dataset.

The quantitative results of all compared methods on portrait image matting are listed in Table I. Overall, DMP is very competitive to BGM and BGMv2, the supervised background matting methods, even it never accesses any ground-truth data during learning. Surprisingly, in terms of SAD, DMP yielded the best result among all compared methods. In terms of MSE, DMP performed worse than BGM and BGMv2 with a small gap, but it showed better performance to other remaining methods. Particularly, DMP outperformed the tripmap-based methods while avoiding the prerequisite on the annotations for trimaps. In addition, DMP outperformed DoubleDIP by a large margin. It indicates that general untrained-network-prior-based methods may not work well for background matting. In comparison, the special treatments introduced by the our DMP, *e.g.*the initialization and learning schemes, are very effective. Also, it can be seen that the salient object detection methods did not performe well on background matting, which is mainly due to their has less information to utilize, which makes the problem setting harder than that of the methods specifically designed for matting. See Fig. 4 for a visual comparison on some results. In comparison to BGM and BGMv2, DMP can better handle the details in the 1^{st} example, and is good at recovering the semi-transparent part in the alpha matte in the 2^{nd} example.

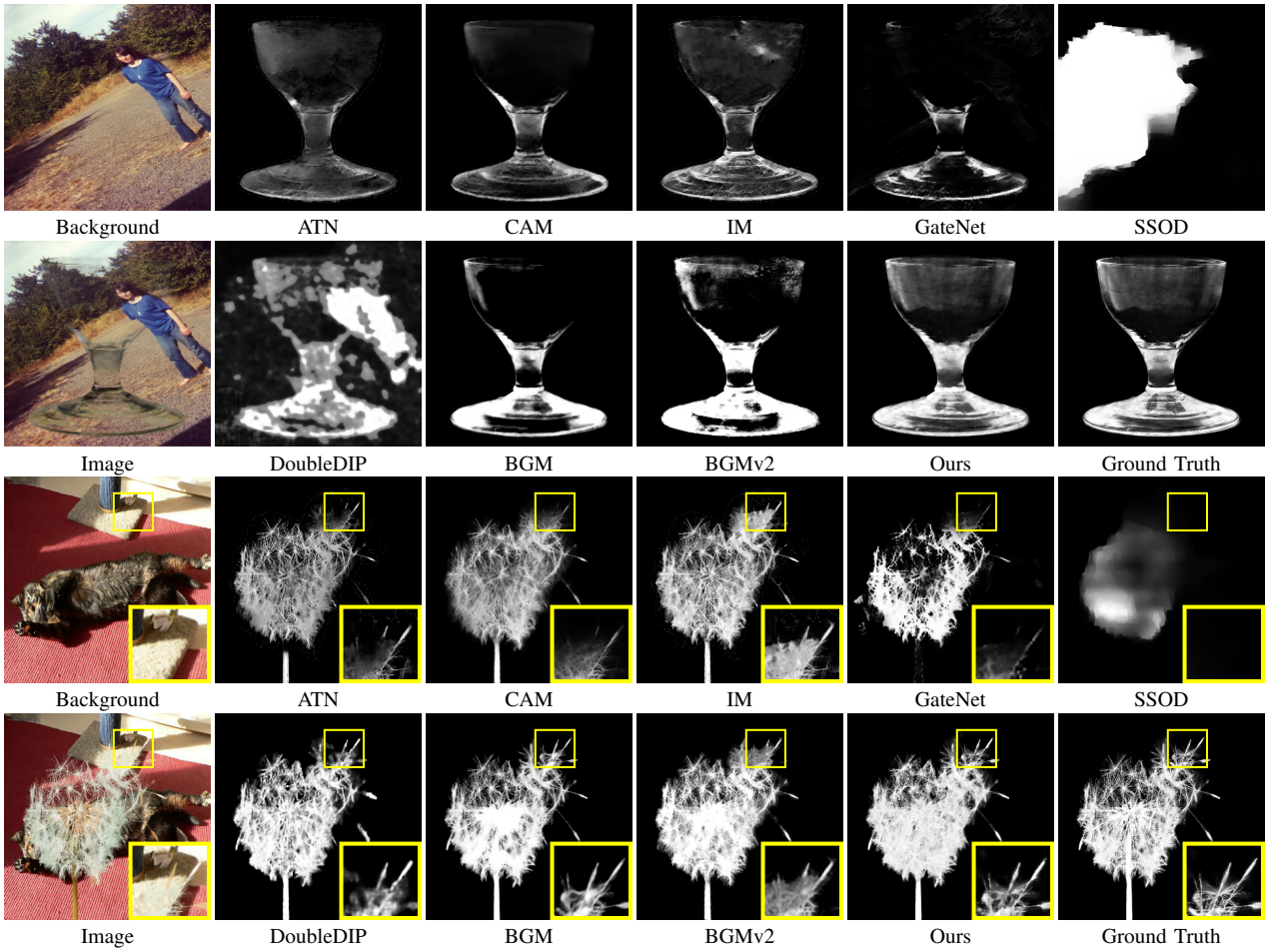


Fig. 5: Comparison of alpha mattes predicted by different methods on non-portrait images from the SC Adobe dataset.

Method	Additional Input	SAD↓	MSE ($\times 10^{-3}$)↓
ATN [27]	Trimap	15.64	26.8
CAM [24]	Trimap	9.85	13.3
IM [23]	Trimap	7.75	8.3
GateNet [32]	n/a	44.66	117.8
SSOD [33]	n/a	47.72	127.1
DoubleDIP [30]	Background	18.71	28.8
BGM [3]	Background	9.27	13.4
BGMv2 [4]	Background	8.23	12.6
DMP [Ours]	Background	7.30	10.0

TABLE II: Evaluation on non-portrait images of SC Adobe dataset.

C. Evaluation on Composite Non-Portrait Images

We also evaluate the performance of DMP on non-portrait images, and the previously selected methods are used for comparison. Since BGM needs the human object segmentation estimated by other deep learning methods as additional input, we directly call its model trained on portrait images. For BGMv2, we retrain its model on the non-portrait images. For ATN, CAM and IM, their published models for general object images are used. For salient object detection methods, we also tried several ways to apply these methods to matting and report the best results achieved by retraining using the cross-entropy loss (ℓ_1 loss works slightly worse in this setting).

See Table II for the quantitative comparison, where DMP outperformed all other methods except IM in terms of both metrics, and it outperformed IM in terms of SA. See also Fig. 5 for the visual results on two non-portrait images, where DMP achieved higher accuracy. The superior performance of our DMP is probably due to its learning is not dependent on external training data which may be biased, but dependent on the test image itself, which leads to better adaption. This has demonstrated the benefit of training-data-free unsupervised learning for background matting.

D. Evaluation on Real Data

To evaluate the performance of DMP on images taken in real scenarios, we collected 30 image/background pairs with 20 portrait images and 10 object images, using a smartphone with a tripod. The image resolution is fixed at 1920×1080. We run background matting methods BGM, BGMv2, DoubleDIP and DMP on these images. Then, we use each of their predicted alpha mattes and foregrounds to composite a new image on a green background for easy inspection. In addition, we also include the trimap-based matting methods ATN, CAM and IM for comparison, and we made the effort to do manual annotation to obtain the accurate trimaps required by these methods.

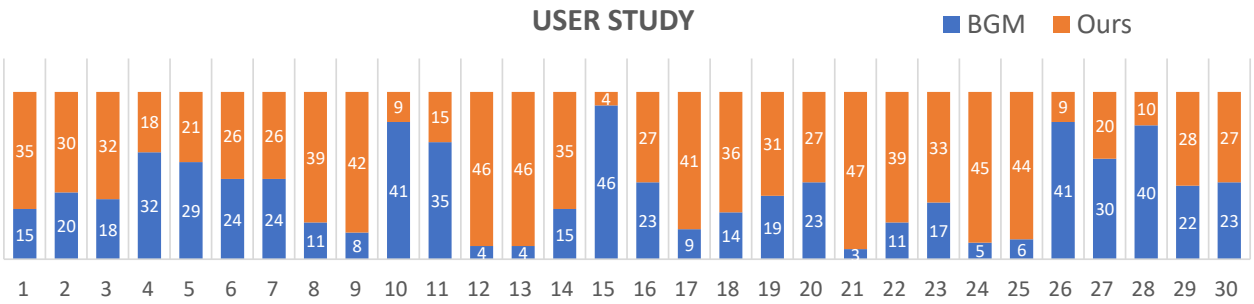


Fig. 6: Voting results in the user study on our collected real images. The horizontal and vertical axes denote the image's sequential number and the number of voting from users repressively.

Fig. 7: Comparison of compositions by several methods on some portrait images from our real data.

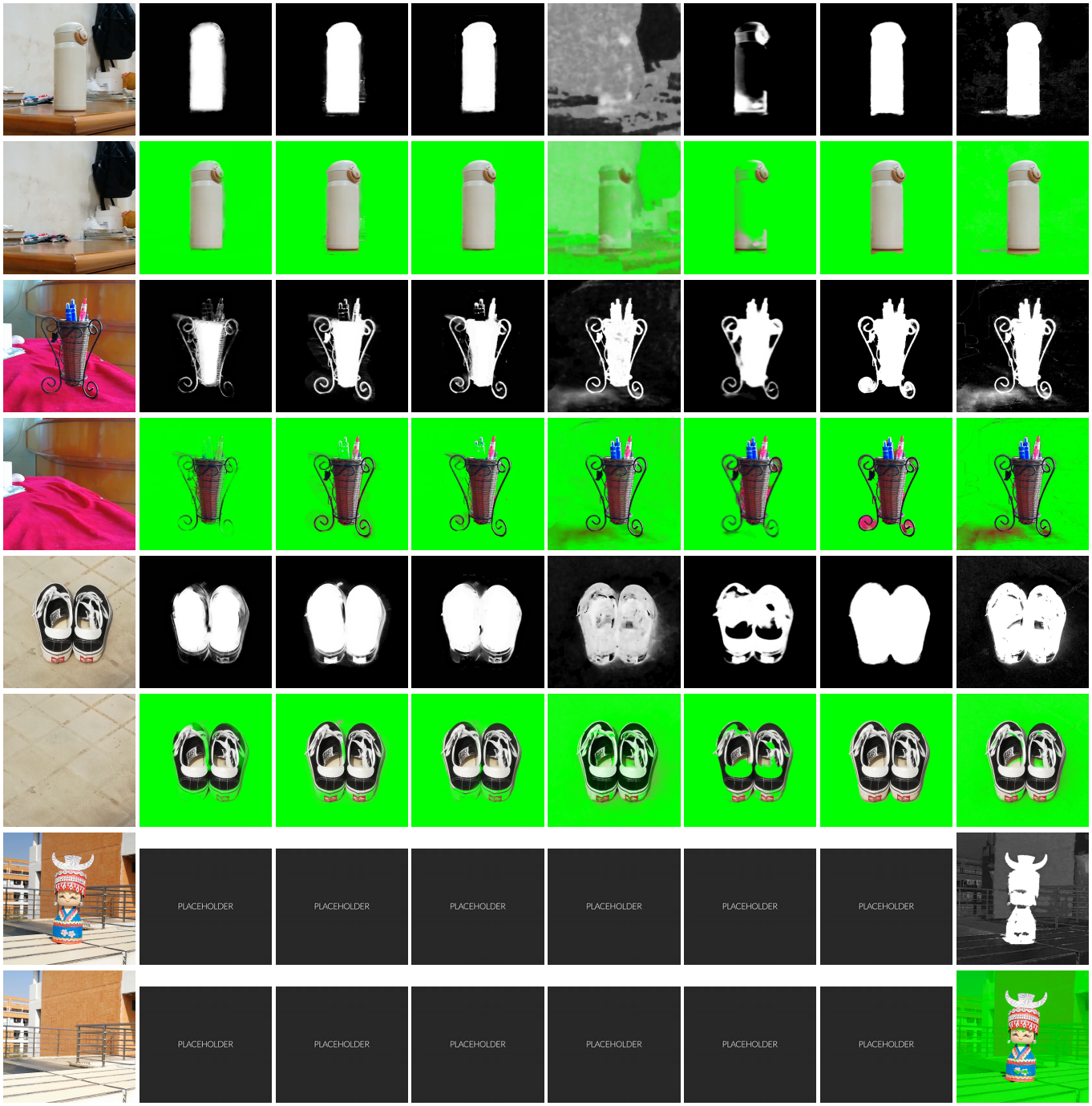


Fig. 8: Comparison of compositions by several methods on some non-portrait images from our real data.

To have a quantitative evaluation, a user study was conducted in the form of electronic questionnaire. We invited 50 persons (22 males and 28 females whose age distribution ranges from 18 to 50) to join the evaluation, each of them is given a pair of composited images from BGM and DMP. Both the order of the pairs as well as the order of the composited images in a pair are random. Each person was asked to vote which composited image is better. Each example is evaluated by all the users. The results are shown in Fig. 6. There are 22 out of 30 images for our method being voted, implying our results are better in terms of visual perception.

See Fig. 7 and Fig. 8 for visual illustraion of sample examples of the predicted alpha mattes and extracted foregrounds. Unlike the synthetic data, the two captures have discrepancy, such as different exposure, focus and noise. As a result, the performance of a background matting method will decrease. Nevertheless, the proposed DMP still performed better than BGM. In addition, DMP is robust to certain degradation effects of real data. For example, for noise supppression, the deep image/matte priors used in our method have good noise resistance (*i.e.* the CNNs prefers smooth output), as long as

the noise level is modest. For background illumination changes due to automatic exposure, as shown in in the last row of Fig. 7 and Fig. 8 respectively, there is only a little graying effect on background areas in our results in practice, thanks to the certain robustness of similarity measurement in (7). Compared to the results of DoubleDIP, ours are of apparently better visual quality. While BGMv2 shows better visual effects than other methods, DMP remains a very competitive performer in terms of visual quality.

E. Ablation Study

Recall the key steps in our learning scheme: *(i)* initialization of alpha matte before learning; *(ii)* using projection to avoid lazy learning in the 1st stage; and *(iii)* Bayesianpost-refinement in the 2nd stage. To analyze the contribution of each of these steps, we form the following several baselines by modifying and disabling one of the steps to conduct ablation study, with results summarized in Table III. See below for the analysis.

Method	SAD↓	MSE ($\times 10^{-3}$)↓
Our full version	1.54	1.7
Simple $\hat{\boldsymbol{\alpha}}^{[0]}$ init.	3.85	4.3
w/o Projection	1.77	1.9
w/o BPR	1.78	1.9
1 round	1.60	1.8
3 rounds	1.54	1.7
Simple $\hat{\boldsymbol{\alpha}}^{[0]}$ update	1.59	1.8

TABLE III: Results in ablation study.

1) Initialization: We replace the initialization scheme of $\hat{\boldsymbol{\alpha}}^{[0]}$ in (8) by a simple scheme: $\hat{\boldsymbol{\alpha}}^{[0]}(j) = 0.5, \forall j$. The resulting model is denoted by 'Simple $\hat{\boldsymbol{\alpha}}^{[0]}$ init.'. It can be seen that the initialization plays a critical role in DMP with a significant impact to the results. This is not surprising as the learning process is highly non-convex and non-smooth whose performance heavily relies on the quality of initialization. A visual comparison is given in Fig. 9(a). Without the proposed initialization, the model tends to believe the background image more and the opacity values of a large part of non-transparent pixels will approach to zero.

2) Projection in 1st-stage training: We remove the projection step defined by (13) in the 1st stage. The resulting model is denoted by 'w/o Projection'. The projection strategy has noticeable contribution to the performance. See Fig. 9(b) for a visual inspection, where training with the projection can suppress sparse errors and induce better local smoothness.

3) Bayesian post refinement: We disable the Bayesian post-refinement (BPR) during training, and the resulting model is denoted by 'w/o Refinement'. It can be seen that BPR does improve the performance. This is probably that the refinement allows the synchronization of the two CNNs which compensates some fine structures to the estimated alpha matte. See Fig. 9(c) for a visual comparison, where BPR brings more details of the hair in the alpha matte.

4) The second round: We run only one round as well as three rounds respectively of the two-stage learning scheme, and the resulting models are denoted by '1 round' and '3 rounds'. It can be seen that the 2nd round is necessary as it does bring performance improvement. The performance saturates after the 2nd round. This is probably because the alpha matte outputted by the 2nd round differs not much from that by the 1st round, unlike that of 1st round versus initialization. Thus, the re-initialization in the 3rd round helps little. We also replace the $\hat{\boldsymbol{\alpha}}^{[0]}$-update in (19) with a simpler one which only relies on current estimate: setting $\hat{\boldsymbol{\alpha}}^{[0]}_{\text{new}}(i)$ to 1 if $\hat{\boldsymbol{\alpha}}^{[0]}_{\text{new}}(i) > 0.95$, 0 if $\hat{\boldsymbol{\alpha}}^{[0]}_{\text{new}}(i) < 0.05$, and 0.5 otherwise. The resulting model is denoted by 'Simple $\hat{\boldsymbol{\alpha}}^{[0]}$ update', which shows a performance decrease. This is probably because such a scheme enlarges the errors when the predictions of some pixels are unreliable. The original scheme combines current prediction with the initialization, which increases the reliability.

F. Evaluation on Background-based Video Matting

The proposed DMP is also applicable to video matting on video sequences whose frames share a single background image [3], [4]. Let T denote the number of frames in the input video. The background image is also duplicated with T times as the input. The processing on the video then can be simply done by setting the output channel dimension of the foreground CNN as well as that of the alpha CNN to T. Such an extension also applies to DoubleDIP. Following [4], the evaluation is done on the test set of VideoMatte240K provided by [4], which provides portrait foreground videos and alpha matte videos extracted with green screens. Those videos are combined with the background images randomly selected from PASCAL VOC 2007 [34] to construct the videos for evaluation. We compare DMP with the video matting models of BGM, BGMv2 and DoubleDIP. Note that the trimap-based methods are not used for comparison as manual trimap annotation of video sequences is very challenging. The proposed DMP takes around 33 minutes to process 30 video frames of size 512×512 on a single RTX 3090 GPU. See Table IV for the quantitative results. Without calling any external video dataset, DMP outperformed the supervised learning-based method BGM and the untrained-network-based method DoubleDIP. Some results are visualized in Fig. 10. It can be seen that the BGM produced over-smooth results while the proposed DMP produced the results with more details.

Method	Video 1	Video 2	Video 3	Video 4	Video 5
BGM [3]	38.71/1.8	44.55/2.4	27.17/1.0	28.73/1.5	47.98/2.6
BGMv2 [4]	8.43/0.2	12.44/0.4	7.55/0.2	3.34/0.1	7.79/0.3
D.DIP [30]	42.81/2.2	49.65/1.4	36.82/2.3	33.15/2.2	52.20/3.0
DMP [Ours]	32.42/1.6	24.71/1.2	18.85/0.3	10.52/0.3	14.41/1.0

TABLE IV: Evaluation on portrait videos from VideoMatte240K in terms of SAD/MSE.

V. Conclusion

As a newly developed approach for image matting, background matting has attractive features over traditional trimap-based methods. Its task is about estimating the alpha matte and foreground from a pair of observed image and background image. This work showed that background matting can be

(a) Comparison of alpha mattes using original and simple initialization scheme of $\hat{\alpha}^{[0]}$.

(b) Comparison of alpha mattes w/ and w/o using projection at the 1st stage.

(c) Comparison of alpha mattes w/ and w/o Bayesian post-refinement.

Fig. 9: Some visualization of Ablation study.

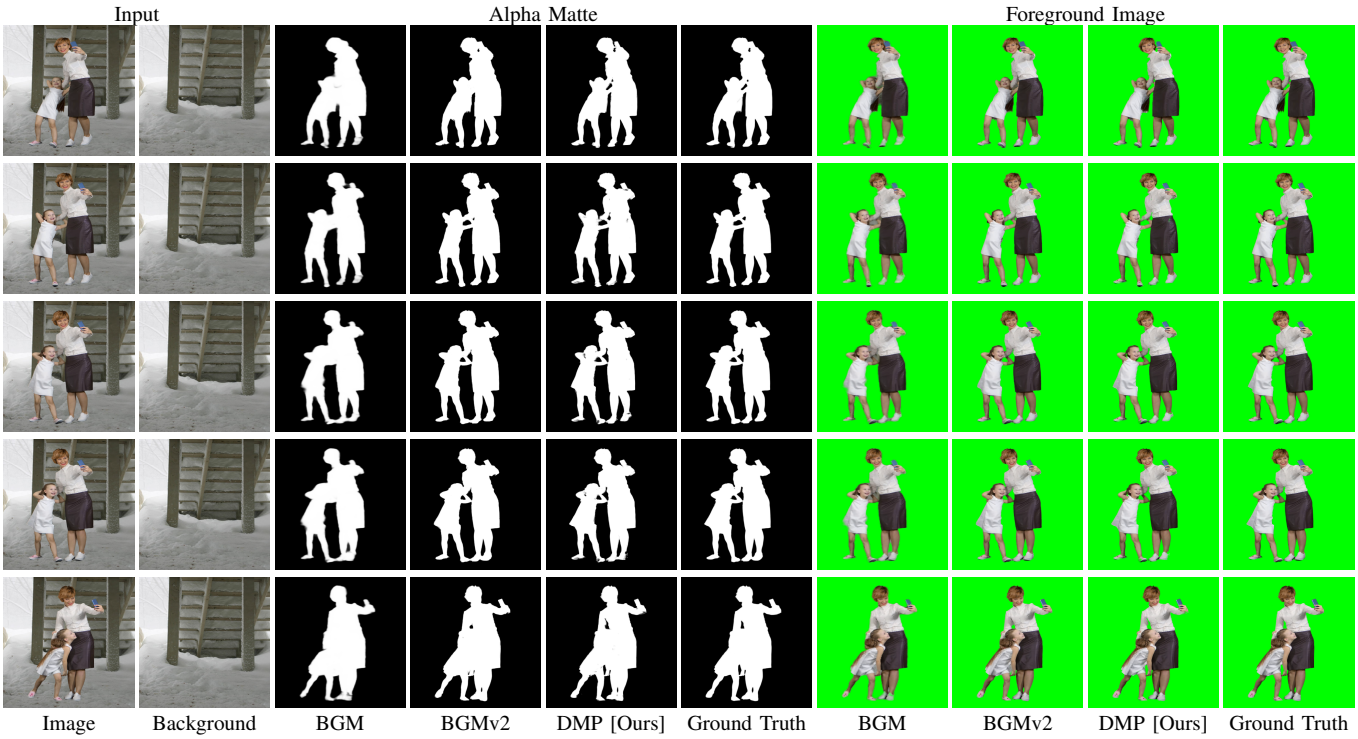


Fig. 10: Comparison of alpha mattes predicted by different methods on selected key frames from a video.

done effectively using unsupervised deep learning without any prerequisite on training data, which provides a complementary approach to existing supervised learning-based methods. There are two key parts in the proposed approach: deep image prior and deep/matte priors for modeling the foreground and alpha matte, and a well-designed two-stage unsupervised learning scheme for overcoming overfitting. The experiments has demonstrated that, in comparison with the latest supervised background matting method, ours performed competitively on portrait images and exhibited superior performance on non-portrait images. It also outperformed supervised trimap-based methods in some settings. We will study the extension to other image processing tasks in future.

REFERENCES

[1] Y. Niu, P. Liu, T. Zhao, and Y. Fan, "Matting-based residual optimization for structurally consistent image color correction," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 30, no. 10, pp. 3624–3636, 2020.

[2] Z. Pei, X. Chen, and Y.-H. Yang, "All-in-focus synthetic aperture imaging using image matting," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 28, no. 2, pp. 288–301, 2018.

[3] S. Sengupta, V. Jayaram, B. Curless, S. M. Seitz, and I. Kemelmacher-Shlizerman, "Background matting: The world is your green screen," in *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 2291–2300.

[4] S. Lin, A. Ryabtsev, S. Sengupta, B. L. Curless, S. M. Seitz, and I. Kemelmacher-Shlizerman, "Real-time high-resolution background matting," in *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 8762–8771.

[5] D. Ulyanov, A. Vedaldi, and V. Lempitsky, "Deep image prior," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2018, pp. 9446–9454.

[6] J. Sun, J. Jia, C.-K. Tang, and H.-Y. Shum, "Poisson matting," in *ACM SIGGRAPH*, 2004, pp. 315–321.

[7] Y.-Y. Chuang, B. Curless, D. H. Salesin, and R. Szeliski, "A bayesian approach to digital matting," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, vol. 2. IEEE, 2001, pp. II–II.

[8] M. Jin, B.-K. Kim, and W.-J. Song, "Adaptive propagation-based color-sampling for alpha matting," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 24, no. 7, pp. 1101–1110, 2014.

[9] Y. Aksoy, T. Ozan Aydin, and M. Pollefeys, "Designing effective inter-pixel information flow for natural image matting," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 29–37.

[10] J. Wang and M. F. Cohen, "Optimized color sampling for robust matting," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2007, pp. 1–8.

[11] E. S. Gastal and M. M. Oliveira, "Shared sampling for real-time alpha matting," in *Computer Graphics Forum*, vol. 29, no. 2. Wiley Online Library, 2010, pp. 575–584.

[12] E. Shahrian, D. Rajan, B. Price, and S. Cohen, "Improving image matting using comprehensive sampling sets," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2013, pp. 636–643.

[13] C. Xiao, M. Liu, D. Xiao, Z. Dong, and K.-L. Ma, "Fast closed-form matting using a hierarchical data structure," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 24, no. 1, pp. 49–62, 2014.

[14] X. Li, K. Liu, Y. Dong, and D. Tao, "Patch alignment manifold matting," *IEEE Transactions on Neural Networks and Learning Systems*, vol. 29, no. 7, pp. 3214–3226, 2018.

[15] A. Levin, D. Lischinski, and Y. Weiss, "A closed-form solution to natural image matting," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 30, no. 2, pp. 228–242, 2007.

[16] Y. Zheng and C. Kambhamettu, "Learning based digital matting," in *2009 IEEE 12th International Conference on Computer Vision*, 2009, pp. 889–896.

[17] A. R. Smith and J. F. Blinn, "Blue screen matting," in *Proceedings of Annual Conference on Computer Graphics and Interactive Techniques*, 1996, pp. 259–268.

[18] X. Feng, X. Liang, and Z. Zhang, "A cluster sampling method for image matting via sparse coding," in *Proceedings of European Conference on Computer Vision*, B. Leibe, J. Matas, N. Sebe, and M. Welling, Eds. Cham: Springer International Publishing, 2016, pp. 204–219.

[19] D. Zou, X. Chen, G. Cao, and X. Wang, "Unsupervised video matting via sparse and low-rank representation," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 42, no. 6, pp. 1501–1514, 2020.

[20] N. Xu, B. Price, S. Cohen, and T. Huang, "Deep image matting," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2017, pp. 2970–2979.

[21] Y. Wang, Y. Niu, P. Duan, J. Lin, and Y. Zheng, "Deep propagation based image matting." in *Proceedings of International Joint Conference on Artificial Intelligence*, vol. 3, 2018, pp. 999–1006.

[22] J. Tang, Y. Aksoy, C. Oztireli, M. Gross, and T. O. Aydin, "Learning-based sampling for natural image matting," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, 2019, pp. 3055–3063.

[23] H. Lu, Y. Dai, C. Shen, and S. Xu, "Indices matter: Learning to index for deep image matting," in *Proceedings of IEEE International Conference on Computer Vision*, 2019, pp. 3266–3275.

[24] Q. Hou and F. Liu, "Context-aware image matting for simultaneous foreground and alpha estimation," in *Proceedings of IEEE International Conference on Computer Vision*, 2019, pp. 4130–4139.

[25] S. Cai, X. Zhang, H. Fan, H. Huang, J. Liu, J. Liu, J. Liu, J. Wang, and J. Sun, "Disentangled image matting," in *Proceedings of IEEE International Conference on Computer Vision*, 2019, pp. 8819–8828.

[26] Y. Qiao, Y. Liu, X. Yang, D. Zhou, M. Xu, Q. Zhang, and X. Wei, "Attention-guided hierarchical structure aggregation for image matting," in *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, June 2020, pp. 13 673–13 682.

[27] F. Zhou, Y. Tian, and Z. Qi, "Attention transfer network for nature image matting," *IEEE Transactions on Circuits and Systems for Video Technology*, vol. 31, no. 6, pp. 2192–2205, 2021.

[28] J. Liu, Y. Yao, W. Hou, M. Cui, X. Xie, C. Zhang, and X.-S. Hua, "Boosting semantic human matting with coarse annotations," in *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 8560–8569.

[29] R. Heckel and P. Hand, "Deep decoder: Concise image representations from untrained non-convolutional networks," in *Proceedings of International Conference on Learning Representations*, 2018.

[30] Y. Gandelsman, A. Shocher, and M. Irani, "Double-dip": Unsupervised image decomposition via coupled deep-image-priors," in *Proceedings of IEEE Conference on Computer Vision and Pattern Recognition*, vol. 6, 2019, p. 2.

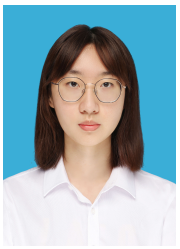
[31] M. Forte and F. Pitié, "f, b, alpha matting," *arXiv preprint arXiv:2003.07711*, 2020.

[32] X. Zhao, Y. Pang, L. Zhang, H. Lu, and L. Zhang, "Suppress and balance: A simple gated network for salient object detection," in *Proceedings of European Conference on Computer Vision*. Springer, 2020, pp. 35–51.

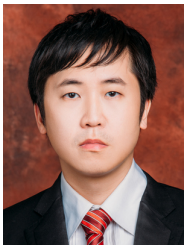
[33] J. Zhang, X. Yu, A. Li, P. Song, B. Liu, and Y. Dai, "Weakly-supervised salient object detection via scribble annotations," in *Proceedings of IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 12 546–12 555.

[34] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman, "The PASCAL Visual Object Classes Challenge 2007 (VOC2007) Results," http://www.pascal-network.org/challenges/VOC/voc2007/workshop/index.html.

Yong Xu received the B.S., M.S., and Ph.D. degrees in mathematics from Nanjing University, Nanjing, China, in 1993, 1996, and 1999, respectively. He is currently a professor in Computer Science at South China University of Technology. His research interests include image processing and analysis.



Baoling Liu is currently an M.sc candidate Computer Science at South China University of Technology. She is working on image recovery.



Yuhui Quan received the Ph.D. degree in Computer Science from South China University of Technology in 2013. He worked as the postdoctoral research fellow in Mathematics at National University of Singapore from 2013 to 2016. He is currently the associate professor in Computer Science at South China University of Technology. His research interests include image processing, sparse representation and deep learning.

Hui Ji received the B.Sc. degree in Mathematics from Nanjing University in China, the M.Sc. degree in Mathematics from National University of Singapore and the Ph.D. degree in Computer Science from the University of Maryland, College Park. In 2006, the joined National University of Singapore as an assistant professor in Mathematics. Currently, he is an associate professor in mathematics at National University of Singapore. His research interests include computational harmonic analysis, optimization, image processing and machine learning.