

# DART: Depth-Enhanced Accurate and Real-Time Background Matting

Hanxi Li<sup>1,†</sup>, Guofeng Li<sup>1,†</sup>, Bo Li<sup>2,\*</sup>, Lin Wu<sup>3</sup>, and Yan Cheng<sup>1</sup>

<sup>1</sup>Jiangxi Normal University, Jiangxi, China

<sup>2</sup>Northwestern Polytechnical University, Shaanxi, China

<sup>3</sup>Swansea University, United Kingdom

arXiv:2402.15820v1 [cs.CV] 24 Feb 2024

**Abstract**—Matting with a static background, often referred to as “Background Matting” (BGM), has garnered significant attention within the computer vision community due to its pivotal role in various practical applications like webcasting and photo editing. Nevertheless, achieving highly accurate background matting remains a formidable challenge, primarily owing to the limitations inherent in conventional RGB images. These limitations manifest in the form of susceptibility to varying lighting conditions and unforeseen shadows.

In this paper, we leverage the rich depth information provided by the RGB-Depth (RGB-D) cameras to enhance background matting performance in real-time, dubbed DART. Firstly, we adapt the original RGB-based BGM algorithm to incorporate depth information. The resulting model’s output undergoes refinement through Bayesian inference, incorporating a background depth prior. The posterior prediction is then translated into a “trimap,” which is subsequently fed into a state-of-the-art matting algorithm to generate more precise alpha mattes. To ensure real-time matting capabilities, a critical requirement for many real-world applications, we distill the backbone of our model from a larger and more versatile BGM network. Our experiments demonstrate the superior performance of the proposed method. Moreover, thanks to the distillation operation, our method achieves a remarkable processing speed of 33 frames per second (fps) on a mid-range edge-computing device. This high efficiency underscores DART’s immense potential for deployment in mobile applications<sup>1</sup>

**Index Terms**—Background matting; RGB-Depth images.

## I. INTRODUCTION

**I**MAGE Image matting is a well-established problem in computer vision, with applications spanning image and video editing, video conferencing, and more. Researchers have dedicated substantial efforts to realizing automatic and high-quality image matting under real-world conditions, resulting in advancements such as deep learning-based approaches [1–11]. A significant challenge in automatic image matting lies in the requirement of a “trimap,” a crucial input for conventional matting algorithms. Without a proper trimap, the matting problem becomes ill-posed, as distinguishing between the “foreground” and “background” can be highly ambiguous [11–18]. To address this issue, pioneering work such as the “Background Matting” (BGM) algorithm [5, 19] was introduced. BGM conducts image matting against a fixed background, ensuring

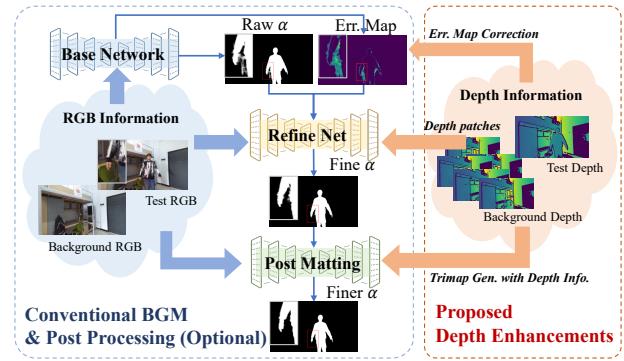


Fig. 1. The illustration on the proposed depth-enhanced accurate and real-time background matting (DART). Left: The conventional BGM and the optional post-matting process. Right: our depth-based enhancement approach.

a well-defined foreground that can be accurately extracted even without human-labeled trimaps. However, challenges persist, especially when dealing with shadows introduced by foreground objects or unexpected lighting variations.

In this paper, we present an innovative approach that leverages depth information obtained from a standard RGB-D camera to significantly improve the accuracy and robustness of image matting with static backgrounds. This novel utilization of depth data addresses some of the limitations inherent in traditional methods like BGM (Background Matting) and enhances the precision and reliability of matting results. We provide a schematic overview of our proposed method in Figure 1. Our approach builds upon the foundation of the BGMv2 algorithm [19] but introduces several key modifications to incorporate depth information. We refer to this method as “Depth-enhanced Accurate and Real-Time BackGround Matting,” or simply “DART” for brevity. Specifically, we utilize depth information to correct the “error map” estimated by the base network of the original BGM model [19] and refine the “trimap” used in the post-matting process. Additionally, we replace the use of traditional RGB patches with RGB-D patches in the refining network [19]. Furthermore, we introduce the concept of distillation, where we derive a smaller base network [20] from the original ResNet50-based model used in BGMv2. These modifications collectively result in significantly improved matting performance compared to state-of-the-art methods, all while maintaining exceptional

<sup>†</sup> These authors contributed equally to this work.

<sup>\*</sup> Corresponding author.

<sup>1</sup> Source code are available at <https://github.com/Fenghoo/DART>.

computational efficiency. Our DART algorithm can achieve a remarkable speed of 125 FPS on an affordable desktop GPU and still maintains a respectable 33 FPS on a mid-range edge-computing platform.

## II. THE PROPOSED METHOD

The workflow of DART is illustrated in Fig. 2. As Fig. 2, we follow BGMv2 as the base network, and then we employ a smaller network (MobileNetv2 [21]) for higher efficiency and the network parameter is learned via distilling the knowledge from the base network of BGMv2, followed by a fine-tuning on the current background. Besides, for different stages of the background matting process, the depth information is exploited in different manners to increase the matting accuracy and robustness. The remaining part of this section will detail the proposed depth-enhanced BGM algorithm.

### A. Efficient base network from distillation

In the original BGMv2 algorithm, the ResNet50-based [22] “base network” of BGMv2 processes an input image  $\mathbf{I} \in \mathbb{Z}^{H \times W \times 3}$  as the following function:

$$\Phi_r(\mathbf{I}) \rightarrow \{\mathbf{A}_{raw}^* \in \mathbb{R}^{\frac{H}{4} \times \frac{W}{4}}, \mathbf{E}_{RGB}^* \in \mathbb{R}^{\frac{H}{4} \times \frac{W}{4}}\}, \quad (1)$$

where  $\mathbf{A}_{raw}^* \in \mathbb{R}^{\frac{H}{4} \times \frac{W}{4}}$  denotes the raw prediction of the alpha matte while  $\mathbf{E}_{RGB}^* \in \mathbb{R}^{\frac{H}{4} \times \frac{W}{4}}$  is the predicted “error map” or in other words, the uncertain region of  $\mathbf{A}_{raw}^*$  [19].

In this work, we employ a MobileNetv2-based [21] network to predict the raw alpha of the test image. The similar inference process of this smaller model can be denoted as:

$$\Phi_m(\mathbf{I}) \rightarrow \{\mathbf{A}_{raw} \in \mathbb{R}^{\frac{H}{4} \times \frac{W}{4}}, \mathbf{E}_{RGB} \in \mathbb{R}^{\frac{H}{4} \times \frac{W}{4}}\}. \quad (2)$$

where  $\Phi_m(\mathbf{I})$  denotes the MobileNetv2-based model which is learned via distillation. Following the SOTA distillation algorithm for segmentation [23], the knowledge transfer from  $\Phi_r$  to  $\Phi_m$  can be realized via minimizing the following loss function:

$$\begin{aligned} L_{distill} = & KL(\mathbf{A}_{raw}, \mathbf{A}_{raw}^*) \\ & + \|\mathbf{A}_{raw} - \mathbf{A}_{GT}\|_{l_1} + \|\mathbf{E}_{RGB} - \mathbf{E}_{GT}\|_{l_2}, \end{aligned} \quad (3)$$

where  $\mathbf{A}_{GT}$  and  $\mathbf{E}_{GT}$  stand for the ground-truth alpha matte and error map respectively;  $KL(\cdot)$  denotes the Kullback–Leibler divergence between two prediction maps. Note that in this paper we do not impose different weights for the three losses as the straightforward summation can already lead to sufficiently good results.

### B. Error map correction in a Bayesian style

Besides the RGB images, the depth channel is also informative enough to estimate the foreground region roughly. In this work, we conduct the depth-based matting in the Bayesian manner. In particular, given the (resized) depth map of the test image  $\mathbf{D} \in \mathbb{R}^{\frac{H}{4} \times \frac{W}{4}}$  and the background depth map set  $\mathcal{D}_b = \{\forall \mathbf{D}_b^i \in \mathbb{R}^{\frac{H}{4} \times \frac{W}{4}} \mid i = 1, 2, \dots, N\}$ <sup>2</sup>, one can first fill the unknown depth pixels with the special value  $-1$ . Then

<sup>2</sup>In this paper we assume that multiple static background RGB-D images are captured before matting. Note that this is trivial to realize considering that most RGB-D cameras can run faster than 30 FPS.

the mean value and the standard deviation of the background depth on the coordinate  $[r, c]$  can be calculated as:

$$\begin{aligned} \bar{\mathbf{D}}_b^{r,c} &= \begin{cases} \frac{1}{|\mathcal{K}|} \sum_{i \in \mathcal{K}} \mathbf{D}_b^i(r, c), & |\mathcal{K}| > 0 \\ d_{max}, & |\mathcal{K}| = 0 \end{cases} \\ \sigma_b^{r,c} &= \begin{cases} \sqrt{\frac{1}{|\mathcal{K}|-1} \sum_{i \in \mathcal{K}} (\mathbf{D}_b^i(r, c) - \bar{\mathbf{D}}_b^{r,c})^2}, & |\mathcal{K}| > 1 \\ \psi \cdot \bar{\mathbf{D}}_b^{r,c}, & |\mathcal{K}| = 1 \\ \psi \cdot d_{max}, & |\mathcal{K}| = 0 \end{cases} \end{aligned} \quad (4)$$

where  $d_{max}$  is the maximum detecting distance of the depth sensor;  $\psi$  is a small ratio;  $\mathcal{K} = \{\forall i \mid \mathbf{D}_b^i(r, c) > 0\}$  and  $|\cdot|$  denotes the set cardinality. Consequently, the conditional probability of an observed pixel depth  $d$  on the coordinate  $[r, c]$ , given this pixel belongs to the foreground is calculated as:

$$P_F^{r,c}(d) = P(\mathbf{D}(r, c) = d \mid F) = \begin{cases} \frac{1}{\bar{\mathbf{D}}_b^{r,c}}, & d \in (0, \bar{\mathbf{D}}_b^{r,c}] \\ 0, & \text{else} \end{cases} \quad (5)$$

Similarly, the conditional probability under the background condition writes:

$$P_B^{r,c}(d) = P(\mathbf{D}(r, c) = d \mid B) = \begin{cases} \mathcal{N}_{r,c}^+(\bar{\mathbf{D}}_b^{r,c}, \sigma_b^{r,c}), & d > 0 \\ 0, & \text{else} \end{cases} \quad (6)$$

where  $\mathcal{N}_{r,c}^+(\bar{\mathbf{D}}_b^{r,c}, \sigma_b^2)$  denotes the estimated normal distribution of the background depth value on the pixel  $[r, c]$ <sup>3</sup>.

In a Bayesian way, the posterior probability that the pixel  $[r, c]$  belongs to the foreground can be calculated as:

$$\tilde{P}_F^{r,c}(d) = \frac{P(F \mid \mathbf{D}(r, c) = d)}{P_F^{r,c}(d) \cdot P_B + P_F^{r,c}(d) \cdot P_F + \zeta} \quad (7)$$

where  $P_F$  and  $P_B = 1 - P_F$  are the pre-defined prior probabilities of foreground and background, respectively;  $\zeta$  is a small-valued parameter to ensure the foreground probability of a depth-unknown pixel equals  $P_F$ . We then arrive at the foreground posterior map  $\mathbf{A}_D \in \mathbb{R}^{\frac{H}{4} \times \frac{W}{4}}$  with each element defined as:

$$\mathbf{A}_D(r, c) = \tilde{P}_F^{r,c}(\mathbf{D}(r, c)), \quad \forall r, c. \quad (8)$$

In practice, we post-process  $\mathbf{A}_D(r, c)$  by gaussian blurring and small region removal for more robust prediction.

Recall that the RGB-based raw alpha matte defined in Equ. 2, we can compare the two foreground probability maps to generate a residual map as

$$\mathbf{E}_D = |\mathbf{A}_D - \mathbf{A}_{raw}|^\circ \in \mathbb{R}^{\frac{H}{4} \times \frac{W}{4}}, \quad (9)$$

where  $|\cdot|^\circ$  denotes the element-wise absolute value. The residual map is used as a complement to the RGB-based error map  $\mathbf{E}_{RGB}$  and the corrected error map is given by:

$$\mathbf{E}_{RGBD} = \beta \cdot \mathbf{E}_D + (1 - \beta) \cdot \mathbf{E}_{RGB}, \quad (10)$$

where  $\beta$  is the balancing parameter for fusing the two error maps.

<sup>3</sup>Note that the negative part of the distribution is truncated and the probability function is scaled so that the integral value over the domain  $[0, \infty)$  is 1.

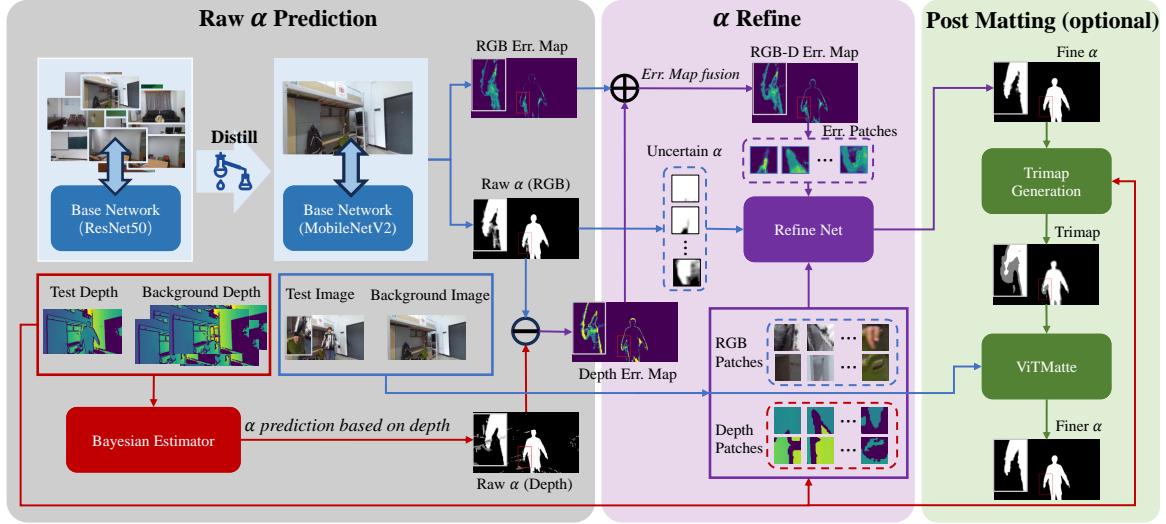


Fig. 2. The workflow of the proposed DART algorithm. The three stages of DART are shown in the gray, purple, and green regions. Better view in color.

### C. Alpha refinement with RGB-D patches

The second stage of the conventional BGM algorithm is the patch-level correction to the raw prediction. In the RGB-D scenario of this work, we propose to employ the RGB-D patch, rather than the RGB path as the input of the refinement net. Considering that the depth information is relatively independent to the RGB content, the RGB-D patch could lead to a more accurate correction. This benefit is also proved empirically in the experiment part. In a mathematical way, the refining model in this work can be defined as

$$\Omega(\mathbf{I}, \mathbf{D}, \mathbf{A}_{\text{raw}}, \mathbf{E}_{\text{RGBD}}) \rightarrow \mathbf{A}_{\text{fine}} \in \mathbb{R}^{H \times W} \quad (11)$$

### D. Post matting with depth information

Given the sufficient computational or time budget, the optional post-matting can significantly increase the matting accuracy. In this paper, we employ the depth information to generate a better “trimap” for the SOTA matting algorithm [24]. Recall that the refined alpha prediction is denoted as  $\mathbf{A}_{\text{fine}}$  which can be viewed as the “prior” foreground probability to the next inference stage, thus the posterior map considering the depth evidence is calculated as:

$$\tilde{\mathbf{A}}_{\text{fine}}^{r,c} = \frac{P_F^{r,c}(d) \cdot \mathbf{A}_{\text{fine}}^{r,c}}{P_F^{r,c}(d) \cdot (1 - \mathbf{A}_{\text{fine}}^{r,c}) + P_F^{r,c}(d) \cdot \mathbf{A}_{\text{fine}}^{r,c}}. \quad (12)$$

We then generate a trimap based on the posterior map  $\tilde{\mathbf{A}}_{\text{fine}}$  as

$$\tilde{\mathbf{A}}_{\text{fine}} \xrightarrow{\text{Gaussian Blur}} \tilde{\mathbf{A}}_{\text{fine}}^{\dagger} \xrightarrow{\text{Two Thresholds}} \mathbf{T} \in \mathbb{R}^{H \times W}, \quad (13)$$

where  $\mathbf{T}$  is a triple-valued map with 1-valued pixels indicating the foreground objects, 0-valued ones indicating the background area and 0.5-valued ones standing for the unknown region. This trimap is finally fed into the SOTA ViTMatte algorithm [24] for generating more accurate foreground masks. Note that this post-matting process is optional as the better performance is achieved at the cost of decreasing.

### E. Proposed Dataset

As we introduced before, no RGB-D background matting dataset is available so far. We thus design and produce a specific dataset for this task. The proposed dataset, termed “JXNU RGBD Background Matting” or JXNU-RGBD for short, contains 12 indoor scenes, with each one involves 100 pure background RGB-D images and 5 RGB-D images with foreground objects for testing. Only the alpha matte on the test images is manually labeled for algorithm evaluation, and the labels are strictly unseen during training. Fig. 3 illustrates 8 out of the 12 scenes of the proposed dataset.



Fig. 3. The proposed RGB-D background matting dataset. 8 scenes are illustrated here, each with one RGB image pair (test and background) and one depth map pair. The dark blue pixels of the depth map stand for the depth-unknown region. Better view in color.

### F. Training strategy and implementation detail

Following the standard training protocol of background matting [19], we train our DART model only based on real background (RGB-D) images and synthetic (RGB-D) foreground objects. In particular, the ResNet50-based base network is trained based on the training set proposed in BGMv2 [19] augmented with the foreground objects (RGB) extracted from the X-Humans dataset [25]. Secondly, the ResNet50 model is then fine-tuned on the proposed JXNU-RGBD dataset with

only background images (RGB) and artificial foregrounds. Finally, the MobileNetv2-based base network is distilled from the fine-tuned larger model based upon the background images (RGB) merely from the target scene. As to the refine net of DART, we first fuse the rendered RGB-D samples from the X-Humans dataset [25] with all the background RGB-D images of JXNU-RGBD to train the raw model and then slightly fine-tune it using only the information from the target scene.

As to the hyperparameters, we set  $\beta = 0.05$  to fuse the two error map,  $\psi = 0.01$ ,  $d_{max} = 5460$ ,  $\zeta = 0.001$ . The two thresholds for generating trimap are  $[0.25, 0.8]$ . When distilling the base network, the batch size is set to 16 and the learning rates for the three submodules of the base network are  $[1e - 4, 5e - 4, 5e - 4]$ .

### III. EXPERIMENT

#### A. Basic settings

In this section, we perform a series of experiments to evaluate the proposed method, compared with the BGM-V2 [19] (the SOTA background matting method); ViTMatte [24] (the SOTA general purpose matting algorithm); HIM [8], SGHM [26] and P3M-Net [10] (three SOTA human matting algorithms).

Four commonly used matting metrics, namely the sum of absolute difference (SAD), mean square error(MSE), gradient error (Grad), and connectivity error (Conn) are employed for scoring the comparing methods. We also report the FPS numbers for each matting algorithm. Most experiments are conducted on a desktop PC with an Intel i5-13490F CPU, 32G DDR4 RAM , and an NVIDIA RTX4070Ti GPU. To evaluate the compatibility of the proposed method on the edge-computing devices, we also run the proposed method on an NVIDIA Jetson Orin NX development board, with the reimplemented code employing the TensorRT SDK [27] and the FP16 data format.

#### B. Quantitative Results

The matting accuracies of the involved methods are reported in Tab. III-B, along with the corresponding running FPS numbers. Note that ViTMatte requires a trimap as a part of its input, we thus generate a trimap for it based on the posterior map defined in Equ. 7.

TABLE I

THE MATTING PERFORMANCES OF THE PROPOSED METHOD AND THE COMPARING SOTA ALGORITHMS. THE RUNNING FPS OF EACH COMPARING ALGORITHM IS ALSO ILLUSTRATED IN THE LAST COLUMN. NOTE THAT MSE IS DIVIDED BY 1000 FOR EASE OF READING. THE SPEED IS MEASURED ON THE DESKTOP ENVIRONMENT DESCRIBED IN SEC. III-A.

Methods	SAD ↓	MSE ↓	Grad. ↓	Conn↓	FPS ↑
ViTMatte[24]	17.71	11.16	21.67	16.60	5
P3M-Net [10]	18.78	8.60	10.9	18.57	4
SGHM [26]	6.95	2.67	8.51	6.41	12
HIM [8]	4.28	1.09	6.92	3.55	4
BGM [19]	4.78	1.86	10.05	4.67	81
DART	3.39	1.22	8.89	3.33	125
DART + ViTMatte[24]	2.90	0.61	6.02	2.42	5

According to the table, one can clearly see the superiority of the proposed DART algorithm. In particular, the DART with post-processing beats all the SOTA methods evaluated by all four metrics; the vanilla DART algorithm achieves slightly

worse matting accuracy compared with its sophisticated version but enjoys a 25-time faster speed.

Fig. 4 demonstrates the performances of the compared methods from another angle. The marker's location indicates the matting accuracy while the algorithm speed is represented by the marker's color (the redder, the faster) and size (the larger, the faster). From the figure, we can see the proposed DART algorithm performs fastest among all the comparing methods. On the edge-computing platform (NVIDIA Jetson Orin NX), our method can also achieve real-time speed (33 FPS) while still enjoying a remarkable accuracy advantage to the fast BGM-V2 algorithm [19].

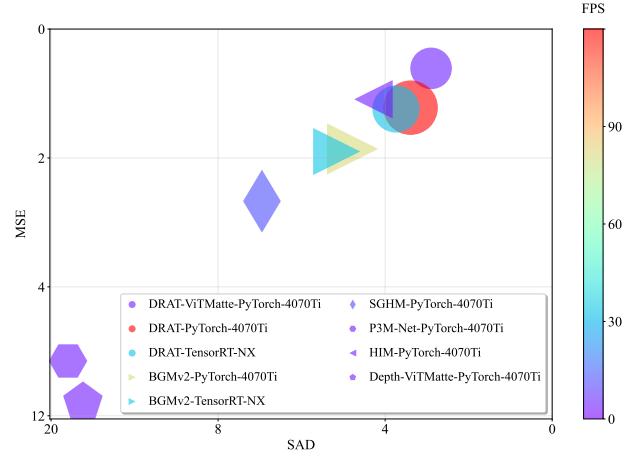


Fig. 4. Speed and accuracy comparison of involved matting methods. Better view in color.

#### C. Ablation study

The ablation study is shown in Tab. II from where we can see a consistent increasing trend of matting error when the proposed modules are removed from the DART algorithm one by one.

TABLE II  
THE ABLATION STUDY OF THE PROPOSED METHOD.

Methods	SAD ↓	MSE ↓	Grad ↓	Conn↓
DART + ViTMatte[24]	2.9	0.61	6.02	2.42
DART	3.39	1.22	8.89	3.33
DART-NoErrorMap	3.56	1.26	9.09	3.45
DART-NoDepthRefine	4.07	1.75	10.31	3.75
DART-NoErrorMap-NoDepthRefine	4.27	1.8	10.43	3.92

### IV. CONCLUSION

This paper introduces a fixed-background matting algorithm that is enhanced by depth information. By smartly exploiting the useful and complementary depth channel of an RGB-D image, the proposed DART algorithm achieves more accurate results compared with the existing SOTA matting approaches. Meanwhile, thanks to the successful distillation process, our method is fast: it runs at 125 FPS on a GPU-equipped desktop PC and 33 FPS on a mid-range edge-computing platform. To evaluate the proposed algorithm, we make a dedicated dataset, termed “JXNU-RGBD” for the RGB-D background matting task. To the best of our knowledge, this paper is the first work in the literature that explores the RGB-D matting problem with a fixed background. It paves the way to future better solutions for this new but realistic computer vision problem.

## REFERENCES

- [1] X. Shen, X. Tao, H. Gao, C. Zhou, and J. Jia, “Deep automatic portrait matting,” in *European Conference on Computer Vision*. Springer, 2016, pp. 92–107.
- [2] B. Zhu, Y. Chen, J. Wang, S. Liu, B. Zhang, and M. Tang, “Fast deep matting for portrait animation on mobile phone,” in *Proceedings of the 25th ACM international conference on Multimedia*, 2017, pp. 297–305.
- [3] Q. Chen, T. Ge, Y. Xu, Z. Zhang, X. Yang, and K. Gai, “Semantic human matting,” in *Proceedings of the 26th ACM international conference on Multimedia*, 2018, pp. 618–626.
- [4] 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 the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 8563–8572.
- [5] S. Sengupta, V. Jayaram, B. Curless, S. M. Seitz, and I. Kemelmacher-Shlizerman, “Background matting: The world is your green screen,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2020, pp. 2291–2300.
- [6] J. Li, J. Zhang, S. J. Maybank, and D. Tao, “Bridging composite and real: towards end-to-end deep image matting,” *International Journal of Computer Vision*, pp. 246–266, 2022.
- [7] Z. Ke, J. Sun, K. Li, Q. Yan, and R. W. Lau, “Modnet: Real-time trimap-free portrait matting via objective decomposition,” in *Proceedings of the AAAI Conference on Artificial Intelligence*, 2022, pp. 1140–1147.
- [8] Y. Sun, C.-K. Tang, and Y.-W. Tai, “Human instance matting via mutual guidance and multi-instance refinement,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 2647–2656.
- [9] Y. Dai, B. Price, H. Zhang, and C. Shen, “Boosting robustness of image matting with context assembling and strong data augmentation,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 11707–11716.
- [10] S. Ma, J. Li, J. Zhang, H. Zhang, and D. Tao, “Rethinking portrait matting with privacy preserving,” *International journal of computer vision*, pp. 1–26, 2023.
- [11] J. Li, J. Zhang, and D. Tao, “Deep image matting: A comprehensive survey,” *arXiv preprint arXiv:2304.04672*, 2023.
- [12] N. Xu, B. Price, S. Cohen, and T. Huang, “Deep image matting,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2017, pp. 2970–2979.
- [13] X. Fang, S.-H. Zhang, T. Chen, X. Wu, A. Shamir, and S.-M. Hu, “User-guided deep human image matting using arbitrary trimaps,” *IEEE Transactions on Image Processing*, pp. 2040–2052, 2022.
- [14] Y. Sun, C.-K. Tang, and Y.-W. Tai, “Ultrahigh resolution image/video matting with spatio-temporal sparsity,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 14112–14121.
- [15] G. Park, S. Son, J. Yoo, S. Kim, and N. Kwak, “Matteformer: Transformer-based image matting via prior-tokens,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2022, pp. 11696–11706.
- [16] Q. Liu, H. Xie, S. Zhang, B. Zhong, and R. Ji, “Long-range feature propagating for natural image matting,” in *Proceedings of the 29th ACM International Conference on Multimedia*, 2021, pp. 526–534.
- [17] Y. Zheng, Y. Yang, T. Che, S. Hou, W. Huang, Y. Gao, and P. Tan, “Image matting with deep gaussian process,” *IEEE Transactions on Neural Networks and Learning Systems*, 2022.
- [18] Y. Liu, J. Xie, X. Shi, Y. Qiao, Y. Huang, Y. Tang, and X. Yang, “Tripartite information mining and integration for image matting,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 7555–7564.
- [19] 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 the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2021, pp. 8762–8771.
- [20] G. Hinton, O. Vinyals, and J. Dean, “Distilling the knowledge in a neural network,” in *NIPS Deep Learning and Representation Learning Workshop*, 2015.
- [21] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, “Mobilenetv2: Inverted residuals and linear bottlenecks,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2018, pp. 4510–4520.
- [22] K. He, X. Zhang, S. Ren, and J. Sun, “Deep residual learning for image recognition,” in *Proceedings of the IEEE conference on computer vision and pattern recognition*, 2016, pp. 770–778.
- [23] C. Shu, Y. Liu, J. Gao, Z. Yan, and C. Shen, “Channel-wise knowledge distillation for dense prediction,” in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, 2021, pp. 5311–5320.
- [24] J. Yao, X. Wang, S. Yang, and B. Wang, “Vitmatte: Boosting image matting with pretrained plain vision transformers,” *arXiv preprint arXiv:2305.15272*, 2023.
- [25] K. Shen, C. Guo, M. Kaufmann, J. J. Zarate, J. Valentini, J. Song, and O. Hilliges, “X-avatar: Expressive human avatars,” in *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, 2023, pp. 16911–16921.
- [26] X. Chen, Y. Zhu, Y. Li, B. Fu, L. Sun, Y. Shan, and S. Liu, “Robust human matting via semantic guidance,” in *Proceedings of the Asian Conference on Computer Vision (ACCV)*, 2022, pp. 2984–2999.
- [27] S. Migacz, “8-bit inference with tensorrt,” in *GPU technology conference*, 2017, p. 5.