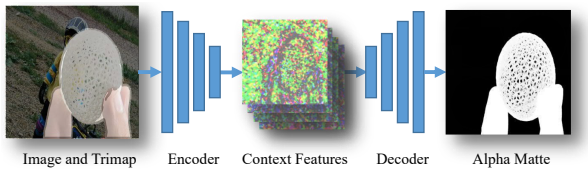
Revisiting Context Aggregation for Image Matting

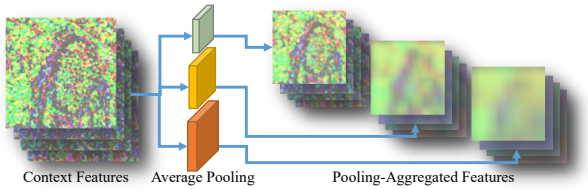
Qinglin Liu [1] **Xiaoqian Lv** [1] **Quanling Meng** [1] **Zonglin Li** [1] **Xiangyuan Lan** [2] **Shuo Yang** [3]
Shengping Zhang [1 2] **Liqiang Nie** [4]

Abstract

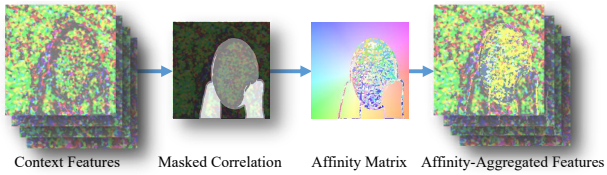
Traditional studies emphasize the significance of context information in improving matting performance. Consequently, deep learning-based matting methods delve into designing pooling or affinity-based context aggregation modules to achieve superior results. However, these modules cannot well handle the context scale shift caused by the difference in image size during training and inference, resulting in matting performance degradation. In this paper, we revisit the context aggregation mechanisms of matting networks and find that a basic encoder-decoder network without any context aggregation modules can actually learn more universal context aggregation, thereby achieving higher matting performance compared to existing methods. Building on this insight, we present AEMatter, a matting network that is straightforward yet very effective. AEMatter adopts a Hybrid-Transformer backbone with appearance-enhanced axis-wise learning (AEAL) blocks to build a basic network with strong context aggregation learning capability. Furthermore, AEMatter leverages a large image training strategy to assist the network in learning context aggregation from data. Extensive experiments on five popular matting datasets demonstrate that the proposed AEMatter outperforms state-of-the-art matting methods by a large margin. The source code is available at https://github.com/aipixel/AEMatter.



(a) Basic Encoder-Decoder Matting Network



(b) Pooling-based Context Aggregation Module



(c) Affinity-based Context Aggregation Module

Figure 1. Illustration of a basic matting network and context aggregation modules. (a) The basic matting network uses an encoder to extract context features from inputs, and a decoder to predict alpha mattes. Our AEMatter also follows this scheme. (b) Pooling-based context aggregation module uses pooling operations to aggregate contexts from surrounding regions. (c) Affinity-based context aggregation module uses affinity operations to aggregate contexts from globally related regions.

1. Introduction

Natural image matting is a classic problem that involves estimating the alpha matte of the foreground in a given image. This technology has numerous real-world applications, such as image editing (Chen et al., 2009; 2018) and film post-production (Gong et al., 2015; Wang et al., 2021). Formally, a given image $\boldsymbol{I}$ can be represented as a combination of a foreground $\boldsymbol{F}$ and background $\boldsymbol{B}$ as

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

where α_i is the alpha matte at pixel i. Therefore, matting involves the challenge of regressing alpha matte $\boldsymbol{\alpha}$ based on image $\boldsymbol{I}$. This process not only necessitates distinguishing

[1]School of Computer Science and Technology, Harbin Institute of Technology, Weihai, China [2]Peng Cheng Laboratory, Shenzhen, China [3]Department of Computer Science, The University of Hong Kong, Hong Kong, China [4]School of Computer Science and Technology, Harbin Institute of Technology, Shenzhen, China. Correspondence to: Shengping Zhang <s.zhang@hit.edu.cn>.

Proceedings of the 41st International Conference on Machine Learning, Vienna, Austria. PMLR 235, 2024.

between foreground and background but also determining the weights of the foreground, making it an intricate task.

To address the matting challenge, early researchers (Berman et al., 1998; Ruzon & Tomasi, 2000; Grady & Westermann, 2005; Levin et al., 2008) explore to estimate alpha mattes based on location and color similarity or by propagating color information within a local region. To improve matting performance, context aggregation technologies, such as global sampling or non-local propagation, are developed to leverage context information away from the foreground boundaries. Recently, deep learning-based methods (Xu et al., 2017; Lu et al., 2022) employ basic encoder-decoder networks to extract context features from input data and estimate alpha mattes, as depicted in Figure 1(a). Due to the formidable learning capability of neural networks, these methods outperform traditional matting methods by a substantial margin. To further improve prediction accuracy, researchers emulate traditional methods in designing context aggregation modules to effectively exploit context information (Li & Lu, 2020; Forte & Pitié, 2020). These modules adopt pooling or affinity based operations, as illustrated in Figures 1(b) and 1(c), to aggregate context information. However, it is rarely acknowledged that these modules cannot well handle the context scale shift caused by the difference in image size during training and inference, resulting in matting performance degradation.

In this paper, we revisit the context aggregation mechanisms of matting networks to inspire future research on high-performance matting methods. Specifically, we first evaluate existing matting networks, revealing that networks with context aggregation modules usually exhibit more errors when inferring on larger images, compared to networks without such modules. This observation underscores that while context aggregation modules can effectively aggregate contexts, their sensitivity to context scale restricts their universality. Subsequently, our assessment extends to basic encoder-decoder networks, where we observe their impressive performance. These results suggest that basic networks possess the capability to aggregate contexts for high-performance matting. Further exploration reveals that enhancing context aggregation capability can be achieved through training with large image patches and incorporating network layers with a larger receptive field. Building on these insights, we introduce AEMatter, a matting network that is both simpler and more powerful than existing methods. AEMatter adopts a Hybrid-Transformer backbone and integrates appearance-enhanced axis-wise learning (AEAL) blocks to build a basic network with strong context aggregation learning capability. Furthermore, AEMatter employs a large image training strategy to facilitate the network in learning context aggregation. Extensive experiments on five matting datasets demonstrate that AEMatter outperforms state-of-the-art methods by a large margin.

To summarize, the contributions of this paper are as follows:

- We pioneer an experimental analysis to evaluate the effectiveness and mechanisms of context aggregation modules within existing matting networks. Our findings reveal that while context aggregation modules can effectively aggregate contexts, their sensitivity to the context scale restricts their universality.
- We empirically find that basic encoder-decoder matting networks can learn to aggregate contexts for high-performance matting. Moreover, we demonstrate that this capability can be enhanced through training with large image patches and the adoption of network layers with a larger receptive field.
- We introduce AEMatter, a straightforward yet effective matting network that expands the receptive field with appearance-enhanced axis-wise learning (AEAL) blocks and is trained using large image patches. Experimental results demonstrate that AEMatter significantly outperforms state-of-the-art methods.

2. Related Work

Traditional matting methods. Traditional matting methods can be categorized into two approaches: sampling-based methods and propagation-based methods. Sampling-based methods involve sampling candidate foreground and background colors for pixels in unknown regions to estimate the alpha matte. Bayesian Matting(Chuang et al., 2001) models foreground and background colors with a Gaussian distribution and incorporates spatial location information to enhance accuracy. Global Matting (He et al., 2011) takes a different approach by sampling pixels in all known regions to prevent information loss and improve robustness. Propagation-based methods rely on the assumption that foreground and background colors exhibit smoothness in local regions for alpha matte estimation. Poisson Matting (Sun et al., 2004) utilizes boundary information from trimap to solve the Poisson equation, making it capable of estimating the alpha matte even with a rough trimap. Closed-form matting (Levin et al., 2008) introduces a color-line assumption and provides a closed-form solution for estimation.

Deep learning-based matting methods. Deep learning-based methods train the networks on image matting datasets to estimate the alpha matte. Early methods (Xu et al., 2017; Lu et al., 2022) typically employ a basic encoder-decoder network for matting. DIM (Xu et al., 2017) introduced a refinement module to the decoder to improve the performance. IndexNet (Lu et al., 2022) retains the indices of the downsampled features for improving the gradient accuracy. Recent advancements in deep image matting methods have designed pooling-based or affinity-based context aggregation modules to refine context features and adopt other

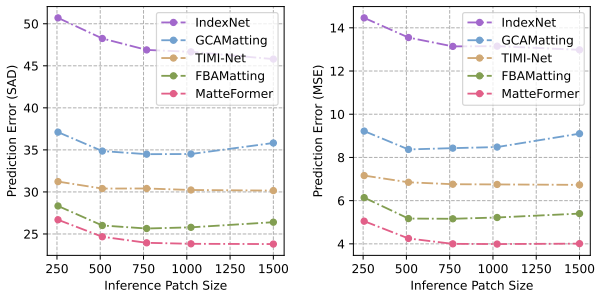


Figure 2. **Inference Patch Size** vs **Prediction Errors**. As the inference patch size increases, the prediction errors of the compared matting methods first decrease and then show different trends.

Table 1. Comparison of state-of-the-art matting methods trained on Adobe Composition-1K using image patches of different sizes.

Method	Patch Size	SAD	MSE	Grad	Conn
IndexNet (Lu et al., 2022)	256	38.52	8.74	18.02	36.43
IndexNet (Lu et al., 2022)	512	33.64	7.05	14.35	30.21
IndexNet (Lu et al., 2022)	768	31.12	6.40	12.83	27.63
IndexNet (Lu et al., 2022)	1024	30.91	6.73	13.72	27.17
FBAMatting (Forte & Pitié, 2020)	256	43.18	10.41	21.13	42.39
FBAMatting (Forte & Pitié, 2020)	512	33.36	7.26	15.75	29.84
FBAMatting (Forte & Pitié, 2020)	768	29.89	5.73	14.05	26.18
FBAMatting (Forte & Pitié, 2020)	1024	30.76	5.74	15.19	27.03
MatteFormer (Park et al., 2022)	256	28.52	5.51	12.00	24.06
MatteFormer (Park et al., 2022)	512	23.61	3.78	9.23	18.52
MatteFormer (Park et al., 2022)	768	22.78	3.59	8.38	17.50
MatteFormer (Park et al., 2022)	1024	23.68	3.62	8.81	18.66

techniques to improve performance. Pooling-based methods (Forte & Pitié, 2020; Yu et al., 2021a; Sun et al., 2021; Liu et al., 2021a; Park et al., 2022; Cai et al., 2022) use average pooling to aggregate contexts from surrounding regions for context feature refinement. FBAMatting (Forte & Pitié, 2020) adopts pyramid pooling module (PPM) (Zhao et al., 2017) and introduces the groupnorm (Wu & He, 2018) and weight standardization (Qiao et al., 2019) tricks to improve the matting performance. MGMatting (Yu et al., 2021a) adopts ASPP and designs a progressive refinement decoder to estimate fine alpha mattes from coarse segmentation. MatteFormer (Park et al., 2022) proposes a trimap-guided token pooling module and adopts the Swin-Tiny (Liu et al., 2021c) backbone to improve the prediction. Affinity-based methods (Li & Lu, 2020; Yu et al., 2021; Yu et al., 2021b; Dai et al., 2022) use the masked correlation to construct an affinity matrix and enhance the context features with the contexts from globally related regions. GCAMatting (Li & Lu, 2020) adopts the guided context attention module to improve the prediction in the transparent region. TIMI-Net (Liu et al., 2021b) proposes a tripartite information module and multi-branch architecture to improve predictions.

3. Empirical Study

In this section, we perform experimental analyses on existing matting networks and basic encoder-decoder matting networks to explore the context aggregation mechanisms of matting networks and identify the key factors contributing to the performance of matting networks.

3.1. Exploring Existing Matting Networks

We assess the performance and robustness of existing matting networks, observing that both the encoder-decoder and the context aggregation module within these networks can effectively aggregate contexts for matting. Nevertheless, the sensitivity of context aggregation modules to context scale restricts their universality.

Patch-based Inference. Existing matting networks usually include an encoder-decoder network with a context aggregation module. The context aggregation modules, built with hard-crafted structures, are considered to exhibit better context aggregation capability across images of various sizes compared to the encoder-decoder network. To validate this understanding, we conduct a patch-based inference evaluation for existing matting networks. We evaluate existing matting methods, including IndexNet (Lu et al., 2022) without a context aggregation module and GCAMatting (Li & Lu, 2020), TIMI-Net (Liu et al., 2021b), FBAMatting (Forte & Pitié, 2020), and MatteFormer (Park et al., 2022) with a context aggregation module. The evaluation was conducted on image patches of varying sizes, ranging from 256×256, 512×512, 768×768, and 1024×1024, and on the whole images. As the results summarized in Figure 2, the IndexNet method without context aggregation modules exhibits a monotonically decreasing error trend. In contrast, the matting methods with context aggregation modules experience a reduction in errors initially as the patch size increases, followed by a subsequent increase or stabilization. This observation contradicts our understanding and suggests that both the encoder-decoder network and context aggregation modules help aggregate contexts. However, it is evident that context aggregation modules are highly sensitive to the variations in context scale due to the differences in image sizes between the training and inference phases. This sensitivity proves detrimental to the performance of matting networks employing such modules.

Patch-based Training. Matting networks learn to aggregate context information from the data, during the training phase. The context aggregation modules in the network, with a larger receptive field compared to the network layers in the encoder-decoder, are believed to enhance the utilization of context information for better predictions. To validate this understanding, we evaluate matting networks with and without context aggregation modules that are trained on image patches of different sizes. Specifically, we evaluated IndexNet without a context aggregation module, and FBA-

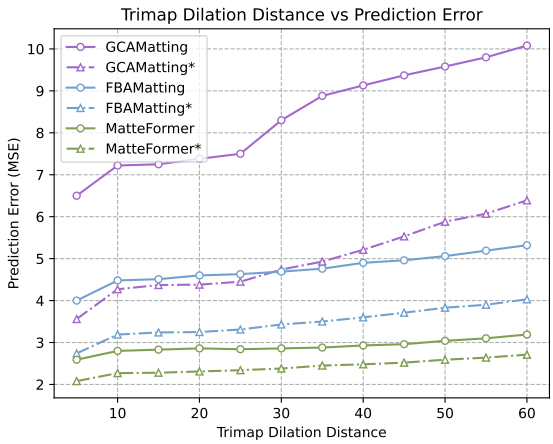


Figure 3. **Trimap Dilation Distance** vs **Prediction Error**. Note that, * denotes the network does not incorporate context aggregation modules. As the trimap dilation distance increases, the prediction errors (MSE) of all compared matting methods increase.

Matting as well as MatteFormer with a context aggregation module. All compared methods are first trained on image patches with sizes of 256×256, 512×512, 768×768, and 1024×1024, and then evaluated on the validation set. Note that, we train more epochs for those networks that are trained on smaller image patches. The results are summarized in Table 1. Remarkably, we a decrease in error for all networks with an increase in patch size, signifying the advantageous impact of larger training data sizes on matting networks. Furthermore, the performance of FBAMatting and MatteFormer, both having context aggregation modules, does not show further improvement beyond the training image sizes specified in their papers, which suggests that the context aggregation modules are limited by manually tuned designs, thereby restricting their universality.

Robustness to Coarse Trimap. Recent advancements in matting research (Yu et al., 2021a; Dai et al., 2022) underscore the importance of robustness to coarse trimaps as a critical performance metric. To assess the impact of context aggregation modules on handling coarse trimap scenarios, we evaluate existing state-of-the-art matting methods, including GCAMatting, FBAMatting, and MatterFormer, on a modified Adobe Composition-1K dataset featuring trimaps with varying dilation distances. The trimap annotations of this dataset are generated by applying morphological erosion and dilation operations to the ground truth. Additionally, we evaluate the network variants without context aggregation modules. The network variants are trained on 1024×1024 image patches. In Figure 3, we present the results of compared methods, where * denotes network variants without context aggregation modules. As depicted in the figure, the performance trend of all matting networks consistently degrades as the dilation distance increases, suggesting that the robustness to coarse trimaps is correlated with the encoder-decoder architecture rather than the presence of context aggregation modules. Furthermore, matting methods with context aggregation modules do not outperform basic networks without such modules, further highlighting their limited universality due to the sensitivity of context aggregation modules to context scale.

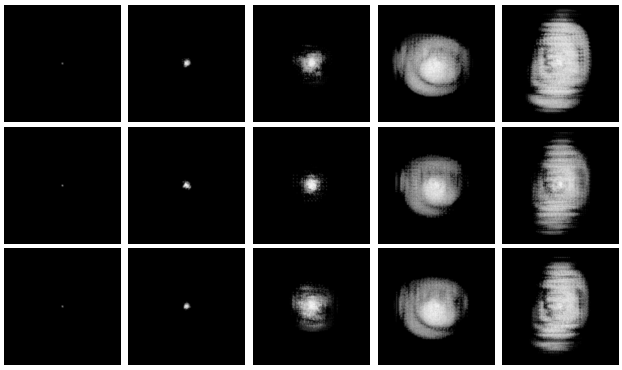
Table 2. Comparison of the basic matting networks with state-of-the-art matting methods on Adobe Composition-1K. * denotes the backbone adopts the dilated convolution trick.

Method	Backbone	SAD	MSE	Grad	Conn
IndexNet (Lu et al., 2022)	MobileNet	45.80	13.00	25.90	43.70
BasicNet (Ours)	MobileNet	**30.91**	**6.73**	**13.72**	**27.17**
GCAMatting (Li & Lu, 2020)	ResNet-34	35.28	9.00	16.90	32.50
A2UNet (Dai et al., 2021)	ResNet-34	32.10	7.80	16.33	29.00
TIMI-Net (Liu et al., 2021b)	ResNet-34	29.08	6.00	11.50	25.36
BasicNet (Ours)	ResNet-34	**28.08**	**5.06**	**11.39**	**24.32**
SIM (Sun et al., 2021)	ResNet-50*	28.00	5.80	10.8	24.80
FBAMatting (Forte & Pitié, 2020)	ResNet-50*	26.40	5.40	10.6	21.50
BasicNet (Ours)	ResNet-50	**23.82**	**4.27**	**8.08**	**19.02**
Transmatting (Cai et al., 2022)	Swin-Tiny	26.83	5.22	10.62	22.14
MatteFormer (Park et al., 2022)	Swin-Tiny	23.80	4.03	8.68	18.90
BasicNet (Ours)	Swin-Tiny	**19.72**	**2.97**	**6.27**	**14.43**

3.2. Exploring Basic Matting Networks

Based on the above experiments, we observe that the encoder-decoder component in matting networks is less sensitive to context scale compared to the context aggregation modules, indicating better universality. To explore the feasibility of building basic matting networks using encoder-decoder, we delve into evaluating basic encoder-decoder networks with various configurations.

Performance of Basic Matting Networks. We first evaluate the performance of the basic encoder-decoder matting network without context aggregation modules. Specifically, we adopt the MobileNet (Sandler et al., 2018), ResNet-34 (He et al., 2016), ResNet-50 (He et al., 2016), and Swin-Tiny (Liu et al., 2021c) backbones to construct basic matting networks without any context aggregation modules. Note that, we simply adopt IndexNet as the MobileNet based basic matting network. Then, we follow the training pipeline of TIMI-Net to train these basic matting networks on image patches with the size of 1024×1024. Finally, we compare these basic networks with state-of-the-art networks including, IndexNet, GCAMatting, FBAMatting, A2UNet (Dai et al., 2021), TIMI-Net, FBAMAtting, and MatteFormer. As shown in Table 2, the basic matting networks (referred to as BasicNet) outperform state-of-the-art methods, which suggests the feasibility of building basic matting networks using encoder-decoder. Furthermore, the Swin-Tiny and ResNet-50 based networks outperform the MobileNet and ResNet-34 based networks, which suggests that basic matting networks with a larger receptive field may learn better context aggregation to achieve higher performance.



(a) (b) (c) (d) (e)

Figure 4. Visualization of the receptive field of matting networks trained on image patches of different sizes. (a) Untrained network. (b) Network trained on 256×256 patches. (c) Network trained on 512×512 patches. (d) Network trained on 768×768 patches. (e) Network trained on 1024×1024 patches.

Table 3. Experiment on the training image patch sizes.

Backbone	Patch Size	SAD	MSE	Grad	Conn
Resnet-34 (He et al., 2016)	256	41.74	12.51	22.51	40.14
Resnet-34 (He et al., 2016)	512	33.16	7.08	15.27	29.80
Resnet-34 (He et al., 2016)	768	27.70	5.41	11.23	23.89
Resnet-34 (He et al., 2016)	1024	28.08	5.06	11.39	24.32
Swin-Tiny (Liu et al., 2021c)	256	27.99	5.30	11.23	23.96
Swin-Tiny (Liu et al., 2021c)	512	22.42	3.72	7.46	17.54
Swin-Tiny (Liu et al., 2021c)	768	20.37	2.96	6.55	16.89
Swin-Tiny (Liu et al., 2021c)	1024	19.72	2.97	6.27	14.43

Training Image Patch Sizes. In our previous experiments on existing matting methods, we observe that matting networks trained with larger image patches may achieve better performance. To explore whether basic matting networks can benefit from large training images, we train the ResNet-34 (He et al., 2016) and Swin-Tiny (Liu et al., 2021c) based basic matting networks with image patches of various sizes, including 256×256, 512×512, 768×768, and 1024×1024. Subsequently, we evaluate the performance of these networks. The results, presented in Table 3, confirm that the performance of matting networks improves with larger training image patches, providing empirical backing for our hypothesis. To delve deeper into the impact of training image patch sizes on matting networks, we employ the methodology proposed by Luo et al.(Luo et al., 2016) to visualize the effective receptive field of ResNet-34 based networks trained on image patches of different sizes using gradient feedback, as shown in Figure4. The visualization demonstrates that basic matting networks can learn enhanced context aggregation from large image patches.

Receptive Field of Network Layers. In our assessment of basic matting networks, we observe a positive correlation between larger receptive fields and improved network performance. This observation leads us to hypothesize that the context aggregation capability of a network is positively correlated with its receptive field size. To verify this hypothesis, we compare the performance of basic matting networks with different kernel sizes. Specifically, we build basic matting networks with ResNet-34 and ResNet-50 backbones. Then, we replace half of 3×3 convolutions in these networks with 1×1 convolutions and 5×5 convolutions to control the receptive field. Finally, we evaluate the modified networks and summarize the results in Table 4. The results indicate that matting networks with larger convolution kernels achieve better performance, providing evidence that supports our hypothesis that networks with larger receptive fields exhibit enhanced context aggregation capability.

Table 4. Experiment on the convolution kernel sizes.

Backbone	Kernel Size	SAD	MSE	Grad	Conn
Resnet-34 (He et al., 2016)	1×1	31.28	6.14	13.41	28.05
Resnet-34 (He et al., 2016)	3×3	28.08	5.06	11.39	24.32
Resnet-34 (He et al., 2016)	5×5	26.72	4.74	10.08	22.75
Resnet-50 (He et al., 2016)	1×1	28.70	5.79	10.96	24.98
Resnet-50 (He et al., 2016)	3×3	23.82	4.27	8.08	19.02
Resnet-50 (He et al., 2016)	5×5	23.34	3.92	7.42	18.89

3.3. Experimental Findings

Based on the above results, we distill two insights to help design effective matting networks: (1). Due to manual designs, context aggregation modules are sensitive to changes in context scale, leading to a lack of universality. (2). Basic encoder-decoder networks possess the capability to learn universal context aggregation. This capability can be further enhanced through training with large image patches and incorporating network layers with a large receptive field.

4. Proposed Method

Based on our findings, we present a simple yet effective matting network, named AEMatter. AEMatter adopts a Hybrid-Transformer backbone with appearance-enhanced axis-wise learning blocks to build a basic network with strong context aggregation learning capability, as illustrated in Figure 5. Additionally, AEMatter leverages a large image training strategy to help learn context aggregation.

4.1. Encoder

To extract low-level features and context features from the inputs and enlarge the receptive field of AEMatter, we adopt a Hybrid-Transformer backbone with appearance-enhanced axis-wise learning blocks to construct the encoder.

Hybrid-Transformer Backbone. Although the Swin-Tiny (Liu et al., 2021c) based matting network performs best in the above experiments, Swin-Tiny is primarily designed for high-level semantic tasks and ignores extracting low-level features, which limits its effectiveness in image matting. Prior studies (Park et al., 2022; Dai et al., 2022)

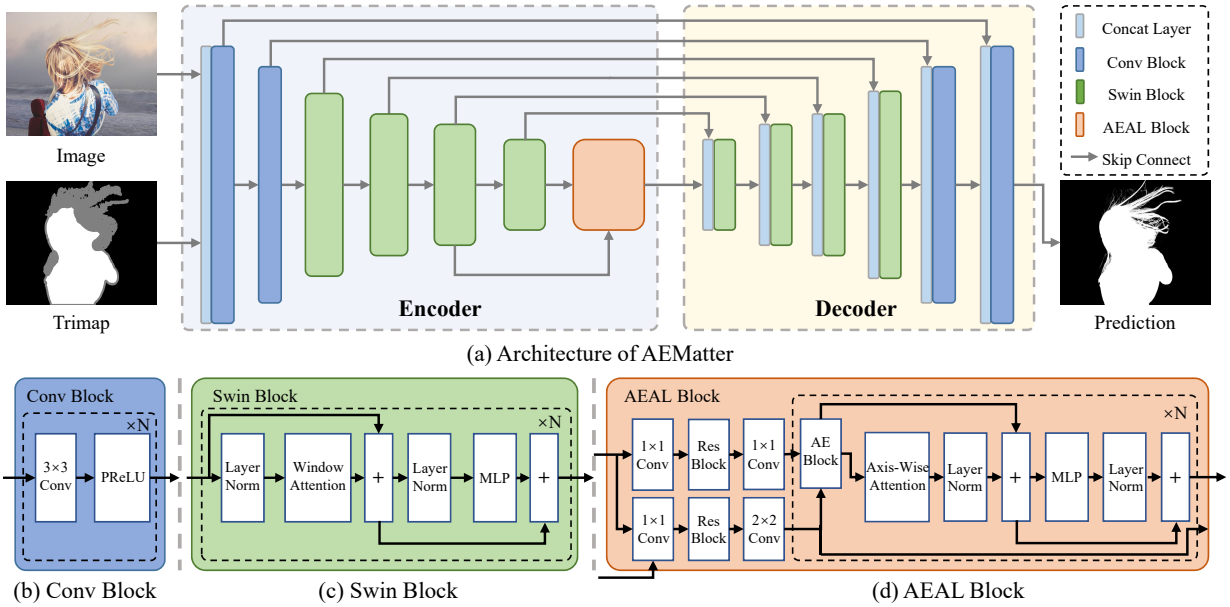


Figure 5. Overview of AEMatter. The encoder adopts a Hybrid-Transformer backbone with appearance-enhanced axis-wise learning blocks to extract context features. The decoder adopts Swin blocks to refine the context features and estimate the alpha matte.

address this issue by incorporating additional shortcut modules to extract low-level features, but their backbones cannot utilize the shortcut features, resulting in subpar performance. In contrast, we replace the patch-embedding stem with convolution blocks to extract rich low-level features. The structure of the convolution block is illustrated in Figure 5(b). To preserve the image details, we omit the normalization layers in the stem as they affect the information in local regions, which hurts the matting performance. In addition, we incorporate PReLU (He et al., 2015) as the activation function, which introduces learnable negative slopes to facilitate network training. Afterward, we use the Swin blocks of Swin-Tiny to extract high-level context features.

Appearance-Enhanced Axis-Wise Learning. The backbone of AEMatter adopts a hierarchical structure that is effective in capturing and integrating context features across large spatial regions. However, the receptive field of the Swin blocks adopted is still not large enough to cover high-resolution images, which limits the context aggregation capability of the matting network, resulting in sub-optimal performance. While one possible solution is to employ many downsampling layers and Swin blocks to extract context features across larger regions, such an approach can hinder the training and increase the risk of overfitting. To address this issue, we incorporate a few appearance-enhanced axis-wise learning (AEAL) blocks after the backbone, which leverages an appearance-enhanced (AE) block to facilitate training and axis-wise attention to enlarge the receptive fields.

The structure of the AEAL block is illustrated in Figure 5(d). To mitigate high computational overheads incurred by the high-dimension context features from the backbone, we use residual blocks and 1×1 convolutions to produce the compact context features $\boldsymbol{F}_c$ from the fourth-stage features $\boldsymbol{F}_4$ of the backbone. Additionally, we use $\boldsymbol{F}_4$ to guide the extraction of appearance features from third-stage features $\boldsymbol{F}_3$ of the backbone with convolution and residual blocks, generating the context-guided appearance features $\boldsymbol{F}_a$. Subsequently, we employ three cascaded learning modules to process $\boldsymbol{F}_c$ and $\boldsymbol{F}_a$. To facilitate network training, we first introduce an AE block to generate the appearance-enhanced context features $\boldsymbol{F}_{ac}$ with $\boldsymbol{F}_c$ and $\boldsymbol{F}_a$ as

$$\boldsymbol{F}_{ac} = \boldsymbol{F}_c + \mathrm{Conv}(\mathrm{Res}(\mathrm{Conv}(\mathrm{Cat}(\boldsymbol{F}_c, \boldsymbol{F}_a)))) \quad (2)$$

where $\mathrm{Cat}(\cdot,\cdot)$, $\mathrm{Conv}(\cdot)$, and $\mathrm{Res}(\cdot)$ denote the concatenation, 1×1 convolution, residual block, respectively. To capture context features over large regions, we propose axis-wise attention, which divides $\boldsymbol{F}_{ac}$ into axis-wise rectangular regions and then applies multi-head self-attention. Specifically, we first zero-pad $\boldsymbol{F}_{ac}$ to a size that is an integer multiple of width W and split the padded feature $\boldsymbol{F}_{acp}$ into features $\boldsymbol{F}_{acpx}$ and $\boldsymbol{F}_{acpy}$ along the channel dimension as

$$(\boldsymbol{F}_{acpx}, \boldsymbol{F}_{acpy}) = \mathrm{Split}_{\mathrm{Channel}}(\mathrm{Pad}(\boldsymbol{F}_{ac})) \quad (3)$$

where $\mathrm{Split}_{\mathrm{Channel}}(\cdot)$ and $\mathrm{Pad}(\cdot)$ denote the channel wise splitting and zero padding, respectively. Next, we further split $\boldsymbol{F}_{acpx}$ and $\boldsymbol{F}_{acpy}$ into two sets of axis-wise features, applying multi-head self-attention to extract context features over large regions. These features are then reassembled to form the refined context feature $\boldsymbol{F}_{rc}$ as:

$$\begin{aligned} \boldsymbol{F}_{rc} = &\mathrm{Cat}(\mathrm{MHA}(\mathrm{Split}_{\text{Axis-X}}(\boldsymbol{F}_{acpx}(X))), \\ &\mathrm{MHA}(\mathrm{Split}_{\text{Axis-Y}}(\boldsymbol{F}_{acpy}(X)))) \end{aligned} \quad (4)$$

where MHA$(\cdot)$ denotes the multi-head attention operation. Split$_{\text{Axis-X}}(\cdot)$ and Split$_{\text{Axis-Y}}(\cdot)$ denote the x-axis wise splitting and y-axis wise splitting, respectively. Finally, we utilize the MLP network as the feed-forward network (FFN) for feature transformation, following the vanilla Transformer (Vaswani et al., 2017).

4.2. Decoder

To enlarge the receptive field of AEMatter and improve the alpha matte estimation, we adopt a Transformer-based decoder that employs Swin blocks which have a large receptive field to refine the context features from the encoder. Specifically, we first concatenate the refined context feature $\boldsymbol{F}_{rc}$ with the fourth-stage features $\boldsymbol{F}_4$ from the encoder, and apply Swin blocks to generate the initial decoder feature $\boldsymbol{F}_d$. We then upsample $\boldsymbol{F}_d$ and concatenate it with the features of the corresponding scale of the encoder, and apply another Swin block for feature refinement. This process is repeated three times to obtain the refined decoder features $\boldsymbol{F}_{rd}$. To fuse the image details for alpha matte estimation, we upsample $\boldsymbol{F}_{rd}$ and concatenate it with the low-level features extracted by the stem of the encoder, and process it using convolution blocks that omit the normalization layers to prevent the mean or variance of the whole feature map from affecting the estimation in local regions. We perform this process twice and then use a 3×3 convolution to predict the alpha matte $\boldsymbol{\alpha}$. Finally, we clip the predicted alpha matte $\boldsymbol{\alpha}$ to the range of 0 to 1.

4.3. Training Strategy

In our empirical study, we observe that basic encoder-decoder networks can acquire better context aggregation capability when trained on larger image patches, leading to improved matting performance. Therefore, we propose to train the AEMatter network on 1024×1024 image patches, which are larger than the existing methods. To help AEMatter learn to predict alpha mattes, we define the loss function as

$$\mathcal{L}_{\alpha} = \mathcal{L}_{l1} + \mathcal{L}_{cb} + \mathcal{L}_{lap} \tag{5}$$

where $\mathcal{L}_{l1}$, $\mathcal{L}_{cb}$, and $\mathcal{L}_{lap}$ are the L1 loss, Charbonnier L1 loss, and Laplacian loss, which are defined as

$$\mathcal{L}_{l1} = \left|\boldsymbol{\alpha} - \boldsymbol{\alpha}^{gt}\right| \tag{6}$$

$$\mathcal{L}_{cb} = \frac{1}{|\mathcal{T}^U|} \sum_{i \in \mathcal{T}^U} \sqrt{(\alpha_i - \alpha_i^{gt})^2 + \epsilon^2} \tag{7}$$

$$\mathcal{L}_{lap} = \sum_{j} 2^j \left|\mathrm{L}_j(\boldsymbol{\alpha}) - \mathrm{L}_j(\boldsymbol{\alpha}^{gt})\right| \tag{8}$$

where $\boldsymbol{\alpha}$ and $\boldsymbol{\alpha}^{gt}$ are the predicted alpha matte and ground truth alpha matte of the input image $\boldsymbol{I}$, respectively. Additionally, we adopt training data augmentation techniques, similar to those employed by FBAMatting and MGMatting, to enhance the matting performance.

Table 5. Quantitative results on Adobe Composition-1K. TTA denotes the method adopts the test-time augmentation trick.

Method	SAD	MSE	Grad	Conn
DIM (Xu et al., 2017)	50.40	17.00	36.70	55.30
IndexNet (Lu et al., 2022)	45.80	13.00	25.90	43.70
GCAMatting (Li & Lu, 2020)	35.28	9.00	16.90	32.50
TIMI-Net (Liu et al., 2021b)	29.08	6.00	11.50	25.36
SIM (Sun et al., 2021)	27.70	5.60	10.70	24.40
FBAMatting (Forte & Pitié, 2020)	26.40	5.40	10.60	21.50
TransMatting (Cai et al., 2022)	24.96	4.58	9.72	20.16
LFPNet (Liu et al., 2021a)	23.60	4.10	8.40	18.50
MatteFormer (Park et al., 2022)	23.80	4.03	8.68	18.90
dugMatting (Wu et al., 2023)	23.40	3.90	7.20	18.80
DiffusionMat (Xu et al., 2023)	22.80	4.00	6.80	18.40
DiffMatte-ViTS (Hu et al., 2023)	20.52	3.06	7.05	14.85
ViTMatte-B (Yao et al., 2024)	20.33	3.00	6.74	14.78
AEMatter (Ours)	17.53	2.26	4.76	12.46
AEMatter + TTA (Ours)	**16.89**	**2.06**	**4.24**	**11.72**

Table 6. Generalization results on Distinction-646. All methods are trained on Adobe Composition-1K.

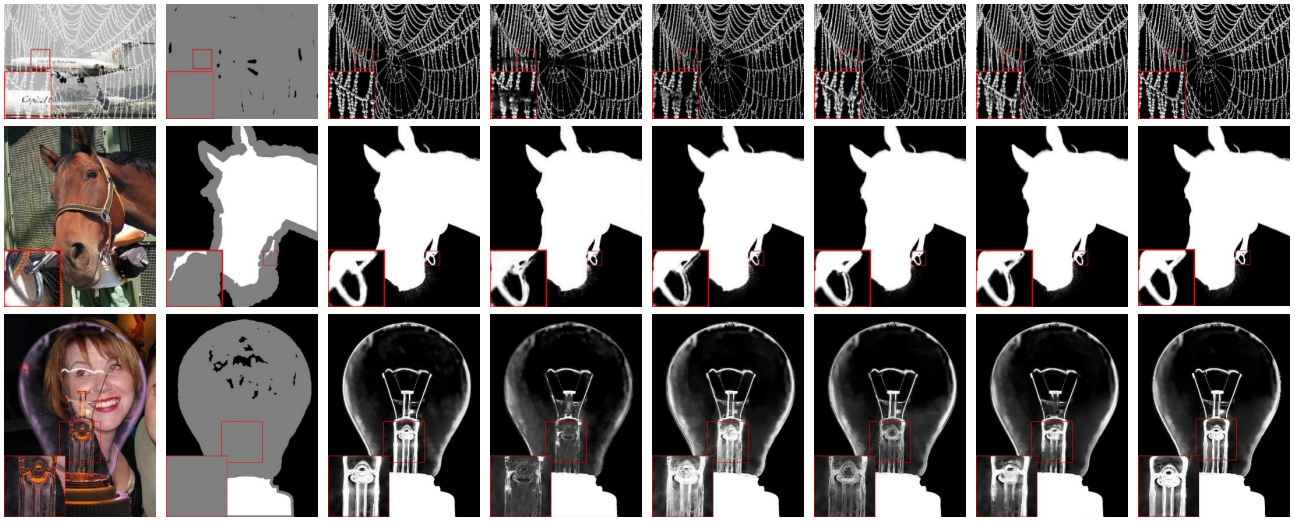
Method	SAD	MSE	Grad	Conn
DIM (Xu et al., 2017)	63.88	25.77	53.23	66.31
IndexNet (Lu et al., 2022)	44.93	9.23	41.30	44.86
TIMI-Net (Liu et al., 2021b)	42.61	7.75	45.05	42.40
GCAMatting (Li & Lu, 2020)	36.37	8.19	32.34	36.00
FBAMatting (Forte & Pitié, 2020)	32.28	5.66	25.52	32.39
LFPNet (Liu et al., 2021a)	22.36	3.41	14.92	20.50
Matteformer (Park et al., 2022)	23.60	3.12	13.56	21.56
AEMatter (Ours)	**16.95**	**1.81**	**8.28**	**14.83**

5. Experiments

In this section, we compare the performance of AEMatter with existing matting methods on the Adobe Composition-1K dataset. Additionally, we evaluate the generalization ability of AEMatter on the Distinctions-646 (Qiao et al., 2020), Transparent-460 (Cai et al., 2022), Semantic Image Matting (Sun et al., 2021), and Automatic Image Matting-500 (Li et al., 2021) datasets. More experimental results and ablation studies are provided in the appendix.

5.1. Results on Adobe Composition-1K

We compare AEMatter against state-of-the-art methods, such as MatteFormer (Park et al., 2022), dugMatting (Wu et al., 2023),and ViTMatte-B (Yao et al., 2024) on the Adobe Composition-1K dataset. Table 5 and Figure 13 summarize the quantitative and qualitative results of all compared methods. TTA denotes the method that adopts the test-time augmentation trick. Quantitative results show that AEMatter significantly outperforms state-of-the-art methods in terms



(a) Image (b) Trimap (c) Ground Truth (d) IndexNet (e) FBAMatting (f) LFPNet (g) MatteFormer (h) AEMatter (Ours)

Figure 6. Qualitative comparison of the alpha matte results on the Adobe Composition-1K dataset.

Table 7. Generalization results on Transparent-460. All methods are trained on Adobe Composition-1K.

Method	SAD	MSE	Grad	Conn
DIM (Xu et al., 2017)	356.20	49.68	146.46	296.31
IndexNet (Lu et al., 2022)	434.14	74.73	124.98	368.48
TIMI-Net (Liu et al., 2021b)	328.08	44.20	142.11	289.79
MGMatting (Yu et al., 2021a)	344.65	57.25	74.54	282.79
TransMatting (Cai et al., 2022)	192.36	20.96	41.80	158.37
AEMatter (Ours)	**122.27**	**6.92**	**27.42**	**112.02**

Table 8. Generalization results on Semantic Image Matting. All methods are trained on Adobe Composition-1K.

Method	SAD	MSE	Grad	Conn
DIM (Xu et al., 2017)	95.96	54.25	29.84	100.65
IndexNet (Lu et al., 2022)	66.89	25.75	22.07	67.61
GCAMatting (Li & Lu, 2020)	51.84	19.46	24.16	51.98
FBAMatting (Forte & Pitié, 2020)	26.87	5.61	9.17	22.87
TIMI-Net (Liu et al., 2021b)	54.08	16.59	18.91	53.79
LFPNet (Liu et al., 2021a)	23.05	4.28	23.30	18.19
Matteformer (Park et al., 2022)	23.90	4.73	7.72	19.01
AEMatter (Ours)	**19.51**	**2.82**	**4.62**	**14.37**

of SAD, MSE, Grad, and Conn metrics. Furthermore, qualitative results show AEMatter delivers a visually appealing alpha matte, especially in regions where the foreground and background colors are similar.

5.2. Generalization on Various Datasets

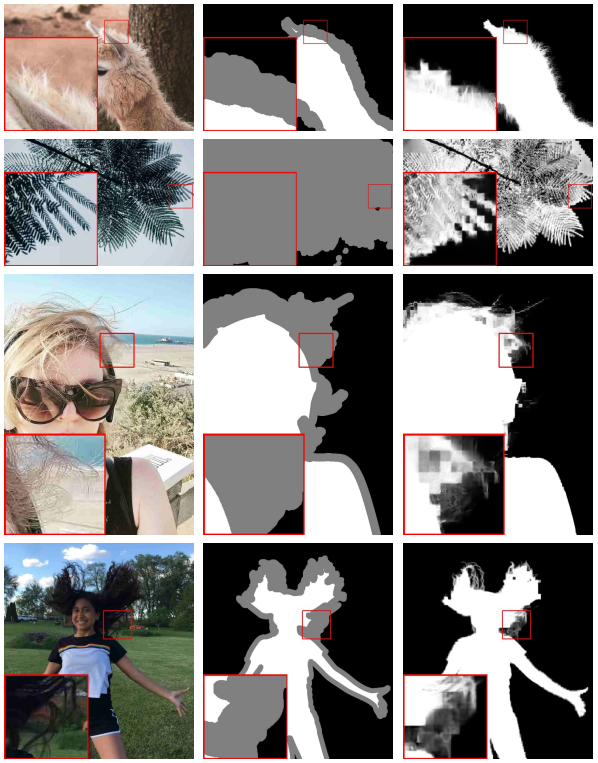
To evaluate the generalization ability of AEMatter, we compare AEMatter against existing matting methods on the Distinctions-646, Transparent-460, Semantic Image Matting, and Automatic Image Matting-500 datasets. It should be noted that all compared matting methods are pre-trained on the Adobe Composition-1K dataset for fair comparison. We evaluate all compared methods and summarized the quantitative results in Tables 6, 7, 8, and 9. The quantitative results underscore the significant performance advantages of AEMatter compared to existing methods, indicative of its exceptional generalization ability.

Table 9. Generalization results on Automatic Image Matting-500. All methods are trained on Adobe Composition-1K.

Method	SAD	MSE	Grad	Conn
DIM (Xu et al., 2017)	39.97	52.83	28.92	40.66
IndexNet (Lu et al., 2022)	26.95	26.32	16.41	26.25
GCAMatting (Li & Lu, 2020)	34.78	38.93	25.73	35.14
SIM (Sun et al., 2021)	27.05	31.10	23.68	27.08
FBAMatting (Forte & Pitié, 2020)	19.43	16.37	12.65	18.75
Matteformer (Park et al., 2022)	26.87	29.00	23.00	26.63
AEMatter (Ours)	**14.76**	**11.69**	**11.20**	**14.19**

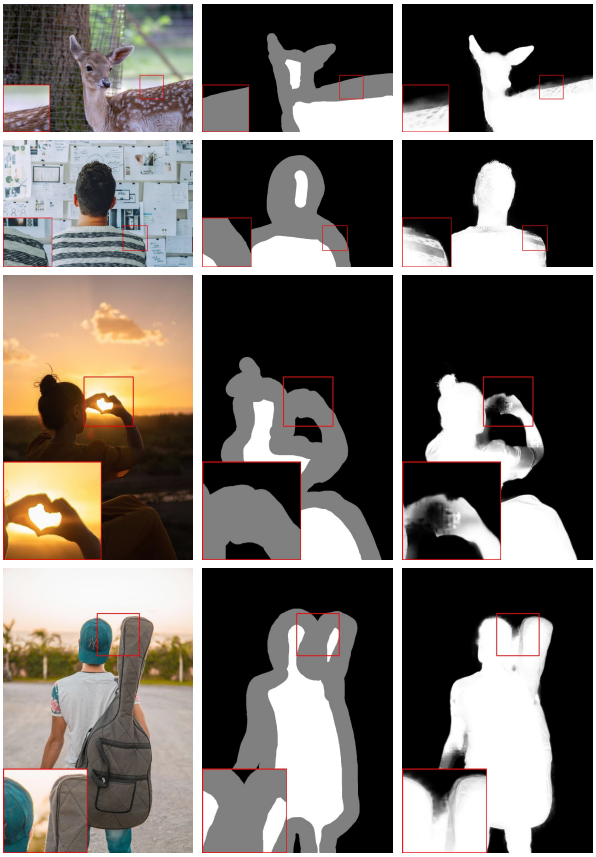
6. Limitations

Although AEMatter performs well on multiple datasets, we have identified some limitations when applying it to real-world scenarios, which could be due to the network learning non-generalizable differences between foreground and background from synthetic data. Specifically, AEMatter may exhibit lower matting accuracy in scenarios where the image is affected by degradation such as JPEG compression or lens blur, or where the trimap is coarse. Qualitative results illustrating these failure cases are shown in Figures 7 and 8. Future research directions could include adopting



(a) Image (b) Trimap (c) Alpha Matte

Figure 7. Qualitative results on real-world degraded images. AEMatter has lower accuracy at the boundary regions.



(a) Image (b) Trimap (c) Alpha Matte

Figure 8. Qualitative results on images with coarse trimaps. AEMatter cannot accurately predict the foreground and background.

strong data augmentation strategies or domain adaptation techniques and exploring auxiliary semantics networks and image restoration networks to address these limitations.

7. Conclusion

In this paper, we revisit the context aggregation mechanisms of matting networks and discover that discover that a basic encoder-decoder network itself can learn universal context aggregations to achieve high matting performance. Specifically, we experimentally reveal that while context aggregation modules can effectively aggregate contexts, their sensitivity to context scale restricts the universality. Simultaneously, we notice that basic encoder-decoder networks can learn context aggregation, leading to impressive matting performance. Further exploration uncovers that enhancing the context aggregation capability of the network can be achieved through training using large image patches and adopting network layers with a larger receptive field. Building upon these insights, we introduce a simple yet very effective matting network, named AEMatter, which expands the receptive field of the network with simple structures and is trained using large image patches. Experimental results on five datasets demonstrate our AEMatter outperforms state-of-the-art matting methods by a large margin.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China (Nos. 62272134, 62236003 and 62072141), the Shenzhen College Stability Support Plan (No. GXWD20220817144428005) and Peng Cheng Laboratory (PCL2023A10-2 and PCL2023A08).

Impact Statement

This paper conducts an analysis of the context aggregation mechanisms in deep image matting methods, presenting a straightforward yet highly effective design and training strategy for matting networks. The insights derived from this paper can serve as inspiration for future research in image matting, making a significant contribution to the image editing research community. However, since matting tools are widely used in video and image editing tasks, the proposed method could potentially be misused for creating fraudulent or misleading content. Therefore, it is crucial to regulate the application of this technology to prevent misuse and mitigate potential negative impacts.

References

Berman, A., Dadourian, A., and Vlahos, P. Method for removing from an image the background surrounding a selected object, 1998.

Cai, H., Xue, F., Xu, L., and Guo, L. TransMatting: Enhancing Transparent Objects Matting with Transformers. In *European Conference on Computer Vision (ECCV)*, 2022.

Chen, T., Cheng, M. M., Tan, P., Shamir, A., and Hu, S. M. Sketch2photo: internet image montage. In *ACM SIGGRAPH Conference and Exhibition on Computer Graphics and Interactive Techniques in Asia (SIGGRAPH Asia)*, 2009.

Chen, Y., Guan, J., and Cham, W. K. Robust multi-focus image fusion using edge model and multi-matting. *IEEE Transactions on Image Processing (TIP)*, 27(3):1526–1541, 2018.

Chuang, Y.-Y., Curless, B., Salesin, D., and Szeliski, R. A bayesian approach to digital matting. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2001.

Dai, Y., Lu, H., and Shen, C. Learning Affinity-Aware Upsampling for Deep Image Matting. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 6841–6850, 2021.

Dai, Y., Price, B., Zhang, H., and Shen, C. Boosting robustness of image matting with context assembling and strong data augmentation. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11707–11716, 2022.

Deng, J., Dong, W., Socher, R., Li, L.-j., Li, K., and Fei-fei, L. Imagenet: A large-scale hierarchical image database. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2009.

Forte, M. and Pitié, F. F, B, Alpha Matting. *ArXiv preprint arXiv:2003.07711*, 2020.

Gong, M., Qian, Y., and Cheng, L. Integrated foreground segmentation and boundary matting for live videos. *IEEE Transactions on Image Processing (TIP)*, 24(4):1356–1370, 2015.

Grady, L. and Westermann, R. Random walks for interactive alpha-matting. *IASTED International Conference on Visualization, Imaging and Image Processing (VIIP)*, 2005.

He, K., Rhemann, C., Rother, C., Tang, X., and Sun, J. A global sampling method for alpha matting. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2011.

He, K., Zhang, X., Ren, S., and Sun, J. Delving deep into rectifiers: Surpassing human-level performance on imagenet classification. In *International Conference on Computer Vision (ICCV)*, 2015.

He, K., Zhang, X., Ren, S., and Sun, J. Deep residual learning for image recognition. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

Hu, Y., Lin, Y., Wang, W., Zhao, Y., Wei, Y., and Shi, H. Diffusion for natural image matting. *ArXiv preprint arXiv:2312.05915*, 2023.

Levin, A., Lischinski, D., and Weiss, Y. A closed-form solution to natural image matting. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 30 (2):228–242, 2008.

Li, J., Zhang, J., and Tao, D. Deep Automatic Natural Image Matting. In *International Joint Conferences on Artificial Intelligence (IJCAI)*, 2021.

Li, Y. and Lu, H. Natural image matting via guided contextual attention. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2020.

Liu, L., Jiang, H., He, P., Chen, W., Liu, X., Gao, J., and Han, J. On the Variance of the Adaptive Learning Rate and Beyond. In *International Conference on Learning Representations (ICLR)*, April 2020.

Liu, Q., Xie, H., Zhang, S., Zhong, B., and Ji, R. Long-Range Feature Propagating for Natural Image Matting. In *ACM International Conference on Multimedia (ACM MM)*, 2021a.

Liu, Y., Xie, J., Shi, X., Qiao, Y., Huang, Y., Tang, Y., and Yang, X. Tripartite information mining and integration for image matting. In *International Conference on Computer Vision (ICCV)*, pp. 7555–7564, October 2021b.

Liu, Z., Lin, Y., Cao, Y., Hu, H., Wei, Y., Zhang, Z., Lin, S., and Guo, B. Swin Transformer: Hierarchical Vision Transformer using Shifted Windows. In *International Conference on Computer Vision (ICCV)*, 2021c.

Lu, H., Dai, Y., Shen, C., and Xu, S. Index networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI)*, 44(1):242–255, 2022.

Luo, W., Li, Y., Urtasun, R., and Zemel, R. S. Understanding the Effective Receptive Field in Deep Convolutional Neural Networks. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2016.

Park, G., Son, S., Yoo, J., Kim, S., and Kwak, N. Matteformer: Transformer-based image matting via prior-tokens. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 11696–11706, 2022.

Paszke, A., Gross, S., Massa, F., Lerer, A., Bradbury, J., Chanan, G., Killeen, T., Lin, Z., Gimelshein, N., Antiga, L., Desmaison, A., Kopf, A., Yang, E., Devito, Z., Raison, M., Tejani, A., Chilamkurthy, S., Steiner, B., Fang, L., Bai, J., and Chintala, S. Pytorch: An imperative style, high-performance deep learning library. In *Conference on Neural Information Processing Systems (NeurIPS)*, 2019.

Qiao, S., Wang, H., Liu, C., Shen, W., and Yuille, A. Weight standardization. *ArXiv preprint arXiv:1903.10520*, 2019.

Qiao, Y., Liu, Y., Yang, X., Zhou, D., and Wei, X. Attention-Guided Hierarchical Structure Aggregation for Image Matting. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2020.

Ruzon, M. and Tomasi, C. Alpha estimation in natural images. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2000.

Sandler, M., Howard, A., Zhu, M., Zhmoginov, A., and Chen, L.-c. Mobilenetv2: Inverted residuals and linear bottlenecks. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4510–4520, 2018.

Sun, J., Jia, J., keung Tang, C., and yeung Shum, H. Poisson matting. In *ACM Special Interest Group on Computer Graphics (SIGGRAPH)*, 2004.

Sun, Y., keung Tang, C., and wing Tai, Y. Semantic image matting. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, Ł., and Polosukhin, I. Attention Is All You Need. In *Conference on Neural Information Processing Systems (NeurIPS)*, volume 30, 2017.

Wang, T., Liu, S., Tian, Y., Li, K., and Yang, M.-h. Video Matting via Consistency-Regularized Graph Neural Networks. In *International Conference on Computer Vision (ICCV)*, 2021.

Wu, J., Zhang, C., Li, Z., Fu, H., Peng, X., and Zhou, J. T. dugMatting: Decomposed-uncertainty-guided matting. In Krause, A., Brunskill, E., Cho, K., Engelhardt, B., Sabato, S., and Scarlett, J. (eds.), *International Conference on Machine Learning (ICML)*, volume 202, pp. 37846–37859, 2023.

Wu, Y. and He, K. Group normalization. In *European Conference on Computer Vision (ECCV)*, 2018.

Xu, N., Price, B., Cohen, S., and Huang, T. Deep image matting. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

Xu, Y., He, S., Shao, W., Wong, K.-y. K., Qiao, Y., and Luo, P. Diffusionmat: Alpha matting as sequential refinement learning. *ArXiv preprint arXiv:2311.13535*, 2023.

Yao, J., Wang, X., Yang, S., and Wang, B. Vitmatte: Boosting image matting with pre-trained plain vision transformers. *Information Fusion*, 103:102091, 2024.

Yu, H., Xu, N., Huang, Z., Zhou, Y., and Shi, H. High-Resolution Deep Image Matting. In *AAAI Conference on Artificial Intelligence (AAAI)*, 2021.

Yu, Q., Zhang, J., Zhang, H., Wang, Y., Lin, Z., Xu, N., Bai, Y., and Yuille, A. Mask Guided Matting via Progressive Refinement Network. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2021a.

Yu, Z., Li, X., Huang, H., Zheng, W., and Chen, L. Cascade image matting with deformable graph refinement. In *International Conference on Computer Vision (ICCV)*, 2021b.

Zhao, H., Shi, J., Qi, X., Wang, X., and Jia, J. Pyramid scene parsing network. In *IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2017.

A. Supplementary Introduction to AEAL Block

The appearance-enhanced axis-wise learning (AEAL) block utilizes axis-wise attention to achieve a large receptive field. In the main text, we introduce the calculation process of axis-wise attention using Equations 3 and 4. For better understanding, we supplement Figure 9 with an illustration of axis-wise attention. As shown in the figure, axis-wise attention first slices the features into axis-wise rectangular regions and then applies multi-head self-attention processing. Finally, it merges the processed features. Since axis-wise attention operates on sliced features, it has a computational complexity of $\mathcal{O}(hw(h+w))$, which is lower than the $\mathcal{O}((hw)^2)$ complexity of self-attention.

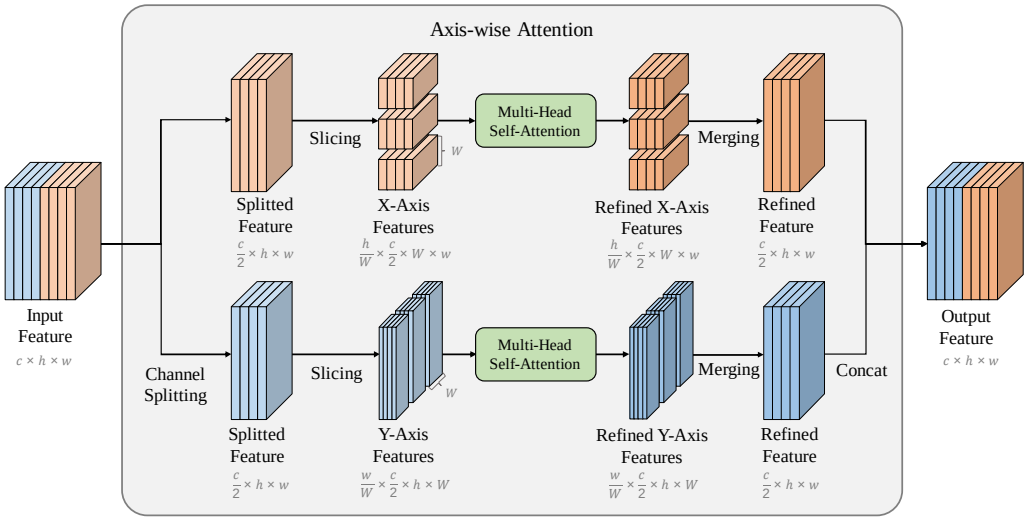


Figure 9. Illustration of Axis-wise Attention

B. Implementation details of AEMatter

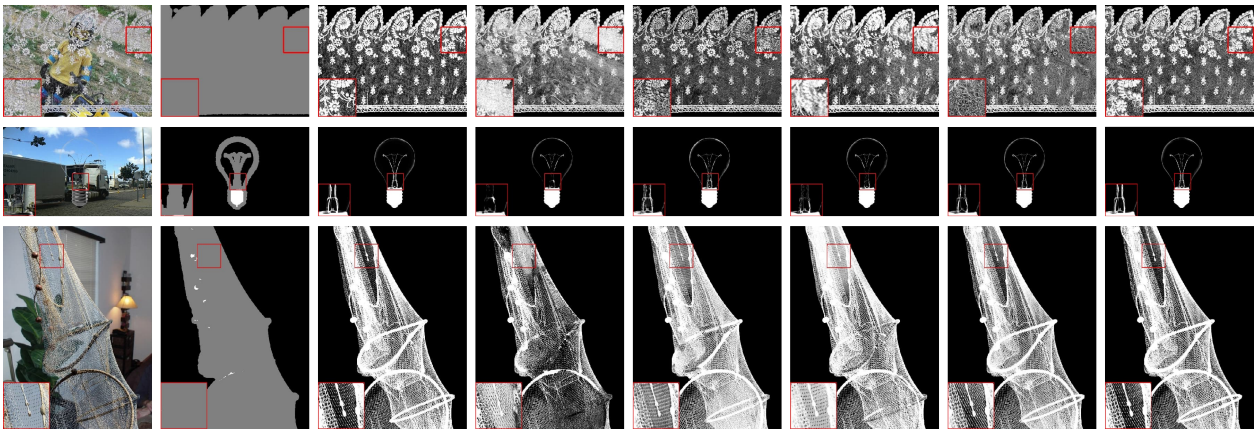
The proposed AEMatter is implemented using the PyTorch (Paszke et al., 2019) framework. Axis-wise attention with a width of $W = 5$ is used in the implementation. The coefficients in the loss functions are set as $\epsilon = 10^{-6}$, and $j = 4$. The network weights are initialized using the Kaiming initializer (He et al., 2015). To avoid overfitting, the backbone weights are initialized with the weights pre-trained on the ImageNet (Deng et al., 2009) dataset. The training is conducted on the Adobe Composition-1K dataset (Xu et al., 2017), using an NVIDIA RTX 3090 GPU with a batch size of 2 for 100 epochs. An RAdam optimizer (Liu et al., 2020) is employed to optimize the network weights with weight decay of 10^{-6} and betas of $(0.5, 0.999)$. The initial learning rate is set to 2.5×10^{-5} and decays to zero using a cosine annealing scheduler. Data augmentation techniques, including random affine transformation, random saturation transformation, random grayscale transformation, random gamma transformation, random contrast transformation, and random composition are applied to the training data. The trimap is generated from the alpha matte ground truth using erosion and dilation with kernel sizes ranging from 1 to 30 pixels. To facilitate network training, the image and trimap are randomly cropped into patches of size 1024×1024 and fed to the network. The code and model of AEMatter will be made available to the public later.

C. Additional Experimental Results

In this section, we present the qualitative results of AEMatter on the Distinctions-646, Transparent-460, Semantic Image Matting, and Automatic Image Matting-500 datasets. Additionally, we explore the potential of the insights of this paper in the automatic matting tasks with the Distinctions-646 and P3M datasets.

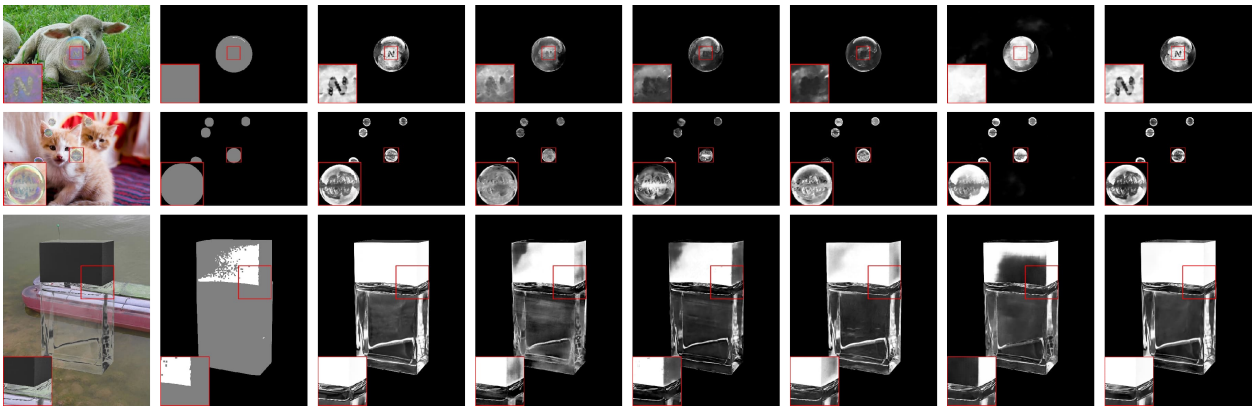
C.1. Generalization on Various Datasets

In the main text, we present the quantitative results of AEMatter across the Distinctions-646, Transparent-460, Semantic Image Matting, and Automatic Image Matting-500 datasets. Here, we supplement the quantitative results of AEMatter and existing methods as illustrated in Figures 10, 11, 12, and 13. The figures clearly demonstrate that AEMatter surpasses



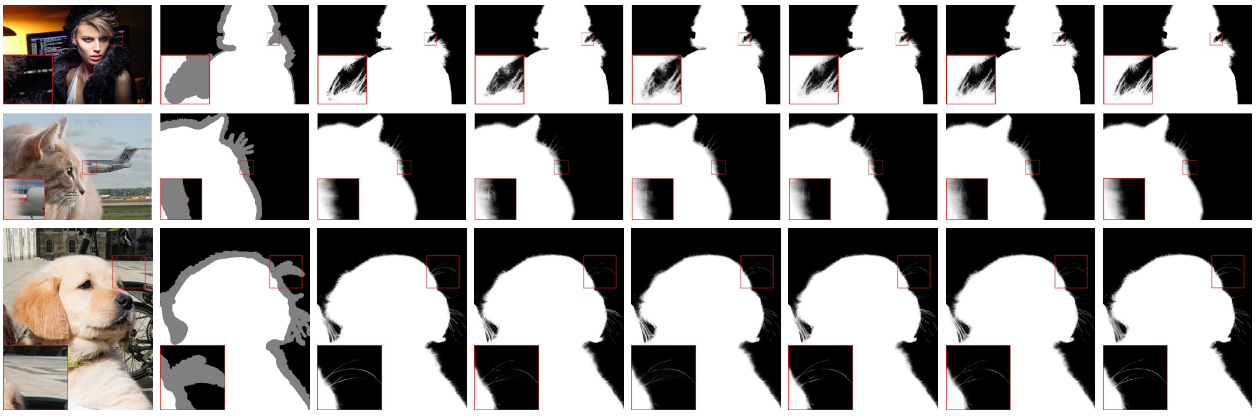
(a) Image (b) Trimap (c) Ground Truth (d) GCAMatting (e) FBAMatting (f) LFPNet (g) MatteFormer (h) AEMatter (Ours)

Figure 10. Qualitative comparison of the alpha matte results on the Distinction-646 dataset.



(a) Image (b) Trimap (c) Ground Truth (d) DIM (e) IndexNet (f) TIMI-Net (g) MGMatting (h) AEMatter (Ours)

Figure 11. Qualitative comparison of the alpha matte results on the Transparent-460 dataset.



(a) Image (b) Trimap (c) Ground Truth (d) GCAMatting (e) FBAMatting (f) LFPNet (g) MatteFormer (h) AEMatter (Ours)

Figure 12. Qualitative comparison of the alpha matte results on the Semantic Image Matting dataset.

existing methods, particularly in scenarios with similar foreground and background colors or when encountering foreground blur. These results underscore the strong generalization ability of AEMatter.

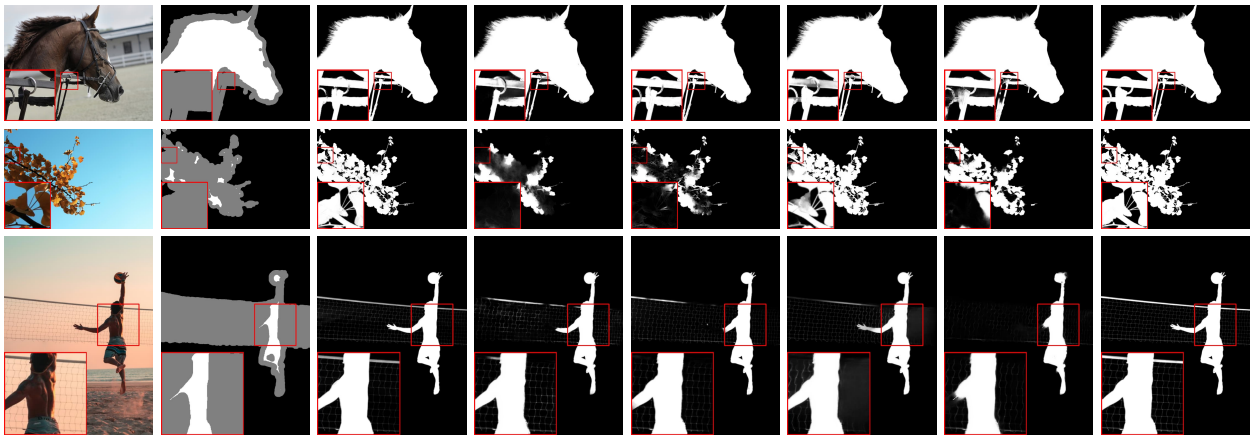


Figure 13. Qualitative comparison of the alpha matte results on the Automatic Image Matting-500 dataset.

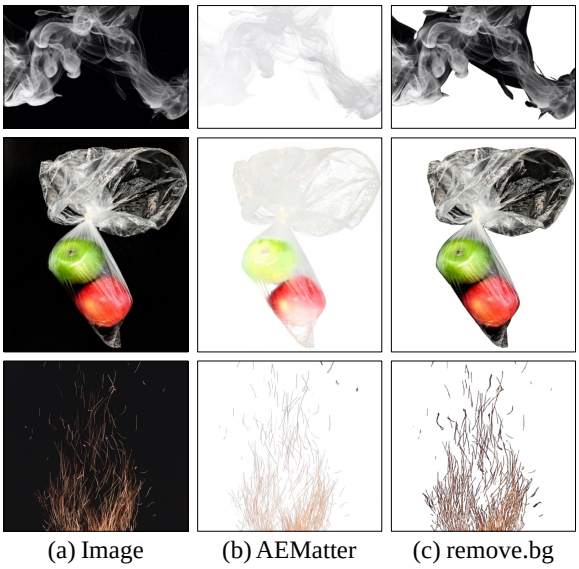


Figure 14. Results on images of semi-transparent objects.

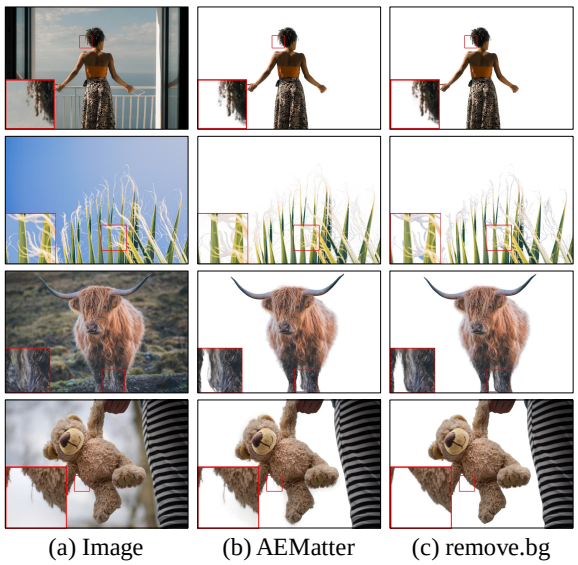


Figure 15. Results on images of fuzzy objects.

C.2. Comparison with remove.bg

To assess the practicality of AEMatter, we compare it with remove.bg, a commercial automatic matting tool trained on a private dataset. Since we do not have access to the remove.bg model for evaluation on public matting datasets, we used the paid service of remove.bg to extract the foreground and qualitatively compared it with the foreground extracted by AEMatter. It is important to note that AEMatter relies on manually annotated trimaps. The qualitative results are shown in Figures C.1 and C.1. Despite requiring a trimap, AEMatter extracts a finer foreground compared to remove.bg. remove.bg struggles to eliminate background interference when handling semi-transparent objects such as smoke and flames, and it fails to accurately extract detailed foreground of fuzzy objects. In contrast, AEMatter performs well in these scenarios, producing accurate and clean foreground.

D. Ablation Study

To evaluate the effectiveness of the network designs and training hyperparameters of AEMatter, we conduct ablation studies on the Adobe Composition-1K dataset.

Table 10. Ablation study on the decoder architecture and additional learning blocks. Decoder denotes the decoder adopted, AL denotes the additional learning block adopted, and AE denotes whether the appearance-enhanced block is used. The additional learning blocks considered are Vanilla, Window, and Axis, representing vanilla self-attention, window attention, and our axis-wise attention, respectively.

Encoder	Decoder	AL	AE	SAD	MSE	Grad	Conn
Resnet-50	Convolution	-	×	23.82	4.27	8.08	19.02
Swin-Tiny	Convolution	-	×	19.72	2.97	6.27	14.43
Hybrid-Transformer	Convolution	-	×	19.57	2.76	5.84	14.36
Hybrid-Transformer	Residual	-	×	19.23	2.82	5.73	14.09
Hybrid-Transformer	Transformer	-	×	18.91	2.66	5.13	13.87
Hybrid-Transformer	Transformer	Self-Attention	×	19.07	2.65	5.86	13.92
Hybrid-Transformer	Transformer	Window Attention	×	18.30	2.61	5.56	13.11
Hybrid-Transformer	Transformer	Axis-wise Attention	×	17.68	2.33	4.71	12.55
Hybrid-Transformer	Transformer	Axis-wise Attention	✓	17.53	2.26	4.76	12.46

Table 11. Ablation study on the sizes of training image patches.

Patch Size	SAD	MSE	Grad	Conn
256×256	24.40	4.06	8.09	19.95
512×512	21.03	3.26	6.46	15.93
768×768	19.43	2.79	5.57	14.23
1024×1024	17.53	2.26	4.76	12.46

D.1. Hybrid-Transformer Backbone

We introduce a Hybrid-Transformer backbone to enlarge the receptive field of AEMatter and helping extract rich low-level details. To assess the effectiveness of this design, we train and evaluate three AEMatter variants on the Adobe Composition-1K dataset, utilizing ResNet-50, Swin-Tiny, and our Hybrid-Transformer backbones, respectively. As depicted in Table 10, the Swin-Tiny backbone outperforms the ResNet-50 backbone due to its larger receptive field, benefitting from its larger receptive field. Moreover, the Hybrid-Transformer backbone of our AEMatter excels over the Swin-Tiny backbone in capturing low-level details, resulting in higher performance.

D.2. Transformer-based Decoder

We introduce a Transformer-based decoder to enlarge the receptive field of AEMatter for improving matting performance. To evaluate the effectiveness of this design, we train AEMatter variants with convolution, residual block, and transformer based decoders on the Adobe Composition-1K dataset. Then, we evaluate these variants and present the results in Table 10. Experimental results demonstrate that the transformer-based decoder of our AEMatter outperforms the other designs.

D.3. Appearance-Enhanced Axis-Wise Learning Block

We introduce an appearance-enhanced axis-wise learning (AEAL) block to further enlarge the receptive field of the encoder, which adopts appearance-enhanced (AE) blocks to enhance the appearance information of context features and axis-wise attention to learn large-scale context. To assess the effectiveness of these designs, we train AEMatter under different configurations: AEMatter without AE blocks, AEMatter with self-attention (Vaswani et al., 2017), and AEMatter with window attention (Liu et al., 2021c) on the Adobe Composition-1K dataset. Subsequently, we evaluate these variants and summarize the results in Table 10. Experimental findings reveal that AEMatter with axis-wise attention surpasses AEMatter with vanilla self-attention and window attention. Furthermore, the AE block leads to further performance improvement, underscoring the effectiveness of the proposed AEAL block.

D.4. Training Image Patch Size

We observe that training image matting networks with large image patches contributes to learning context aggregation, thereby resulting in improved matting performance. To validate the applicability of this observation to AEMatter, we evaluate AEMatter models trained with image patches of varying sizes. It is noteworthy that we train AEMatter on smaller image patches for an extended number of epochs to ensure network convergence. The results, summarized in Table 11,

Table 12. Comparison of the computational complexity and parameter amounts of matting methods.

Method	MACs (T)	Params (M)	SAD	MSE
LFPNet (Liu et al., 2021a)	6.16	112.2	23.60	4.10
MatteFormer (Park et al., 2022)	0.86	44.9	23.80	4.03
VitMatte-S (Yao et al., 2024)	1.69	25.8	21.46	3.30
VitMatte-B (Yao et al., 2024)	6.85	89.2	20.33	3.00
DiffMatte (Hu et al., 2023)	2.08	29.0	20.52	3.06
AEMatter (Ours)	1.15	48.7	17.53	2.26

demonstrate an improvement in network performance as the size of the training image patches increases. This confirms our observation that training with large image patches enhances the context aggregation capability of matting networks.

E. Model Complexity Analysis

In this section, we compare the computational complexity and parameter amounts of AEMatter and state-of-the-art matting methods. Specifically, the computational complexity is quantified by the count of multiply-accumulates (MACs) necessitated by each method for the inference of a 2048×2048 image. The parameter amounts denote the number of trainable parameters in each model. The results, summarized in Table 12, demonstrate that AEMatter shares a similar computational complexity and parameter amounts with existing matting methods. This suggests that the performance enhancement achieved by AEMatter is not attributed to an increase in model complexity.