This article has been accepted for publication in IEEE Transactions on Multimedia. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TMM.2022.3183403

Enabling Trimap-Free Image Matting with a Frequency-Guided Saliency-Aware Network via Joint Learning

Linhui Dai*, Xiang Song*, Xiaohong Liu, Chengqi Li, Zhihao Shi, Jun Chen, *Senior Member, IEEE,* Martin Brooks

***Abstract*—This paper presents a strategic approach to tackling trimap-free natural image matting. Specifically, to address the false detection issue of existing trimap-free matting algorithms when the foreground object is not uniquely defined, we design a novel tangled structure (TangleNet) to handle foreground detection and matting prediction simultaneously. TangleNet enables information exchange between foreground segmentation and alpha prediction, producing high-quality alpha mattes for the most salient foreground object based on RGB inputs alone. TangleNet boosts network performance with a frequency-guided attention mechanism utilizing wavelet data. Additionally, we pretrain for salient object detection to aid in the foreground segmentation. Experimental results demonstrate that TangleNet is on par with the state-of-the-art matting methods requiring additional inputs, and outperforms all previous trimap-free algorithms in terms of both qualitative and quantitative results.**

***Index Terms*—image matting, joint-task learning, frequency-guided attention.**

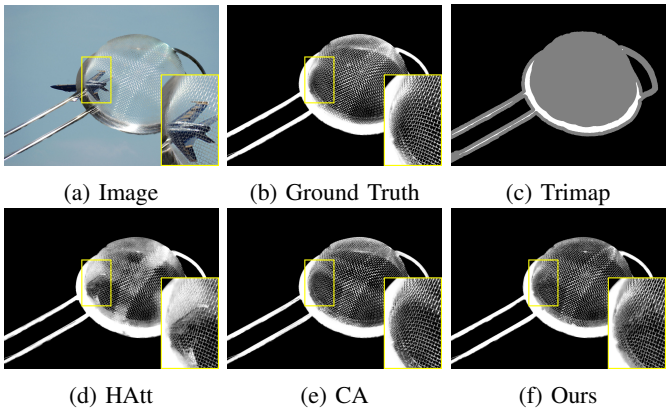
I. Introduction

IMAGE matting aims to predict an alpha matte that can be leveraged to accurately extract the target foreground object from an image with miscellaneous background objects. It has a broad range of applications in industrial tasks including image composition, video editing, and film production. To formulate the problem precisely, consider the following equation that relates composite image $\mathcal{I}$, foreground image $\mathcal{F}$, background image $\mathcal{B}$, and alpha value α at pixel i:

$$\mathcal{I}_i = \alpha_i \mathcal{F}_i + (1 - \alpha_i)\mathcal{B}_i, \quad \alpha_i \in [0, 1]. \tag{1}$$

Image matting seeks to determine α based on $\mathcal{I}$ perhaps together with additional user-provided information. Since $\mathcal{F}$ and $\mathcal{B}$ are unknown, the matting problem is in general mathematically ill-posed.

Early works on image matting making Eq. (1) solvable by requiring solid background color [3]. Later, hand-crafted algorithms identify alpha values by exploiting correlations among pixels in small regions. For example, propagation-based algorithms [4], [5], [6] make assumptions of local smoothness, and sampling-based methods [7], [8], [9], [10] utilize user-identified foreground and background patches. These methods suffer from performance degradation on images having complex background. Starting with [11], data-driven matting algorithms [2], [12], [13], [14], [15] leverage user-provided trimaps to generate high-quality alpha mattes. However, manual provision of trimaps is tedious and is infeasible for real-time applications such as video matting. Thus, there has been a growing interest in trimap-free approaches [1], [16], [17], [18], [19] that can predict alpha mattes based solely on RGB inputs. Most of the trimap-free matting methods [16], [17], [18] are designed with a specific kind of target object (e.g., human) in mind. Moreover, without user inputs, the SOTA trimap-free algorithms can easily fail to produce an accurate alpha matte for an image with a complicated foreground (see an example in Fig. 1).



(a) Image (b) Ground Truth (c) Trimap

(d) HAtt (e) CA (f) Ours

Fig. 1: Without the location clue, the state-of-the-art trimap-free algorithms (e.g., HAtt [1]) may make a false prediction by extracting a wrong object (the jet). Our TangleNet mitigates this effect and produces high-quality alpha mattes that are comparable to other recent trimap-based methods such as CA [2].

In this paper we propose a tangled structure named TangleNet to tackle the trimap-free matting problem. The rationale behind the present work is as follows: First, we argue that it is essential to endow the network with a certain prior knowledge of object saliency, otherwise the process of

L. Dai, X. Song, C. Li, Z. Shi and J. Chen are with the Department of Electrical and Computer Engineering, McMaster University, Hamilton, ON L8S 4K1 (e-mail: {dail5, songx5, lic222, shiz31, chenjun}@mcmaster.ca.)

X. Liu (corresponding author) is with the John Hopcroft Center, Shanghai Jiao Tong University, Shanghai, 200240, China (e-mail: xiaohongliu@sjtu.edu.cn).

M. Brook is with ShapeVision Inc. (email: brooks.martin@sympatico.ca).

This work was supported in part by the Natural Sciences and Engineering Research Council (NSERC) of Canada through a Discovery Grant. ⋆ Authors contributed equally.

foreground localization will be unreliable. A related research area in computer vision is called salient object detection. Surprisingly, utilizing object saliency in trimap-free matting is largely unexplored. In the present work, we pretrain the model on a salient object binary segmentation dataset before performing matting-oriented training. Second, we exploit correlation between salient object detection (binary segmentation) and matting (alpha prediction) by joint learning in two tangled sub-networks that reinforce each other, using a hybrid loss function. Lastly, we make judicious use of the frequency-domain feature as it is useful in revealing the fine image details and consequently is highly informative for image matting. In summary, the main contributions of this paper are:

- We exploit the connection between trimap-free matting and salient object detection by pretraining our network on a salient object detection dataset and leveraging this prior knowledge to fine-tune our model for trimap-free natural image matting.
- We introduce a novel network structure for joint learning of alpha prediction and binary segmentation, implementing the two tasks with dedicated sub-networks which are "tangled" by means of decoder and task switcher modules that optimize information flow between them, together with a mutli-scale hybrid loss function.
- We present an effective attention mechanism that leverages the frequency information to guide the behavior of the model. With the alpha prediction task directed to the high-frequency area, the binary segmentation task can focus more on the low-frequency counterpart.

II. Related Work

Natural image matting. Most natural image matting algorithms utilize user input to help solve the problem, typically given in the form of scribbles [20] or trimaps [21] that identify the foreground, background, and unknown regions.

Traditional matting algorithms can be roughly divided into two categories: propagation-based methods, and sampling-based methods. Both types rely on image regularity conditions. Propagation-based methods [4], [5], [6], [22] solve an optimization problem by assuming local smoothness of color distribution. Sampling-based methods [7], [8], [9], [10], [23] solve Eq. (1) by inferring each pixel's foreground or background membership based on user-provided image patches. The quality of alpha predictions generated by traditional methods varies significantly from case to case, depending on the applicability of those hypothetical regularity conditions. The data-driven approach in image matting has seen increasing popularity over the past few years. Cho et al. [11] combine some traditional matting algorithms and CNNs to form an end-to-end training pipeline. Xu et al. [14] propose an encoder-decoder structure to make alpha predictions based on RGB images and trimaps. This opens the floodgates to a host of learning-based matting algorithms with various innovations [2], [12], [13], [24], [25], [26], [27].

Trimap-free matting. It is desirable to predict alpha mattes using only RGB images because the provision of user input typically ranges from inconvenient to impractical. To tackle this challenging problem, some works [16], [17] leverage CNNs to segment images and guide alpha predictions. In contrast, Sengupta et al. [18] utilize different priors (soft segmentation and background) to replace trimaps. Zhang et al. [19] fuse the foreground and background predictions to obtain alpha mattes. Qiao et al. [1] utilize an attention mechanism to guide alpha predictions. However, most of the trimap-free methods focus on a specific type of foreground (e.g., human). Moreover, for natural image matting tasks, the lack of object location clues can cause false detection when extracting the foreground from a complex background. The present work resolves the issue by taking object saliency into account.

Attention mechanisms. Attention mechanisms in deep learning has enhanced performance of machine translation [28], [29] and computer vision [30], [31], [32], [33]. Wang et al. [32] introduce a non-local attention module that measures spatial information using a correlation matrix, using it to guide contextual information aggregation. Following this idea, a series of papers [34], [35], [36] leverage the non-local module to guide spatial or channel-wise learning. In image matting, Li et al. [12] introduces the idea of guided contextual attention that uses low-level features from different areas of the RGB image to guide propagation of the alpha prediction. In trimap-free matting we exploit frequency domain information to guide the feature propagation process. Inspired by [12] and [34], we design a frequency-guided attention module to facilitate our alpha prediction and binary segmentation processes.

Multi-task learning. Prediction accuracy for multiple tasks can be improved by multi-task learning when the tasks mutually reinforce each other, e.g. depth estimation and semantic segmentation [37], [38], [39]) In image matting, Cai et al. [24] leverage multi-task learning to refine trimap input and predict alpha matte simultaneously. However, we argue that due to loosely connected model structures, the alpha prediction branch does not make full use of the task correlation, and therefore their method is not suitable for trimap-free matting. Inspired by [39], we propose a tangled structure that facilitates joint learning of alpha prediction and binary segmentation by structuring information flow between the two tasks.

Salient object detection. This task aims to recognize the most attention-grabbing foreground in an image [40], [41], [42], [43], [44]. Our work exploits the knowledge from this task by performing pretraining on a salient object detection dataset and introducing a customized loss to couple the learning processes for salient object detection and trimap-free image matting.

III. Our Method

A. Method overview

Trimap-free natural image matting is a challenging problem as one has to locate the foreground and extract the fine details from the target object concurrently. Most existing trimap-free methods start by generating a trimap, thereby providing a location estimate for the foreground.

In contrast, the present work uses interactive learning, with switcher modules to direct network flow during the joint learning of alpha prediction and binary segmentation. A

This article has been accepted for publication in IEEE Transactions on Multimedia. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TMM.2022.3183403

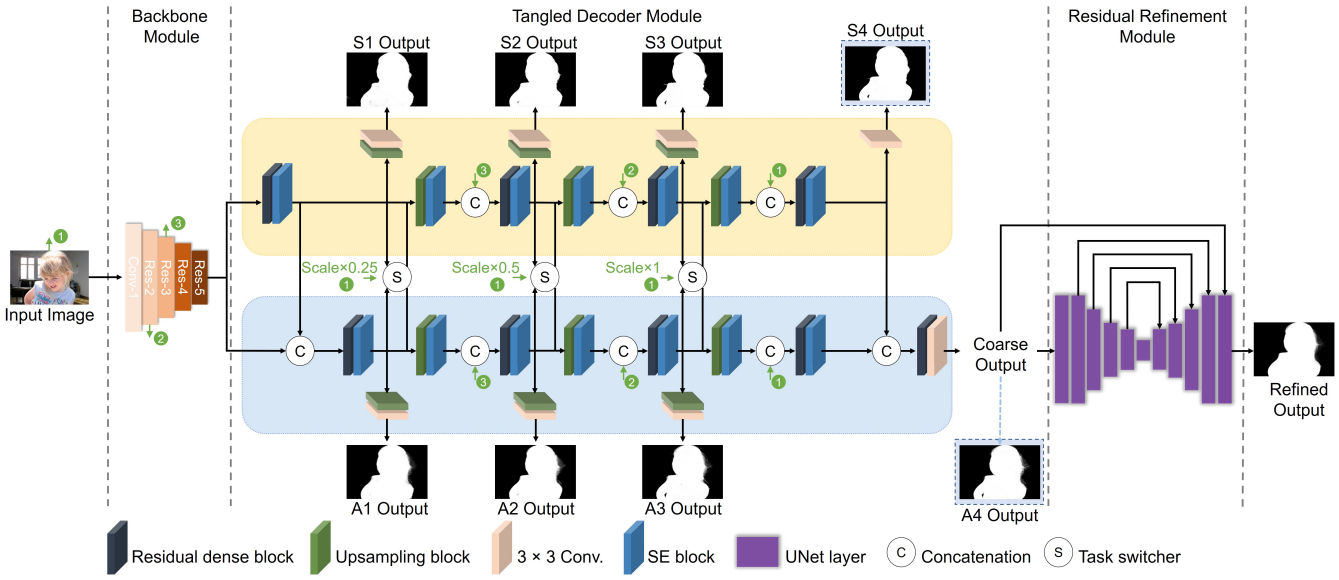


Fig. 2: Architecture of the proposed TangleNet. The orange and blue boxes indicate the binary segmentation branch and the alpha prediction branch, respectively. The two modules are tangled together to form a joint learning pipeline, where information is exchanged via the task switchers. A residual refinement module completes the process.

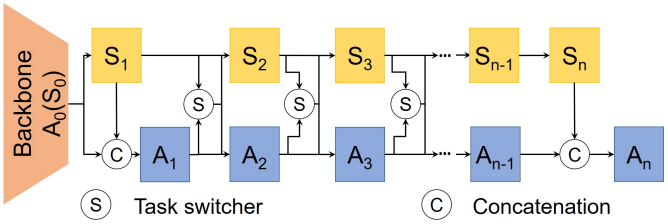


Fig. 3: Flow chart of interactive joint learning. We denote the alpha prediction task at stage i as A_i and the binary segmentation task as S_i. For all cases except $i = 0, 1,$ and n, we use task switchers to distribute information from previous stages. Note that $i = 0$ denotes the backbone stage, which feeds features to $A_1(S_1)$.

graphical illustration is shown in Fig. 3. Formally, we denote the alpha prediction and binary segmentation tasks as A and S, respectively.

Using γ to represent the task switcher function (to be discussed later), our interactive learning process can be expressed as:

$$\mathcal{F}_i^S = \begin{cases} \phi_i^S(\mathcal{F}_{i-1}^S,\ \mathcal{W}_i^S), & \text{if } i = 1,\ n, \\ \phi_i^S(\gamma(\mathcal{F}_{i-1}^A,\ \mathcal{F}_{i-1}^S) + \mathcal{F}_{i-1}^S,\ \mathcal{W}_i^S), & \text{otherwise,} \end{cases}$$
$$\mathcal{F}_i^A = \begin{cases} \phi_i^A(\mathcal{F}_{i-1}^A \copyright \mathcal{F}_i^S,\ \mathcal{W}_i^A), & \text{if } i = 1,\ n, \\ \phi_i^A(\gamma(\mathcal{F}_{i-1}^A,\ \mathcal{F}_{i-1}^S) + \mathcal{F}_{i-1}^A,\ \mathcal{W}_i^A), & \text{otherwise.} \end{cases} \quad (2)$$

Note that © denotes the feature map concatenation; ϕ_i^A (ϕ_i^S) is a prediction function with learnable parameter $\mathcal{W}_i^A$ ($\mathcal{W}_i^S$) for stage i; $\mathcal{F}_{i-1}^A$ ($\mathcal{F}_{i-1}^S$) is the output from the previous stage of the alpha prediction (binary segmentation) task; when $i = 0$, $\mathcal{F}_i^A$ and $\mathcal{F}_i^S$ are the same, which is the output from the last layer of backbone. To extract fine details for alpha prediction, a frequency-guided attention (FGA) module is used to direct network attention to high-frequency regions. We obtain frequency domain information through the discrete wavelet transform (DWT), and realize a frequency-guided self-attention mechanism following the idea of [12] and [32]. In addition, we pretrain our model on a salient object detection dataset and tailor our multi-task loss function to accommodate the binary segmentation task.

B. Network architecture

We introduce a tangled structure that jointly learns binary object segmentation and alpha matte prediction by sharing information between the two tasks. As shown in Fig. 2, our network takes the form of an encoder-decoder structure. For the encoder, we adopt a modified version of ResNet-50 that integrates SE blocks into all stages of residual blocks as suggested in [31]. The reason for choosing this design option is to obtain useful channel features while extracting semantic information. The core of our TangleNet is the decoder, which can be depicted as three parts, namely, the tangled decoder, task switcher (TS), and the frequency-guided attention module (FGA). We enhance the result with a UNet residual refinement module.

C. Tangled decoder

The tangled decoder aims to decode the feature maps and learn binary segmentation and alpha prediction simultaneously. As shown in Fig. 2, the blue and orange boxes indicate the corresponding branches for the two tasks, which are supervised by a customized multi-scale loss function using two kinds of ground truth. Inspired by the residual structure from [45] and [46], we use residual dense blocks (RDB) followed by squeeze

This article has been accepted for publication in IEEE Transactions on Multimedia. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TMM.2022.3183403

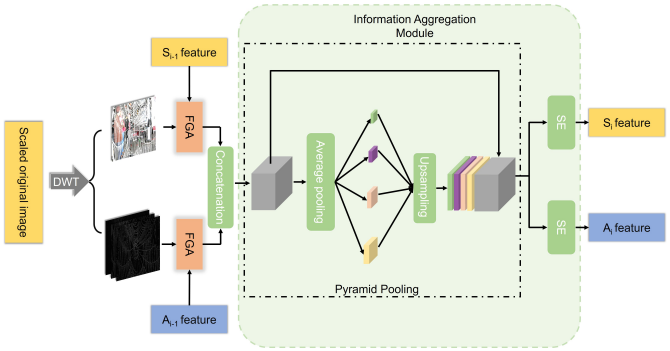


Fig. 4: The overview of our task switcher module.

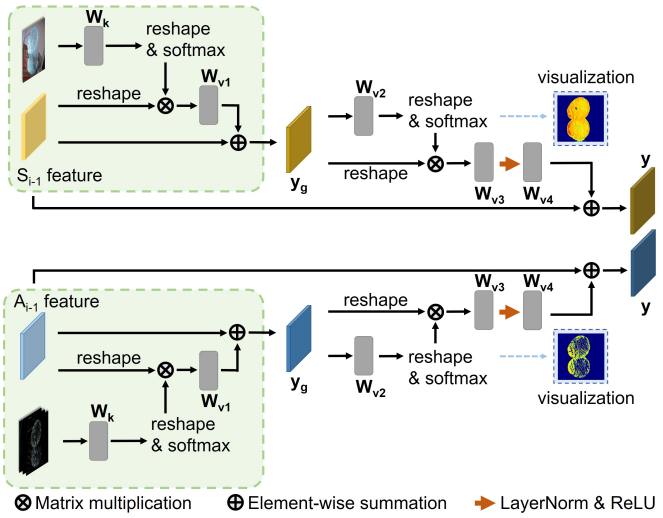


Fig. 5: Overview of FGA modules for each tasks. Green boxes indicate the frequency-guided operation.

and excitation (SE) blocks [31] as our main processing units (ϕ_i^A and ϕ_i^S) for each stage in Eq. (2). Here, RDB serves as a strong feature extractor that decodes the features for TS modules and the rest of network units.

D. Task switcher (TS)

In image matting, alpha matte can be thought of as an extension of binary segmentation because it contains alpha values and solid foreground information. We use TS modules to exploit the correlation between binary segmentation and alpha prediction, thereby improving alpha matte. Specifically, we employ the TS modules during interactive learning ($i \in [2, n-1]$ in Eq. (2)) to weigh and distribute the decoded feature maps. As shown in Fig. 4, our TS module consists of two frequency-guided attention modules (FGA) and an information aggregation module. First, we utilize frequency-guided attention to direct spatial-level and channel-level attention, using correlation between decoded features and original image frequency domain features to measure the spatial and channel information. Next, we use a pyramid pooling operation to aggregate weighted features, and we leverage channel-wise attention to perform task-specific modulation of feature maps. Using γ to indicate the task switcher function, the TS module can be formulated as:

$$\gamma(\mathcal{F}_{i-1}^A,\ \mathcal{F}_{i-1}^S) = \begin{cases} f_{SE}(f_{PSP}(\psi(\mathcal{F}_{i-1}^S,\ \mathcal{I}_L))), & \text{for task S,} \\ f_{SE}(f_{PSP}(\psi(\mathcal{F}_{i-1}^A,\ \mathcal{I}_H))), & \text{for task A,} \end{cases} \tag{3}$$

where ψ denotes the FGA, which takes $(\mathcal{F}_{i-1}^S, \mathcal{I}_L)$ in binary segmentation task S, and $(\mathcal{F}_{i-1}^S, \mathcal{I}_H)$ in alpha prediction task A. In addition, $\mathcal{I}_L$ and $\mathcal{I}_H$ indicate the low and high-frequency subbands, respectively; f_{PSP} is the pyramid pooling operation for aggregating information; f_{SE} denotes the squeeze and excitation (SE) [31] operation that weighs channel information according to the target task. Thus, our TS module employs attention mechanisms on the spatial level, channel level, and task level.

E. Frequency-guided attention

As part of the TS module, the frequency-guided attention module (FGA) directs spatial and channel attention. In the trimap-free matting problem, the main issue is that lack of foreground location information allows for false detection of foreground objects and consequently the extraction of wrong image details. Also, it is a heavy load for the network to solve both object segmentation and alpha prediction. This motivates methods that locate the foreground and relieve the learning load. By inspecting the image matting dataset, we notice that most of the fine details are located in high-frequency areas, which makes the frequency domain information a useful hint for inferring the alpha value. For the frequency-domain analysis, we adopt DWT in view of its ability to preserve information inside the CNN structures, as proved in the literature [47], [48], [49]. To be specific, we take the 2D Haar wavelet for DWT with the four filters defined as $f_{LL}=\begin{pmatrix}1&1\\1&1\end{pmatrix}$, $f_{LH}=\begin{pmatrix}-1&-1\\1&1\end{pmatrix}$, $f_{HL}=\begin{pmatrix}-1&1\\-1&1\end{pmatrix}$, $f_{HH}=\begin{pmatrix}1&-1\\-1&1\end{pmatrix}$. The resulting four subbands can be viewed as a low-frequency component, a high-frequency component along the x-axis, a high-frequency component along the y-axis, and a high-frequency component along the diagonal. As shown in Fig. 5, we use the low-frequency component in binary segmentation, and the high-frequency components in alpha prediction. We implement FGA using the idea of non-local operation [32], which can be modelled as $y_i = \sum_{j=1}^{Np} \frac{f(x_i,x_j)}{C(X)} g(x_j)$. Note that y_i and x_i are the respective values of input and output features at query position i; j is the index that enumerates all possible positions; Np denotes the total number of positions ($H \times W$ for image); $g(\cdot)$ denotes the linear transformation function, e.g., 1×1 convolution; $f(x_i, x_j)$ denotes the function that measures the similarity between x_i and x_j; $C(X)$ is the normalization function. In FGA, we first reshape the RGB image to make the size of frequency subbands the same as the input feature maps. Then we use Embedded Gaussian [32] to compute the similarity between position i on feature maps and position j on frequency subbands. To reduce computational complexity, we adopt the simplified design from [34] to implement FGA. The frequency-guided feature y_i^g produced by frequency-guided operation (green boxes in Fig. 5) at location i can be expressed

as

$$y_i^g = x_i^{feat} + W_{v1} \sum_{j=1}^{Np} \frac{e^{W_k x_j^{freq}}}{\sum_{m=1}^{Np} e^{W_k x_m^{freq}}} x_j^{feat}, \quad (4)$$

where W_{v1} and W_k are 1×1 convolutions; x_j^{freq} and x_j^{feat} denote respective values of location j on the frequency subband and input feature map; x_i^{feat} is the value of query location i on the input feature map. Note that frequency guidance takes effect by influencing the fraction term in Eq. (4). For example, at high-frequency subbands, the low-frequency part will have a small value of x_j^{freq}, making the fraction term very small as well, and vice versa for the low-frequency area. After the frequency-guided operation, we invoke another simplified non-local operation with y^g=$\{y_i^g\}_{j=1}^{Np}$ as the input to model the pixel-wise relationship within the guided feature map. In addition, we add a bottleneck term to model the channel attention, as in [34]. By denoting the output of FGA at location i as y_i, the entire FGA module can be defined as:

$$y_i = x_i^{feat} + W_{v4} f_{Re}\left(f_{LN}\left(W_{v3} \sum_{j=1}^{Np} \frac{e^{W_{v2} y_j^g}}{\sum_{m=1}^{Np} e^{W_{v2} y_m^g}} y_j^g\right)\right), \quad (5)$$

where W_{v2-v4} are convolutional operations with kernel size as 1 while $W_{v4} f_{Re}(f_{LN}(W_{v3}(\cdot)))$, f_{Re}, and f_{LN} denote the bottleneck operation, ReLU function, and LayerNorm, respectively. More details of the attention visualization can be found in Fig. 9 and Section V-D.

F. Loss function and pretraining strategy

Since trimap-free matting inevitably faces the problem of locating the foreground object, we design a hybrid loss for binary segmentation and alpha prediction.

For binary segmentation, we adopt the loss combination of [42] due to its promising performance on salient object detection. Segmentation loss ℓ^{seg} is defined as

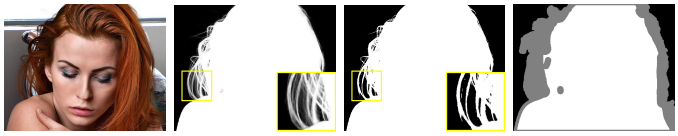
$$\ell^{seg} = \ell^{bce} + \ell^{ssim} + \ell^{iou}, \quad (6)$$

where ℓ^{bce}, ℓ^{ssim}, and ℓ^{iou} are respectively BCE loss [50], SSIM loss [51], and IoU loss [52]. The IoU loss measures the set similarity, and is given by $\ell^{iou} = 1 - \frac{\sum_i p_i p_i^*}{\sum_i [p_i + p_i^* - p_i p_i^*]}$, where p_i and p_i^* are predicted probability and ground truth labels at location i, respectively. For the image matting part, we use $\ell^{\alpha} = \sum_i |\tilde{\alpha_i} - \alpha_i|$ and $\ell^{comp} = \sum_i |\tilde{\mathcal{I}_i} - \mathcal{I}_i|$ from [14], and compose them with gradient loss $\ell^{grad} = \sum_i |\tilde{g_i} - g_i|$ and aforementioned SSIM loss ℓ^{ssim} to form ℓ_m:

$$\ell^m = \ell^{\alpha} + \ell^{comp} + \ell^{grad} + \ell^{ssim}, \quad (7)$$

where i specifies pixel location, $\tilde{\alpha_i}$ indicates predicted alpha value, $\tilde{\mathcal{I}_i}$ is the composite RGB image value defined as $\tilde{\mathcal{I}_i} = \tilde{\alpha_i}\mathcal{F}_i + (1-\tilde{\alpha_i})\mathcal{B}_i$, and $\tilde{g_i}$ (g_i) denotes the gradient of the predicted (ground-truth) alpha matte. TangleNet uses multi-scale loss during training; we denote the composed losses at scale j as ℓ_j^{seg} and ℓ_j^m. Consequently, the overall loss during alpha prediction is

$$\mathcal{L}^{total} = \sum_{j=1}^{k_1} \ell_j^{seg} + \sum_{j=1}^{k_2} \ell_j^m. \quad (8)$$



(a) Image (b) Alpha Matte (c) Binary Map (d) Trimap

Fig. 6: Visualization of our customized binary segmentation map and other types of ground truth. It is evident that our binary segmentation map (c) preserves more semantic information than the foreground area of the trimap (d) does.

Here $k_1 = 4$ as there are 4 different scales for the output segmentation map while $k_2 = 5$ as there is an additional refined alpha matte from the residual refinement module.

To further address the problem of how to locate the foreground object, we pretraining on the DUTS-TR dataset [53] to let the network acquire some prior knowledge before proceeding to training for alpha matting. This pretraining enables the network to identify the salient objects and further facilitates the alpha prediction task in the later stage. More implementation details can be found in the subsequent section.

IV. Implementation Details

A. Experiment setup

TangleNet is trained using the PyTorch framework. We first conduct a pretraining on the DUTS-TR dataset with images resized to 256×256 and then randomly cropped to 224×224. We initialize the backbone for pretraining using ImageNet weights. For image matting training, we randomly choose whether to crop an image or directly resize it to 320×320 (as the resized image can retain more semantic information). For the first choice, the input image is randomly cropped to 320×320, 480×480, 640×640, and 720×720 along the unknown region in the trimap (smaller patches reveal more fine details). Then we resize all image patches to 320×320. Both training stages use vertical, horizontal, and diagonal flipping as an additional augmentation strategy. For the ground truth of the segmentation branch, we use a customized binary map (Fig. 6c), in lieu of the normal trimap, with each pixel value marked to be 255 if the corresponding alpha value is greater than 0.5. The reason is that the morphological methods used for trimap generation can distort the original object shape whereas the customized binary map can better retain the shape of the foreground . For both stages, we use batch size of 4 and "poly" learning rate decay, where $lr = lr_{init}(1 - \frac{iter}{max\_iter})^p$ with $lr_{init} = 1e^{-4}$ and $p = 0.9$. For the pretraining stage, we train $50k$ iterations and take only the segmentation outputs (bounded by the orange box in Fig. 2) for calculating the loss, and deactivate the residual refinement module. For the image matting training stage, we train for 10^6 iterations and make use of all outputs.

B. Datasets and evaluation metrics

For the pretraining stage, we use the DUTS-TR dataset, which is derived from ImageNet and contains $10,553$ images with salient objects. Our main focus is on the matting training

This article has been accepted for publication in IEEE Transactions on Multimedia. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TMM.2022.3183403

stage, where we adopt the Adobe Composition-1k dataset from [14] for training and testing. The training set consists of 431 foregrounds, each of which is composited randomly with 100 unique background images from the MS COCO dataset [54]. For testing, the dataset provides 50 foregrounds,

TABLE I: Numerical results on the Composition-1k testing set calculated on the unknown region indicated by the trimap.

Methods	MSE↓	SAD↓	Grad↓	Conn↓
KNN [7]	0.103	175.45	124.13	176.39
ClosedForm [6]	0.091	168.13	126.88	167.92
BGM [18]	0.022	56.81	74.79	56.21
Late Fusion† [19]	0.020	49.02	34.33	50.60
DIM [14]	0.014	50.40	31.00	50.80
IndexNet [13]	0.013	45.48	25.9	433.7
ATNet [15]	0.013	40.50	21.50	39.40
GCA [12]	0.009	35.28	16.92	32.53
CA [2]	0.008	35.80	17.30	33.20
PIIA [27]	0.009	36.40	16.90	31.50
A^2U [55]	0.008	32.15	16.39	29.25
TIMI [56]	0.006	29.08	11.50	25.36
SD†	0.027	75.51	47.28	73.47
TD†	0.021	51.71	40.32	50.94
TD+TS†	0.018	49.24	38.93	47.33
TD+TS+FGA†	0.015	48.06	29.46	46.12
TD+TS+FGA+msloss†	0.013	45.23	25.31	45.23
TD+TS+FGA+pretrain†	0.013	45.52	26.11	46.06
TD+TS+FGA+pretrain +msloss†	0.011	43.03	22.52	44.69
TD+TS+FGA+msloss+Ref w/o pretrain†	0.011	43.36	22.12	41.17
TD+TS+FGA$_{rgb}$ +msloss+Ref+pretrain†	0.011	42.01	22.82	39.55
Ours†	0.010	40.16	18.87	37.31

each of which is composited with 20 background images from PASCAL VOC [57] that are randomly picked without replacement. Additionally, we train the model on the Dist-646 dataset [1] for comparison with other state-of-the-art natural image matting methods. Note that the training set has 596 foreground images and the testing set has 50 foregrounds; both of them are composited using the same composition rules as Composition-1k.

We use four quantitative metrics to evaluate our model: sum of absolute differences (SAD), mean squared error (MSE), gradient (Grad), and connectivity (Conn) [58]. To make a fair comparison and avoid bias, the reported performance results are based on official model weights, images, and numbers provided by the relevant papers.

V. Experiments

A. Composition-1k dataset

We compare TangleNet with ten trimap-based matting algorithms: KNN [7]; ClosedForm [6]; DIM [14]; IndexNet [13]; GCA [12]; CA [2]; PIIA [27]; ATNet [15]; A^2U [55]; and TIMI [56], and to three trimap-free methods: Late Fusion [19]; BGM [18] and HAtt [1]. More precisely, Late Fusion, HAtt, and our method only need RGB inputs; BGM requires the background image as an additional input; and others need both RGB and trimap images, produced according to the procedure described in [14]. Table I and Table II provide quantitative comparisons on the Composition-1k dataset. Here, "†" indicates that the corresponding method takes only RGB

TABLE II: Numerical results on the Composition-1k testing sets calculated on the whole image.

Names	MSE↓	SAD↓	Grad↓	Conn↓
KNN [7]	0.026	126.20	117.17	131.05
ClosedForm [6]	0.023	105.73	91.76	114.55
BGM [18]	0.008	57.21	74.83	56.57
DIM [14]	0.009	47.56	43.29	55.90
IndexNet [13]	0.005	44.52	29.88	42.37
GCA [12]	0.003	32.91	15.33	29.58
CA [2]	0.003	32.62	13.89	28.91
Late Fusion† [19]	0.011	58.34	41.63	59.74
HAtt†* [1]	0.009	48.98	41.57	49.93
SD†	0.019	75.63	46.23	80.54
TD†	0.016	65.31	44.43	72.32
TD+TS†	0.014	55.85	33.52	61.13
TD+TS+FGA†	0.012	50.66	29.63	55.04
TD+TS+FGA+msloss†	0.010	48.96	28.51	51.10
TD+TS+FGA+pretrain†	0.010	48.33	28.67	50.62
TD+TS+FGA+pretrain +msloss†	0.008	46.07	27.71	46.22
TD+TS+FGA+msloss+Ref w/o pretrain†	0.007	44.96	26.22	44.96
TD+TS+FGA$_{rgb}$ +msloss+Ref+pretrain†	0.007	43.13	25.06	44.20
Ours†	0.006	41.31	23.44	43.56

images as inputs. Red denotes the best result for methods that use additional inputs while Blue indicates the best result for methods that use only the RGB images. Several variants of our method are shown in grey rows (see Section V-C for their respective definitions). From these table, we can see that the proposed TangleNet outperforms all trimap-free methods by a large margin. Table I focuses on performance of fine detail extraction, as we only measure errors in the unknown region indicated by the trimap. In this respect, our method is comparable to most state-of-the-art trimap-based algorithms and is only inferior to GCA and CA. This is expected because these trimap-based algorithms use strong user input to guide the prediction. In contrast, the present method does not have user input, utilizing only intrinsic image features for alpha prediction, yet it performs competitively against the trimap-based methods. The qualitative results in Fig. 7 further prove effectiveness of our method. For other trimap-free methods, performance is limited by the need to learn both semantic and alpha features. Benefiting from joint training and frequency guidance, the proposed TangleNet is more effective in predicting alpha values while retaining good semantic features. In fact, our results are very close to ground truth and visually comparable to those of the state-of-the-art trimap-based methods.

B. Dist-646 dataset

We also compare our results with the benchmark provided by [1]. The performance results of several variants of our method on the Dist-646 dataset can be found in Table III. Here, the Boldface indicates the best result and the grey rows are variants of our method described in Section V-C. These results follow the same trend from the Composition-1k dataset in the sense that our method outperforms all other trimap-free algorithms and makes comparable prediction against the state-of-the-art trimap-based algorithms. It is noticeable that

This article has been accepted for publication in IEEE Transactions on Multimedia. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TMM.2022.3183403

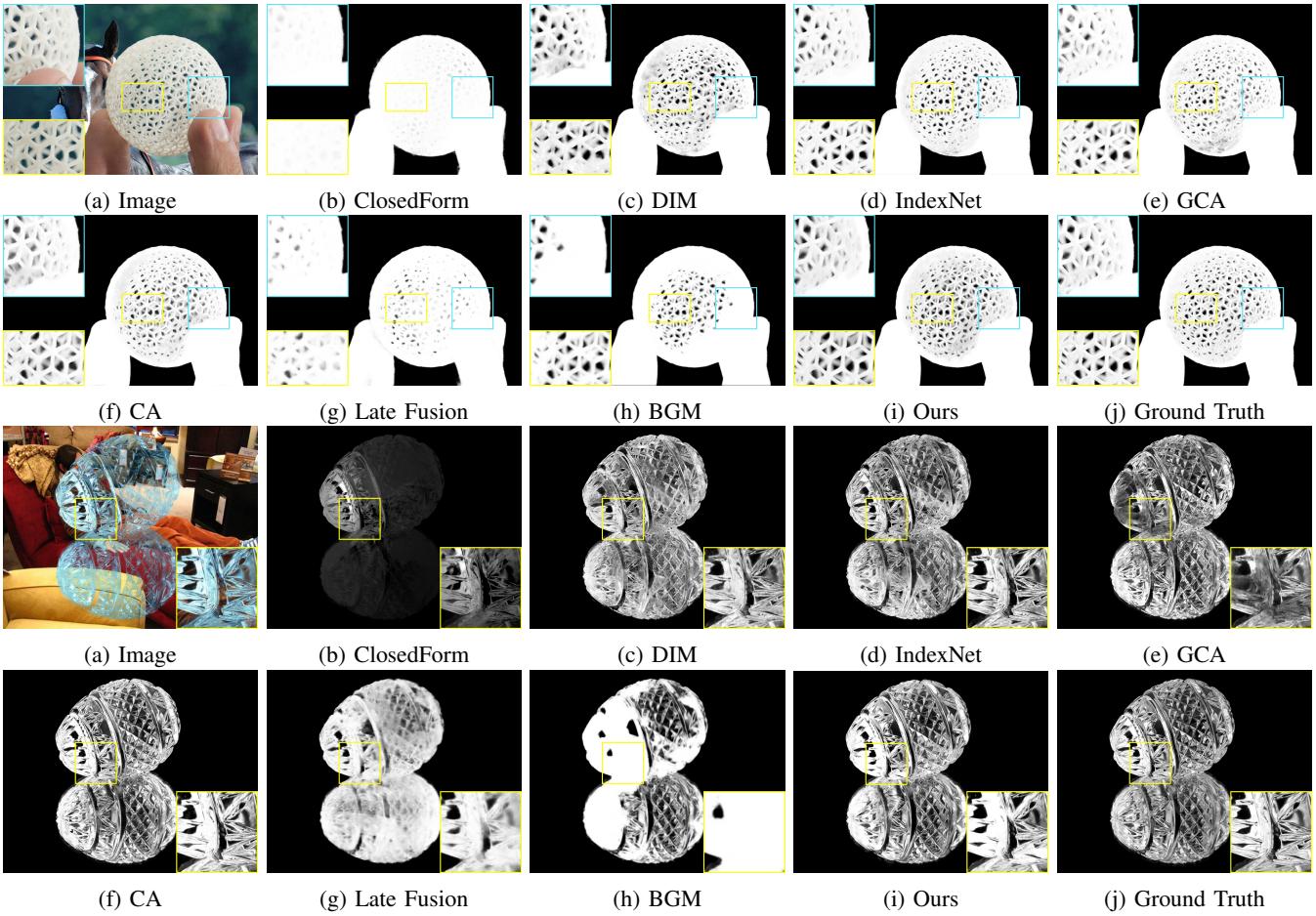


Fig. 7: The visual comparison results on the Adobe Composition-1k dataset. See supplementary material for more results.

TABLE III: Numerical results the Dist-646 testing sets calculated on the whole image. * means that the relevant numbers are quoted from [1].

Names	MSE↓	SAD↓	Grad↓	Conn↓
KNN* [7]	0.025	116.68	103.15	121.45
ClosedForm* [6]	0.023	105.73	91.76	114.55
DIM* [14]	0.009	47.56	43.29	55.90
HAtt*† [1]	0.009	48.98	41.57	49.93
SD†	0.021	84.33	61.04	89.88
TD†	0.017	66.64	52.41	78.53
TD+TS†	0.015	59.24	48.32	66.12
TD+TS+FGA†	0.014	52.06	41.63	58.13
TD+TS+FGA+msloss†	0.012	48.06	38.32	54.20
TD+TS+FGA+pretrain†	0.011	48.52	39.72	51.20
TD+TS+FGA+pretrain +msloss†	0.009	44.23	37.01	47.61
TD+TS+FGA+msloss+Ref w/o pretrain†	0.008	42.54	36.99	45.74
TD+TS+FGA$_{rgb}$ +msloss+Ref+pretrain†	0.007	41.98	37.54	43.17
Ours†	**0.007**	**40.88**	**36.22**	**40.23**

our method outperforms HAtt, which sets a new state-of-the-art result for trimap-free matting on this dataset. See supplementary material for qualitative results.

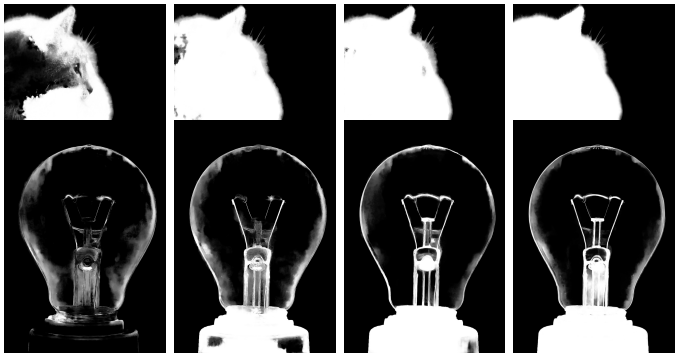


Fig. 8: Qualitative results for different designs of our method.

C. Ablation study

To justify our design, we examine nine alternatives: single decoder (**SD**); tangled decoder (**TD**); tangled decoder with task switcher module (**TD+TS**); tangled decoder with task switcher module and FGA (**TD+TS+FGA**); tangled decoder with task switcher module and FGA with multi-scale loss (**TD+TS+FGA+msloss**); tangled decoder with task switcher module and FGA with pretraining

(**TD+TS+FGA+pretrain**); our design without refinement module (**TD+TS+FGA+pretrain+msloss w/o Ref**); our design without pretraining (**TD+TS+FGA+msloss+Ref w/o pretrain**); and FGA with RGB input (**TD+TS+FGA$_{rgb}$+msloss+Ref+pretrain**). In the single decoder case, we remove the segmentation branch and only train the alpha prediction branch; in the tangled decoder case, the binary segmentation branch is added back, but the alpha prediction branch takes segmentation information directly without the assistance from the TS module. Without refinement module means that the refinement module is removed from Fig. 4; in the w/o pretrain case, the alpha prediction task is trained from scratch. FGA$_{rgb}$ means FGA with wavelet components in its input replaced by the original RGB image.

Some quantitative and qualitative results are shown in Table I, Table II, Table III, and Fig. 8. We make the following observations: 1) Removing the TS module jeopardizes the performance of the tangled decoder as the information from different tasks cannot be utilized efficiently. 2) The TS module can be viewed as an information regulator that selects useful information to facilitate training. In fact, by adding the TS module, we observe 13% and 12% performance gains in MSE for the whole image error on Composition-1k and Dist-646 datasets, respectively. 3) The FGA module contributes significantly to the model performance under the Gradient metric as evidenced by 12% improvement for the gradient error on the Composition-1k dataset and 14% for the gradient error on the Dist-646 dataset (see Table II and Table III.). 4) By comparing **TD+TS+FGA**, **TD+TS+FGA+msloss**, and **TD+TS+FGA+msloss+Ref w/o pretrain**, we see that adding the refinement module and incorporating multi-scale loss yield about 30% and 20% MSE performance gain, respectively. 5) The contribution of pretraining is most evident under the MSE metric, leading to about 22% performance improvement on the Dist-646 dataset (Table III) and 13% on the Adobe dataset (Table II) as shown by the comparison between **TD+TS+FGA w/o pretrain** and **TD+TS+FGA+pretrain**. This further supports our assumption that prior knowledge of image saliency is useful for trimap-free matting. In addition, in order to assess the proposed framework in the absence of pretraining data, we also test our full design without pretraining. Experimental results indicate that our w/o pretraining version already outperforms the SOTA trimap-free method, HAtt, on both Composition-1k and Dist-646 datasets, which again demonstrates the power of our network design. 6) To measure the contribution of the frequency input, we replace the wavelet components with the original RGB image and feed it directly to the FGA module. By comparing **TD+TS+FGA$_{rgb}$+msloss+Ref+pretrain** with **Ours**, we see that frequency input contributes greatly to our final design option. Table I, where the error is calculated on the unknown area, shows that FGA with frequency input can greatly improve gradient error. This meets our expectations that frequency information is leveraged as prior knowledge for improving model performance on fine image details.

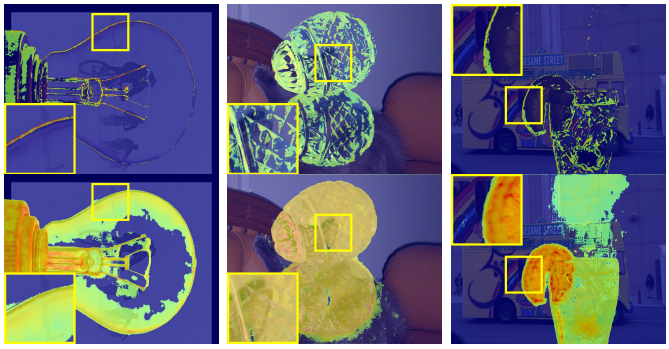


Fig. 9: Visualizations of the FGA module. The top row demonstrates the high-frequency guided attention maps while the bottom row shows the low-frequency guided ones.

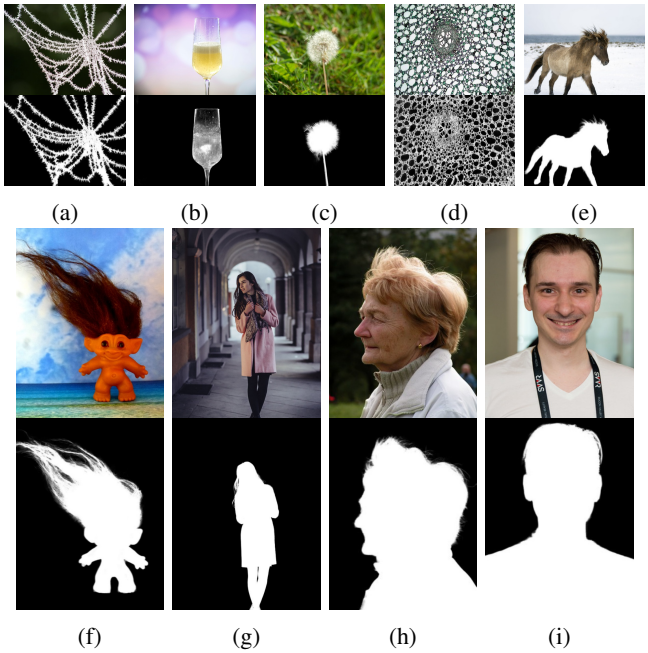
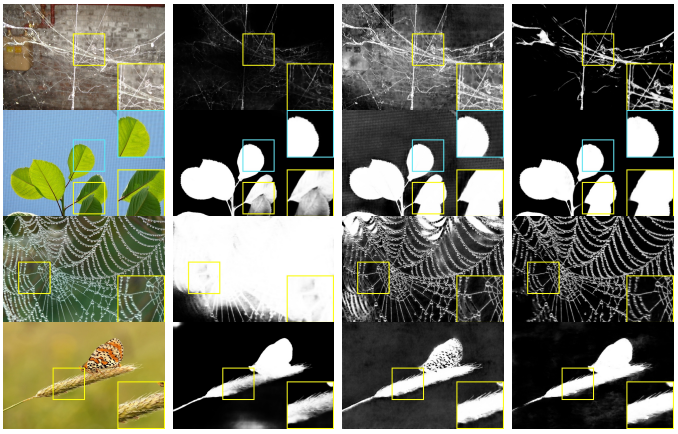


Fig. 10: TangleNet matting results on real-world images.

D. Attention map visualization

In Fig. 9, we visualize the learned FGA attention map in both high-frequency and low-frequency branches. Here we take the softmax output of the simplified non-local block as shown in Fig. 5, as this is the last stage of modelling the pixel spatial relationship in FGA. The visualization displays regions to which FGA provides larger weights as brighter color. By showing the enlarged details in yellow boxes from Fig. 9, we see that high-frequency guided attention maps assign more weights to high-frequency regions such as object edges or contours. In contrast, low-frequency guided maps focus more on the rough object silhouette and tend to highlight solid foreground areas. These attentive features help the network to adaptively treat different parts of the image and thus facilitate the tasks of alpha prediction and image segmentation.

This article has been accepted for publication in IEEE Transactions on Multimedia. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TMM.2022.3183403



(a) Orig (b) Late Fusion (c) DIM* (d) Ours

Fig. 11: Qualitative comparison with other matting algorithms on real-world images. DIM* means DIM with only RGB images as input.

E. Real-world image matting

The present method performs very well on synthetic datasets; to ensure this is not a consequence of overfitting we conduct comprehensive experiments on real-world data, including the well-known online benchmarking dataset, *alphamatting.com* dataset, as well as some public internet images.

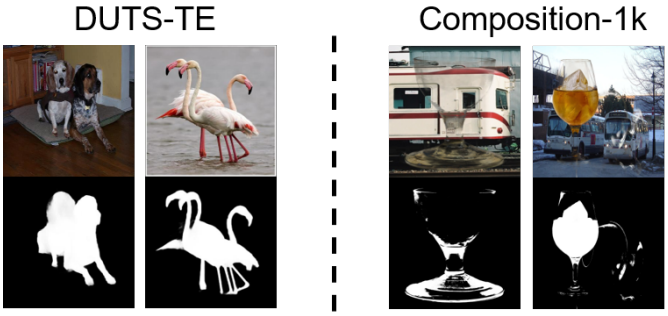
Internet images. The performance of TangleNet on real-world data is demonstrated in Fig. 10. We test TangleNet on different real-world objects, including glass, nets, plants, animals, and human (full-body and half-body portraits). We reuse the weights from previous experiments and test several real-world images from the internet. We observe that TangleNet produces accurate alpha mattes for complex foreground images without user input.

Comparison with other matting methods on real-world images. To further quantify robustness of the present method on real-world images, we conduct a qualitative comparison with other matting algorithms, such as Late Fusion [19] and DIM [14], on real-world scenarios, using their official implementations and weights. Fig. 11 shows that our method outperforms Late Fusion and DIM on various foreground objects by capturing more image details and extracting more accurate foregrounds. In addition, with the aid of saliency detection, our method excels at identifying the most salient object in the given image and tends to detect the salient foreground as an enclosed entity. Row 5 and Row 6 from Fig. 11 illustrate such scenarios where our method detects the true foreground object and produces a clear alpha matte whereas other matting methods which do not explicitly use foreground priors mistakenly recognize the foreground as separated parts (Row 5) or pick up part of the background.

Comparison with trimap-free methods trained without saliency prior. By explicitly exploiting saliency detection, our method is more sensitive to the most salient object in the given image and tends to detect the salient foreground as an enclosed entity. Fig. 12 shows scenarios where our method detects the true foreground object and produces an accurate alpha matte, whereas Late Fusion, a trimap-free method which does not explicitly use foreground priors, will recognize the foreground as separated parts (Row 1) or pick up part of the background (Row 2). Therefore, our experiment shows the benefit of using saliency knowledge as prior for matting tasks.

(a) Orig (b) Late Fusion (c) Ours

Fig. 12: Comparisons on real-world images with a trimap-free matting algorithm that makes no use of explicit saliency prior.



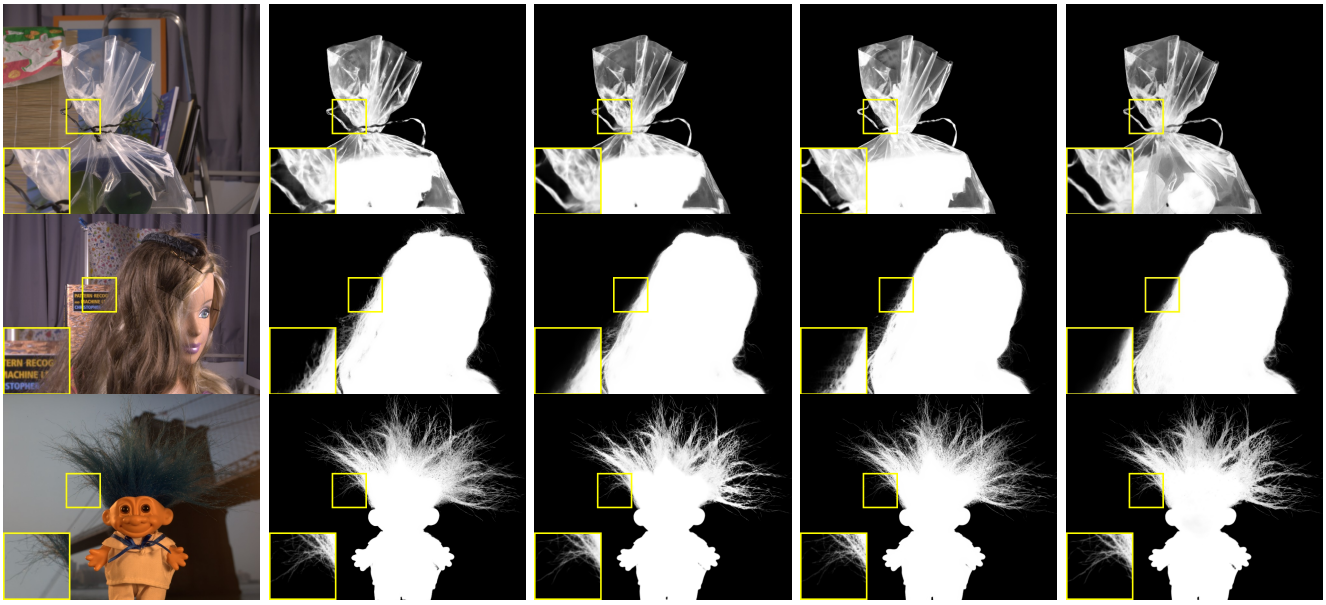
(a) Top row shows images with multiple foreground objects from DUTS-TE and Composition-1k datasets. Bottom row shows the corresponding matting results.

(b) Images. (c) w/o pretraining. (d) w/ pretraining.

Fig. 13: Results when multiple foreground objects are presented.

***alphamatting.com* dataset**. We show our results on the alphamatting.com benchmark to make comparisons with other SOTA trimap-based matting algorithms. Our qualitative results can be viewed from Fig. 14. Since our results are independent of different trimap settings given in the alphamatting.com, we use the same matting result for all different settings. Some numerical comparisons can be found in Table IV. Note that by taking only RGB images, our method is already outperforms some SOTA trimap-based methods (e.g. AlphaGAN [25], IndexNet, and DIM) in terms of MSE, Gradient, and Connectivity errors.

This article has been accepted for publication in IEEE Transactions on Multimedia. This is the author's version which has not been fully edited and content may change prior to final publication. Citation information: DOI 10.1109/TMM.2022.3183403



(a) Image (b) AlphaGAN (c) DIM (d) GCA (e) Ours

Fig. 14: Qualitative results on the alphamatting.com testing dataset.

TABLE IV: Our average ranking scores for four error metrics on the alphamatting.com benchmark together with other state-of-the-art and closely related methods. *S*, *L*, *U* denote average rankings of three trimap types, small, large and user, as given in the benchmark. Red, Green, and Blue indicate best, second best and third best performance. '†" indicates that the corresponding method takes only RGB images as input.

	Gradient↓				Connectivity↓				MSE↓				SAD↓			
Methods	*Overall rank*	*S*	*L*	*U*	*Overall rank*	*S*	*L*	*U*	*Overall rank*	*S*	*L*	*U*	*Overall rank*	*S*	*L*	*U*
CA [2]	15.6	16.9	16.5	13.5	27.2	29.3	26.3	26.1	18.6	22.4	20	13.4	24.4	28.5	22.8	22
DIM [14]	24.8	21.6	21.5	31.3	21.7	21.1	21	23	20.2	18.8	19.1	22.8	16.9	18	16.3	16.5
IndexNet [13]	19.5	18.3	18.3	22.1	26.8	25.3	27.8	27.3	21.2	23.8	19.8	20	20.6	22.9	19.4	19.5
AlphaGAN [25]	24.8	23.8	22.8	27.8	38.3	43.3	37.3	35	25.4	25.8	26.8	23.6	22.1	22.9	22.6	20.8
GCA [12]	14.2	12.8	15.6	12.9	23.6	27	21.3	22.6	16.2	16.3	15	17.4	15.3	16.3	12.6	16.9
Ours†	16.6	20.9	15	13.9	12.9	16.4	12.1	10	19.6	24.1	19.3	15.4	22.9	27.5	21.4	19.8

F. Performance in the presence of multiple foreground objects

Trimap-free natural image matting inevitably faces certain ambiguity with respect to the definition of foreground. To address this issue, we assume that the foreground consists of the "most salient object(s)", and consequently prior knowledge can be acquired via pretraining on salient object detection tasks. As such, our network will extract at least one foreground object from the given image if it is considered a salient object. There are many examples in DUT-TE and Composition-1k testing sets that have multiple foreground objects in one image. We demonstrate some of them in Fig. 13a. We also show the importance of pretraining on salient object detection by making a qualitative comparison between the w/ pretraining and w/o pretraining results. We conduct our experiment on real-world images. Fig. 13a and Fig. 13c show that the network may pick up some background objects from the scene (such as the candle in Fig. 13b) if it is only trained on the matting task. In contrast, the pretrained counterpart is able to successfully distinguish foreground and background objects.

Fig. 15: Some failure cases of our method.

G. Limitations

Our method generalizes well on the real-world data, particularly for the cases where only one salient object exists in the image or the levels of saliency across different foreground

objects are evident. On the other hand, it has some difficulties in the presence of multiple foreground objects that have similar levels of saliency, as there is a higher chance for saliency detection to misidentify the end user's actual target. For example, if two foreground objects (goal net and player) overlap as shown in the first row of Fig. 15, our network extracts the most salient foreground object (goal net) but also takes into account part of the less salient object (number sign on the player) that stands out due to its similar color. In the second row, because the two objects (horse and lawn) are attached, our method recognizes them as a single salient foreground which is a failure case if the user only wants to extract the horse. As such, our method is most suitable for images where the target foreground object(s) can be unambiguously identified via saliency detection. We leave the problem of addressing more challenging scenarios (e.g., images with overlapping/attached foreground objects) for future work.

H. Runtime comparison

Our naive implementation takes about $0.4s$ to predict one image from the Adobe Composition-1k dataset on average. We also evaluate the computational efficiency of 3 other SOTA matting methods by running their official implementation on the Composition-1k dataset and record their average runtimes in Table V. It can be seen that the proposed TangleNet has a good balance between performance and runtime efficiency. Note that our method operates in a trimap-free manner which is more difficult than the trimap-based matting, thus requires more computing resources to handle. Moreover, though our model has the most number of parameters, its inference time is still comparable to other SOTA matting methods.

TABLE V: Runtime comparison with other SOTA methods. The input images are resized to 800×800 and the implementation is carried out on a workstation with one NVIDIA RTX 2080Ti GPU. "†" indicates that the corresponding method takes only RGB images as inputs.

Methods	Parameters(10^6)	Runtime(s)
Late Fusion [19]†	37.9	0.37
GCA [12]	25.0	0.31
DIM [14]	28.2	0.22
Ours†	53.9	0.40

VI. Conclusion

We have presented a novel learning framework that can perform trimap-free natural image matting. Equipped with task switchers (TS), the proposed network implements joint learning to better utilize correlated and shared feature information of alpha prediction and binary segmentation. The frequency-guided attention (FGA) module leverages frequency domain features to guide alpha prediction. The network is pretrained for salient object detection, and we introduce a multi-scale hybrid loss to exploit the benefits of salient object detection to trimap-free image matting. Extensive experiments demonstrate that the proposed method outperforms all other trimap-free matting algorithms and achieves comparable results against state-of-the-art trimap-based algorithms. We hope further progress can be made on trimap-free image matting using the ideas and experience of the present work.

References

[1] Y. Qiao, Y. Liu, X. Yang, D. Zhou, M. Xu, Q. Zhang, and X. Wei, "Attention-guided hierarchical structure aggregation for image matting," in *CVPR*, 2020.

[2] Q. Hou and F. Liu, "Context-aware image matting for simultaneous foreground and alpha estimation," in *ICCV*, 2019.

[3] A. R. Smith and J. F. Blinn, "Blue screen matting," in *SIGGRAPH*, 1996.

[4] Y. Aksoy, T. Ozan Aydin, and M. Pollefeys, "Designing effective inter-pixel information flow for natural image matting," in *CVPR*, 2017.

[5] L. Grady, T. Schiwietz, S. Aharon, and R. Westermann, "Random walks for interactive alpha-matting," in *VIIP*, 2005.

[6] A. Levin, D. Lischinski, and Y. Weiss, "A closed-form solution to natural image matting," *PAMI*, 2007.

[7] Q. Chen, D. Li, and C.-K. Tang, "Knn matting," *PAMI*, 2013.

[8] X. Feng, X. Liang, and Z. Zhang, "A cluster sampling method for image matting via sparse coding," in *ECCV*, 2016.

[9] K. He, C. Rhemann, C. Rother, X. Tang, and J. Sun, "A global sampling method for alpha matting," in *CVPR*, 2011.

[10] E. Shahrian and D. Rajan, "Weighted color and texture sample selection for image matting," in *CVPR*, 2012.

[11] D. Cho, Y.-W. Tai, and I. Kweon, "Natural image matting using deep convolutional neural networks," in *ECCV*, 2016.

[12] Y. Li and H. Lu, "Natural image matting via guided contextual attention," in *AAAI*, 2020.

[13] H. Lu, Y. Dai, C. Shen, and S. Xu, "Indices matter: Learning to index for deep image matting," in *ICCV*, 2019.

[14] N. Xu, B. Price, S. Cohen, and T. Huang, "Deep image matting," in *CVPR*, 2017.

[15] F. Zhou, Y. Tian, and Z. Qi, "Attention transfer network for nature image matting," *TCSVT*, 2021.

[16] Q. Chen, T. Ge, Y. Xu, Z. Zhang, X. Yang, and K. Gai, "Semantic human matting," in *ACMMM*, 2018.

[17] J. Liu, Y. Yao, W. Hou, M. Cui, X. Xie, C. Zhang, and X.-s. Hua, "Boosting semantic human matting with coarse annotations," in *CVPR*, 2020.

[18] S. Sengupta, V. Jayaram, B. Curless, S. M. Seitz, and I. Kemelmacher-Shlizerman, "Background matting: The world is your green screen," in *CVPR*, 2020.

[19] Y. Zhang, L. Gong, L. Fan, P. Ren, Q. Huang, H. Bao, and W. Xu, "A late fusion cnn for digital matting," in *CVPR*, 2019.

[20] J. Wang and M. F. Cohen, "An iterative optimization approach for unified image segmentation and matting," in *ICCV*, 2005.

[21] Y.-Y. Chuang, B. Curless, D. H. Salesin, and R. Szeliski, "A bayesian approach to digital matting," in *CVPR*, 2001.

[22] J. Sun, J. Jia, C.-K. Tang, and H.-Y. Shum, "Poisson matting," in *SIGGRAPH*, 2004.

[23] J. Wang and M. F. Cohen, "Optimized color sampling for robust matting," in *CVPR*, 2007.

[24] S. Cai, X. Zhang, H. Fan, H. Huang, J. Liu, J. Liu, J. Liu, J. Wang, and J. Sun, "Disentangled image matting," in *ICCV*, 2019.

[25] S. Lutz, K. Amplianitis, and A. Smolic, "Alphagan: Generative adversarial networks for natural image matting," in *BMVC*, 2018.

[26] J. Tang, Y. Aksoy, C. Öztireli, M. Gross, and T. O. Aydın, "Learning-based sampling for natural image matting," in *CVPR*, 2019.

[27] Y. Liu, J. Xie, Y. Qiao, Y. Tang, and X. Yang, "Prior-induced information alignment for image matting," *IEEE Transactions on Multimedia*, 2021.

[28] D. Bahdanau, K. Cho, and Y. Bengio, "Neural machine translation by jointly learning to align and translate," *arXiv preprint arXiv:1409.0473*, 2014.

[29] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *NeurIPS*, 2017.

[30] J. Hu, L. Shen, S. Albanie, G. Sun, and A. Vedaldi, "Gather-excite: Exploiting feature context in convolutional neural networks," in *NeurIPS*, 2018.

[31] J. Hu, L. Shen, and G. Sun, "Squeeze-and-excitation networks," in *CVPR*, 2018.

[32] X. Wang, R. Girshick, A. Gupta, and K. He, "Non-local neural networks," in *CVPR*, 2018.

[33] S. Woo, J. Park, J.-Y. Lee, and I. S. Kweon, "Cbam: Convolutional block attention module," in *ECCV*, 2018.

[34] Y. Cao, J. Xu, S. Lin, F. Wei, and H. Hu, "Gcnet: Non-local networks meet squeeze-excitation networks and beyond," in *ICCVW*, 2019.

[35] J. Fu, J. Liu, H. Tian, Y. Li, Y. Bao, Z. Fang, and H. Lu, "Dual attention network for scene segmentation," in *CVPR*, 2019.

[36] Z. Huang, X. Wang, L. Huang, C. Huang, Y. Wei, and W. Liu, "Ccnet: Criss-cross attention for semantic segmentation," in *ICCV*, 2019.

[37] A. Kendall, Y. Gal, and R. Cipolla, "Multi-task learning using uncertainty to weigh losses for scene geometry and semantics," in *CVPR*, 2018.

[38] S. Vandenhende, S. Georgoulis, and L. Van Gool, "Mti-net: Multi-scale task interaction networks for multi-task learning," in *ECCV*, 2020.

[39] Z. Zhang, Z. Cui, C. Xu, Z. Jie, X. Li, and J. Yang, "Joint task-recursive learning for semantic segmentation and depth estimation," in *ECCV*, 2018.

[40] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *PAMI*, 1998.

[41] Y. Pang, X. Zhao, L. Zhang, and H. Lu, "Multi-scale interactive network for salient object detection," in *CVPR*, 2020.

[42] X. Qin, Z. Zhang, C. Huang, C. Gao, M. Dehghan, and M. Jagersand, "Basnet: Boundary-aware salient object detection," in *CVPR*, 2019.

[43] J.-X. Zhao, J.-J. Liu, D.-P. Fan, Y. Cao, J. Yang, and M.-M. Cheng, "Egnet: Edge guidance network for salient object detection," in *ICCV*, 2019.

[44] B. Jiang, Z. Zhou, X. Wang, J. Tang, and B. Luo, "cmsalgan: Rgb-d salient object detection with cross-view generative adversarial networks," *IEEE Transactions on Multimedia*, 2021.

[45] X. Liu, Y. Ma, Z. Shi, and J. Chen, "Griddehazenet: Attention-based multi-scale network for image dehazing," in *ICCV*, 2019.

[46] Y. Zhang, Y. Tian, Y. Kong, B. Zhong, and Y. Fu, "Residual dense network for image super-resolution," in *CVPR*, 2018.

[47] L. Dai, X. Liu, C. Li, and J. Chen, "Awnet: Attentive wavelet network for image isp," in *ECCVW*, 2020.

[48] P. Liu, H. Zhang, K. Zhang, L. Lin, and W. Zuo, "Multi-level wavelet-cnn for image restoration," in *CVPRW*, 2018.

[49] X. Luo, J. Zhang, M. Hong, Y. Qu, Y. Xie, and C. Li, "Deep wavelet network with domain adaptation for single image demoireing," in *CVPRW*, 2020.

[50] P.-T. De Boer, D. P. Kroese, S. Mannor, and R. Y. Rubinstein, "A tutorial on the cross-entropy method," *Annals of Operations Research*, 2005.

[51] Z. Wang, E. P. Simoncelli, and A. C. Bovik, "Multiscale structural similarity for image quality assessment," in *ACSSC*, 2003.

[52] G. Máttyus, W. Luo, and R. Urtasun, "Deeproadmapper: Extracting road topology from aerial images," in *ICCV*, 2017.

[53] L. Wang, H. Lu, Y. Wang, M. Feng, D. Wang, B. Yin, and X. Ruan, "Learning to detect salient objects with image-level supervision," in *CVPR*, 2017.

[54] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, and C. L. Zitnick, "Microsoft coco: Common objects in context," in *ECCV*, 2014.

[55] Y. Dai, H. Lu, and C. Shen, "Learning affinity-aware upsampling for deep image matting," in *CVPR*, 2021.

[56] Y. Liu, J. Xie, X. Shi, Y. Qiao, Y. Huang, Y. Tang, and X. Yang, "Tripartite information mining and integration for image matting," in *ICCV*, 2021.

[57] M. Everingham, L. Van Gool, C. K. Williams, J. Winn, and A. Zisserman, "The pascal visual object classes (voc) challenge," *IJCV*, 2010.

[58] C. Rhemann, C. Rother, J. Wang, M. Gelautz, P. Kohli, and P. Rott, "A perceptually motivated online benchmark for image matting," in *CVPR*, 2009.

Linhui Dai received the B.E. degree (Hons.) in the year 2019 from Electrical Engineering at McMaster University. In 2021 he received the M.A.Sc. degree from the Department of Electrical and Computer Engineering at McMaster University, supervised by Dr. Jun Chen. His research interests include image matting, low-level vision, and machine-learning algorithms. He was the recipient of the 2019 Vector Institute Scholarship.

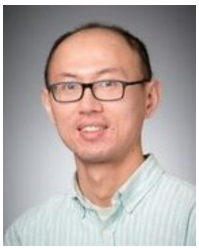
Xiang Song received the B.E degree in Electrical Engineering from McMaster University, in 2019, and the M.A.Sc. degree in Electrical and Computer Engineering from McMaster, in 2021. His research interests include image dehazing/deraining, video super-resolution, and other low-level computer vision problems.

Xiaohong Liu received the B.E. degree in communication engineering from Southwest Jiaotong University, China, in 2014, the M.A.Sc. degree in electrical and computer engineering from University of Ottawa, Canada, in 2016, and the Ph.D. degree in electrical and computer engineering from McMaster University, Canada, in 2021.

Dr. Liu is a tenure-track assistant professor with the John Hopcroft Center at Shanghai Jiao Tong University. His research interests include video super-resolution/interpolation, image dehazing/deraining, and image forgery detection. He received the Ontario Graduate Scholarship in 2019, NSERC Alexander Graham Bell Canada Graduate Scholarship-Doctoral, and Borealis AI Global Fellowship award in 2020. He serves as the reviewer for several IEEE journals, including IEEE TRANS. PATTERN ANAL. MACH. INTELL., IEEE TRANS. IMAGE PROCESS., IEEE TRANS. MULTIMEDIA, IEEE TRANS. CIRCUITS SYST. VIDEO TECHNOL., and IEEE TRANS. INTELL. TRANSP. SYST.

Chengqi Li received the B.E. degree in microelectronic from San Yat-sen University, Guangzhou, China, in 2012, and the M.A.Sc. degree in electrical and computer engineering from the McMaster University, Canada, in 2021. He is currently pursuing the Ph.D. degree with the Department of Computing and Software, McMaster University, Canada. His research areas include video super-resolution and video frame interpolation.

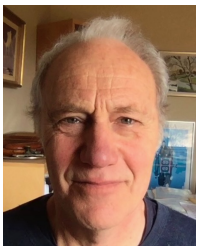
Zhihao Shi received the B.E. degree in communication engineering from Zhengzhou University, China, in 2018. He is currently pursuing the Ph.D. degree in the Department of Electrical and Computer Engineering, McMaster University, Canada. His research interests include image dehazing/deraining, video super-resolution and other low-level computer vision problems.



Jun Chen (Senior Member, IEEE) received the B.E. degree in communication engineering from Shanghai Jiao Tong University, Shanghai, China, in 2001 and the M.S. and Ph.D. degrees in electrical and computer engineering from Cornell University, Ithaca, NY, USA, in 2004 and 2006, respectively.

He was a Postdoctoral Research Associate with the Coordinated Science Laboratory, University of Illinois at Urbana-Champaign, Urbana, IL, USA, from September 2005 to July 2006, and a Postdoctoral Fellow at the IBM Thomas J. Watson Research Center, Yorktown Heights, NY, USA, from July 2006 to August 2007. Since September 2007, he has been with the Department of Electrical and Computer Engineering, McMaster University, Hamilton, ON, Canada, where he is currently a Professor. His research interests include information theory, machine learning, wireless communications, and signal processing.

Dr. Chen was a recipient of the Josef Raviv Memorial Postdoctoral Fellowship in 2006, the Early Researcher Award from the Province of Ontario in 2010, the IBM Faculty Award in 2010, the ICC Best Paper Award in 2020, and the JSPS Invitational Fellowship in 2021. He held the title of the Barber-Gennum Chair in Information Technology from 2008 to 2013 and the Joseph Ip Distinguished Engineering Fellow from 2016 to 2018. He served as an Editor of the IEEE TRANSACTIONS ON GREEN COMMUNICATIONS AND NETWORKING from 2020 to 2021 and is currently an Associate Editor of the IEEE TRANSACTIONS ON INFORMATION THEORY



Martin Brooks was born in Montreal in 1952. He received the B.A. degree in mathematics from MIT in 1974, and Ph.D. degree in computer science from Stanford University in 1980. During the 1980s he carried out robotics and AI research at the Centre for Industrial Research (now SINTEF) in Oslo, and from 1991 - 2007 at National Research Council Canada, where he led the machine learning and AI research groups. From 2007 - 2014 Dr. Brooks performed mathematical research and wrote automated trading algorithms for hedge fund Apollo Systems Research, after which he founded startup ShapeVision Inc. His current research focuses on applications of computational topology to images, including use in AI, medical imaging, traffic management, and art.