JCSE Journal of Computing Science and Engineering

Regular Paper

Journal of Computing Science and Engineering,
Vol. 18, No. 2, June 2024, pp. 93-115

TGMatting: Automatic Image Matting Based on Trimap Generation

Jianming Wang*
School of Mathematics and Computer, Dali University, Dali, China;
Yunnan Provincial Key Laboratory of Entomological Biopharmaceutical R&D, Dali University, Dali, China
wjm@dali.edu.cn

Xiao Jiang, Yuhang Zhang
School of Mathematics and Computer, Dali University, Dali, China
jiangxiao@stu.dali.edu.cn,zyh@stu.dali.edu.cn

Jiting Yin
Dali Forestry and Grassland Science Research Institute, Dali, China
yinjiting234@dali.edu.cn

Zizhong Yang
Yunnan Provincial Key Laboratory of Entomological Biopharmaceutical R&D, Dali University, Dali, China
yangzizhong@dali.edu.cn

Abstract
At present, automatic image matting methods primarily focus on portraits and hard edge targets, which gives them a limited ability to deal with low-resolution, complex, and blurred non-portrait targets. To address these issues, the current paper proposes an automatic image matting method called TGMatting. This method automatically optimizes Trimap generation through three modules: the U^2S-Net pre-segmentation module, which is based on the U^2-Net network, enhances segmentation by removing null convolutions and reducing oversampling interference; the BGTrimap module, which is also based on U^2-Net, refines edge regions using optimized dilation-erosion methods and Manhattan distance for seed point sparsification, thus ensuring accurate region growth and background information removal; and in the last module, edges are transformed into mixed-pixel regions using Sobel operator and non-local-means denoising binarization, and a Trimap map is automatically generated by combining OTSU segmentation with pre-segmentation results, thus achieving fully automated processing. Finally, a transparency mask is obtained via FBAMatting, which enables interaction-free automatic matting. The experimental results demonstrate that the improved U^2S-Net network reduces MAE by 0.003 on the SOD-Spider test set, enhances accurate detection of significant regions in low-resolution images compared to U^2S-Net, and reduces BGTrimap's sum of absolute difference value by about 10% compared to other Trimap generation methods in IFMatting and KNN Matting.

Category: Computer Graphics / Image Processing

Keywords: Image matting; Trimap generation; Deep learning; Saliency object detection (SOD)

Open Access **http://dx.doi.org/10.5626/JCSE.2024.18.2.93** **http://jcse.kiise.org**

Received 22 May 2024; **Accepted** 10 June 2024
*Corresponding Author

 pISSN: 1976-4677 eISSN: 2093-8020

I. INTRODUCTION

Image matting, the purpose of which is to finely segment the foreground and background, has a wide range of applications in the field of data enhancement as well as the film and television special effects industries [1]. Unlike other image segmentation methods, semantic segmentation [2] and salient object detection (SOD) [3] can achieve coarse segmentation of foreground and background with 0/1 classification, while image matting, and specifically the image matting method used in one study that is based on the Trimap [4], can achieve segmentation effects for details as fine as hair. This fine segmentation is key to achieving precise results, but it can be more difficult to obtain the trimap. Therefore, this paper strives to realize an automatic image fine-matting method based on image matting that does not require user interaction.

For general image segmentation, although semantic segmentation and saliency target detection—in addition to other coarse segmentation methods—can achieve 0/1 classification of the foreground and background of the image, they fail to produce a smooth transition at the boundary [4]. Recent years have seen significant advancements in segmentation networks based on large models. Examples of such advancements include Segment Anything Model (SAM) [5], OpenSeeD [6], CLIP-Driven Universal Model [7], DSC-Net [7], and other segmentation models, all of which are still coarse segmentation methods. Although they have achieved better results in panoramic segmentation and can understand semantics to a large extent, they cannot capture fine detail information such as hairs. However, image matting is the only method that has been shown to be capable of achieving automatic fine segmentation. It mainly relies on predicting the transparency of mixed pixel regions, with its core being the auxiliary input of the Trimap [4]. However, obtaining a Trimap is challenging, as it typically requires manual interaction based on the original image. Such methods entail heavy workloads and cumbersome operations, and the need for manual interaction limits the ability to implement automatic image matting [1].

The existing Trimap generation methods involving human interaction mainly rely on traditional image processing methods. For instance, Yao [8] utilizes dynamic threshold segmentation to generate a Trimap based on human annotations. Li et al. [9] employ super-resolution to predict three regions of the image based on strokes and set a threshold to generate the Trimap. While these methods can retain more details, they obtain limited amounts of information from annotations. Consequently, the generated Trimap may contain significant interference from the background region, thus hindering automatic matting.

Automatic image matting methods mainly comprise Trimap-based methods and end-to-end training methods. The automatic Trimap generation methods are based on pre-segmentation results, which are used as a priori information in automatic Trimap generation. Therefore, the key to achieving image matting without human interaction lies in the automatic generation of pre-segmentation and the Trimap. The end-to-end training method integrates Trimap generation into the overall model architecture of image matting, ultimately establishing a separate Trimap generation module. This module automatically utilizes Trimap information during the matting process to achieve interaction-free automation [10]. Some studies have achieved automatic Trimap generation and image matting based on it. For portrait images, automatic Trimap generation is mainly achieved either through pre-segmentation based on three classifications of portraits or by detecting portrait information. For hard edge targets, automatic Trimap generation primarily involves pre-segmentation followed by expansion and erosion. Ran and Feng [11] used the pre-segmentation result of semantic segmentation as well as mapping from 0/1 to 0/255 to generate the Trimap using a fixed threshold to achieve automatic portrait matting. However, the quality of the Trimap obtained using this method is poorer for more complex portrait foregrounds, and it can only target portraits, which limits its applicability. In an approach focusing on hard edge non-portrait images, Wang et al. [12] employed semantic segmentation to pre-segment the foreground of the car. Through expansion and erosion, they obtained the Trimap. As the edges of the car image belong to hard edges, simple morphological processing can yield better results. However, this is not applicable in the complex image edges' goal of unrestricted semantics. Some methods aim to achieve the automatic generation of semantically unrestricted Trimap without interaction by mainly relying on pre-segmentation using traditional image processing methods. For instance, Henry and Lee [13] generate the Trimap by combining pre-segmentation results with FCM clustering, while Gupta and Raman [14] generate the Trimap by over-segmenting the image and combining it with k-means clustering. Cho et al. [15] generate the Trimap by expanding the pre-segmentation results and then employing dynamic brushes and downsampling. However, most of the above methods require processing times ranging from more than 10 seconds to 1 minute while also being less effective for images with low resolution and complex backgrounds, which ultimately makes them unsuitable for automatic matting tasks.

The aforementioned matting methods, which rely on automatic Trimap generation, are implemented through pre-segmentation techniques with the ultimate aim of providing rudimentary binary classification details. At present, the principal pre-segmentation approaches encompass semantic segmentation and saliency target detection. However, semantic segmentation methodologies rely on extensive training data, which constrains their performance and scalability [16]. Conversely, saliency

target detection methods tend to delineate significant image components [3], which proves advantageous in mitigating redundant background details while also offering greater adaptability in single-objective dichotomous classifications, ultimately rendering them more suitable for subsequent Trimap generation. Early saliency target detection strategies typically rely on fully convolutional networks [17]. However, with the advent of U-Net [18], U-Net-based algorithms, as exemplified by U^2-Net devised by Qin et al. [19], demonstrate superior results, which can be attributed to their nested U-Net architecture, which is adept at capturing contextual nuances. By contrast, modified GAN-cAED [20] leverages a novel generative adversarial network (GAN) approach for image fusion, although the segmentation efficacy of this model may suffer from insufficient training data or inaccurate labeling. SThy-Net [21] encompasses multiple hyperparameters, which complicates the quest for optimal configurations and heightens the intricacy and challenge of its implementation. Meanwhile, DeepDR Plus [22] harnesses convolutional neural network (CNN) models for feature extraction from fundus images, yet its predictive outcomes merely represent best estimations that are based on existing data and model capabilities. A novel fuzzy extensive learning system introduced by Ali et al. [23] offers promising prospects, although channel extraction is needed to maximize its segmentation effectiveness, and its limited generalization ability poses challenges in extending its application to complex image segmentation tasks.

Matting methods based on end-to-end training have become an important research topic in recent years. However, most of these methods focus on end-to-end portrait matting. For instance, the Alibaba team proposed semantic human matting (SHM) [10], which achieves end-to-end portrait matting by predicting rough classification information of the human body using the T-Net module to obtain a Trimap. Another method, MODNet [24], performs automatic matting of human images by predicting rough information of the human body within the network and then distinguishing between mixed pixel regions and foreground backgrounds.

There have been fewer studies examining end-to-end matting of non-human nature images. Li et al. [25] proposed an end-to-end matting network, AIM, which predicts the generalized Trimap of any image containing humans and animals in the form of a unified semantic representation. The learned semantic features guide the matting network to focus on the transition region through an attention mechanism. In another study, Li et al. [26] proposed a scanning and focusing matting network (GFM) that employs a shared encoder and two independent decoders to collaboratively learn the two tasks. This method utilizes three semantic and transition region representations to achieve end-to-end image matting. An animal matting dataset (AM-2k) was also created to facilitate the end-to-end matting task.

However, such methods have higher requirements for image saliency. In particular, it is not possible for most tiny, non-significant targets to be captured in the foreground, which results in either a blank segmentation foreground or the presence of a large residual. To achieve better results, it is necessary to use more samples of non-significant fine segmentation of such targets for training, which incurs an extremely high labeling cost.

The deep image matting method proposed by Xu et al. [4] utilizes deep learning models to achieve accurate segmentation of images, thereby capturing subtle details in the image and accurately extracting foreground objects. However, it typically requires the use of a large amount of annotated data in training the deep learning models, which requires manual annotation of foreground objects and backgrounds, thus incurring high costs and lengthy training times. The method proposed by Yu et al. [27] can also accurately extract the boundaries between the foreground and the background, thus achieving high-quality image cutout effects. However, there are also issues that arise with high training costs and complexity.

In summary, the current Trimap-based automatic matting methods and end-to-end matting methods primarily focus on portraits or hard edge objects, which poses limitations when matting non-portrait targets with low resolution and complex edges [28]. Precise segmentation methods for non-human images also face problems such as long training times and high complexity. In particular, matting soft-edge targets such as insects, spiders, and other non-human subjects prove more challenging due to their complex edges. These targets are also nocturnal hunters and easily startled, which makes image acquisition difficult; images of these subjects often suffer from deficiencies in clarity, resolution, and other aspects. Most of these targets feature long limbs, spindles, and other intricate organs, often with dense hair, thus causing them to be classified as complex soft-edge objects [31]. Therefore, it is difficult to achieve automatic matting for such images with low resolution or complexity, or with blurred foreground edges.

To address these issues, the current paper proposes an automatic image fine-matting method that combines the improved U^2-Net network and automatic optimization of Trimap generation. First, based on U^2-Net, we enhance the effectiveness of the model with low-resolution images and construct the U^2S-Net pre-segmentation module. Then, using the pre-segmentation results, we employ region growing algorithms and Sobel operator edge detection to swiftly generate the high-quality Trimap. Finally, we utilize the image matting method to complete the automatic matting process. Using this approach, we successfully achieve automatic fine matting of low-resolution non-portrait images with complex and blurred foreground edges.

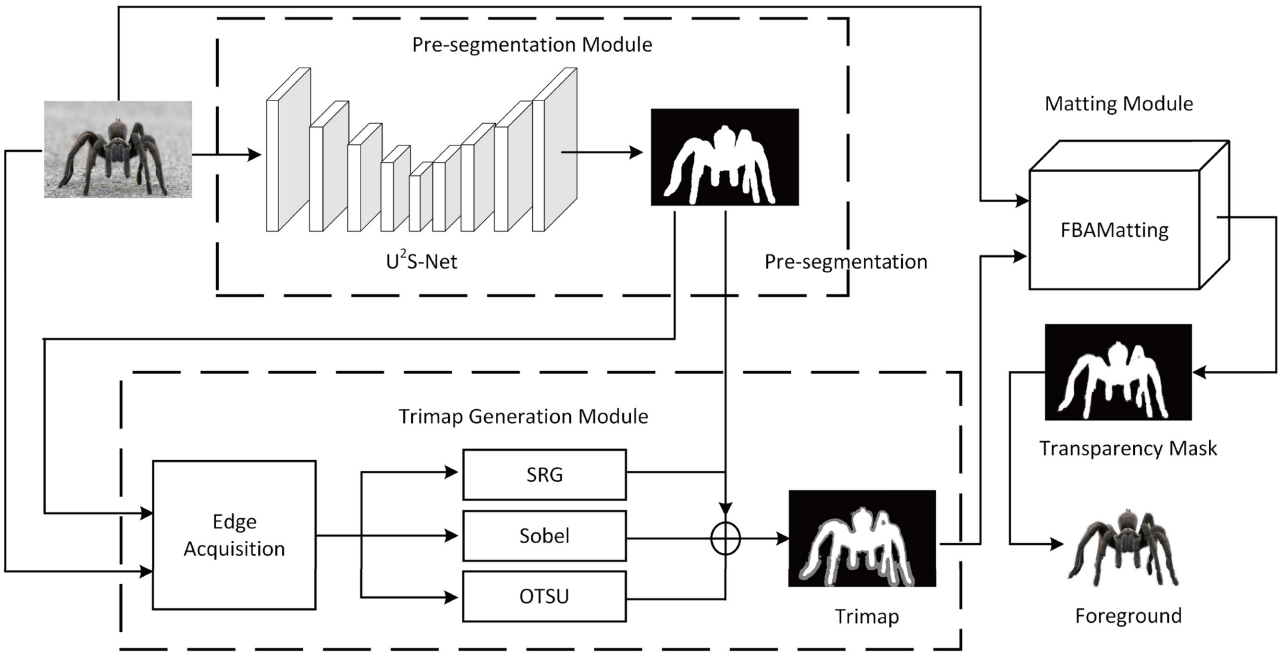

Fig. 1. TGMatting.

II. TGMATTING AUTOMATIC MATTING METHOD

In this paper, to address the problems described in the previous paper, an automatic matting network based on Trimap Generates Matting Network (TG-Matting) is constructed that consists of the U^2S-Net pre-segmentation module, BGTrimap automatic Trimap generation module, and FBAMatting matting module, as shown in Fig. 1.

The U^2S-Net pre-segmentation module is based on the saliency target detection method, which generates the corresponding saliency map by calculating the saliency value of each pixel in the image, and this reduces the interference caused by the excessive downsampling based on U^2S-Net. Meanwhile, the BGTrimap module obtains the edge region by optimizing the expansion and erosion algorithm, removes the background redundant information by using the seed point set with Manhattan distance constraints, detects the edge by using the Sobel operator, and then transforms it into a mixed-pixel region by using NML denoising binarization. Finally, it obtains the Trimap based on the results of the pre-segmentation and Otsu thresholding results. The matting module uses the FBAMatting method to obtain the final transparency mask. The spider species segmentation dataset SOD-Spider and the spider species matting dataset Alpha-Spider are used for validation. Most of the above data suffer from problems such as low resolution, complex and fuzzy foreground edges, etc. The current paper conducts a comparative study in which low-resolution images are defined as images with a foreground target area lower than 200×200 pixels or less, and where foreground targets account for 20% or less of the overall pixels. This provides a better way to validate the effect of the TGMatting method on this type of data.

A. U^2S-Net Pre-segmented Network

Pre-segmentation is the first step of the automatic matting method described in this paper. It aims to obtain the rough classification information regarding foreground and background in the image, wherein each pixel point of the image is classified in a binary manner. This paper is based on the U^2S-Net network and constructed using the saliency target detection U^2S-Net pre-segmentation module.

Saliency target detection methods generally use VGG or ResNet networks that have been trained on the ImageNet dataset as the backbone network, which is used to replace the coding module to extract deep features. However, these networks are designed for image classification, and the semantic features they extract are different from the global and local information needed for segmentation tasks. This makes it difficult to obtain the feature information in regions where saliency is not obvious. Meanwhile, the U^2-Net model constructed by Qin et al. [19] does not use a backbone network to extract features but instead uses the traditional U-Net splicing structure to obtain contextual information. This structure can better learn the semantic information in the training images and reduce the interference of the traditional backbone network in feature extraction. This also allows it to effectively improve the training speed and final results of the single semantic pre-segmentation work.

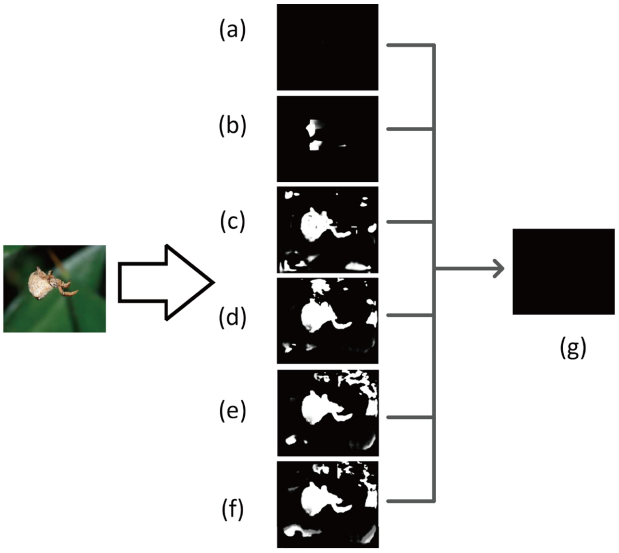

Fig. 2. Output of each layer U^2-Net.

However, the codec of U^2-Net employs six layers of RSU modules (Residual U-blocks), in a manner similar to the U-Net structure, for stacking. This structure downsamples the image to 1/24 of the original image, resulting in a low-resolution image. However, excessive downsampling can lead to a loss of too much detail information in the bottom layer. Simultaneously, the model splices the output of each RSU module in the decoding layer, jointly calculates the loss of each layer to obtain the final loss, and outputs the final result in the top layer RSU module. However, when applying this method to low-resolution images as was done in the current work, it is difficult to distinguish the saliency value of the entire image, which results in a poor final output. This can also cause significant interference, as illustrated in Fig. 2.

The resolution of the input image in Fig. 2 is 240×204 pixels. Outputs a to f represent the results obtained by the decoder from the low layer to the high layer. However, outputs a and b lose many details, which can interfere with the final result when spliced. The bottom layer of the cavity convolution module downsamples the image from 1/12 to 1/24; it also calculates the output image loss during this process. While this method allows for deeper feature extraction in image coding and decoding, for low-resolution images, the model lacks a valuable reference that can assist in computing the final output loss. This results in a loss of significance prediction in the final output region. When used with low-resolution images with complex and blurred foreground edges, the U^2-Net method can easily lead to a loss of key details.

To address the aforementioned issues, this paper enhances the U^2-Net model, which serves as the pre-segmentation module in this study to ensure accurate coarse segmentation results. The specific structure of the constructed U^2S-Net is depicted in Fig. 3.

The overall structure of U^2S-Net is based on U^2-Net, which employs a nested U-Net structure. However, U^2S-Net excludes the sixth layer of the RSU module, as it is replaced with a null convolution layer, a pooling layer, and a ReLU function. The RSU structure of the sixth layer is consistent with that of the preceding and subsequent layers. Instead of the RSU module from the bottom layer, the RSU module from the fifth layer is used to reduce model depth, which enhances the segmentation efficacy of each layer and improves the obtained results, particularly for low-resolution images after final splicing.

The loss function of U^2-Net aggregates the losses of the six RSU modules in the decoding layer to form the final loss. The training loss of the enhanced U^2S-Net with reduced depth is as follows:

$$L = \sum_{m=1}^{M} w_m l_m + w_s l_s . \tag{1}$$

Here, unlike in the original loss function, the number

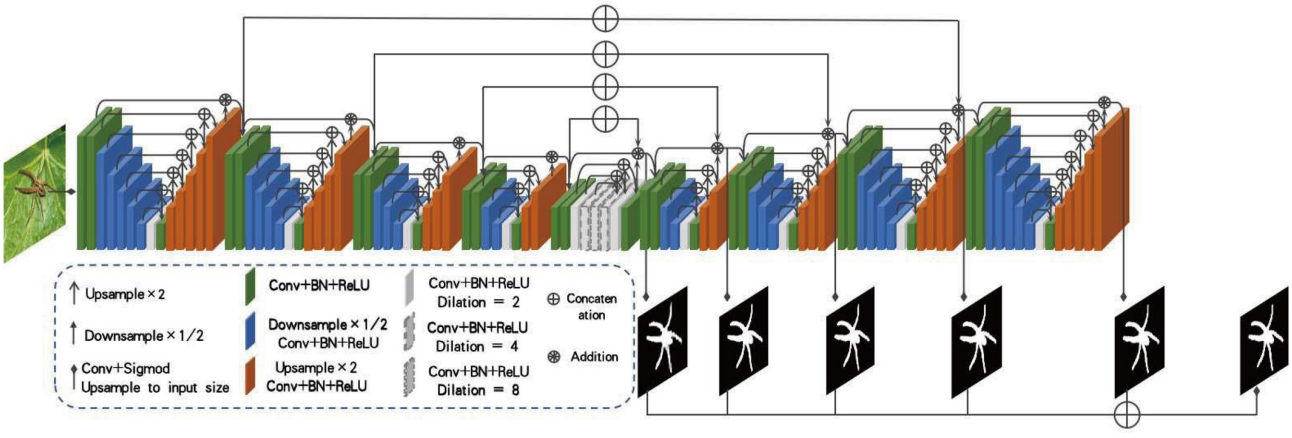

Fig. 3. U^2S-Net model structure.

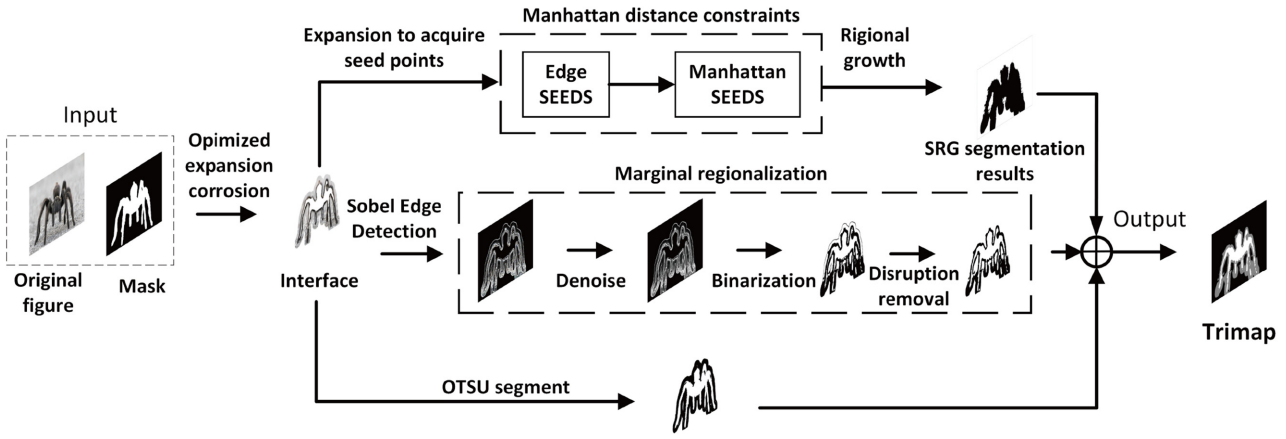

Fig. 4. BGTrimap overall process.

of layers M in RSU is 5. Moreover, l_m denotes the loss of each layer's output, while w_m signifies the weight that is assigned to each layer. Lastly, l_s and w_s, respectively represent the loss and the weight of the final result after splicing.

B. BGTrimap AutoTrimap Generation Module

As the core method of automatic matting in this paper, the Trimap module needs to accurately classify the foreground, background, and mixed pixels of an image. It is difficult to obtain an accurate Trimap, and it typically requires manual annotation, particularly for images with low resolution, complex details, and blurred foreground edges. To address these challenges, this paper presents an automatic Trimap generation algorithm based on edge growth without human interaction - BGTrimap. The overall flow of the algorithm is illustrated in Fig. 4.

Based on the pre-segmentation results, BGTrimap first uses the optimized expansion-corrosion method to identify the edge region. Then, it automatically derives seed points from this region. Initially, Manhattan distance is used to sparsify the set of edge seed points, which facilitates more precise region growth and eliminates redundant background information. To mitigate the impact of low image resolution on background information, the Sobel operator is applied to detect edge details. The gradient information is then transformed into mixed- pixel regions through NML denoising binarization. Finally, in a process aided by Otsu segmentation and the pre-segmentation results, the Trimap is generated without human intervention.

The first step of the BGTrimap algorithm is the acquisition of edge regions. Most of the blended pixel regions in images are concentrated at the junction of the foreground and the background. Therefore, BGTrimap focuses on this region, which effectively enhances the algorithm's effectiveness and speed. The edge region is often expanded using the erosion algorithm while

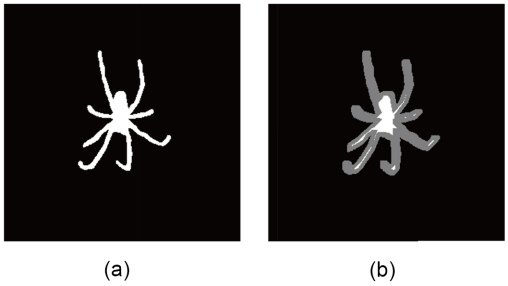

Fig. 5. Expansion corrosion of low-resolution images: (a) original figure and (b) trimap.

adjusting the kernel size. For low-resolution images, even if the kernel size of the erosion operation is minimized to 2×2, there will still be a loss of foreground information. For example, consider the spider's legs; the target area is only 2 to 3 pixels wide, as depicted in Fig. 5, where the white area represents the result of erosion and the grey area signifies the difference between the expansion area and the erosion area. With a kernel size of 2×2, there is a significant loss in the foreground features of the spider's leg region, which causes the entire leg to be inaccurately labeled as an edge region. This reduction in accuracy compromises the acquisition of mixed pixel regions in subsequent processing.

To address this issue, this paper first upsamples the original image to an appropriate resolution and then applies an optimized expansion and erosion algorithm to delineate the edge region. Initially, the faster bilinear interpolation method is used to upsample the binary map. Then, the expansion and erosion operations are conducted. Fig. 6 depicts the effect of this expansion using bilinear interpolation. The details of the process are as follows: through bilinear interpolation, Fig. 6(a) is upsampled to Fig. 6(b), after which Fig. 6(c) is obtained by mean downsampling Fig. 6(b) to restore the results, which

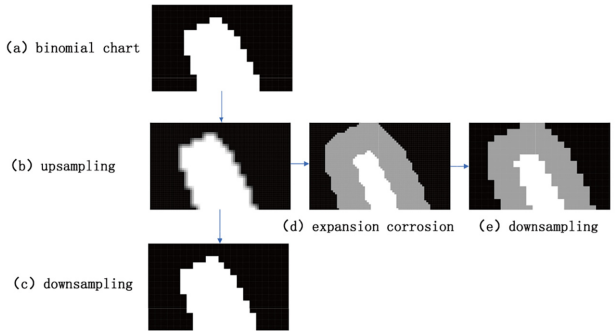

Fig. 6. Bilinear interpolation method for obtaining edge region.

ultimately reveals no discernible difference from a binary map (Fig. 6(a)). This shows that there is no loss of grayscale information in the upsampling and downsampling processes. Consequently, the expansion and erosion algorithm can be applied to Fig. 6(b) to derive a high-quality edge expansion region.

However, with traditional expansion-corrosion methods, it is necessary to customize the kernel size based on different images or by fixing semantics, such as by uniformly using smaller kernels for expansion-corrosion in scenarios like cars to acquire edge regions. Nevertheless, for low-resolution complex edge targets, where the target's proportion in the image and the complexity of the edge contour vary, it is not feasible to directly acquire the edge region through simple expansion-corrosion operations. Therefore, this paper proposes an improved expansion and erosion method based on an adaptive kernel. This method automatically adjusts the size of the expansion and erosion kernels by detecting the proportion of foreground pixels as a whole and determining whether the image data belongs to a small target image. Once the kernel size has been determined, the traditional expansion and erosion operations are performed, after which the resolution is restored through downsampling using the mean value method. The formula used to determine the expansion-corrosion kernel size is as follows:

$$I_{kernel} = \left\lceil \left| \frac{1000}{max(W,H)} \right| \left\lceil \frac{F^{\frac{3}{2}}}{\beta X} \right\rceil \right\rceil, \tag{2}$$

where I_{kernel} is the dimension of the kernel, $max(W,H)$ is the number of pixels on the longest side of the pre-segmentation result, F is the number of foreground pixels in the pre-segmented image, and X is the number of pixels in the overall image. Parameter β controls the expansion ratio of the expansion corrosion. Through experimental comparison studies, β is obtained as 5, which indicates a 1/5 expansion.

The optimized expansion corrosion algorithm is based on a coarse segmentation mask to obtain the edge grayscale map, and linear operations are executed on the original image to extract the edge region in the RGB channels. Since most of the mixed pixel regions in the image correspond to the edge region, the goal of obtaining the edge region is to reduce the subsequent computation time of different algorithms while also minimizing interference from redundant foreground and background information. The calculation formula used in this process is as follows:

$$GB = \sum_{i,j \in D} I(i,j) - \sum_{i,j \in E} I(i,j),\ I \in A, \tag{3}$$

where I represents a single pixel point, A is an RGB image, and i, j denote the position of the pixel point. $I(i, j)$ represents the pixel value of the pixel point located at column j and row i in image I. $i, j \in D$ denotes the set of pixels in the dilated region, while $i, j \in E$ denotes the set of pixels in the eroded area. This operation is performed across all channels of a three-channel RGB image.

After obtaining the edge region GB in the initial step, the second step is to remove redundant background information in the GB image; we use the seeded region growing (SRG) algorithm for this purpose. The SRG algorithm is beneficial because it uses the expanded and eroded edge region as the growth range and selects outer edges as seed points. This approach effectively reduces the computational burden of rejecting background information while also enhancing accuracy. The key aspects of the SRG algorithm are seed point selection (Seeds) and the choice of the growth threshold. Initially, BGTrimap expands and erodes the boundary for pre-processing, while also collecting seed points S_d. However, the seed points obtained through this method are overly dense, which leads to poor background rejection when using a low threshold during growth. In this paper, we use Manhattan distance to sparsify the seed point set S_d, which is defined as follows:

$$d(I^1, I^2) = \left|I_i^1 - I_j^2\right| + \left|I_i^1 - I_j^2\right|,\ I^1, I^2 \in S_d, \tag{4}$$

$$S_m = \sum_{d(I^1, I^2)=3} I(i,j),\ I \in S_d, \tag{5}$$

where $d(I^1, I^2)$ denotes the Manhattan distance between the pixel points of I^1 and I^2, and the set of seed points after thinning is denoted as S_m. The aforementioned method makes it possible to automatically obtain seed points without requiring interaction.

Region growing is performed using the set of seed points S_m. During region growing, pixels are expanded at each seed point, and those that are smaller than the threshold are added to the set. Subsequently, the sets corresponding to all seed points are merged. Due to the small threshold value and the large number of seed

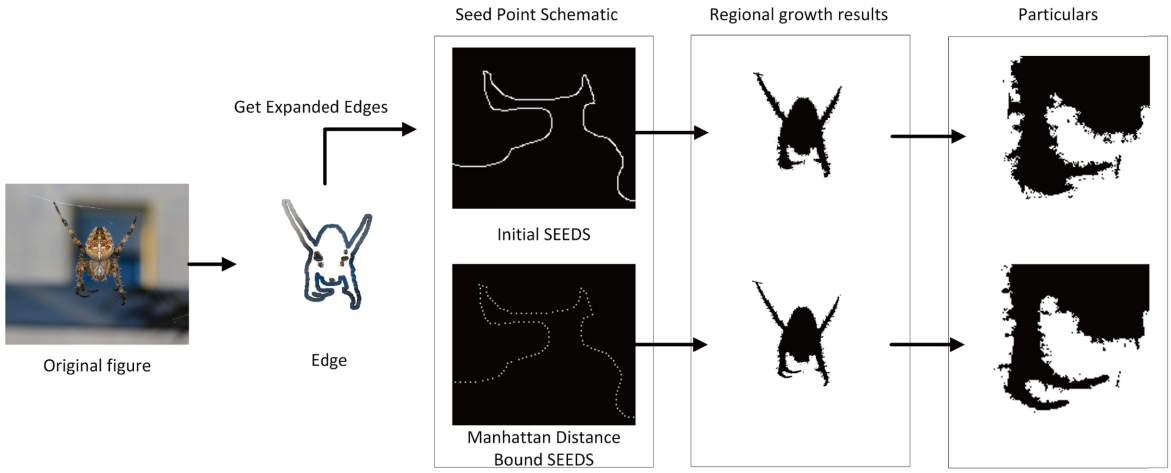

Fig. 7. Comparison of different seed points.

points, the segmented image can better retain the foreground and mixed pixel regions while eliminating background redundant information, all without requiring interaction.

Fig. 7 shows the comparison of seed points before and after sparsification. It can be observed from the detailed areas that the seed points that were obtained directly by expanding the edges for region growth contain more redundant information in the foreground and mixed pixel regions. For example, in Fig. 7, the background between the forelimbs of the spider is misclassified as a foreground and mixed pixel region, which is not conducive to subsequent classification of the mixed region. Sparsification using the Manhattan distance removes redundant pixel information in the background region while accurately retaining foreground and mixed pixel regions.

Building upon the aforementioned steps, edge detection is then used to enhance the details within GB. The mixed pixel region typically encompasses numerous complex edge textures, which can be refined through edge detection methods to extract more detailed information based on edge gradients. Subsequently, the edge details are converted into regions to extract mixed pixel information. The edge detection algorithm employed in this paper utilizes the Sobel operator, which yields coarser edges, thereby enhancing edge detail consistency while also maintaining faster processing speeds.

The Sobel operator detects edges in the image by computing the grayscale values. For each pixel, both horizontal and vertical convolution operators are applied to perform the convolution operation. The formulations for these two convolution operators are as follows:

$$G_x = \begin{bmatrix} -1 & 0 & +1 \\ -2 & 0 & +2 \\ -1 & 0 & +1 \end{bmatrix} \times A, \tag{6}$$

$$G_y = \begin{bmatrix} +1 & +2 & +1 \\ 0 & 0 & 0 \\ -1 & -2 & -1 \end{bmatrix} \times A, \tag{7}$$

where A represents the original image, and G_x and G_y respectively represent the horizontal and vertical gradient images. The output is constrained as a grayscale image with 16-bit symbolic shaping. Subsequently, gradients in both directions are combined, after which the 16-bit image is converted to an 8-bit image to obtain the overall gradient image G. The formula used for gradient combination is as follows:

$$G = \sqrt{G_x^2 + G_y^2}. \tag{8}$$

The obtained edge gradient map contains numerous details, which primarily consist of point and line information. To obtain specific mixed pixel regions, these details need to be segmented into regions. To address this issue, BGTrimap initially employs the non-local average denoising (non-local means [NLM]) algorithm to denoise the image. The NLM algorithm has been shown to be effective in noise reduction while preserving texture details, ultimately resulting in superior outcomes. Minimizing the loss of details during denoising operations can significantly enhance the accuracy of subsequent image binarization.

The purpose of binarization is to distinctly differentiate edge regions from non-edge regions based on the NML denoising results. In this study, it was found that a threshold of 50 for binarization effectively eliminates redundant background while retaining more details. The specific formula that was used is as follows:

$$Z(i,j) = \begin{cases} 255, & G(i,j) > 50 \\ 0, & G(i,j) \le 50 \end{cases}. \tag{9}$$

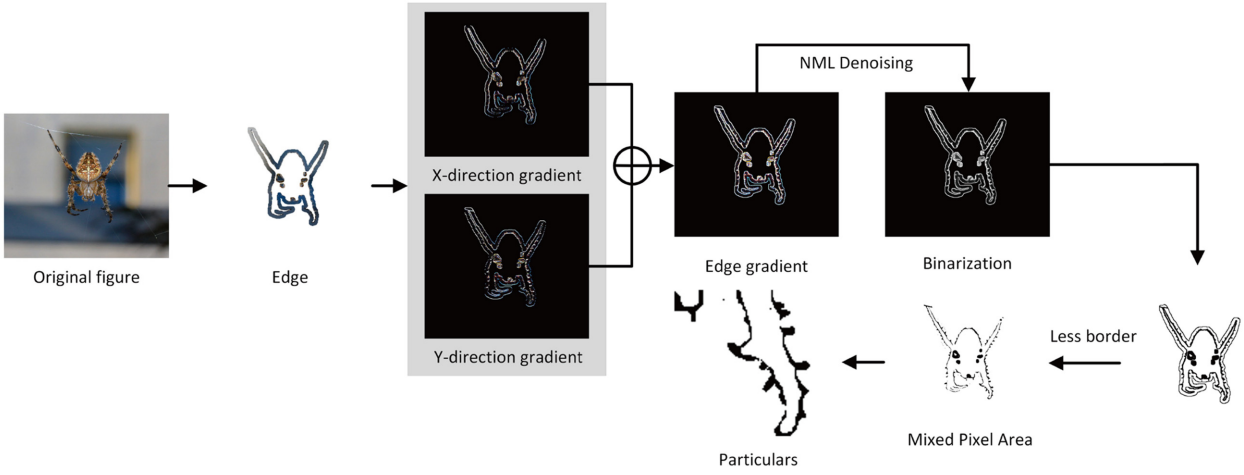

Fig. 8. Mixed pixel area acquisition.

Experimental comparative studies showed that better results are achieved by defining the threshold value as 50. If the grayscale value of pixel $G(i, j)$ is greater than 50, threshold value will be set to 255; if it is smaller than or equal to 50, threshold value will be set to 0. The resulting image, Z, still contains interference from the boundary between the foreground of the GB image and the transparent region, which needs to be subtracted. The specific process used to obtain the mixed pixel region is shown in Fig. 8.

Performing Sobel edge detection on the edge region yields its edge gradient. As can be seen in the detailed region, the gradient map contains abundant information about edge details, which can be used as preprocessing features for the mixed pixel region of Trimap. However, the gradient map also includes numerous scattered points, which can primarily be attributed to the fact that the background's redundant information was not removed. By employing the NML algorithm, the entire gradient map is treated as a denoising region. This denoising operation effectively eliminates redundant scattered points. Subsequently, the gradient map is binarized. By treating the entire gradient map as a grayscale image, along with discrete points, it can be converted into regions consisting of points, lines, and other areas, ultimately resulting in a cleaner and more accurate mixed pixel region.

Afterwards, BGTrimap also uses the maximum interclass variance method (Otsu) to refine the aforementioned results and generate Trimap. The aim is to confine the foreground region in Trimap, thereby preventing the mixed pixel region from including parts of the foreground. Otsu belongs to the adaptive thresholding segmentation algorithm, which operates based on the grayscale characteristics of the image.

Assuming the existence of a threshold, TH, that divides all pixels of an image into the following two categories: regions smaller than the threshold D_1 and regions larger than the threshold D_2. The mean grey value of each region is m_1 and m_2, respectively, and the probabilities that a pixel will be classified into the two regions are p_1 and p_2. The threshold selection formula for Otsu is as follows:

$$\sigma^2 = p_1 p_2 (m_1 - m_2)^2 . \tag{10}$$

The threshold value that maximizes σ^2 in the above equation serves as the Otsu threshold. The image can then be segmented using this threshold value.

Through the aforementioned processes, we obtain the following images: M_S, which represents the background removal of the GB image using the region growing method; M_B, which denotes the edge detection after blending pixel regions; and M_O, which is obtained through Otsu algorithm threshold segmentation. We then combine M_S, M_B, and M_O through pre-segmentation constraints to generate Trimap. In the M_O and pre-segmentation results, M is found to constrain the generation of Trimap. The black regions in M_S and M_O represent the foreground and part of the background, while the black region in M_B indicates the edges of the blended pixel region. The specific formulas are defined as follows:

$$T(i, j) = 0 \begin{cases} M_B(i, j) = 255,\ M(i, j) = 0 \\ M_S(i, j) = 255,\ M_O(i, j) = 0,\ M(i, j) = 0 \\ GMB(i, j) = 0 \end{cases}, \tag{11}$$

$$T(i, j) = 255 \begin{cases} M_S(i, j) = 0,\ M_O(i, j) = 255,\ M(i, j) = 255 \\ GMB(i, j) = 255 \end{cases}, \tag{12}$$

$$T(i, j) = 128,\ M_B(i, j) = 0,\ GBM(i, j) = 255 , \tag{13}$$

where GBM is the grayscale label map of GB, where

white represents the foreground, black represents the background, and grey represents the boundary region. The results obtained through the aforementioned linear operations make it possible for us to accurately classify foreground, background, and mixed pixels to generate the Trimap map.

In summary, the algorithm steps of BGTrimap outlined in this paper are as follows:

Input: RGB image I, Coarse segmentation result M.
Output: Trimap diagram T.
Step 1: Read image M and utilize the optimized swell-corrosion algorithm to derive the swollen region D and corroded region E.
Step 2: Obtain the edge pixel region GB from D and E using Eq. (4).
Step 3: Generate the initial seed point set S_d from the edge of region D, sparsify S_d using the Manhattan distance (Eq. 5), and obtain the final seed point set S_m.
Step 4: Grow the region of GB based on the fixed threshold seed point set S_m to acquire M_S.
Step 5: Conduct Sobel operator edge detection on region GB to produce the gradient map G.
Step 6: Apply NML denoising on the gradient map G, then use binarization to derive the edge-mixed pixel region M_B.
Step 7: Perform Otsu threshold segmentation on GB to obtain M_O.
Step 8: Utilize the linear operations described in Eqs. (11)-(13) on M_S, M_B, and M_O to derive the final Trimap diagram T.

III. EXPERIMENTAL RESULTS AND ANALYSES

The TGMatting automatic image matting method proposed in this paper comprises the U^2S-Net pre-segmentation module, the BGTrimap automatic Trimap generation module, and the final matting module. Therefore, the effectiveness, advantages, and disadvantages of this method are verified by conducting comparative experiments for U^2S-Net, BGTrimap, and the final matting module.

A. Data Sources and Pre-processing

To evaluate the performance of TGMatting on low-resolution images with complex and blurred foreground edges, spiders were selected as experimental subjects. Obtaining a large number of high-quality live images of spider species is challenging, as most existing images suffer from issues such as low resolution, complex foreground edges, and blurring. Spiders also exhibit characteristics such as body hairs, long and thin limbs, and complex morphology, which makes them suitable for verifying the method. In this study, we constructed an image dataset of spider species through web collection, specimen shooting, and field image collection. The spider dataset used in this paper was created by selecting spider images from two public datasets, ImageNet [30] and Animal-10 on Kaggle, and combining them with images captured in the field along with specimen images to form the Spider-1100 dataset, ultimately consisting of 1,100 original images. The distribution table of spider families is presented in Table 1.

Table 1. Spider-1100 dataset distribution

Spider categories	Number of images
Thomisidae	200
Theraphosidae	100
Missulena	100
Araneidae	250
Salticidae	100
Sparassidae	50
Lycosidae	100
Other	200

To validate the pre-segmentation and matting results, this paper constructs the SOD-Spider coarse segmentation dataset and the Alpha-Spider fine segmentation test set based on the Spider-1100 dataset; these datasets are used to evaluate the performance of the matting method.

1) SOD-Spider Dataset

In the field of image segmentation, there is a shortage of datasets that are exclusively dedicated to spider species, making it challenging to train high-quality models using pre-segmentation methods. SOD-Spider is one of the few segmentation datasets that has been specifically designed for spiders: It comprises 1,000 training images and 100 test images, where the foreground and background of the pre-processed images are manually classified using Photoshop. The foreground is labeled as white, while the background is labeled as black, resulting in the generation of the ground-truth (GT) map. Several images from the SOD-Spider dataset are depicted in Fig. 9.

Due to the small size of the spider subjects, in most scenes depicted in spider images, they constitute a small proportion of the overall image, often resulting in lower resolution. In the training set, a significant portion of the SOD-Spider dataset consists of images with a unilateral resolution of 300 to 400 pixels, which aligns with the resolution distribution that is found in many current datasets [31]. The SOD-Spider test dataset, Spider-test, also includes low-resolution images with complex and blurred foreground edges to evaluate the model's segmentation capability on such images [32]. It encompasses spiders from at least 28 families and 40 genera, thus

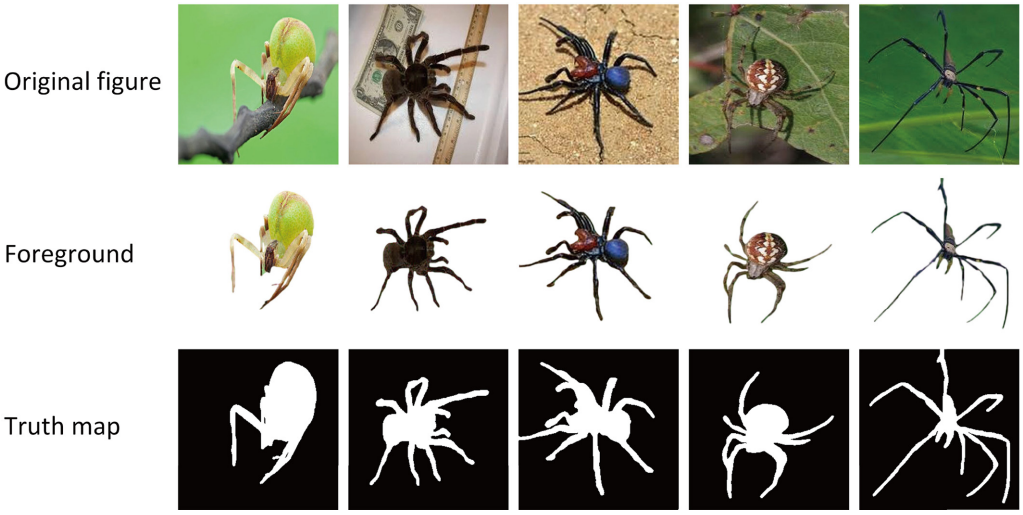

Fig. 9. Selected images of the SOD-Spider dataset.

providing better coverage of spiders with various morphological features. This helps address issues related to the model's limited generalization ability, and ultimately enhances its robustness.

2) Alpha-Spider Dataset

In this paper, 60 spider images with various features from the Spider-1100 dataset are selected to construct the Alpha-Spider spider image matting test dataset. This dataset is used to evaluate the performance of the matting method. Image processing professionals use transparency masks to finely segment the spider's hairs and other details based on the original images. Several images from the Alpha-Spider dataset are presented in Fig. 10. The first row shows the original images, while the second row displays the processed transparency masks. The dataset includes images that have been captured at different distances and features various types of edge details.

3) Alphamatting Dataset

Most current research has assessed the performance of image matting methods using the Alphamatting dataset [33], which is test data that is publicly available on the Image Matting Evaluation website. The Alphamatting dataset comprises around 30 images with mixed pixel regions exhibiting various characteristics, such as portraits, hair, and dolls, making it suitable for method evaluation. However, due to its limited size, the dataset is not adequate for training purposes. Since there is no suitable criterion for directly evaluating Trimap generation, many methods opt to first generate Trimap using the Alphamatting dataset and then derive transparency masks using a fixed image matting method. The quality of Trimap generation is then assessed based on the quality of the resulting transparency masks.

In this paper, the BGTrimap algorithm is implemented as the Trimap generation module of TGMatting. Experiments are tested on the Alphamatting public dataset to verify the Trimap generation effectiveness of the BGTrimap algorithm. Fig. 11 shows some images from the Alphamatting dataset.

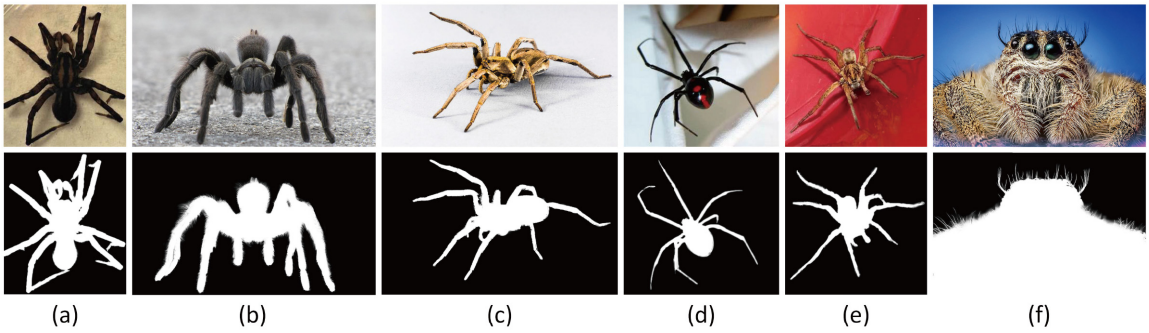

Fig. 10. (a-f) Alpha Spider dataset images.

Fig. 11. Partial images from the Alphamatting dataset.

B. U²S-Net Pre-segmentation Module Results and Analysis

The purpose of pre-segmentation is to accurately classify foreground and background regions. Pre-segmentation is achieved in this paper using the salient target detection method, which typically involves a salient target detection dataset containing original images as well as corresponding salient region labels. This dataset can be used for both model training and testing. The method described in this paper is trained and tested using the public dataset DUTS and the SOD-Spider spider dataset that has been constructed in this study.

1) Evaluation Criteria

Commonly used evaluation metrics for salient target detection include mean absolute error (MAE) [34], precision-recall (P-R) curves, and F-measure [35].

The MAE directly calculates the mean absolute error between the saliency map and the ground truth map of the model output, which is calculated using the following formula:

$$MAE = \frac{1}{W \times H}\sum_{x=1}^{W}\sum_{y=1}^{H}\left|S(x, y) - G(x, y)\right|, \tag{14}$$

where W is the width of the image, H is the height of the image, $S(x, y)$ denotes the significance result, and $G(x, y)$ denotes the ground truth value.

The P-R curve refers to the probability that the predicted value of the model is in the GT, while the R rate refers to the probability that the significant region in GT will still be significant in the model prediction. It involves comparing the model (M) and GT pixel by pixel to calculate the precision and recall values. The specific formula used for this process is as follows:

$$Precision = \frac{|M \cap G|}{M}$$

$$Recall = \frac{M \cap G}{G}. \tag{15}$$

Sometimes, neither the precision rate nor the recall rate can fully capture the performance of the model; hence it was proposed that the F-measure be used. The F-measure represents the weighted harmonic mean (WHM) of precision and recall, with non-negative weights denoted by β. The F-measure is calculated using the following formula:

$$F_\beta = \frac{(1+\beta^2)Precision \times Recall}{\beta^2 Precision \times Recall}. \tag{16}$$

A value of 0.3 is commonly used for β^2, implying that more emphasis is placed on precision than recall. This suggests that accuracy is considered to be more important than completeness. This choice is made because, when the model labels all pixels as significant regions, the precision rate will be equal to 100% whereas the recall rate will be very low.

2) Analysis of Coarse Segmentation Results of U²S-Net Module

In the experiments, the improved U^2S-Net is compared with U^2S-Net and U^2-Netp [20]. Since the parameter count of the improved method in this paper is 82.62M, which is only half of that of U^2-Net, another method with a smaller parameter count—U^2-Netp—is chosen from the U^2-Net paper for simultaneous comparison and validation. The U^2-Net method, the improved U^2S-Net, and U^2-Netp are all trained using the DUTS-TR training set, and they are finally tested on the DUTS-TE test set with each evaluation index calculated. Moreover, to compare the effect of the improved U^2S-Net method and U^2-Net on low-resolution small target datasets, we compare the performance of these two methods on the SOD-Spider dataset in this paper. The values of MAE and F metrics are listed in Table 2. Further, Fig. 12 depicts the P-R curves and F metric curves of the methods on both the DUTS-TE and Spider-test test sets.

The results indicate that, on the DUTS-TE test set, the evaluation metrics of U^2S-Net are slightly lower than those of U^2-Net. This is primarily attributed to the diverse range of saliency targets that is present in DUTS-TE. Moreover, the U^2S-Net model has a lower parameter

Table 2. SOD results of the models

Model	Training set	Test set	MAE	Max F_β	Mean F_β
U^2-Net [19]	DUTS-TR	DUTS-TE	0.044	0.873	0.848
	SOD-Spider	Spider-test	0.034	0.873	0.843
U^2-Netp [19]	DUTS-TR	DUTS-TE	0.054	0.852	0.763
U^2S-Net	DUTS-TR	DUTS-TE	0.048	0.855	0.828
	SOD-Spider	Spider-test	0.031	0.899	0.862

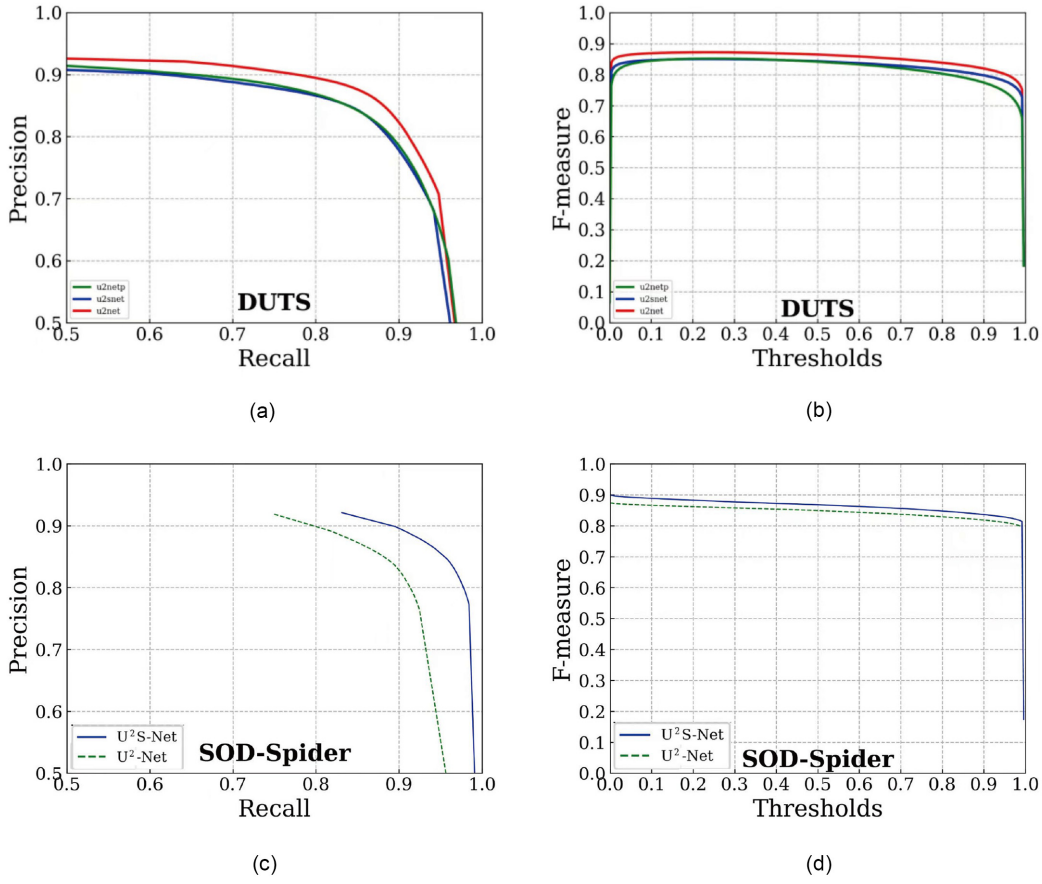

Fig. 12. Comparison of P-R curve and F measurement curve: (a) precision, (b) F-measure in DUTS, and (c) precision and (d) F-measure in SOD-Spider.

count of 82.6M than U^2-Net's parameter count of 176.8M. This makes it challenging for the model with fewer parameters to effectively learn complex saliency regions. However, the improved U^2S-Net outperforms the model with fewer parameters, U^2-Netp, across all evaluation metrics. Moreover, in terms of processing time, the average processing time of U^2S-Net per image is approximately 10% faster than that of U^2-Net. This suggests that the improved U^2S-Net achieves similar efficiency and effectiveness compared to the original method. As a result of the comparison using the SOD-Spider dataset, the MAE, max F_β, and mean F_β values of U^2S-Net are still found to be better than those of the U^2-Net method, despite having fewer parameters. Specifically, the MAE is reduced by 0.003, while max F_β and mean F_β are both improved by approximately 0.02. The Spider-test dataset includes images with low resolution, small targets, and complex and blurred edges, so the results demonstrate the advantage of the U^2S-Net method on such images.

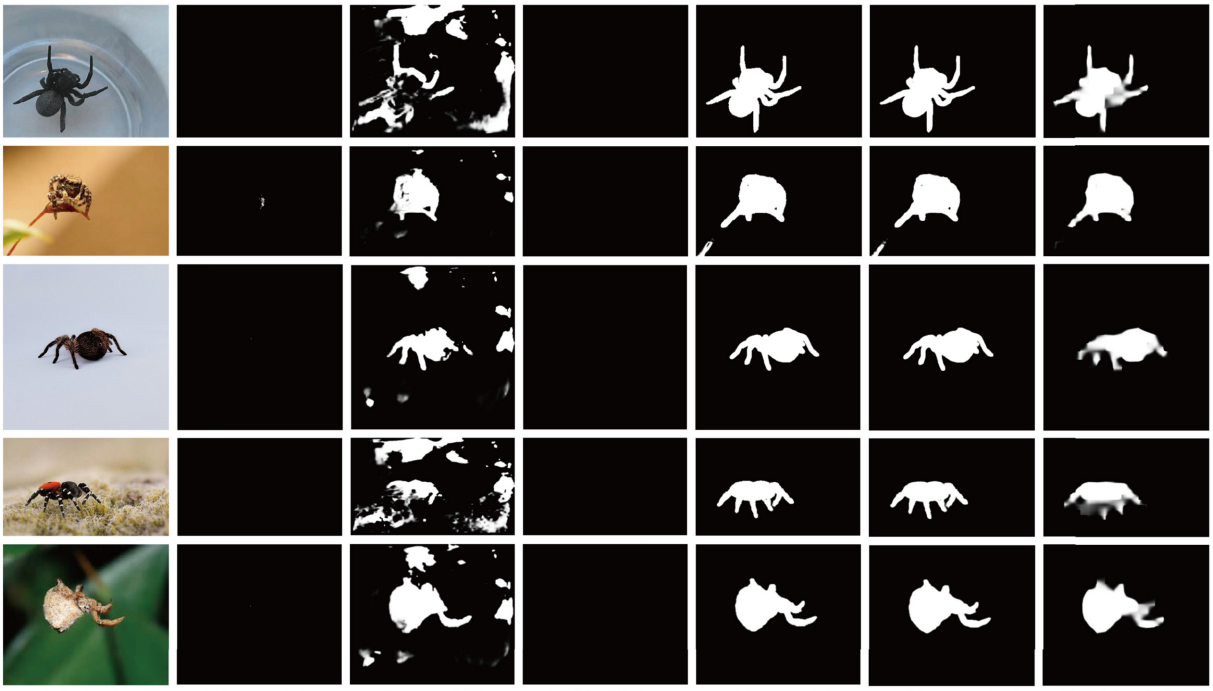

Fig. 13. SOD results in low-resolution spider images.

In the Spider-test test dataset, specific comparisons between U^2S-Net and U^2-Net in low-resolution images are shown in Fig. 13.

From left to right, Fig. 13 shows the original image, the outputs of U^2-Net's high-level, middle-level, and bottom-level layers, and the outputs of U^2S-Net's high-level, middle-level, and bottom-level layers, respectively. The resolutions of these five test images are all below 200 pixels, while the proportion of spider pixels in the images is less than 20%. In the low-resolution images, U^2-Net's predictions for saliency values of the foreground and background are unclear. The bottom layer predicts almost all pixels as non-significant regions, while the middle layer fails to distinguish specific significant and non-significant regions. Moreover, the top layer predicts that most significant regions are non-significant. Due to the reduction in model depth, U^2S-Net performs better in predicting saliency for images with different resolutions. Its outputs are improved across all levels, and especially improved for low-resolution and small target images. This improvement is particularly crucial for saliency target detection in low-resolution images with complex and blurred foreground edges, such as those of spider species.

The results of the quantitative comparison presented above indicate that the improved U^2S-Net achieves better results. This improvement primarily stems from the fact that the downsampling and joint loss computation of U^2-Net are less effective for very low-resolution images. By reducing the burden on the model's bottom layer, U^2S-Net alleviates excessive downsampling, which leads to significant improvements in performance for low-resolution images. Although the downsampling method aims to increase the receptive field of the model while reducing memory and computation requirements, it requires the use of multiple multi-scale features to achieve optimal results. While U^2-Net combines encoding and decoding layers to fuse results from the same scale RSU modules to leverage multi-scale information, excessive downsampling in low-resolution images results in the loss of crucial features in the model's bottom layers, which leads to feature confusion. Although increasing the receptive field can to some extent enhance multi-category segmentation capabilities, excessive downsampling is unnecessary for single-category segmentation tasks. The U^2S-Net method proposed in this paper reduces the model's reliance on bottom-layer features by replacing the underlying RSU model, thereby improving computational speed while maintaining effectiveness.

C. BGTrimap Module Trimap Generation Results and Analysis

1) Evaluation Criteria

The common evaluation metric for image matting methods is the sum of absolute differences (SAD) [31].

SAD is a measure that is frequently used in regression analysis, where it is primarily employed to assess the extent of data changes. A smaller value indicates better performance. The specific formula is as follows:

$$SAD(x, y) = \sum_{x=1}^{W}\sum_{y=1}^{H}\left|mask(x, y) - gt(x, y)\right|, \qquad (17)$$

where H is the height of the image, W is the width of the image, $mask(x, y)$ is the mask generated by the algorithm, and $gt(x, y)$ is the ground truth transparency mask. SAD is the most commonly used and intuitive evaluation metric in image matting methods. It is frequently employed in Trimap generation tasks to assess the quality of the transparency mask that has been obtained [1].

2) BGTrimap Generation Results

To begin with, the Trimap generation efficacy of BGTrimap is assessed on the images from the Alpha-matting dataset to subjectively analyze the quality of the generated results. The intention of doing this is to observe and compare the Trimap generation performance of the method across images with varied characteristics. Some of the generated results for the Alphamatting dataset are depicted in Fig. 14. The first column represents the original image, and the second column shows the SOD model generated by U^2S-Net trained on the public dataset DUTS-TR. Meanwhile, the third column displays the manually produced Trimap, and the fourth column exhibits the Trimap generated by BGTrimap. The findings show that the method proposed in this paper performs exceptionally well on this dataset, as it demonstrates results comparable to those obtained by the manually produced Trimap. The Trimap generated by BGTrimap effectively distinguishes hard edges of mixed pixels and hairy regions, as evidenced by the intricate details. Notably, subtle foreground regions such as buttons and zippers are accurately classified, thereby showcasing the method's superior performance over manually produced Trimaps.

The results that have been generated for the Alpha-Spider dataset are presented in Fig. 15. The second column, labeled SOD, displays the results of saliency target detection generated by U^2S-Net trained on SOD-Spider. Spider species present significant challenges in Trimap generation due to their complex morphological features, including small limbs and spinners. The dataset also features low resolution and complex fuzzy edges. The results obtained from BGTrimap demonstrate effective detection of edge-mixed pixels in spider images. For instance, the results in Fig. 15 demonstrate improved classification accuracy on the hair edge of the tarantula in (1) and the hard edge of the black widow in (2). The algorithm minimizes uncertainty while ensuring precise delineation of foreground and background regions. In images (3) and (4), the hairs and spines of the spider are classified with high accuracy, even in tiny soft-edge regions.

Fig. 14. Alphamatting generated results.

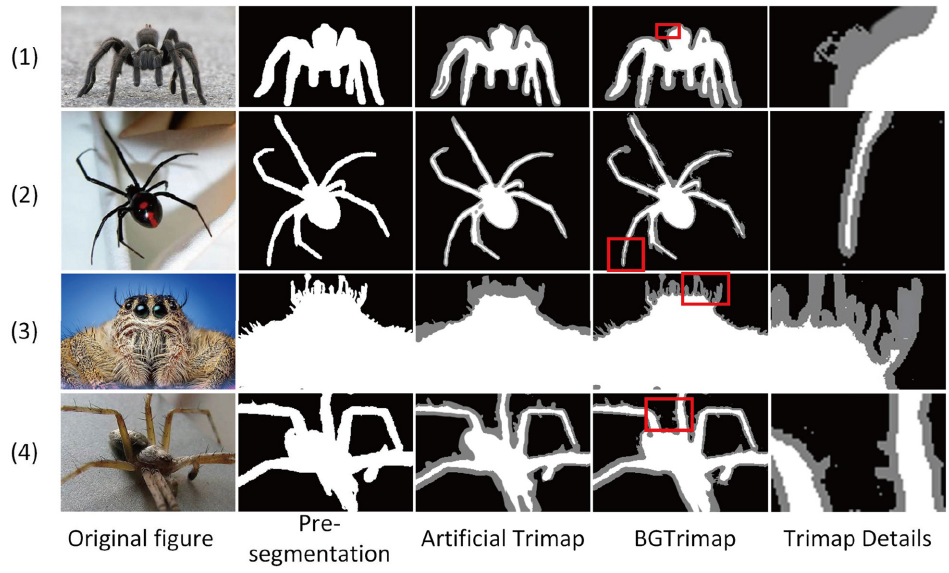

Fig. 15. Generation results of spider images.

3) Quantitative Comparison

The Alphamatting dataset is currently considered to be the most authoritative dataset for image matting testing. This paper conducts a quantitative comparison with existing methods on Alphamatting to evaluate the effectiveness of the BGTrim algorithm proposed herein. To provide a comprehensive assessment, this study compares BGTrim with several typical Trimap generation algorithms that do not require interaction. These algorithms include the fusion-based approach proposed by Li et al. [36], the clustering-based approach described by Henry and Lee [13], the edge-based approach introduced by Shahrian et al. [37], the patch-based approach detailed by Aksoy et al. [38], and the expansion-based approach used by Gupta and Raman [14].

Since the quality of the Trimap affects the performance of image matting methods, we first assess the different Trimaps generated by different methods. Two traditional image matting methods, KNN Matting [39] and Information Flow Matting (IFMatting) [38], are selected for this comparison. Transparency masks are generated from the results of each Trimap generated using these two matting methods, and the SAD values between the transparency masks and the ground truth maps are calculated. The selection of traditional matting methods aims to mitigate dataset interference on model performance in deep learning, thereby improving model robustness and reducing bias in performance across different data types. This approach ensures that more informative results are obtained. Specific results are presented in Table 3.

The SAD comparison of the IFMatting method is presented in the first section of Table 3. It can be observed that the BGTrimap method proposed in this paper achieves the lowest SAD values in GT02, GT17, and GT22, indicating superior performance on images with relatively few edge-mixed regions. This can be attributed to the effectiveness of the region growing algorithm, particularly in handling images with significant color variations, as it performs better in removing redundancy from the background. Conversely, the clustering Trimap generation method yields the best results in GT05, GT09, and GT18, which is attributed to its ability to accurately classify large opaque regions. The pruning Trimap generation method achieves better results in GT06. Conversely, the expansion-based Trimap generation method performs poorly, resulting in higher SAD values. Notably, the average SAD value of BGTrimap is 6,923, which is lower than that of the clustering-based method (8,316) and all other Trimap generation methods. These results indicate that, among the transparency masks lost in the final IFMatting methods, BGTrimap aligns closely with the true value of the map, which demonstrates its superior effectiveness.

The SAD results for the KNN Matting method are presented in the second section of Table 3. Notably, this method achieves the lowest SAD values in GT02, GT05, GT17, and GT22, which indicates its effectiveness, particularly in images with fewer edge-blending regions. Conversely, the clustering-based Trimap generation method yields the best results in GT06, GT09, and GT18, which is consistent with the observations from the IFMatting method. This suggests that the performance of KNN Matting can be significantly enhanced in images with fewer regions of edge blending. The average SAD value for KNN Matting is 7,566, which is also lower than the average SAD values of the other methods.

The experimental comparison results demonstrate that, overall, the BGTrimap method developed in this paper

Table 3. SAD values for different trimap generation methods

Model	Test pictures	BGTrimap	Co-fusion-based method [36]	Clustering-based method [13]	Patch-based method [38]	Edge-based method [37]	Dilation-based method [14]
IFMatting [36]	GT02	7526	3550	7259	7291	6702	5544
	GT05	3594	6088	2619	4015	2727	6650
	GT06	6126	20296	3421	2996	3044	5112
	GT09	10982	14393	9483	15112	10737	11231
	GT17	7330	13492	20349	20309	22954	23563
	GT18	6800	4573	4582	4624	8850	11478
	GT22	9147	7040	10499	11602	13714	15685
	Average	7358	9918	8316	9421	9818	11323
KNN Matting [39]	GT02	6573	5247	7316	9778	7776	11592
	GT05	1522	6244	2688	4056	2955	4923
	GT06	4698	10005	3630	4071	3802	4477
	GT09	12813	16091	10094	14832	12635	13440
	GT17	15268	19666	19549	18824	22909	19053
	GT18	5335	8622	4674	4974	6310	6423
	GT22	6736	10272	10592	12530	14471	13009
	Average	7566	10878	8363	9866	10122	10416
Time (s)	Average	3.09	4.21	5.11	14.85	4.34	8.19

achieves the lowest loss in Trimap generation, as it ultimately accurately generates Trimap maps even for images with mixed pixel regions concentrated at the edges.

As can be seen in Table 3, from the input image to the output Trimap, the average Trimap generation time for the test images above is only about 3.09 seconds, while many current methods take up to 10 seconds or even a minute [40]. This reduction in running time can significantly enhance the efficiency of automatic matting.

D. Image Matting Module Results and Analysis

Although traditional image matting methods may be effective for validating Trimap generation methods, they primarily rely on low-level features related to image color or structure. For instance, the propagation-based matting method, KNN Matting, by Chen et al. [39] and the information flow-based IFMatting by Aksoy et al. [38] both have processing times that are too long for them to be used in practice as automatic matting modules.

However, the introduction of deep neural networks has largely addressed this issue. FBAMatting by Forte and Pitie [41] was proposed to predict foreground and background maps simultaneously by calculating their respective losses. Similarly, Liu et al. [42] proposed TIMI Net, which is a collaborative matting method that integrates global and local information. This approach reduces the model's focus on the uncertain areas of the Trimap by extracting features from the original image and the Trimap in separate processes using different modules.

1) Comparison Experiment of Different Image Matting Methods

In the matting module, this paper compares four image matting methods—KNN Matting, IFMatting, TIMI Net, and FBAMatting—based on the BGTrimap method using the Alpha-Spider Spider matting test set. A visual comparison is presented in Fig. 16.

From the six spider images in Fig. 16, it is evident that the TIMI Net method exhibits more loss in the detail region of the spider. In image (b), the matting effect on the hair detail region of the tarantula is poor. KNN Matting seems to achieve better hair detail effects but it performs poorly in shadow classification. Although the IFMatting method excels in hair segmentation, it incurs more loss in the fine hard edge region. Overall, FBAMatting performs better in the six spider images, which is primarily attributed to its joint loss approach, which enhances its ability to segment mixed regions in low-resolution images.

The quantitative comparisons are presented in Table 4, where the method with the lowest average SAD value is FBAMatting; this finding is consistent with the previous visual comparison evaluation. The best-performing method in spider (a) is TIMI Net, with a SAD value of 745, which is mainly attributed to the fact that this method can

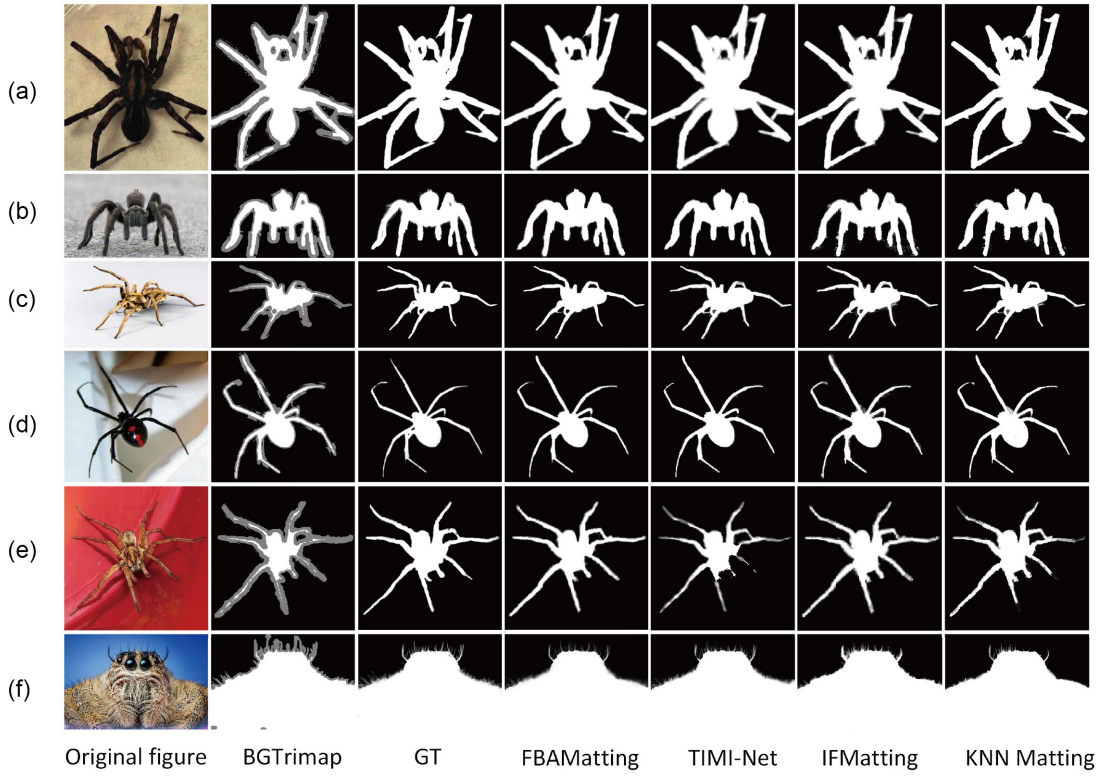

Fig. 16. Results of different image matting methods.

Table 4. Different image matting methods

Test set	Test pictures	FBA-Matting [41]	TIMI Net [42]	IFMatting [38]	KNN Matting [39]
Alpha-Spider	a	908	745	892	1163
	b	1197	1427	1153	1138
	c	2644	3951	3209	3395
	d	1433	3444	2080	1482
	e	222	359	399	502
	f	1326	1397	1615	1451
	Average	1288	1887	1558	1521

better segment the edge information of the spider overlapping with the shadows. The best-performing method in spider (b) is KNN Matting, while the lowest SAD value in all other images is FBAMatting. The SAD value of the TIMI Net method in this dataset is the highest at 1,887, indicating that TIMI Net performs relatively poorly in low-resolution images, which is attributed to its global and local information mining method. On the other hand, FBAMatting, due to its foreground and background joint loss, excels in extracting detailed features. It performs best in low-resolution small target images and also provides better segmentation results for spider images with more hairs at the edges.

2) Comparison of End-to-End Natural Image Matting Methods

Based on the research described above, the automatic image matting method TGMatting, which relies on Trimap generation, can now be used to generate precise transparency masks without human interaction. Unlike the automatic Trimap-based matting approach in this paper, numerous methods dedicated to end-to-end non-portrait matting are currently available. In this study, we have chosen two fully end-to-end non-portrait matting methods, AIM and GFM, for comparison with TGMatting.

We compare and verify the final segmentation and background replacement effects using selected images

Table 5. Comparison of SAD values for end-to-end natural image matting methods

Test set	Test pictures	TGMatting	AIM [42]	GFM [38]
Alpha-Spider	a	908	1349	1460
	b	1197	1284	1774
	c	2644	4259	7831
	d	1433	3083	6084
	e	222	566	1634
	f	1326	2912	3339
	Average	1288	2242	3687

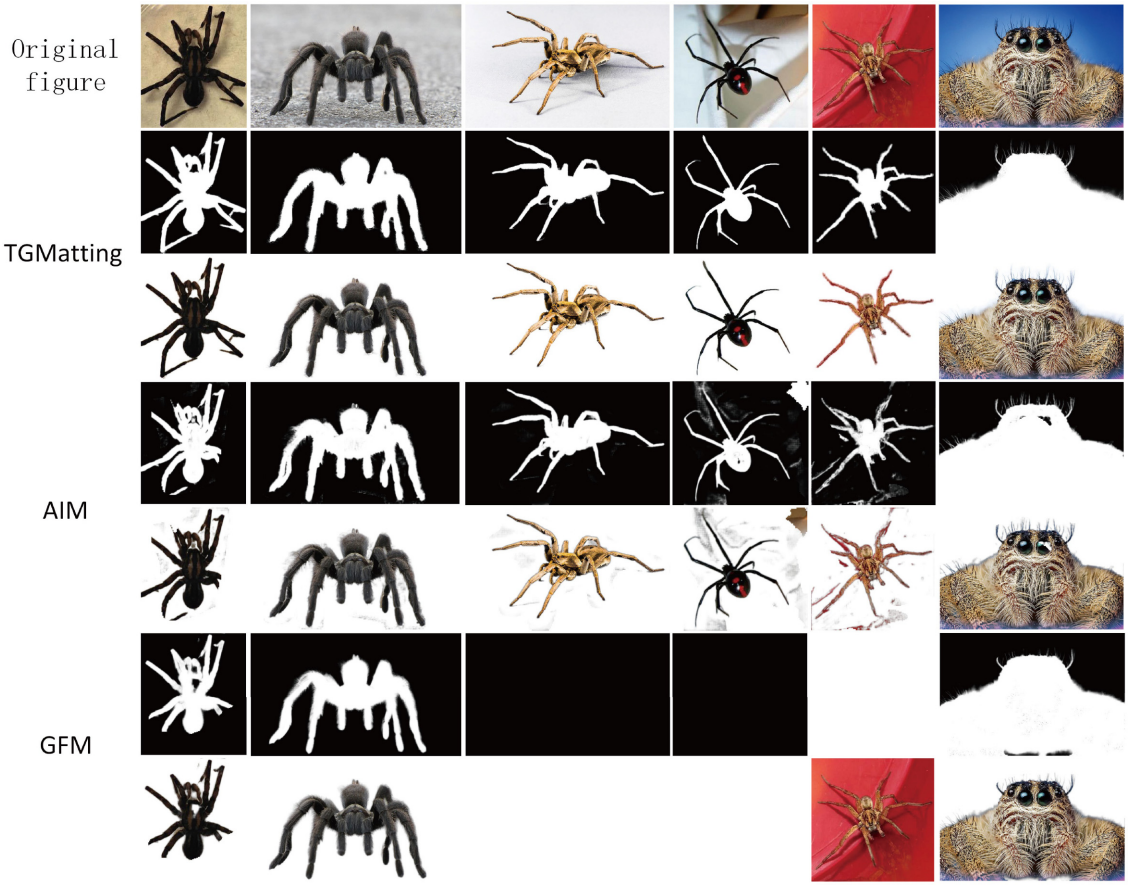

Fig. 17. Comparison of end-to-end image matting methods.

from the Alpha-Spider dataset. Table 5 and Fig. 17 present the specific results. We compare the performances of TGMatting, AIM, and GFM, two end-to-end natural image matting methods, on several significant images. The matting mask and foreground results are showcased. For non-significant targets, AIM and GFM struggle to identify significant regions, so they cannot be visually compared in an effective manner.

The AIM method performs better at finding the foreground for targets with well-defined saliency regions. However, for spiders, due to the dataset's lack of semantics and structural model defects, AIM tends to lose leg details and may misclassify background noise as foreground, thus resulting in a poorer overall matting effect. GFM also struggles to predict salient foregrounds effectively. Specifically, it tends to misclassify significant regions, sometimes predicting the entire image as either foreground or background, thus leading to errors in the

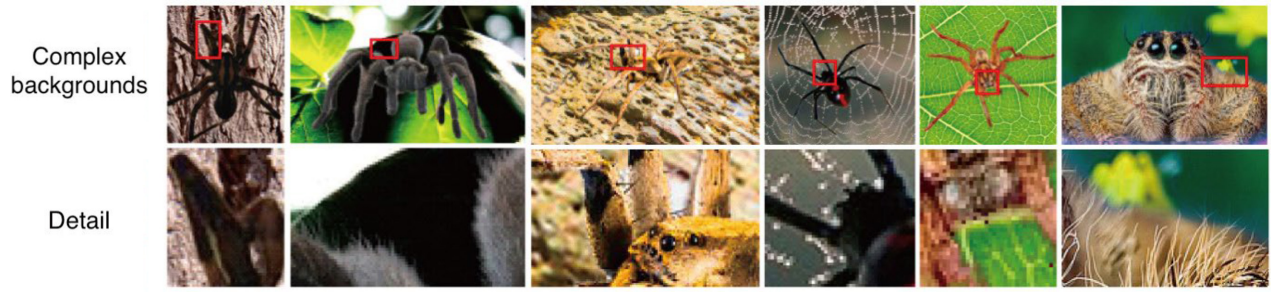

Fig. 18. Image matting details after background replacement.

matting process. The SAD results in Table 5 indicate that our method achieves the lowest values in predicted targets such as (a), (b), and (f), thus demonstrating superior performance.

Based on Fig. 16, this paper demonstrates a process that can be used to obtain a transparent background for the foreground image using the transparency mask. To enhance visibility, the method described in this paper first replaces the background with a solid color—in this case, white—to showcase the synthesized effect with various complex backgrounds. The transparent foreground of each spider is also replaced with different complex backgrounds, as depicted in Fig. 18. The results reveal a seamless integration of edge areas with the background, minimal artifacts, and hair details that maintain their original image context without incorporating background information.

IV. CONCLUSION

The purpose of image matting is to achieve fine segmentation of images. However, it is challenging to achieve automatic matting due to the difficulty of Trimap production. Unlike end-to-end training methods, this paper addresses the issue of automatic matting by focusing on obtaining high-quality Trimap. To tackle the challenges posed by low resolution, small targets, and complex foreground edge images, we designed an automatic image matting algorithm called TGMatting that is built upon Trimap generation as a foundation.

TGMatting comprises the U^2S-Net pre-segmentation module, BGTrimap automatic Trimap generation module, and FBAMatting image matting module. This paper has constructed a spider segmentation dataset, SOD-Spider, and a spider matting test dataset, Alpha-Spider, to evaluate the method's effectiveness on low-resolution, small-target, and complex foreground-edge images.

The U^2S-Net pre-segmentation module, which is based on the U^2-Net method, reduces model depth to mitigate over-downsampling of low-resolution images, and it modifies the loss function to diminish interference from bottom loss, thereby enhancing coarse segmentation results for such images. The BGTrimap automatic Trimap generation module leverages algorithms like region growing, threshold segmentation, and edge detection. Initially, seed points are automatically obtained using the region growing method from expanding edges, which is constrained by Manhattan distance for precise segmentation with a low threshold, and to eliminate background redundant information. Subsequently, the Sobel operator facilitates edge detection, and the denoised and binarized edge results yield the mixed pixel region. Finally, Trimap is generated with the assistance of Otsu segmentation and the pre-segmentation results.

TGMatting successfully automates matting for low-resolution, small-target, and complex foreground-edge images like spiders. However, it faces challenges in generating Trimap for images with extensive transparency areas, such as glass, grids, or flames. The primary obstacle here arises from the concentration of blended pixel areas within the foreground target, which renders conventional edge acquisition methods ineffective in identifying these regions. To address this issue in the future, we propose exploring transparent region detection methods or optimizing saliency target detection algorithms. By obtaining pre-segmentation results from various perspectives, we ultimately aim to enhance Trimap generation for such semantic elements.

CONFLICT OF INTEREST

The authors have declared that no competing interests exist.

ACKNOWLEDGEMENTS

This research was funded by Yunnan Fundamental Research Projects (No. 202201AT070006), Yunnan Post-doctoral Research Fund Projects (No. ynbh20057), and Major Science and Technology Project of Yunnan Province (No. 202002AA100007).

REFERENCES

1. Y. Liang, H. Huang, Z. Cai, Z. Hao, and F. Feng, "A review of natural image matting techniques," *Application Research of Computers*, vol. 38, no. 5, pp. 1294-1301, 2021. https://doi.org/10.19734/j.issn.1001-3695.2020.06.0111
2. X. Lu and Z. Liu, "A review of image semantic segmentation technology based on deep learning," *Software Guide*, vol. 20, no. 1, pp. 242-244, 2021.
3. C. Shi, W. Zhang, H. Chen, and L. Ge, "Survey of salient object detection based on deep learning," *Journal of Frontiers of Computer Science & Technology*, vol. 15, no. 2, pp. 219-232, 2021. https://doi.org/10.3778/j.issn.1673-9418.2007074
4. N. Xu, B. Price, S. Cohen, and T. Huang, "Deep image matting," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, HI, USA, 2017, pp. 2970-2979. https://doi.org/10.1109/CVPR.2017.41
5. A. Kirillov, E. Mintun, N. Ravi, H. Mao, C. Rolland, L. Gustafson, et al., "Segment anything," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, Paris, France, 2023, pp. 4015-4026. https://doi.org/10.1109/ICCV51070.2023.00371
6. H. Zhang, F. Li, X. Zou, S. Liu, C. Li, J. Yang, and L. Zhang, "A simple framework for open-vocabulary segmentation and detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision* Paris, France, 2023, pp. 1020-1031. https://doi.org/10.1109/ICCV51070.2023.00100
7. J. Liu, Y. Zhang, J. N. Chen, J. Xiao, Y. Lu, B. A. Landman, et al., "Clip-driven universal model for organ segmentation and tumor detection," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, Paris, France, 2023, pp. 21152-21164. https://doi.org/10.1109/ICCV51070.2023.01934
8. G. L. Yao, "survey on pre-processing in image matting," *Journal of Computer Science and Technology*, vol. 32, no. 1, pp. 122-138, 2017. https://doi.org/10.1007/s11390-017-1709-z
9. J. Li, G. Yuan, and H. Fan, "Robust trimap generation based on manifold ranking," *Information Sciences*, vol. 519, pp. 200-214, 2020. https://doi.org/10.1016/j.ins.2020.01.017
10. Q. Chen, T. Ge, Y. Xu, Z. Zhang, X. Yang, and K. Gai, "Semantic human matting," in *Proceedings of the 26th ACM International Conference on Multimedia*, Seoul, Republic of Korea, 2018, pp. 618-626. https://doi.org/10.1145/3240508.3240610
11. Q. Ran and J. Feng, "Automatic cutout algorithm for human foreground," *Journal of Computer Aided Design and Computer Graphics*, vol. 32, no. 2, pp. 277-286, 2020.
12. X. Wang, Q. Wang, G. Yang, and X. Guo, "Automatic image cutout algorithm based on attention mechanism and feature fusion," *Journal of Computer Aided Design and Computer Graphics*, vol. 32, no. 9, pp. 1473-1483, 2020.
13. C. Henry and S. W. Lee, "Automatic trimap generation and artifact reduction in alpha matte using unknown region detection," *Expert Systems with Applications*, vol. 133, pp. 242-25, 2019. https://doi.org/10.1016/j.eswa.2019.05.019
14. V. Gupta and S. Raman, "Automatic trimap generation for image matting," in *Proceedings of 2016 International Conference on Signal and Information Processing (IConSIP)*, Nanded, India, 2016, pp. 1-5. https://doi.org/10.1109/ICONSIP.2016.7857477
15. D. Cho, S. Kim, Y. W. Tai, and I. S. Kweon, "Automatic trimap generation and consistent matting for light-field images," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 8, pp. 1504-1517, 2017. https://doi.org/10.1109/TPAMI.2016.2606397
16. X. Xie, D. Yin, X. Xu, X. Liu, C. Luo, and G. Xie, "A review of weakly supervised image semantic segmentation based on image-level labels," *Journal of Taiyuan University of Technology*, vol. 52, no. 6, pp. 894-906, 2021.
17. J. Long, E. Shelhamer, and T. Darrell, "Fully convolutional networks for semantic segmentation," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Boston, MA, USA, 2015, pp. 3431-3440. https://doi.org/10.1109/CVPR.2015.7298965
18. O. Ronneberger, P. Fischer, and T. Brox, "U-Net: convolutional networks for biomedical image segmentation," in *Medical image computing and computer-assisted intervention–MICCAI 2015*. Cham, Switzerland: Springer, 2015, pp. 234-241. https://doi.org/10.1007/978-3-319-24574-4_28
19. X. Qin, Z. Zhang, C. Huang, M. Dehghan, O. R. Zaiane, and M. Jagersand, "U2-Net: going deeper with nested U-structure for salient object detection," *Pattern Recognition*, vol. 106, article no. 107404, 2020. https://doi.org/10.1016/j.patcog.2020.107404
20. M. N. Cheema, A. Nazir, P. Yang, B. Sheng, P. Li, H. Li, et al., "Modified GAN-cAED to minimize risk of unintentional liver major vessels cutting by controlled segmentation using CTA/SPET-CT," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 12, pp. 7991-8002, 2021. https://doi.org/10.1109/TII.2021.3064369
21. A. H. Al-Jebrni, S. G. Ali, H. Li, X. Lin, P. Li, Y. Jung, et al., "SThy-Net: a feature fusion-enhanced dense-branched modules network for small thyroid nodule classification from ultrasound images," *The Visual Computer*, vol. 39, pp. 3675-3689, 2023. https://doi.org/10.1007/s00371-023-02984-x
22. L. Dai, B. Sheng, T. Chen, Q. Wu, R. Liu, C. Cai, et al., "A deep learning system for predicting time to progression of diabetic retinopathy," *Nature Medicine*, vol. 30, pp. 584-594, 2024. https://doi.org/10.1038/s41591-023-02702-z
23. R. Ali, B. Sheng, P. Li, Y. Chen, H. Li, P. Yang, Y. Jung, J. Kim, and C. L. P. Chen, "Optic disk and cup segmentation through fuzzy broad learning system for glaucoma screening," *IEEE Transactions on Industrial Informatics*, vol. 17, no. 4, pp. 2476-2487, 2021. https://doi.org/10.1109/TII.2020.3000204
24. Z. Ke, J. Sun, K. Li, Q. Yan, and R. W. Lau, "MODNet: real-time trimap-free portrait matting via objective decomposition," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 36, no. 1, pp. 1140-1147, 2022. https://doi.org/10.1609/aaai.v36i1.19999
25. J. Li, J. Zhang, and D. Tao, "Deep automatic natural image matting," 2021 [Online]. Available: https://arxiv.org/abs/2107.07235.
26. J. Li, J. Zhang, S. J. Maybank, and D. Tao, "Bridging composite and real: towards end-to-end deep image matting," *International Journal of Computer Vision*, vol. 130, pp. 246-266, 2022. https://doi.org/10.1007/s11263-021-01541-0
27. H. Yu, N. Xu, Z. Huang, Y. Zhou, and H. Shi, "High-resolution deep image matting," *Proceedings of the AAAI Conference on Artificial Intelligence*, vol. 35, no. 4, pp. 3217-3224, 2021. https://doi.org/10.1609/aaai.v35i4.16432
28. X. Wen, "A review of research methods based on high-resolution large-size image matting," *Modern*

Computer, vol. 2021, no. 5, pp. 75-802, 2021. https://doi.org/10.3969/j.issn.1007-1423.2021.05.016
29. Z. Fu, "Research on small target detection method of insect growth stage based on convolutional neural network," M.S. thesis, Changchun University of Technology, Jilin, China, 2021.
30. F. F. Li, J. Deng, and K. Li, "ImageNet: constructing a large-scale image database," *Journal of Vision*, vol. 9, no. 8, pp. 1037-1037, 2009. https://doi.org/10.1167/9.8.1037
31. 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 *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, HI, USA, 2017, pp. 136-145. https://doi.org/10.1109/CVPR.2017.404
32. Z. Yao and S. Li, "Annual report of Chinese spider taxonomy in 2020," *Biodiversity Science*, vol. 29, no. 8, pp. 1058-1063, 2021. https://doi.org/10.17520/biods.2021140
33. C. Rhemann, C. Rother, J. Wang, M. Gelautz, P. Kohli, and P. Rott, "A perceptually motivated online benchmark for image matting," in *Proceedings of 2009 IEEE Conference on Computer Vision and Pattern Recognition*, Miami, FL, USA, 2009, pp. 1826-1833. https://doi.org/10.1109/CVPR.2009.5206503
34. J. Shi, Q. Yan, L. Xu, and J. Jia, "Hierarchical image saliency detection on extended CSSD," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 38, no. 4, pp. 717-729, 2016. https://doi.org/10.1109/TPAMI.2015.2465960
35. G. Li and Y. Yu, "Visual saliency based on multiscale deep features," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Boston, MA, USA, 2015, pp. 5455-5463. https://doi.org/10.1109/CVPR.2015.7299184
36. J. Li, G. Yuan, and H. Fan, "Generating trimap for image matting using color co-fusion," *IEEE Access*, vol. 7, pp. 19332-19354, 2019. https://doi.org/10.1109/ACCESS.2019.2896084
37. E. Shahrian, D. Rajan, B. Price, and S. Cohen, "Improving image matting using comprehensive sampling sets," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Portland, OR, USA, 2023, pp. 636-643. https://doi.org/10.1109/CVPR.2013.88
38. Y. Aksoy, T. Ozan Aydin, and M. Pollefeys, "Designing effective inter-pixel information flow for natural image matting," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, Honolulu, HI, USA, pp. 29-37. https://doi.org/10.1109/CVPR.2017.32
39. Q. Chen, D. Li, and C. K. Tang, "KNN matting," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 35, no. 9, pp. 2175-2188, 2013. https://doi.org/10.1109/TPAMI.2013.18
40. S. Deng and Y. Huang, "Fast algorithm for dilation and erosion of binary images," *Computer Engineering and Applications*, vol. 53, no. 5, pp. 207-211, 2017. https://doi.org/10.3778/j.issn.1002-8331.1508-0007
41. M. Forte and F. Pitie, "F, B, Alpha Matting," 2020 [Online]. Available: https://arxiv.org/abs/2003.07711.
42. Y. Liu, J. Xie, X. Shi, Y. Qiao, Y. Huang, Y. Tang, and X. Yang, "Tripartite information mining and integration for image matting," in *Proceedings of the IEEE/CVF International Conference on Computer Vision*, Montreal, Canada, 2021, pp. 7555-7564. https://doi.org/10.1109/ICCV48922.2021.00746

Jianming Wang https://orcid.org/0009-0004-7729-7284

Jianming Wang received B.S., and M.S., degrees from the Department of Computer and Information Science and the School of Resources at Southwest Forestry University in 2010 and 2013, respectively, and received Ph.D. degree from the School of Information at Beijing Forestry University in 2017. Since July 2017, he has been an associate professor for the Department of Mathematics and Computer Science at Dali University, Dali, China. His research focuses on artificial intelligence, computer vision, geographic information systems, and forestry statistical modeling.

Jiang Xiao https://orcid.org/0009-0004-4885-2808

Jiang Xiao is a master's student majoring in Computer technology at the Department of Mathematics and Computer Science, Dali University since 2023. His research interests are in deep learning and computer vision.

Yuhang Zhang

Yuhang Zhang obtained a master's degree from the School of Mathematics and Computer Science at Dali University in 2022. His research focuses on artificial intelligence and computer vision.

Jiting Yin

Jiting Yin is a forestry engineer with a master's degree. She graduated from Jilin Agricultural University with a Bachelor of Science degree in Plant Science and Technology in 2010. In July 2013, she graduated from Southwest Forestry University with a Master's degree in Forestry. She has been working at the Forestry and Grassland Science Research Institute of Dali Prefecture since September 2014, mainly focusing on forestry informatization.

Zizhong Yang

Zizhong Yang received B.S. and Ph.D. degrees in Biology Education and Zoology from the College of Life Sciences of Yunnan Normal University in 1994 and 2006, respectively. Since 2005, he has been a professor at the School of Pharmacy, Dali University, Dali, China. His research focuses on the construction of medicinal insect resources in Southwest China, the investigation and classification of spider resources in Yunnan, and the systematics, toxins, and evolution of the spiders of the Macrotheloidae family in China.