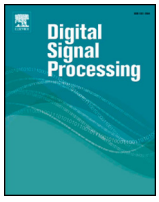
Digital Signal Processing 133 (2023) 103849

Contents lists available at ScienceDirect

Digital Signal Processing

journal homepage: www.elsevier.com/locate/dsp



Semi-supervised portrait matting using transformer

Xinyue Zhang [a], Changxin Gao [b], Guodong Wang [a,*], Nong Sang [b], Hao Dong [a]

[a] College of Computer Science and Technology, Qingdao University, Qingdao, 266071, China
[b] Huazhong University of Science and Technology, Wuhan, 430074, China

ARTICLE INFO

Article history:
Available online 25 November 2022

Keywords:
Portrait matting
Semi-supervised
Transformer
Global features
Local features

ABSTRACT

Transformer is successfully applied in many tasks but is not sensible to directly embed in the portrait matting. Such an operation can effectively consider global features ignoring local features, which leads to the difficulty of identifying the complete and edge-refined portraits. What is more, the existing supervised architectures are feeble against the unlabeled images. To achieve the effective excavation of both local and global features, we design a semi-supervised network that leverages Transformer to capture the global features. The arduous task of compensating more local features is left to the portrait detailed decoding module (PDDM) that we designed. In addition, to provide the possibility of improving effectiveness when faced with unlabeled images, we design an intelligent pseudo-label generation strategy to embed in our semi-supervised network. This strategy can generate more detailed pseudo-labels than predicted results through redundant foreground filtering and edge adjustment. Compared to existing portrait matting methods, our network successfully achieves performance improvements with a small number of datasets and has the ability to train on unlabeled datasets. The training models and the code will be released at https://github.com/XinyueZhangqdu/SSPMTransformer/.

1. Introduction

Portrait matting is the task of predicting detailed alpha mattes through the given images containing some portraits. Although portrait matting has urgent application requirements in real life, it has been limited by the number of datasets and cannot be further improved in deep learning. It is time-consuming to manually mark the hair of each portrait, so the existing portrait matting models are unable to obtain a large amount of training data and suffer from the bottleneck of improving the effect. In addition, the existing portrait matting methods are still based on CNN to construct the network architecture, ignoring the global features. CNN-based architectures cannot capture the same abundant global features as Transformer architectures. The Transformer architecture, which is weak in capturing local features, cannot directly apply to the portrait matting task with demanding details.

The recent adoption of semi-supervised networks in other areas circumvents the problem of over-reliance on manually annotated datasets. Besides, Transformer [1,2] is better at capturing global features while CNN-based modules [3,4] will pay more attention to local features. Based on the above two existing technical supports, we build a semi-supervised network based on Transformer in the field of portrait matting. It is worth noting that our network embeds the Transformer at the front of our network to capture global features of the images (like shape features), and we embed a portrait detailed decoding module at the back of the network to compensate for local features lost in Transformer (like some hair details). The embedding of this portrait detailed decoding module allows us to capture more hair detail features and turn them into alpha mattes for display. By combining the front-end Transformer architecture with the back-end portrait detailed decoding module of the network, our network gives consideration to both global and local features. In addition, to improve the semi-supervised learning effect in the face of unlabeled data, we design a pseudo-label generation strategy. In this strategy, we attain more refined pseudo-labels by algorithmically removing redundant foregrounds and tweaking the image edges. Our main contributions can be summarized as follows:

1). A semi-supervised network (as shown in Fig. 1.) that is compatible with global and local features through the embedded Transformer and the **p**ortrait **d**etail **d**ecoding **m**odule (PDDM) designed by us. With the help of PDDM, the network enhances the possibility of capturing complete and edge-elaborate portraits,
2). An intelligent strategy for generating pseudo-label with unlabeled images. By removing redundant foregrounds and rearranging the edges of the portraits, pseudo-labels are generated.

* Corresponding author.
E-mail address: doctorwgd@gmail.com (G. Wang).

https://doi.org/10.1016/j.dsp.2022.103849
1051-2004/

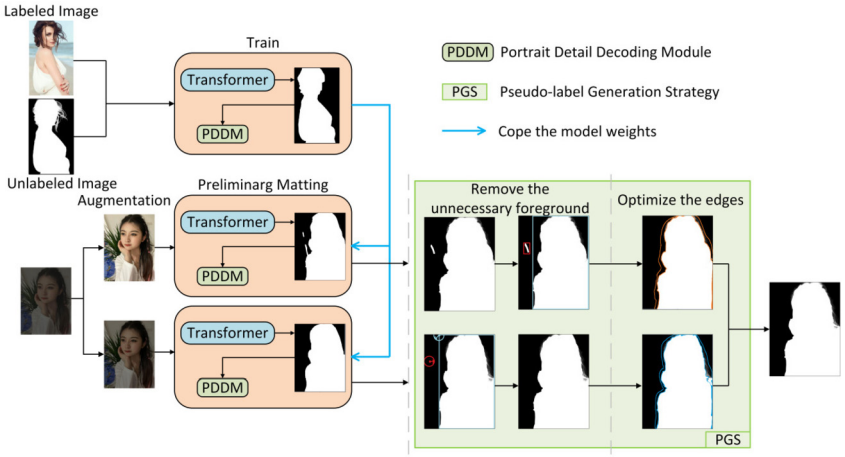


Fig. 1. A diagram of the architecture of our proposed Transformer based semi-supervised network (SemiTrans). Pseudo-label Generation Strategy is responsible for adjusting the foreground and edge parts of the network prediction results on the unlabeled datasets. The Portrait Detailed Decoding Module has the ability to extract local features in the network, such as some hair details, which is complementary to Transformer. The design of the semi-supervised network gives the network the ability to improve performance when faced with unlabeled datasets.

With the assistance of these pseudo-labels, the unlabeled images also have the optimization standards that can be referred to in the training, and the effects of matting on the data without labels are improved.

3). Break the bottleneck of effectiveness on labeled data. With the same training strategy, we make progress across multiple datasets both in terms of metrics and in terms of visual comparisons.

2. Related work

With the development of deep learning, more and more computer vision tasks [5] begin to adopt neural networks for modeling and implementation. In this session, we introduce some terms related to our work.

2.1. Transformer

Transformer's [1] revolutionary improvement in the field of computer vision has attracted wide attention from the academic world [6–8]. The Transformer's general modeling [9,10] capabilities come from two aspects: On the one hand, Transformer [1] can be seen as a graph modeling approach. Graphs are fully connected and the relationships between nodes are learned in a data-driven manner. Transformer [1], on the other hand, utilizes the philosophy of validation to establish relationships between graph nodes. However, the direct application of Transformer [1] in portrait matting can only pay attention to the global information similar to the shape of the portrait, and can not pay attention to the local information of the hair. The architecture named NMW [11] ensembles Swin Transformer and DetectorRS with Resnet backbone. This effective combination has been proven to perform better compared to the respective baseline model. In another paper [12], the author proposes to transmit scale-consistent feature and position feature to the encoder in Transformer [1] to mine deep correlation patterns. Both of these methods have achieved remarkable breakthroughs in classification and detection respectively. However, in the field of portrait matting, more details (like hair strands in a portrait) should be given more attention. In addition, the above two methods ignore the establishment of residual edges when further extracting abstract features between the two features when combining the features extracted by Transformer and those obtained based on CNN. Thus, the above two methods can only carry out simple fusions, but cannot carry out multiple fusions to continuously make up for each other's previous high-level abstract features. Recently, some scholars [13–15] have verified that Transformer [1] is more inclined to obtain global features in images, and they are trying to design modules that can better extract local features to reduce the loss of effective features in their field. Therefore, it is urgent to design a detailed decoding module for portrait matting. [11]

2.2. Portrait matting

Depending on the type of input, portrait matting can be roughly divided into three categories. (1) Architectures [18,19] that require additional background image input: In order to reduce the difficulty of the task, some scholars need to input the corresponding background images in addition to the complete images when constructing the architecture. These architectures, which require the background images, have good application prospects in some specific background environments but are not suitable for situations where the background is unknown. (2) Architectures that require the assistance of trimaps [20–22]: Trimaps roughly divide the images into the foreground, background, and unknown areas. With the assistance of trimaps, the networks are given more space optimization details. But trimaps take time to annotate manually. (3) Architectures that do not require additional input [23–28]: For simplicity of matting, many recent architectures do not require additional input. Portrait matting is accomplished by simply inputting the images to be processed into the network architectures. However, due to the difference in construction architectures, there are significant differences in the effects of these methods in image matting. In particular, visually, some small areas cannot be identified and displayed in alpha mattes.

3. Methods

The proposed semi-supervised network (as shown in Fig. 1) is used to enhance the effect of image matting with small amounts of data. This architecture consists of two main parts: (1) Transformer [1] architecture for capturing the general outline of the portrait and (2) a portrait detailed decoding module for capturing local features such as hair details for portrait matting. The embedding of these two modules enables the network to acquire global features and local features to complement each other. This semi-supervised network architecture (SemiTrans) also offers the possibility of further improvements in unlabeled datasets. Besides, an intelligent pseudo-label generation strategy is designed and successfully applied to obtain fine training references from unlabeled

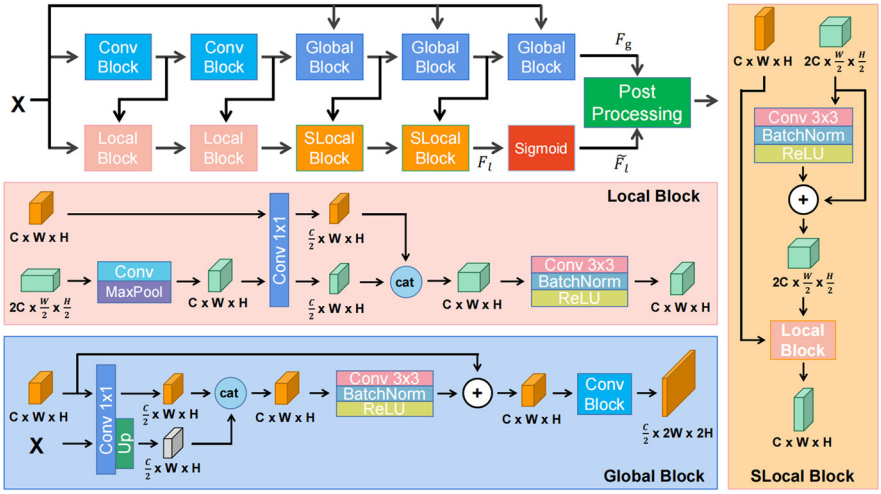


Fig. 2. A detailed representation of the portrait detailed decoding module. Conv Block refers to the Conv + Batchnormalization [16] + ReLu [17] operations. This module makes up for local features that Transformer [1] architecture cannot capture, and integrates global features with local features.

data. In the following, we introduce the mentioned contents in detail.

3.1. The network architecture

Since existing matting network architectures are still built on CNN and ignore global features, Transformer [1] has been successfully embedded in many other areas and proven to be more friendly to capturing global features. Therefore, Transformer [1] architecture is embedded in the front part of the network to obtain the global portrait features in the input images, and the portrait detailed decoding module based on CNN design is embedded in the back end of the network to analyze the local features (as shown in Fig. 1). Our embedded Transformer [1] module includes the encoder and decoder parts of Transformer architecture [1]. With the cooperation of Transformer [1] and the proposed portrait detailed decoding module, the network avoids the defect of not being able to take into account both global and local features. In addition, in Fig. 1, we mention that images need to go through two kinds of augmentations before being transported to the network. When we perform the augmentations of the images, we will first determine whether they need to be adjusted. First, the images are converted into HSV format and saved in HSV_image, and then the mean value of HSV_image [:,:,2] is calculated. Compared to RGB color space, images in HSV format are closer to the way humans perceive color. By calculating the HSV, we can obtain the brightness value saved in the last channel [29,4]. If the brightness value is greater than 130, then the image brightness is within the normal range for processing images. Therefore, when the value of HSV is greater than 130, the image we send to the network includes the original image I_o and the darkened image I_d. Conversely, when the brightness information of the image is less than 130, the network input is the brightened image I_b and the original image I_o. The purpose of this two-stage delivery is to achieve the following two objectives. (1) Avoid insufficient light and exposure to some areas that can not be detected. (2) According to the different lights, the initial acquisition of pictures transmitted to the network will have different results. Prepare for cross-referencing adjustments using the pseudo-label generation strategy. Below, we give a detailed introduction to the proposed detailed decoding module and the pseudo-label generation strategy.

3.1.1. Portrait detailed decoding module

As shown in Fig. 2, our detailed decoding module can be roughly divided into two branches. A branch performs basic convolution operations on features acquired by Transformer [1] and then tranches them to the Global Block for processing. The other branch is processed by the complex Local Block and SLocal Block based on CNN thought design. At the same time, this branch continuously incorporates global features extracted from the other branch. We show the structure and internal size changes of the Local Block, Global Block, and SLocal block in detail in Fig. 2. After two branches of processing, we perform the Post Processing operation (as shown in Algorithm 1) represented by the green square in Fig. 2. The implications of the operations in Algorithm 1 are described in section 3.1.3.

Algorithm 1 The algorithm description for the Post Processing operation in Fig. 2.

Input: The results of the two branches in Fig. 2: named F_g and $\tilde{F}_l$.
Output: Fusion_sigmoid for matting final result prediction.
$v_g, i_g \leftarrow Max(F_g, 1)$
$\hat{i}_g \leftarrow i_g[:, None, :, :].float()$
trimap_mask $\leftarrow \hat{i}_g$.clone()
trimap_mask[trimap_mask==2] $\leftarrow$ 0
foreground_mask $\leftarrow$ index.clone()
foreground_mask[foreground_mask==1] $\leftarrow$ 0
foreground_mask[foreground_mask==2] $\leftarrow$ 1
fusion_sigmoid $\leftarrow \tilde{F}_l \times$ trimap_mask+foreground_mask

The core function of Local Block is to receive the abstract features acquired by the previous layer and the features extracted by Conv Block for further detail extraction. Global Block is responsible for establishing long-distance links and feature extraction for the features input to the detailed decoding module and the features of the previous layer of this module. It tends to capture more global features. SLocal Block attempts to further observe some local detail features hidden from the features obtained from the global perspective, and complement each other with the features of the previous layer. In the following, we describe the details of how the output of Local Block F_l, Global Block F_g, and SLocal Block F_s is formed. To facilitate the introduction of the equations, some abbreviations in the equations are introduced here. Assume that the characteristics transported to the Local Block are $Input_{l1}$ and $Input_{l2}$, respectively. The width and height of $Input_{l2}$ are half that of $Input_{l1}$ and the number of channels is twice that of $Input_{l1}$. In addition, assume that χ and κ in convolution function Conv(χ, κ) are the number of channels to be convolved and the size of convolution kernel respectively. Concat(ψ, ζ, γ) indicates that ψ and ζ are stacked on the γ channel. CBR stands for 3×3 convolution,

Batchnormalization, and processing of the ReLu function. Upsampling means doubling the width and height of features.

$$F_1 = Conv(Input_{l1}, 1) \tag{1}$$

$$F_2 = Conv(MaxPool(Conv(Input_{l2}, 3)), 1) \tag{2}$$

$$F_l = CBR(Concat(F_1, F_2, -1)) \tag{3}$$

Why do Local Blocks have the ability to focus on more detail than Global Blocks? This is due to the inclusion of MaxPool [30] in the Local Block prior to feature fusion. Max Pooling means that several feature values are extracted from a filter, only the largest Pooling layer is obtained as the reserved value, and all other feature values are discarded. The maximum value means that only the strongest features are retained and other weak features are discarded. In addition, Local Block combines the features of the previous layer and those processed by Conv Block to extract the features again, so as to ensure the richness of the features. Assume that the characteristics transported to the Global Block are $Input_{g1}$ and $Input_{g2}$, respectively.

$$G_1 = Conv(Input_{g1}, 1) \tag{4}$$

$$G_2 = Upsampling(Conv(Input_{g2}, 1)) \tag{5}$$

$$F_g = CBR\big(CBR(Concat(G_1, G_2, -1)) \oplus Input_{g1}\big) \tag{6}$$

In the Global Block, a long-distance residual edge is established from the input end to avoid the characteristic losses obtained from different observation angles during the series operation. In addition, the features processed by Transformer [1] are conveyed to the features processed by the previous layer in this module for addition. It also complements the global features captured in the previous Transformer [1] module. Assume that the characteristics transported to SLocal Block are $Input_{s1}$, $Input_{s2}$. Then the output F_s of SLocal Block can be obtained by the following calculation:

$$F_s = LB\big(CBR(Input_{s2}) \oplus Input_{s2}, Input_{s1}\big) \tag{7}$$

where LB is the abbreviation of Local Block.

3.1.2. Pseudo-label generation strategy (PGS)

In order to obtain pseudo-labels comparable to manual labeling in semi-supervised learning, we design an intelligent pseudo-label generation strategy. This pseudo-label generation strategy can be divided into the following steps:(1) Train a network (denoted by N) with a small number of labeled datasets and retain its model parameters δ. (2) The network $\widehat{N}$ after loading the parameters δ saved during training is sent with two images I_o, I_p that are enhanced to obtain two initial alpha mattes, denoted as α^1 and α^2 respectively.

$$\alpha^1, \alpha^2 = \widehat{N}([I_o, I_p], \delta) \tag{8}$$

(3) Divide the foreground areas by α^1 and α^2. Assume that the foreground region existing in α^1 is α_i^1, and the foreground region existing in α^2 is α_i^2. So there are multiple foreground regions in both alpha mattes. It is assumed that α^1 and α^2 each have n sub-foreground regions, which can be formulated as:

$$\alpha^\tau = \alpha_0^\tau, \alpha_1^\tau, \ldots, \alpha_n^\tau \tau = 1, 2 \tag{9}$$

Then, calculate the outermost boundary box of each sub-foreground region, and record the position of the boundary box with its upper-left and lower-right coordinate points respectively. By comparing the location regions between boundary boxes in different alpha mattes, if the upper-left coordinate point of a boundary box in α^1 is the center of the circle, and the location with a radius of 60 pixels does not find a sub-foreground region in α^2, then this region should be divided into background region. Similarly, other redundant foreground regions can be renaturalized as background regions. (4) The foreground areas in two different alpha mattes are fused to complement each other's missing edge areas. At the same time, the hair details of the portrait can be better supplemented to provide meticulous pseudo-labels ϖ.

$$\varpi = \tilde{\alpha^1} \oplus \tilde{\alpha^2} \tag{10}$$

where $\tilde{\alpha^1}$ and $\tilde{\alpha^2}$ mean the resulting graphs with black background.

Compared with the previous pseudo-label generation strategy in semi-supervision, we mainly have the following differences: (1) In the existing semi-supervised methods, the model is first trained based on the labeled datasets, and then the results obtained by the trained model prediction are directly used as pseudo labels. Unlike the above methods, our method further adjusts and optimizes the obtained results based on the foreground and background application algorithm. (2) In the process of removing the extra foreground, we mentioned that the bounding box is established in different sub-foreground areas and the bounding circle is established with the upper left corner of the bounding box as the center. Then compare the same circular area with the result of another stage. As far as we know, there is currently no architecture for such design optimization prospects.

3.1.3. Post processing

Algorithm 1 describes how to obtain the final output according to the architecture global features and local features. Assume that the feature obtained by the Global Block is F_g, and the feature obtained by the last SLocal Block is F_l. Firstly, F_g is convolved once and the number of channels is adjusted to 3 with the kernel size of 3. At this time, the shape of F_g can be expressed as [1,3,256,256] according to the layout of features in Pytorch. Then the features in F_l are processed by the following equation.

$$\tilde{F}_l = \frac{1}{1 + e^{-F_l}} \tag{11}$$

where the shape of $\tilde{F}_l$ is [1,1,256,256]. In the algorithm, we first use the Max function to obtain the maximum value v_g and index value i_g based on the second dimension of the feature. Obviously, the second dimension here points to the dimension that the channel is in. The value stored in i_g is a combination of the values 0, 1, and 2. In order to establish the relationship with the local feature region score F_l, we adjust the shape of i_g into four dimensions to form $\hat{i_g}$. With the help of the record operation of algorithm 1, the value of $fusion\_sigmoid$ is divided into 0, 1, and υ (υ is a decimal between 0 and 1) by the value stored in $\hat{i_g}$. The value of $fusion\_sigmoid$ is associated with local features F_l only in the boundary area (neither the white foreground area nor the black background area). In the foreground and background regions, it is more dependent on the calculation of F_g. Finally, multiply the value of $fusion\_sigmoid$ by 255 to obtain the alpha matte consisting of three parts.

3.2. Key points of using a small amount of data to achieve improvement

In the field of portrait matting, there has been a lack of training datasets that can support fully supervised learning due to the time-consuming and complicated manual annotation, so we start training based on semi-supervised learning. The semi-supervised network we design is first trained using a batch of alpha mattes with manually marked labels and corresponding images. In the

face of a new batch of data without labels, pseudo-labels are generated by the proposed pseudo-label generation strategy to assist training. When confronted with the labeled datasets, the network is trained in the following way to calculate losses:

$$edge\_loss = ||e_i - \alpha_e||_1 \tag{12}$$

$$alpha\_loss = ||a_i - \alpha_o||_1 \tag{13}$$

$$Loss = edge\_loss + alpha\_loss \tag{14}$$

where α_o stands for alpha mattes in the datasets. e_i represents the edge graph of human portrait generated after the expansion and corrosion operation of alpha mattes a_i predicted by the network, and α_e is the edge graph obtained after the expansion and corrosion operation of α_o in datasets. Equation (3) tries to calculate the edge gap between the predicted results and the label in the real datasets, while equation (4) focuses more on calculating the overall gap between them. In the training of unlabeled datasets, the pseudo-labels generated by the pseudo-label generation strategy replace the role of labels in the original datasets to assist the training.

4. Experimental results

Some experimental details and the effect of our proposed portrait matting architecture on different datasets are reported in this section.

4.1. Evaluate metrics

In this paper, we use seven different evaluate metrics. Below, we give some equation descriptions of these seven different metrics. It is assumed that the final alpha mattes obtained after model prediction are α^p, and the alpha mattes to be referenced in the datasets are α^d. Then the equation for calculating the 'sum of absolute differences' (SAD) can be expressed as:

$$SAD = \sum_i (|\alpha_i^p - \alpha_i^d|), \tag{15}$$

where $'\sum'$ means to calculate the sum of all the elements of the matrix. '||' symbol represents the absolute value of the operation. The 'sum of absolute differences in the foreground' (SFG), and the 'sum of absolute differences in the background' (SBG) only need to transmit the corresponding calculation of foreground region and background region respectively to equation (1). The calculation method of 'mean squared error' (MSE) and 'mean absolute difference' (MAD) can be formulated as:

$$MSE = \frac{\sum_i ((\alpha_i^p - \alpha_i^d)^2)}{\rho}, \tag{16}$$

$$MAD = \frac{\sum_i (|\alpha_i^p - \alpha_i^d|)}{\rho}, \tag{17}$$

where ρ represents the product of the width and height of α_p. The calculation of 'grad error' (Grad) can be expressed as:

$$Grad = \sum_i (\nabla\alpha_i^p - \nabla\alpha_i^d)^2, \tag{18}$$

where $\nabla\alpha_i^p$ and $\nabla\alpha_i^p$ represent the normalized gradient of the corresponding alpha matte, which is calculated by convolving matte with the first order Gaussian derivative filter. The calculation of 'connectivity error' (Conn) can be formulated as:

$$Conn = \sum_i (\phi(\alpha_i^p, \Omega), \phi(\alpha_i^d, \Omega)), \tag{19}$$

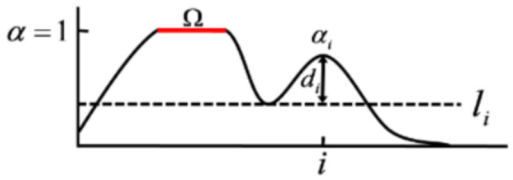


Fig. 3. An illustration of the variables used in the calculation of connectivity error. l_i is the maximum threshold at which pixel i can quadruple connect to Ω.

where Ω represents the completely opaque part of both α_i^p and the corresponding α_i^d (as shown in Fig. 3).

$$\phi(\alpha_i, \omega) = 1 - (\lambda_i \cdot \delta(d_i \geq \theta) \cdot d_i), \tag{20}$$

$$\lambda_i = \frac{\sum_{k \in K} dist_k(i)}{|K|} \tag{21}$$

where θ is set to 0.15. The K here represents the set of discrete α_i values between α_i. If the threshold is k, $dist_k$ calculates the normalized Euclidean distance between the pixel i closest connected to the source domain and itself.

4.2. Training details

SPDDataset [35] and AutomaticPortraitMattingDataset [37] are utilized as our training sets. SPDDataset contains 3210 portrait images and the corresponding detailed annotation files manually, AutomaticPortraitMattingDataset contains 1700 pairs of such data. We adopt these 4910 pairs of labeled data to conduct preliminary training on our proposed semi-supervised network. The learning rate during training decays from 0.03 to 0.0001. The training defects caused by direct attenuation are replaced by gradual attenuation. When passing 100 epochs each time, the current learning rate is multiplied by 0.45 as the learning rate of the next epoch. Each epoch uses the Adam optimizer [38] to optimize parameters. In the face of unlabeled datasets, pseudo-labels can be generated through the cooperation of the above pre-trained network and pseudo-label generation strategy. Then perform the same training as above when confronted with the labeled datasets.

4.3. Comparisons of state-of-the-art methods

Experiments comparing the effects of recent portrait matting structures are shown in Table 1. In Table 1, a total of 28 comparative experiments are performed on four different datasets. It is important to note that all the comparative tests followed the principle of a single variable. In addition to the differences in method architecture, each group of experiments undergoes the same training strategy as we mention above. Seven different measures [23] are utilized to measure the effectiveness of the experiments. It is worth noting that not only do we compare the results with those [23–25] built on CNN, but we also compare the results with Vitae [26] built on Transformer [1] with the same training strategy we mentioned above. Obviously, our architecture named SemiTrans has outperformed existing methods on multiple datasets. Both visually (as shown in Fig. 4) and across seven different measures, our approach is strongly demonstrated to be effective. In terms of MODNet [34] effects, MODNet [34] and our architecture use different training sets, so it cannot be directly compared with the effect of the paper architecture in MODNet [34]. The comparative experiments we show in Table 1 are trained on only SPDDataset [35] and AutomaticPortraitMattingDataset [37]. When MODNet [34] is trained with the same training strategy as ours, our architecture outperforms MODNet [34] on several datasets. In terms of the PPM-100 dataset [34], after performing the same training strategy, the effect of our architecture in the SAD, MSE, MAD, Grad,

Table 1
Comparison experiments. SAD, MSE, MAD, Grad, and Conn are the abbreviation of the 'sum of absolute differences', 'mean squared error', 'mean absolute difference', 'grad error', and 'connectivity error'; SFG/SBG is: the sum of absolute differences in the foreground/background [31]. The comparison architectures in the table are GFM [23], P3M [24], Mgm [25], Vitae [26], Vmf [32], and Mfm [33]. The datasets mentioned in the table including PPM-100 [34], SPDDataset [35], AdobeDataset [36], and AutomaticDataset [37].

Dataset	PPM-100							SPDDataset						
Method	GFM	P3M	Mgm	Vitae	Vmf	Mfm	Ours	GFM	P3M	Mgm	Vitae	Vmf	Mfm	Ours
Year	2022	2021	2021	2022	2022	2022	2022	2022	2021	2021	2022	2022	2022	2022
SAD	52.00	65.39	34.06	32.91	14.72	4.503	**3.288**	49.69	23.57	22.02	19.27	16.67	10.01	**9.032**
MSE	0.1936	0.1973	0.0502	0.0761	0.2762	0.0091	**0.0076**	0.1868	0.0681	0.0501	0.3107	0.3081	0.0920	**0.0307**
MAD	0.1983	0.2491	0.1293	0.1256	0.3041	0.0161	**0.0125**	0.1895	0.0952	0.0771	0.3244	0.3143	0.0911	**0.0344**
Grad	19.07	18.98	14.48	16.46	13.67	6.483	**6.418**	27.21	23.18	12.72	23.42	15.86	**12.16**	15.81
Conn	50.30	65.81	32.87	32.99	14.08	4.062	**2.990**	49.56	21.72	19.42	17.92	15.73	9.946	**7.289**
SFG	36.37	20.44	12.08	13.91	5.959	1.033	**0.8013**	32.11	5.225	6.174	5.310	6.095	4.594	**3.165**
SBG	7.727	10.36	10.59	8.204	2.663	0.060	**0.0102**	7.577	14.05	15.09	10.94	4.507	2.463	**1.129**
Dataset	AdobeDataset							AutomaticDataset						
Method	GFM	P3M	Mgm	Vitae	Vmf	Mfm	Ours	GFM	P3M	Mgm	Vitae	Vmf	Mfm	Ours
Year	2022	2021	2021	2022	2022	2022	2022	2022	2021	2021	2022	2022	2022	2022
SAD	77.09	28.54	25.37	25.62	10.25	4.981	**4.933**	60.77	47.58	45.03	47.27	23.81	6.224	**4.727**
MSE	0.2839	0.2131	0.0676	0.1072	0.1036	0.0103	**0.0097**	0.2214	0.2151	0.1765	0.1072	0.0541	0.0315	**0.0107**
MAD	0.2940	0.3325	0.2105	0.2613	0.1071	0.0179	**0.0188**	0.2318	0.2195	0.1912	0.1280	0.0927	0.0391	**0.0180**
Grad	24.13	20.22	16.60	17.58	12.38	6.565	**5.394**	17.89	15.33	12.62	13.59	15.60	6.0732	**3.9955**
Conn	70.17	28.18	24.85	25.21	10.04	4.991	**4.351**	59.62	47.43	44.19	24.44	23.09	4.691	**4.444**
SFG	21.66	12.10	5.019	2.311	4.649	1.015	**0.8885**	52.14	28.69	22.62	14.80	4.095	6.020	**1.480**
SBG	15.47	9.242	3.628	2.870	2.597	0.6572	**0.4535**	2.569	2.815	4.180	6.392	2.167	1.051	**0.639**

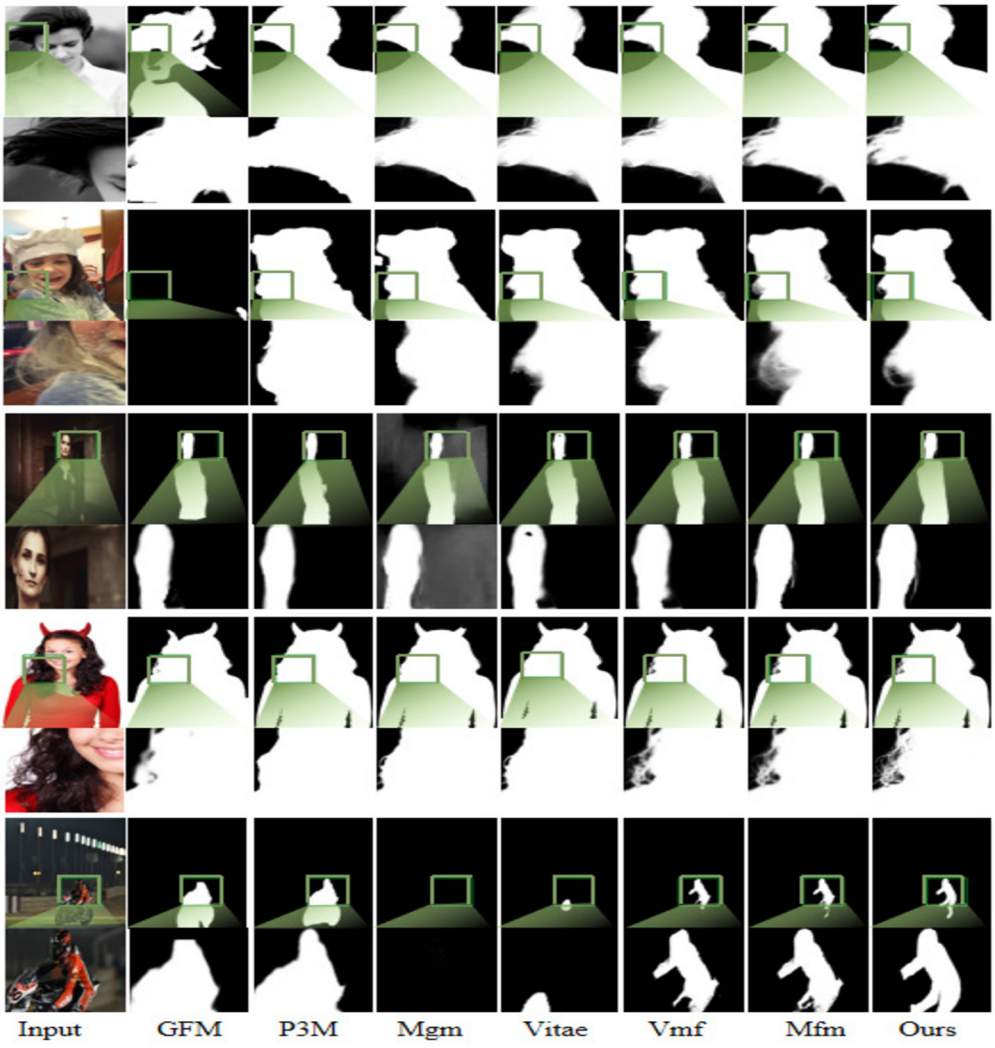
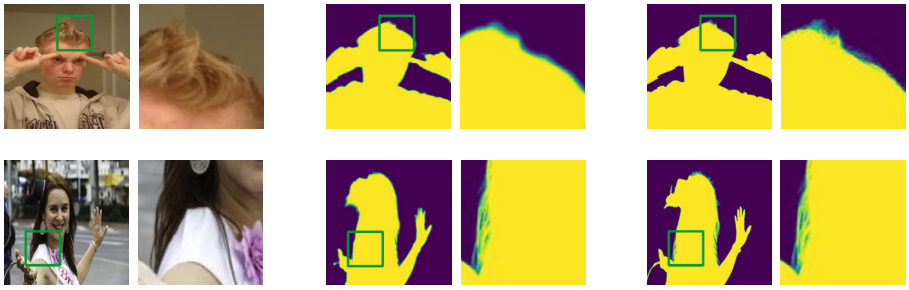


Fig. 4. Visual comparisons for portrait matting. Compared to the state-of-the-art methods, our SemiTrans captures more perceptually faithful images.



(d) Images (e) Ours w/o Local Block (f) Ours

Fig. 5. Visual comparisons for ablation studies.

Table 2
Comparison of the parameters of different architectures and the processing seconds of individual images. These comparisons are calculated based on 512× 512. The comparison architectures in the table are GFM [23], P3M [24], Mgm [25], Vitae [26], Vmf [32], and Mfm [33].

Method	GFM	P3M	Mgm	Vitae	Vmf	Mfm	Ours
Year	2022	2021	2021	2022	2022	2022	2022
Parameter	76.05	39.48	4.54	45.08	44.37	27.55	26.93
Time	0.32	0.29	0.05	0.09	0.13	0.08	0.10

Conn, SAD - FG, and SAD - BG beyond the effect of MODNet respectively 30.45, 0.0441, 0.1118, 7.492, 9.380, 10.81, 7.410. On the SPDDataset [35], the values of our architecture in seven indicators are respectively improved by 10.43, 0.0585, 0.0257, 15.75, 4.344, 11.02, 13.03, and 9.261 compared with MODNet. Besides, to test the model size of the other methods, we initialize a tensor and the shape is [1,3,512,512]. Through testing, we know that the model sizes of GFM [23], P3M [24], Vitae [26] are 76.058M, 39.480M, and 45.080M, respectively. The model size of our model is only 26.931M.

To obtain the inference time for each architecture, we test the inference time of six architectures at the resolution of 512 × 512. After testing, we conclude that the inference time of a single image required by GFM [23], P3M [24], Mgm [25], Vitae [26], Vmf [32] and Mfm [33] is 0.32 s, 0.29 s, 0.05 s, 0.09 s, 0.13 s, and 0.08 s, respectively. The single image inference time of our architecture is 0.10 s. Although our architecture is not the fastest architecture available, it has a significant improvement in performance in a relatively fast time (as shown in Table 1 and Table 2).

5. Ablation studies

In this article, we design a portrait detailed decoding module to compensate for local features (like some hair details) that are missing in Transformer [1]. The pseudo-label generation strategy is proposed to assist the semi-supervised training of unlabeled data. In this section, some ablation experiments are performed on the above designs to prove their effectiveness.

5.1. Portrait detailed decoding module

To verify the effectiveness of this module, the following ablation experiments are performed. Eight groups of ablation experiments are performed on the PPM-100 dataset [34] to verify the validity of the sub-modules of the portrait detailed decoding module. Sub-modules include Local Block, Global Block, and SLocal Block. We partially removed these sub-modules and tested the effect after the same training strategy as above. The results of these eight ablation trials are recorded in Table 3 using different measures. As can be seen from the data in Table 3, Local Block has the greatest influence on the network effect, and the removal of this module will greatly reduce the effect. We also perform some visual demonstrations. In Fig. 5, we show some of the resulting graphs without the Local Block. In addition, the resulting graph without adding a Global Block is shown in Fig. 6. It can be seen intuitively that in the absence of Local blocks, the hair details in some portraits cannot be paid attention to and displayed in the final alpha mattes. In the absence of a Global Block, some portrait contours become missing. The validity of the construction of the portrait detailed decoding module has been fully proved, both visually and in terms of measurement indicators.

Table 3
The ablation experiment of blocks in portrait detailed decoding module.

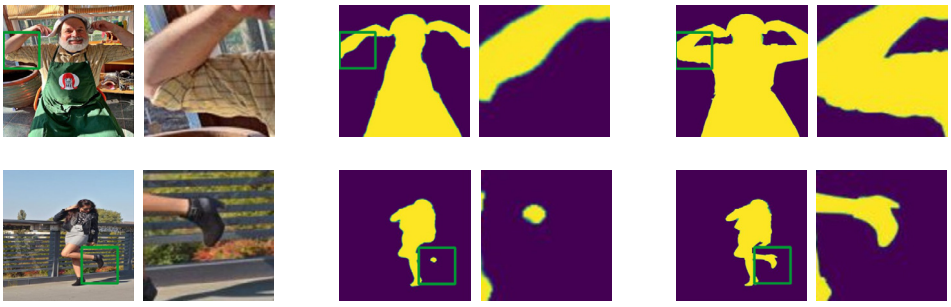
Module \ Dataset			PPM-100						
Local Block	Global Block	SLocal Block	SAD	MSE	MAD	Grad	Conn	SFG	SBG
Y	N	N	50.06	0.1816	0.1943	18.94	50.05	35.21	8.531
N	Y	N	52.57	0.2056	0.2068	19.64	51.08	37.57	14.58
N	N	Y	59..81	0.2882	0.3614	22.61	53.58	40.75	12.53
Y	Y	N	30.57	0.0247	0.1073	13.83	31.24	10.68	7.685
Y	N	Y	36.82	0.0581	0.1653	14.95	33.23	13.82	10.18
N	Y	Y	46.25	0.0762	0.1653	15.71	43.82	14.16	10.34
Y	Y	Y	3.288	0.0076	0.0125	6.418	2.990	0.8013	0.0102

5.2. Pseudo-label generation strategies

The pseudo-label generation strategy used for unlabeled data is added to several other existing methods for unlabeled data training. The generated pseudo label is used to assist the training of these no-label data, and the test effects without the pseudo label generation strategy and the test data using this strategy are recorded in Table 4. Unlabeled data means that only original images are used without labeling data. Labeled data means that both images and manually labeled mattes are needed. The unlabeled dataset used for ablation experiments mentioned in this subsection is P3M Test Set [24]. It contains a total of 1,000 images. We only use the images in the dataset and discard the labeled masks to simulate the effect that can be obtained on the unlabeled data. Experiments show that this pseudo-code generation strategy can improve the performance of unlabeled data. And this strategy has been validated in different architectures.

6. Conclusion

To improve the image matting effect with a small number of datasets, we build a semi-supervised network (called SemiTrans) based on the Transformer [1]. Transformer [1] has succeeded in



(d) Images (e) Ours w/o Global Block (f) Ours

Fig. 6. Visual comparisons for ablation studies.

Table 4
Comparison results with the pseudo-label generation strategy.

Dataset	P3M Test set											
	Before (the original algorithms)						After(algorithms with the strategy)					
Method	GFM	P3M	Mgm	Vitae	Vmf	Mfm	GFM	P3M	Mgm	Vitae	Vmf	Mfm
Year	2022	2021	2021	2022	2022	2022	2022	2021	2021	2022	2022	2022
SAD	66.50	30.74	35.21	23.75	11.54	6.304	25.74	13.54	16.51	8.643	3.913	3.107
MSE	0.2500	0.1071	0.1246	0.0898	0.1196	0.0329	0.1457	0.0183	0.0194	0.0061	0.0081	0.0068
MAD	0.2536	0.1172	0.1366	0.0917	0.1195	0.0387	0.1668	0.0172	0.0215	0.0090	0.0086	0.0069
Grad	20.22	33.31	28.42	29.91	12.62	6.174	13.27	18.67	16.28	14.38	4.102	3.918
Conn	64.86	36.89	32.06	22.41	11.15	6.256	22.36	14.75	10.07	8.044	3.902	3.196
SFG	54.79	13.04	10.63	8.635	4.805	6.164	20.03	5.483	3.581	2.091	0.8065	0.7926
SBG	2.436	6.303	14.81	10.62	2.684	1.196	1.443	0.1890	2.811	0.0132	0.4163	0.0225

capturing global features in other areas, but it has also proved inadequate to capture local features as rich as CNN. Therefore, we embed a portrait detailed decoding module in the network to make up for the local features lost by Transformer [1] and fuse them with the global features acquired by Transformer [1] based on CNN. In addition, a new strategy is proposed to generate pseudo-labels in the face of unlabeled data, and its effectiveness is proven by several groups of experiments. This generation strategy can adjust the results of network prediction by removing redundant foreground estimation and optimizing image edges. After comparing twenty-eight groups of seven measures across four different datasets, our architecture outperforms other methods in portrait matting. In addition, the effect of the proposed detailed decoding module and the pseudo-label generation strategy has also been verified by multiple ablation experiments.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

I have shared the link to my code at the manuscript. All code will be provided in the link when the paper is published.

Acknowledgment

This work was supported by the Natural Science Foundation of Shandong Province (No. ZR2019MF050) and the Shandong Province colleges and universities youth innovation technology plan innovation team project (No. 2020KJN011).

References

[1] A. Vaswani, N.M. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A.N. Gomez, L. Kaiser, I. Polosukhin, Attention is all you need, arXiv:1706.03762 [abs], 2017.
[2] Y. Xu, H. Wei, M. Lin, Y. Deng, K. Sheng, M. Zhang, F. Tang, W. Dong, F. Huang, C. Xu, Transformers in computational visual media: a survey, Comput. Vis. Media 8 (1) (2021) 33–62, https://doi.org/10.1007/s41095-021-0247-3.
[3] M. Chen, X. Zhao, B. Fu, L. Zhang, X. Xue, Rethinking local and global feature representation for semantic segmentation, in: BMVC, 2021.
[4] B. Chen, P. Li, C. Li, B. Li, L. Bai, C. Lin, M. Sun, J. Yan, W. Ouyang, GLiT: neural architecture search for global and local image transformer, in: 2021 IEEE/CVF International Conference on Computer Vision (ICCV), IEEE, 2021.
[5] V. Guimarães, V.S. Costa, Online learning of logic based neural network structures, in: Inductive Logic Programming, Springer International Publishing, 2022, pp. 140–155.
[6] Y. Li, H. Mao, R. Girshick, K. He, Exploring plain vision transformer backbones for object detection, arXiv preprint, arXiv:2203.16527, 2022.
[7] S. He, W. Liao, H.R. Tavakoli, M. Yang, B. Rosenhahn, N. Pugeault, Image captioning through image transformer, in: Proceedings of the Asian Conference on Computer Vision (ACCV), 2020.
[8] W. Xie, T. Huang, M. Wang, MNSRNet: multimodal transformer network for 3D surface super-resolution, in: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2022.
[9] K. Islam, Recent advances in vision transformer: a survey and outlook of recent work, arXiv preprint, arXiv:2203.01536, 2022.
[10] K. Han, Y. Wang, H. Chen, X. Chen, J. Guo, Z. Liu, Y. Tang, A. Xiao, C. Xu, Y. Xu, et al., A Survey on Vision Transformer, IEEE, 2022.
[11] W.F. Hendria, Q.T. Phan, F. Adzaka, C. Jeong, Combining transformer and CNN for object detection in UAV imagery, ICT Express (2022), https://doi.org/10.1016/j.icte.2021.12.006, in press, available online 28 December 2021.
[12] Z. Li, G. Chen, T. Zhang, A CNN-Transformer Hybrid Approach for Crop Classification Using Multitemporal Multisensor Images, vol. 13, Institute of Electrical and Electronics Engineers (IEEE), 2020, pp. 847–858.
[13] J. Ji, Y. Luo, X. Sun, F. Chen, G. Luo, Y. Wu, Y. Gao, R. Ji, Improving image captioning by leveraging intra- and inter-layer global representation in transformer network, Proc. AAAI Conf. Artif. Intell. 35 (2) (2021) 1655–1663, https://doi.org/10.1609/aaai.v35i2.16258.
[14] T. Xian, Z. Li, C. Zhang, H. Ma, Dual global enhanced transformer for image captioning, Neural Netw. 148 (2022) 129–141, https://doi.org/10.1016/j.neunet.2022.01.011.
[15] H. Yan, Z. Li, W. Li, C. Wang, M. Wu, C. Zhang, Contnet: why not use convolution and transformer at the same time?, arXiv preprint, arXiv:2104.13497, 2021.

[16] S. Ioffe, C. Szegedy, Batch normalization: accelerating deep network training by reducing internal covariate shift, in: International Conference on Machine Learning, in: PMLR, 2015, pp. 448–456.
[17] T. Kessler, G. Dorian, J.H. Mack, Application of a rectified linear unit, (ReLU) based artificial neural network to cetane number predictions, https://doi.org/10.31224/osf.io/7wmha.
[18] Y. Xu, B. Liu, Y. Quan, H. Ji, Unsupervised deep background matting using deep matte prior, IEEE Trans. Circuits Syst. Video Technol. 32 (7) (2022) 4324–4337, https://doi.org/10.1109/tcsvt.2021.3132461.
[19] S. Sengupta, V. Jayaram, B. Curless, S.M. Seitz, I. Kemelmacher-Shlizerman, Background matting: the world is your green screen, in: 2020 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2020.
[20] C. Rhemann, C. Rother, A. Rav-Acha, T. Sharp, High resolution matting via interactive trimap segmentation, in: 2008 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2008.
[21] Z. Yi, W. Song, S. Li, A. Hao, Automatic image matting and fusing for portrait synthesis, Sci. China Inf. Sci. 65 (2) (2022) 124101, https://doi.org/10.1007/s11432-021-3279-y.
[22] J. Wang, M. Cohen, An iterative optimization approach for unified image segmentation and matting, in: Tenth IEEE International Conference on Computer Vision (ICCV'05), Vol. 1, IEEE, 2005.
[23] J. Li, J. Zhang, S.J. Maybank, D. Tao, Bridging composite and real: towards end-to-end deep image matting, Int. J. Comput. Vis. 130 (2) (2022) 246–266, https://doi.org/10.1007/s11263-021-01541-0.
[24] J. Li, S. Ma, J. Zhang, D. Tao, Privacy-preserving portrait matting, in: Proceedings of the 29th ACM International Conference on Multimedia, ACM, 2021.
[25] Q. Yu, J. Zhang, H. Zhang, Y. Wang, Z. Lin, N. Xu, Y. Bai, A. Yuille, Mask guided matting via progressive refinement network, in: 2021 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2021.
[26] S. Ma, J. Li, J. Zhang, H. Zhang, D. Tao, Rethinking portrait matting with privacy preserving, arXiv preprint, arXiv:2203.16828, 2022.
[27] Y. Xu, B. Sun, X. Yan, J. Hu, M. Chen, Multi-focus image fusion using learning based matting with sum of the Gaussian-based modified Laplacian, Digit. Signal Process. 106 (2020) 102821, https://doi.org/10.1016/j.dsp.2020.102821.
[28] I. Molodetskikh, M. Erofeev, A. Moskalenko, D. Vatolin, Temporally coherent person matting trained on fake-motion dataset, Digit. Signal Process. 126 (2022) 103521, https://doi.org/10.1016/j.dsp.2022.103521.
[29] C.P. Yanti, I.G. Andika, HSV image classification of ancient script on copper Kintamani inscriptions using GLRCM and SVM, J. Tekn. Sist. Komput. 8 (2) (2020) 94–99, https://doi.org/10.14710/jtsiskom.8.2.2020.94-99.
[30] N. Murray, F. Perronnin, Generalized max pooling, in: 2014 IEEE Conference on Computer Vision and Pattern Recognition, IEEE, 2014.
[31] D.C. Lepcha, B. Goyal, A. Dogra, Image matting: a comprehensive survey on techniques, comparative analysis, applications and future scope, https://doi.org/10.1142/S0219467823500110, 2020.
[32] J. Li, V. Goel, M. Ohanyan, S. Navasardyan, Y. Wei, H. Shi, Vmformer: end-to-end video matting with transformer, arXiv:2208.12801 [abs], 2022.
[33] G. Park, S. Son, J. Yoo, S. Kim, N. Kwak, MatteFormer: transformer-based image matting via prior-tokens, in: 2022 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR), IEEE, 2022.
[34] Z. Ke, J. Sun, K. Li, Q. Yan, R.W. Lau, MODNet: real-time trimap-free portrait matting via objective decomposition, Proc. AAAI Conf. Artif. Intell. 36 (1) (2022) 1140–1147, https://doi.org/10.1609/aaai.v36i1.19999.
[35] supervise.ly. supervisely person dataset. supervise.ly, 2018.
[36] N. Xu, B. Price, S. Cohen, T. Huang, Deep image matting, in: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, 2017, pp. 2970–2979.
[37] X. Shen, X. Tao, H. Gao, C. Zhou, J. Jia, Deep automatic portrait matting, in: Computer Vision – ECCV 2016, Springer International Publishing, 2016, pp. 92–107.
[38] Z. Zhang, Improved Adam optimizer for deep neural networks, in: 2018 IEEE/ACM 26th International Symposium on Quality of Service (IWQoS), IEEE, 2018.

Xinyue Zhang is pursuing master degree in computer technology from Qingdao University. She has participated in multiple target detection and tracking projects and is familiar with target detection, target tracking, image recovery, 3D reconstruction, human pose estimation related algorithms. Her current research interests focus on computer vision problems and deep learning.

Changxin Gao received the PH.D degree in pattern recognition and intelligent systems from the Huazhong University of Science and Technology in 2010. He is currently an Associate Professor with the School of Artificial Intelligence and Automation, Huazhong University. His research interests are recognition and surveillance video analysis. He served as a member of IEEE, a member of China image and graphics society of youth work committee communications committee member, China society of imaging detection and perception of image and graphics, image and graphics society, deputy secretary-general of branch of visual data of members, the branch of computer vision committee member of China computer society, the hybrid intelligence branch of China automation society member.

Guodong Wang received the PH.D degree in pattern recognition and intelligent systems from the Huazhong University of Science and Technology in 2008. He is currently an Associate Professor with the College of Computer Science and Technology Qingdao University. His research interests are artificial intelligence, variational image science, deep learning, face recognition, intelligent video surveillance and analysis, 3D reconstruction, etc. He is the reviewer of several SCI international journals, such as IEEE Transactions on Image Processing, IEEE Transactions on Systems, Man and Cybernetics - Part A: Systems and Humans, International Journal of Pattern Recognition and Artificial Intelligence, Neurocomputing, IET Image Processing, Mathematical Problems in Engineering, Multimedia Tools and Applications, Abstract and Applied. He presided over one project of the National Natural Science Foundation of China, one project of the Natural Science Foundation of Shandong Province, one project of Qingdao Science and Technology Development Plan, and participated in many projects including the National Key Basic Research Program of China ("973 Program"), the National Natural Science Foundation of China, and the National Science and Technology Support Program. He has published more than 100 academic papers, including more than 20 SCI papers and more than 70 EI papers. He has obtained three invention patents.

Nong Sang graduated from Huazhong University of Science and Technology and received his BE degree in computer science and engineering in 1990, MS degree in pattern recognition and intelligent control in 1993, and PhD degree in pattern recognition and intelligent systems in 2000. He is currently a professor at the School of Artificial Intelligence and Automation, Huazhong University of Science and Technology, Wuhan, China. His research interests include object detection and recognition, object tracking, image/video semantic segmentation, intelligent processing and analysis of surveillance videos.

Hao Dong is a postgraduate student at Qingdao University. His major is computer science and technology, and his research direction is texture recognition. He has participated in many AI projects based on deep learning, including but not limited to object detection, instance segmentation, texture recognition, and so on. At present, his research interests are texture recognition and human pose recognition.