ISSN 2616-5775 Vol. 6, Issue 11: 60-71, DOI: 10.25236/AJCIS.2023.061109

Image Matting Algorithm Using Deep Generative Adversarial Networks

Yanlong Xu[1,a], Limin Sun[1,b], Qian Guo[1,c,*]

[1]*School of Information and Intelligent Engineering, University of Sanya, Sanya, 572022, China*
[a]*yanlongxu@sanyau.edu.cn*, [b]*liminsun@sanyau.edu.cn*, [c]*qianguo@sanyau.edu.cn*
[*]*Corresponding author*

Abstract: *In recent years, Generative Adversarial Network (GAN) has been widely used in the field of computer vision due to its superior performance. Influenced by AlphaGAN, we propose U²Net-GAN for image matting algorithm, which is the basis of image synthesis technology. The purpose of matting algorithm is to obtain alpha matte of the foreground in a natural image. In practical applications, natural images may have similar foreground and background and more complex textures, and matting algorithms need to accurately extract a large number of detailed features of images. In order to solve the above problems, we use U²Net as generator of GAN. The generator U²Net cascades multi-layer depth features to accurately extract high-level semantic features of image. Self-Calibrated convolutions (SC) replaces standard convolution of the Residual Ublocks(RSU), which improves performance of the network to extract image detail features without significantly increasing the amount of calculation. And we use PatchGAN discriminator to train with the expanded public datasets and evaluate it on the benchmark dataset. Experiments show that our algorithm achieves the best results in both quantitative and qualitative aspects compared with other algorithms.*

Keywords: *Image Matting, GAN, PatchGAN, RSU, U²Net, Self-Calibrated Convolutions*

1. Introduction

A good matting algorithm can accurately solve alpha matte of foreground objects(regions of interest) in images or video sequences. With the latest development of mobile technology, both professional users and ordinary users need high-quality matting algorithms to complete image synthesis tasks. As shown in (1), each pixel of input image can be modeled as a convex combination of foreground and background colors[1]. Among the equation, I_z is observed color value of pixel z, F_z is foreground color value of pixel z, B_z is background color value of pixel z. The α_z represents the weight of F_z and B_z in pixel z, which is an arbitrary real number in [0, 1], and its physical meaning is the opacity of pixels corresponding to foreground objects.

$$I_z = \alpha_z F_z + (1-\alpha_z) B_z \tag{1}$$

When α_z takes 0, pixel z is background. When α_z takes 1, pixel z is foreground. When α_z takes [0, 1], the pixel z is translucent, which is a mixed pixel. For a RGB image, the three channels of red, green and blue are substituted into the (1) respectively.

$$\begin{aligned} I_z^r &= \alpha_z F_z^r + (1-\alpha_z) B_z^r \\ I_z^g &= \alpha_z F_z^g + (1-\alpha_z) B_z^g \\ I_z^b &= \alpha_z F_z^b + (1-\alpha_z) B_z^b \end{aligned} \tag{2}$$

Through the (2), it is known that the matting problem is serious underconstrained. Each pixel has only three known quantities, namely I_z^r, I_z^g and I_z^b, which are intensity values of the pixel z in three channels, respectively can be obtained by observing the input image. There are seven unknown quantities, namely , which are much larger $F_z^r, F_z^g, F_z^b, B_z^r, B_z^g, B_z^b$ than the number of equations. Therefore, the user needs to provide prior information to help solve the α_z. Common prior information can be provided by trimap [2] and Subscribes [3]. This paper provides prior information through the trimap.

In the trimap of Figure.1(b), the white area is determined foreground, black area is determined

ISSN 2616-5775 Vol. 6, Issue 11: 60-71, DOI: 10.25236/AJCIS.2023.061109

background, gray part is mixed pixel to be solved. Image matting algorithm generally focuses on the gray's unknown area and solves the α value of the area.

Figure 1: Input image(left); Trimap(right) .

Traditional matting algorithms are mainly divided into two categories according to the different calculation methods of value, which are propagation-based and sampling-based matting algorithm. The propagation-based matting [5, 6,8, 9] method assumes that the image has local smoothness or local continuity, and has poor performance in natural images with discontinuities such as holes. The sampling based matting algorithm [2,4] develops the sampling points from local sampling to global sampling through the statistical hypothesis obtained by the color value of the marked area, which increases the computational complexity. Both sampling-based and propagation-based matting algorithms need to be based on certain assumptions, and the α value of each pixel is calculated in isolation. This approach does not fully take into account the correlation between adjacent pixels, and is also easily disturbed by sample selection. The weaker the assumptions, the wider the image range that the matting algorithm can solve. If the assumptions in the image are not true, the matting result will be very bad.

Traditional matting algorithms rely heavily on low-level color and texture feature of the image, which makes these algorithms unable to effectively process details such as hair, holes, and foreground edges in the image. Deep learning solves many problems in computer vision. The matting algorithms based on deep learning use deep neural network to extract image features, which solves the shortcomings of traditional matting algorithms that rely too much on lowlevel features of images to a certain extent, and improves the efficiency and generalization performance of matting. However, due to the neglect of the multi-level connection of deep features and multi-scale fusion, these algorithms still have large errors when processing some images with complex structure and rich details.

Therefore, a matting algorithm based on Generative Adversarial Network(GAN) is proposed for the existing problems of natural image matting algorithm. We use U^2Net network, the variant of UNet, as the generator of GAN, which can fully extract local features of different scales by cascading multilayer deep features. Because the traditional random noise will make the GAN too 'free' in the training process, it is difficult to train the network or even difficult to converge. Therefore, the random noise as input of generator is abandoned, and the four-dimensional matrix spliced by trimap and RGB image is used as input of generator. This can not only fully utilize prior information of foreground and background in the trimap and the low-level feature in original RGB image, but also add constraints in a good direction to generator, so that generator can output satisfactory alpha matte on the basis of constraints of the input. In the network, we use Self-Calibrated convolutions(SC)[7] instead of standard convolution. SCNet can further enhance the network's ability to extract image depth feature without markedly increasing the amount of calculation. Experiments show that our algorithm can receive more competitive matting results.

2. Related Work

The early representative works of the propagation-based matting algorithm are the KNN matting[8] and local & non-local criteria[29] matting algorithm. The former utilizes the K-nearest neighbor algorithm to select similar pixels, which is enlightening to the design of non-local criterion transparency propagation form. The latter proposes the idea of combining local and non-local criteria for transparency propagation. The Poisson matting algorithm[5] based on local criteria assumes that foreground and background colors of mixed pixels and adjacent pixels have a specific relationship, such as the two are equal or the former is a linear combination of the latter. The α value is propagated according to the assumption between adjacent pixels. In the propagation-based method, the propagation criterion between pixels is designed, and the affinity model between pixels is defined to measure the similarity, so as to achieve the purpose of spreading from known region to unknown region. Nevertheless, for images with discontinuous target objects (such as objects with holes), the matting effect cannot achieve satisfactory results. For high resolution images, longer processing time is required, which also becomes a limiting factor for the further development of such algorithms.

ISSN 2616-5775 Vol. 6, Issue 11: 60-71, DOI: 10.25236/AJCIS.2023.061109

Sampling-based matting algorithms mainly include KL and Bayesian algorithms. Mishima[10] et al. first proposed image matting by sampling representative foreground and background colors and calculating its statistical rules. Ruzon et al.[4] analyzed the statistical distribution of foreground and background samples to accurately calculate alpha matte. Bayesian matting[2] method raises questions in a well defined Bayesian framework and uses maximum a posteriori estimation to complete matting. The 'belief propagation' matting method[9] and the 'easy matting' method[11] directly solve the foreground region from some user-specified subscribes without using the Trimap as the common input. However, in this case, the user input is very sparse, so the resulting color samples are not sufficient. Shahrian et al.[12] proposed a comprehensive sampling algorithm using Gaussian model. The accuracy of the obtained foreground semi-transparency is further improved. Johonson et al.[13] cluster the known regions into superpixels and form a sparse coding dictionary from the perspective of pixel coding, so the unknown part sparse coding can be solved in a short time. Karacan et al.[14] proposes a matting algorithm based on KL divergence. The algorithm not only samples from the global, improves the diversity of the selected samples, but also controls the size of the sample set to a certain extent. In the sampling-based algorithm, different sampling methods are set to expand the sampling area, enrich the diversity of the collected samples, and find the best foreground-background sample pair for unknown pixels. However, sampling robustness cannot be guaranteed by using artificially designed sampling criteria.

The matting algorithm using deep learning can often get more accurate results. Xu et al.[15] expanded dataset through image fusion, and introduced the En-Decoder network structure to predict alpha matte of image foreground, and used refinement network to deal with the over-smoothing problem of the prediction graph. Hou et al.[16] used dual encoders to predict semi-transparency and foreground images, thus completely solving the task of matting. Lutz et al.[17] introduced the adversarial neural network into the matting task, and proposed AlphaGAN matting algorithm. The generator and discriminator were set up respectively. And the skip connection in En-Decoder of the generator can enhance the network 's perception of local detail information. Cai et al.[18] added the step of reprocessing the trimap when predicting the semi-transparency of the foreground. Its advantage is that it has a good perception of the details and structural features of the foreground, and has a certain correction effect on the manually labeled trimap. Background Matting[19]Using background images, segmentation results and continuous frames as prior information, good results have been achieved on video matting. MG Matting[20] uses a rough segmentation graph as an auxiliary graph input to the network for prediction. HAtt Matting[21] introduces attention mechanism to solve the problem of semantic asymmetry in deep networks, and combines high-level semantic features and low-level texture features to jointly improve prediction details. MoDnet[22] decomposes the matting task into subtasks to achieve semantic estimation and detail prediction respectively, and then uses the fusion branch to complete the prediction task, but ignores the correlation between different sub-tasks.

Matting algorithm based on deep learning can obtain depth feature information of the image through the deep neural network, and has a better expression of the deep features of the image, which improves the efficiency and generalization performance of the matting. However, these matting algorithms often ignore the connection of multi-level deep features and the fusion of multi-scale deep features in the network model, resulting in the inability to accurately predict the alpha matte of the image when dealing with some images with complex background structures. To address the problems mentioned above, we propose a matting algorithm based on GAN.

3. Model Construction

The GAN includes two sections: generator and discriminator. The U^{2}Net is used as the generator of the GAN. And we use Self-Calibrated convolutions module to replace standard convolution block of RSU. The generator uses RGB image and its trimap as input. We use PatchGAN as discriminator to output the predicted alpha matte.

3.1. The Generator of GAN

We use the U^{2}Net as generator, as shown in Figure.2. The generatoris generally an U-shaped network, where encoder of each layer is an UNet network. The feature map from each encoder UNet is input into a symmetric decoder UNet to cascade multi-level depth features. And then we can extract multiscale features of image. The dilated convolution operation in network can promote the network to capture the local and non-local feature in image to the greatest extent.

ISSN 2616-5775 Vol. 6, Issue 11: 60-71, DOI: 10.25236/AJCIS.2023.061109

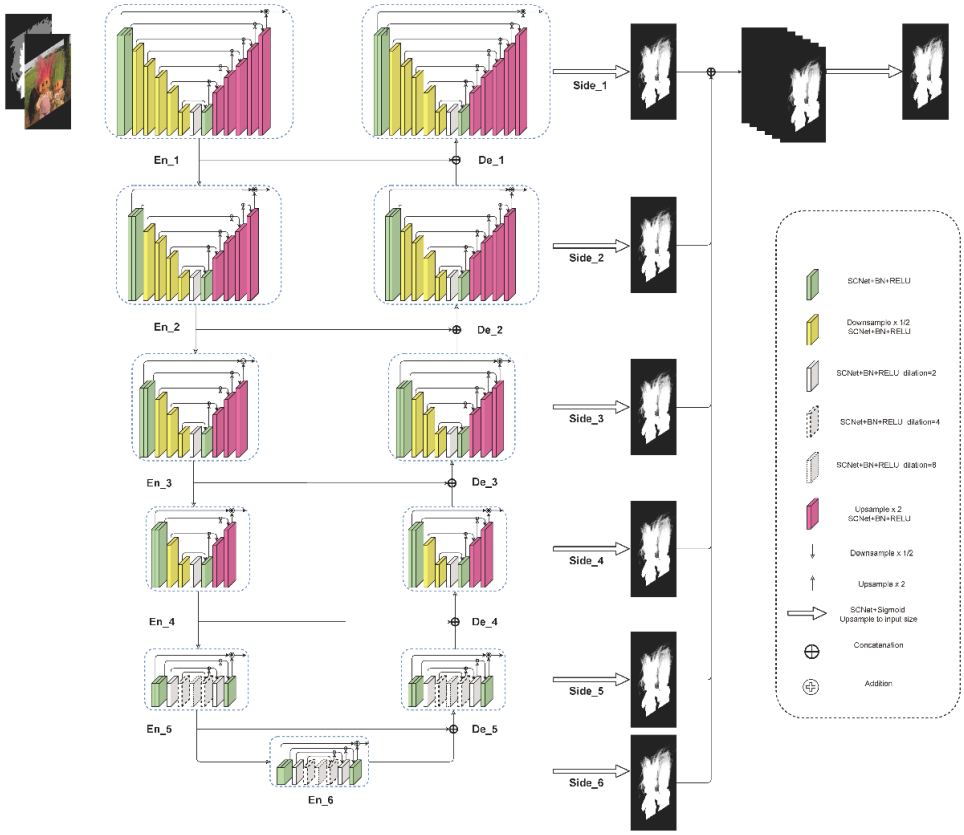

Figure 2: The generator of GAN.

Figure.3 shows the common deep neural network models, such as ResNet[23], DenseNet[24], VGG [25],etc. So as to decrease the amount of parameters and improve the computational efficiency, a smaller convolution kernel is used to extract local feature. However, due to the small receptive field, global features of image cannot be effectively captured. Therefore, RSU(ReSidual Ublock) block is used to replace the naive convolution block of UNet.

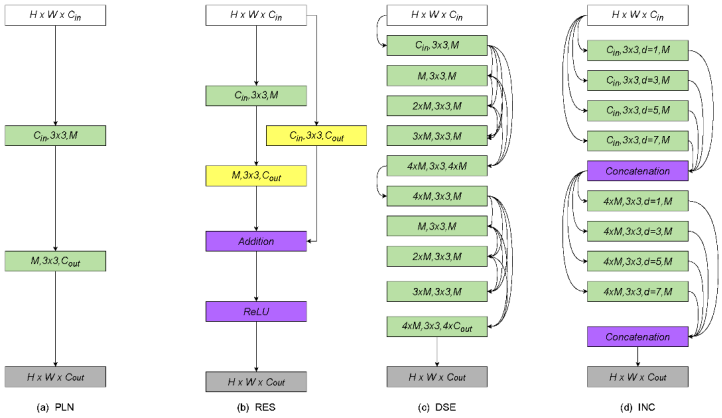

Figure 3: Common convolution modules: (a)Plain Convolution module (b)Residual Convolution module (c)Dense Convolution module (d)Inception Convolution module.

In Figure.4, L is the amount of network layers of RSU blocks, C_{in} and C_{out} is the amount of input and output channels of the RSU module, respectively. M means the amount of characteristic channels of RSU blocks. RSU block has three main features:

▪ Initialize the ordinary convolution of the input. Input feature map size is $x(H \times W \times C_{in})$, and output feature map is $F_1(x)$ with C_{out} channel.

▪ It The RSU realizes the mixing of different receptive field feature maps through a symmetric En-Decoder structure with layers, and captures global information at different scales. And using the residual

ISSN 2616-5775 Vol. 6, Issue 11: 60-71, DOI: 10.25236/AJCIS.2023.061109

idea, it is well known that ResNet can do identity mapping at least two layers, otherwise it is an invalid linear transformation. The Block in the RSU itself contains several layers, so it only needs to cross one Block to be the complete residual block. In terms of calculation amount, the calculation amount of RSU is linearly related to the increase of depth, but the correlation coefficient is small, so the calculation amount will not increase too much even if the network stacks deep.

▪ Concatenation operation. Splicing $F_1(x)$ and $U(F_1(x))$ aggregate local information and multi-scale information of depth features.

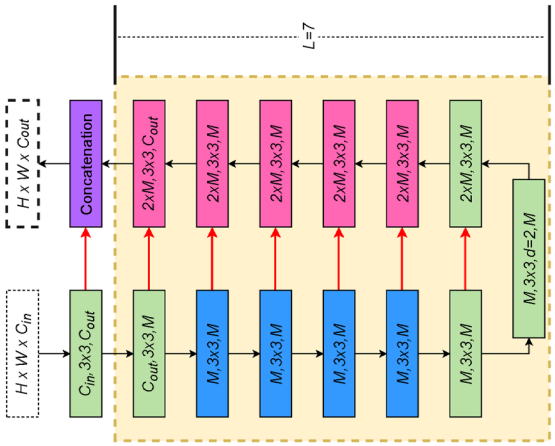

Figure 4: RSU block.

We use Self-Calibrated convolutions instead of the ordinary convolution of RSU blocks. We can understand the structure of SCNet through Figure.5. An advantage of Self-Calibrated convolutions is that each pixel in feature space has information of its nearby area and the interaction information between the channels, which can further extract image depth feature without obvious growth in amount of calculation. Then, by adjusting the parameter L, multi-scale features are extracted from the feature map of stepwise downsampling, and upsampling is performed by the feature map generated during the cascade downsampling process. This step effectively avoids the problem of detail loss caused by large-scale upsampling.

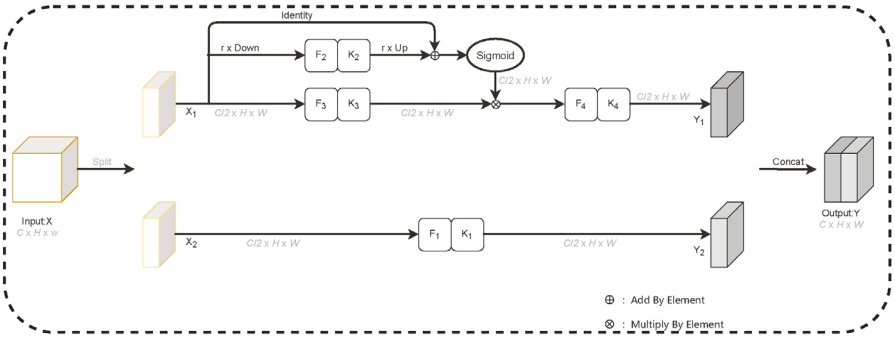

Figure 5: Self-Calibrated convolutions operation.

3.2. The Discriminator of GAN

We use PatchGan as the discriminator of the GAN, as shown in Figure.6. The discriminator divides the input image into $N*N$ blocks, and then maps it into a matrix X with a size of $(N*N)$ by full convolution operation. In this experiment, the size of is 30. Finally, final discriminant result is received by averaging matrix X. Each value in the matrix X corresponds to the discriminant value of the discriminator for each block in the input image. By averaging the matrix X as the final discriminant result, the network focus on the details of the image during training. Because the prediction of α needs to link the loss value with the details of the image. A good matting algorithm can effectively deal with extremely detailed areas such as hair and edges in the image, so it is appropriate to use PatchGan as the discriminator of the GAN in our paper.

In order to enable the discriminator to focus on the mixed pixels of foreground and background and

ISSN 2616-5775 Vol. 6, Issue 11: 60-71, DOI: 10.25236/AJCIS.2023.061109

guide the generator to generate more accurate α values. We use the alpha matte map generated by the generator to synthesize a new image by combining it with a new background. The input of the discriminator is a four-channel matrix consists of the synthesized image and the trimap. The introduction of the fourth-channel trimap can help the network to focus on the uncertain region of the image.

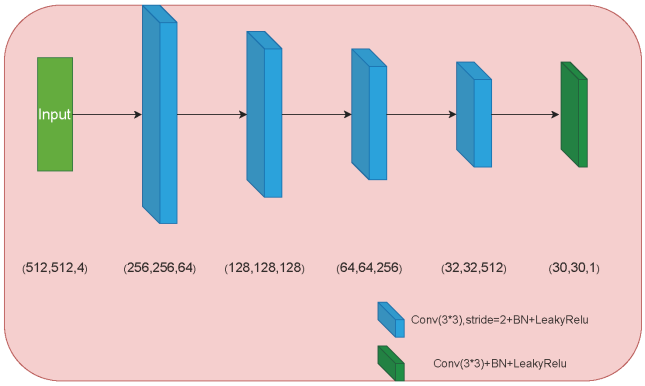

Figure 6: The discriminator of GAN.

3.3. Loss Function

In loss function of the network, we introduce α loss L_α, component loss L_c and adversarial loss L_{GAN} to train the GAN:

$$L_{total}(G,D) = L_\alpha(G) + L_C(G) + L_{GAN}(G,D) \tag{3}$$

The α loss function is defined as:

$$L_\alpha = \sum_i^n \sqrt{(\alpha^i{}_p - \alpha^i{}_{gt})^2 + \varepsilon^2} \quad \alpha^i{}_p, \alpha^i{}_{gt} \in [0,1] \tag{4}$$

Amoun them, α^i_p and $\alpha^i{}_{gt}$ denote the $i-th$ pixel's α value in predicted alpha matte and groundtruth alpha matte, respectively. And ε is the regularization term.

Component loss L_c is defined as:

$$L_c = \sum_j^m \sum_i^n \sqrt{(c^i{}_{j,p} - c^i{}_{j,gt})^2 + \varepsilon^2} \tag{5}$$

Among them, m denotes the amount of channels in synthesized image. n denotes the amount of pixels. $c^i_{j,p}$ denotes the i-th pixel's intensity value in the j-th channel in composite image of the predicted alpha matte and the new background. And $c^i_{j,gt}$ denotes the i-th pixel's intensity value in the j-th channel in composite image of the groundtruth alpha matte and the new background.

The adversarial loss L_{GAN} is defined as:

$$\min_G \max_D L_{GAN}(G,D) = \log D(x) + \log(1 - D(C(G(x)))) \tag{6}$$

Amoun them, x is a four-dimensional matrix, which is composed of the groundtruth alpha matte, image and trimap. G(x) is generator that tries to generate a near-real alpha matte. C(y) is synthesis function that uses predicted alpha matte generated by G(x) and new background image to synthesize a new image. While D as a discriminator will try to distinguish between real input images and synthetic images. D(x) makes L_{GAN} larger. Therefore, the generator G(x) needs to constantly adjust itself to minimize L_{GAN} to offset the impact of D(x). The above loss function constitutes the target loss function $L_{total}(G,D)$ of our network.

ISSN 2616-5775 Vol. 6, Issue 11: 60-71, DOI: 10.25236/AJCIS.2023.061109

4. Experimental Results

We conducted this experiment on a server with eight GeForce RTX 2080Ti graphics cards. So as to settle the underfitting problem caused by insufficient training data, this experiment randomly selects 1000 images from the Pascal VOC [26] dataset as the background, and fuses with the sample images provided by the benchmark dataset [27] as the foreground. The dataset of this paper is amplified by this method. The amplified dataset contains a total of 27000 images, and the synthesized images are shown in Figure.7. Figure.8 shows partial test set used in this experiment.

We enlarged the details of the image to show the test results more intuitively. To prove the effectiveness of our matting algorithm, qualitative and quantitative methods are used to evaluate and analyze the experimental results. Different matting algorithms such as Bayesian [2], CF [6], KL [14], KNN[8], Poisson [5] are compared with our matting algorithm.

Figure 7: Some images of this dataset.

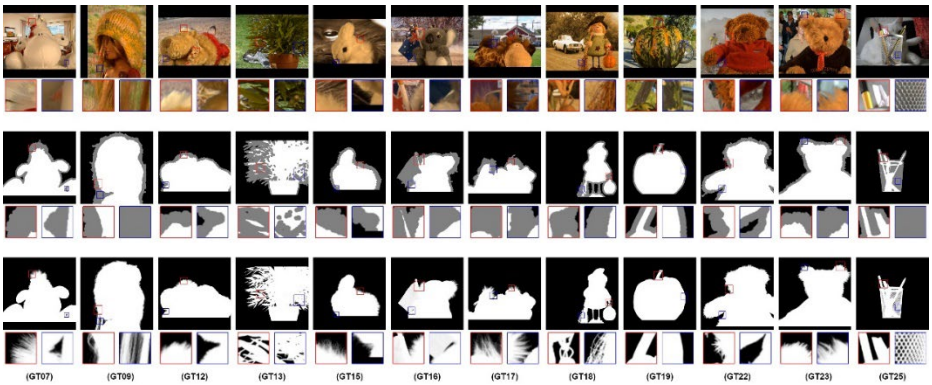

Figure 8: The first row is input RGB image, the second row is its trimap, and the third row shows the groundtruth alpha matte corresponding to the RGB image. The blue box and red box in the image are the magnification of the image detail area.

4.1. Quatitative Analysis

The quantitative analysis of this experiment uses RMSE(Root Mean Square Error) and SSIM(Structural Similarity)[28] as evaluation indicators. SSIM is based on the structure, brightness and contrast of the image. It is an image quality evaluation method that conforms to human vision. Equations (7) defines SSIM.

ISSN 2616-5775 Vol. 6, Issue 11: 60-71, DOI: 10.25236/AJCIS.2023.061109

$$
\begin{aligned}
u_x &= \frac{1}{R\times C}\sum_{p=1}^{R}\sum_{q=1}^{C}X(p,q)\\
u_y &= \frac{1}{R\times C}\sum_{p=1}^{R}\sum_{q=1}^{C}Y(p,q)\\
\sigma_x^2 &= \frac{1}{R\times C-1}\sum_{p=1}^{R}\sum_{q=1}^{C}(X(p,q)-u_x)\\
\sigma_y^2 &= \frac{1}{R\times C-1}\sum_{p=1}^{R}\sum_{q=1}^{C}(Y(p,q)-u_y)\\
\sigma_{xy} &= \frac{1}{R\times C-1}\sum_{p=1}^{R}\sum_{q=1}^{C}(X(p,q)-u_x)(Y(p,q)-u_y)\\
SSIM(X,Y) &= \frac{(2u_xu_y+C_1)(2\sigma_{xy}+C_2)}{(u_x^2+u_y^2+C_1)(\sigma_x^2+\sigma_y^2+C_2)}
\end{aligned}
\tag{7}
$$

Among them, X represents predicted alpha matte, Y represents groundtruth alpha matte of image, R and C represent the rows and columns of alpha matte, u_x and u_y represent the mean of predicted alpha matte and groundtruth alpha matte, σ_x^2 and σ_y^2 represent the variance of predicted alpha matte and groundtruth alpha matte, while σ_{xy} represents their covariance. C1 and C2 are constants to avoid the denominator becoming 0. The value of SSIM is positive correlation to the predicted alpha matte.

The role of RMSE is to measure the similarity between predicted image and groundtruth image. The RMSE value is inversely proportional to the image similarity and matting quality. Equations (8) is the definition of RMSE:

$$
\sqrt{\frac{1}{N}\sum_{z}^{N}(\widehat{\alpha_z}-\overline{\widetilde{\alpha_z}})^2} \tag{8}
$$

Among the equation, $\widehat{\alpha_z}$ represents predicted alpha matte, $\widetilde{\alpha_z}$ represents groundtruth alpha matte of the image, z represents the pixel index. And N is the amount of pixels in the image.

Table 1 reveals the quantitative evaluation results of Baysian, CF, KL, KNN, Poisson and our matting algorithms on RMSE and SSIM indicators. In Table 1, we can find that our algorithm has the best matting results on every image except image GT17.

Table 1: Baysian, CF, KL, KNN, Poisson and our matting algorithms are evaluated on RMSE and SSIM respectively

	Baysian		CF		KL		KNN		Poisson		Our	
	RMSE	SSIM	RMSE	SSIM	RMSE	SSIM	RMSE	SSIM	RMSE	SSIM	RMSE	SSIM
GT07	37.472	0.901	14.889	0.962	13.304	0.957	19.865	0.937	31.095	0.917	10.852	0.979
GT09	24.172	0.891	19.25	0.926	17.327	0.918	24.934	0.909	38.337	0.862	14.536	0.948
GT12	23.812	0.948	13.788	0.969	14.388	0.964	19.685	0.96	23.118	0.949	12.204	0.978
GT13	27.5	0.898	26.822	0.911	15.587	0.952	19.484	0.945	45.237	0.809	13.933	0.958
GT15	27.498	0.933	13.708	0.964	15.803	0.955	18.846	0.956	21.087	0.947	9.712	0.975
GT16	64.548	0.851	32.699	0.934	26.428	0.929	36.866	0.924	49.173	0.863	9.144	0.959
GT17	28.475	0.939	11.996	0.966	11.93	0.964	20.785	0.951	32.217	0.917	15.046	0.962
GT18	37.334	0.903	11.411	0.972	10.532	0.965	15.559	0.955	27.442	0.923	8.339	0.976
GT19	18.071	0.961	7.331	0.981	7.511	0.979	10.455	0.97	22.963	0.952	5.112	0.995
GT22	24.976	0.926	7.952	0.98	7.889	0.977	11.291	0.969	23.981	0.911	7.147	0.981
GT23	30.681	0.92	13.362	0.962	12.903	0.956	20.438	0.938	34.639	0.902	9.94	0.971
GT25	36.149	0.887	20.972	0.93	20.774	0.934	21.647	0.914	38.088	0.904	18.511	0.941

Figure.9 and Figure.10 show the quantitative evaluation results of RMSE and SSIM of our algorithm and other matting algorithms on different test samples in the form of line charts. The lines of different color represent quantitative evaluation values of different algorithms. Our algorithm is represented by a red line.

By observing the Table 1, Figure.9 and Figure.10, we can find that U^{2}Net-GAN keeps good performance when dealing with images of different scenes. From the above figures and table, we can see

ISSN 2616-5775 Vol. 6, Issue 11: 60-71, DOI: 10.25236/AJCIS.2023.061109

that experimental results obtained by our matting algorithm are slightly inferior to the CF algorithm on the RMSE when dealing with the image GT17 of the Figure.8. However, by comparing the predicted alpha matte of image GT17 obtained by different matting algorithms in the Figure.12 with the groundtruth alpha matte of image GT17 in the Figure.8, it can be found that our experimental results are close to the groundtruth alpha matte. Although the quantitative evaluation results are slightly insufficient, on the whole, our algorithm U^{2}Net-GAN can achieve relatively good results compared with other matting algorithms. It can be concluded that our matting algorithm is more stable and robust.

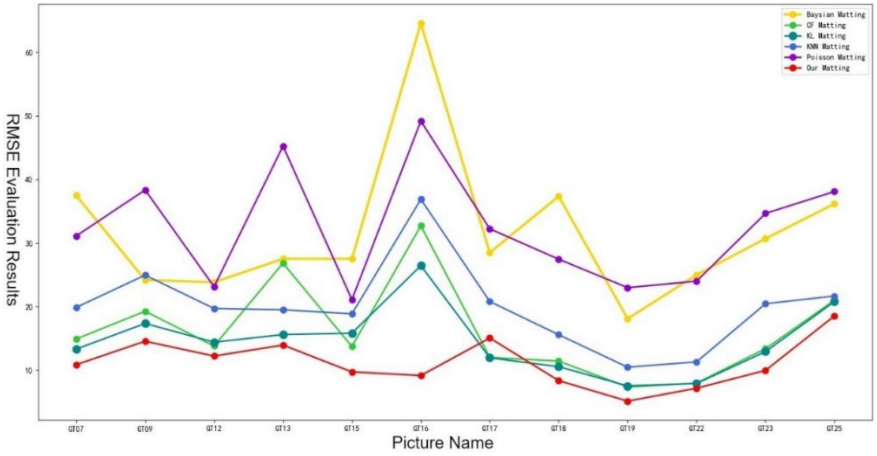

Figure 9: The comparison results of different matting algorithms on RMSE when dealing with different test samples.

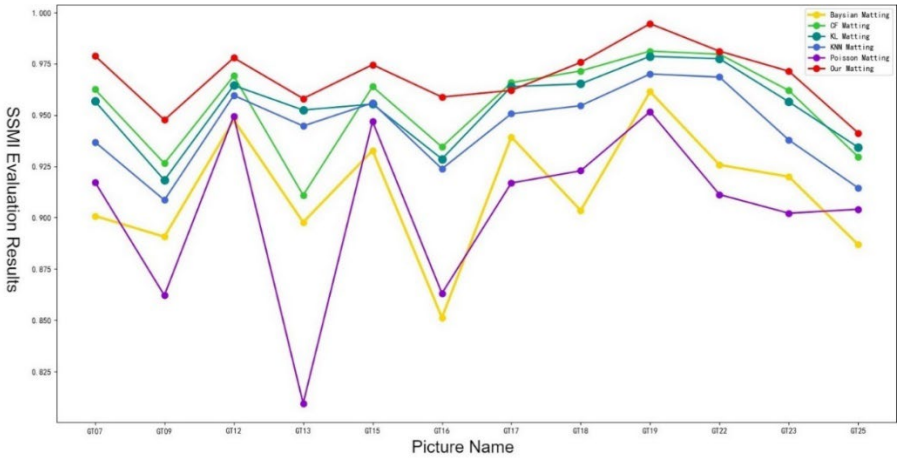

Figure 10: The comparison results of different matting algorithms on SSMI when dealing with different test samples.

4.2. Quantitative Analysis

As shown in Figure.11, Figure.12, our algorithm is more accurate than other algorithms in dealing with details such as hair and holes. When dealing with the hair of girls in image GT09 and the hair of rabbits in image GT15, this algorithm can extract hair information more accurately. When dealing with the holes between the leaves of image GT13, our algorithm and KNN matting algorithm get relatively accurate results, indicating that our algorithm is effective in dealing with the holes in foreground target. From the experimental results of image GT18 in Figure.12, we can see that the algorithm U^{2}Net-GAN also has a good effect on images with similar background colors. All the above results prove that the generator of GAN can extract the local detail features of the image more effectively by using U^{2}Net. When dealing with the flag of image GT17, the groundtruth alpha matte is semi-transparent, but the mask obtained by our algorithm is completely opaque. We can find that our algorithm has poor performance in the prediction of this kind of image alpha matte.

When processing image GT18, our algorithm and CF algorithm get more accurate results, but U^{2}Net-GAN algorithm performs better on the small holes of the handbag in the red box of the image. When processing image GT19 with similar foreground and background structure, our algorithm is more precise

ISSN 2616-5775 Vol. 6, Issue 11: 60-71, DOI: 10.25236/AJCIS.2023.061109

and accurate than other algorithms. When processing image GT25, the processing of transparency is more accurate, eliminating the interference of background pixels. In general, although our matting algorithm is insufficient, it can still obtain competitive results compared with other matting algorithms.

5. Conclusions

We proposes a matting algorithm based on GAN in this paper. Using U^{2}Net as the generator of GAN, it can effectively cascade the deep features and multi-scale features extracted by the network. Meanwhile, we use the Self-Calibrated convolutions instead of the standard convolution of the RSU to obtain the channels information of each pixel and its interaction with the nearby area without obvious grouth in the amount of calculation, and to facilitate the network to extract depth features of input image. Our algorithm has good performance in processing images with similar background colors, images with complex structures and tiny details in images. From the above experimental results, we can see that our matting algorithm is better than the traditional matting algorithm.

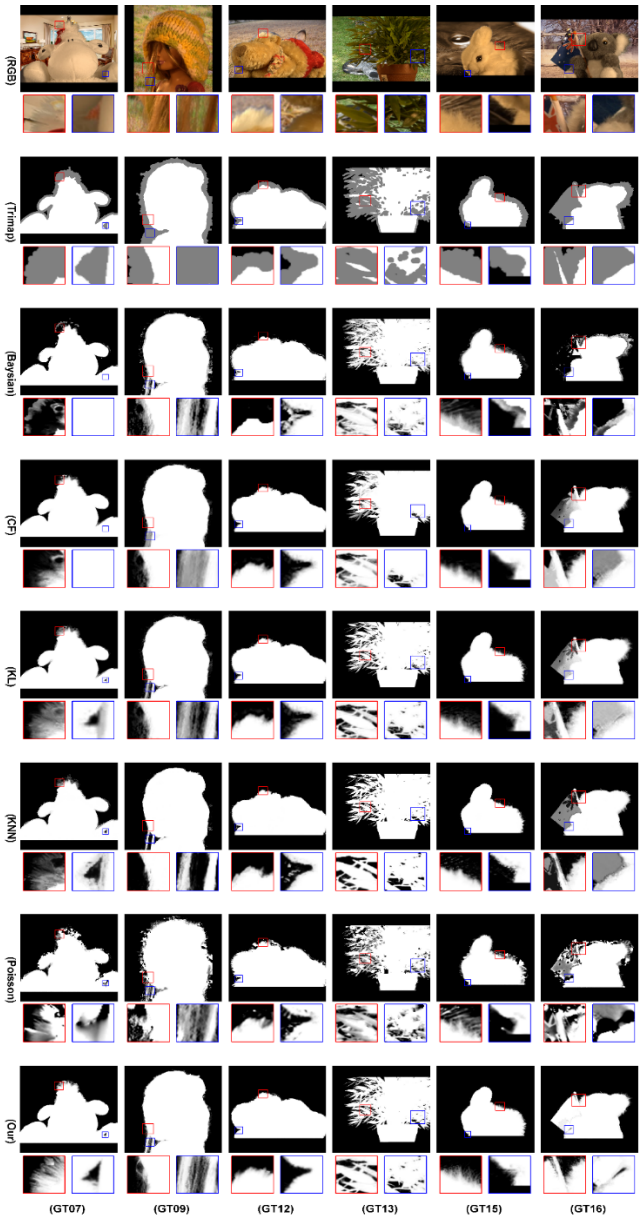

Figure 11: The alpha matte of different matting algorithms on image GT07, GT09, GT12, GT13, GT15 and GT16. The first and second rows are the input image and corresponding trimap respectively, the third to seventh rows are the alpha matte obtained by Bayesian, CF, KL, KNN and Poisson matting algorithm, respectively. The eighth row is the alpha matte of this algorithm, and the local magnification of all experimental results.

ISSN 2616-5775 Vol. 6, Issue 11: 60-71, DOI: 10.25236/AJCIS.2023.061109

Figure 12: The alpha matte of different matting algorithms on image GT17, GT18, GT19, GT22, GT23 and GT25. The first and second rows are the input image and corresponding trimap respectively, the third to seventh rows are the alpha matte obtained by Bayesian, CF, KL, KNN and Poisson matting algorithm, respectively. The eighth row is the alpha matte of this algorithm, and the local magnification of all experimental results.

Acknowledgements

This work was supported in part by University-level research projects of Sanya University. The research project is titled: Research on matting algorithm based on superpixels (project number: USYJSPY2243).

References

[1] Porter T, Duff T. Compositing digital images[C]//Proceedings of the 11th annual conference on Com-

ISSN 2616-5775 Vol. 6, Issue 11: 60-71, DOI: 10.25236/AJCIS.2023.061109

puter graphics and interactive techniques. 1984:253-259.
[2] Chuang Y Y, Curless B, Salesin D H, et al. A bayesian approach to digital matting[C]//Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition. CVPR 2001. IEEE, 2001, 2: II-II.
[3] Wang J, Cohen M F. An iterative optimization approach for unified image segmentation and matting [C] //Tenth IEEE International Conference on Computer Vision (ICCV'05) Volume 1. IEEE, 2005, 2: 936-943.
[4] Ruzon M A, Tomasi C. Alpha estimation in natural images[C]//Proceedings IEEE Conference on Computer Vision and Pattern Recognition. CVPR 2000 (Cat. No. PR00662). IEEE, 2000, 1: 18-25.
[5] Sun J, Jia J, Tang C K, et al. Poisson matting [M]//ACM SIGGRAPH 2004 Papers. 2004: 315-321.
[6] Levin A, Lischinski D, Weiss Y. A closed-form solution to natural image matting [J]. IEEE transacttions on pattern analysis and machine intelligence, 2007, 30(2): 228-242.
[7] Liu J J, Hou Q, Cheng M M, et al. Improving convolutional networks with self-calibrated convolutions[C]//Proceedings of the IEEE/CVF conference on computer vision and pattern recognition. 2020: 10096-10105.
[8] Chen Q, Li D, Tang C K. KNN matting[J]. IEEE transactions on pattern analysis and machine intelligence, 2013, 35(9): 2175-2188.
[9] Rhemann C, Rother C, Rav-Acha A, et al. High resolution matting via interactive trimap segmentation [C]//2008 IEEE Conference on Computer Vision and Pattern Recognition. IEEE, 2008:1-8.
[10] Mishima Y. Soft edge chroma-key generation based upon hexoctahedral color space: U.S. Patent 5,355,174[P]. 1994-10-11.
[11] Guan Y, Chen W, Liang X, et al. Easy Matting-A Stroke Based Approach for Continuous Image Matting[C]//Computer Graphics Forum. Oxford, UK and Boston, USA: Blackwell Publishing, Inc, 2006, 25(3): 567-576.
[12] Shahrian E, Rajan D, Price B, et al. Improving image matting using comprehensive sampling sets [C]//Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition. 2013: 636-643.
[13] Johnson J,Varnousfaderani E S, Cholakkal H, et al. Sparse coding for alpha matting[J]. IEEE Transactions on Image Processing, 2016, 25(7): 3032-3043.
[14] Karacan L, Erdem A, Erdem E. Alpha matting with kl-divergence-based sparse sampling[J]. IEEE Transactions on Image Processing, 2017, 26(9):4523-4536.
[15] Xu N, Price B, Cohen S,et al. Deep image matting[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 2970-2979.
[16] Hou Q, Liu F. Context-aware image matting for simultaneous foreground and alpha estimation [C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019: 4130-4139.
[17] Lutz S, Amplianitis K, Smolic A. Alphagan: Generative adversarial networks for natural image matting [J]. arXiv preprint arXiv:1807.10088,2018.
[18] Cai S, Zhang X, Fan H, et al.Disentangled image matting[C]//Proceedings of the IEEE/CVF International Conference on Computer Vision. 2019:8819-8828.
[19] Sengupta S, Jayaram V, Curless B, et al. Background matting: The world is your green screen [C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 2291-2300.
[20] Yu Q, Zhang J, Zhang H, et al. Mask guided matting via progressive refinement network[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2021: 1154-1163.
[21] Qiao Y, Liu Y, Yang X, et al. Attention-guided hierarchical structure aggregation for image matting[C]//Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. 2020: 13676-13685.
[22] Ke Z, Li K, Zhou Y, et al. Is a green screen really necessary for real-time portrait matting?[J]. arXiv preprint arXiv:2011.11961, 2020.
[23] He K, Zhang X, Ren S, et al. Deep residual learning for image recognition[C]//Proceedings of the IEEE conference on computer vision and pattern recognition. 2016: 770-778.
[24] Huang G, Liu Z, Van Der Maaten L, et al. Densely connected convolutional networks[C] //Proceedings of the IEEE conference on computer vision and pattern recognition. 2017: 4700-4708.
[25] Simonyan K, Zisserman A. Very deep convolutional networks for largescale image recognition[J]. arXiv preprint arXiv:1409.1556, 2014.
[26] Everingham M, Van Gool L,Williams C K I, et al. The pascal visual object classes (voc) challenge[J]. International journal of computer vision, 2010, 88(2): 303-338.
[27] Rhemann C, Rother C, Wang J, et al. A perceptually motivated online benchmark for image matting[C]//2009 IEEE conference on computer vision and pattern recognition. IEEE, 2009: 1826-1833.
[28] Wang Z, Bovik A C, Sheikh H R, et al. Image quality assessment: from error visibility to structural similarity[J]. IEEE transactions on image processing, 2004, 13(4): 600-612.
[29] Chen X , Zou D , Zhou S Z , et al. Image Matting with Local and Nonlocal Smooth Priors[C]//Computer Vision & Pattern Recognition. IEEE, 2013.