Video Matting via Consistency-Regularized Graph Neural Networks

Tiantian Wang[1], Sifei Liu[2], Yapeng Tian[3], Kai Li[4], Ming-Hsuan Yang[1,5,6]
[1]UC Merced, [2]NVIDIA, [3]University of Rochester, [4]Northeastern University, [5]Google Research, [6]Yonsei University

Abstract

Learning temporally consistent foreground opacity from videos, i.e., video matting, has drawn great attention due to the blossoming of video conferencing. Previous approaches are built on top of image matting models, which fail in maintaining the temporal coherence when being adapted to videos. They either utilize the optical flow to smooth frame-wise prediction, where the performance is dependent on the selected optical flow model; or naively combine feature maps from multiple frames, which does not model well the correspondence of pixels in adjacent frames. In this paper, we propose to enhance the temporal coherence by Consistency-Regularized Graph Neural Networks (CRGNN) with the aid of a synthesized video matting dataset. CRGNN utilizes Graph Neural Networks (GNN) to relate adjacent frames such that pixels or regions that are incorrectly predicted in one frame can be corrected by leveraging information from its neighboring frames. To generalize our model from synthesized videos to real-world videos, we propose a consistency regularization technique to enforce the consistency on the alpha and foreground when blending them with different backgrounds. To evaluate the efficacy of CRGNN, we further collect a real-world dataset with annotated alpha mattes. Compared with state-of-the-art methods that require hand-crafted trimaps or backgrounds for modeling training, CRGNN generates favorably results with the help of unlabeled real training dataset. The source code and datasets are available at https://github.com/TiantianWang/VideoMatting-CRGNN.git.

1. Introduction

Video matting aims to estimate the foreground opacity (alpha matte) of each video frame. It has drawn much attention recently due to the blossoming of video conferencing. Typically, the predicted alpha matte can be utilized to create new composites for video editing. Unlike the binary segmentation task, matting produces soft masks that better represent object boundaries or transparent material. Simply segmenting the foreground regions does not synthesize realistic image or video composition results due to the neglect of the transition zone. To obtain accurate video matting, we need to guarantee that: (i) alpha mattes extracted on individual frames should accurately represent the object to be extracted, i.e., the spatial accuracy, and (ii) extracted mattes should not result in noticeable temporal jitter, i.e., the temporal coherence. Compared to spatial accuracy, temporal coherence is often more important in video matting as the human visualization system is more sensitive to temporal inconsistency when watching a video [42].

Figure 1: Matting results of different models. The first row shows the image and ground truth. The second row represents the predictions of a video matting method [34] (Left) and our method (Right). The third row shows the blended image generated by the foreground and predicted alpha. Clearly, our method can predict more subtle details on the hairs.

However, due to the lack of a large-scale video matting dataset, previous methods usually build video matting systems on top of image matting models. For instance, one naive way is to directly apply an image matting approach frame by frame. However, this will cause inconsistent alpha prediction across frames. To improve the temporal coherence of alpha mattes, the previous methods usually utilize

the optical flow [25, 39, 27, 35] to smooth frame-wise prediction, or leverage a stack of nearby video frames to exploit motion cues [34]. These methods still lead to several issues. First, warping information from the reference frame to the query frame relies on the quality of optical flow being used. Normally, a faster solution of optical flow produces inaccurate propagation, while a more accurate one is usually time-consuming. Furthermore, merely combining multiple frames in the feature level ignores the interactions between frames, and does not model the motion flow of pixels in time.

In this paper, we focus on the two challenges for video matting. First, how to produce temporally coherent alpha predictions with the existing image matting dataset [48]? Second, how to mitigate the domain gap when transferring the model trained on the composited dataset to the real videos? We propose the Consistency-Regularized Graph Neural Networks (CRGNN) to address these two challenges. We first design a graph neural network, in space and time, with the aid of a composited video matting dataset to enhance the temporal coherence. Second, a consistency regularization technique is proposed to generalize our model pretrained on the composited dataset to the real one.

In particular, we construct a fully-connected graph neural network to enhance temporal coherence by exploiting the interactive relation between different frames. In this graph, the nodes denote video frames and edges link a pair of neighboring frames which are represented by the pairwise relation. With the graph structure, we encourage information to be propagated across frames, in order to complement the information for the missing pixels in the current frame and smooth the predictions over time. As shown in Figure 1, the proposed method can generate more detailed structures compared to the video-based method [34] that does not exploit the interaction between frames, which demonstrates the advantages of the graph neural network for recovering missing pixels assisted by neighboring frames. As another important contribution to assist the above training process, we also propose a new composited video matting dataset in which alphas are manually annotated against the green screen videos.

To address the second challenge, we need to adapt our model – supervised trained on the composited dataset, to real videos. As such, we introduce a consistency-regularized adversarial learning scheme. On the one hand, we enforce a consistency loss: we blend the prediction of the alpha and the foreground with a random new background, forwarding this new image to have a new version of alpha/foreground pairs, and encouraging them to be consistent. On the other hand, we introduce a discriminator to better differentiate the composited frames and real ones in an adversarial manner. To verify the efficacy of the proposed method, we evaluate our method on a new real-world dataset in which the alpha mattes are carefully extracted from the background.

Compared to the existing methods which either utilize trimaps or backgrounds as the input for modeling training, our background-free method achieves better performance against the state-of-the-arts on the composited and real datasets with the help of unlabeled real training dataset.

Our contributions can be summarized in three aspects:

- We propose a graph neural network to fully exploit the interactive relationship between multiple video frames to enhance the temporal coherence with the assist of a composited video matting dataset.
- We present a consistency regularization technique to adapt the model trained on the composited video frames to the real ones, which can enhance the consistency on the alpha and foreground.
- We propose two large-scale composited datasets and one manually annotated real dataset for the future development of this area. Extensive experiments are conducted on the proposed datasets, showing that the proposed method performs favorably against the state-of-the-arts.

2. Related Work

In this section, we review methods closely related to this work including image matting, video matting, and graph neural networks.

Image matting. Early image matting methods can be roughly categorized into color sampling-based techniques [12, 18, 19, 21, 4, 36] and alpha propagation-based approaches [1, 10, 20, 26, 37]. Recently, methods based on the convolutional neural networks have achieved state-of-the-art results in the image matting task [14, 11, 11, 48, 16, 46, 15, 8, 38, 49, 7, 28, 22]. For example, Xu et al. [48] propose to learn the alpha matte from the input image and trimap based on the alpha and composited image losses. Indexed pooling and upsampling operation are introduced by Lu et al. [28] to recover boundary details.

Video matting. Different from image matting, video matting [2, 6, 39, 27, 25, 30, 34] aims to estimate temporally-coherent alpha mattes. Existing methods usually utilize propagation modules to maintain the coherence among different frames. For instance, Lee et al. [2] first generate trimaps on some key frames in an interactive manner, and then propagate the trimaps to all other frames. Schahrian et al. [35] take each video frame and trimap as the input, and use the matting Laplacian to refine the sampled background and foreground regions. Soumyadip et. al. [34] propose a trimap-free method that utilizes an additional background image and segmentation map as the input and utilizes the image matting dataset for network pretraining.

Graph neural networks. Graph Neural Networks (GNN) are proposed to handle graph-structured data with deep

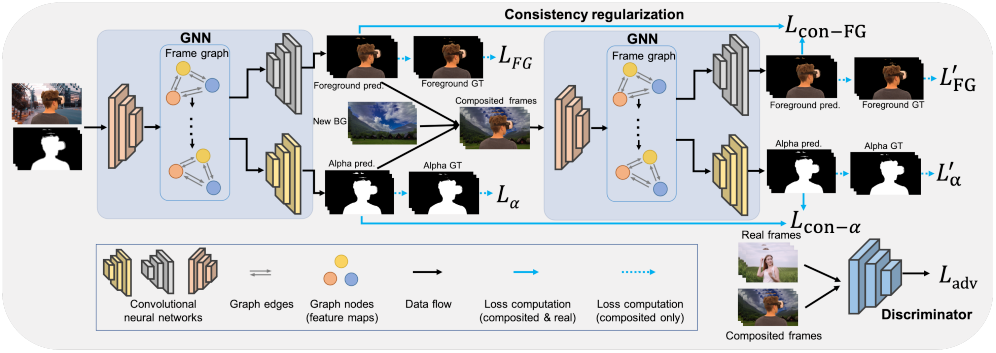


Figure 2: Overview of the proposed method. Given video frames and (pseudo) trimaps, the proposed model first predicts the foreground color and alpha mattes via the GNN by leveraging the frame-wise interaction. Then the predicted foregrounds and alphas are blended with new backgrounds to generate new images, which are forwarded into the same GNN to generate new foregrounds and alphas. The consistency regularization and discriminator are proposed to generalize the model trained on the labeled composited videos to the unlabeled real videos.

learning, which has been applied to fields such as detection [32], segmentation [43, 29] and classification [41]. The previous GNN based video object segmentation method [43] utilizes the GNN to mine the inter-frame relationship over graphs to predict the segmentation map for each frame. Though the motivation using the GNN to exploit the inter-frame relationship is similar, our method shows significant differences compared to [43]. First, we exploit the inter-frame relationship by utilizing the locally-connected information in contrast with the non-local structure in [43], which can generate more clear boundaries than the non-local structure. Second, we enhance the graph neural network by introducing the consistency-regularization and adversarial learning, which can help the network trained on the composited dataset be adapted better to the real dataset.

3. Proposed Algorithm

Video matting is the task that given a video $\mathcal{V} = \{I_i\}_{i=1}^{V}$ of V frames, the goal is to decompose each frame $I_i \in \mathcal{V}$ as:

$$I_i = A_i * F_i + (1 - A_i) * B_i, \tag{1}$$

where A_i, F_i and B_i are the alpha matte, foreground color and background color, respectively. The symbol $*$ means the Hadamard product. Video matting is a challenging task because it entails obtaining high-quality details of each individual frame while maintaining favorable temporal consistency across frames.

We tackle this task by collecting a large-scale composited dataset (introduced in the next section) and proposing a novel model which utilizes graph neural networks to associate pixels in space and time. As a result, the learned model is supposed to produce video matting results with temporal coherence enhanced. To generalize our GNN based model from composited videos to real videos where the backgrounds are arbitrary and no real ground truths are available for model training, we propose a novel consistency regularization approach which enforces the consistency of the extracted foregrounds and alpha mattes under different backgrounds. The learned model is thus capable of addressing the variety and complexity of backgrounds in real videos. Moreover, we adopt adversarial training to further mitigate the domain gap between composited and real videos. The framework can be found in Figure 2.

3.1. Composited Video Matting

Given a video $\mathcal{V} = \{I_i\}_{i=1}^{V}$ with ground truth labels $\mathcal{Y}_i = (A_i, F_i, B_i)$ for each frame I_i, we generate a trimap T_i from A_i which provides coarse information of the foreground, background and unknown regions, following existing image matting methods [48, 28]. An encoder network E takes as input I_i and T_i, producing a latent representation $\mathbf{x}_i \in \mathbb{R}^{H \times W \times C}$ as

$$\mathbf{x}_i = E([I_i \oplus T_i]), \tag{2}$$

where $\oplus$ denotes the concatenation operator. H, W and C represent the height, width and channel of the feature map, respectively. We propose to use the GNN to model the temporal consistency among frames. The core idea is to exploit the inter-frame relationship by performing feature aggregations so that vertex features can be updated by aggregating features of the associated nodes weighted by the connectivity (edge).

We define a graph with K vertices at the t-step, $\mathcal{G}^t = (\mathcal{V}^t, \mathcal{E}^t)$, where the vertices $\mathcal{V}^t = \{\mathbf{x}_i^t\}_{i=1}^{K}$ represent the latent feature for the i-th frame in the graph and the edges $\mathcal{E}^t = \{\{\mathbf{e}_{i,j}^t\}_{i=1}^{K}\}_{j=1}^{K}$ denote the relationship between two vertices,

$$\mathbf{e}_{i,j}^t = f_t(\mathbf{x}_i^t, \mathbf{x}_j^t), \tag{3}$$

where $f_t(\cdot)$ denotes the aggregation function at the t-th step.

Feature aggregation. Here we adopt the deformable alignment [40, 44], which utilizes the deformable convolution

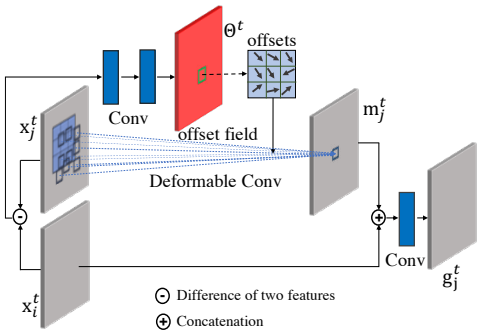


Figure 3: Deformable alignment-based feature aggregation.

to implement the feature aggregation. Different from the standard 2D offsets to the regular grid sampling locations, deformable convolution enables free form deformation of the sampling grid, which is implemented on an irregular grid augmented with data-conditioned offsets. Given two feature embeddings $\mathbf{x}_i$ and $\mathbf{x}_j$, the offsets on the regular convolution kernels (such as 3×3) are calculated by

$$\Theta^t = f_\theta(\mathbf{x}_i^t, \mathbf{x}_j^t), \tag{4}$$

where $\Theta^t = \{\triangle p_n \mid n = 1, \ldots, |\mathcal{R}|\}$ represents the offsets of the convolution kernels. $\mathcal{R} = \{(-1,-1), (-1,0), ..., (0,1), (1,1)\}$ denotes the regular grid of a convolutional kernel.

With the predicted Θ^t and the feature embedding $\mathbf{x}_j^t$, the aligned feature map $\mathbf{m}_j^t$ for each position p_0 can be formulated by the following operation:

$$\mathbf{m}_j^t(p_0) = \sum_{p_n \in \mathcal{R}} w(p_n) \mathbf{x}_j^t(p_0 + p_n + \triangle p_n). \tag{5}$$

Since the offset $\triangle p_n$ is typically fractional, the operation above is implemented using bilinear interpolation, similar to [13]. This yields an offset map that has the same spatial resolution as the input feature map. The learned offset can capture motion cues and also explore neighboring features to maintain the temporal coherence among the whole video.

Then the aggregated feature for the i-th frame is calculated by:

$$\mathbf{g}_j^t = f_a(\mathbf{m}_j^t \oplus \mathbf{x}_i^t), \tag{6}$$

where $\oplus$ denotes the concatenation operation and f_a means the convolutional operation. Figure 3 illustrates the feature aggregation process.

Node-state updating. Each vertex aggregates information from its neighboring vertices to update its original representation. In the t-th passing step, we model the node-state updating process using the ConvGRU [3, 43] as

$$\mathbf{g}_i^t = f_g(f_c(\mathbf{g}_0^t \oplus \cdots \mathbf{g}_{i-1}^t \oplus \mathbf{g}_{i+1}^t \cdots \oplus \mathbf{g}_K^t), \mathbf{x}_i^t), \tag{7}$$

where f_c is a convolution operator for dimensional reduction. $f_g(\cdot,\cdot)$ stands for the Gated Recurrent Unit (GRU). The feature aggregation and node-state updating procedures will be executed alternatively up to T times. Other models such as ConvLSTM can also be used for the node-state updating. Here we use ConvGRU because its has fewer parameters and can be trained more efficiently.

Network prediction. After T message passing iterations, all K node representations are updated. Then the updated representations are used to predict the alpha matte and foreground using the decoders D_a and D_f as

$$\hat{A}_i = D_a(\mathbf{g}_i^T), \quad \hat{F}_i = D_f(\mathbf{g}_i^T). \tag{8}$$

The input frame is reconstructed by

$$\hat{I}_i = \hat{A}_i * F_i + (1 - \hat{A}_i) * B_i. \tag{9}$$

We train our model by minimizing the sum of the prediction errors of the alpha matte, foreground and input frame as

$$L_{gt} = L_\alpha + L_{FG} + L_{Frm}, \tag{10}$$

where $L_\alpha = \frac{1}{K}\sum_{i=1}^K \|\hat{A}_i - A_i\|_{\mathrm{F}}^2$, $L_{FG} = \frac{1}{K}\sum_{i=1}^K \|\hat{F}_i - F_i\|_{\mathrm{F}}^2$, and $L_{Frm} = \frac{1}{K}\sum_{i=1}^K \|\hat{I}_i - I_i\|_{\mathrm{F}}^2$.

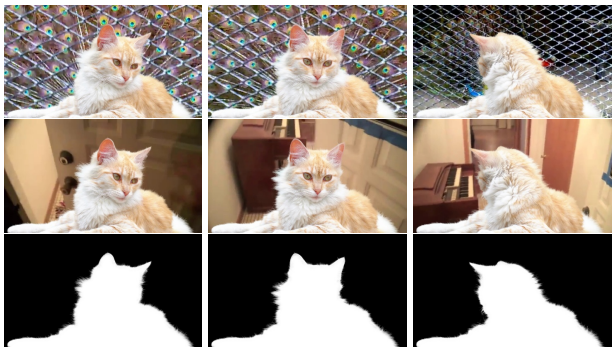
3.2. Real Video Matting

The proposed GNN based model trained on the composited video dataset can help improve the temporal coherence compared to the model pretrained on the image matting dataset [48]. However, it may still fail when applied to real videos due to the domain gap. To avoid this, we propose a novel regularization approach that enforces consistency on the alpha and foreground, when blending them with different backgrounds. Besides, we adopt an adversarial training scheme to further mitigate the domain gap between composited videos and real ones.

Specifically, let $\mathcal{V} = \{I_i\}_{i=1}^V$ be a video drawn from the composited set and $\mathcal{R} = \{U_i\}_{i=1}^U$ be a video drawn from the real set. $\mathcal{V}$ is labeled but $\mathcal{R}$ is not.

Consistency regularization. The proposed consistency regularization does not require labels so that it can be identically applied on our model when utilizing both $\mathcal{V}$ and $\mathcal{R}$ as the input except the way of producing the trimap, which is another input for our GNN based model. For each frame $I_i \in \mathcal{V}$, we directly generate the trimap T_i using the ground truth alpha matte, following the previous image matting methods [48, 28]. For $\mathcal{R}$, since the ground truth alpha mattes are not available for training, the pseudo trimap is generated by the segmentation map based on the DeepLabv3 [9].

Taking $\mathcal{V}$ as an example, our GNN model utilizes I_i and T_i as the input and generates the alpha matte $\hat{A}_i$ and foreground $\hat{F}_i$. Then, $\hat{F}_i$ is composited with a random new background B by the alpha $\hat{A}_i$ to generate a new frame, $\hat{I}_i = \hat{A}_i * \hat{F}_i + (1 - \hat{A}_i) * B$. The composited frame $\hat{I}_i$ is fed into the GNN model again and generates a new alpha



(a) Examples from the composited dataset

(b) Examples from the real dataset

Figure 4: Video matting dataset. (a) The first two rows show the composited video frames with the same foreground objects. The foregrounds are first generated from the videos with simple background and then composited with two different backgrounds. (b) The first row shows the original real video frames and the second row indicates the objects are blended with a new background using the annotated foreground and alpha. The third row in (a) and (b) represents the annotated alpha.

matte $\bar{A}_i$ and foreground prediction $\bar{F}_i$. $\hat{F}_i$ and $\bar{F}_i$ should be consistent with each other, as they represent the same object against different backgrounds. The same goes for $\hat{A}_i$ and $\bar{A}_i$. Besides, a new frame can be composited by $\bar{I}_i = \bar{A}_i * \bar{F}_i + (1 - \bar{A}_i) * B$, and $\bar{I}_i$ should also be consistent with $\hat{I}_i$. Thus, we define the consistency regularizer as

$$L^c_{con} = L_{con-\alpha} + L_{con-FG} + L_{con-Frm}, \tag{11}$$

where $L_{con-\alpha} = \frac{1}{K}\sum_{i=1}^{K} \|\hat{A}_i - \bar{A}_i\|_{\mathrm{F}}^2$, $L_{con-FG} = \frac{1}{K}\sum_{i=1}^{K} \|\hat{F}_i - \bar{F}_i\|_{\mathrm{F}}^2$, and $L_{con-Frm} = \frac{1}{K}\sum_{i=1}^{K} \|\hat{I}_i - \bar{I}_i\|_{\mathrm{F}}^2$. Similarly, we can get the consistency regularizer L^r_{con} when the network utilizes the real frame as the input.

With the consistency regularization losses L^c_{con} and L^r_{con}, we can reach our learning objective as

$$L_{adapt} = L^c_{con} + L^r_{con} + L_{gt} + L'_{gt}, \tag{12}$$

where L_{gt} is calculated by Eq. (10) using $(\hat{A}_i, \hat{F}_i)$ and the ground truth label (A_i, F_i). L'_{gt} is calculated by Eq. (10) as well, but using $(\bar{A}_i, \bar{F}_i)$ and (A_i, F_i).

Adversarial learning. Adversarial learning has been widely used for addressing the domain adaptation problem. Here, we introduce adversarial learning to further mitigate the domain gap. Motivated by [5], we augment the data by translating the foreground objects with an arbitrary small shift $\delta \sim \mu([-\sigma, \sigma] \times [-\sigma, \sigma])$, where the σ defines the range of the local shift. We can synthesize a composited image as

$$\hat{U}_i = \hat{A}^u_i[p + \delta] * \hat{F}^u_i[p + \delta] + (1 - \hat{A}^u_i[p + \delta]) * B. \tag{13}$$

where $\hat{A}^u_i$ and $\hat{F}^u_i$ are predicted from real frame U_i. $\hat{A}^u_i[p]$ indexes the image pixel at the specified localization and p indicates the coordinates. By compositing the alpha and foreground predictions of real frames with random backgrounds, we can obtain composited images that are hard for a discriminator to distinguish whether it is real or composited. This in return enhances domain alignment results. We optimize the adversarial loss L_{adv} by

$$\min_{\theta_D} \mathop{\mathbb{E}}_{\hat{U}_i \sim P_{\mathcal{R}}, B \sim P_{\mathcal{B}}} [D(\hat{U}_i)^2] + \mathop{\mathbb{E}}_{U^r \sim P_{\mathcal{R}}} [(D(U^r) - 1)^2], \tag{14}$$

where $P_{\mathcal{R}}$ and $P_{\mathcal{B}}$ are the distributions of real frames and background images, respectively. U^r represents a random frame sampled from the real videos. D is the discriminator and θ_D represents the parameters of D.

Though [34] also utilizes a discriminator, the proposed method differs from that for the input. [34] uses the original frame while the proposed method randomly samples one frame from any real video as the real input. The diverse inputs can help the discriminator better differentiate between the composited and real frames.

4. Datasets

As far as we know, there is only one labeled dataset for video matting [33]. It contains 3 training videos and 10 test videos, which is not enough for training a deep learning model and is hard for researchers to evaluate on it because of the inaccessibility to the ground truths. Sengupta et al. [34] capture a human video matting dataset with only the videos provided but no annotations. Because of the shortage of annotation, they propose to utilize the model pretrained on the image-matting dataset [48] to predict the pseudo-label for training a video matting model. However, this will generate temporal jitters and cannot maintain temporal coherence. Labeled data is becoming a bottleneck for the development of this topic.

In this paper, we propose two synthesized datasets to alleviate this problem. Furthermore, to evaluate the generalization of the proposed method trained on the labeled composited dataset to the real dataset, we also provide a real-world dataset. These datasets contain high-resolution (HD)

	MSE	SAD	Gradient	Connectivity	MESSDdt
DIM [48]	10.69	79.87	74.54	72.75	7.676
IM [28]	9.216	81.56	64.97	63.72	5.595
IM* [28]	5.734	54.31	43.82	44.68	3.297
LF [49]	20.61	113.0	168.7	108.2	13.90
CAM [23]	20.97	145.5	147.5	116.2	9.867
BM [34]	13.57	90.15	130.8	84.85	7.388
Ours	**3.770**	**45.77**	**30.80**	**33.81**	**2.475**

(a) Composited dataset.

	MSE	SAD	Gradient	Connectivity	MESSDdt
DIM [48]	13.32	98.92	129.1	88.56	17.48
IM [28]	10.91	95.07	120.0	73.05	14.45
IM* [28]	13.84	97.09	136.9	84.57	17.89
LF [49]	29.61	141.4	168.5	131.7	32.58
CAM [23]	11.62	101.0	123.9	78.21	14.93
Ours	**9.224**	**73.50**	**112.1**	**58.49**	**12.23**

(b) Real dataset.

Table 1: Quantitative results on the two human matting datasets. To better show the performance difference, the numbers for the above measures have been scaled up or scaled down. The scaling factors of the five measures from left to right are 1000, 0.01, 0.01, 0.01, 1000. IM* means we re-train IM using the proposed dataset. The best results are in **bold**.

	MSE	SAD	Gradient	Connectivity	MESSDdt
DIM [48]	25.03	402.1	167.4	407.4	16.47
IM [28]	37.30	582.8	115.3	597.1	16.67
LF [49]	49.25	478.7	339.0	466.3	25.07
CAM [23]	25.95	461.3	92.97	468.6	11.70
Ours	**20.65**	**378.8**	**87.54**	**365.0**	**10.41**

Table 2: Results on the auxiliary category dataset.

videos and the annotations are carefully manually created using Adobe After Effects and Photoshop. Figure 4 shows some examples from the proposed datasets.

Composited video dataset. Because of the increasing interest in human matting on videos, we propose a composited dataset with the human category (composited human matting dataset). We also provide a dataset with categories except for the human (auxiliary category dataset) to verify the generalization of our model on both the human category and other categories. Videos in these two datasets are annotated against the green screen or simple background. Because of the simplicity of the backgrounds, it is easy to generate high-quality alpha mattes and the corresponding foregrounds for each video. For the human matting dataset, there are 20 training videos (6312 frames) and 10 test videos (3807 frames). For the auxiliary category dataset (e.g., cat, plant), 20 training videos (3983 frames) and 10 test videos (1722 frames) are provided. To enlarge the diversity of the dataset, each foreground video is composited with varied backgrounds using the groundtruth alpha mattes.

Real video dataset. To measure the performance of natural videos, we also collect a real-world human matting dataset with 19 videos. The alpha and foreground are manually annotated at every 10 frames with a frame rate of 30 fps for each video, which in total results in 711 frames being labeled.

5. Experiments

We use the data augmentation scheme to increase the diversity of the input data. First, we randomly crop the image and trimap pairs centered on pixels in the unknown regions with varied resolutions (e.g. 480×480, 640×640, 960×960) and resize them to 480×480 due to the memory constraint. We also utilize random rotation, scaling, shearing as well as the vertical and horizontal flipping for the affine transformation. Our model is first pretrained on the image matting dataset [48] and then finetuned using the labeled composited data and unlabeled real data. For the image matting dataset, we use the random affine transformation to generate a short video clip with 3 frames to imitate the motion flow of the objects. Because it is hard to generate the pseudo trimap with the category like transparency, we only utilize the proposed graph neural network on the auxiliary category dataset and adopt the full model on the human matting dataset for training and inference. In the test stage, the trimaps for all datasets are generated from the ground-truth alpha mattes by thresholding and the unknown region is dilated with the kernel size 25.

We adopt the similar encoder and decoder structures introduced in [28]. We remove the last two pooling layers so the output size of the encoder is 1/8 of the input image. The decoder D_a and D_f for predicting the alpha and foreground have same structures except for the prediction layer. The output channels for the prediction layer to predict the alpha and foreground are set to 1 and 3. For the discriminator, we adopt the structure proposed in PatchGAN [24]. All the weights in objective L_{adapt} and L_{adv} used to balance different losses are set to 1. The number of vertices K and the number of iteration step T are set to 3. The running speed is about 1 fps on one single Nvidia 2080 Ti GPU.

5.1. Comparative Results

Evaluation metrics. To show the effectiveness of the proposed method, we evaluate the results on five popular metrics, including the SAD, MSE, Gradient [33], Connectivity [33] and temporal coherence (MESSDdt) [17]. These metrics can be used to evaluate the accuracy of the alpha matte for every single frame and the temporal coherence within a video. The first four metrics are widely used for image-level matting evaluation. However, long-range videos own more features compared to the image. One key feature is the temporal coherence which means the objects move among different frames should be consistent for better human perceptibility.

Results on the composited dataset. We first evaluate the proposed algorithm and state-of-the-art methods on the pro-

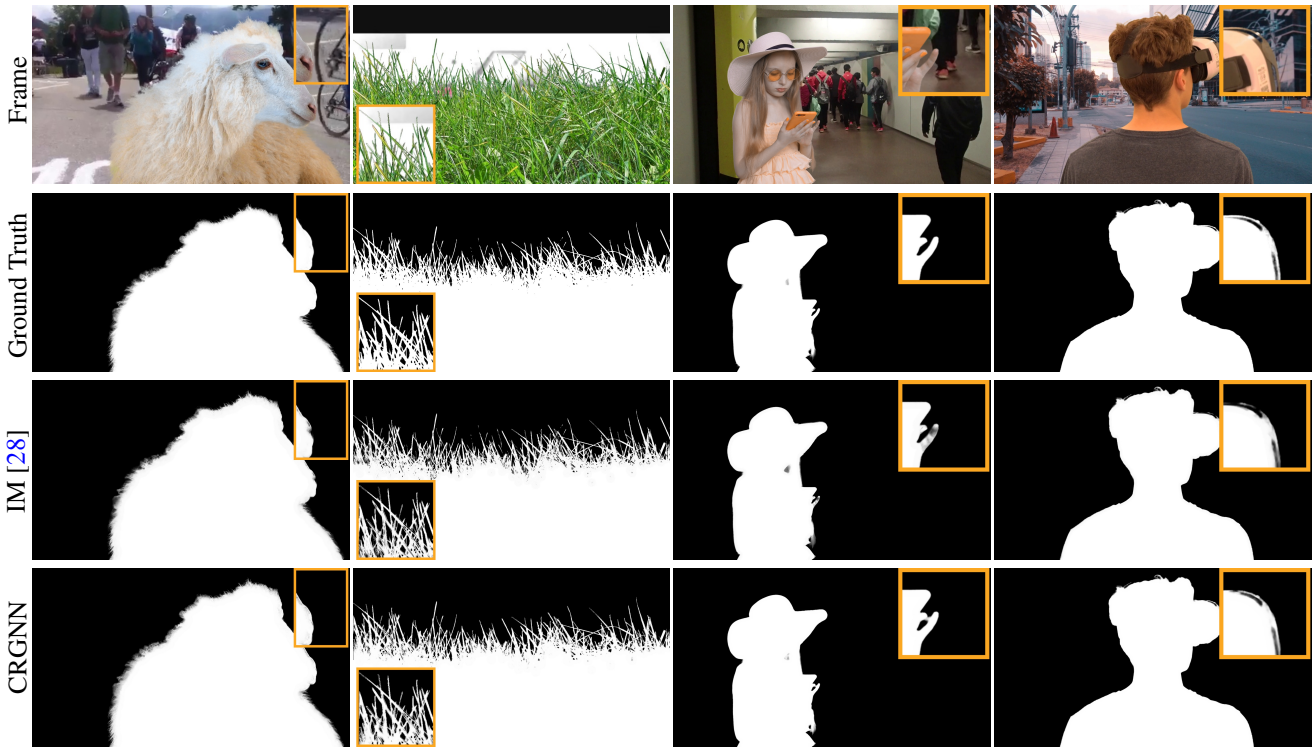


Figure 5: Visual comparison on the composited dataset.

Figure 6: Visual comparison on the real dataset.

posed composited human matting dataset and auxiliary category dataset. We include the existing image based matting methods [48, 49, 28, 23] and video based method [34]. It can be observed from Table 1a and Table 2 that the proposed method achieves better performance compared to all other methods evaluated on all five metrics. Compared to the image-based methods, the performance gain is derived from the utilization of the CRGNN, which leverages multi-frame

information among the whole video and help recover the missing predictions by the feature aggregation. Compared to the video-based method BM [34], the proposed method achieves better performance because the CRGNN assisted by the deformable feature aggregation can fully mine the interactions between frames.

Results on the real dataset. To further verify the efficacy of the proposed method, we evaluate the results on the proposed real-world dataset. The quantitative results are shown in Table 1b. We see that our CRGNN performs best among all methods, which demonstrates the efficacy of our core idea of formulating the video matting as the combination of GNN and consistency regularization technique.

Qualitative results. Figure 5 and 6 show the visual results on the composited and real video datasets. From these results, we can clearly see that the proposed method predicts more subtle details of the frames, such as the grass in the second column of Figure 5 and suppresses the background better as shown in the second column of Figure 6. These further substantiate the superiority of the proposed method for the video matting task.

		MSE	SAD	Gradient	Connectivity	MESSDdt
Variants	Baseline	10.21	90.23	130.7	67.23	15.31
	+GNN	9.480	78.38	123.2	62.81	13.45
	+Consistency	9.260	73.21	115.4	60.75	12.69
	+Discriminator	9.223	73.49	112.1	58.49	12.23
Number of nodes	#5	9.230	74.62	115.7	58.53	12.30
	#7	9.228	73.77	115.2	58.50	12.27
Non-local agg.	-	9.954	89.45	128.9	65.68	13.56

Table 3: Ablation study on the variants of the proposed network. 'Baseline' means the image-level model without using the GNN. '+' means the progressive connection of different modules.

5.2. Ablation Study

We perform an ablation study to investigate the effect of each essential component of the proposed method.

Effectiveness of the proposed graph neural network. To analyze the contribution of our CRGNN, we introduce a baseline model by removing the inter-frame relationship, that is, the image-level baseline using the encoder-decoder structure similar to [28]. Each video frame is forwarded into our baseline model frame by frame. As shown in the second row of Table 3, GNN indeed brings significant performance improvements compared to the image-level model in the first row, which benefits from the introduction of multiple frames in enhancing the temporal coherence.

Effectiveness of the consistency regularization strategy. To investigate the effectiveness of the consistency scheme, we provide the results with and without prediction consistency in Table 3. Compared to the results without utilizing the alpha, foreground and frame consistency (the second row), utilizing the prediction consistency can generate better result, (e.g. MSE: 9.260 v.s. 9.480). The performance gain is derived from the better feature representation enhanced by the consistency regularization.

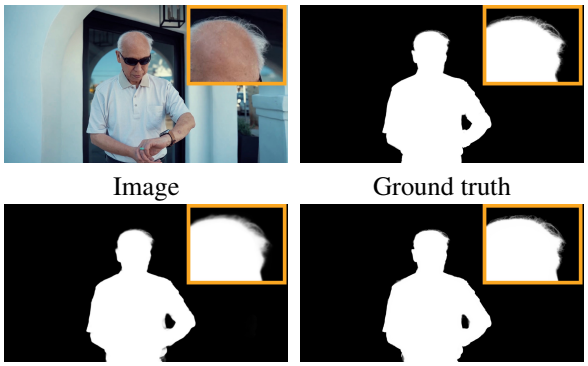


Figure 7: Visual comparison of deformable aggregation and the non-local aggregation on the real dataset.

Effectiveness of the adversarial learning scheme. The fourth row in Table 3 shows that the introduction of the discriminator can further improve the performance based on the consistency regularization, which benefits from the advantages of the discriminator to distinguish if the image belongs to the composited image or the real one.

Comparison of different number of nodes. We report the performance using the different number of nodes during the test stage. As shown in Table 3, increasing the number of nodes generates comparable results.

Comparison with the non-local structure. The non-local structure [45] has been widely used for feature aggregation on various tasks, such as video object segmentation [31] and object detection [47]. Features are aggregated by enumerating all possible positions in the embedding space. As shown in Table 3, the proposed method can generate better results comparing to utilize the non-local structure for aggregation.

6. Conclusion

In this paper, we focus on enhancing the temporal coherence for matting in videos. Different from the previous methods built on the image matting models, we propose to maintain the temporal consistency by fully exploiting the inter-frame relationship among the whole video. We use a graph neural network to relate adjacent frames with the aid of annotated synthesized video matting datasets. To generalize the proposed model from synthesized videos to real-world videos, we propose a regularization scheme to enforce the consistency on the alpha, foreground and predicted frames. In addition, we annotate a real-world dataset with alpha mattes to evaluate the efficacy of the proposed method. Extensive experiments on the synthesized and real datasets show the proposed CRGNN model performs favorably against the state-of-the-art methods.

7. Acknowledgements

This work is supported in part by the NSF CAREER Grant #1149783.

References

[1] Yagiz Aksoy, Tunc Ozan Aydin, and Marc Pollefeys. Designing effective inter-pixel information flow for natural image matting. In *CVPR*, 2017. 2

[2] Xue Bai, Jue Wang, and David Simons. Towards temporally-coherent video matting. In *International Conference on Computer Vision/Computer Graphics Collaboration Techniques and Applications*, 2011. 2

[3] Nicolas Ballas, Li Yao, Chris Pal, and Aaron Courville. Delving deeper into convolutional networks for learning video representations. *arXiv preprint arXiv:1511.06432*, 2015. 4

[4] Arie Berman, Arpag Dadourian, and Paul Vlahos. Method for removing from an image the background surrounding a selected object, 2000. US Patent 6,134,346. 2

[5] Adam Bielski and Paolo Favaro. Emergence of object segmentation in perturbed generative models. In *NeurIPS*, 2019. 5

[6] Nicole Brosch, Asmaa Hosni, Christoph Rhemann, and Margrit Gelautz. Spatio-temporally coherent interactive video object segmentation via efficient filtering. In *Joint DAGM (German Association for Pattern Recognition) and OAGM Symposium*, 2012. 2

[7] Shaofan Cai, Xiaoshuai Zhang, Haoqiang Fan, Haibin Huang, Jiangyu Liu, Jiaming Liu, Jiaying Liu, Jue Wang, and Jian Sun. Disentangled image matting. In *ICCV*, 2019. 2

[8] Guanying Chen, Kai Han, and Kwan-Yee K. Wong. Tom-net: Learning transparent object matting from a single image. In *CVPR*, pages 9233–9241, 2018. 2

[9] Liang-Chieh Chen, George Papandreou, Florian Schroff, and Hartwig Adam. Rethinking atrous convolution for semantic image segmentation. *arXiv preprint arXiv:1706.05587*, 2017. 4

[10] Qifeng Chen, Dingzeyu Li, and Chi-Keung Tang. Knn matting. *TPAMI*, 2013. 2

[11] Donghyeon Cho, Yu-Wing Tai, and Inso Kweon. Natural image matting using deep convolutional neural networks. In *ECCV*, 2016. 2

[12] Yung-Yu Chuang, Brian Curless, David H Salesin, and Richard Szeliski. A bayesian approach to digital matting. In *CVPR*, 2001. 2

[13] Jifeng Dai, Haozhi Qi, Yuwen Xiong, Yi Li, Guodong Zhang, Han Hu, and Yichen Wei. Deformable convolutional networks. In *ICCV*, 2017. 4

[14] Henghui Ding, Xudong Jiang, Ai Qun Liu, Nadia Magnenat Thalmann, and Gang Wang. Boundary-aware feature propagation for scene segmentation. In *Proceedings of the IEEE/CVF International Conference on Computer Vision*, pages 6819–6829, 2019. 2

[15] Henghui Ding, Xudong Jiang, Bing Shuai, Ai Qun Liu, and Gang Wang. Context contrasted feature and gated multi-scale aggregation for scene segmentation. In *CVPR*, 2018. 2

[16] Henghui Ding, Xudong Jiang, Bing Shuai, Ai Qun Liu, and Gang Wang. Semantic correlation promoted shape-variant context for segmentation. In *CVPR*, 2019. 2

[17] Mikhail Erofeev, Yury Gitman, Dmitriy Vatolin, Alexey Fedorov, and Jue Wang. Perceptually motivated benchmark for video matting. In *BMVC*, 2015. 6

[18] Xiaoxue Feng, Xiaohui Liang, and Zili Zhang. A cluster sampling method for image matting via sparse coding. In *ECCV*, 2016. 2

[19] Eduardo SL Gastal and Manuel M Oliveira. Shared sampling for real-time alpha matting. In *Computer Graphics Forum*, 2010. 2

[20] Leo Grady, Thomas Schiwietz, Shmuel Aharon, and Rüdiger Westermann. Random walks for interactive alpha-matting. In *VIIP*, 2005. 2

[21] Kaiming He, Christoph Rhemann, Carsten Rother, Xiaoou Tang, and Jian Sun. A global sampling method for alpha matting. In *CVPR*, 2011. 2

[22] Qiqi Hou and Feng Liu. Context-aware image matting for simultaneous foreground and alpha estimation. In *ICCV*, 2019. 2

[23] Qiqi Hou and Feng Liu. Context-aware image matting for simultaneous foreground and alpha estimation. In *ICCV*, 2019. 6, 7

[24] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *CVPR*, 2017. 6

[25] Sun-Young Lee, Jong-Chul Yoon, and In-Kwon Lee. Temporally coherent video matting. *Graphical Models*, 2010. 2

[26] Anat Levin, Dani Lischinski, and Yair Weiss. A closed-form solution to natural image matting. *TPAMI*, 2007. 2

[27] Dingzeyu Li, Qifeng Chen, and Chi-Keung Tang. Motion-aware knn laplacian for video matting. In *ICCV*, 2013. 2

[28] Hao Lu, Yutong Dai, Chunhua Shen, and Songcen Xu. Indices matter: Learning to index for deep image matting. In *ICCV*, 2019. 2, 3, 4, 6, 7, 8

[29] Ao Luo, Xin Li, Fan Yang, Zhicheng Jiao, Hong Cheng, and Siwei Lyu. Cascade graph neural networks for rgb-d salient object detection. In *ECCV*, 2020. 3

[30] J Moon, D Kim, and R Park. Video matting based on background estimation. *Proc. World Acad. Sci., Eng. Technol*, 2005. 2

[31] Seoung Wug Oh, Joon-Young Lee, Ning Xu, and Seon Joo Kim. Video object segmentation using space-time memory networks. In *ICCV*, 2019. 8

[32] Siyuan Qi, Wenguan Wang, Baoxiong Jia, Jianbing Shen, and Song-Chun Zhu. Learning human-object interactions by graph parsing neural networks. In *ECCV*, 2018. 3

[33] Christoph Rhemann, Carsten Rother, Jue Wang, Margrit Gelautz, Pushmeet Kohli, and Pamela Rott. A perceptually motivated online benchmark for image matting. In *CVPR*, 2009. 5, 6

[34] Soumyadip Sengupta, Vivek Jayaram, Brian Curless, Steve Seitz, and Ira Kemelmacher-Shlizerman. Background matting: The world is your green screen. In *CVPR*, 2020. 1, 2, 5, 6, 7, 8

[35] Ehsan Shahrian, Brian Price, Scott Cohen, and Deepu Rajan. Temporally coherent and spatially accurate video matting. In *Computer Graphics Forum*, 2014. 2

[36] Ehsan Shahrian, Deepu Rajan, Brian Price, and Scott Cohen. Improving image matting using comprehensive sampling sets. In *CVPR*, 2013. 2

[37] Jian Sun, Jiaya Jia, Chi-Keung Tang, and Heung-Yeung Shum. Poisson matting. In *ACM Transactions on Graphics (ToG)*, 2004. 2

[38] Jingwei Tang, Yagiz Aksoy, Cengiz Oztireli, Markus Gross, and Tunc Ozan Aydin. Learning-based sampling for natural image matting. In *CVPR*, 2019. 2

[39] Zhen Tang, Zhenjiang Miao, Yanli Wan, and Dianyong Zhang. Video matting via opacity propagation. *The Visual Computer*, 2012. 2

[40] Yapeng Tian, Yulun Zhang, Yun Fu, and Chenliang Xu. Tdan: Temporally-deformable alignment network for video super-resolution. In *CVPR*, pages 3360–3369, 2020. 3

[41] Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Lio, and Yoshua Bengio. Graph attention networks. *arXiv preprint arXiv:1710.10903*, 2017. 3

[42] Paulo Villegas and Xavier Marichal. Perceptually-weighted evaluation criteria for segmentation masks in video sequences. *TIP*, 13(8):1092–1103, 2004. 1

[43] Wenguan Wang, Xiankai Lu, Jianbing Shen, David J Crandall, and Ling Shao. Zero-shot video object segmentation via attentive graph neural networks. In *ICCV*, 2019. 3, 4

[44] Xintao Wang, Kelvin CK Chan, Ke Yu, Chao Dong, and Chen Change Loy. Edvr: Video restoration with enhanced deformable convolutional networks. In *CVPRW*, pages 0–0, 2019. 3

[45] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *CVPR*, 2018. 8

[46] Yu Wang, Yi Niu, Peiyong Duan, Jianwei Lin, and Yuanjie Zheng. Deep propagation based image matting. In *IJCAI*, 2018. 2

[47] Fanyi Xiao and Yong Jae Lee. Video object detection with an aligned spatial-temporal memory. In *ECCV*, 2018. 8

[48] Ning Xu, Brian Price, Scott Cohen, and Thomas Huang. Deep image matting. In *CVPR*, 2017. 2, 3, 4, 5, 6, 7

[49] Yunke Zhang, Lixue Gong, Lubin Fan, Peiran Ren, Qixing Huang, Hujun Bao, and Weiwei Xu. A late fusion cnn for digital matting. In *CVPR*, 2019. 2, 6, 7