Onion-Peel Networks for Deep Video Completion

Seoung Wug Oh
Yonsei University

Sungho Lee
Yonsei University

Joon-Young Lee
Adobe Research

Seon Joo Kim
Yonsei University

Abstract

We propose the onion-peel networks for video completion. Given a set of reference images and a target image with holes, our network fills the hole by referring the contents in the reference images. Our onion-peel network progressively fills the hole from the hole boundary enabling it to exploit richer contextual information for the missing regions every step. Given a sufficient number of recurrences, even a large hole can be inpainted successfully. To attend to the missing information visible in the reference images, we propose an asymmetric attention block that computes similarities between the hole boundary pixels in the target and the non-hole pixels in the references in a non-local manner. With our attention block, our network can have an unlimited spatial-temporal window size and fill the holes with globally coherent contents. In addition, our framework is applicable to the image completion guided by the reference images without any modification, which is difficult to do with the previous methods. We validate that our method produces visually pleasing image and video inpainting results in realistic test cases.

1. Introduction

Recently, we have seen rapid progress in the image completion task – synthesizing missing pixels in an image – using deep neural networks [14, 37, 25, 20, 36]. Generative adversarial networks [8] are often used to *hallucinate* missing contents and have shown impressive results. Despite the progress in image completion, video completion has been rarely explored using deep networks and still remains in an early stage. In video completion, we need to fill missing regions with coherent contents through time rather than synthesizing contents at each frame independently. When removing an object in a video, the region occluded by the object may be visible in other (potentially distant) frames, and filling the region without considering the original content in other frames will severely break the temporal consistency. Therefore, it is difficult to directly extent ideas of image inpainting networks to the video completion problem using 3D convolutional or recurrent networks because their temporal receptive fields are too limited or directional.

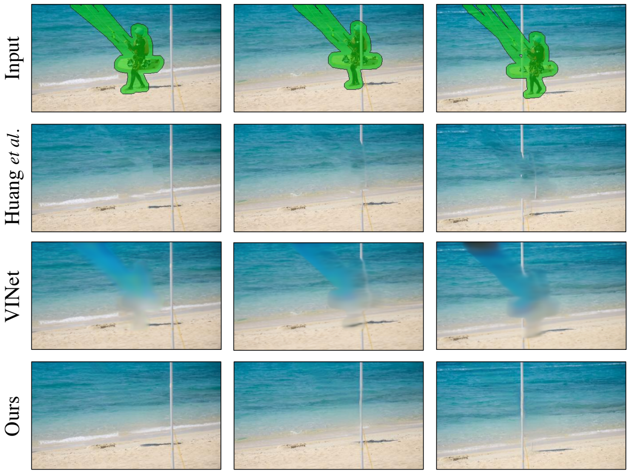


Figure 1: Video completion results from Huang *et al*. [13], VINet [17], and our method. Our method is able to handle challenging cases where the flow computation is not easy due to occlusions or large holes.

For video completion, the traditional approach like the global flow field based optimization technique [13] produces outputs with state-of-the-art quality. Although it performs well in many cases, it is slow because it is computationally intensive and difficult to parallelize. In addition, it suffers from challenging cases where optical flows are noisy (*e.g*. occlusions or large holes). Recently, two deep learning based video completion methods have been proposed. CombCN [30] is built by combining 2D and 3D CNNs, but it is only validated on low-resolution videos of aligned faces and vehicles with fixed square holes, making it difficult to be used for real object removal cases. VINet [17] is designed as a recurrent network and internally computes the flow fields from 5 adjacent frames[1] to the target frame. Due to the network design, its temporal window is restricted to the nearby frames, therefore, it is difficult to inpaint holes with the faithful content that can be visible outside the temporal window. Both works are meaningful in tackling the challenging video completion task using deep networks and

[1] [t-6, t-3, t-1, t+3, t+6], where the current frame is at t

produce video inpainting results much faster than the traditional methods, however, they hardly improve the output quality over the traditional methods.

In this paper, we propose a novel deep network called Onion-Peel Network (OPN) for video completion. Given video frames with inpainting masks, we use a set of (sampled) frames as the reference and fill the hole of the target frame by taking the contents from the reference frames, or synthesizing coherent contents if there is no missing content visible in the reference. Our network inpaints the hole region one layer (peel) at a time by gradually eroding the hole [23, 24][2]. By doing so, our network can exploit richer contextual information for the missing regions at each step. Given a sufficient number of recurrences, even a large hole can be inpainted successfully.

To pick up the missing contents that are visible in the reference images, we propose an asymmetric attention block that computes similarities between the peel pixels in the target and the non-hole pixels in the references in a non-local manner. With our attention block, our network can have an unlimited spatial-temporal window to the reference and inpaint holes with globally coherent contents. Our attention-based approach is also free from optical flow computations, thus we are able to handle challenging scenes with occlusions or large holes hardly modeled by flows.

In addition to video completion, our network is also applicable to image completion with reference images. Unlike the single-image inpainting task, we take a target image with additional reference images as input and we aim to complete the target image faithfully to the reference. It is useful for image editing as people tend to take multiple photographs of a scene in different angles and time. Using a group of photos, our method enables one to remove undesired objects without damaging the original contents. While this scenario can be considered as a special case of video completion with only a few frames, previous video completion methods have difficulties in handling this because computing optical flow between distant frames is challenging.

To summarize, we make the following contributions for accurate video completion:

- We propose the novel onion-peel network for flow-free video completion that works by the spatio-temporal attention mechanism.
- We propose the asymmetric attention block that allows non-local matching between the hole and the non-hole pixels.
- Our framework is applicable to the image completion guided by the reference images that is hard to be achieved by the previous image and video completion methods.
- We validate that our method produces comparable results to the state-of-the-art methods with fast computational time.

2. Related Work

Given the usefulness and difficulty of the problem, there is a large body of literature on the image and video completion. We review representative and recent studies that are closely related to our method. We refer the survey papers – image [9] and video [15] completion – for a comprehensive review.

2.1. Image Completion.

Example-based methods. Example-based methods rely on the self-examples to inpaint the missing regions [4, 11, 21]. In specific, a missing region is reconstructed by referencing the information outside the missing region. Furthermore, PatchMatch [1], a randomized search method for a fast approximation of the nearest neighbor field, accelerates the inpainting for the high quality results [24, 12]. A commercial solution in Adobe Photoshop belongs to this category.

Learning-based Methods. For learning-based methods, deep generative models are trained to synthesize missing contents in a data-driven manner. Generative adversarial networks [8] are often adopted for this purpose [25, 37, 14, 36]. To specifically handle incomplete data in the inpainting problem, the partial [20] and gated convolution [36] were proposed as an alternative to the normal convolution operator. We use the gated convolution layer in our network.

In this work, we tackles the multi-image completion that shares the similar concept of the example-based methods. The difference is that we find the example information from the reference images rather than from the same image.

2.2. Video Completion

Patch-based Methods. Wexler *et al*. [34] present a global optimization approach that alternates between the nearest neighbor search for the 3D spatio-temporal patches and the pixel reconstruction. Newson *et al*. [23] extends [34] by employing the PatchMatch [1] to accelerate the nearest neighbor search. Huang *et al*. [13] presented the non-parametric optimization method based on the dense optical flow field. The authors combine the spatial patch-based optimization and the flow field estimation to achieve spatially and temporally coherent inpainting. While this method is recognized as the state-of-the-art, the execution time is extremely long due to the computational complexity.

Compared to the above methods that perform the global optimization, our method runs an order of magnitude faster as it is based on feed-forward neural network. In addition, our data-driven approach is more flexible in handling unseen backgrounds.

[2] The term ”onion peel” originated from the initialization technique in [23, 24] for the randomized search through the PatchMatch [1]

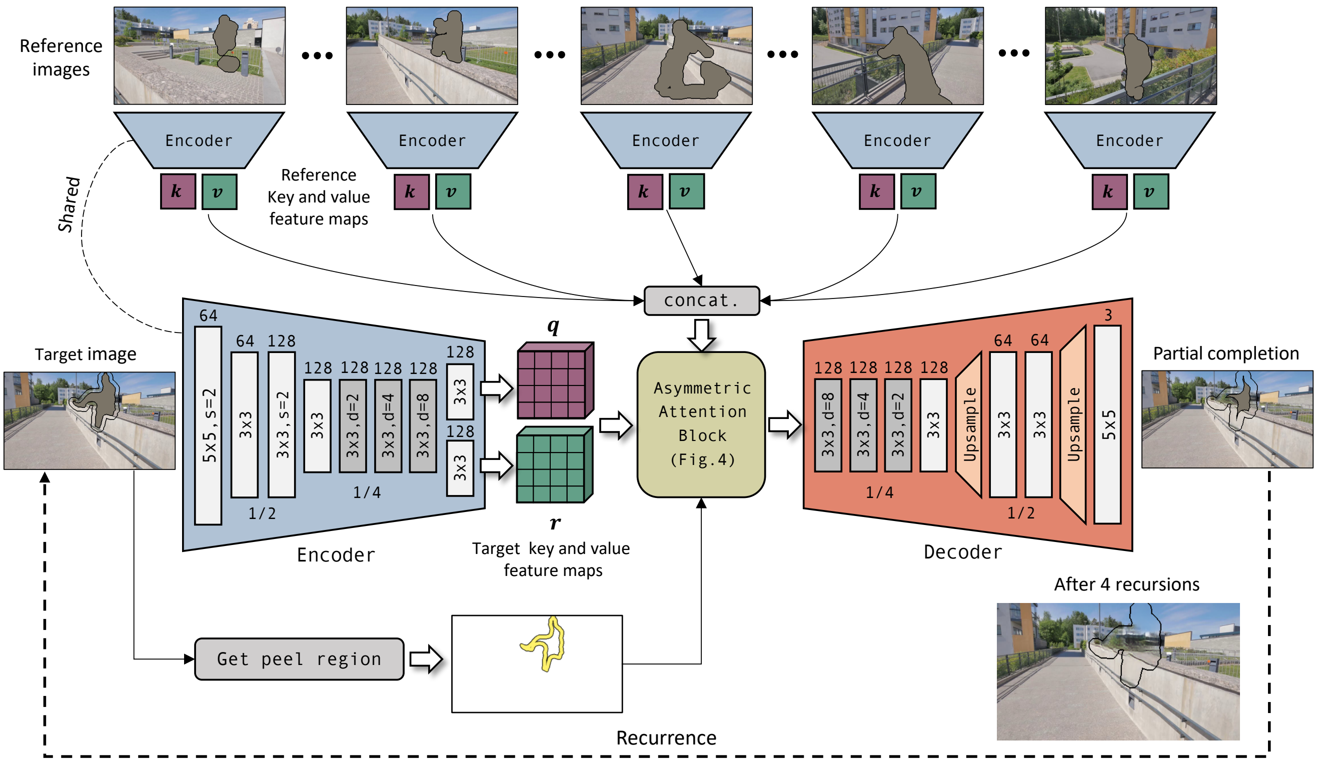


Figure 2: Overview of our network. We use the gated convolution layer as a basic building block. s and d indicate the stride and the filter dilation of the gated convolution layer. If not indicated, we set s=1 and d=1.

Data-driven Methods. Recently, data-driven methods for video completion through the deep networks have been proposed [30, 17]. In [30], the authors propose to use a combination of 3D and 2D CNNs, where the 3D sub-network is working with low-resolution video to reduce the computation. In [17], an image-based encoder-decoder network is designed to collect information from the nearby frames. Specifically, a set of the optical flow is estimated to aggregate information from the neighbor frames.

As mentioned earlier, those approaches have a limitation on their spatial-temporal window, which is crucial for the global coherency. Therefore, these methods are not expected to produce plausible results when a wide temporal window is required (*e.g.*, the object is big or moving slowly). On the other hand, our approach exploits a non-local pixel matching to have a global temporal window, thus it is free from such problems.

3. Onion-peel Network

The overview of our network for video completion is shown in Fig. 2. Given each video frame annotated with regions to be filled, the goal is to fill the hole region by looking at the other frames in the video for the right pixels. We call the image to be filled as the *target* image and the other images as the *reference* images. At each step, the onion-peel network fills only the *peel* region of the target image by referring to valid regions on the reference images. The peel region is defined as the set of pixels that are on the boundary of the hole region. We process the target image recursively to gradually fill the hole using the reference images.

In our network, the target image as well as the reference images are first embedded into the *key* and the *value* feature maps through the shared encoder network. We use the key features to find correspondences between the pixels in the target and the reference images. The key features on the peel region in the target image will be matched to every key feature on all valid (non-hole) regions of the reference images using the asymmetric attention block. The result of the matching is a spatio-temporal attention map, giving information about which pixel in which frame is important to fill the pixels in the peel region. The value is an abstract feature representation of an image to be used for reconstructing output images. Using the attention map from the key matching, we retrieve value features on the non-hole region of the reference frames according to the matching score. An example of the key matching and the value retrieval is shown in Fig. 3. The retrieved value features from the reference feature map are added to corresponding value features on

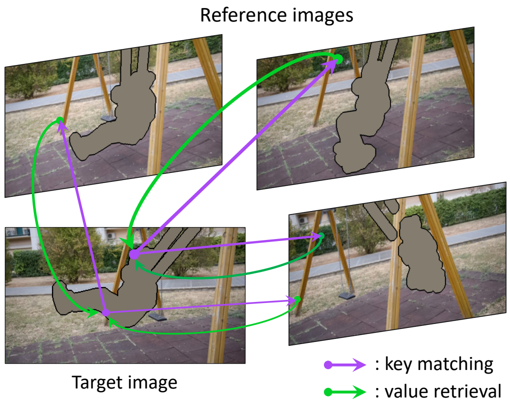


Figure 3: A visualization of attentions computed in our asymmetric attention block. The thickness of arrows represents the matching (softmax) score.

the peel region in the target feature map. Then the decoder takes the updated target value feature map with the peel region mask and inpaints the peel region of the target image.

3.1. Inference

We present two applications of the proposed onion-peel network: reference-guided image completion and video completion. For the image completion, reference images are embedded into key and value feature maps through the encoder. Then, the whole network (encoder, asymmetric attention block, and decoder) is applied to the target image recursively to fill the peel region iteratively until the hole is completely filled. For the video completion, the procedure of the image inpainting is looped over every frame sequentially. In the case of video, a set of the reference images is sampled from the video. In our implementation, we sampled every 5-th frame as the references.

The detailed procedure of the video inpainting is described in Alg. 1. Image completion is a specific case of the video completion with one target image and a set of reference images. Here, we define i-th video frame as X_i and its hole mask as H_i. The peel region P is defined as the hole pixels that are within the Euclidean distance of p to the nearest non-hole pixel. We set p to 8. The validity map V_i indicates genuine non-hole pixels that are not filled by the algorithm. Because the reference frames are not changed during inpainting a video, we run the encoder for the reference frames only once.

3.2. Network Design

Encoder. The input to the encoder network consists of an RGB image, the hole mask, and the validity map. The hole pixels on the RGB image are filled with the neutral (grey) values. These inputs are concatenated along the channel axis to form a 5-channel image before being fed into the first layer. We use the gated convolutional layer as a basic building block as it is useful for handling void information like a hole [36]. The encoder downsamples the feature map up to the 1/4 scale of the original size to secure the high-frequency details. The dilated convolution is used to further enlarge the receptive field size, and similar design choices can be found in recent image completion works [14, 37]. The encoder has two output heads each for the key and the value embedding, and this is implemented as two parallel gated convolutional layers.

Algorithm 1: Onion-peel hole filling.

```
Inputs : video $X$, hole $H$, validity $V$, peel width $p$
Output: completion video $X$
for $l$ in reference frame indices do
    $\mathbf{k}_l, \mathbf{v}_l = \text{Encoder}(X_l, H_l, V_l)$
end
for $i$ in target frame indices do
    $X_i^0 = X_i$
    $H_i^0 = H_i$
    $j = 0$
    while $\|H_i^j\|$ do
        $P^j = \text{get\_peel}(H_i^j, p)$
        $\mathbf{q}, \mathbf{r} = \text{Encoder}(X_i^j, H_i^j, V_i)$
        $\mathbf{z} = \text{AsymAtteBlk}(\mathbf{q}, \mathbf{r}, P^j, \mathbf{k}_{\forall\setminus i}, \mathbf{v}_{\forall\setminus i}, V_{\forall\setminus i})$
        $\hat{X}_i^{j+1} = \text{Decoder}(\mathbf{z})$
        $X_i^{j+1} = (1 - P^j) \odot X_i^j + P^j \odot \hat{X}_i^{j+1}$
        $H_i^{j+1} = H_i^j - P^j$
        $j = j + 1$
    end
end
```

The same encoder network is used for both the target image and the reference images to embed them into the key and the value feature maps. We denote the key and the value map from the target frame as $\mathbf{q}$ and $\mathbf{r}$, and the key and the value maps from the reference frames as $\mathbf{k}$ and $\mathbf{v}$.

Asymmetric Attention Block. The key and the value feature maps from the target image and the reference images are fed into the asymmetric attention block. A detailed illustration is shown in Fig. 4. In this module, inspired by recent attention mechanisms [29, 33], the key feature maps from the target and the reference are non-locally matched to compute a soft attention map over the reference images. Then, the attention map is used for the retrieval of the reference value features. As the network aims to reconstruct the peel region of the target image by looking the non-hole region of the reference images, we constrain the matching targets accordingly using the peel P and the validity map V. Specifically, before the key matching, pixels on the target and the reference embedding maps are indexed by P and

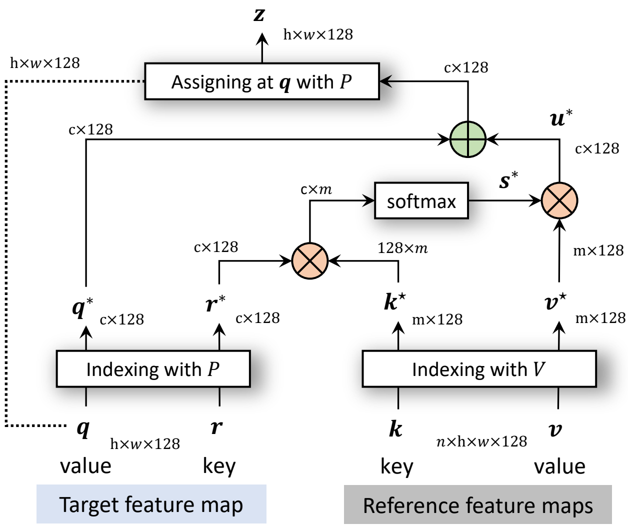


Figure 4: Asymmetric attention block. Variables after indexing are marked with high asterisks. h and w are the spatial resolution of feature maps, and n is the number of the reference frame. c and m is the number of pixels on the peel region and non-hole regions. $\otimes$ and $\oplus$ represent the matrix multiplication and element-wise addition, respectively.

V, respectively. This results in smaller matrices (marked with high asterisks in Fig. 4) where the column size is the number of pixels on the peel and the valid region, respectively. By doing so, we can specify the region of interests and the computational cost is significantly reduced as the key matching and value retrieval are implemented as a matrix inner-product. The matching scores $\mathbf{s}^*$ are cosine similarities normalized by the softmax function. The reference value features are then retrieved by the weighted summation with the computed matching scores. The retrieved values $\mathbf{u}^*$ are added to the target values $\mathbf{q}^*$ and assigned back to the original peel position (P). An example of this attention mechanism is visualized in Fig. 3.

Decoder. The decoder takes the output of the asymmetric attention block to reconstruct the peel region of the target image. We make the decoder to focus on the recovery of the peel region through our loss function which put more weight on that area (see Sec. 3.3 for detail). The decoder network is designed to be symmetric to the encoder network. The nearest neighbor upsampling is used to enlarge the feature map and the peel region of raw decoder output is combined with the non-hole region of the input target frame.

3.3. Training

Synthetic Training Data. Our framework is designed to retrieve the missing information from spatially and temporally distant pixels on the reference images. This design choice allows us to easily synthesize the training data. For each training sample, we prepare one target and four reference images taken from the same scene. The network inputs are images with holes made using masks, and the network is trained to reconstruct the image pixels of the hole region in the target image.

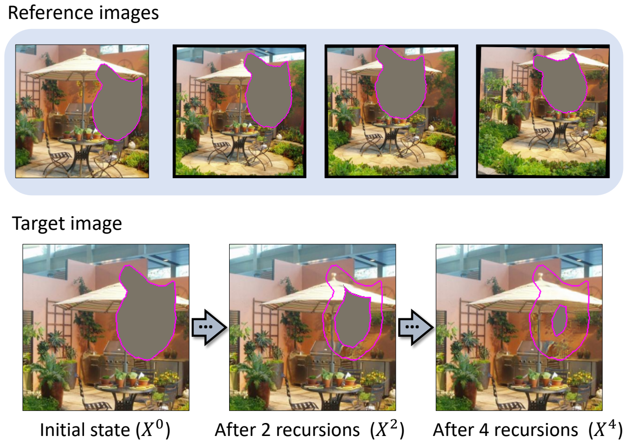


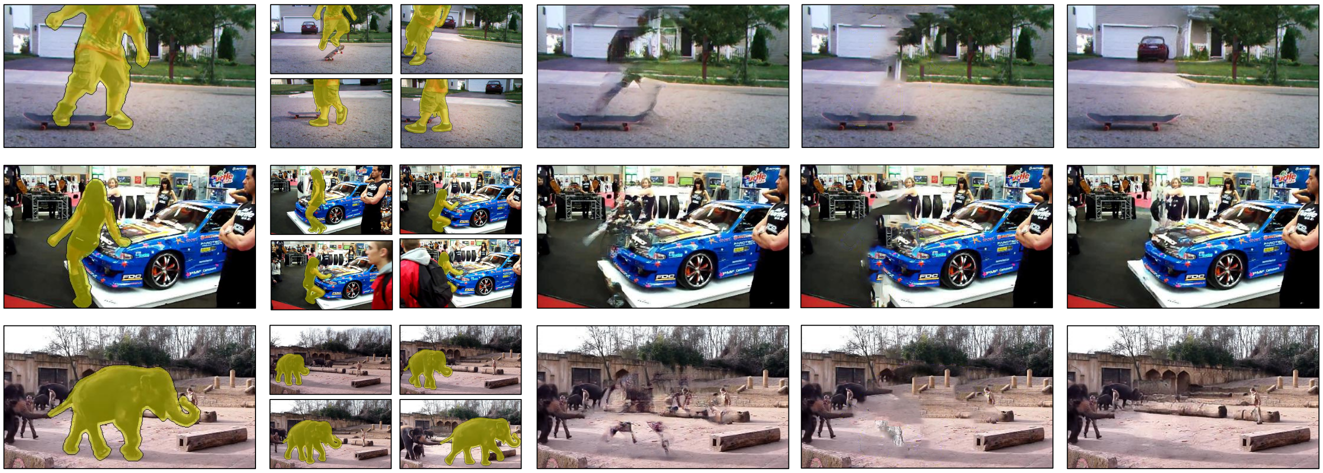
Figure 5: Example of the training sample and the intermediate outputs. All the intermediate outputs are used for computing losses.

To synthesize such training samples, two ingredients are required: a set of images and masks. For images, we used two sources: Places2 dataset [38] and YouTube videos that we collected ourselves. Places2 dataset [38] consists of 1.8 million images from 365 scene categories. With this dataset, we synthesized 5 background images by applying random affine transforms to an image such as rotation, translation, and cropping. The variety of contents in this dataset is useful for learning the generalization for various types of scenes.

In addition, we collected YouTube videos to learn a realistic scene transition between video frames. A total of 351 videos were collected from YouTube by searching for keywords like cities, nature, and world. Then, each video is chopped into short clips by detecting scene changes resulting in 7000 video clips. During the training, 5 frames are randomly sampled from each video clip. The source of training data is randomly picked at an equal chance.

For masks, we collected various object-shaped masks from the image datasets annotated with object masks (instance segmentation [6, 10] and salient object detection [27, 3]). Then, these masks are randomly moved and deformed by random affine transforms to simulate moving objects. An example of our training data is shown in Fig. 5.

Loss Functions. The loss function is designed to capture both the per-pixel reconstruction accuracy and the perceptual similarity. To achieve this, we minimize the L1 distance to the groundtruth in the pixel space and the deep feature



Input image

Reference images

Yu *et al*.

Content aware fill

Ours

Figure 6: Examples of image completion using a group of photos. The images are from Youtube-VOS [35].

space at each recurrence step. Example intermediate outputs are shown in Fig. 5.

The pixel losses are defined as follows:

$$\begin{aligned} \mathcal{L}_{\text{peel}} &= \sum_j \|P^j \odot (\hat{X}^j - Y)\|_1, \\ \mathcal{L}_{\text{valid}} &= \sum_j \|V \odot (\hat{X}^j - Y)\|_1, \end{aligned} \tag{1}$$

where j indexes over the number of the recurrence, $\hat{X}^j$ is the raw output of the decoder before the composition at j-th recursion, Y is the groundtruth, and $\odot$ indicates the element-wise multiplication.

Similarly, the perceptual loss is defined as the combination of the content and the style loss as follows:

$$\begin{aligned} \mathcal{L}_{\text{content}} &= \sum_{s=1}^{3} \sum_j \|\phi_s(X^j) - \phi_s(Y)\|_1, \\ \mathcal{L}_{\text{style}} &= \sum_{s=1}^{3} \sum_j \|\mathrm{G}(\phi_s(X^j)) - \mathrm{G}(\phi_s(Y))\|_1, \end{aligned} \tag{2}$$

where $\phi_s(\cdot)$ is the mapping to s-th pooled feature map of VGG-16 network [28] pre-trained on ImageNet, and $\mathrm{G}(\cdot)$ is a function for computing Gram matrix [7].

We compute the pixel losses directly on the decoder output and the perceptual loss after merging the peel pixels of the output with the non-hole region input. The total loss is the weighted summation of all the loss functions.

$$\begin{aligned} \mathcal{L}_{\text{total}} = {} & 100 \cdot \mathcal{L}_{\text{peel}} + \mathcal{L}_{\text{valid}} \\ & + 0.05 \cdot \mathcal{L}_{\text{content}} + 120 \cdot \mathcal{L}_{\text{style}} + 0.01 \cdot \mathcal{L}_{\text{tv}}, \end{aligned} \tag{3}$$

where $\mathcal{L}_{\text{tv}}$ is the total variation regularization term [16]. The balancing hyperparameters are adopted from [20].

Training Details. We used 256×256 images for the training and set the maximum number of the recursion as 5. For all experiments, the mini-batch size is set to 4 per GPU and Adam optimizer [18] is used. The learning rate starts with 1e-4 and divided by 10 every 100,000 iteration. The training takes about 5 days using four NVIDIA V100 GPUs.

3.4. Video Temporal Consistency

In the case of video completion, a video is processed frame-by-frame and the output videos often show flickering artifacts. To remedy this, we post-process our video outputs with an additional temporal consistency networks. The network is designed and trained for stabilizing videos inspired by [19] . Specifically, an encoder-decoder network equipped with a convolutional GRU at the core is trained to balance between the temporal stability with the previous frame and the perceptual similarity with the current frame. We modified the original method to match our need which is to stabilize the inpainted contents. Further details are covered in the supplementary materials and the effect of this post-processing is discussed in Sec. 4.3.

4. Experiments

In this section, we present our results on two experimental settings: reference-guided image completion and video completion.

We conducted our experiments using the test images/videos with a resolution of 424×240. Our code and model will be available online.N To obtain the results of other methods, we either used the available official codes or asked the authors for their results[3].

[3] Yu *et al*. [37]: https://github.com/JiahuiYu/generative_inpainting, Huang *et al*. [13]: https://filebox.ece.vt.edu/~jbhuang/project/vidcomp/index.html, and VINet [17]: requested to the authors.

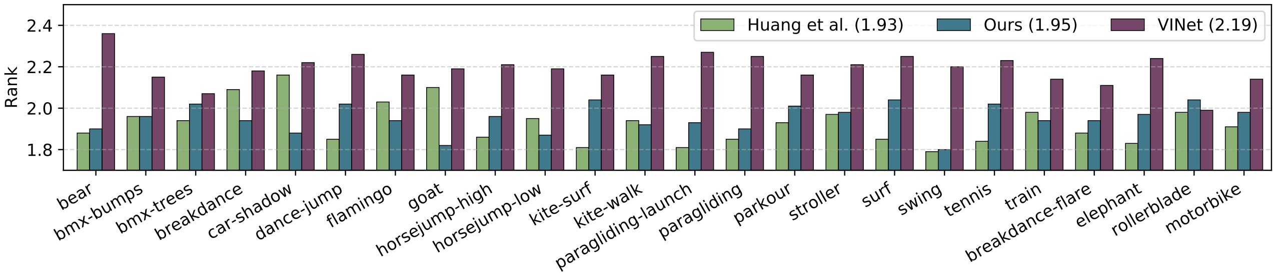


Figure 7: Result of our user study. The averaged ranks are shown in the figure legend.

4.1. Reference-guided Image Completion

In this setting, an algorithm aims to reconstruct the hole regions on an image using several reference images where the missing parts are partially visible.

Test Data. There are few related inpainting methods [31, 22, 5, 2] that leverage reference views, but those are either limited to specific configurations or patch-based methods. To the best of our knowledge, there is no deep learning method and standard test data specifically targeting this task. Therefore, we built a test set that consists of groups of photos, with each group having a set of photos of the same scene but in a different camera angle and time. Each photo also needs to be annotated with the undesired objects.

Rather than collecting images that meet the requirements from scratch, we start from a video dataset labeled with objects. We take Youtube-VOS [35] as the starting point. From there, we manually select videos that contain the appropriate scenes, and 5 frames are uniformly sampled from each video to form a group of photos. Among 5 images, one is used as the target image and other four images are used as the reference images.

Comparison. We compare our method against the state-of-the-art learning-based [37] and example-based [1] completion methods.

– Yu *et al*. [37] is a generative model for single-image completion built upon the generative adversarial network [8]. We used a model trained on Places2 data [38].

– Content aware fill [1], a feature built into Adobe Photoshop, is the most popular tool for the image completion. To allow it to take advantages of the reference frames, we first concatenated the target and reference images, then we let it fill all the holes using all the available cues.

As shown in Fig. 6, the previous methods have difficulties in inpainting a large missing area and show artifacts. In comparison, our method is able to deal with a large hole through the onion peeling. As expected, the single-image based method (Yu *et al*.) tends to hallucinate contents, and inpainted results are not consistent with the reference frames. The example-based method (content aware fill) also fails to deliver pleasing results as it sometimes pastes wrong contents due to errors in the patch matching. On the other hand, our method is able to fill holes with the actual contents by exploiting the reference images. In the top row of Fig. 6, the red car completely occluded by a boy is recovered.

4.2. Video Completion

In this section, our method is evaluated on the video completion task both quantitatively and qualitatively.

User Study. We conducted a user study to subjectively compare our method against the state-of-the-art optimization-based [13] and learning-based [17] video completion methods.

For the test videos, we used videos from DAVIS [26], where every video frame has pixel-wise annotation for an object. We used 24 videos from DAVIS with additional annotations that include the shadow regions provided by [13].

We conducted the user study through Amazon Mechanical Turk. For each video, the input and results from the 3 methods were shown to the participants, and the participants were asked to rank the results from 1 to 3. We allowed the participants to give a tie. Each test video was evaluated by 100 participants, and the averaged rank was as follows:

Huang *et al*. [13]	VINet [17]	Ours
1.93	2.19	1.95

(lower is better)

By the average rank, the quality of VINet [17] is not on par with the other two methods. The average ranks were very close between our method and [13], although [13] had a slight edge. Note that our methods runs more than 50 times faster than [13]. Per-video statistics are shown in Fig. 7 and some sample results are shown in Fig. 8.

Quantitative Evaluation. It is widely known that quantitatively evaluating the object removal is a hard task as the ground truth is not available. To make test videos with known backgrounds, we synthesized imaginary objects on top of the existing videos. To achieve this, we shuffled the pairs of video and mask from DAVIS. A total of 26 test videos were obtained this way, and used for comparing the pixel-wise reconstruction accuracy.

In Table 1, we quantitatively compare our method against the single-image [37] and the video completion [13] method. In addition to PSNR and SSIM, we provide Video

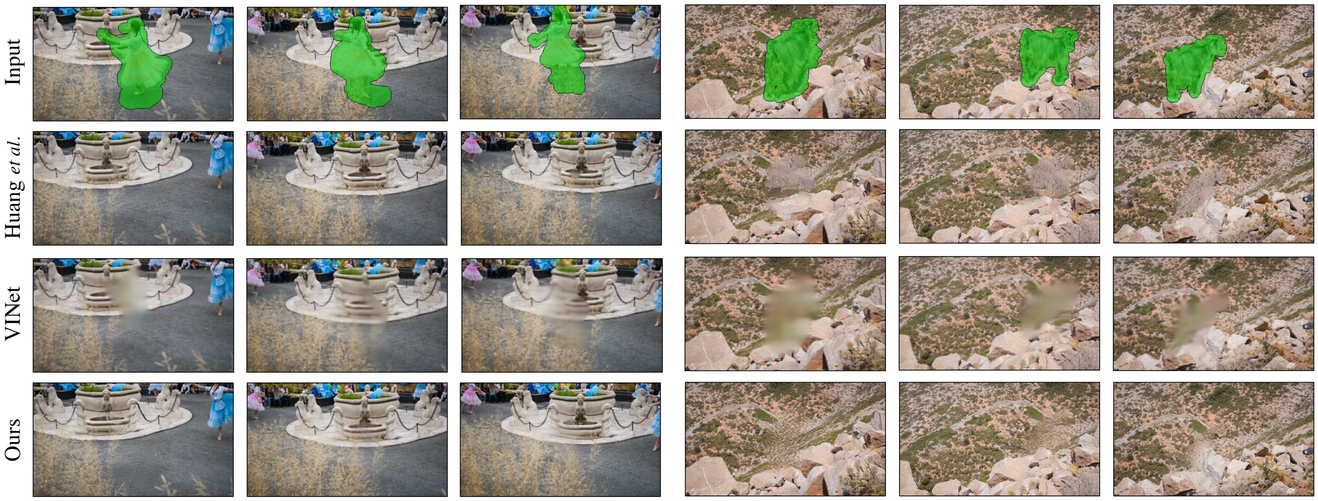


Figure 8: Visual comparison with video completion methods [13, 17] on DAVIS [26].

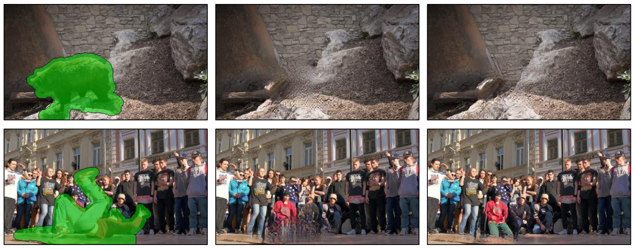


Figure 9: The effect of the onion peeling.

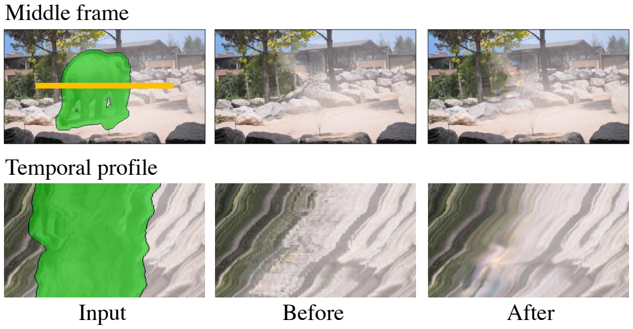


Figure 10: Our results before and after applying the post-processing. (top) the input and results at the middle frame. (bottom) temporal profile of the orange scan line. The slices of video frames were stacked along the vertical-axis.

Fréchet Inception Distance (VFID) [32] which is a video perceptual measure known to match well with human perception. Our method shows the best accuracy on all the measures. While the test videos are not real, we believe this synthetic experiments are helpful to understand the capability of different methods.

4.3. Analysis

Onion Peeling. The core of our algorithm is the progressive image completion called the onion peeling. To validate its effect, we trained a variant of our model that inpaints the whole mask at one-shot. The one-shot variant runs about 3× faster than our onion-peel network (p=8), but the output quality is not as good. One-shot model often fails to handle large holes as shown in Fig. 9.

	PSNR	SSIM	VFID	Time
Yu *et al.* [37]	23.34	0.824	0.059	**1.40**s
Huang *et al.* [13]	28.04	0.858	0.035	910s
Ours	**30.19**	**0.900**	**0.026**	15.8s

Table 1: Quantitative evaluation. The PSNR, SSIM and VFID are shown along with the execution time (sec/video).

Post-processing. For video completion, we run post-processing for temporal consistency as described in Sec. 3.4. In Fig. 10, we show the completion results before and after the post-processing. It clearly shows that the post-processing makes our results more temporally coherent as the temporal profiles are more smooth. However, as a side effect, the temporal consistency network tends to blur video frames (Fig. 10).

5. Conclusion

We have presented a novel application of deep network for the image/video completion. For the video completion, our method shows a competitive quality to the state-of-the-art optimization based method while running in a fraction of time. Without any modification, our network is applicable to the image completion guided by the reference frames, which is hardly achieved by the existing methods.

Acknowledgment. This work was supported by the ICT R&D program of MSIT/IITP (2014-0-00059) and the Technology Innovation Program (10073129) funded By the Ministry of Trade, industry & Energy (MOTIE, Korea).

References

[1] Connelly Barnes, Eli Shechtman, Adam Finkelstein, and Dan B Goldman. Patchmatch: a randomized correspondence algorithm for structural image editing. *ACM Transactions on Graphics (TOG)*, 28(3):24, 2009. 2, 7

[2] Connelly Barnes, Fang-Lue Zhang, Liming Lou, Xian Wu, and Shi-Min Hu. Patchtable: Efficient patch queries for large datasets and applications. *ACM Transactions on Graphics (TOG)*, 34(4):97, 2015. 7

[3] Ming-Ming Cheng, Niloy J Mitra, Xiaolei Huang, Philip HS Torr, and Shi-Min Hu. Global contrast based salient region detection. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 37(3):569–582, 2015. 5

[4] Antonio Criminisi, Patrick Pérez, and Kentaro Toyama. Region filling and object removal by exemplar-based image inpainting. *IEEE Transactions on image processing*, 13(9):1200–1212, 2004. 2

[5] Soheil Darabi, Eli Shechtman, Connelly Barnes, Dan B Goldman, and Pradeep Sen. Image melding: Combining inconsistent images using patch-based synthesis. *ACM Transactions on Graphics (TOG)*, 31(4):82–1, 2012. 7

[6] Mark Everingham, Luc Van Gool, Christopher KI Williams, John Winn, and Andrew Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2):303–338, 2010. 5

[7] Leon A Gatys, Alexander S Ecker, and Matthias Bethge. Image style transfer using convolutional neural networks. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 2414–2423, 2016. 6

[8] Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. Generative adversarial nets. In *Advances in neural information processing systems*, pages 2672–2680, 2014. 1, 2, 7

[9] Christine Guillemot and Olivier Le Meur. Image inpainting: Overview and recent advances. *IEEE signal processing magazine*, 31(1):127–144, 2014. 2

[10] Bharath Hariharan, Pablo Arbeláez, Lubomir Bourdev, Subhransu Maji, and Jitendra Malik. Semantic contours from inverse detectors. In *IEEE International Conference on Computer Vision (ICCV)*, pages 991–998. IEEE, 2011. 5

[11] James Hays and Alexei A Efros. Scene completion using millions of photographs. *ACM Transactions on Graphics (TOG)*, 26(3):4, 2007. 2

[12] Jia-Bin Huang, Sing Bing Kang, Narendra Ahuja, and Johannes Kopf. Image completion using planar structure guidance. *ACM Transactions on graphics (TOG)*, 33(4):129, 2014. 2

[13] Jia-Bin Huang, Sing Bing Kang, Narendra Ahuja, and Johannes Kopf. Temporally coherent completion of dynamic video. *ACM Transactions on Graphics (TOG)*, 35(6), 2016. 1, 2, 6, 7, 8

[14] Satoshi Iizuka, Edgar Simo-Serra, and Hiroshi Ishikawa. Globally and Locally Consistent Image Completion. *ACM Transactions on Graphics (Proc. of SIGGRAPH 2017)*, 36(4):107:1–107:14, 2017. 1, 2, 4

[15] Shachar Ilan and Ariel Shamir. A survey on data-driven video completion. In *Computer Graphics Forum*, volume 34, pages 60–85. Wiley Online Library, 2015. 2

[16] Justin Johnson, Alexandre Alahi, and Li Fei-Fei. Perceptual losses for real-time style transfer and super-resolution. In *European conference on computer vision*, pages 694–711. Springer, 2016. 6

[17] Dahun Kim, Sanghyun Woo, Joon-Young Lee, and In So Kweon. Deep video inpainting. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, 2019. 1, 3, 6, 7, 8

[18] Diederik Kingma and Jimmy Ba. Adam: A method for stochastic optimization. In *International Conference on Learning Representations*, 2015. 6

[19] Wei-Sheng Lai, Jia-Bin Huang, Oliver Wang, Eli Shechtman, Ersin Yumer, and Ming-Hsuan Yang. Learning blind video temporal consistency. In *European Conference on Computer Vision*, 2018. 6

[20] Guilin Liu, Fitsum A. Reda, Kevin J. Shih, Ting-Chun Wang, and Andrew Tao Bryan Catanzaro. Image inpainting for irregular holes using partial convolutions. In *IEEE International Conference on Computer Vision (ICCV)*, 2018. 1, 2, 6

[21] Yunqiang Liu and Vicent Caselles. Exemplar-based image inpainting using multiscale graph cuts. *IEEE transactions on image processing*, 22(5):1699–1711, 2013. 2

[22] Bryan Morse, Joel Howard, Scott Cohen, and Brian Price. Patchmatch-based content completion of stereo image pairs. In *2012 Second International Conference on 3D Imaging, Modeling, Processing, Visualization & Transmission*, pages 555–562. IEEE, 2012. 7

[23] Alasdair Newson, Andrs Almansa, Matthieu Fradet, Yann Gousseau, and Patrick Prez. Video inpainting of complex scenes. *SIAM Journal on Imaging Sciences, Society for Industrial and Applied Mathematics*, 7(4):1993–2019, 2014. 2

[24] Alasdair Newson, Andrés Almansa, Yann Gousseau, and Patrick Pérez. Non-local patch-based image inpainting. *Image Processing On Line*, 7:373–385, 2017. 2

[25] Deepak Pathak, Philipp Krahenbuhl, Jeff Donahue, Trevor Darrell, and Alexei A. Efros. Context encoders: Feature learning by inpainting. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 1, 2

[26] Federico Perazzi, Jordi Pont-Tuset, Brian McWilliams, Luc Van Gool, Markus Gross, and Alexander Sorkine-Hornung. A benchmark dataset and evaluation methodology for video object segmentation. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016. 7, 8

[27] Jianping Shi, Qiong Yan, Li Xu, and Jiaya Jia. Hierarchical image saliency detection on extended cssd. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 38(4):717–729, 2016. 5

[28] Karen Simonyan and Andrew Zisserman. Very deep convolutional networks for large-scale image recognition. *arXiv preprint arXiv:1409.1556*, 2014. 6

[29] Sainbayar Sukhbaatar, Jason Weston, Rob Fergus, et al. End-to-end memory networks. In *Advances in neural information processing systems*, pages 2440–2448, 2015. 4

[30] Chuan Wang, Haibin Huang, Xiaoguang Han, and Jue Wang. Video inpainting by jointly learning temporal structure and spatial details. In *Proceedings of the 33th AAAI Conference on Artificial Intelligence*, 2019. 1, 3

[31] Liang Wang, Hailin Jin, Ruigang Yang, and Minglun Gong. Stereoscopic inpainting: Joint color and depth completion from stereo images. In *2008 IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–8. IEEE, 2008. 7

[32] Ting-Chun Wang, Ming-Yu Liu, Jun-Yan Zhu, Guilin Liu, Andrew Tao, Jan Kautz, and Bryan Catanzaro. Video-to-video synthesis. In *Advances in neural information processing systems*, 2018. 8

[33] Xiaolong Wang, Ross Girshick, Abhinav Gupta, and Kaiming He. Non-local neural networks. In *IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2018. 4

[34] Yonatan Wexler, Eli Shechtman, and Michal Irani. Space-time video completion. In *Proceedings of IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, 2004. 2

[35] Ning Xu, Linjie Yang, Dingcheng Yue, Jianchao Yang, Yuchen Fan, Yuchen Liang, and Thomas Huang. Youtube-vos: A large-scale video object segmentation benchmark. In *arXiv preprint arXiv:1809.03327*, 2018. 6, 7

[36] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang. Free-form image inpainting with gated convolution. *arXiv preprint arXiv:1806.03589*, 2018. 1, 2, 4

[37] Jiahui Yu, Zhe Lin, Jimei Yang, Xiaohui Shen, Xin Lu, and Thomas S Huang. Generative image inpainting with contextual attention. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 5505–5514, 2018. 1, 2, 4, 6, 7, 8

[38] Bolei Zhou, Agata Lapedriza, Aditya Khosla, Aude Oliva, and Antonio Torralba. Places: A 10 million image database for scene recognition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2017. 5, 7