https://doi.org/10.2352/EI.2023.35.9.IPAS-285

ORCA: An End-to-End Video Object Removal framework with Cropping interested region and Quality Assessment

Minseong Son, Hansol Lee, Sungkeun Kwak, Jihwan Woo
AI Research, CJ OliveNetworks, Republic of KOREA

Abstract

Recently, various types of Video Inpainting models have been released. Video Inpainting is used to naturally erase the object you want to erase in the video. In order to use inpainting models, we usually need frames extracted from a video and masks and most people make these data manually. We propose a novel End-to-End Video Object Removal framework with Cropping Interested Region and Video Quality Assessment (ORCA). ORCA is built in an end-to-end way by combining the Detection, Segmentation, and Inpainting modules. The characteristics of proposed framework focus going through the cropping step before inpainting step. In addition, we propose our own video quality assessment since ORCA use two models for inpainting. Our new metric indicates the higher quality of the results between two models. Experimental results show the superior performance of the proposed methods.

Introduction

Object removal is a complex set of modules that removes undesired objects within images or videos. In the US, movies and televisions are rated by entities such as Motion Picture Association and TV Parental Guidelines, respectively. Likewise, Republic of Korea imposes even more strict restrictions when it comes to rating television contents. Inappropriate materials for children such as cigarettes, liquor, sexual materials, etc needs to be blurred or erased to pass the Parental Guidelines. Even commercial logo like Nike should be covered up to be able to go on the air. Fig. 1(a) is a cropped still image from some movie. For this movie to be televised or to be served in OTT media services, cigarettes in movies should be blurred. We go step further to propose an end-to-end model that can remove cigarettes (Fig. 1(b)). In fig. 1(c), commercial logo on the baseball cap is covered by a black tape and aesthetically, it does not look good. Our end-to-end model can automatically remove corresponding logo with short time (Fig. 1(d)). Blurring or removing inappropriate objects is an extremely labor intensive task. Therefore, the movie&tv industries have growing needs of automating this whole process. This growing demand led us to develop an end-to-end model which can automatically remove target objects in videos. Therefore, we propose our automated end-to-end model consists of Detection, Segementation, and Inpainting modules that can automatically remove specific object (Fig. 2).

The first step of our end-to-end model is detection. Our detection model uses YOLOv5 [1] as a baseline. The next step of our end-to-end model is segmentation. Unfortunately, there were cases in which objects go undetected. To minimize this risk, we had to find a segmentation model that can also track objects at the same time. We were able to find SiamMask [2] that can track objects and simultaneously can create detailed segmentation masks. The last step of our end-to-end model is inpainting. We use an ensemble model that comprises of STTN [3] and LaMa [4] models. We carefully devised a metric to compare the performances of STTN and LaMa under given conditions and choose either one that performs better. We also formulated a new cropping method to improve inpainting results.

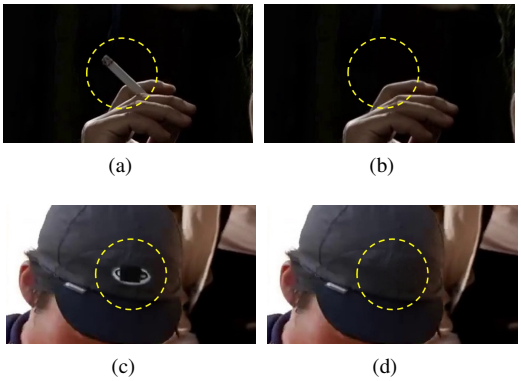


Figure 1. *The proposed model is able to remove uncomfortable objects in the videos, such as the left figures ((a), (c)), to look good like the right figures ((b), (d)).*

Related Works

Discrete efforts have been made in object detection, segmentation and inpainting. However, there has not been any effort to merge these modules together to accomplish automated object removal process. The demand for end-to-end object removal model in tv&movie industry kept growing. This growing demand led us to suggest an end-to-end model for object removal.

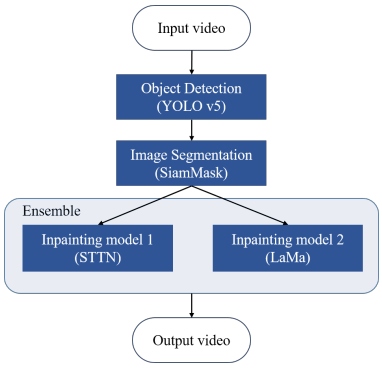


Figure 2. *The flowchart of the proposed model. Three modules are connected in one pipeline to compose an end-to-end model.*

Detection methods

Great deal of studies on object detection has been conducted. Fast R-CNN [5] used region based convolution network to work out detection back in 2015. Fast R-CNN uses ROI pooling with fixed feature map to extract feature vectors. Faster R-CNN [6] improved Fast R-CNN by introducing Region Proposal Network to efficiently compute Region of Interest. R-CNN based models have been studied rigorously and in 2018, META introduces Detectron [7] that includes implementation of Mask R-CNN, RetinaNet, Faster R-CNN and other detection algorithms.

Along with Detectron series, another branch of object detection called YOLO has been developed. Using the idea of speed-accuracy tradeoff, YOLO series were able to devise detection algorithm with great efficiency and accuracy. In 2021, YOLO Detector [8] was switched to anchor-free and used decoupled head technique. Although Detectron2 had edge over YOLOv5 in accuracy but it required more computing power and was slower. Speed mattered in our end-to-end model. Therefore, we decided to use YOLOv5 as our baseline model for the Detection Module.

However, YOLOv5 needed some adjustments because there were some limitations. First, YOLOv5 does not work well with high-resolution videos such as Full HD and 4K videos. Typical YOLOv5 deals with this problem by training the model with higher resolution in the beginning. But, in order to train the model with Full HD or 4K images, it required a very large GPU. To fix this issue, we combined techniques like patch sliding method and resizing. Second, YOLOv5 cannot detect objects with very small size and fast moving objects. Therefore, we had to develop new methods on top of YOLOv5 to enhance the overall quality of detection. In order to better detect small-sized objects like cigarettes, we took extra careful approach to create our custom dataset; for example, we used images of an object viewed in multiple angles and different distances. Moreover, in order to detect objects in motion better, we used interpolation and adopted object tracking method. In this way, we were able to detect fast moving objects.

Segmentation methods

Much studies has been conducted on segmentation. One of many segmentation model such as OCRNET [9] classifies the object to certain region by using object-contextual representation. Google Inc. also published Deeplab for segmentation tasks. It uses Atrous Spatial Pyramid Pooling Module in which at the end of the CNN atrous convolution is applied. [10] We implemented multiple segmentation models using custom dataset. However, there were difficulties adding object classes that were not used in the original model. While it showed great performance with the benchmark dataset, it showed poor result when custom dataset was used. Additionally, we had to use a model that can track objects to complement YOLOv5 which could not detect fast moving objects.

For the purpose of our model, SiamMask [11] was a really good fit. SiamMask was able to track objects and showed great performance in object segmentation. As stated before, YOLOv5 fails to detect object with fast movement due to heavy motion blur. By using SiamMask, we can track objects so we do not lose any frames with object going undetected. Although SiamMask does a good job in tracking objects and making segmentation masks, there were still cases in which the object cannot be tracked. In addition, if there were multiple objects in a single frame it stumbles. Therefore, We added a module to resolve these obstacles. The boundary box information from the previous detected frame is used along with the boundary box information from the next detected frame. Considering the time span and the boundary box information of the past and the future we were able to predict the boundary box in the current working frame. Then, the predicted boundary box goes into the SiamMask to create segmentation masks for the later module; the Inpainting Module.

Inpainting methods

Adobe has spot healing brush tool that can remove unwanted objects. In addition, Samsung Galaxy Phones have functions to remove unwanted object in photos. These works well under certain conditions but fails to provide good results in many cases. Our end-to-end model should work under every given condition and moreover it should match the quality of broadcasting requirements. Therefore, we had to make extra effort to improve the Inpainting Module. NVIDIA corporation proposed partial convolution instead of traditional convolution network in which convolution is applied only to the hole that needs to be inpainted. [12] Adobe further developed image inpainting using gated convolution. [13] It proposes feature gating mechanism for both channel and spatial locations. Implementing many other models, we found that STTN [14] and LaMa [15] give best results for image inpainting.

Between the two models, it was hard to distinguish which model is better. Therefore, we created an ensemble modeling to get better inpainting result. We devised a new metric called No-Reference Video Quality Assessment(NR-VQA) metric that integrates the Variance of Laplacian [16] and BRISQUE [17] to get better inpainting results.

Furthermore, we introduced the use of new cropping method in inpainting module. By using cropping, we added more flexibility to the use of inpainting model. The inpainting model can only intake image size of 432×240. Cropping helps us to determine the best 432×240 size image and segmentation mask that gives us the best result. Moreover, existing inpainting models use global spatial information in which the model takes account every pixels of an image. By examining every pixel in the image, it can wrongfully fill in the hole region with awkward colors or patterns. In order to sustain the homogeneity of the inpainting region, our model uses local spatial information to fill in the hole. In this way, the model only takes pixel information adjacent to the hole that needs to be filled in. Therefore, it gives more smooth and natural inpainting results. The inpainting result that uses cropping and local spatial information showed better result quantitatively and qualitatively compared to the result of original models.

Proposed Approach

Overview. The process of erasing objects in a video consists of various modules (Fig. 3). First, input video is converted to frames, and it goes through three main steps. (1) Detection : This is a step of detecting the object to be erased in each frame. We use a YOLOv5 trained by a custom dataset. Through the detection process, we can get the object's bounding box coordinates. (2) Segmentation: After detection, the object in the bounding box is segmented. In this process, we use the SiamMask [2]. For using SiamMask, we need to input annotation box. We use the bound-

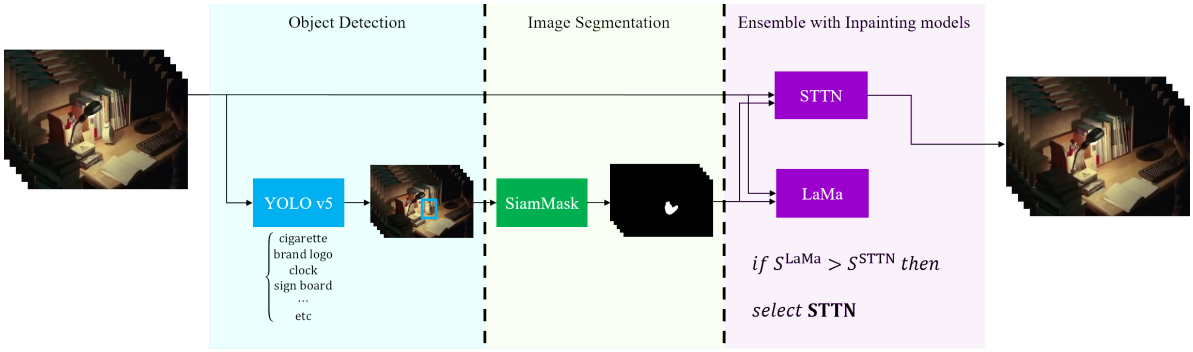


Figure 3. *The architecture of our model. YOLOv5 takes frames of an original video as input, and feeds their results to SiamMask to output the mask images. Then, two different inpainting modules take original frames and their mask images as input and produce respective results. The best of these is determined by the final result using proposed NR-VQA.*

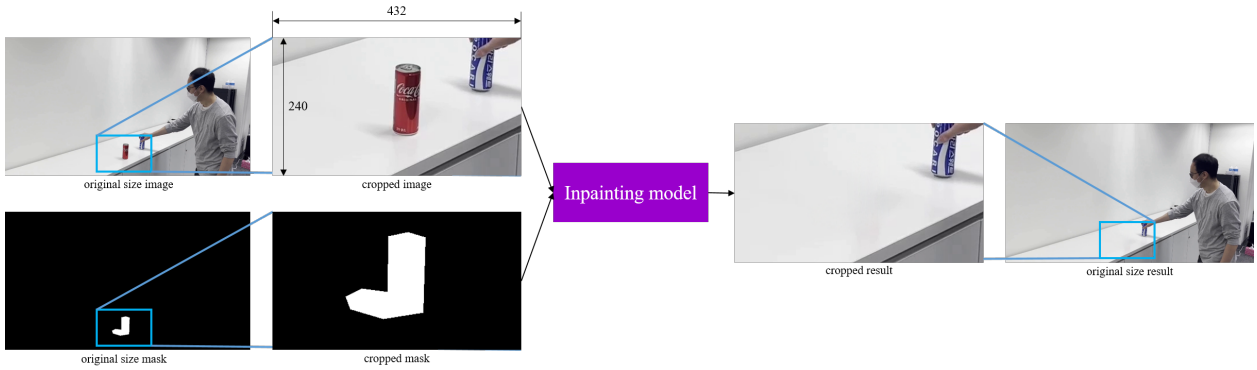


Figure 4. *To maintain the original quality of the videos, we address our cropping approach, which takes a small region of the image to be removed as input of the inpainting module.*

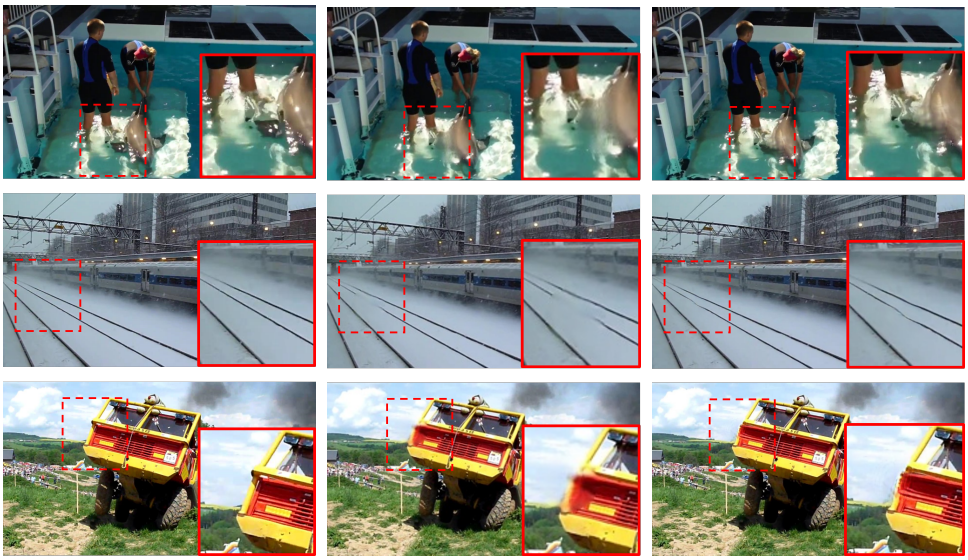
ing box obtained at the prior step and consequently we can get the mask images in this step. (3) Inpainting: Then, based on the coordinates of the centroid of the masked area, cropped images and cropped masks are obtained and these are used as input for both inpainting models; STTN [3] and LaMa [4]. Finally, the better result is selected by comparing the quality of each model result.

Cropping Strategy. We propose to crop the frames and masks as specific size of images with the centroid of the masked area as the center (Fig. 4). We use them as an input for the inpainting models. 2-dimensional coordinates in the cropped region (x, y) is expressed below. $(\bar{x}, \bar{y})$ is the coordinates of the centroid of masked area. W, H are width and height of the frame.

$$\max(0, \bar{x} - \frac{W}{2}) \leq x \leq min(W, \bar{x} + \frac{W}{2}),$$
$$\max(0, \bar{y} - \frac{H}{2}) \leq y \leq min(H, \bar{y} + \frac{H}{2}). \quad (1)$$

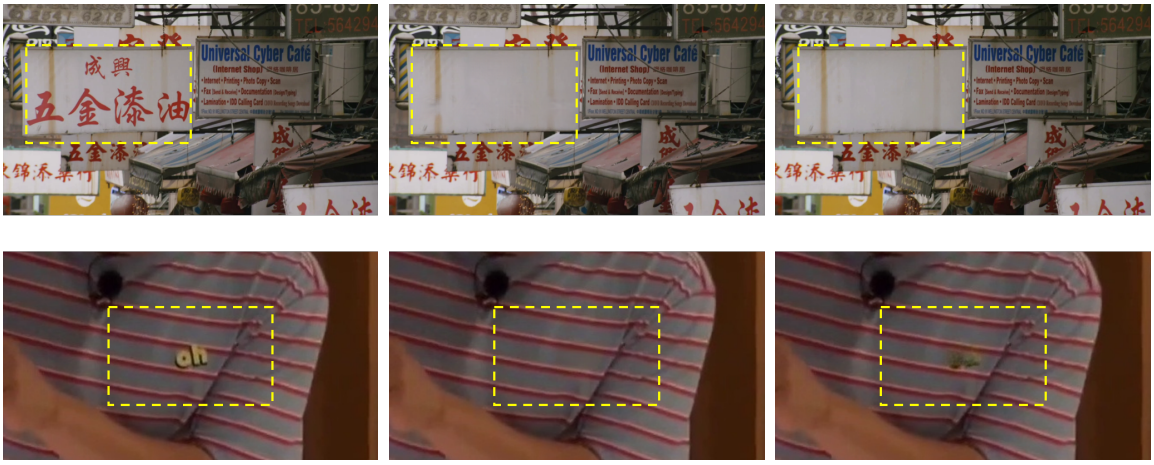
There are two advantages of cropping images (Fig. 3). First, it is possible to maintain the high quality of the video. If the input image is larger than the designated input size of the inpainting model, it can not be used. In this case, the methods to solve this problem are resizing or cropping. Among them, we decide to use the cropping method. We crop the image to fit the input size. After inpainting, newly created hole regions is pasted it back into the cropped hole regions. Using this method, we can maintain the quality of the frames. Second, the model uses local spatial information around the object to be removed, rather than global spatial information. To compare the inpainting results of cropped input images with the results of uncropped input images, we used metric called PSNR and SSIM [18]. If uncropped images are inputs for the model, we resize them to 432×240 and proceed with inpainting. As a result, the size of the result is also 432×240. To compare the results with other inpainting results using cropped input, we resize the result images to use cropped inputs of 432×240. Then, we compare the results quantitatively using two metrics, PSNR and SSIM.

No-Reference Video Quality Assessment. In the inpainting step, two models are used and it is impossible to determine which one is better. This is because the superiority of the results varies depending on the objects (Fig. 6). Better results should be selected as the final results. For this, we create our own video quality assessment metric using the image quality assessment metric. The result videos are evaluated with a combination of the Variance of Laplacian [16], which indicates the degree of blur of images, and BRISQUE [17], which is used as an image quality evaluation metric. We use a gap between the measured values. Because the larger the quality gap between neighboring frames, the more awkward it is. We measure the metrics for N result images from M inpainting models and we define the Variance of Laplacian α_n^m and BRISQUE β_n^m where $m = 1, 2, ..., M$, $n = 1, 2, ..., N$. The gap between metric values of neighboring frames can be expressed using frame number n and inpainting model m:



(a) Original Image (b) Input: Uncropped Image (c) Input: Cropped Image

Figure 5. *Experiment results to demonstrate the effectiveness of our cropping approach on the Youtube-VOS videos. The yellow box in figure (a) shows which region is cropped.*



(a) Original Image (b) STTN (c) LaMa

Figure 6. *Experiment results to compare the results of two inpainting modules on a test video.*

$$\Delta\alpha_n^m = \alpha_n^m - \alpha_{n-1}^m, \quad \Delta\beta_n^m = \beta_n^m - \beta_{n-1}^m. \tag{2}$$

We set $M = 2$ since we use two models STTN and LaMa. The gap metric of each model can be expressed as $\Delta\alpha_n^{\text{STTN}}$, $\Delta\alpha_n^{\text{LaMa}}$, $\Delta\beta_n^{\text{STTN}}$, $\Delta\beta_n^{\text{LaMa}}$. Maximum values of each value can be expressed as $\alpha_{max}^{\text{STTN}}$, $\alpha_{max}^{\text{LaMa}}$, $\beta_{max}^{\text{STTN}}$, $\beta_{max}^{\text{LaMa}}$. Finally, our proposed video quality assessment metric is computed in the form of weighted sum of α_{max}^m and β_{max}^m.

$$S^{\text{STTN}} = \alpha_{max}^{\text{STTN}} + w * \beta_{max}^{\text{STTN}}, \tag{3}$$

$$S^{\text{LaMa}} = \alpha_{max}^{\text{LaMa}} + w * \beta_{max}^{\text{LaMa}}, \tag{4}$$

where w denotes weight.

After calculating S^{STTN} and S^{LaMa}, we compare the values sequentially. First, if $\alpha_{max}^{\text{STTN}}$ is larger than $\alpha_{max}^{\text{LaMa}}$ and $\beta_{max}^{\text{STTN}}$ is larger than $\beta_{max}^{\text{LaMa}}$, we choose STTN results. However, we choose LaMa results in the opposite case. If both conditions are not met, we compare S^{STTN}, S^{LaMa} to check which result is better. The model with smaller metric values is better than the other one. The process for selecting the better result is shown as Alg. 1.

Experiments and Results

Dataset

Youtube-VOS. We use Youtube-VOS dataset to check whether the inpainting model using cropped input images generate better results. We use the validation set of Youtube-VOS(2019). However in the case of Youtube-VOS, it gets only one mask from the first frame. Therefore, we need to make a masks for every frame and we make masks by placing a random,

Table 1. Comparison of evaluated Mean Opinion Score (MOS) for the results of each inpainting module. Higher scores are highlighted in bold.

Methods	#1	#2	#3	#4	#5	#6	#7	#8	#9	#10	#11	#12	#13	#14	#15	#16	#17	#18	#19	#20
STTN [3]	**3.56**	3.44	3.5	**3.53**	**3.85**	**4.18**	3.88	**3.15**	**4.24**	**4.29**	**3.85**	**4.06**	**4.29**	**4**	**3.76**	**4.24**	**4.41**	**3.06**	**3.71**	**4.12**
LaMa [4]	1.53	**4.12**	**4**	2.97	2.24	2.12	**4.12**	1.56	1.47	1.47	1.03	1.03	1.12	1.59	1.38	1.71	1.41	1.29	1.26	1.97

Algorithm 1 Self-Selection using Video Quality Assessment

Input: $\alpha_{max}^{STTN}, \alpha_{max}^{LaMa}, \beta_{max}^{STTN}, \beta_{max}^{LaMa}, S^{STTN}, S^{LaMa}$

Output: selected model result $\hat{sel}$

Calculate $\alpha_{max}^{STTN}, \alpha_{max}^{LaMa}, \beta_{max}^{STTN}, \beta_{max}^{LaMa}, S^{STTN}, S^{LaMa}$.
if $\alpha_{max}^{STTN} > \alpha_{max}^{LaMa}$ *and* $\beta_{max}^{STTN} > \beta_{max}^{LaMa}$ **then**
    $\hat{sel} \leftarrow STTN$
else
    if $\alpha_{max}^{LaMa} > \alpha_{max}^{STTN}$ *and* $\beta_{max}^{LaMa} > \beta_{max}^{STTN}$ **then**
        $\hat{sel} \leftarrow LaMa$
    else
        if $S^{LaMa} > S^{STTN}$ **then**
            $\hat{sel} \leftarrow STTN$
        end if
        if $S^{STTN} > S^{LaMa}$ **then**
            $\hat{sel} \leftarrow LaMa$
        end if
    end if
end if

Table 2. Results of our cropping approach on Youtube-VOS dataset.

Methods	PSNR	SSIM (%)
STTN [3]	34.69	96.13
Ours (cropping)	**44.02**	**99.49**

Table 3. Accuracy (%) of our ensemble with two inpainting modules on the 26 test videos.

Methods	LaMa	STTN	Total
Pech et al. [16]	16.67	**95**	76.92
BRISQUE [17]	**100**	20	38.46
Ours (proposed)	**100**	85	**88.46**

fixed-size circle at an arbitrary position within the frame.

Inpainting Results. We generate STTN [3] and LaMa [4] results from 26 videos. In addition, we conduct Mean Opinion Score (MOS) to evaluate which results are better. Table 1 shows 20 MOS results among all inpainting results and based on this, we set the ground truth for each result that are more selected between STTN and LaMa results. This survey is evaluated by 34 participants.

Experiment Results

Influence of Cropping Interested Region. The inpainting results for the cropped input fill the hole more naturally with the surrounding background. The improved results by using cropping method are shown in fig. 5. As a result, we can see the inpainted

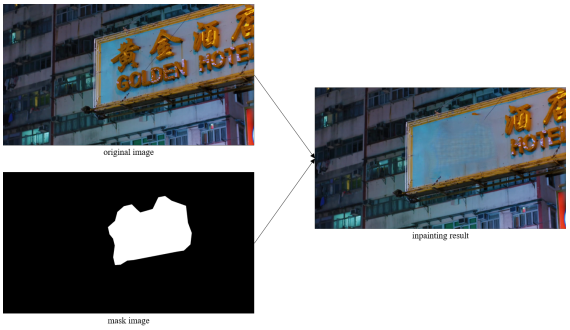


Figure 7. *A failure case. The left figures are the original image and the corresponding mask image, respectively, and the right figure is the result.*

area is more clear. Table 2 presents the comparison results of PSNR and SSIM. According to the table, our proposed method increases PSNR and SSIM to 44.02 and 99.49 respectively. It demonstrates cropping input images can improve the quality of inpainting results.

Comparison of STTN and LaMa. Depending on quality assessment metrics, we select the better results of each pair of STTN and LaMa results. As a result of the MOS, the evaluation score of STTN were higher than those of LaMa (Table 1). Inpainting results with a higher score for LaMa were 6 of 26. Table 3 shows the accuracy of selecting a better result. Our proposed metric achieved 88.46% accuracy. Additionally, our proposed metric almost predicted correct answer for all the questions regardless of models (LaMa : 100 %, STTN : 85 %). Other metrics showed a strong bias for only one model. Consequently, it can be seen that our metric increased the video quality using our newly proposed video quality assessments.

Conclusion

In this paper, we presented a new model that automatically finds and blurs or removes some objects in the videos. The proposed model has the advantage of processing three successive components end-to-end. We described the cropping approach, which cuts out a small region including the object to be processed and merges it into the same location on the original image after passing through our model. This method was able to maintain the original quality of the images. Furthermore, the ensemble of the two state-of-the-art approaches in the inpainting task is leveraged to refer to richer resourced, and our model adopts more natural results using NR-VQA metrics. We evaluated the performance of the proposed model visually and quantitatively and demonstrated the efficient removal of some objects such as signboards, brand logo, and cigarettes.

Even though the cropping technique described above provides a great strength to secure the original resolution of the images, it is accompanied by a limitation that the larger the hole size, the worse the result is. As can be seen in fig. 7, when the

hole of the mask exceeds a certain size, the inpainting result is very sloppy. Therefore, it is necessary to do further researches on the inpainting task such that it is not affected by the size of the object to be removed. Our end-to-end model is difficult to handle high-quality videos such as 4k and 8k, because it is a heavy model with three modules combined and each module is calculated with many parameters. Since recent content often requires high resolution, we note that this is important to improve for actual service delivery. We will focus on expanding to the powerful model that can provide a service to users. Furthermore, our future work notes an extension to the end-to-end model, which adds the replacement task, removes the object and then replaces the new object naturally on it.

References

[1] G. Jocher *et al.*, "ultralytics/yolov5: v6.1," Feb. 2022. `https://doi.org/10.5281/zenodo.6222936`.

[2] Q. Wang, L. Zhang, L. Bertinetto, W. Hu, and P. H. Torr, "Fast online object tracking and segmentation: A unifying approach," in *Proceedings of the IEEE/CVF conference on Computer Vision and Pattern Recognition*, pp. 1328–1338, 2019.

[3] Y. Zeng, J. Fu, and H. Chao, "Learning joint spatial-temporal transformations for video inpainting," in *European Conference on Computer Vision*, pp. 528–543, Springer, 2020.

[4] R. Suvorov, E. Logacheva, A. Mashikhin, A. Remizova, A. Ashukha, A. Silvestrov, N. Kong, H. Goka, K. Park, and V. Lempitsky, "Resolution-robust large mask inpainting with fourier convolutions," in *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pp. 2149–2159, 2022.

[5] R. B. Girshick, "Fast R-CNN," *CoRR*, vol. abs/1504.08083, 2015.

[6] S. Ren, K. He, R. B. Girshick, and J. Sun, "Faster R-CNN: towards real-time object detection with region proposal networks," *CoRR*, vol. abs/1506.01497, 2015.

[7] R. Girshick, I. Radosavovic, G. Gkioxari, P. Dollár, and K. He, "Detectron." `https://github.com/facebookresearch/detectron`, 2018.

[8] Z. Ge, S. Liu, *et al.*, "YOLOX: exceeding YOLO series in 2021," *CoRR*, vol. abs/2107.08430, 2021.

[9] Y. Yuan, X. Chen, and J. Wang, "Object-contextual representations for semantic segmentation," *CoRR*, vol. abs/1909.11065, 2019.

[10] L. Chen, G. Papandreou, F. Schroff, and H. Adam, "Rethinking atrous convolution for semantic image segmentation," *CoRR*, vol. abs/1706.05587, 2017.

[11] Q. Wang, L. Zhang, L. Bertinetto, W. Hu, and P. H. S. Torr, "Fast online object tracking and segmentation: A unifying approach," *CoRR*, vol. abs/1812.05050, 2018.

[12] G. Liu, F. A. Reda, K. J. Shih, T. Wang, A. Tao, and B. Catanzaro, "Image inpainting for irregular holes using partial convolutions," *CoRR*, vol. abs/1804.07723, 2018.

[13] J. Yu, Z. Lin, J. Yang, X. Shen, X. Lu, and T. S. Huang, "Free-form image inpainting with gated convolution," *CoRR*, vol. abs/1806.03589, 2018.

[14] Zeng *et al.*, "Learning joint spatial-temporal transformations for video inpainting," in *The Proceedings of the European Conference on Computer Vision (ECCV)*, 2020.

[15] Suvorov *et al.*, "Resolution-robust large mask inpainting with fourier convolutions," *arXiv preprint arXiv:2109.07161*, 2021.

[16] J. L. Pech-Pacheco, G. Cristóbal, J. Chamorro-Martinez, and J. Fernández-Valdivia, "Diatom autofocusing in brightfield microscopy: a comparative study," in *Proceedings 15th International Conference on Pattern Recognition. ICPR-2000*, vol. 3, pp. 314–317, IEEE, 2000.

[17] A. Mittal, A. K. Moorthy, and A. C. Bovik, "No-reference image quality assessment in the spatial domain," *IEEE Transactions on image processing*, vol. 21, no. 12, pp. 4695–4708, 2012.

[18] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli, "Image quality assessment: from error visibility to structural similarity," *IEEE transactions on image processing*, vol. 13, no. 4, pp. 600–612, 2004.

Author Biography

Minseong Son received his BS in electrical and electronic engineering from Yonsei University(2018). He is currently working at CJ OliveNetworks as an AI engineer and his research area is computer vision.

Hansol Lee received her BS in electronic and IT media engineering from Seoul National University of Science and Technology (Seoultech, 2019) and her MS in meida IT engineering at the same university (2021). She is currently working at CJ OliveNetworks as an AI engineer and her research area is computer vision and media analysis.

Sung Keun Kwak received his BS in Mathematics and Computer Science from Duke University (2018). He is currently working at CJ OliveNetworks as an AI engineer and his research area is computer vision.

Jihwan Woo is currently working at CJ OliveNetworks as a director of AI Research. He is pursuing his research on providing AI services for art and media.