Unsupervised Video Frame Interpolation using Online Refinement

Seungmin Lee[1], Seongwook Yoon[2] and Sanghoon Sull[3]

[1, 2, 3] School of Electrical Engineering, Korea University
Seongbuk-Gu, Seoul, 02841, Republic of Korea
E-mail : [1]smlee@mpeg.korea.ac.kr, [2]swyoon@mpeg.korea.ac.kr, [3]sull@korea.ac.kr

Abstract: Video frame interpolation is a method that estimates an intermediate frame between two consecutive frames and makes the video playable at a high frame rate or slow motion. Since video frame interpolation using deep neural networks should produce good results for unseen arbitrary videos, we apply an online refinement, which refines a network's weights by learning new instances in online manner. Since effective online refinement should select useful subset of instances, we introduce several rules using simple metric to select frames from the entire video. The metric is cycle consistency error of all instances composed of three frames as the metrics. Then the network is trained by the selected instance using its original unsupervised learning method. We investigated the performances of several methods at UCF 101 dataset.

Keywords: unsupervised learning, video frame interpolation, online refinement

1. Introduction

Recently, the usage of a display playing a high frame rate video has increased, and the frame rate of the video has also increased. Similarly, various high-frame rate cameras have also appeared, but in reality, shooting all scenes with a high-frame rate camera is impossible due to limitations of both memory and time. Instead, video frame interpolation, which can increase the frame rate of videos regardless of such limitation, is being studied.

Generally speaking, the video frame interpolation task is to generate several frames between given two frames as shown in Figure 1. In order to create intermediate frame, motion between the two frames must be estimated. Then, the middle frame is generated based on the motion. Therefore, the accuracy of optical flow greatly affects the performance of video frame interpolation.

The video frame interpolation should work properly for any arbitrary video while the learning-based optical flow computation is often fine-tuned for a specific dataset. The video frame interpolation should be able to produce good results for general video smoothing or slow motion.

Thus, we need to think about online refinement for video frame interpolation. Rather than using only few recent frames, we propose a method considering the tendency of given video which allows to both reduce the calculation time and enhance the performance of online refinement.

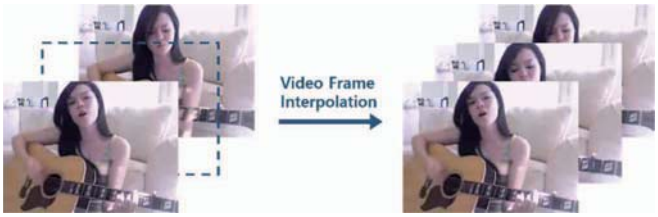

Figure 1. Video frame interpolation.

There are two main concepts in our method: Metric and rules. Firstly, we need a metric to decide whether the frame should be used to refinement or not, though it is not sufficient to predict the true refinement results. In particular, we observe the interpolation error(IE), in the training process of [2], which is the difference between existing intermediate frame and generated intermediate frame. Secondly, we introduce several rules based on the metric and randomness to compensate the insufficiency of the metric. Note that our formulation may include more general online refinement method of instance-based unsupervised learning than the specific method for video frame interpolation task. In addition, the overfitting problem may be raised in online refinement. However, overfitting problem in such a good direction may enable fast and even stable training, while the generalization error increases. Instead, we should return to the pre-trained generalized weight for every new video.

In this paper, we investigate how to design the online refinement for video frame interpolation with an easier concept to extend to general online refinement problem. To do this, we monitor the metric induced from corresponding unsupervised learning to predict the true refinement result. Also, we designed several random rules to select most important frames among entire video for efficient online refinement.

2. Related Works

In a recent study [5, 8], video frame interpolation is divided into two phases. First, the bi-directional flow between two consecutive frames is estimated. Then, the intermediate frame is improved from an initial intermediate frame warped using the flow and occlusion reasoning. *Reda at al.*[8] presents an unsupervised learning method using cycle-consistency loss among three consecutive frames. Since the method is instance-based, we use it for online refinement.

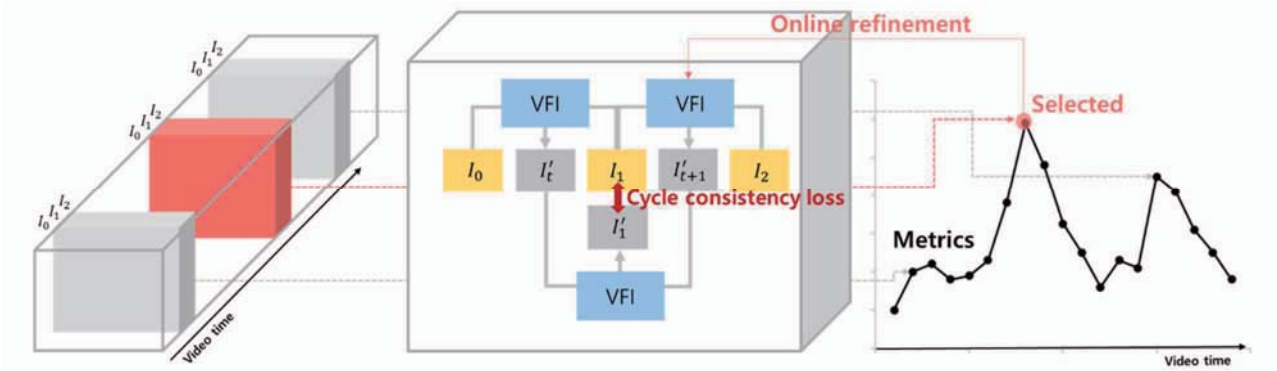

Figure 2. We observed cycle consistency error of all instances composed of three frames as the metrics. Then, an instance is selected by rules using the metrics of the entire video frame. Finally, the network is trained by the selected instance using its original unsupervised learning method. Also, such process is repeated until they meet a terminate condition.

The online refinement is a strategy of fine-tuning data at test time. Recently, it was used in video object segmentation tasks in research [4]. The key frame list is generated and learned by observing the motion boundary in the current frame and the previous frame and excluding the frame of the unclear motion boundary. This method works well for segmentation, but it uses task-specific metric which is not directly induced from the original learning method. In another case [7], online refinement is used for ego-motion estimation. Since it should be real-time, only the early part of video was used. However, the online refinement for video frame interpolation can view the entire video, because real time is not a requirement for slow motion. To our knowledge, there is not online refinement method for video frame interpolation.

3. Proposed Method

In this section, we describe about our proposal, but we mainly introduce how to design the online refinement algorithm efficient in general. First, we describe the concept of metric which is monitored to select instances(frames) from given dataset(video). Second, we describe the concept of rule to perform actual online refinement using the metric and randomness, also in general. Briefly speaking, our proposal repeats single instance online refinement. Assuming that every stage of online refinement is equivalent, we can apply same rule recursively.

3.1 Metric

As mentioned above, we choose the metric for online refinement induced directly from the unsupervised learning of video frame interpolation in [8]. In detail, the unsupervised learning takes place in three consecutive frames. First, an intermediate frame I'_t is generated in frames I_0 and I_1, and simultaneously an intermediate frame I'_{t+1} between I_1 and I_2 is generated. Second, intermediate frame I'_1 is created again using the generated frame I'_t and I'_{t+1}. At this time, the generated I_1 and I'_1 should be equal, the training can be performed unsupervisedly through such cycle consistency loss from the three frames. The process is shown in Figure 2.

Although the actual learning procedure includes other losses such as smoothing flows and perceptual loss, we choose the cycle consistency loss for the metric to simplify and generalize the problem. However, it needs to be verified that the cycle consistency loss is related to the true interpolation loss, which is slightly different from the case that the photometric loss is nearly equal to the true performance in optical flow estimation task.

In order to confirm the relationship between the cycle consistency loss and the true interpolation error, we observed the metric values before online refinement and the interpolation loss for the entire video. Note that this is just for experiment and the actual online refinement process only observes not the true interpolation error but the metric values before and after the online refinement.

As shown in Figure 3, it can be seen that the true interpolation error of the entire video is highly related with the cycle consistency loss. However, while the most of instances produces good decrement in error, some of instances do not or even increase the error largely. Showing at (c) in Figure 3, the two graphs are closely related in most areas, but not in a specific area (gray highlight). Also, as shown in (d) in Figure 3, the result is completely opposite in certain areas. While most of instances in our dataset shows the consistent relationships, we need to handle such wrong instances.

Thus, it is important to prevent the wrong instance from spoiling the online refinement. First, we can try to design more stable metric. Otherwise, we can make additional rules to choose the instance more diversely considering other factors than the metric only. We propose the latter approach, because it provides more general solution for online refinement.

3.2 Selection rule

As mentioned above, we repeat online refinement for several stages. At each stage, metrics of entire video should be

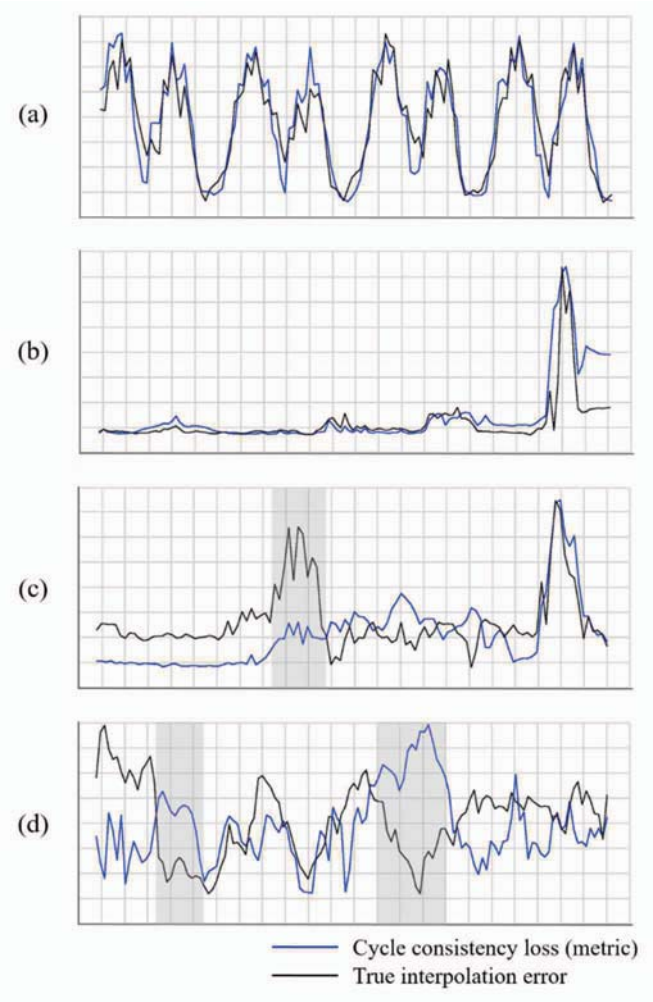

Figure 3. Comparison of cycle consistency loss and true interpolation error.

monitored. Then, we choose an instance using random rule among the rules described as below.

Our first rule is to choose the largest metric value for the entire frame of video. Simply, assuming the original ideal case that all of the metrics are enough to predict the true error correctly, learning the instance with the largest metric value will give good refinement results. By learning the most difficult instance, the network may be able to update the weight in the steepest direction that fits entire video. However, choosing the largest metric value does not always give the best refinement results, as shown in Figure 3. Also, even though the instance of the largest metric give the best refinement result, we need to change the rule if the instance index of the largest metric does not change so that only one instance is used for online refinement. It may cause unintended overfitting even in short video.

The second rule is to observe how much the metric has changed and select an instance with smallest change. This rule can prevent the online refinement from learning only partial features of entire video. If difference of an instance metric values is very small or negative, it means that the refinement in the previous stage had no effect or had a bad effect to those kinds of instances in the video. However, these small changes are valid for sufficiently difficult instance. Since there are lots of easy instances in entire video of which metrics do not change through the refinement, we need to exclude them and choose the instance of which metric is large but unchanged.

Third, in order to escape from a stuck case, we applied randomness. Like the second rule, after thresholding instances whose metric value is higher than a certain level, a frame is randomly selected among them. Finally, we apply above three rules stochastically.

Additionally, we can directly measure the difference between instances to choose another instance sufficiently different from previous ones., However, it requires much more computation $O(n^2)$ than $O(n)$ of metric monitoring.

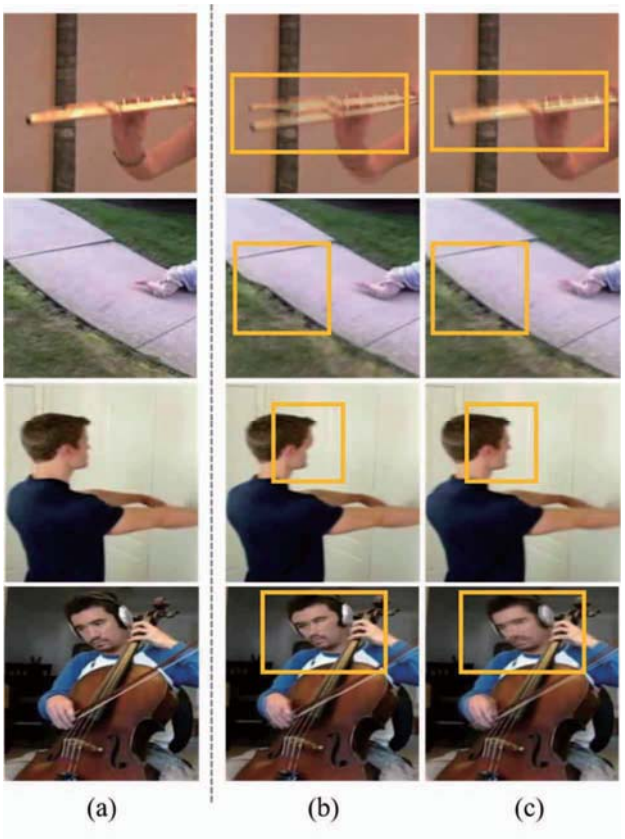

Figure 4. Result of sample from UCF101. (a) ground truth. (b) before online refinement. (c) after online refinement.

4. Experiment

We used the adobe 240-fps dataset[3] to train the network. UCF101 was used for the quantitative evaluation, and additionally tested on several videos collected from YouTube was used to check the image qualities for actual videos. In UCF101, as a single video was randomly selected from all

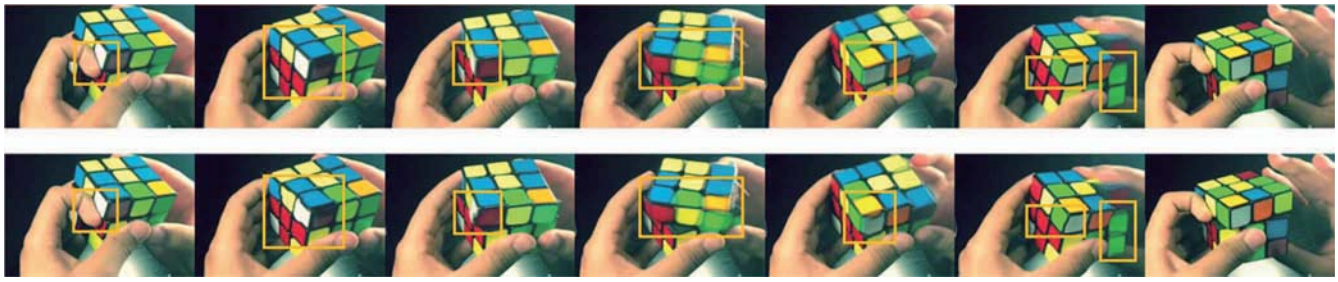

Figure 5. Before(below) and after(above) online refinement in video.

classes, and a total of 101 videos were used for evaluation, and converted 25fps to 50fps. We proceed to online refinement in a total of 20 stages. And, as mentioned above, overfitting in a good direction makes training stable and fast, so the learning rate in refinement is adjusted to be larger than the training time. Also, we used the network structure that connects two U-Net[1], used in *Jiang at al*[5].

We added termination conditions for when the online refinement ends. When the metric values of the entire frame in the recent refinement stages do not change over a certain level, refinement is terminated. For our methods using the termination condition, about 10 steps are performed on average. Since the complete random method cannot check the metric value, we terminated the method at 10 stage equal to our methods on average.

Table 1. Online refinement results on UCF101 dataset.

Method	IE
Complete random	1.2441
Metric only	1.2460
Metric + rule	**1.3319**

Table 1 shows the quantitative results. The first result, complete random, is of the method selecting frames at completely random without any rules, the second result is of the method selecting frames at random after thresholding by their metric values, and the last result is of the method applying all our rules. You can see that our method shows better results than before applying online refinement. Also, we can see that our method with all the rules shows slightly better results than the method of randomly selecting instances after the threshold. If the reason for the inconsistency between the true interpolation loss and the cycle consistency loss mentioned in Figure 3 can be found more clearly, we think that the number could be improved.

In Figure 4, the frames before and after our online refinement method are compared. As you can see from the picture, the boundaries of objects are discontinuous before the online refinement, but they look better after the online refinement. Although blurring occurs after the refinement for large motion is applied, it looks much smoother than flickering when observing actual videos.

Figure 5 shows the result for one video. The phenomenon that the border of the cube looks discontinuous or wavy was reduced after the online refinement. In addition, for frames that produce good results before online refinement, a smooth intermediate frame is generated as well after applying online refinement. From this experiment, we can investigate that the online refinement using only small subset of the entire video enhances overall qualities of the video.

5. Conclusion

We applied online refinement to the video frame interpolation task, it should be generalized to unseen general videos. In particular, we selected some frames from the entire video and used them in the online refinement. We observed the metric before online refinement, and applied various rules to select a frame based on the metric.

As mentioned above, we think that this type of rules can be applied to the online refinement of other tasks. For example, it will be able to apply online refinement to unsupervised depth of flow estimation, if it does not require real-time performance.

Acknowledgement

This work was supported by the MSIT(Ministry of Science and ICT), Korea, under the ITRC(Information Technology Research Center) support program(IITP-2020-2016-0-00464) supervised by the IITP(Institute for Information & communications Technology Planning & Evaluation)

References

[1] O. Ronneberger, P. Fisher, and T. Brox. "U-net: Convolutional networks for biomedical image segmentation," CoRR, vol. abs/1505.04597, 2015.

[2] Simon Baker, Daniel Scharstein, JP Lewis, Stefan Roth, Michael J Black, and Richard Szeliski. "A database and evaluation methodology for optical flow," International Journal of Computer Vision, vol. 92, no. 1, pp.1-31, 2011.

[3] Su, Shuochen and Delbracio, Mauricio and Wang, Jue and Sapiro, Guillermo and Heidrich, Wolfgang and Wang, Oliver. "Deep video deblurring for hand-held

cameras," In Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition, pp.1279-1288, 2017.

[4] Li, Gongyang and Liu, Zhi and Zhou, Xiaofei. "Effective online refinement for video object segmentation," Multimedia Tools and Applications, vol. 78, no. 23, pp.33617-33631, 2019.

[5] H. Jiang, D. Sun, V. Jampani, M. Yang, E. G. Learned-Miller, and J. Kautz. "Super slomo: High quality estimation of multiple intermediate frames for video interpolation," CoRR, vol. abs/1712.00080, 2017.

[6] K. Soomro, A. R. Zamir, and M. Shah. "UCF101: A dataset of 101 human actions classes from videos in the wild," CoRR, vol. abs/1212.0402, 2012.

[7] V. Casser, S. Pirk, R. Mahjourian, and A. Angelova. "Unsupervised monocular depth and ego-motion learning with structure and semantics," CoRR, vol. abs/1906.05717, 2019.

[8] F. A. Reda, D. Sun, A. Dundar, M. Shoeybi, G. Liu, K. J.Shih, A. Tao, J. Kautz, and B. Catanzaro. "Unsupervised video interpolation using cycle consistency," CoRR, vol. abs/1906.05928, 2019.