SR-CL-DMC: P-frame coding with Super-Resolution, Color Learning, and Deep Motion Compensation - A Supplemental Document

Man M. Ho
Hosei University
Tokyo, Japan
man.hominh.6m@stu.hosei.ac.jp

Jinjia Zhou
Hosei University and JST, PRESTO
Tokyo, Japan
jinjia.zhou.35@hosei.ac.jp

Gang He
Xi'dian University
Xi'an, China
ghe@xidian.edu.cn

Muchen Li
Hosei University
Tokyo, Japan
muchen.li.42@hosei.ac.jp

Lei Li
Xi'dian University
Xi'an, China
1849747827@foxmail.com

Loss Function.

We adopt [1] and use the restoration loss for low-resolution as:

$$\mathcal{L}_{restore} = \frac{1}{n}\sum_{i=1}^{n} ||LR_2 - \hat{LR}_{2\_1}||_2^2 \tag{1}$$

and reconstruction loss for high-resolution as:

$$\mathcal{L}_{recon} = \frac{1}{n}\sum_{i=1}^{n} ||HR_2 - \hat{HR}_2||_2^2 \tag{2}$$

In predicting optical flow and warping to generate the target frame, the warped frames are observed as:

$$\begin{aligned}\mathcal{L}_{warp} = \frac{1}{n}\sum_{i=1}^{n} ||LR_2 - \phi(LR_1, OF^{LR}_{1\to2})||_2^2 \\ + \frac{1}{n}\sum_{i=1}^{n} ||HR_2 - \phi(HR_1, OF^{HR}_{1\to2})||_2^2\end{aligned} \tag{3}$$

with ϕ is a flow backward-warping function, $OF^{LR}_{1\to2}, OF^{HR}_{1\to2}$ are the estimated optical flows for low-resolution and high-resolution respectively. The refinement after warping frame is observed under loss function:

$$\begin{aligned}\mathcal{L}_{refine} = \frac{1}{n}\sum_{i=1}^{n} ||LR_2 - \hat{LR}_{2\_2}||_2^2 \\ + \frac{1}{n}\sum_{i=1}^{n} ||HR_2 - \hat{HR}_{2\_1}||_2^2\end{aligned} \tag{4}$$

To enhance the Multi-Scale Structural Similarity (MS-SSIM), we add the MS-SSIM loss for the final output as:

$$\mathcal{L}_{ms\text{-}ssim} = MS\text{-}SSIM(HR_2, \hat{HR}_2) \tag{5}$$

Finally, our total loss is defined as:

$$\begin{aligned}\mathcal{L}_{total} = \alpha * \mathcal{L}_{restore} + \beta * \mathcal{L}_{recon} + \gamma * \mathcal{L}_{warp} \\ + \delta * \mathcal{L}_{refine} + \eta * \mathcal{L}_{ms\text{-}ssim}\end{aligned} \tag{6}$$

where $\alpha, \beta, \gamma, \delta, \eta$ are hyperparameters empirically set as $0.5, 0.5, 0.6, 0.8, 1$.

Estimating scene changes and large motion using perceptual distance for the Youtube UGC dataset. As shown in 1, most of possible pairs is closer to a static pairs. Higher perceptual distance, the less useful information that we can leverage from the reference. We observe that there are levels that the perceptual distance can represent from low to high:

1. Static pairs.
2. Moving object with still background.
3. Moving whole frame, but the contents are the same.
4. At this distance, it's mixed between: a) Losing objects but still have the same color tone. b) The whole pixels are changed with same content.
5. Scene changes or very large motion.

as shown in Figure 2.

Additional results in high resolution. are shown in Figure 3, 4, 5. The MS-SSIM of Y,U,V is shown bellow each result.

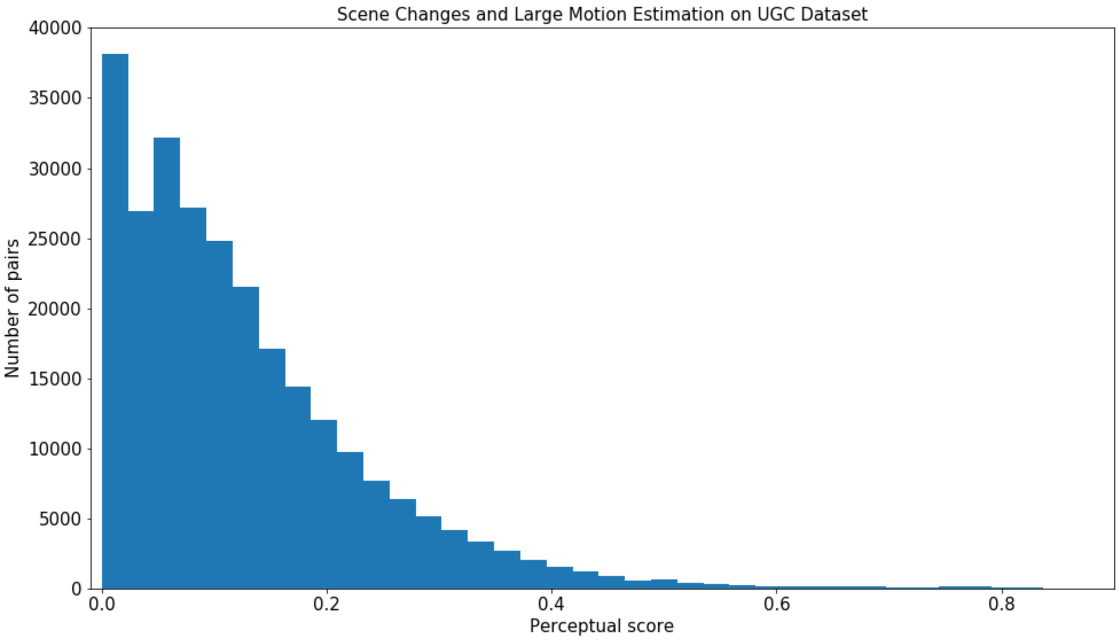


Figure 1. Using the perceptual metric [2] to estimate the number of scene changes or large motion scenes. x-axis represents the perceptual distance, y-axis is for number of possible pairs in YouTube UGC dataset. Lower perceptual score means closer to a static pair.

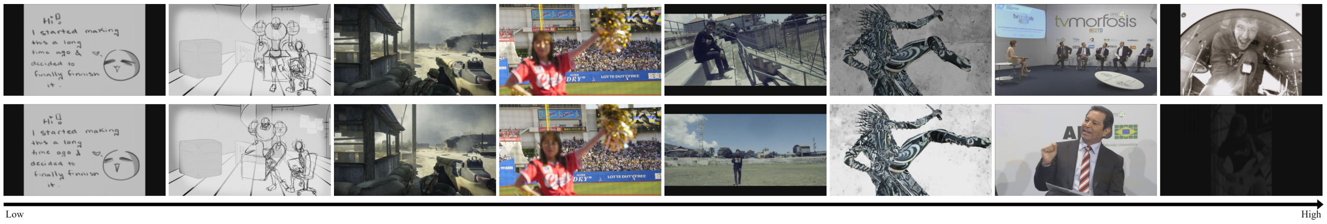
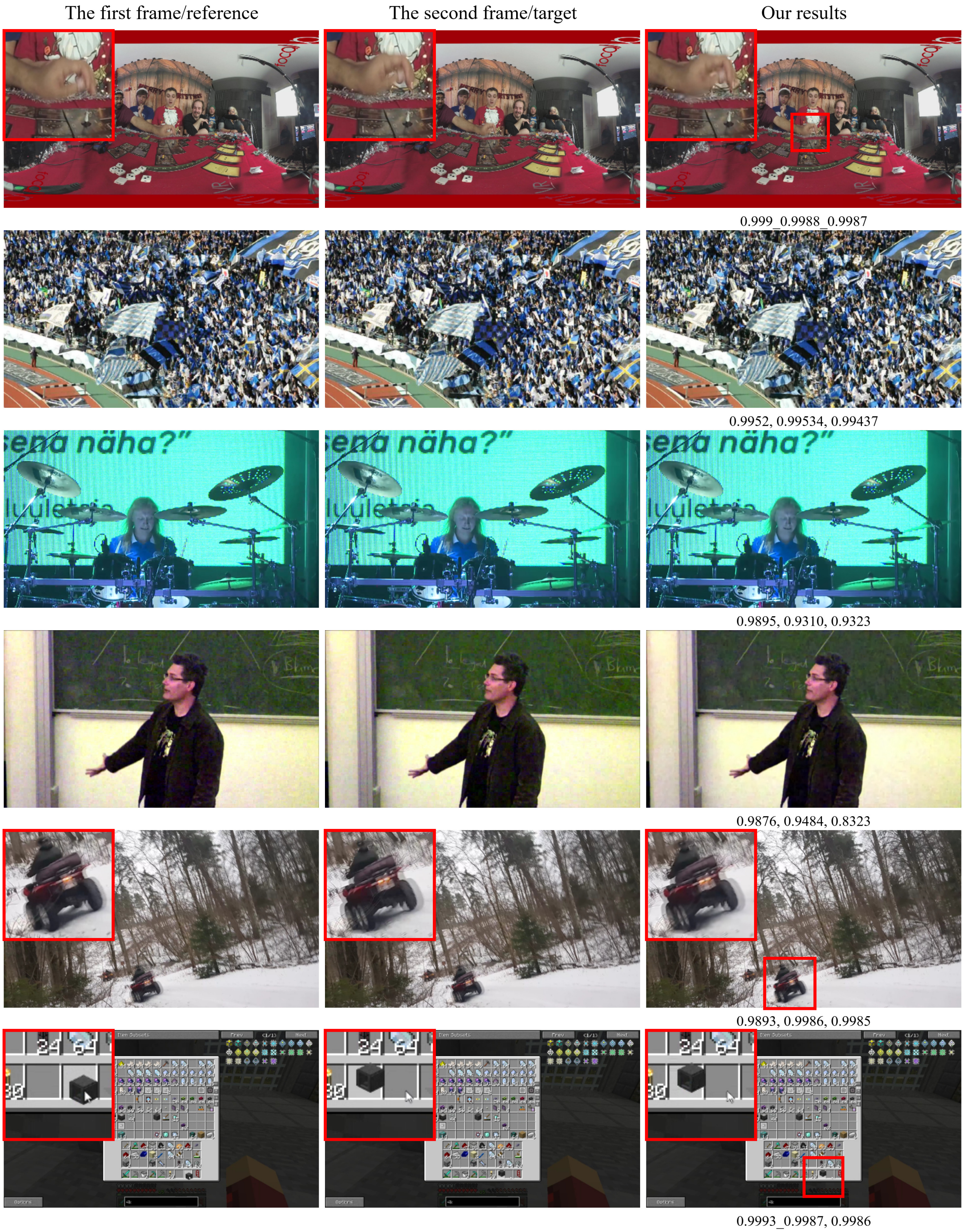


Figure 2. Illustration of the perceptual metric [2] on detecting scene changes or large motion. The perceptual distance is increased from *left to right*. *Top-bottom*: the reference as a previous frame and the current frame.

References

[1] Man M. Ho, Jinjia Zhou, and Gang He. Rr-dncnn v2.0: Enhanced restoration-reconstruction deep neural network for down-sampling based video coding. *arXiv preprint arXiv:2002.10739*, 2020. 1

[2] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 586–595, 2018. 2



The first frame/reference

The second frame/target

Our results

0.999_0.9988_0.9987

0.9952, 0.99534, 0.99437

0.9895, 0.9310, 0.9323

0.9876, 0.9484, 0.8323

0.9893, 0.9986, 0.9985

0.9993_0.9987, 0.9986

Figure 3. Additional Results 1.

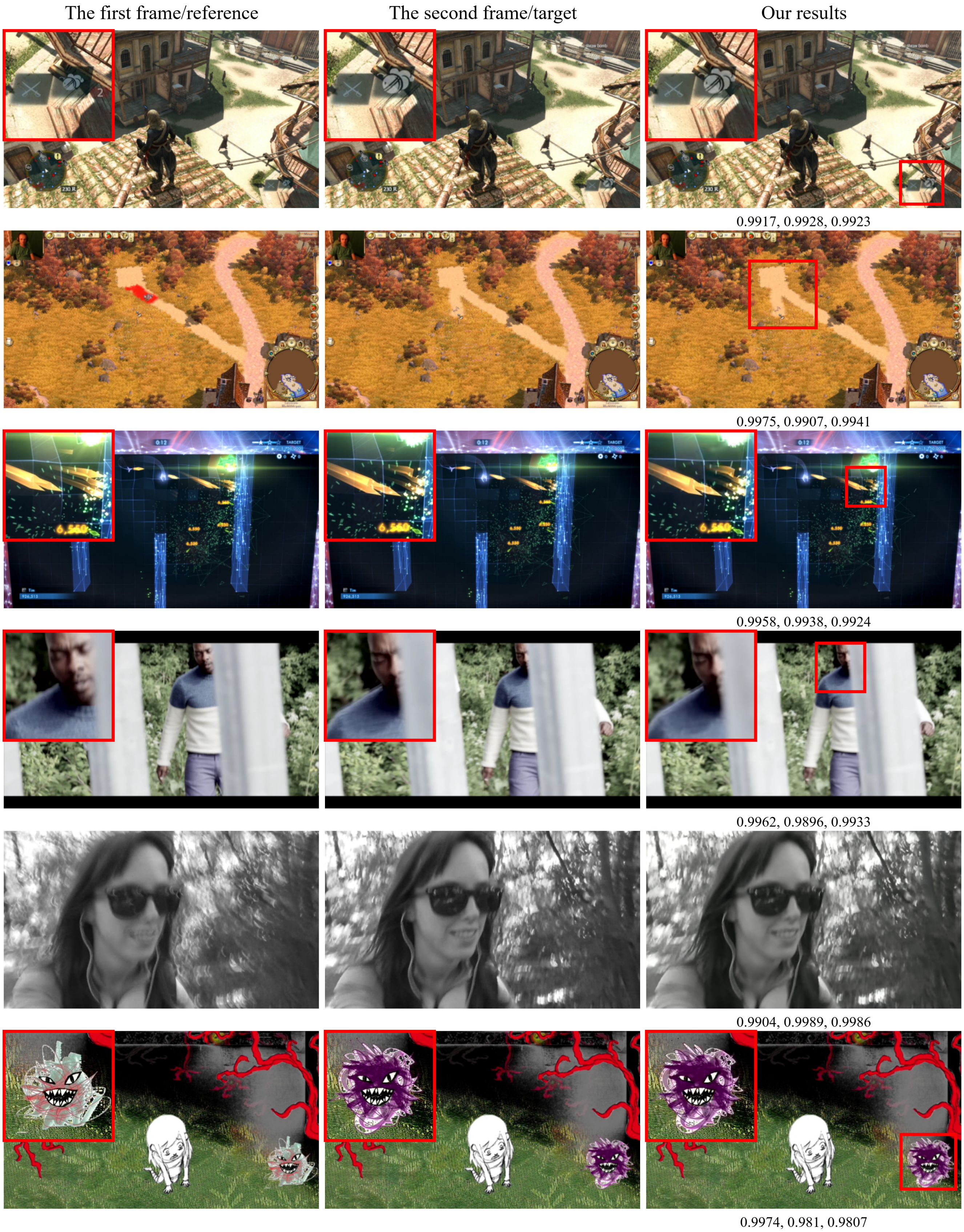
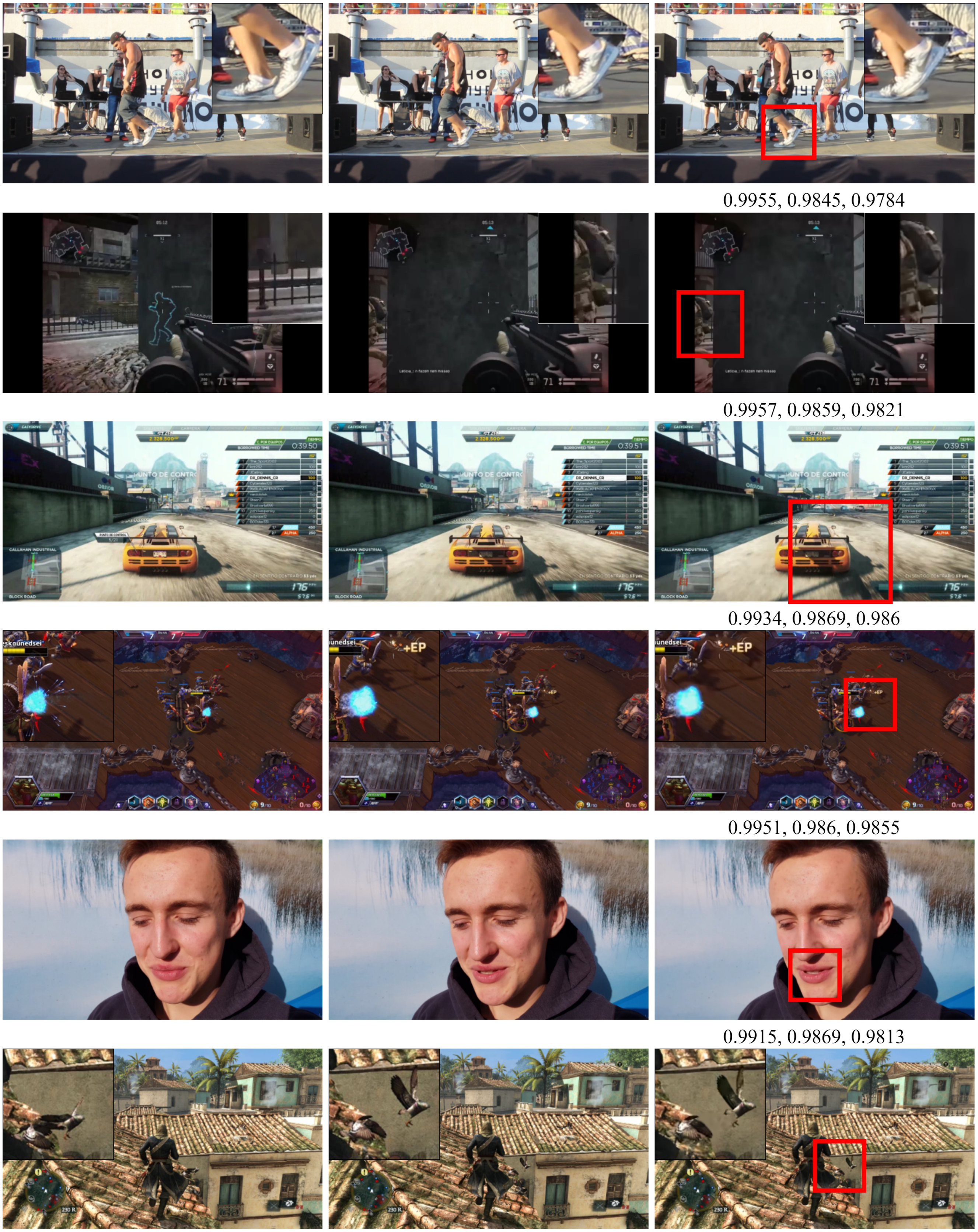


Figure 4. Additional Results 2.



0.9955, 0.9845, 0.9784

0.9957, 0.9859, 0.9821

0.9934, 0.9869, 0.986

0.9951, 0.986, 0.9855

0.9915, 0.9869, 0.9813

0.9917, 0.9896, 0.9869

Figure 5. Additional Results 3.