# RA Report for FIFA project

Hsuan-I Ho and Jie Song

ETH Zürich

**Abstract.** This report summarize the working progress and results in the 3-month part-time RA job. The main objective for this project is synthesizing photo-realistic high-resolution videos for the FIFA players tracking application. However most existing video super-resolution (VSR) models and datasets are focusing on general video scenes and textures which are not sufficient for our task. It is therefore required to propose model adaptation and collect suitable training data for super-resoluting players moving in fast motion. Inspired by the recent VSR pipeline, we further extend their model in to a human-centric VSR scheme alongside several adapted network designs. In our experiments, we showed our proposed training pipeline could produce more realistic results than other video super-resolution baselines on the collected Football4K dataset.

**Keywords:** Video Super-resolution, Generative Model, Self-supervised Learning

## 1 Task Description

With the final goal of delivering a player pose tracking and rendering system (see Fig. 1) on a football stadium, a clear capture of human body in high resolution is required as the input for the system. However cameras, even used for broadcasting, have many physical limitations on capturing human bodies, such as limited native resolution of sensors, insufficient optical zooming range of lens and long distances as mounted on the top the stadium. As a result, most videos have resolution of 2K $\sim$ 4K, where the human bounding boxes detected by the system are around only 400 $\sim$ 2000 pixels on average. This would largely affect the accuracy in the later pose estimation and rendering.

To solve the above mentioned problem, super-resolution (SR) [2] techniques is a possible pre-processing solution to this pipeline. As indicated by the recent study [6], applying a pre-trained SR model would benefits the bounding boxes that are too small for pose estimation. However, we also point out two possible drawbacks by directly using the off-the-shelf SR models in the system. First, most of the existing methods are trained and evaluated on the general image benchmarks, which might be not favorable to the specific human body that has more complicated body articulation. Moreover, these approaches process each frame independently while fail to utilize continuous video information and consider its temporal consistency.

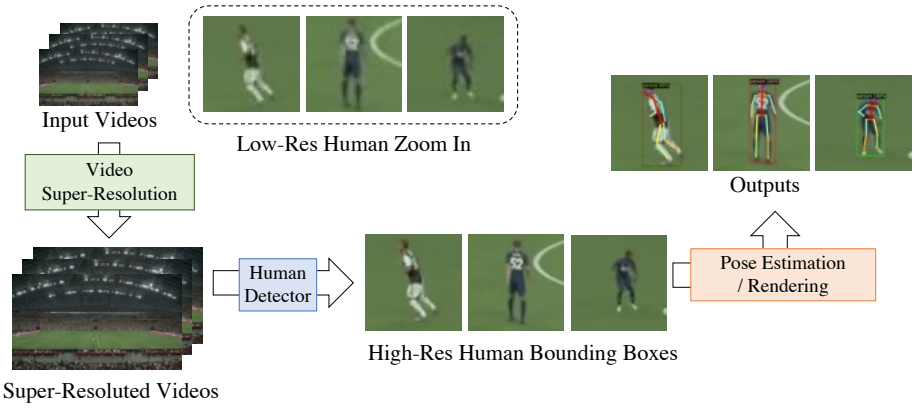


**Fig. 1. System overview.** Our system consists of a video super-resolution model for prepossessing, a learned human detector and a pose estimation model for the final video rendering. Details are explained in Section 1.

Thus, in this project we further model our problem in a human-centric video super-resolution (VSR) scheme. That is to say, we not only need to generate reasonable human body structures from the low-resolution blurry inputs, but also need to consider the overall video consistency and exploit more temporal information when synthesizing high-resolution videos. To this end, we collect a new 4K video dataset from the football broadcasting data which enables the training and evaluation on high quality human-centric video clips. Besides, we also propose several adaptations on the existing VSR framework to enhance our model producing more realistic and smooth high-resolution football videos. Please refer to the following chapters for our implementation details and experimental results.

## 2 Proposed Framework

### 2.1 Architecture survey

We first review the recent neural network architectures used for super-resolution. SRCNN [5] first attempted to model the image super-resolution problem with convolutional neural network (CNN). Later VDSR [8], DRCN [7], RDN [20] have introduced using residual neural network architecture for training much deeper network architectures. They have showed that skip connection and recursive convolution speed up the learning in the SR model. However, these methods all minimized the L1 or L2 loss function during training, which would produced averaged blurry images. SRGAN [9] first utilized the generative adversarial network (GAN) to produce photo-realistic and sharper super-resolution images. To solve the problem of fixed up-sampling scale during training, MDSR [10] used a technique of multi-scale aggregation and recent LIIF [3] learned a continuous image representation for unlimited upscale.

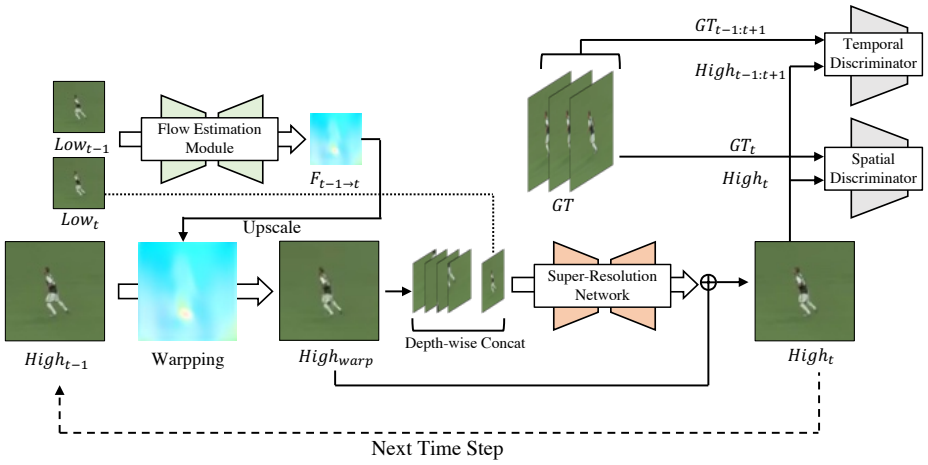


**Fig. 2. Video super-resolution model overview.** Our model consists of a optical flow estimation module plus a super-resolution network to predict video frames iteratively. We also adopt adversarial training to ensure the sharpness and smoothness of the generated videos.

Besides the single image super-resolution model, video super-resolution (VSR) aims at using more neighboring frames to improve the temporal alignment and consistency. For instance, TDAN [16] uses deformable convolutions to align the video frames at the temporal dimension. EDVR [18] further aggregate multi-scale, multi-frame motion information with a pyramid cascading architecture and a spatial temporal attentional alignment network. Instead of using an iterative sliding window, BasicVSR [18] adopts a bidirectional recurrent network coupled with a simple optical flow-based feature alignment to align the nearby video frames. More recent VSR-Transformer [1] further replaces the recurrent network by a transformer backbone with spatial-temporal self-attention module. TecoGAN [4] first integrate the VSR architecture with spatial-temporal GAN to synthesize perceptually realistic videos in a self-supervised learning fashion.

### 2.2 Network architecture

We adopt the TecoGAN [4] network as the backbone of our framework since its temporal smoothness and sharper results learned by GAN could better benefit human detection and pose estimation. The overview of the pipeline is depicted in Fig 3, including a flow estimation module, SR network as well as spatial-temporal discriminators.

Our pipeline is working in an autoregressive fashion (i.e., a sliding window of two frames) to produce the high-resolution video output. The flow estimation module first estimate the optical flow between two consecutive low-resolution input frames $(Low_{t-1}, Low_t)$. The predicted flow $F_{t-1\to t}$ is then upscaled to our target resolution by simple linear interpolation. A high-resolution image

prior $High_{warp}$ can be obtained by warping the previous generated frame with the estimated flow. The SR network then take the prior $High_{warp}$ and the low-resolution image $Low_t$ as inputs to refine the final high-resolution frame at time $t$. Similar to the TecoGAN network, we also employ generative adversarial learning in the both spatial and temporal dimensions to ensure sharpness and smoothness of the generated high-resolution videos. Detailed layers can be found in Table 1.

### 2.3 Loss functions

The learning of our framework is done by optimizing several loss functions end-to-end in a special reflective sampling strategy. For each training sample containing T continuous frames, we first duplicated the training batch in a reverse order, resulting the ground-truth image frames of $\{I_0, I_1, \ldots, I_{T-1}, I_T, I'_{T-1}, \ldots, I'_1, I'_0\}$. We train our flow estimation module by calculating the L1 error between warping frames and ground truth frame at the low resolution:

$$\mathcal{L}_{warp} = \sum_{t=0:T} \|Low_t - W(Low_{t-1}, F_{t-1\to t})\|_1, \tag{1}$$

where W(.) is the image warping operation using the predicted flow. The reconstruction loss of our SR network is computed as:

$$\mathcal{L}_{reco} = \sum_{t=0:T} (\|High_t - I_t\|_1 + \lambda_{percep}\|\psi(High_t) - \psi(I_t)\|_1). \tag{2}$$

where $\psi_l$ is a trained deep feature extractor from the VGG-19 network [15] pre-trained on ImageNet and $\lambda_{percep}$ is a scalar weighting the two terms. To further ensure the temporal consistency, we follow the original TecoGAN computing the L1 PingPong Loss:

$$\mathcal{L}_{pp} = \sum_{t=0:T-1} \|High_t - High'_t\|_1. \tag{3}$$

The LSGAN [11] loss is applied to the SR network and the discriminator during training. The adversarial loss applied to the discriminator is formulated as:

$$\mathcal{L}^D_{adv} = \frac{1}{2}\mathbb{E}_{(High_t)}\left[D(High_t)^2\right] + \frac{1}{2}\mathbb{E}_{(I_t)}\left[(D(I_t)-1)^2\right]. \tag{4}$$

The adversarial loss for generator is calculated as:

$$\mathcal{L}^G_{adv} = \mathbb{E}_{(High_t)}\left[(D(High_t)-1)^2\right] + \lambda_{FM}\mathcal{L}_{FM}, \tag{5}$$

where the discriminator's feature-matching loss $\mathcal{L}_{FM}$, comparing the real and generated image using the activations of the discriminator, is calculated as:

$$\mathcal{L}_{FM} = \mathbb{E}_{(\boldsymbol{p}_t, I_t, High_t)} \sum_{j=1}^{M} \frac{1}{N_j}\|D^{(j)}(High_t) - D^{(j)}(I_t)\|_1. \tag{6}$$

| | layer type(s) | out channels | stride | activation |
|---|---|---|---|---|
| Flow Estimation Module | | | | |
| 1 | 3×3 Conv × 2 | 32 | 1 | LReLU |
| | 2×2 Max. Pooling | | 2 | |
| 2 | 3×3 Conv × 2 | 64 | 1 | LReLU |
| | 2×2 Max. Pooling | | 2 | |
| 3 | 3×3 Conv × 2 | 128 | 1 | LReLU |
| | 2×2 Max. Pooling | | 2 | |
| 4 | 3×3 Conv × 2 | 256 | 1 | LReLU |
| | UpSampling | | 2 | |
| 5 | 3×3 Conv × 2 | 128 | 1 | LReLU |
| | UpSampling | | 2 | |
| 6 | 3×3 Conv × 2 | 64 | 1 | LReLU |
| | UpSampling | | 2 | |
| 7 | 3×3 Conv | 32 | 1 | LReLU |
| 8 | 3×3 Conv | 2 | 1 | Tanh |
| Super-Resolution Network | | | | |
| 1 | 3×3 Conv | 64 | 1 | ReLU |
| 2 | Residual Blk × 10 | 64 | 1 | ReLU |
| 3 | 3×3 ConvT | 64 | 2 | ReLU |
| 4 | 3×3 ConvT | 64 | 2 | ReLU |
| 5 | 3×3 ConvT | 3 | 1 | - |
| Residual Block | | | | |
| | Residual In | | | |
| 1 | 3×3 Conv | 64 | 1 | ReLU |
| 2 | 3×3 Conv | 64 | 1 | - |
| | Residual Out | | | |
| Discriminator | | | | |
| 1 | 3×3 Conv, | 64 | 1 | LReLU |
| 2 | 4×4 Conv, BN | 64 | 2 | LReLU |
| 3 | 4×4 Conv, BN | 64 | 2 | LReLU |
| 4 | 4×4 Conv, BN | 128 | 2 | LReLU |
| 5 | 4×4 Conv, BN | 256 | 2 | LReLU |
| 6 | Linear | 1 | 1 | - |

**Table 1. Network layer details of our model.** 'LReLU' denotes leaky ReLU activation, and 'BN' denotes batch normalization layer, and 'ConvT' denotes transposed convolution layer.

With $M$ being the number of discriminator layers, $N_j$ the number of elements in the j-th layer. The expectation is computed per mini-batch, over the input $\boldsymbol{p}_t$, $I_t$, and $\hat{I}_t$.

The overall loss function is the summation of the above terms:

$$\mathcal{L}_{total} = \mathcal{L}_{adv} + \lambda_w \mathcal{L}_{warp} + \lambda_r \mathcal{L}_{reco} + \lambda_p \mathcal{L}_{pp}, \tag{7}$$

where $\lambda$s are weights that control the interaction of the loss.

### 2.4 Implementation details

Since our pipeline focus on super-resoluting human bodies, we first runs a human detector [19] to localize the center of each human on every frames. We then use the center to perform a human-centered cropping into a patch of $128 \times 128$ pixels, $T = 5$ for each training sample. We then downsample and apply Gaussian blur to the ground-truth image, resulting a $32 \times 32$ low-resolution human body training patch. Random rotation and shifting are applied to the training images for data augmentation.

We refactor the original TencoGAN TensorFlow implementation to Pytorch, which allows us to implement more customized adaptions in our framework. We train our network using Adam optimizer with a learning rate of $10^{-4}$ for the neural rendering network, $4 \times 10^{-4}$ for the discriminator, and first- and second-momentum of 0 and 0.99. For each training iteration, we sample $T = 3$ continuous frames as a training sample and recovery the middle frames in each feed-forward pass. The batch size is set to 4, $T$ is increased by 1 and the learning rate is decayed with a scale of 0.5 for every 10 epochs. We set the weights of the loss terms to be $\lambda_w = 100$, $\lambda_{percep} = 0.2$, $\lambda_r = 100$, $\lambda_{pp} = 50$, $\lambda_{FM} = 1.0$ across all experiments. The training takes one day on a RTX 2080Ti for 60 epochs.

## 3 Dataset

To train our proposed framework and perform better human-centric super-resolution, we collect a new 4K-FHD ($3840 \times 2160$) dataset **Football4K** from the publicly available online football videos in total 1.5 hours. We manually split all the video into 150 continuous clips for ensuring temporal smoothness during training. We also provide human bounding box information in the 4K video frames so that it is possible to preform human-centric cropping during training.

For test, we use another separated dataset **FIFA2K** in a lower resolution, which is collected from the FIFA offical broadcasting data. The dataset captures a complete football games from different camera position in a $1920 \times 1080$ resolution. We manually split the test videos into several clips based on three different human size setting:

- **Large** contains video scenes which most human bounding boxes are larger than 2000 pixels.
- **Medium** contains video clips with human bounding boxes within 1000 to 2000 pixels.

**Fig. 3. Our collected Football4K dataset.** Our dataset contains various football scenes in the 4K resolution. We also the provide human positions of each frame.

– **Smal** includes challenging video scenes of bounding boxes that are lower than 1000 pixels.

For further ablative studies, we split a single 8K (7680 × 4320) video into train/validation sets of different resolution (1K∼8K). We would like to analyze the optimal input range of our proposed framework.

## 4 Experiment

### 4.1 Super-resolution results

To save the GPU memory usage during inference, we downscale the video clips in **FIFA2K** into 1K (960 × 540). From Fig. 4 to FIg. 8 we visualize the super-resoluted human-centered cropping bounding boxes produced by our model from 1K to 4K. In the **Large** and **Medium** split sets, each bounding box contains 64 × 64 pixels. While in the **Small** split test, each bounding box contains only 32 × 32 pixels which is more challenging to recover human bodies at the high resolution.

The results show that our model is able to generate sharp and complete human bodies even in the small bounding boxes and the low resolution condition. We noticed that the best improvement of visual quality appears in the **Medium** split, where the outline of human body and the color intensity are more favorable. In the challenging cases of **Small** set, the outline became more blurry due to limited input information and ambiguous averaged pixels of human bodies and backgrounds.

### 4.2 Compare with SOTA super-resolution models

We also compare the generated image quality with several SR baselines, including **Bicubic**, **MDSR** [10], **LIIF** [3], and the original **TecoGAN** [4] model. As depicted in Fig. 10, it can be seen that general CNN-based image super-resolution


**Fig. 4. Visualization of the Large split set in FIFA2K.** For the images in the Large set, each test bounding box contains 64 × 64 pixels cropping from 1K resolution videos. Our method is able to recover distorted lines, facial expressions, and also clothes patterns into 4K resolution.

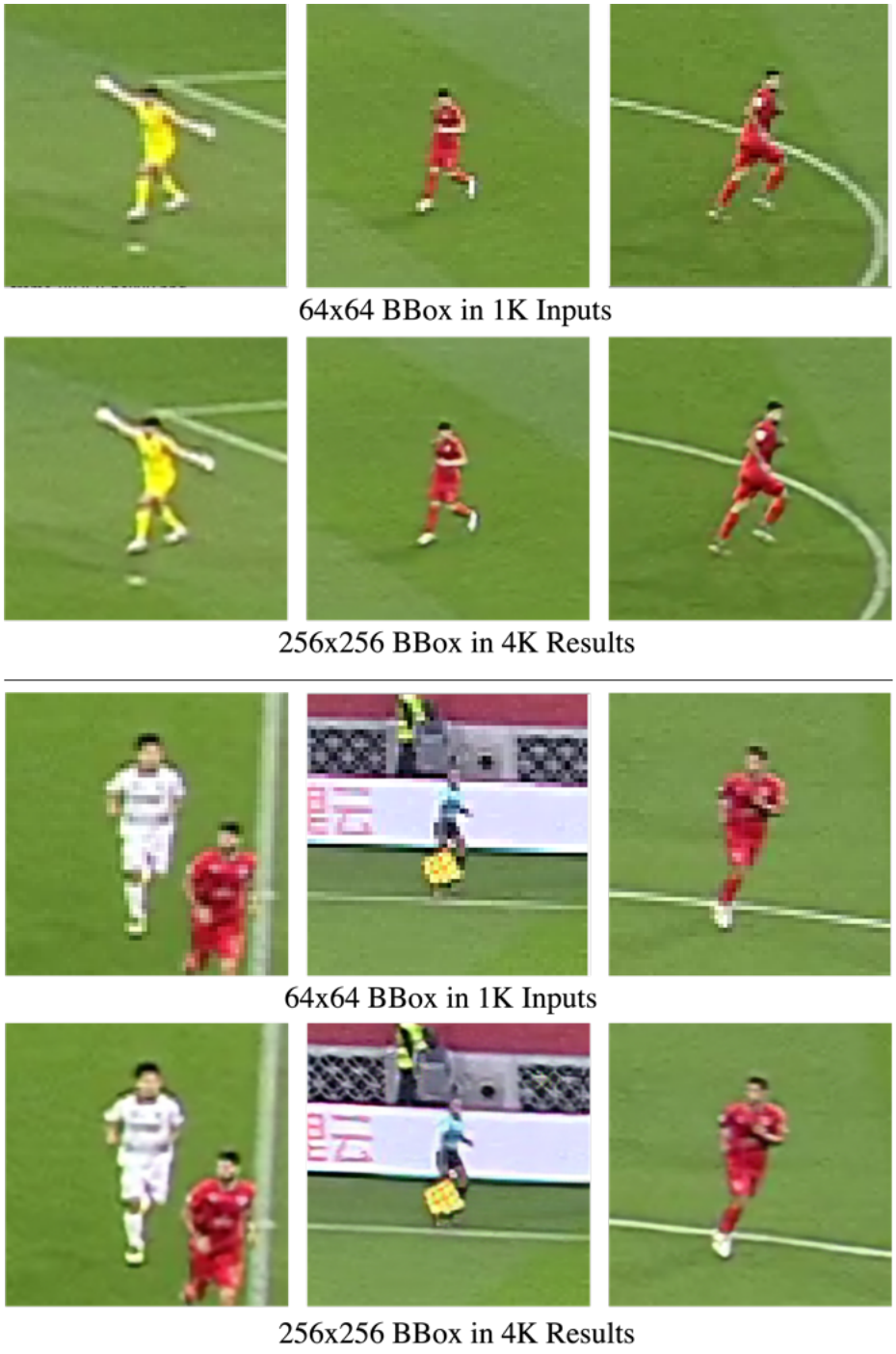


**Fig. 5. Visualization of the Medium split set in FIFA2K.** For the images in the Medium set, each test bounding box contains 64 × 64 pixels cropping from 1K resolution videos. Our method is able to recover distorted lines, clear human outlines, and complete bodies into 4K resolution.

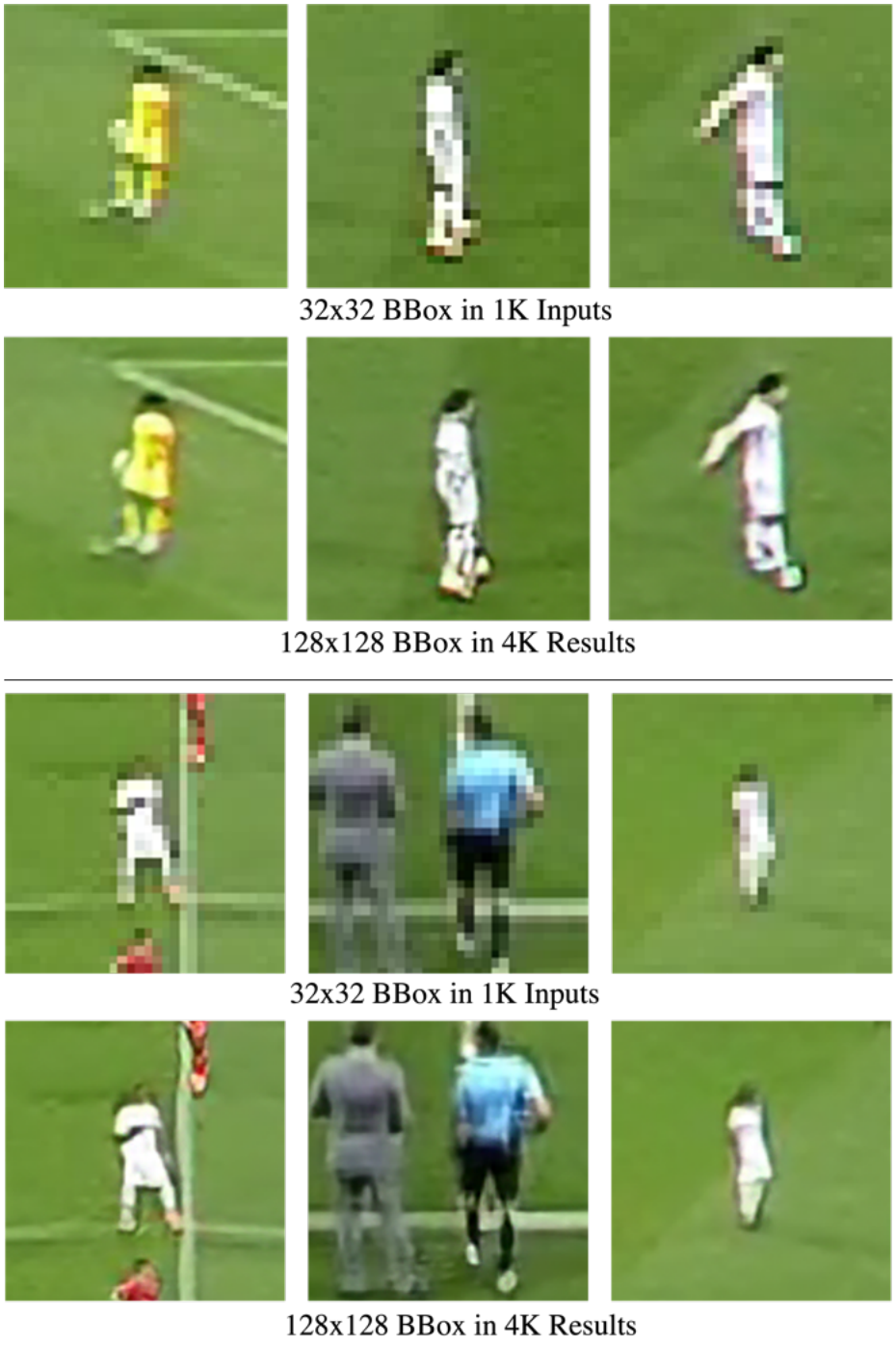


**Fig. 6. Visualization of the Small split set in FIFA2K.** For the images in the Small set, each test bounding box contains $32 \times 32$ pixels cropping from 1K resolution videos. Our method is able to recover distorted lines and synthesize possible human shape from the limited and challenging inputs.


**Fig. 7. Visualization of the Small split set in FIFA2K.** For the images in the Small set, each test bounding box contains $32 \times 32$ pixels cropping from 1K resolution videos. Our method is able to recover distorted lines and synthesize possible human shape from the limited and challenging inputs.


**Fig. 8. Visualization of the Small split set in FIFA2K.** For the images in the Small set, each test bounding box contains $32 \times 32$ pixels cropping from 1K resolution videos. Our method is able to recover distorted lines and synthesize possible human shape from the limited and challenging inputs.


**Fig. 9. Temporal smoothness comparison of state-of-the-art SR models.** We compared our method with baselines on the Small split set. We iteratively synthesized 10 continuous frames and compare the consistency between them. We note that LIFF could not use temporal information and was easily affected by noise. TecoGAN generated the human bodies with many noisy patterns while ours could generate more stable and desirable results.

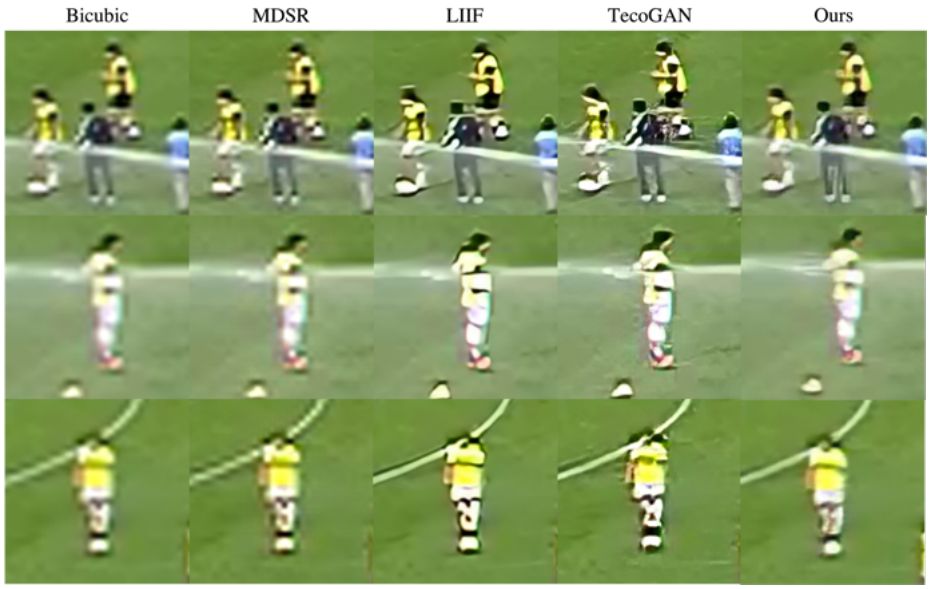


**Fig. 10. Comparison with state-of-the-art SR models.** We compared our method with baselines on the Small split set. We zoomed in the resulting 128 × 128 patches and compared the quality of generated human bodies. Note that Bicubic and MDSR both produced blurry results, while LIIF and TecoGAN could generate more sharper results, they also introduced noise near the human bodies.

methods still yielded blurry results just as the bicubic upsampling. Continuous representation based method LIIF and video based method TecoGAN were able to produce sharper results, however, they also produce noises, artifacts and uncompleted human bodies in the video. This is because these methods are trained on general sequences and without the knowledge of human bodies structures.

In Fig. 9, we also compare the temporal smoothness of single image-based models (LIIF) and video-based models (TecoGAN, Ours). LIIF failed to generate consistent video results as it took no previous frame information as model inputs. TecoGAN also suffered from noises around human bodies and made it difficult to synthesize stable output results. We showed that our methods is more preferable to produce clean, sharp and smooth video clips of human bodies.

### 4.3 Optimal input/output resolution settings

We are also curious about the quality of output videos if given input vdieos of different resolutions. That is, we want to find out is it possible to achieve infinite upscaling by recursively applying our model many times? or there will be a optimal input resolution for the model? To this end, we conduct an ablative experiment which provides different input resolution during inference.

We first analyzed **FIFA2K** under two different input settings during inference, i.e. 1K → 4K and 2K → 8K. The results is shown in Fig. 11, interestingly even the input video has at higher resolution (2K), the quality of output video did not become better than the results of 1K inputs. We initially hypothesize the overfitting to the 4K training videos might cause this problem, thus we retrain our model using a setting of 2K → 4K and perform another similar comparison in Fig. 12. Surprisingly, the results of upsampling 2K video to 4K demonstrated the same behavior like Fig. 11, and 1K to 2K were still better. After ruling out the possibility of overfitting, we argue this is because of the noise level from the original video. That is to say, each video might has its own optimal resolutions that that can be recovered by SR models. This range is usually decided by the original equipment factors and filming conditions. If the noise is about the same scale of the content pixels, then it would be difficult to tell the difference between them.

To further verify this hypothesis, we additionally exploit a 8K video with very low noise level to conduct the ablative study of input resolutions. The result of different resolutions is depicted in Fig. 13. Unlike the previous example which has a limitation of 2K resolution, we are able to achieve super-resolution to 32K from the 8K inputs. However, we still note that the recovered 8K frames still have a gap between the ground-truth 8K frames, which introduce extra noises that might propagate to higher resolutions. Hence we can confirm that the noise level of the original inputs is highly correlated to the optimal output resolutions.

## 5 Future Directions

### 5.1 Multi-frame oversampling strategy

Instead of using single frame during the auto-regressive procedure, we can use a larger sliding window to gather more temporal information. Details can be found in the related works [13,14,18,1]

### 5.2 Cycle-consistency learning

Cycle-consistency [21,22] is a strategy to improve the self-supervised learning scheme. For instance, we can train two different models for upscale the input into x2 and x4 sizes. The cycle consistency can be applied to these two model, such that (x2)x2 should be similar to x4 or (x4)x0.5 should be similar to x2 ...etc.

### 5.3 Attentional matching

The problem can also be solved on database oriented approaches. The existing frame work learns a general inverse function to recover low-resolution frames into high-resolution. We can also apply the concept of memory network or attention mechanism [17,12] so that the query image input can use the information from other highly correlated image patches.

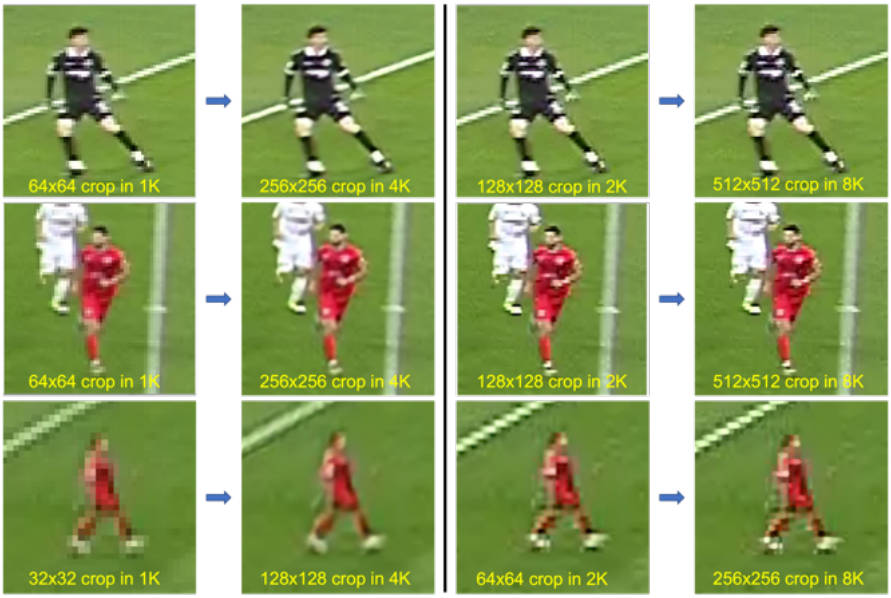


**Fig. 11. Ablative study for different video resolutions during inference.** The model is trained on Football4K 1K → 4K. We tested the model using 1K and 2K input video from FIFA2K. We found that the improvements of 2K → 8K were not obvious.

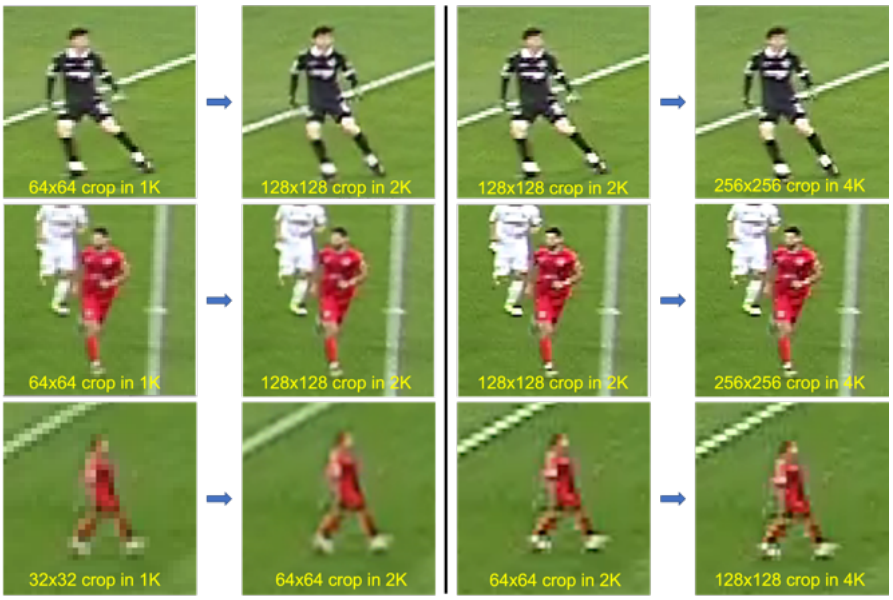


**Fig. 12. Ablative study for different video resolutions during inference.** The model is trained on Football4K 2K → 4K. We tested the model using 1K and 2K input video from FIFA2K. Surprisingly, 1K → 2K showed better visual quality.

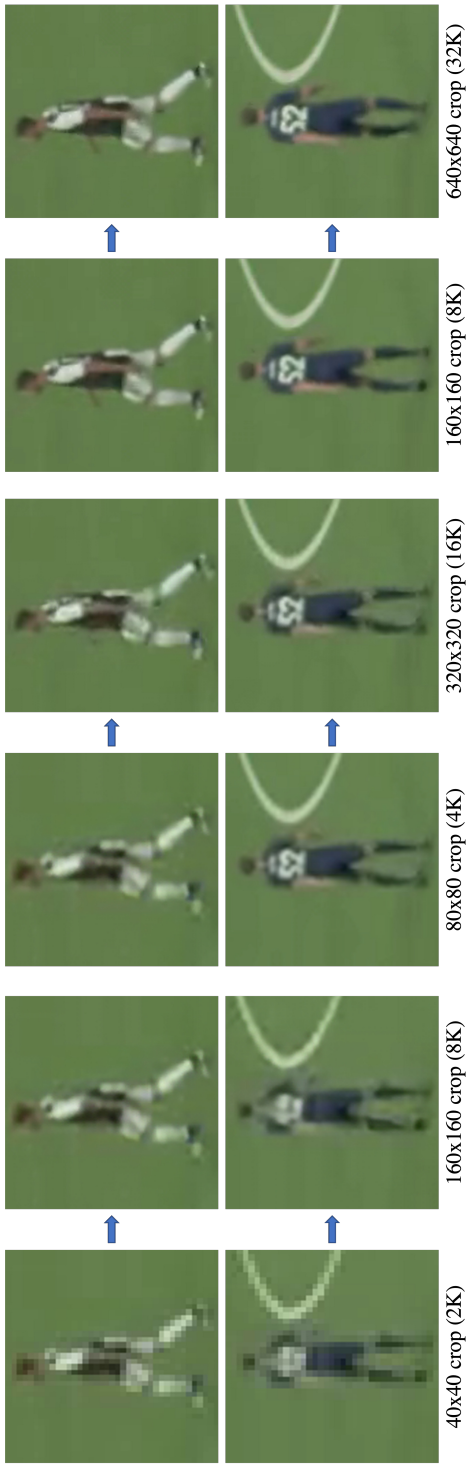


**Fig. 13. Ablative study for different video resolutions using lower noise level 8K video inputs.** We performed inferences of different input resolutions on our collected 8K video by downsample the 8K video into 2K and 4K. Unlike previous experiment, the higher input resolution, the better results we got.

## References


1. Cao, J., Li, Y., Zhang, K., Van Gool, L.: Video super-resolution transformer. arXiv (2021)
2. Chang, H., Yeung, D.Y., Xiong, Y.: Super-resolution through neighbor embedding. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). vol. 1, pp. I–I. IEEE (2004)
3. Chen, Y., Liu, S., Wang, X.: Learning continuous image representation with local implicit image function. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 8628–8638 (2021)
4. Chu, M., Xie, Y., Mayer, J., Leal-Taixé, L., Thuerey, N.: Learning temporal coherence via self-supervision for gan-based video generation. ACM Transactions on Graphics (TOG) **39**(4), 75–1 (2020)
5. Dong, C., Loy, C.C., He, K., Tang, X.: Image super-resolution using deep convolutional networks. IEEE Transactions on Pattern Analysis and Machine Intelligence (TPAMI) **38**(2), 295–307 (2015)
6. Hardy, P., Dasmahapatra, S., Kim, H.: Super resolution in human pose estimation: Pixelated poses to a resolution result? CoRR **abs/2107.02108** (2021), https://arxiv.org/abs/2107.02108
7. Kim, J., Lee, J.K., Lee, K.M.: Deeply-recursive convolutional network for image super-resolution. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)
8. Kim, J., Lee, J.K., Lee, K.M.: Accurate image super-resolution using very deep convolutional networks. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 1646–1654 (2016)
9. Ledig, C., Theis, L., Huszár, F., Caballero, J., Cunningham, A., Acosta, A., Aitken, A., Tejani, A., Totz, J., Wang, Z., et al.: Photo-realistic single image super-resolution using a generative adversarial network. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR). pp. 4681–4690 (2017)
10. Lim, B., Son, S., Kim, H., Nah, S., Lee, K.M.: Enhanced deep residual networks for single image super-resolution. In: Proceedings of IEEE Conference on Computer Vision and Pattern Recognition Workshop (CVPRW) (July 2017)
11. Mao, X., Li, Q., Xie, H., Lau, R.Y., Wang, Z., Paul Smolley, S.: Least squares generative adversarial networks. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV). pp. 2794–2802 (2017)
12. Oh, S.W., Lee, J.Y., Xu, N., Kim, S.J.: Video object segmentation using space-time memory networks. In: Proceedings of the IEEE International Conference on Computer Vision (ICCV). pp. 9226–9235 (2019)
13. Richard, A., Cherabier, I., Oswald, M.R., Tsiminaki, V., Pollefeys, M., Schindler, K.: Learned multi-view texture super-resolution. In: International Conference on 3D Vision (3DV). pp. 533–543. IEEE (2019)
14. Rozumnyi, D., Oswald, M.R., Ferrari, V., Pollefeys, M.: Shape from blur: Recovering textured 3d shape and motion of fast moving objects (2021)
15. Simonyan, K., Zisserman, A.: Very deep convolutional networks for large-scale image recognition. arXiv preprint arXiv:1409.1556 (2014)
16. Tian, Y., Zhang, Y., Fu, Y., Xu, C.: Tdan: Temporally-deformable alignment network for video super-resolution. In: Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition. pp. 3360–3369 (2020)

17. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A.N., Kaiser, Ł., Polosukhin, I.: Attention is all you need. In: Advances in Neural Information Processing Systems (NIPS). pp. 5998–6008 (2017)
18. Wang, X., Yu, K., Chan, K.C., Dong, C., Loy, C.C.: BasicSR: Open source image and video restoration toolbox. https://github.com/xinntao/BasicSR (2020)
19. Wu, Y., Kirillov, A., Massa, F., Lo, W.Y., Girshick, R.: Detectron2. https://github.com/facebookresearch/detectron2 (2019)
20. Zhang, Y., Tian, Y., Kong, Y., Zhong, B., Fu, Y.: Residual dense network for image super-resolution. In: Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)
21. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networkss. In: Computer Vision (ICCV), 2017 IEEE International Conference on (2017)
22. Zhu, J.Y., Zhang, R., Pathak, D., Darrell, T., Efros, A.A., Wang, O., Shechtman, E.: Toward multimodal image-to-image translation. In: Advances in Neural Information Processing Systems (NIPS) (2017)