TurboVSR: Fantastic Video Upscalers and Where to Find Them

Supplementary Material

6. Training details

Video data augmentation. Due to the scarcity of high-quality, high-resolution videos and the availability of high-quality, high-resolution images, we design a video data augmentation method based on static images. This is achieved by generating pseudo-videos from static images using affine transformations, including random translation, rotation, and scaling. We carefully tune the ranges of these transformation parameters to ensure the pseudo-videos exhibit motion comparable to real videos .

Training Strategy. Image-video mixed training is achieved through alternating batches. To mitigate overfitting to training degradation, we employed following improvements: (1) add 0-300 step random DDPM noise degradation to the condition latent; (2) fix the FFN of the network and trained other parameters; (3) while training image/video super-resolution (ISR/VSR), we also train text-to-image and text-to-video generation by randomly dropping the LR condition, for preserving the model's generation capabilities.

7. Qualitative Comparison

We show several qualitative comparison with existing VSR methods in Figure 8. Overall, TurboVSR presents detail generation capability on par with or even superior to state-of-the-art methods.

8. Details on 4K Resolution Image SR

For 4K image SR, we divid the training into two stages, both of which utilize the same training dataset: a private 4K image dataset containing approximately 2 million samples. This dataset includes diverse contents such as portraits, landscapes, and animals, most of which are high-quality professionally generated content (PGC).

In the first stage, we train a our model on text-to-image (T2I) generation task at a resolution of 2048×2048. The primary reason for this choice is that the LTX pre-trained model has not been trained at such high resolutions, making it unsuitable as a direct pre-training model for 4K super-resolution. We initialized the training with the official weights of LTX-DiT and fine-tuned it for 35K iterations with a batch size of 256. Figure 7 shows two examples generated during this stage. We observed that the model trained in this phase is capable of generating high-quality details, especially in domains such as portraits and animals, with sharp and rich details in hair and facial features. However, similar to other text-to-image generation models, it struggles with generating anatomically correct limbs and fingers, and these issues are slightly more pronounced compared to state-of-the-art models. We attribute these limitations to the lack of high-resolution training in LTX and the inherent challenges of learning from highly compressed, high-dimensional latent spaces.

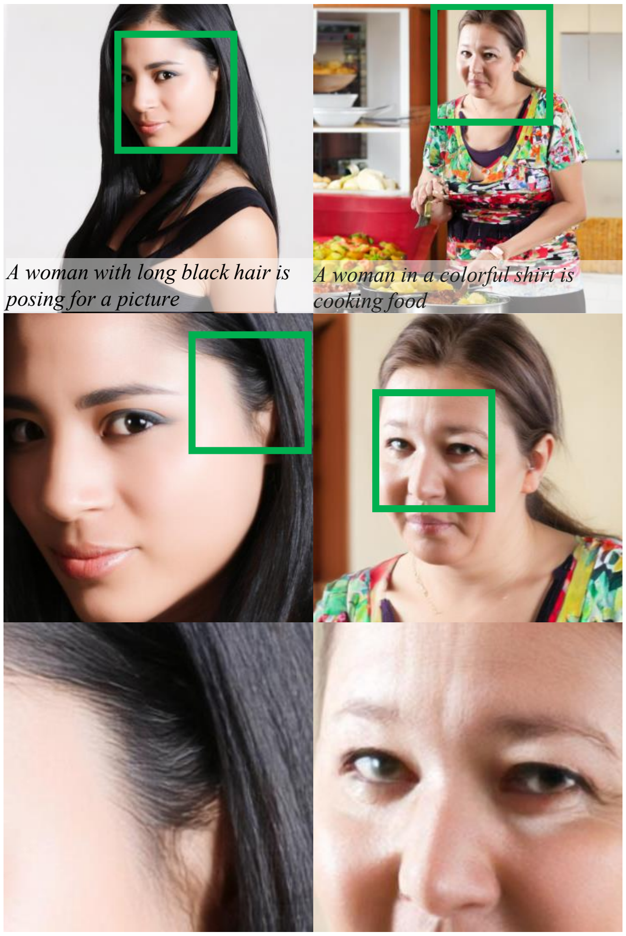
A woman with long black hair is posing for a picture
A woman in a colorful shirt is cooking food

Figure 7. Examples of our high-resolution T2I generation pre-training (2048×2048). Based on our high compression ratio autoencoder, we show satisfactory detail generation capability can be achieved.

In the second stage, we train on image SR, using model weights from the first stage as initialization. The training is conducted with a batch size of 256 for 30K iterations. We show several results in Figures 10, 9, 11, 12, and 13. It can be observed that, in the SR task, our model performs especially well on portraits and also achieves remarkable results on landscapes and architecture. Overall, if a scene is handled well by the text-to-image generation model in the first stage, it is also effectively processed by the super-resolution model in the second stage.

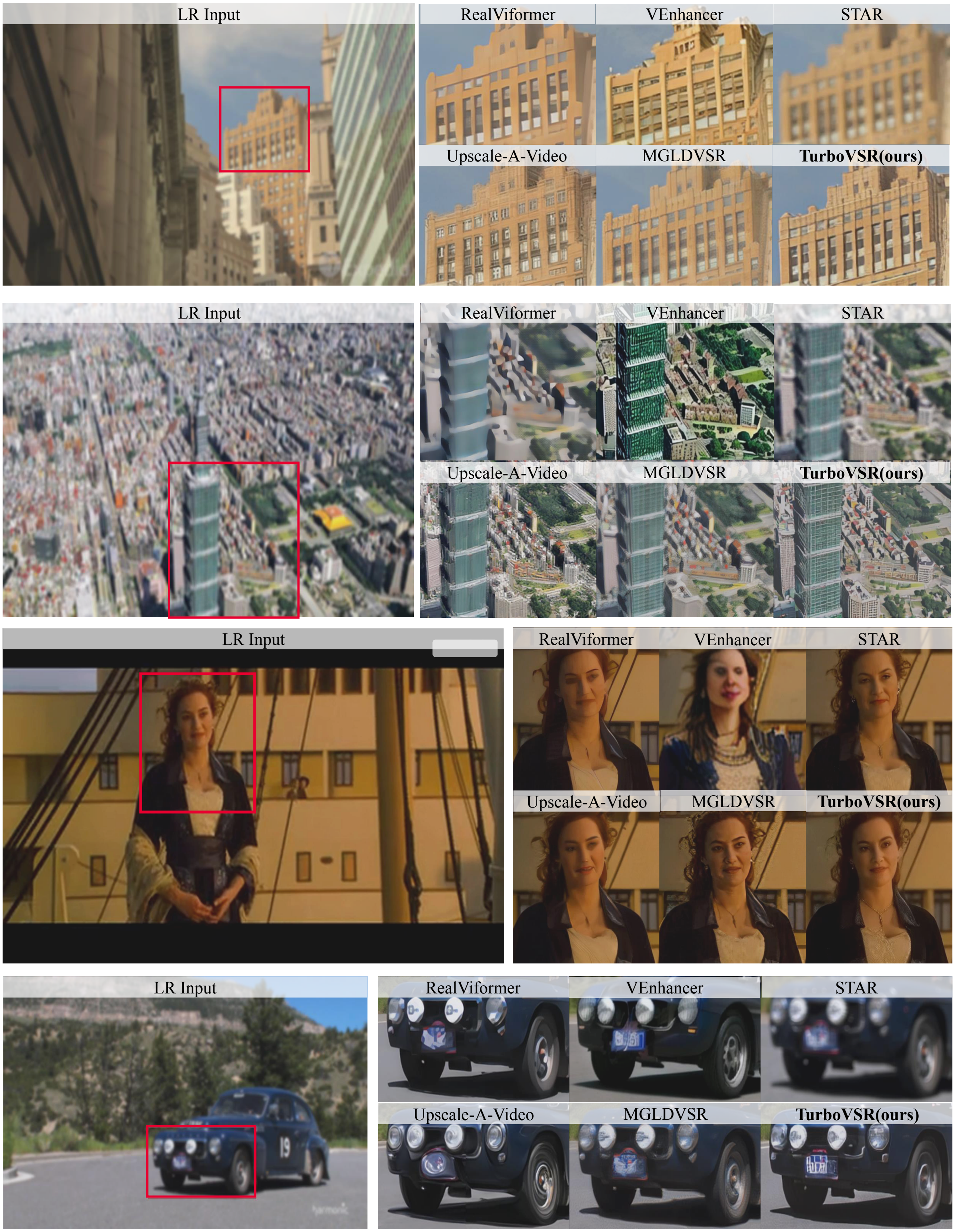

Figure 8. Qualitative comparison with existing VSR methods.

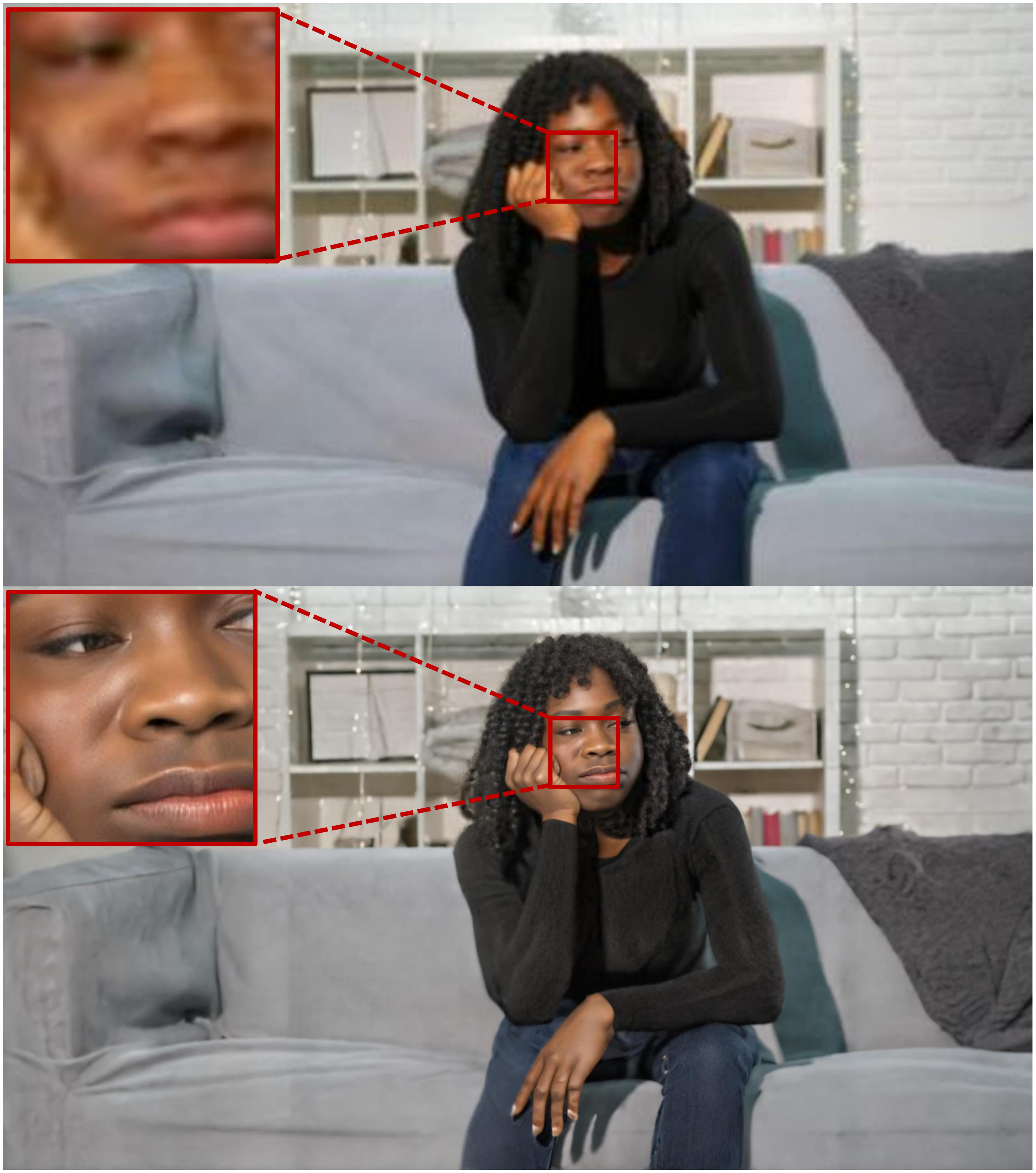

Figure 9. Example results on 4K image super-resolution (3648×2048) .**Top:** Input low resolution image. **Bottom:** TURBOVSR predicted high resolution image.

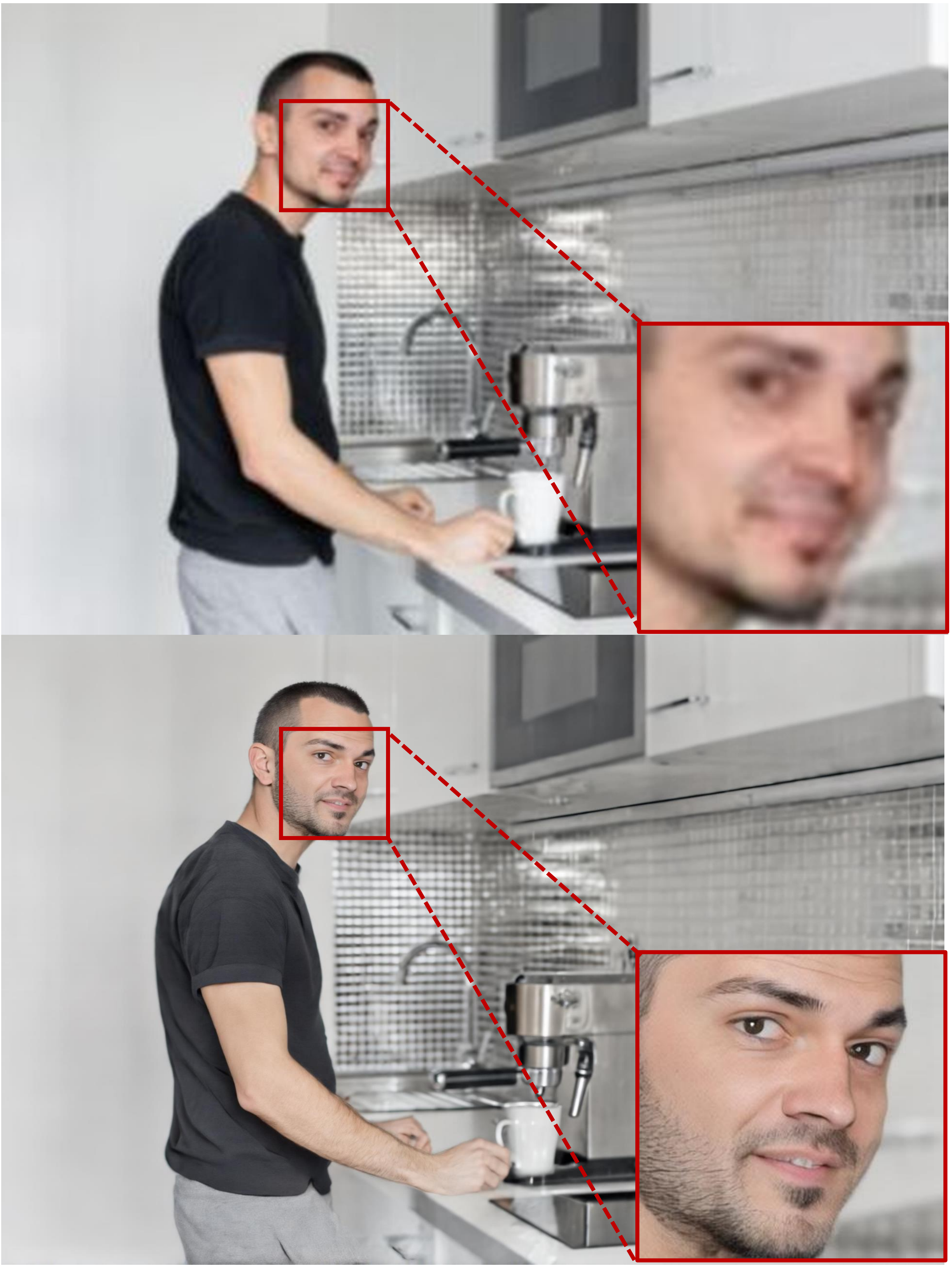

Figure 10. Example results on 4K image super-resolution (3072×2048). **Top:** Input low resolution image. **Bottom:** TURBOVSR predicted high resolution image

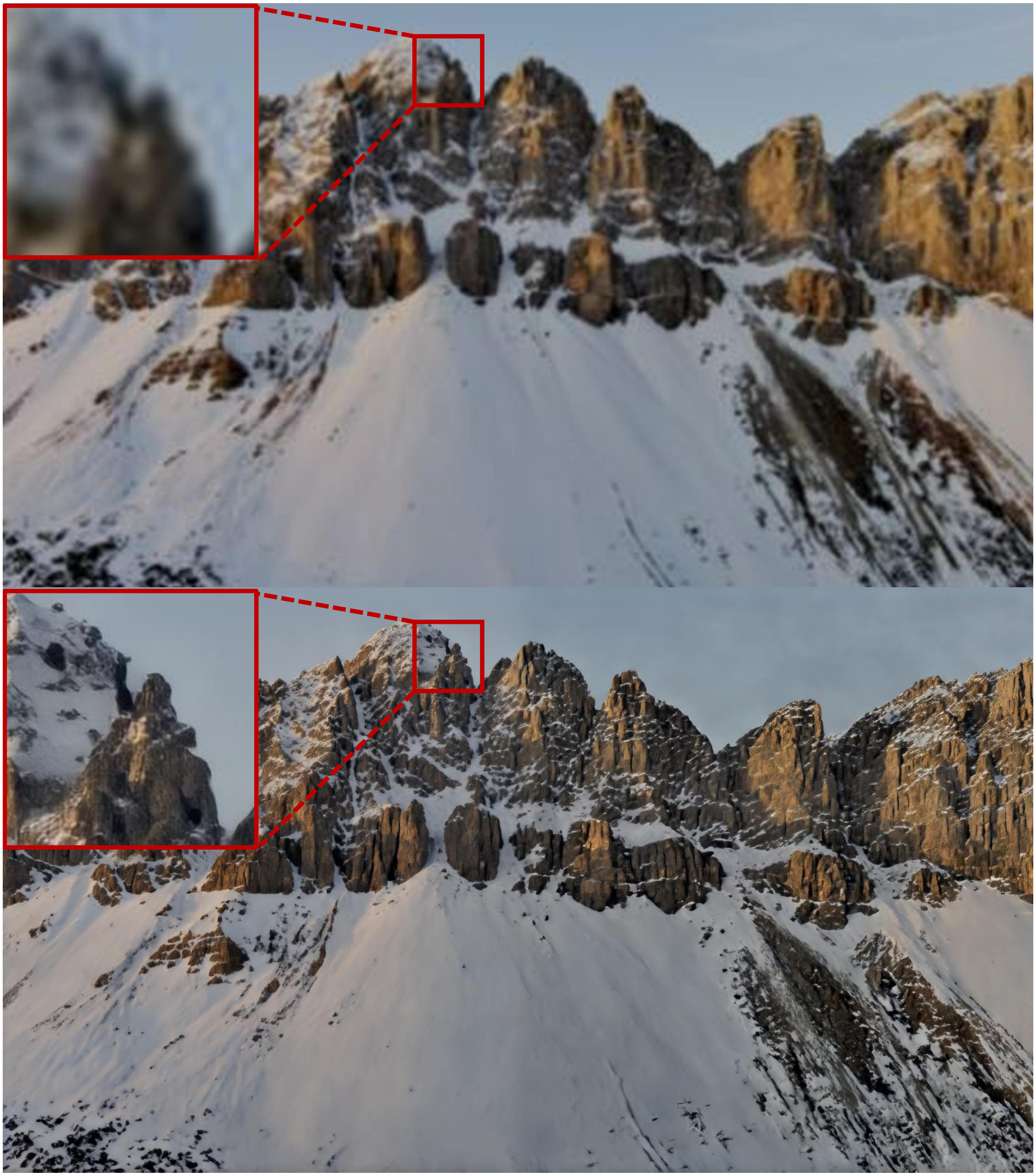

Figure 11. Example results on 4K image super-resolution (3648×2048) .**Top:** Input low resolution image. **Bottom:** TurboVSR predicted high resolution image.

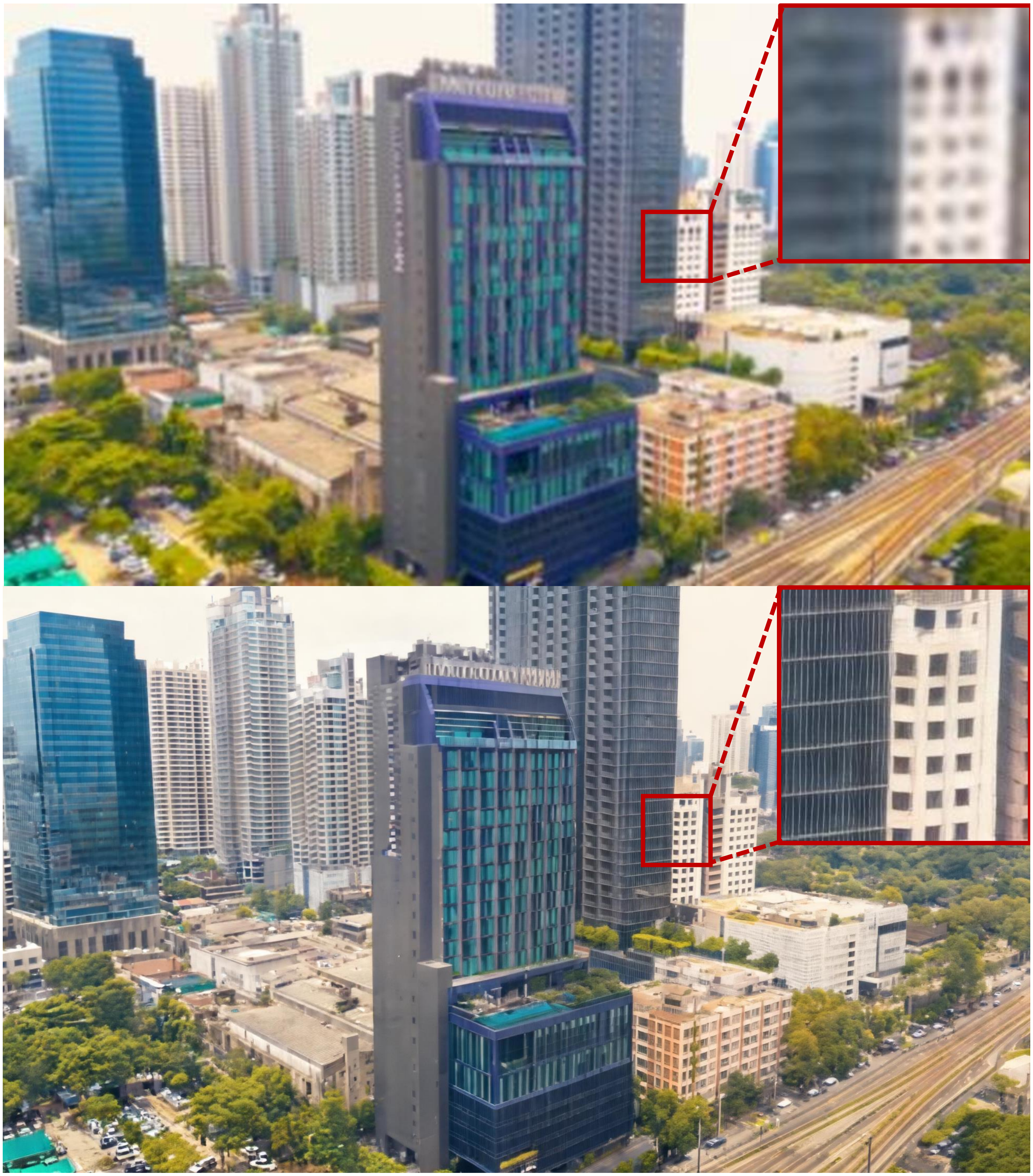

Figure 12. Example results on 4K image super-resolution (3648×2048) .**Top:** Input low resolution image. **Bottom:** TurboVSR predicted high resolution image.

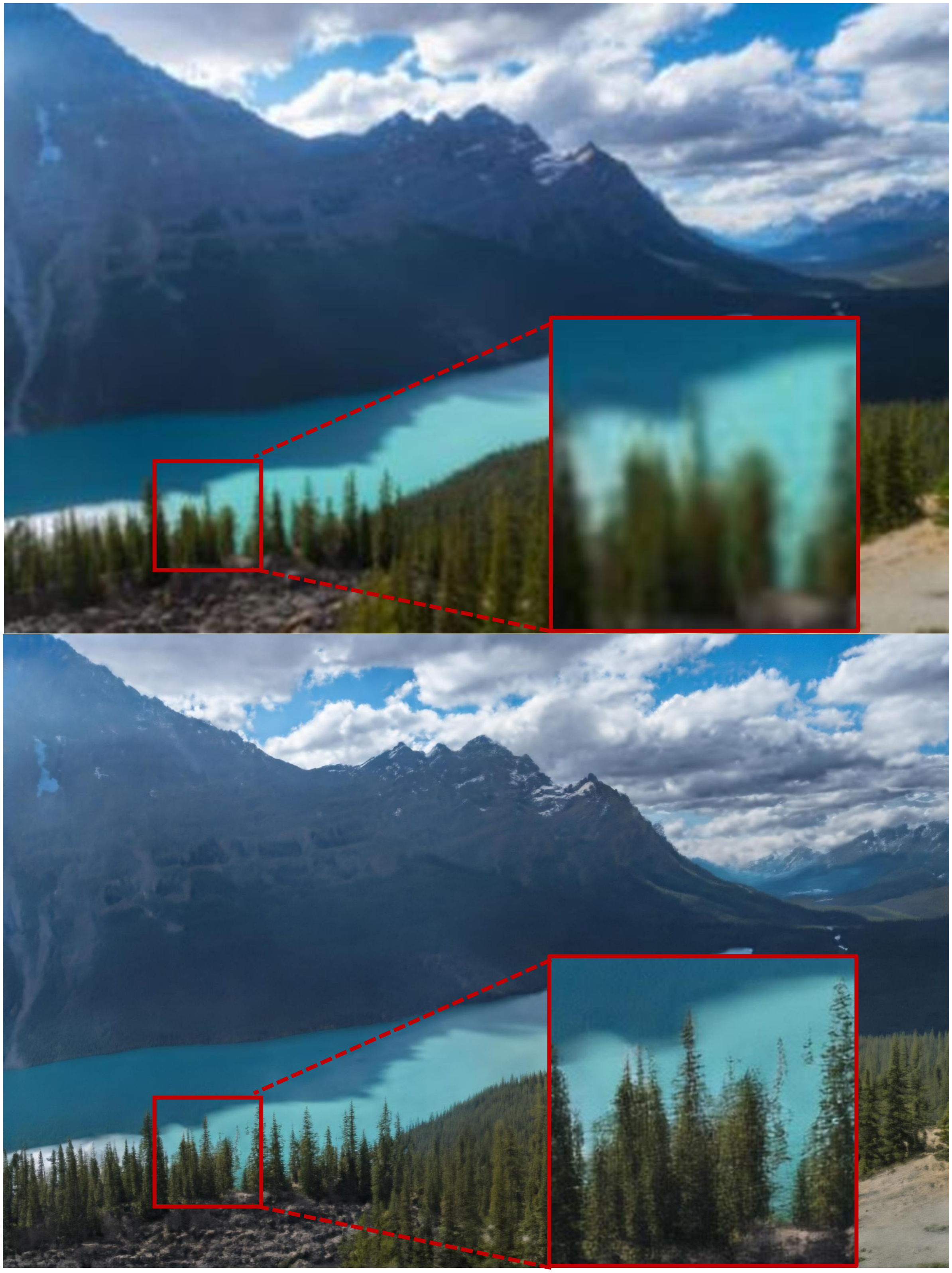

Figure 13. Example results on 4K image super-resolution (3072×2048). **Top:** Input low resolution image. **Bottom:** TURBOVSR predicted high resolution image