Supplementary Material of SASIC: Stereo Image Compression with Latent Shifts and Stereo Attention

Matthias Wödlinger
TU Wien, Vienna, Austria
mwoedlinger@cvl.tuwien.ac.at

Jan Kotera
TU Wien

Jan Xu
Deep Render, London, UK

Robert Sablatnig
TU Wien

1. MS-SSIM rate-distortion curves

Figure 1 contains the rate-distortion curves for the different models. The neural network based models Backbone and SASIC are optimized with MSE as the distortion metric. The results for HESIC [1], and DSIC [2] are the scores reported by their respective authors also when trained with MSE as the distortion metric.

The proposed method either outperforms or is on par with the benchmark methods and outperforms the Backbone alone (without stereo components), especially at low bitrates. The results of HESIC appear very good at MS-SSIM but rather bad at PSNR (see Fig. 4 of the main paper), in both cases with a large margin to other methods. Although it is not specified in the paper, it appears from the available codebase that the authors of HESIC did not use the dataset images in their original resolution but performed some pre-processing, and as a consequence, the obtained results may not be comparable.

2. Runtimes

To compute the runtimes for our method we averaged the runtimes over all 512×512 center crops of the cityscapes test set. The experiments where performed on a NVIDIA GeForce 3090 GPU and a batch size of 1. The results can be found in Tab. 1. Encoding corresponds to the time needed for producing $\hat{\boldsymbol{y}}_1, \hat{\boldsymbol{y}}_{\text{res}}$ from inputs $\boldsymbol{x}_1, \boldsymbol{x}_2$. Decoding corresponds to the time needed for producing $\hat{\boldsymbol{x}}_1, \hat{\boldsymbol{x}}_2$ from $\hat{\boldsymbol{y}}_1, \hat{\boldsymbol{y}}_{\text{res}}$. To measure saving times of the bitstream we measured the average times for saving the latent tensor to disk. Loading times for encoding correspond to the average time needed to load the images and move them to GPU, loading times for decoding measure the average times for loading the latent saved by the encoder.

Operation	Time [s]
Encoding	0.017
Encoding + saving	0.220
Encoding + loading + saving	0.277
Decoding	0.050
Decoding + loading	0.073

Table 1. Runtimes for our method when measured on a NVIDIA GeForce 3090 GPU.

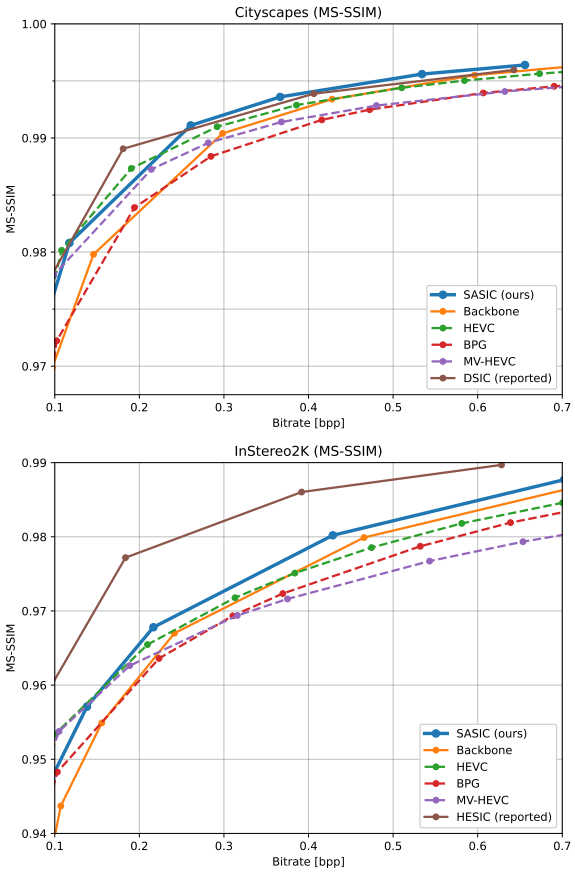

Figure 1. Rate distortion curves of the proposed method and various compression baselines on the Cityscapes (top) and InStereo2K (bottom) datasets measured by MS-SSIM

For comparison, HEVC run on CPU achieves in average approximately 3 seconds for encoding at low bitrates

(bpp=0.1) and 8 seconds at high bitrates (bpp=1.0) and 0.04s for decoding of the same image pairs. The proposed method therefore has the advantage of faster encoding times (especially with bulk processing) that is independent of the target bitrate and disadvantage of slower decoding times and higher hardware demands.

3. Qualitative comparison

We provide qualitative comparisons on example images from the cityscapes and InStereo2K test sets in Fig. 2 and Fig. 3.

References

[1] Xin Deng, Wenzhe Yang, Ren Yang, Mai Xu, Enpeng Liu, Qianhan Feng, and Radu Timofte. Deep homography for efficient stereo image compression. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, pages 1492–1501, June 2021. 1

[2] Jerry Liu, Shenlong Wang, and R. Urtasun. DSIC: Deep stereo image compression. *2019 IEEE/CVF International Conference on Computer Vision (ICCV)*, pages 3136–3145, 2019. 1

Figure 2. A qualitative comparison on an image from the Cityscapes test set. We show the same image in both columns with two different zoomed out regions.

Figure 3. A qualitative comparison on an image from the InStereo2K test set.