Efficient Flow-Guided Multi-frame De-fencing (supplemental material)

Stavros Tsogkas Fengjia Zhang Allan Jepson Alex Levinshtein
Samsung AI Center Toronto
101 College St., Toronto, ON, Canada, M5G 1L7
{stavros.t, f.zhang2, allan.jepson, alex.lev}@samsung.com

1. Data

1.1. Synthetic Data Augmentation

Background augmentation. We source background scenes (which are also used as ground truth during training and evaluation) from Vimeo-90k [6], which consists of videos depicting every day activities in realistic settings, often including people and other objects. We specifically use the original test split of the dataset[1], which contains sequences of seven (7) frames. The original clean frames are used as ground truth for training and evaluation. To increase variability of our synthetically generated data, we apply the following data augmentation steps:

1. random homography transformation
2. center cropping to avoid any black borders caused by (1).
3. random cropping of a 320×192 window, which are the frame dimensions used during training.
4. random horizontal flip.

Foreground augmentation. The foreground fence obstructions are sourced from the De-fencing dataset [2], which contains 545 training and 100 test images with fences, along with corresponding binary masks as ground truth for the fence segmentation. The variability of fences in that dataset is limited, so we also apply various forms of data augmentation on the fence image before fusing it with the background. The types of foreground augmentation we consider are:

1. random downsample of the fence image and segmentation to create fences of different sizes and thickness.
2. random "outer" window crop to focus on a specific subregion of the fence.
3. color jitter to make the network more robust to different fence appearances and lighting conditions.
4. random perspective distortion to obtain a fence sequence of length K.
5. center cropping to avoid black border effects from the homographic distortion.
6. random blur with a gaussian kernel, to simulate defocus aberrations.

Samples from our synthetic burst dataset are shown in Figure 1.

1.2. Real Burst Collection

Although our synthetic data are carefully generated and exhibit considerable realism and diversity, they still cannot fully capture the variability of motion, lighting, and obstruction patterns in scenes captured under realistic conditions, so we collect set of controlled sequences, specifically for quantitative evaluation. As mentioned in the main paper, rather than collecting toy scenes as in Liu et al. [4], we capture outdoors real world hand-held sequences with a fence and a corresponding background ground truth image without a fence.

Data capture. We first capture one image without the fence as the ground-truth frame, by bringing our camera to the centre of a fence cell. We then fix the focus and exposure on the background and move backwards from the fence to capture 5 frames with fences. To minimize misalignment caused by a change in perspective, we capture the first frame as the key-frame, moving backwards along the capturing direction. Then, we capture the remaining four frames by intentionally jittering the camera around.

Keyframe - groundtruth alignment. After capturing the real bursts, we need to align the ground-truth frame to the obstruced key-frame. We do this following an approach

[1] http://data.csail.mit.edu/tofu/testset/vimeo_test_clean.zip

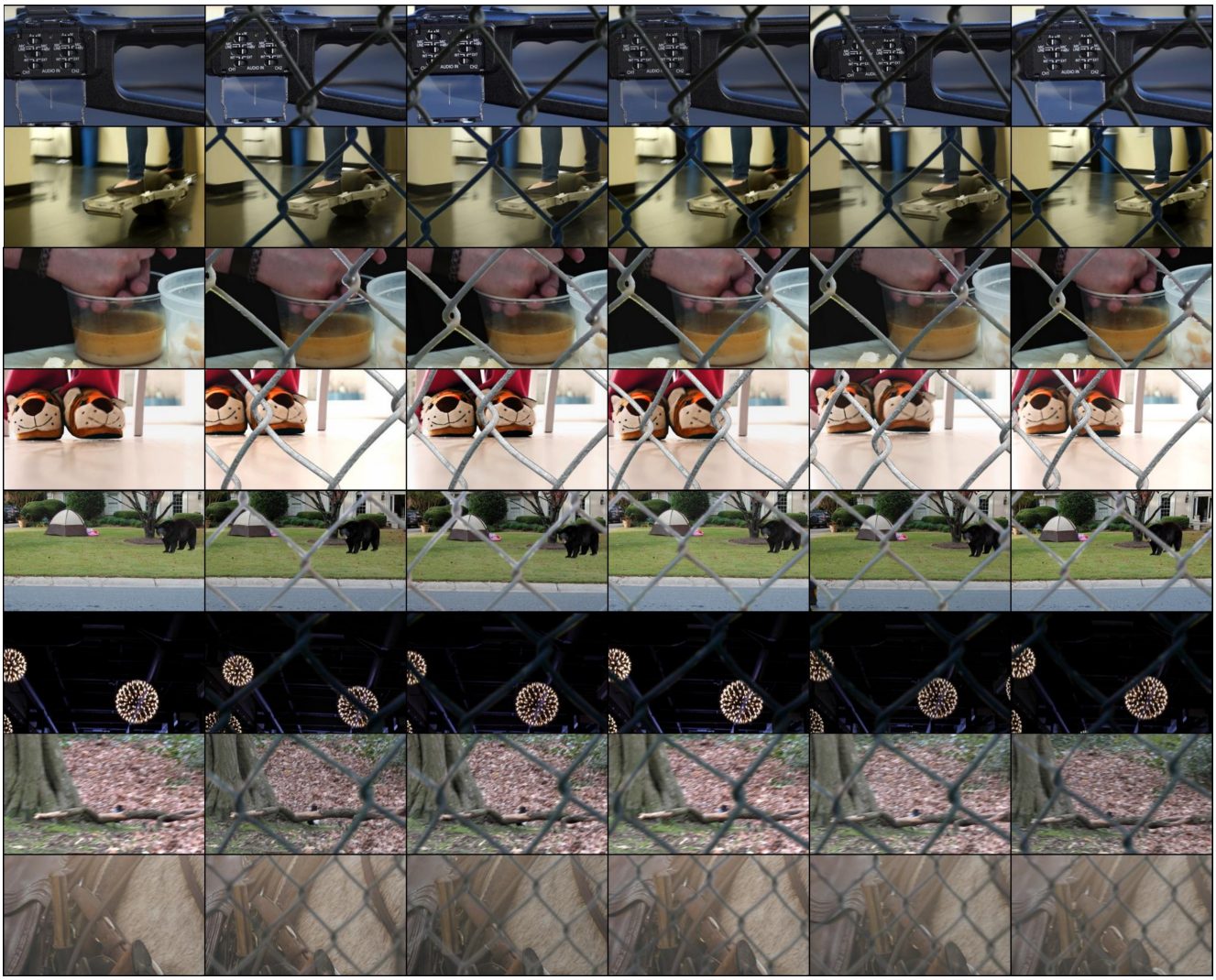
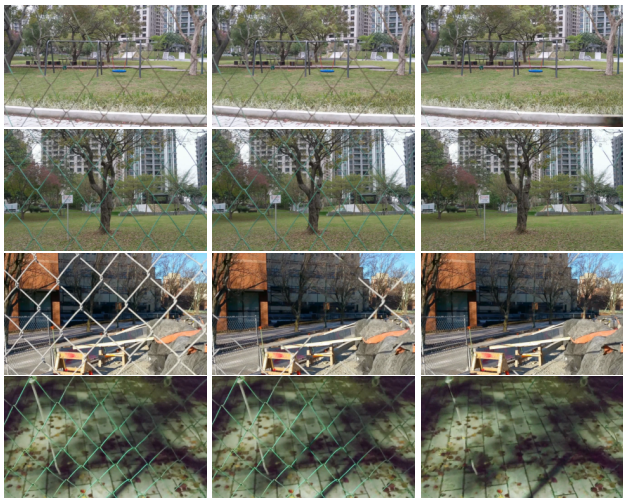


Figure 1: Examples of our synthetically generated data. The leftmost column shows the clean background frame and the next 5 columns show the background burst from Vimeo-90k [6], with overlaid fences from the De-fencing dataset [2].

combining SIFT feature extraction and RANSAC homography estimation, similar to [1]. We start by computing and matching SIFT features in the keyframe and respective clean groundtruth shot. Since the resolution of the original images is high, we extract 320×192 regions in a sliding window fashion, and within such window, P, we compute homography parameters using matched SIFT features in crops of varying sizes: 128^2, 256^2, 512^2, and 1024^2 (larger crop sizes extend beyond the area of the original window). The computed homography parameters are used for global alignment of the keyframe and groundtruth frames, so we have multiple homography "candidates" corresponding to P. The motivation behind computing homographies at different scales is that different parts of a given window P may require different homographies to be aligned more accurately. We assign the best homography to each 128×128 crop C inside P, by computing its respective SSIM score with respect to its warped counterpart in the groundtruth (we use the estimated fence masks to only include non-obstructed areas in the SSIM computation). To ensure a minimum level of quality, if there is at least one C inside P with average SSIM ≤ 0.2 or PSNR ≤ 20, we discard P and move to the next sliding window with stride 128. If there are no "failed" crops, P slides to the next non-overlapping position. In the end, we also manually filter out the crops that are misaligned on and near the fences by visual comparison between input and aligned ground-truth. We also manually filter out crops consisting of mostly homogeneous regions (sky, land, sand), to promote diversity in our dataset. Our final real burst dataset consists of 185 320×192 input

bursts with corresponding ground truth key-frames from 29 scenes. Samples from our real burst dataset are shown in Figure 3.

2. Task Specificity and Comparison with SOLD [4]



(a) Keyframe (b) Output (original) (c) Improved output

Figure 2: Better data augmentation can make the fence segmentation network more robust to varied types of fences, thus improving the quality of frame inpainting on real sequences *without* the need for online finetuning.

One potential criticism towards our approach is our focus on a specific type of obstruction (fences), and the fact that we heavily rely on a specific prior (pre-trained fence segmentation model), which can harm generalization to new inputs, not commonsly seen in the training data. In comparison, SOLD [4] is a multi-frame approach can handle various types of obstructions. However, SOLD is also limited when faced with atypical obstructions (e.g., fences), requiring scene-specific, costly online optimization that takes $\sim$ 3 minutes, to achieve good results, making it impractical for real-world application. Our method trades-off generality for reconstruction and runtime performance (the latter is a feature missing from previous de-fencing works), producing better de-fencing results than SOLD, at a fraction of its runtime, without requiring scene-specific optimization. Besides, de-fencing is an important problem in its own right, with an extensive literature in computer vision (see Section 2.1 in the main paper). Finally, we can make our method more robust to a broader variety of fences (e.g., rhombic rotated fences, etc.) by improving our data augmentation protocol. To showcase this, we have added more scale, rotation, shape and color variation during the training of the fence segmentation model. As shown in Figure 2, after adding these additional data augmentations, the fence segmentation model can accurately segment fences that are rotated, very thin fences, or have low contrast with respect to the background, subsequently improving de-fencing quality. Extending our method to handle other types of obstructions (e.g., reflections), is also a direction we are currently exploring.

3. Qualitative Results

Figure 4 shows additional qualitative results on sequences from our synthetically generated test set. Figure 5 shows results on real sequences, taken from previous works [7, 4]. We are also including some failure cases, where the fence segmentation model encounters fences at scales or shapes that are out of its training distribution, resulting in low de-fencing quality. Finally, in Figure 6 we compare results from our method and other baselines on examples from the real burst dataset we collected.

References

[1] Goutam Bhat, Martin Danelljan, Luc Van Gool, and Radu Timofte. Deep burst super-resolution. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*, pages 9209–9218, 2021.

[2] Chen Du, Byeongkeun Kang, Zheng Xu, Ji Dai, and Truong Nguyen. Accurate and efficient video de-fencing using convolutional neural networks and temporal information. In *2018 IEEE International Conference on Multimedia and Expo (ICME)*, pages 1–6. IEEE, 2018.

[3] Chen Gao, Ayush Saraf, Jia-Bin Huang, and Johannes Kopf. Flow-edge guided video completion. In *European Conference on Computer Vision*, pages 713–729. Springer, 2020.

[4] Yu-Lun Liu, Wei-Sheng Lai, Ming-Hsuan Yang, Yung-Yu Chuang, and Jia-Bin Huang. Learning to see through obstructions with layered decomposition. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 2021.

[5] Roman Suvorov, Elizaveta Logacheva, Anton Mashikhin, Anastasia Remizova, Arsenii Ashukha, Aleksei Silvestrov, Naejin Kong, Harshith Goka, Kiwoong Park, and Victor Lempitsky. Resolution-robust large mask inpainting with fourier convolutions. In *Proceedings of the IEEE/CVF Winter Conference on Applications of Computer Vision*, pages 2149–2159, 2022.

[6] Tianfan Xue, Baian Chen, Jiajun Wu, Donglai Wei, and William T Freeman. Video enhancement with task-oriented flow. *International Journal of Computer Vision (IJCV)*, 127(8):1106–1125, 2019.

[7] Tianfan Xue, Michael Rubinstein, Ce Liu, and William T Freeman. A computational approach for obstruction-free photography. *ACM Transactions on Graphics (TOG)*, 34(4):1–11, 2015.

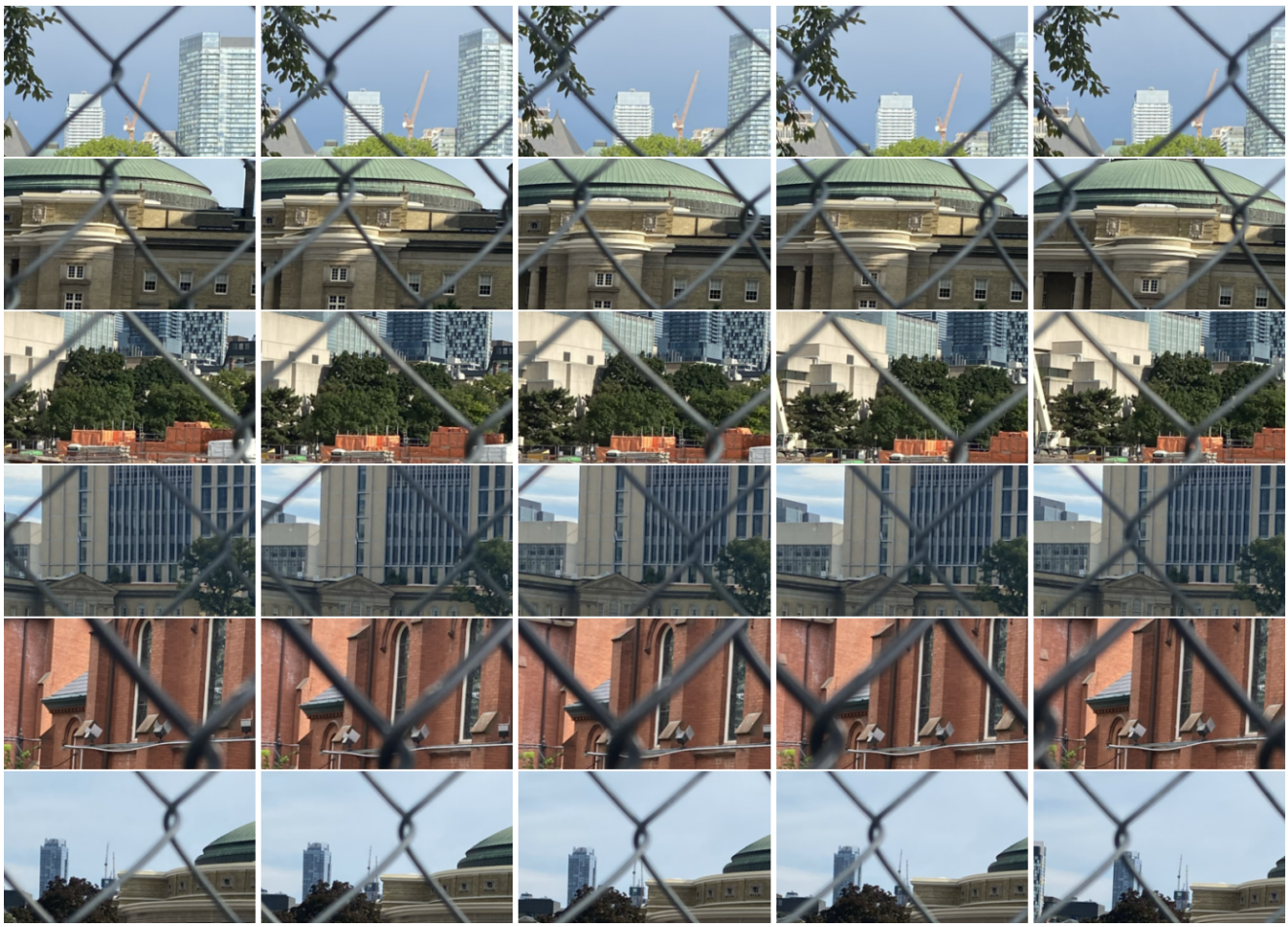
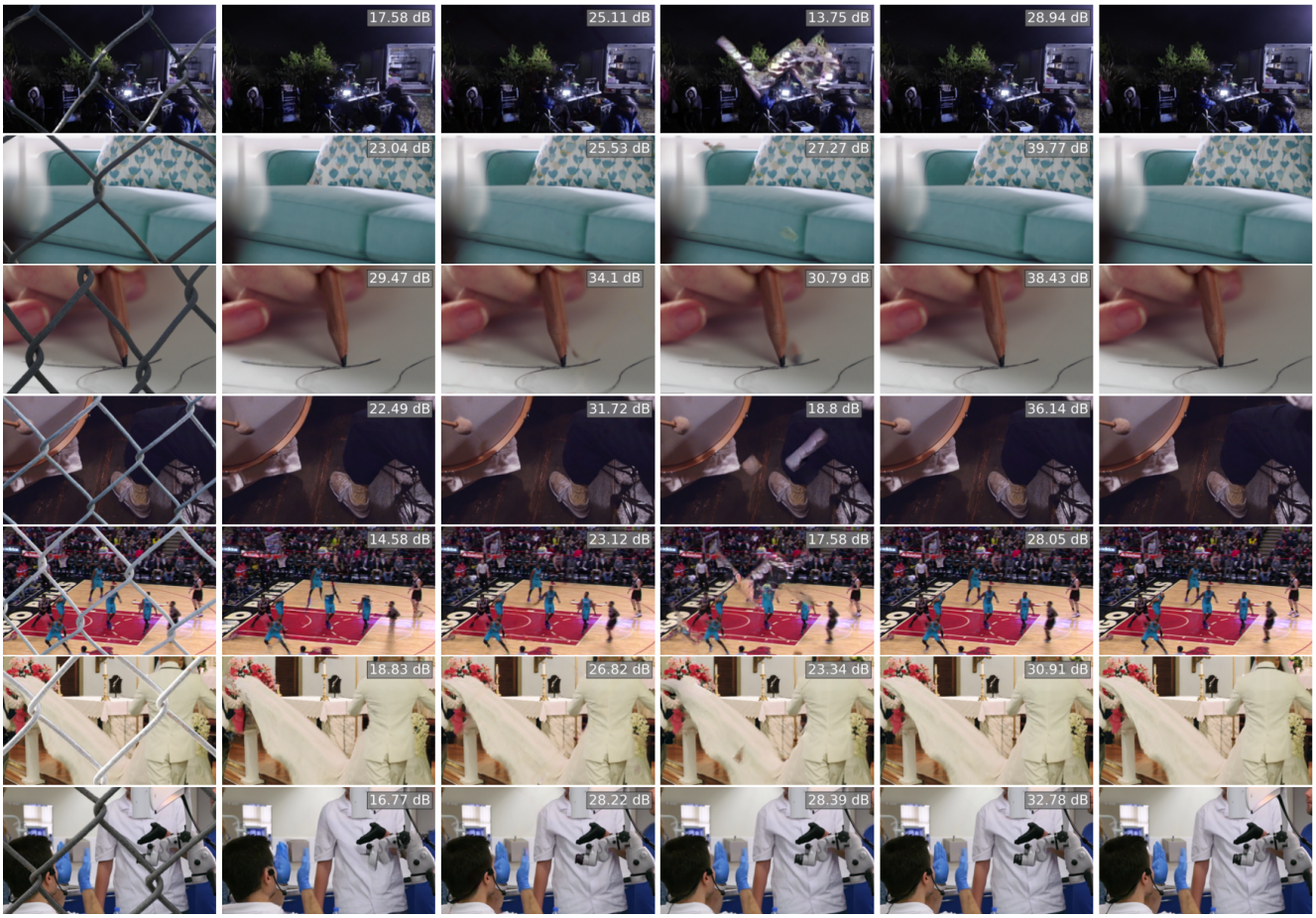


Figure 3: Examples of real bursts we have collected. These are 320×192 crops from the original, high resolution images, after alignment.



(a) Keyframe (b) LaMa [5] (c) SOLD [4] (d) FGVC [3] (e) Ours (f) Ground truth

Figure 4: De-fencing results on sequences from our **synthetic** data, and respective PSNR scores *inside* the fence mask area. The leftmost column shows the obstructed keyframe, and the next 5 rows show its reconstructed version using various baselines and our approach. Zoom in to notice differences in reconstructed frames.

Figure 5: De-fencing results on sequences from [7, 4]. From top to bottom: obstructed keyframe, reconstructed keyframe using our approach, estimated fence segmentation using our U-net fence segmentation model. The second group of results shows failure cases: when the fence obstruction is outside our training distribution (e.g., scale - very thin fences, irregular fence pattern, such as vertical bars, extreme blur etc.) the fence segmentation estimation fails, affecting reconstruction quality. Addressing unusual fence obstructions like these is our main focus for future work.

(a) Keyframe (b) LaMa [5] (c) SOLD [4] (d) FGVC [3] (e) Ours (f) Ground truth

Figure 6: Qualitative de-fencing results on **real** sequences, and respective PSNR scores *inside* the fence mask area. The leftmost column shows the obstructed keyframe, and the next 5 rows show its reconstructed version using various baselines and our approach. Zoom in to notice differences in reconstructed frames.