Learning to Super Resolve Intensity Images from Events

- Supplementary Material -

S. Mohammad Mostafavi I.
GIST, South Korea
mostafavi@gist.ac.kr

Jonghyun Choi
GIST, South Korea
jhc@gist.ac.kr

Kuk-Jin Yoon
KAIST, South Korea
kjyoon@kaist.ac.kr

1. Discussions on Overlapped Stacking

Based on the representation of the event stream stacked over time in Fig. 3 of the main paper, we are able to change the amount of overlap for stacking. This is demonstrated in Fig. 1 where the location of APS frames are shown and events cover different amount of the stream over time as stacks based on how fast the events are fired which is related to the camera or scene speed movement. This means that the size of the colored stacks or the overlaps are not necessarily equal to each other. Two stacks can have common events up to a single event but less common events are desired to produce meaningful different images.

Unlike stacking based on time (SBT), stacking based on number of events (SBN) can consume different amount of time per stack which is related to the amount of events triggered from the scene. Furthermore, a stack might even surpass the location of the previous or next APS frame location and is not bound to the APS. This overlap is useful when there is large amount of scene movement and can prevent short-time fired events from being less effective by having them in more than one stack over the total number of stacks.

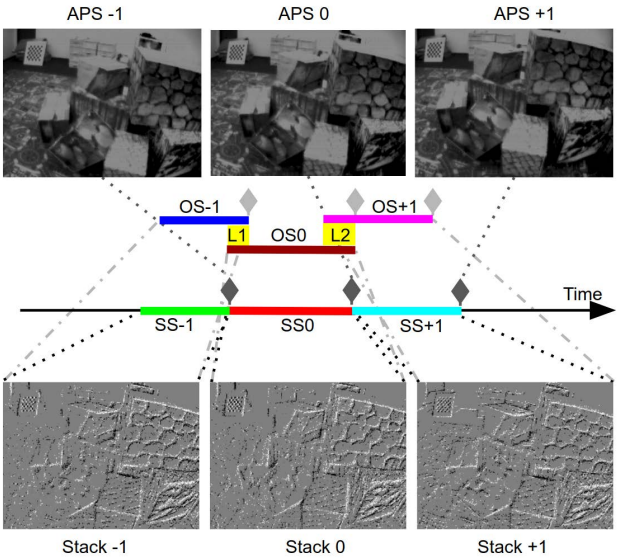

Figure 1. Stacking with separate stacks (SS) or with overlapping Stacks (OS) in a sequence of $3S$. APS frame locations are shown as dark gray diamonds. Light gray diamonds show the location of virtual APS frames which are used in testing and do not respond to an actual APS frame. Central stack is shown as $Stack0$ and the next $(+)$ or previous $(-)$ stacks with regards to the central stack are also shown. The yellow part shows the amount of shared overlapping events ($L1$ and $L2$). Note that the amount of events sets the length of the stack in time which will not necessarily be the same from one stack or overlapping region to another.

2. Design Parameters for $SRNet$

We illustrate the detailed design of our main super resolution network, the $SRNet$, in Fig. 2. The text in each box indicates layer type, number of filters, kernel size, stride and padding respectively (*e.g.*, Conv 64/3/1/1). The projection-wise setting of the recurrent residual modules follows the well-known iterative procedure for super-resolving multiple LR features called back-projection [3]. We adopt the idea to design our $SRNet$; more specifically $RNet$-B performs back-projection from RE_{m+n} to $State_n$ for producing the residual $RNetB(e_n)$.

3. The Synthetic Dataset

3.1. Background

We create a dataset using the event camera simulator (ESIM) [6] for high quality GT as many real world datasets have following issues, making the evaluation less reliable.

Imperfect APS frame as groundtruth (GT). The event camera needs movements of the scene or the camera to produce outputs but rapid movements create motion blur on the intensity image. In addition, the dynamic range of an event camera and an intensity camera are much different which one device might sense parts of the scene that the other device does not. The combination of these factors makes real-world sensing devices prone to errors when used as the training source.

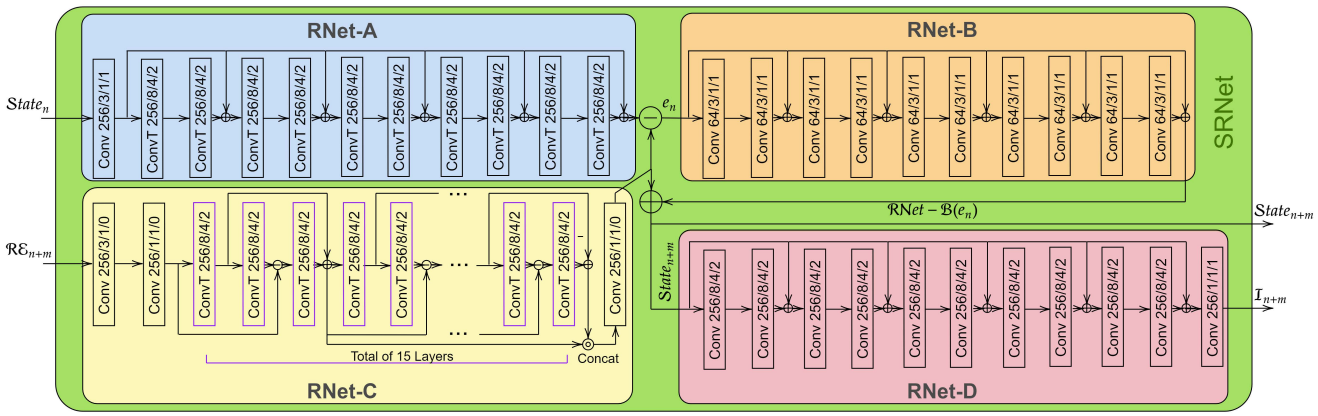

Figure 2. SRNet in detail components. Colors following Fig. 4 of main paper.

Lack of high resolution GT. When working with the same resolution intensity and event device for creating low resolution (LR) event and high resolution (HR) intensity pairs, one might think of resizing the event data to a smaller value. This might work on the training set but when generalizing to the test set that does not have such resized inputs such as the original events of the event camera, the outputs will have much lower quality. The reason behind it is that subsampling algorithms leave unwanted traces on the event stack. This artifacts might seem negligible, but in a learning based solution, it leads to learning erroneous parameters. In our experiments, subsampling the events resulted in a drop of almost $2dB$ in terms of PSNR. This is a crucial step to reach higher quality outputs for cross evaluating on other datasets which we set as a goal.

As a remedy, we utilize a pair of synthetic cameras with different resolutions. We set the LR (event) camera has 128×128 pixels resolution and the HR intensity camera has 256×256 or 512×512 pixels based on the upscale factor ($2\times$ or $4\times$), both sharing the same camera center. To have exactly the same filed of view in both cameras without further warping requirements, the focal length is multiplied to the desired upscale factor when moving from the LR event camera to the HR GT intensity camera.

3.2. Dataset Detail

We created our dataset using $1,000$ different images from the Microsoft COCO 2017 unlabeled images [4] placed on a planar surface while moving the cameras in 6-DoF on top using random trajectories and created almost $120K$ sequences of stacks. Different cameras can have different threshold values, therefore we randomly set the positive and negative threshold independently for each sequence to prevent the network from adapting to this parameter therefore being versatile to the input source all following the implementation details of [7]. Although we train our network only with the simulated dataset, we can fully transfer to real-world scenes without any fine tuning in a complete blind dataset transfer setting.

4. Additional Qualitative Results

Comparison to the State of The Arts. We present more results on real-world and simulated sequences in Fig. 3, 4 and 5.

Results on dataset [7]. Furthermore, we used the new dataset in [7] with includes challenging sequences with high dynamic range and in high-speed scenarios. We showcase a sample in the high speed scenario of popping a water balloon over time in Fig. 14. Our method is able to reconstruct super-resolved details from the background scene and the fast moving foreground objects.

5. Failure Mode Anlaysis

Since the largest number of stacks in a sequence we use was 7 (in $S7$), we are not able to recover the farther events over the 7 stacks due to limited GPU resources. Therefore, our algorithm may miss some background detail when fast foreground moving objects fire large amount of events that make the stacking exceed the 7 stacks. Fig 13 demonstrates a sample condition shown in a sequential manner over time.

Furthermore, if the events in a stream are noisy or dead pixels exist our method will create blurry artifacts in the presentation of those events. Parts of the stream used in Fig. 8 suffer from blurry artifacts. The final reconstruction artifacts are attributed to the lack of events when the camera movement is parallel to the scene structure, therefore events will not fire as shown in Fig. 9. This artifact is often found in many reconstruction methods based on pure events.

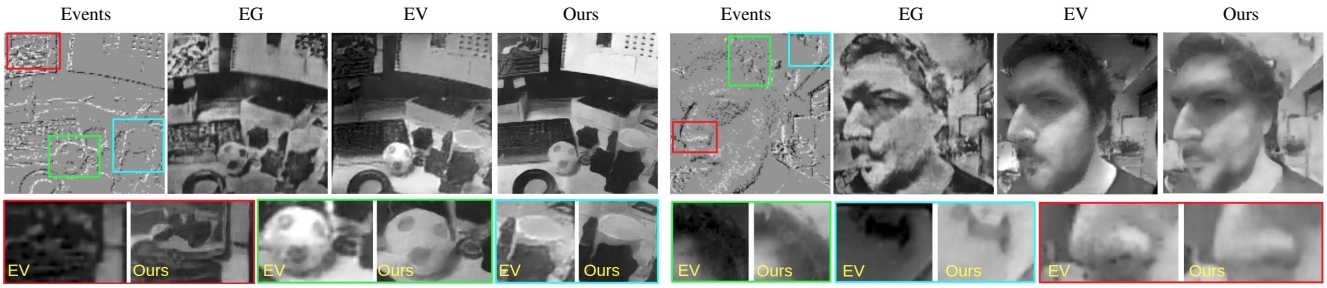

Figure 3. Additional comparison between EV, EG and our results on sequences from [1] (In addition to the Fig. 5 of main manuscript).

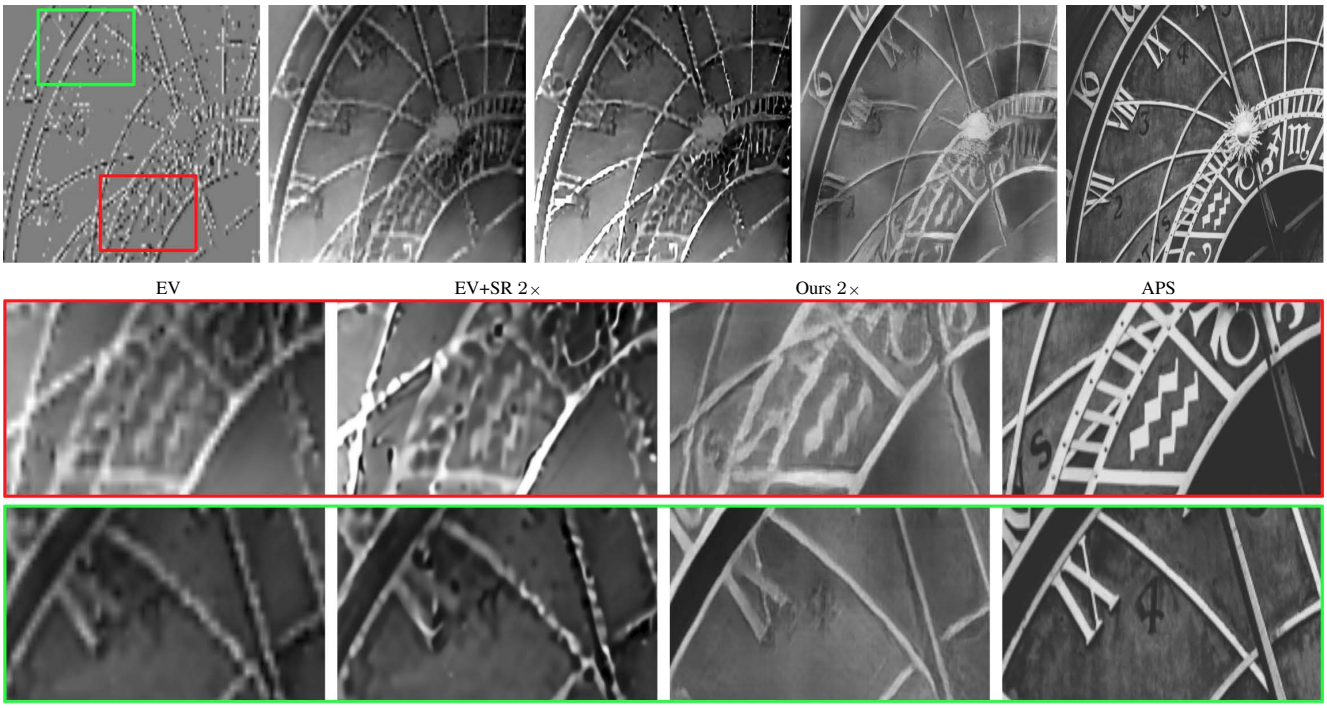

Figure 4. Additional comparison between direct event to SR intensity (ours) and event to image to SR intensity in a hierarchical manner (EV+MISR) on simulated sequences. (In addition to the Fig. 5 of main manuscript)

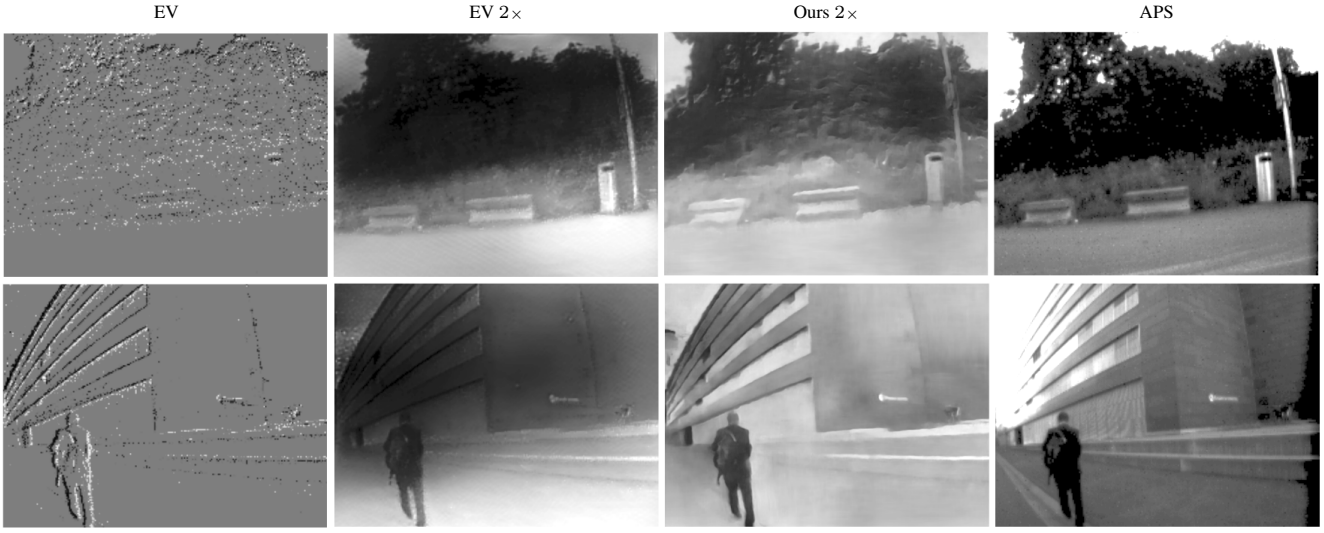

Figure 5. Intensity reconstruction in the presence of background noise from far away objects. (In addition to the Fig. 7 of main manuscript)

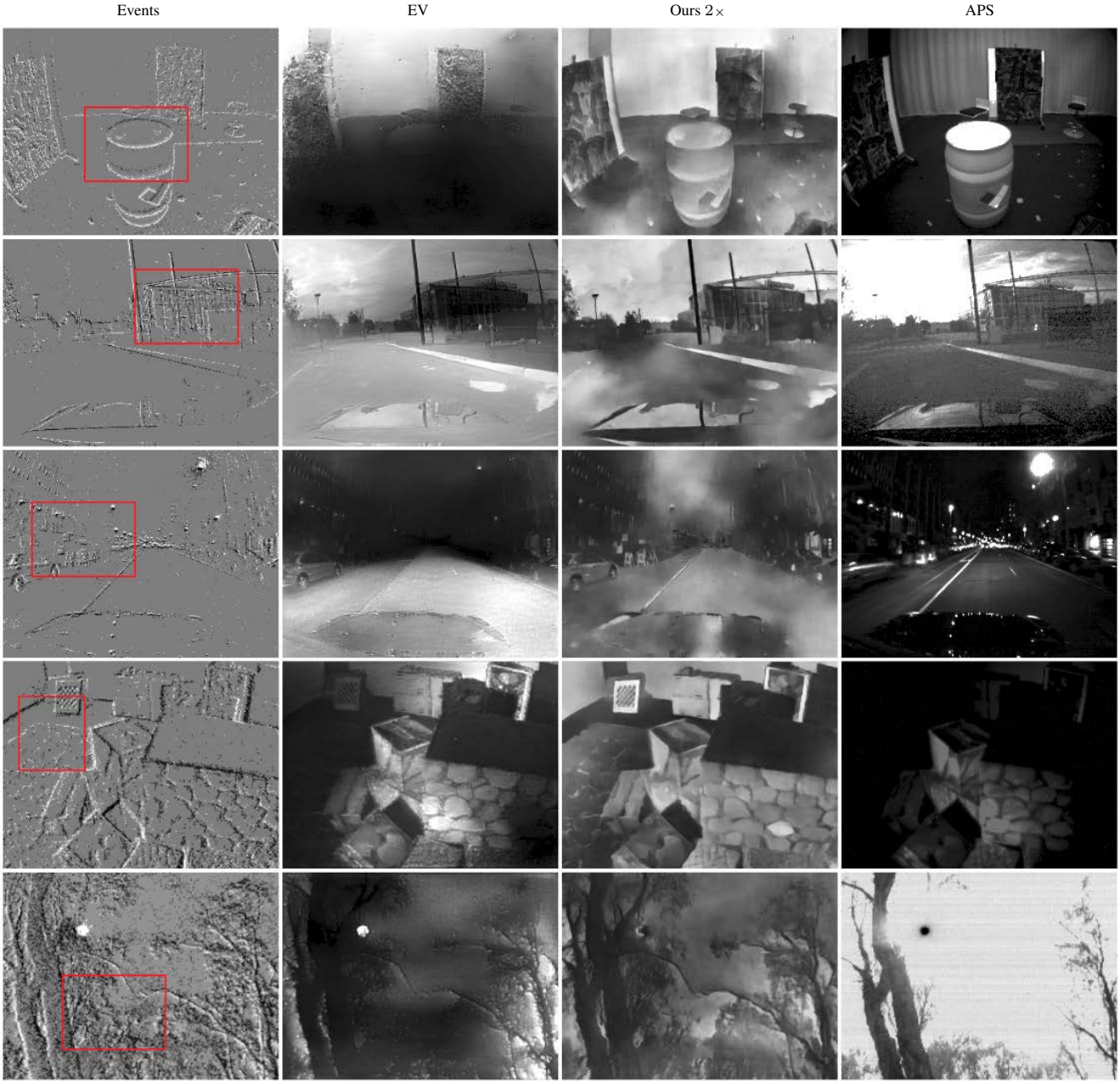

Figure 6. Expressing the robustness of our intensity image reconstruction in challenging scenes. When testing on diverse indoor and outdoor scenes with different lighting conditions and extreme HDR scenarios [9, 10, 5], our method synthesizes more details while producing less artifacts in comparison to EV and the APS. Please zoom in and compare the suggested regions.

6. Details of The Extensions with More Results

6.1. Complimentary

The Complementary extension uses the available APS frame and the events together to make a higher resolution intensity image by fusing the best from both sources. The training process, described in the main manuscript, is shown for seven stacks in a sequence ($S7$) in the green section of Fig. 7. The central stack is highlighted in the middle (SBN_3). The complimentary training also follows the same process but instead of events as a central stack it has the low resolution version of the GT. Each previous or next stack will be fused with the LR GT (APS) and fed to the network. At inference, each event frame will add further HR details to the LR APS frame creating a super-resolved high quality output. Further results are shown in Fig. 10 and Fig. 11.

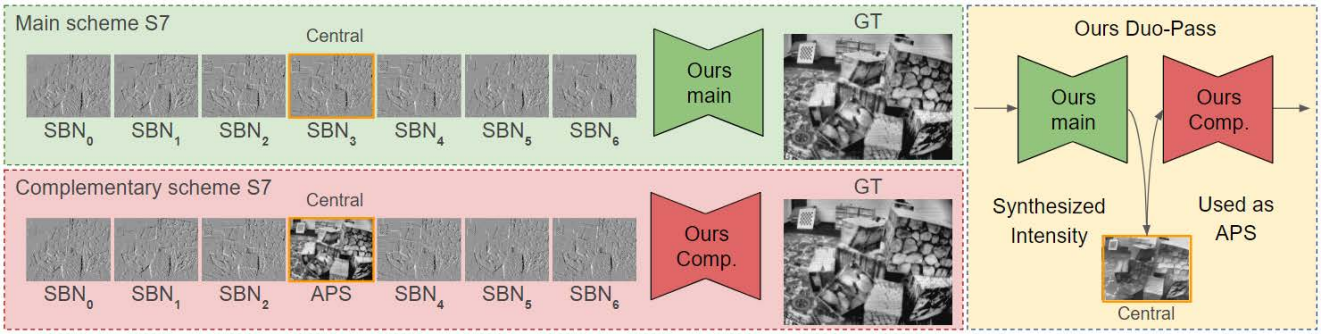

Figure 7. Main and complimentary (Comp.) scheme with $S7$ stacks in a sequence. The central stack is highlighted in the middle and all other stacks will be compared to this stack fore optical flow creation. By putting the main network's output from pure events as a LR input (central stack) for the Comp. network we can have the Duo-Pass network which can add more details to the original intensity image.

Figure 8. Foggy edges in the presence of noisy events.

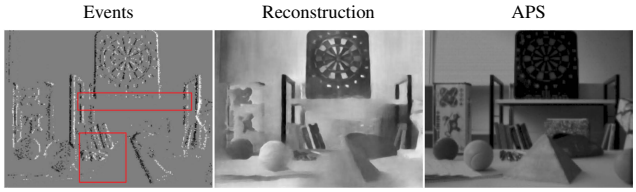

Figure 9. Blur artifact due to lack of events.

There are two downsides of the Complementary approach; (1) we can only reach up to the frame rate of the APS since we use the APS frames, (2) if the LR APS is noisy and blurry, this artifacts will be propagated to the output image.

6.2. Duo-Pass

To avoid the downsides of the Complementary but obtain better quality output, we utilize the output of our method as the LR images in the complimentary extension, we can solve the noise and blur propagation and remove the frame-rate limitation. As shown in Fig. 7, we just place the central stack (or frame) of the complementary method with the synthesised intensity image of our main method. We call this method a Duo-Pass and compare with Complementary and our original method in Fig. 10.

7. Additional Analysis on the Effect of Number of Events Per Stack

The number of events in each stack affects the output reconstruction as shown in Fig. 12. When the number of event are around $5,000$ events for image sizes of 240×180, the output is generally in a reasonable quality. However, adding much more events creates shadow-like outputs or blurred regions. Having much less events results in faded regions due to lack of information. Depending on the scene complexity, more or less events will be required for the best quality result.

To prevent overridden events in cases that the shapes on bottom last row, we stop adding events to the stack if a specific pixel gets overwritten more than 50 times and continue with the next stack in the sequence. This is the general process while hand-tuning this number might get better results for specific cases.

These examples further show that the APS frame is not a good reference for comparing the reconstruction of events in terms of low dynamic range, motion blur and locations where events exist and there is no intensity details corresponding to it (*e.g.*, in Fig 1 of the main manuscript under the table in the 3rd row) or locations where image details exist but no events have fired (tape and paper detailed areas around the shapes in the last row).

8. Additional Analysis on the Effect of using Optical Flow by $FNet$

Stacking events by definition causes loss of temporal relations among events. To recover that loss, we utilize FNet in our design by employing optical flow by following recent MISR techniques for inter-relating images over a sequences [8, 2]. In order to show the usefulness of optical flow on our intensity reconstruction, we ablate its effect by removing it and summarize the results in Table 1. The base network is design for $4\times$ scale and $3S$ stacks with ℓ_1 norm only as the optimization criterion. As shown in the table, without $FNet$ the performances are noticeably decreased in all metrics.

9. Computational Complexity in Time

The average run-time to super resolve an image with the scaling factor of ($2\times$, $4\times$) from the input

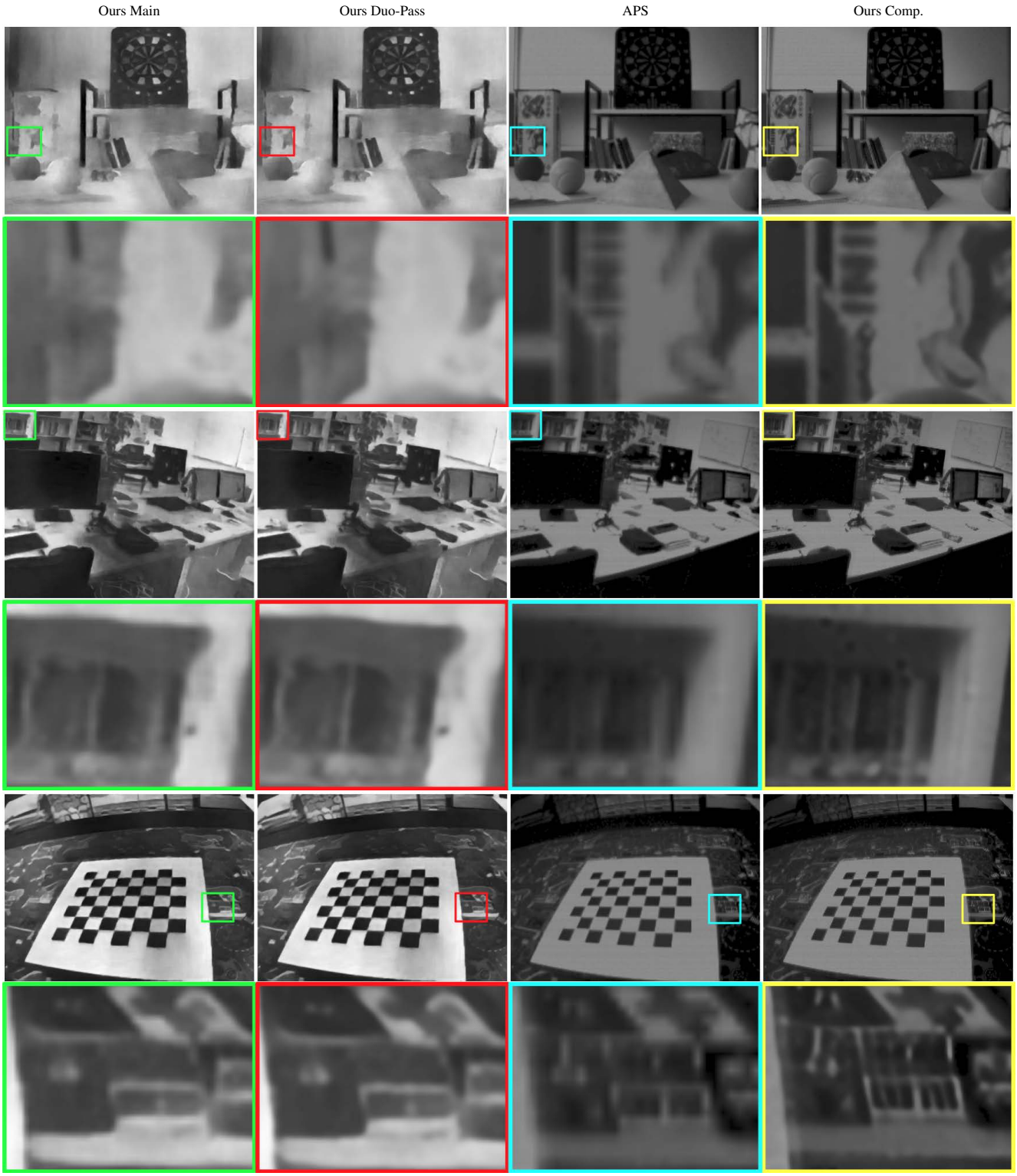

Figure 10. More results of our main and extension methods of double passing (Duo-pass) and complementary processing (Comp.) on real-world dataset [5]. Regions in the colored boxes are zoomed 20× for comparison. APS frames that where very dark are histogram equalized for visualization only. High quality outputs can be achieved when complementing APS frames and events.

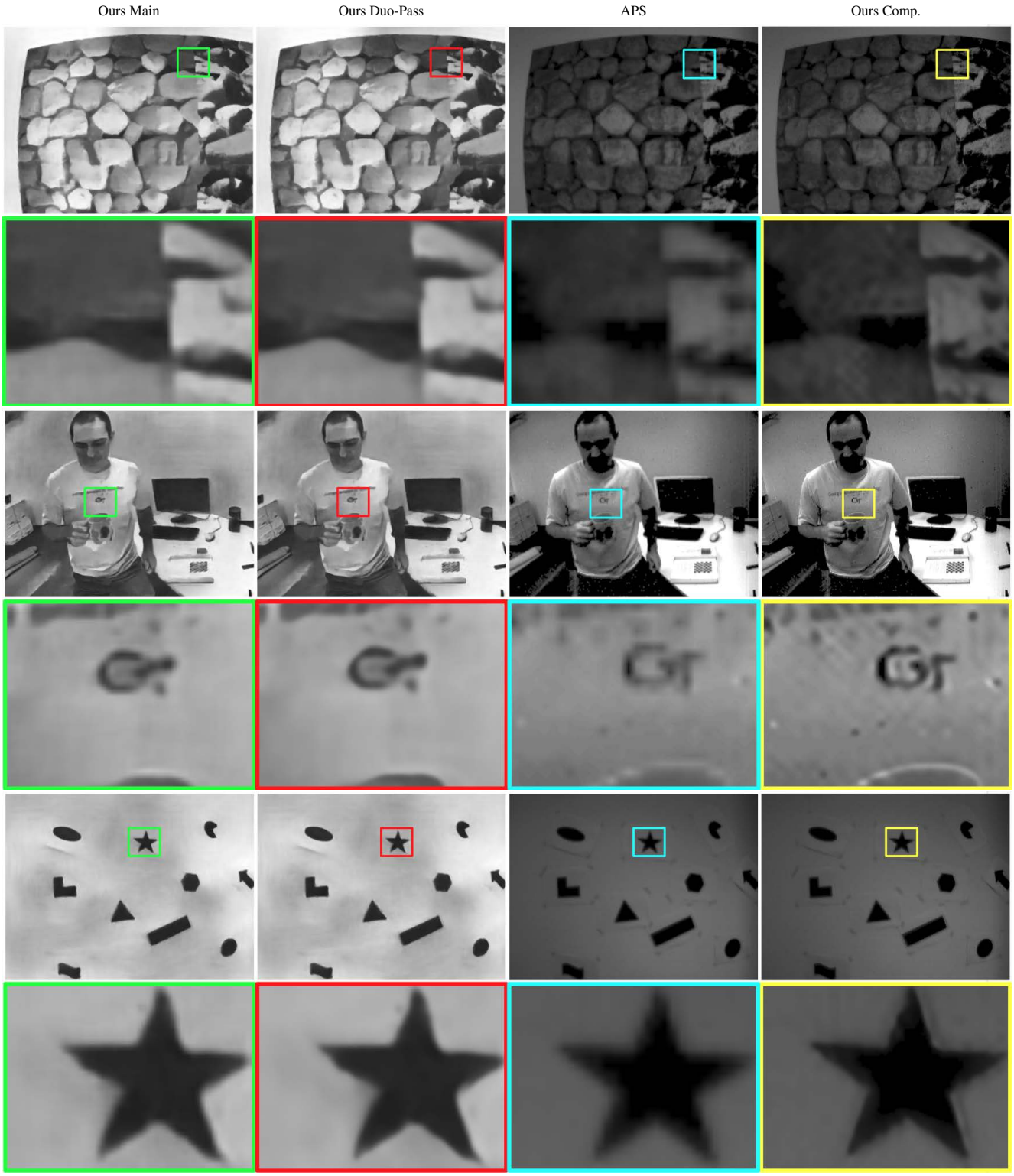

Figure 11. More results of our main and extension methods of double passing (Duo-pass) and complementary processing (Comp.) on real-world dataset [5]. Regions in the colored boxes are zoomed 20× for comparison. APS frames that where very dark are histogram equalized for visualization only. High quality outputs can be achieved when complementing APS frames and events.

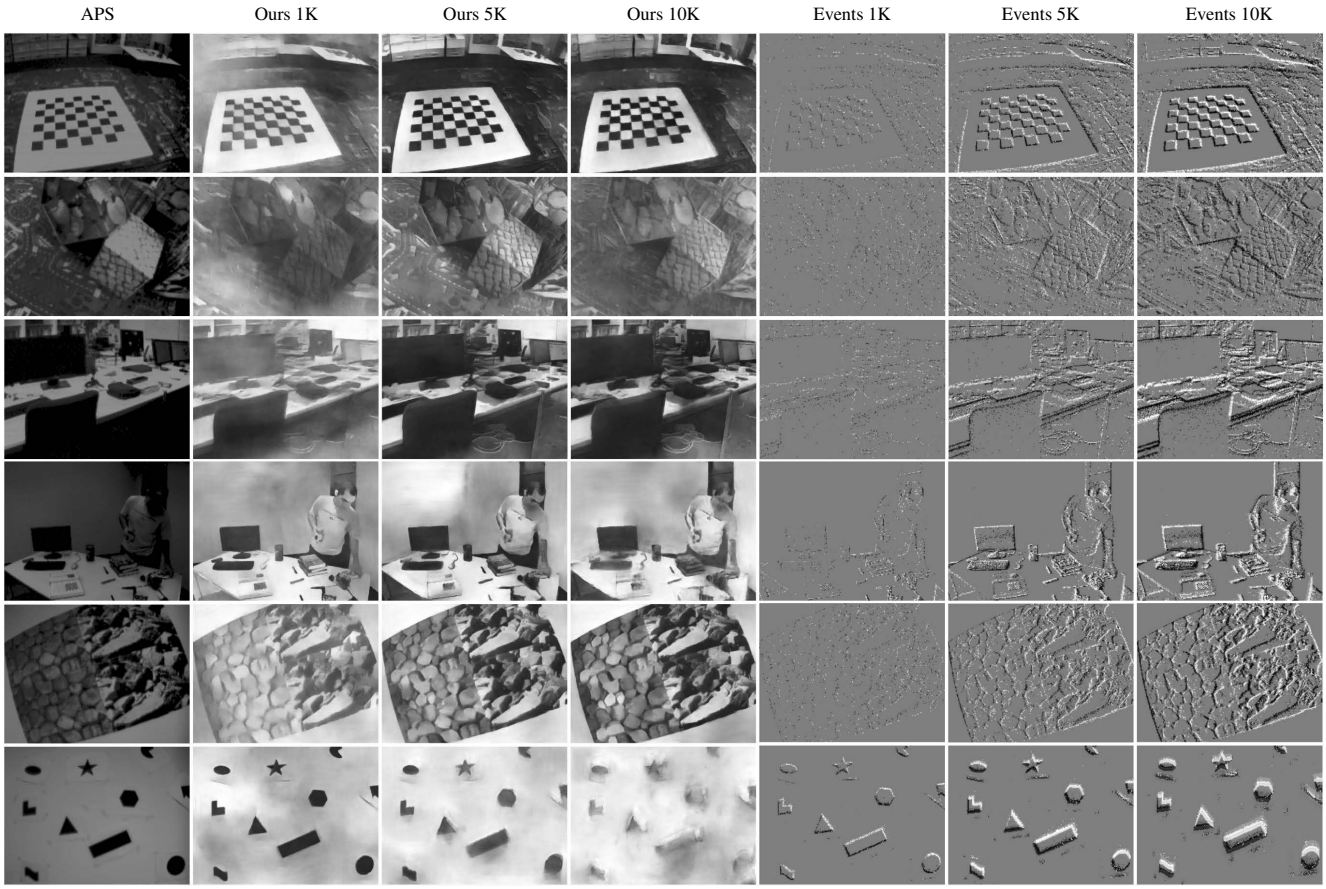

Figure 12. Effect of the number of events on reconstruction quality. APS frame shown is for reference.

Table 1. Ablating the existence of FNet. Adding FNet to ℓ_1 improves all metrics. In the main paper all experiments included FNet and all ablations where performed using $4\times$ scale and 3 stacks ($3S$).

Similarity	PSNR (↑)	SSIM (↑)	MSE (↓)	LPIPS (↓)
without FNet	14.97	0.505	0.036	0.499
with FNet	**15.33**	**0.517**	**0.034**	**0.485**

event stacks with the dimension of $180\times240\times3$ holding 5,000 events per stack on a single Titan-Xp GPU, is: $\{(3S, 2\times),(3S, 4\times),(7S, 2\times),(7S, 4\times)\}\rightarrow\{18.5,\ 19.4,\ 250.8,\ 450.9\}$ (ms) where $3S$ and $7S$ refer to the number of stacks (3 and 7, respectively) in each sequence.

References

[1] Patrick Bardow, Andrew J Davison, and Stefan Leutenegger. Simultaneous optical flow and intensity estimation from an event camera. In *IEEE CVPR*, pages 884–892, 2016. 3

[2] Muhammad Haris, Gregory Shakhnarovich, and Norimichi Ukita. Recurrent back-projection network for video super-resolution. In *IEEE CVPR*, pages 3897–3906, 2019. 5

[3] Michal Irani and Shmuel Peleg. Super resolution from image sequences. In *[1990] Proceedings. ICPR*, volume 2, pages 115–120. IEEE, 1990. 1

[4] Tsung-Yi Lin, Michael Maire, Serge Belongie, James Hays, Pietro Perona, Deva Ramanan, Piotr Dollár, and C Lawrence Zitnick. Microsoft coco: Common objects in context. In *IEEE ECCV*, pages 740–755, 2014. 2

[5] Elias Mueggler, Henri Rebecq, Guillermo Gallego, Tobi Delbruck, and Davide Scaramuzza. The event-camera dataset and simulator: Event-based data for pose estimation, visual odometry, and slam. *The International Journal of Robotics Research*, 36(2):142–149, 2017. 4, 6, 7

[6] Henri Rebecq, Daniel Gehrig, and Davide Scaramuzza. Esim: an open event camera simulator. In *Conference on Robot Learning*, pages 969–982, 2018. 1

[7] Henri Rebecq, René Ranftl, Vladlen Koltun, and Davide Scaramuzza. High speed and high dynamic range video with an event camera. *IEEE T-PAMI*, 2019. 2

[8] Mehdi SM Sajjadi, Raviteja Vemulapalli, and Matthew Brown. Frame-recurrent video super-resolution. In *IEEE CVPR*, pages 6626–6634, 2018. 5

[9] Cedric Scheerlinck, Nick Barnes, and Robert Mahony. Continuous-time intensity estimation using event cameras. In *ACCV*, pages 308–324. Springer, 2018. 4

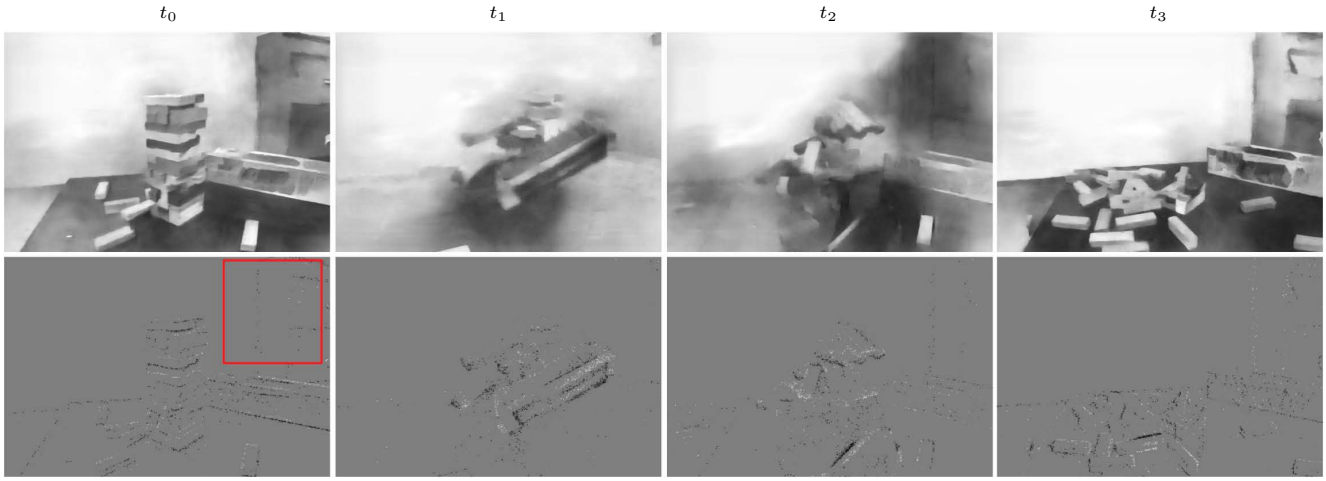

Figure 13. Forgetting background details with SBN in rapid object movement.

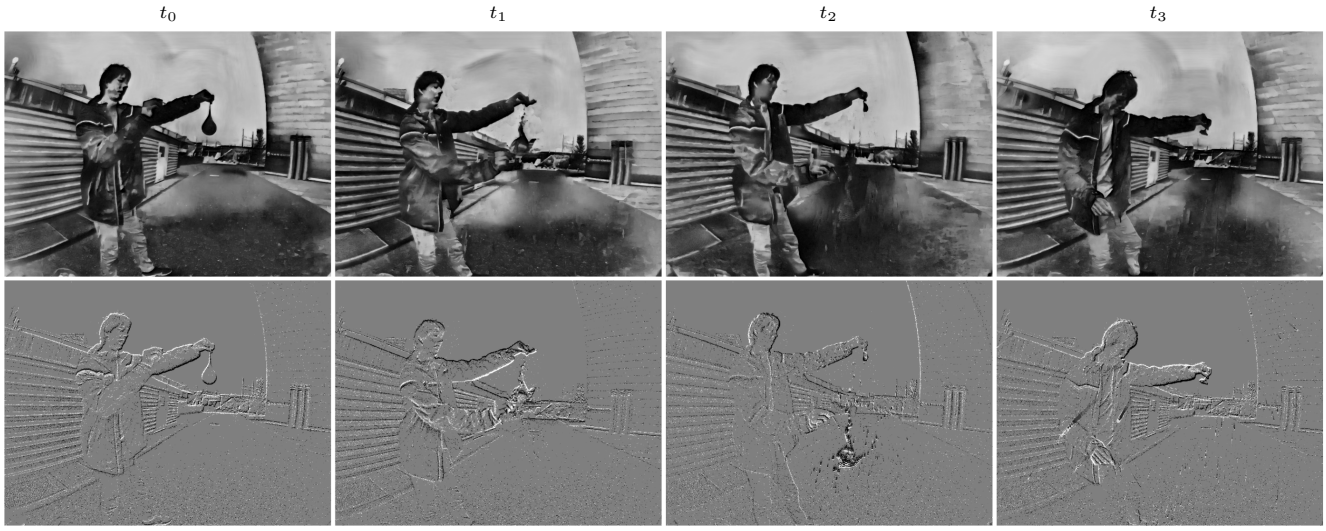

Figure 14. A challenging high-speed scenario of popping a water balloon over time (t_0 to t_3). The intensity details are available in SR dimensions. The background is well reconstructed and the fast moving foreground has been also reconstructed.

[10] Alex Zihao Zhu, Dinesh Thakur, Tolga Özaslan, Bernd Pfrommer, Vijay Kumar, and Kostas Daniilidis. The multi-vehicle stereo event camera dataset: An event camera dataset for 3d perception. *IEEE RA-L*, 3(3):2032–2039, 2018. 4