Supplementary Material for "HDR Video Reconstruction: A Coarse-to-fine Network and A Real-world Benchmark Dataset"

Guanying Chen[1,2] Chaofeng Chen[1] Shi Guo[2,3] Zhetong Liang[2,3]
Kwan-Yee K. Wong[1] Lei Zhang[2,3]

[1]Department of Computer Science, The University of Hong Kong [2]DAMO Academy, Alibaba Group
[3]Department of Computing, The Hong Kong Polytechnic University

Contents

1. More Details for the Proposed Method

1.1. Mathematical Notations Used in the Paper

Table S1 summarizes the mathematical notations used in this paper.

Table S1. Mathematical notations used in this paper.

$\tilde{L}_i$	Original input LDR image with an arbitrary CRF at frame i
L_i	The LDR image with a gamma curve CRF at frame i
H_i	The HDR image at frame i
T_i	Tonemapped HDR image at frame i by a μ-law function
t_i	Exposure time at frame i
$h(L_i, t_i)$	A function converts a LDR image L_i with exposure time t_i to linear radiance domain: $h(L_i, t_i) = L_i^\gamma / t_i$
I_i	Result of converting a LDR image at frame i to linear radiance domain: $I_i = h(L_i, t_i)$
$g_j(I_i)$	A function converts a linear radiance domain image I_i to LDR domain with exposure t_j: $g_j(I_i) = \text{clip}[(I_i t_j)^{1/\gamma}]$

1.2. Network Details of the CoarseNet

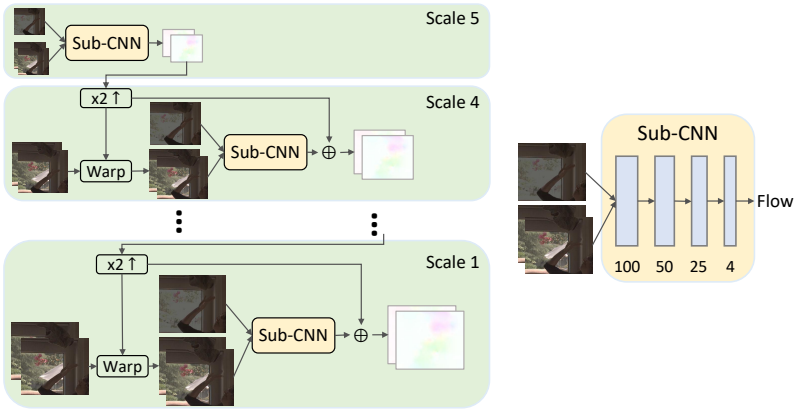


Figure S1. Architecture of the flow network.

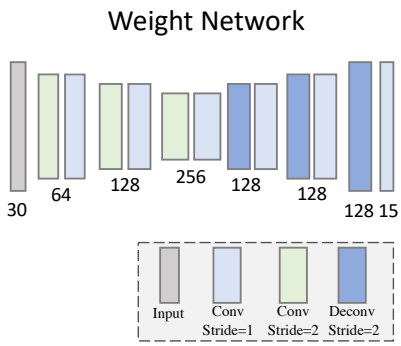


Figure S2. Architecture of the weight network.

Optical flow for image alignment Since the reference frame L_i and neighboring frames (L_{i-1}, L_{i+1}) have different exposures, we adjust the exposure of the reference frame to be the same as the neighboring frame. The reference LDR image is first converted to the linear radiance domain using its exposure t_i:

$$I_i = h(L_i, t_i) = L_i^\gamma / t_i. \tag{1}$$

It is then converted to the LDR domain using the neighboring exposure t_{i+1} as $g_{i+1}(I_i) = \text{clip}[(I_i t_{i+1})^{1/\gamma}]$, where the clip function clips the values to the range of $[0, 1]$.

Traditional flow estimation method takes two images as input and estimates a flow map. However, in our problem, the center frame has a different exposure as the neighboring frames, such that the adjusted reference frame $g_{i+1}(I_i)$ often contains missing contents or noise. We therefore take three consecutive images, *i.e.*, $\{L_{i-1}, g_{i+1}(I_i), L_{i+1}\}$, as input and estimate two flow maps $\{F_{i,i-1}, F_{i,i+1}\}$) as in [6]. Two neighboring frames can then be aligned to the reference frame $(\hat{L}_{i-1,i}, \hat{L}_{i+1,i})$ using backward warping with bilinear sampling [4].

Pixel-blending for HDR reconstruction The HDR image can be computed as a weighted average of the pixels in the aligned images [1, 5]. Note that the two original neighboring frames are also taken into account for pixel blending, as it is reported to be helpful for reducing artifacts in the background regions [6].

Specifically, the input image number for weight network is 5, *i.e.*, $\{L_{i-1}, \hat{L}_{i-1,i}, L_i, \hat{L}_{i+1,i}, L_{i+1}\}$. We provide these five images as input in both the LDR and linear radiance domain, resulting in a stack of 10 images as inputs. The network predicts

five per-pixel weighted maps, *i.e.*, $\{\omega_k | k = 1, \ldots, 5\}$. The coarse HDR at frame i can then be reconstructed as the weighted average of five input images in the linear radiance domain:

$$H_i^c = \frac{\omega_1 I_{i-1} + \omega_2 \hat{I}_{i-1,i} + \omega_3 I_i + \omega_4 \hat{I}_{i+1,i} + \omega_5 I_{i+1}}{\sum_{k=1}^{5} \omega_k}. \tag{2}$$

Similar to [6], we adopt an encoder-decoder architecture to estimate the blending weights.

Flow network Following [6], We adopted a modified version of spatial pyramid network (SPyNet) [9] for optical flow estimation. Our flow network estimated optical flow in 5 different scales in a coarse-to-fine manner (see Fig. S1). The kernel size of the convolutional layers in the flow network was 5×5 and ReLU activation was used.

Weight network Similar to [6], we adopted an encoder-decoder network with three downsampling and three upsampling layers for estimating the blending weights, but with smaller channel numbers in the convolutional layers (see Fig. S2). The kernel size of the convolutional layers in the weight network was 3×3 and Leaky ReLU activation was used. Compared with Kalantari *et al.*'s weight network that contained 8 million parameters [6], ours only contained 2.5 million parameters.

1.3. Network Details of the RefineNet

Figure S3 shows the architecture of RefineNet. The output channel number of the convolution layers was 64, and Leaky ReLU activation was used.

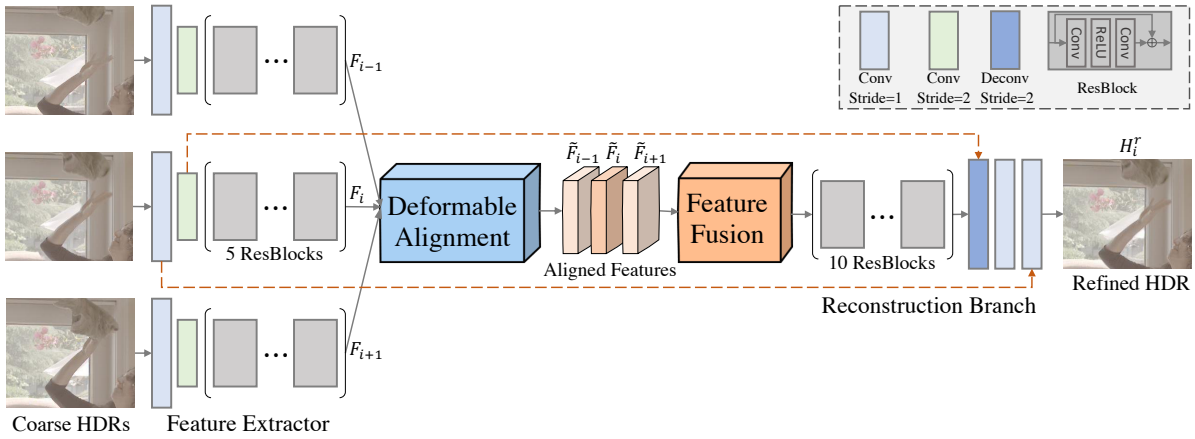


Figure S3. Architecture of the RefineNet.

1.4. Extension to Handle Three Exposures

We have illustrated our method for handling videos captured with *two* alternating exposures in the paper. Here we describe how to adapt our method for handling the case of *three* exposures. Figure S4 compares the overall model architectures for two-exposure and three-exposure videos.

Review of the two-exposure model For sequences captured with two alternating exposures (*e.g.*, {EV-3, EV+3, EV-3, ...}), the CoarseNet takes *three* frames $\{L_{i-1}, L_i, L_{i+1}\}$ as input and estimates the HDR image for the center frame. Given *five* consecutive LDR frames $\{L_i | i = i-2, \ldots, i+2\}$, the CoarseNet can sequentially reconstruct the coarse HDR images for the middle three frames (*i.e.*, H_{i-1}^c, H_i^c, and H_{i+1}^c). The RefineNet then takes these three coarse HDR images as input to produce better HDR reconstruction for the reference frame (*i.e.*, H_i^r), as shown in Fig. S4 (the left part).

CoarseNet for sequences with three-exposure Following [6], for sequences with three alternating exposures (*e.g.*, {EV-2, EV+0, EV+2, EV-2, EV+0, ...}), the CoarseNet takes $five$ frames $\{L_{i-2}, L_{i-1}, L_i, L_{i+1}, L_{i+2}\}$ as input and estimates the HDR image for the center frame. Specifically, the flow network takes $\{L_{i-2}, g_{i+1}(I_i), L_{i+1}\}$[1] and $\{L_{i-1}, g_{i+2}(I_i), L_{i+2}\}$

[1]Remember that I_i is the linear radiance domain image of L_i, which can be computed as $I_i = h(L_i, t_i) = L_i^\gamma / t_i$.

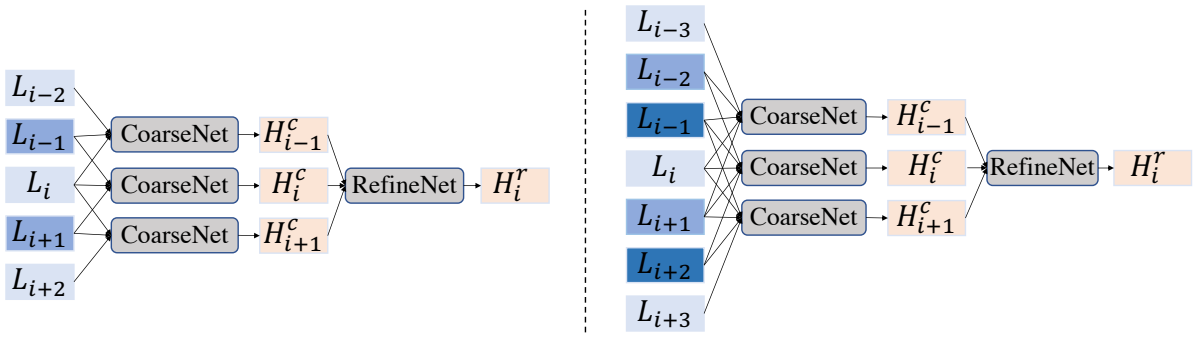
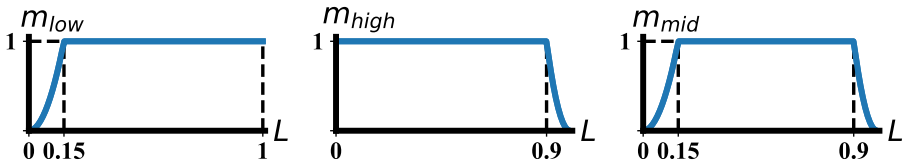


Figure S4. Overview of the models for sequences with two (left) and three (right) exposures. Our method takes 5 (or 7) images as input to produce the HDR image for the center frame for two-exposure (or three-exposure) video, respectively.

as input (totally six images) to estimate four flow maps. The four neighboring frames can then be aligned to the reference frame as $\{\hat{L}_{i-2,i}, \hat{L}_{i-1,i}, \hat{L}_{i+1,i}, \hat{L}_{i+2,i}\}$. The aligned images (4 images) and the original input images (5 images) in both the LDR and linear radiance domain are used as the input (54 channels) for the weight network to estimate 9 weight maps. The coarse HDR image for the reference frame can then be reconstructed as the weighted average of 9 input images in the linear radiance domain. Note that the overall architectures of flow network and weight network are the same for sequences with two and three exposures. The only difference is the channel numbers of the input and output layers.

RefineNet for sequences with three-exposure Similarly to the case of two exposures, given *seven* LDR frames $\{L_i | i = i-3, \ldots, i+3\}$, the CoarseNet can sequentially reconstruct the coarse HDR images for the middle three frames (*i.e.*, H_{i-1}^c, H_i^c, and H_{i+1}^c). The RefineNet then takes these three coarse HDR images as input to produce better HDR reconstruction for the reference frame (*i.e.*, H_i^r). The architecture of RefineNet is the same for two-exposure and three-exposure cases, including the input and output channel numbers. The mask indicating the well-exposed regions for the middle-exposure image can be computed as in Fig. S5 (c).

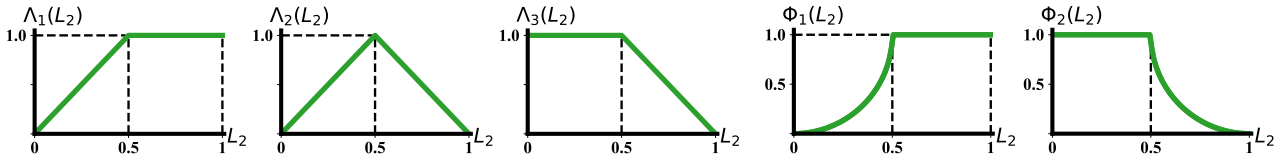


(a) Low exp. (b) High exp. (c) Middle exp.

Figure S5. Weight curves for computing the well-exposed regions for (a) low-, (b) high-, and (c) middle- exposure reference image.

2. More Details for the Captured Real-world Dataset

Figure S6. Samples for the captured static scenes.



(a) Weights for three-exposure scene. (b) Weights for two-exposure scene.

Figure S7. Weighting functions used in real-world HDR generation for the cases of (a) three exposures and (b) two exposures.

We used an off-the-shelf Basler acA4096-30uc camera for capturing videos with two alternating exposures separated by 3 stops, and three alternating exposures separated by 2 stops. The framerate was 26 fps during capturing. Figure S6 shows some samples for the static scenes. For $\mathcal{D}_s^{gt}$ and $\mathcal{D}_d^{gt}$, we inspected and discarded sequences containing unacceptable motions for HDR generation (*e.g.*, motions caused by winds, cars, or human).

Ground-truth HDR Generation For three-exposure images in the order of [low, middle, high] exposures, we used the same triangle weighting functions as in [5]:

$$\omega_1 = \Lambda_1(L_2), \quad \omega_2 = \Lambda_2(L_2), \quad \omega_3 = \Lambda_3(L_2), \tag{3}$$

where Λ_1, Λ_2, and Λ_3 are defined in Fig. S7 (a). The HDR image can be generated as

$$H = \frac{\sum_{i=1}^{3} \omega_i I_i}{\sum_{i=1}^{3} \omega_i}, \tag{4}$$

where I_i is the input image in linear radiance domain.

For two-exposure images in the order of [low, high] exposures, the HDR image can be generated in a similar manner: $H = \sum_{i=1}^{2} \omega_i I_i / \sum_{i=1}^{2} \omega_i$, where $\omega_1 = \Phi_1(I_2)$ and $\omega_2 = \Phi_2(I_2)$ are defined in Fig. S7 (b).

3. More Experimental Results

3.1. More Details for the Synthetic Training Dataset

Since there is no publicly available real video dataset with alternating exposures and their ground-truth HDR, we resort to synthetic data for training. Following [6], we selected 21 HDR videos [3, 8] to synthesize the training dataset. Since the size of the HDR video dataset is limited, we also adopted the high-quality Vimeo-90K dataset [10] to be the source videos. For consecutive frames in an HDR video, we re-exposed them to the LDR domain using alternating exposures and chose the center frame as the reference frame. The exposures are separated by two, or three stops[2], where the low exposure is randomly sampled within a base range.

HDR video dataset Following [6], we selected 21 HDR video clips for synthesizing training dataset:

- 13 videos from [3]: *Bistro 01*, *Bistro 02*, *Bistro 03*, *Cars Close Shot*, *Cars Full Shot*, *Cars Long Shot*, *Fireplace 01*, *Fireplace 02*, *HDR Test Image*, *Showgirl 01*, *Showgirl 02*, *Smith Hammering*, *Smith Welding*.
- 8 videos from [8]: *River*, *Hallway*, *Hallway 2*, *Bridge*, *Bridge 2*, *Students*, *Water*, *Window*.

For consecutive frames, we measured the absolute intensity difference for overlapping patches of size 352×352 with a stride of 176, and saved the patches with top-k largest intensity difference to keep that the patch number for each scene was around 1000. This simple strategy was effective in sampling motion regions. In total, we had $23,926$ samples. We split this dataset into $99:1$ for training and validation.

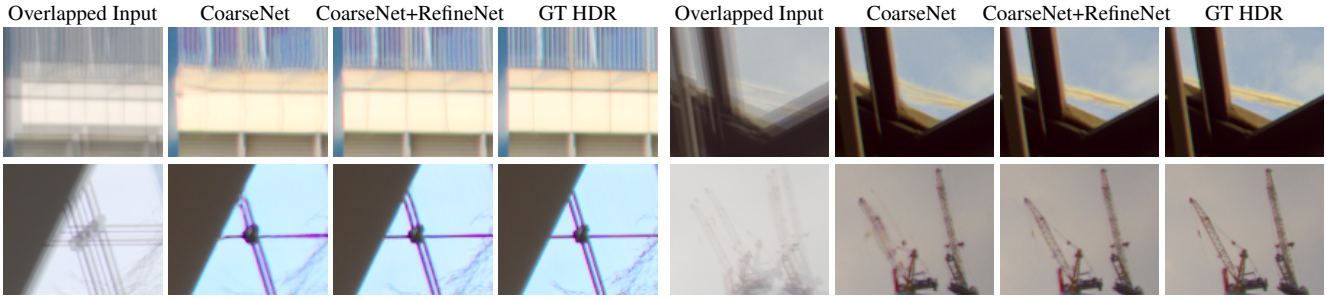
Vimeo-90K video dataset Vimeo-90K dataset [10] consists of $91,701$ preprocessed 7-frame clips. The image resolution of this dataset is 448×256. Following [2], we converted the LDR frames in this dataset to the linear radiance domain by $I_i = \mathcal{F}^{-1}(L_i)$ using randomly sampled parametric camera curves in the form of $\mathcal{F}(x) = (1+\sigma)x^n/(x^n+\sigma)$, where $n \sim \mathcal{N}(0.65, 0.1), \sigma \sim \mathcal{N}(0.6, 0.1)$.

[2]Here, a stop can be considered as a doubling of the exposure time

3.2. More Results for Ablation Study

Comparison between our full model and CoarseNet We first provide more results to verify the effectiveness of the second stage in our method (*i.e.*, RefineNet). Figure S8 compares the visual results of our full model (*i.e.*, CoarseNet + RefineNet) and the CoarseNet on the introduced real-world dataset.

We can see that the CoarseNet, which performs alignment and HDR fusion in the images space, produces results with artifacts (*e.g.*, structure distortion and ghosting) for large motion and over-exposed regions. The RefineNet, performs alignment and fusion in the feature space, can effectively remove the artifacts from the estimated coarse HDRs.



(a) Results on *static scenes* augmented with random global motion.

(b) Results on *dynamic scenes with GT*.

Figure S8. Visual comparison between our full model and CoarseNet on the introduced real-world dataset. For results on each dataset, row 1 is for two-exposure scenes, and row 2 is for three-exposure scenes.

Comparison between our full model and RefineNet† To further validate the necessity of the proposed two-stage architecture, we compared our full model with a single-stage RefineNet†, which directly takes the LDR frames as input and performs alignment and fusion in the feature space to produce the HDR image. We found that RefineNet† cannot produce temporally consistent results for reference frames captured with different exposures (especially for over- and under-exposed regions). Here, we use two examples to illustrate this problem. Please refer to our supplementary video for better visualization.

Figure S9 shows an example for over-exposed regions. We can see that RefineNet† produces clear details for the *low-exposure* image, but produces distorted and blury details for the *high-exposure* image (see the green arrows), resulting in the flickering effect in the reconstructed video. This is because RefineNet† predicts the HDR image by decoding the aligned and fused features. However, it is challenging for a decoder to generate consistent and clear details for well-exposed and over-exposed regions. In contrast, our full model first performs fusion in the image space by blending pixels of the input images to largely remove the color difference caused by different exposures, leading to more consistent and stable results.

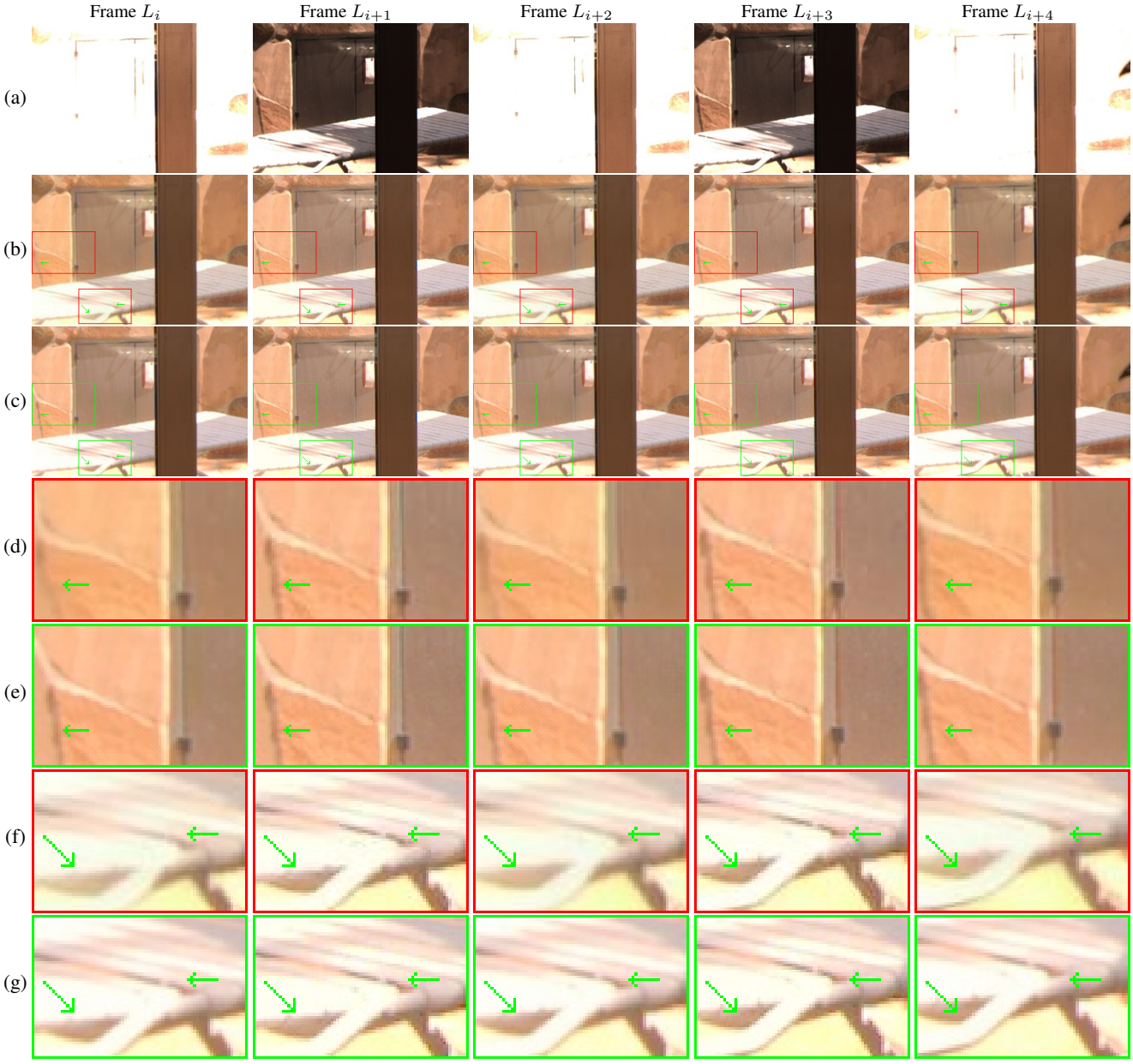


Figure S9. Comparison between our full model and the single-stage RefineNet† on a bright scene. (a) Input sequences with two alternating exposures. (b), (d) and (f) are results of RefineNet†. (c), (e) and (g) are results of our full model.

Figure S10 shows an example for the under-exposed regions. We can see that RefineNet† produces clear details for the *high-exposure* frame, but produces noisy reconstruction for the *low-exposure* frame, causing the flickering effect in the reconstructed sequence. Similar to our previous explanation, it is challenging for a decoder to generate consistent and clear details for reference frames with severe noise and without noise. In contrast, our first stage can largely remove the noise by blending pixel values of the input images, such that our second stage can produce results with better temporal consistency.

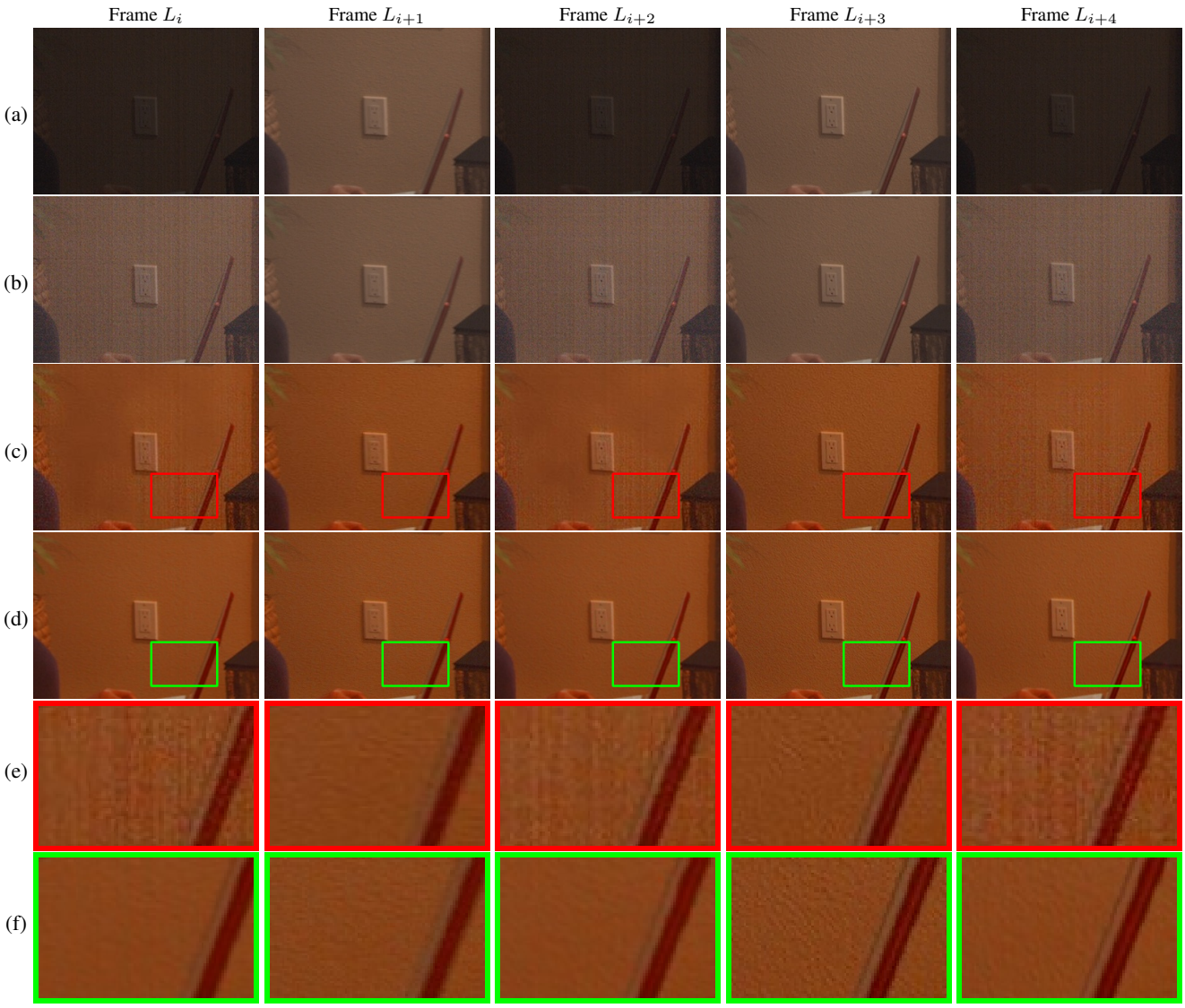


Figure S10. Comparison between our full model and the single-stage RefineNet† on a dark scene. (a) Input sequences with two alternating exposures. (b) The low-exposure images are adjusted to have the same exposure as the high-exposure images. (c) and (e) are results of RefineNet†. (d) and (f) are results of our full model.

3.3. More Comparisons with Previous Methods

We provide more visual comparisons between our method, Kalantari13 [7], Yan19 [11], and Kalantari19 [6].

More results on *static scenes with GT* Figure S11 shows the result on *static scenes* augmented with random global motion. We can see that our method clearly outperforms previous methods.

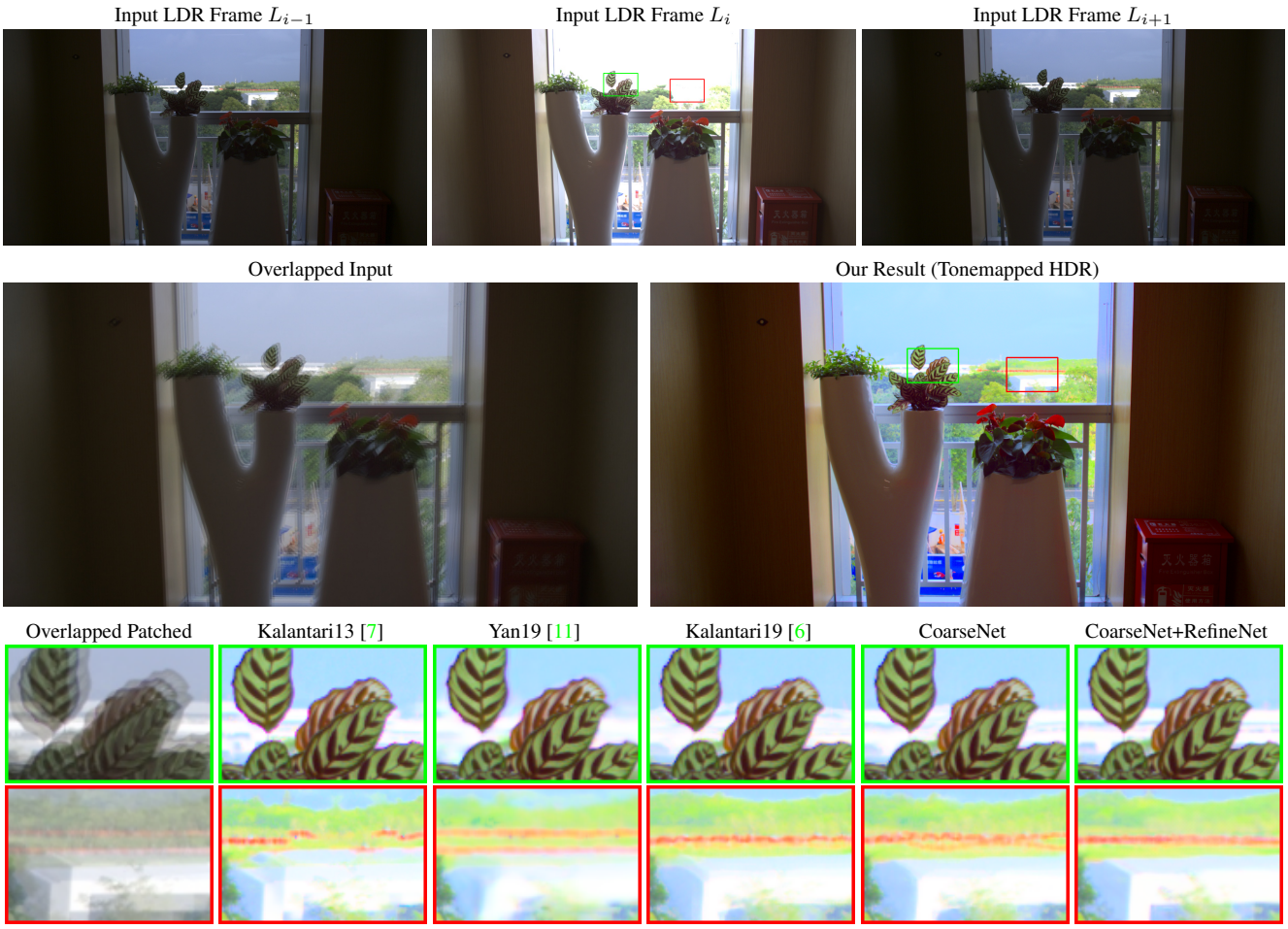


Figure S11. Visual results on *static scenes* augmented with random global motion (two-exposure scene).

More results on *dynamic scenes with GT* Figure S12 shows the result on our *dynamic scenes with GT*.

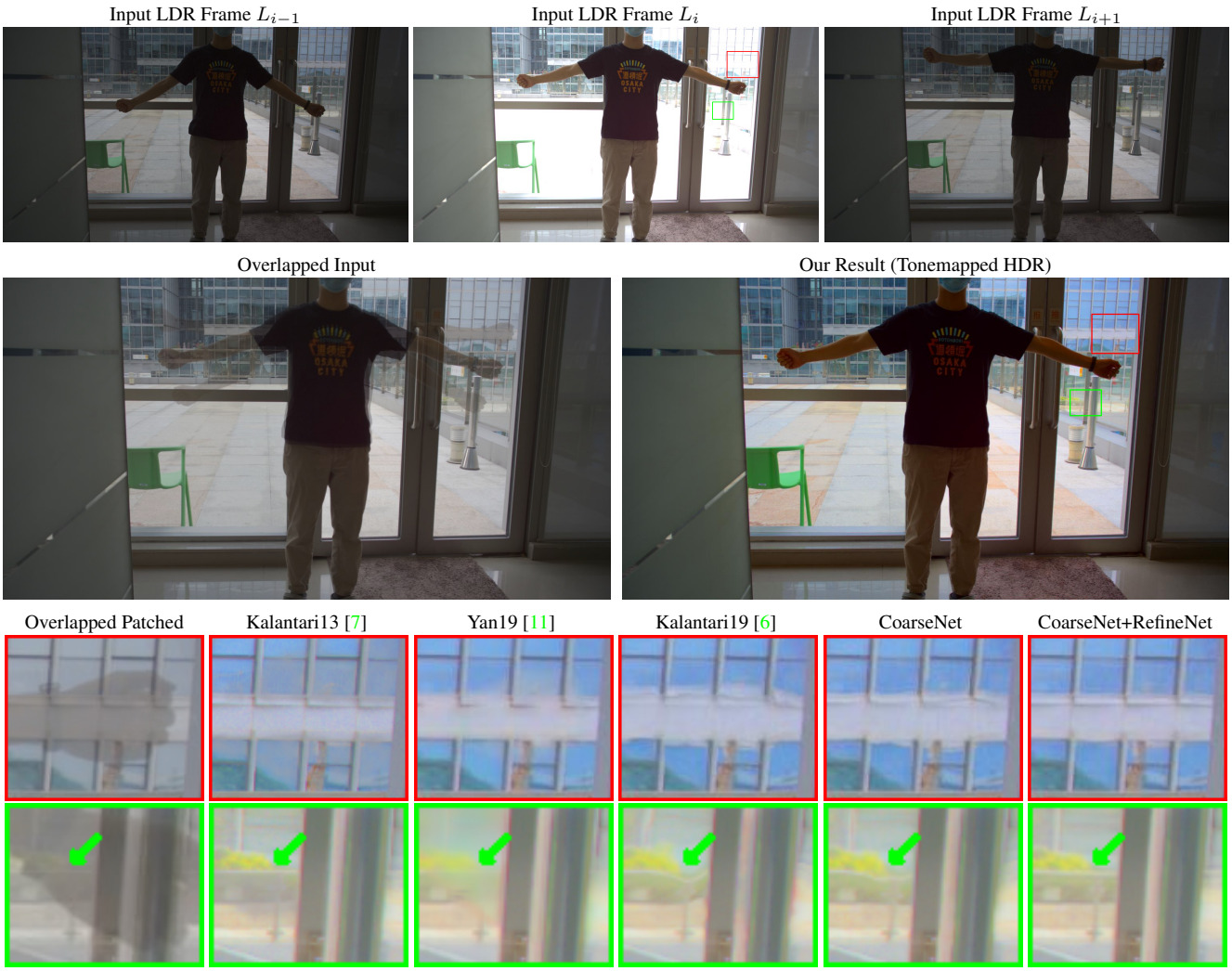
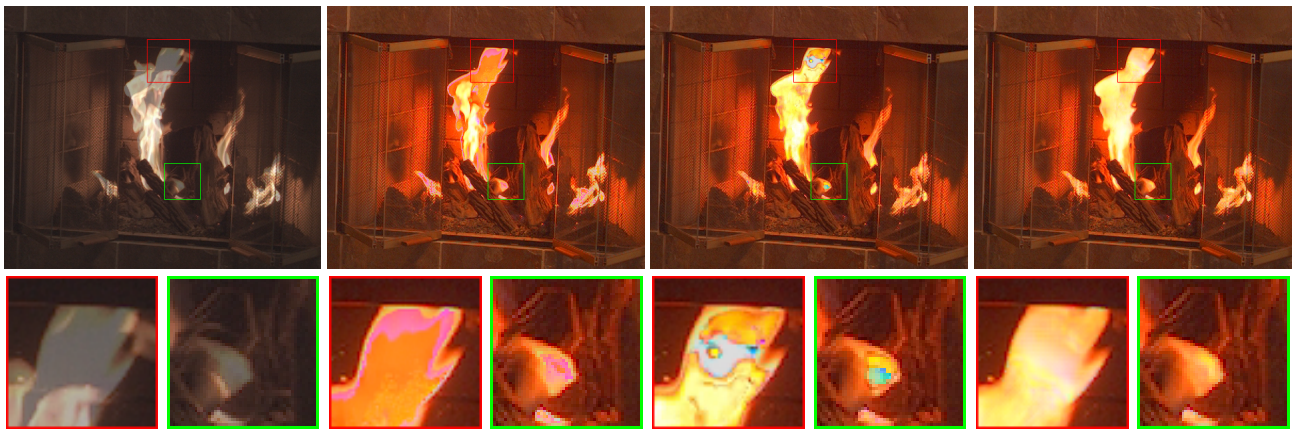


Figure S12. Visual results on *dynamic scenes with GT* (two-exposure scene).

More Results on Kalantari13 Dataset Figure S13 and Figure S14 compare the results of different methods on *Kalantari13* dataset. We can see that our method can effectively reconstruct ghost-free HDR images from sequences captured with two-exposure and three-exposure.

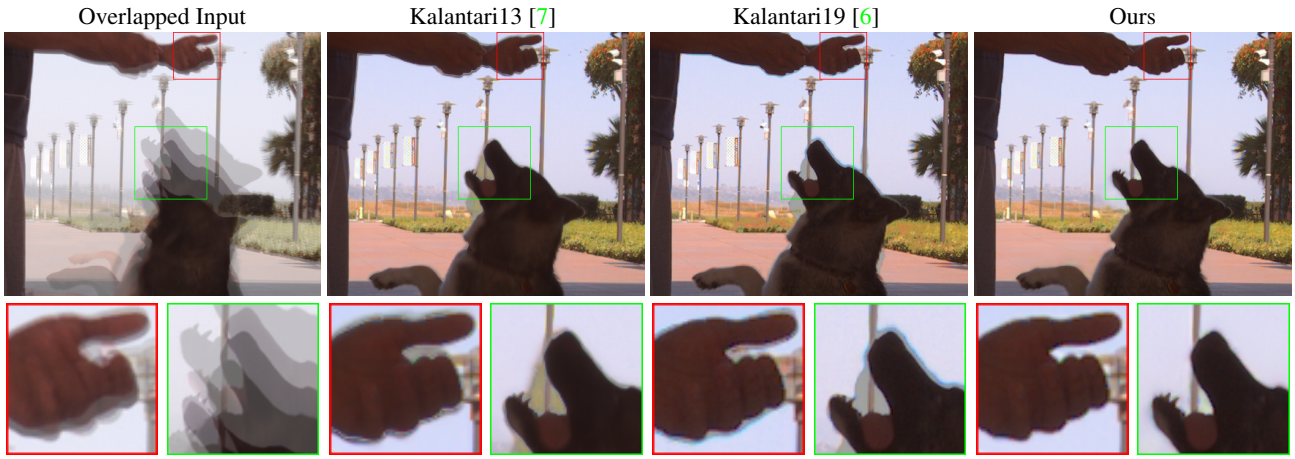
(a) Results on NINJA 2EXP.



(a) Results on FIRE 2EXP.

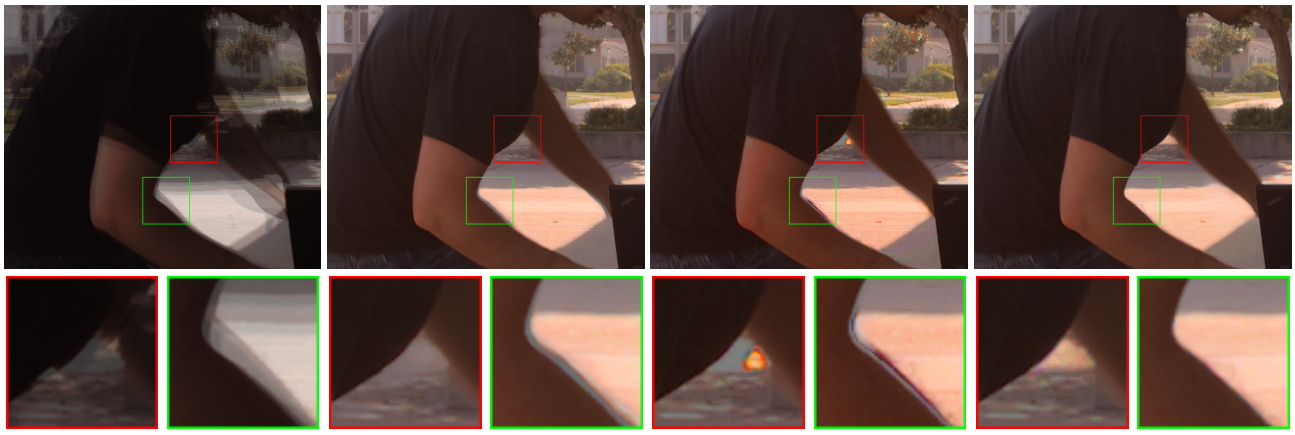
(a) Results on THROWING TOWEL 2EXP.

Figure S13. Visual comparisons on scenes from *Kalantari13* dataset (two-exposure scenes).



(a) Results on DOG 3EXP.

(a) Results on CLEANINING 3EXP.



(a) Results on CHECKING EMAIL 3EXP.

Figure S14. Visual comparisons on scenes from *Kalantari13* dataset (three-exposure scenes).

References

[1] Paul E Debevec and Jitendra Malik. Recovering high dynamic range radiance maps from photographs. In *SIGGRAPH*, 1997. 2

[2] Gabriel Eilertsen, Joel Kronander, Gyorgy Denes, Rafał K Mantiuk, and Jonas Unger. HDR image reconstruction from a single exposure using deep cnns. *TOG*, 2017. 6

[3] Jan Froehlich, Stefan Grandinetti, Bernd Eberhardt, Simon Walter, Andreas Schilling, and Harald Brendel. Creating cinematic wide gamut HDR-video for the evaluation of tone mapping operators and HDR-displays. In *Digital Photography X*, 2014. 6

[4] Max Jaderberg, Karen Simonyan, Andrew Zisserman, et al. Spatial transformer networks. In *NIPS*, 2015. 2

[5] Nima Khademi Kalantari and Ravi Ramamoorthi. Deep high dynamic range imaging of dynamic scenes. *TOG*, 2017. 2, 5

[6] Nima Khademi Kalantari and Ravi Ramamoorthi. Deep HDR video from sequences with alternating exposures. In *Computer Graphics Forum*, 2019. 2, 3, 6, 10, 11, 12, 13

[7] Nima Khademi Kalantari, Eli Shechtman, Connelly Barnes, Soheil Darabi, Dan B Goldman, and Pradeep Sen. Patch-based high dynamic range video. *TOG*, 2013. 10, 11, 12, 13

[8] Joel Kronander, Stefan Gustavson, Gerhard Bonnet, Anders Ynnerman, and Jonas Unger. A unified framework for multi-sensor HDR video reconstruction. *Signal Processing: Image Communication*, 2014. 6

[9] Anurag Ranjan and Michael J Black. Optical flow estimation using a spatial pyramid network. In *CVPR*, 2017. 3

[10] Tianfan Xue, Baian Chen, Jiajun Wu, Donglai Wei, and William T Freeman. Video enhancement with task-oriented flow. *International Journal of Computer Vision*, 2019. 6

[11] Qingsen Yan, Dong Gong, Qinfeng Shi, Anton van den Hengel, Chunhua Shen, Ian Reid, and Yanning Zhang. Attention-guided network for ghost-free high dynamic range imaging. In *CVPR*, 2019. 10, 11