







# Deep High Dynamic Range Imaging with Large Foreground Motions

Shangzhe Wu<sup>1,3</sup>, Jiarui Xu<sup>1</sup>, Yu-Wing Tai<sup>2</sup>, and Chi-Keung Tang<sup>1</sup>

<sup>1</sup> The Hong Kong University of Science and Technology, Kowloon, Hong Kong  
{swuai, jxuat}@connect.ust.hk, cktang@cs.ust.hk

<sup>2</sup> Tencent Youtu, Shanghai, China  
yuwingtai@tencent.com

<sup>3</sup> University of Oxford, Oxford, UK

**Abstract.** This paper proposes the first non-flow-based deep framework for high dynamic range (HDR) imaging of dynamic scenes with **large-scale foreground motions**. In state-of-the-art deep HDR imaging, input images are first aligned using optical flows before merging, which are still error-prone due to occlusion and large motions. In stark contrast to flow-based methods, we formulate HDR imaging as an image translation problem **without optical flows**. Moreover, our simple translation network can automatically hallucinate plausible HDR details in the presence of total occlusion, saturation and under-exposure, which are otherwise almost impossible to recover by conventional optimization approaches. Our framework can also be extended for different reference images. We performed extensive qualitative and quantitative comparisons to show that our approach produces excellent results where color artifacts and geometric distortions are significantly reduced compared to existing state-of-the-art methods, and is robust across various inputs, including images without radiometric calibration.

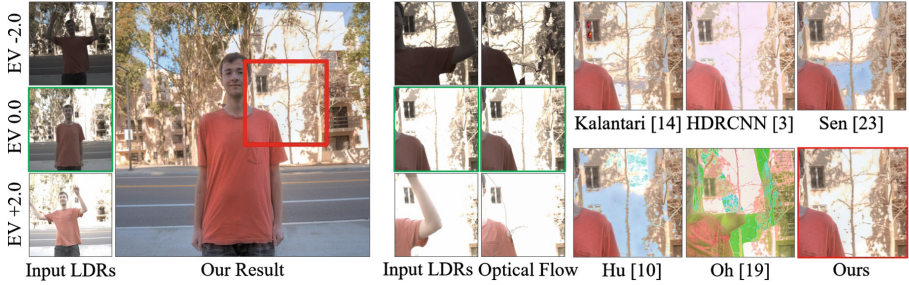
**Keywords:** High dynamic range imaging  
Computational photography

## 1 Introduction

Off-the-shelf digital cameras typically fail to capture the entire dynamic range of a 3D scene. In order to produce high dynamic range (HDR) images, custom captures and special devices have been proposed [8, 24, 25]. Unfortunately, they are usually too heavy and/or too expensive for capturing fleeting moments to cherish, which are typically photographed using cellphone cameras. The other more practical approach is to merge several low dynamic range (LDR) images captured at different exposures. If the LDR images are perfectly aligned, in other words no camera motion or object motion is observed, the merging problem is considered almost solved [1, 17]. However, foreground and background misalignments

---

This work was partially done when Shangzhe Wu was an intern at Tencent Youtu.



**Fig. 1.** Our goal is to produce an HDR image from a stack of LDR images that can be corrupted by large foreground motions, such as images shown on the left. Our resulted HDR image is displayed after tonemapping. On the right, the first two columns show that the optical flow alignment used by Kalantari [14] introduces severe geometric distortions and color artifacts, which are unfortunately preserved in the final HDR results. The last three columns compare the results produced by other state-of-the-art methods and ours where no optical flow alignment is used. Our simple network produces high quality ghost-free HDR image in the presence of large-scale saturation and foreground motions.

are unavoidable in the presence of large-scale foreground motions in addition to small camera motions. While the latter can be resolved to a large extent by homography transformation [26], foreground motions, on the other hand, will make the composition nontrivial. Many existing solutions tackling this issue are prone to introducing artifacts or ghosting in the final HDR image [14, 15, 31], or fail to incorporate misaligned HDR contents by simply rejecting the pixels in misaligned regions as outliers [9, 16, 19], see Fig. 1.

Recent works have been proposed to learn this composition process using deep neural networks [14]. In [14], they first used optical flow to align input LDR images, followed by feeding the aligned LDRs into a convolutional neural network (CNN) to produce the final HDR image. Optical flows are often unreliable, especially for images captured with different exposure levels, which inevitably introduce artifacts and distortions in the presence of large object motions. Although in [14] it was claimed that the network is able to resolve these issues in the merging process, failure cases still exist as shown in Fig. 1, where color artifacts and geometry distortions are quite apparent in the final results.

In contrast, we regard merging multiple exposure shots into an HDR image as an image translation problem, which have been actively studied in recent years. In [11] a powerful solution was proposed to learn a mapping between images in two domains using a Generative Adversarial Network (GAN). Meanwhile, CNNs have been demonstrated to have the ability to learn misalignment [2] and hallucinate missing details [30]. Inspired by these works, we believe that optical flow may be an overkill for HDR imaging. In this paper, we propose a

simple end-to-end network that can learn to translate multiple LDR images into a ghost-free HDR image even in the presence of large foreground motions.

In summary, our method has the following advantages. First, unlike [14], our network is trained end-to-end without optical flow alignment, thus intrinsically avoiding artifacts and distortions caused by erroneous flows. In stark contrast to prevailing flow-based HDR imaging approaches [14], this provides a novel perspective and significant insights for HDR imaging, and is much faster and more practical. Second, our network can hallucinate plausible details that are totally missing or their presence is extremely weak in all LDR inputs. This is particularly desirable when dealing with large foreground motions, because usually some contents are not captured in all LDRs due to saturation and occlusion. Finally, the same framework can be easily extended to more LDR inputs, and possibly with any specified reference image. We perform extensive qualitative and quantitative comparisons, and show that our simple network outperforms the state-of-the-art approaches in HDR synthesis, including both learning based or optimization based methods. We also show that our network is robust across various kinds of input LDRs, including images with different exposure separations and images without correct radiometric calibration.

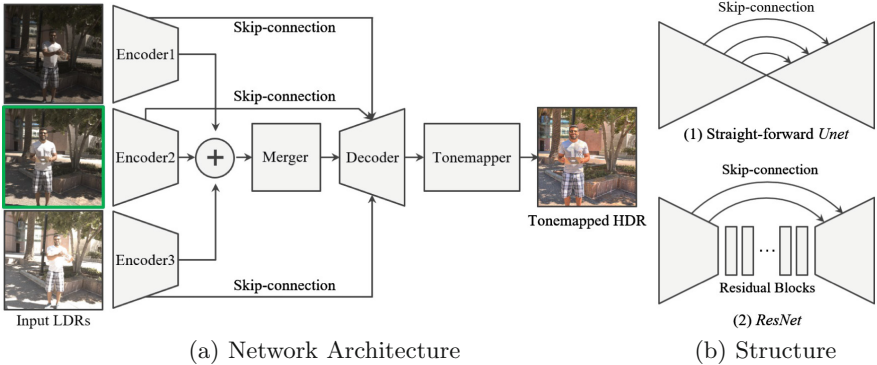
## 2 Related Work

Over the past decades, many research works have been dedicated to the problem of HDR imaging. As mentioned above, one practical solution is to compose an HDR image from a stack of LDR images. Early works such as [1, 17] produce excellent results for static scenes and static cameras.

To deal with camera motions, previous works [12, 15, 26] register the LDR images before merging them into the final HDR image. Since many image registration algorithms depend on the brightness consistence assumptions, the brightness changes are often addressed by mapping the images to another domain, such as luminance domain or gradient domain, before estimating the transformation.

Compared to camera motions, object motions are much harder to handle. A number of methods reject the moving pixels using weightings in the merging process [9, 16]. Another approach is to detect and resolve ghosting after the merging [4, 21]. Such methods simply ignore the misaligned pixels, and fail to fully utilize available contents to generate an HDR image.

There are also more complicated methods [15, 31] that rely on optical flow or its variants to address dense correspondence between image pixels. However, optical flow often results in artifacts and distortions when handling large displacements, introducing extra complication in the merging step. Among the works in this category, [14] produces perhaps the best results, and is highly related to our work. The authors proposed a CNN that learns to merge LDR images aligned using optical flow into the final HDR image. Our method is different from theirs in that we do not use optical flow for alignment, which intrinsically avoids the artifacts and distortions that are present in their results. We provide concrete comparisons in the later sections.



**Fig. 2.** Our framework is composed of three components: encoder, merger and decoder. Different exposure inputs are passed to different encoders, and concatenated before going through the merger and the decoder. We experimented with two structures, *Unet* and *ResNet*. We use skip-connections between the mirrored layers. The output HDR of the decoder is tonemapped before it can be displayed.

Another approach to address the dense correspondence is patch-based system [10, 23]. Although these methods produce excellent results, the running time is much longer, and often fail in the presence of large motions and large saturated regions.

A more recent work [3] attempts to reconstruct a HDR image from one single LDR image using CNN. Although their network can hallucinate details in regions where input LDRs exhibit only very weak response, one intrinsic limitation of their approach is the total reliance on one single input LDR image, which often fails in highly contrastive scenes due to large-scale saturation. Therefore, we intend to explore better solutions to merge HDR contents from multiple LDR images, which can easily be captured in a burst, for instance, using cellphone cameras.

Typically, to produce an HDR image also involves other processing, including radiometric calibration, tone-mapping and dynamic range compression. Our work is focused on the merging process. Besides, there are also more expensive solutions that use special devices to capture a higher dynamic range [8, 24, 25] and directly produce HDR images. For a complete review of the problem, readers may refer to [5].

### 3 Approach

We formulate the problem of HDR imaging as an image translation problem. Similar to [14], given a set of LDR images  $\{I_1, I_2, \dots, I_k\}$ , we define a reference image  $I_r$ . In our experiments, we use three LDRs, and set the middle exposure shot as reference. The same network can be extended to deal with more LDR inputs, and possibly with any specified reference image. We provide results in Sect. 5.3 to substantiate such robustness.

Specifically, our goal is to learn a mapping from a stack of LDR images  $\{I_1, I_2, I_3\}$  to a ghost-free HDR image  $H$  that is aligned with the reference LDR input  $I_r$  (same as  $I_2$ ), and contains the maximum possible HDR contents. These contents are either obtained directly from LDR inputs, or from hallucinations when they are completely missing. We focus on handling large foreground motions, and assume the input LDR images, which are typically taken in a burst, have small background motions.

### 3.1 Network Architecture

We capitalize on a translation network to learn such a mapping. As shown in Fig. 2, our framework is essentially a symmetric encoder-decoder architecture, with two variants, *Unet* and *ResNet*.

*Unet* [22] is a common tool for translation learning. It is essentially an encoder-decoder architecture, with skip-connections that forward the output of the encoder layer (conv) directly to the input of the corresponding decoder layer (deconv) through channel-wise concatenation. In recent image translation works, such as [11], *Unet* has been demonstrated to be powerful in a wide range of tasks. However, unlike [11] where *Unet* was used in an adversarial setting, we may not need a discriminator network in HDR imaging, because the mapping from LDR to HDR is relatively easy to learn, compared to other scenarios in [11], where the two images domains are much more distinct, such as *edge*  $\leftrightarrow$  *photo*.

In addition to simple *Unet*, we also experimented with another structure, *ResNet*, similar to *Image Transformation Networks* proposed in [13], which simply replaces the middle layers with residual blocks [7]. Similar structure is also used in recent translation works [29]. In this paper, we name the this structure *ResNet*, as opposed to the previous one, *Unet*. We compare their performance in later sections.

The overall architecture can be conceptually divided into three components: encoder, merger and decoder. Since we have multiple exposure shots, intuitively we may have separate branches to extract different types of information from different exposure inputs. Instead of duplicating the whole network, which may defer the merging, we separate the first two layers as encoders for each exposure inputs. After extracting the features, the network learns to merge them, mostly in the middle layers, and to decode them into an HDR output, mostly in the last few layers.

### 3.2 Processing Pipeline and Loss Function

Given a stack of LDR images, if they are not in RAW format, we first linearize the images using the estimated inverse of Camera Response Function (CRF) [6], which is often referred to as radiometric calibration. We then apply gamma correction to produce the input to our system.

Although this process is technically important in order to recover the accurate radiance map, in practice, our system could also produce visually plausible approximation without radiometric calibration, such as examples shown in

Fig. 10. This is because the gamma function can be a rough approximation of the CRF.

We denote the set of input LDRs by  $\mathcal{I} = \{I_1, I_2, I_3\}$ , sorted by their exposure biases. We first map them to  $\mathcal{H} = \{H_1, H_2, H_3\}$  in the HDR domain. We use simple gamma encoding for this mapping:

$$H_i = \frac{I_i^\gamma}{t_i}, \gamma > 1 \quad (1)$$

where  $t_i$  is the exposure time of image  $I_i$ . Note that we use  $H$  to denote the target HDR image, and  $H_i$  to denote the LDR inputs mapped to HDR domain. The values of  $I_i$ ,  $H_i$  and  $H$  are bounded between 0 and 1.

We then concatenate  $\mathcal{I}$  and  $\mathcal{H}$  channel-wise into a 6-channel input and feed it directly to the network. This is also suggested in [14]. The LDRs facilitate the detection of misalignments and saturation, while the exposure-adjusted HDRs improve the robustness of the network across LDRs with various exposure levels. Our network  $f$  is thus defined as:

$$\hat{H} = f(\mathcal{I}, \mathcal{H}) \quad (2)$$

where  $\hat{H}$  is the estimated HDR image, and is also bounded between 0 and 1.

Since HDR images are usually displayed after tonemapping, we compute the loss function on the tonemapped HDR images, which is more effective than directly computed in the HDR domain. In [14] the author proposed to use  $\mu$ -law, which is commonly used for range compression in audio processing:

$$\mathcal{T}(H) = \frac{\log(1 + \mu H)}{\log(1 + \mu)} \quad (3)$$

where  $H$  is an HDR image, and  $\mu$  is a parameter controlling the level of compression. We set  $\mu$  to 5000. Although there are other powerful tonemappers, most of them are typically complicated and not fully differentiable, which makes them not suitable for training a neural network.

Finally, our loss function is defined as:

$$\mathcal{L}_{U_{net}} = \|\mathcal{T}(\hat{H}) - \mathcal{T}(H)\|_2 \quad (4)$$

where  $H$  is the ground truth HDR image.

## 4 Datasets

We used the dataset provided by [14] for training and testing. Although other HDR datasets are available, many of them either do not have ground truth HDR images, or contain only a very limited number of scenes. This dataset contains 89 scenes with ground truth HDR images. As described in [14], for each scene, 3 different exposure shots were taken while object was moving, and another 3 shots were taken while object remained static. The static sets are used to

**Table 1.** Comparison of average running time on the test set under CPU environment.

	Sen [23]	Hu [10]	Kalantari [14]	HDRCNN [3]	Ours <i>Unet</i>	Ours <i>ResNet</i>
Time (s)	261	137	72.1	12.6	11.9	14.7

**Table 2.** Quantitative comparisons of the results on Kalantari’s test set [14]. The first two rows are PSNR/SSIM computed using tonemapped outputs and ground truth, and the following two rows are PSNR/SSIM computed using linear images and ground truth. The last row is HDR-VDP-2 [18] scores. All values are the average across 15 testing images in the original test set.

	Sen [23]	Hu [10]	Kalantari [14]	Ours <i>Unet</i>	Ours <i>ResNet</i>
PSNR-T	40.80	35.79	<b>42.70</b>	40.81	41.65
SSIM-T	0.9808	0.9717	<b>0.9877</b>	0.9844	0.9860
PSNR-L	38.11	30.76	<b>41.22</b>	40.52	40.88
SSIM-L	0.9721	0.9503	0.9845	0.9837	<b>0.9858</b>
HDR-VDP-2	59.38	57.05	63.98	64.88	<b>64.90</b>

produce ground truth HDR with reference to the medium exposure shot. This medium exposure reference shot then replaces the medium exposure shot in the dynamic sets. All images are resized to  $1000 \times 1500$ . Each set consists of LDR images with exposure biases of  $\{-2.0, 0.0, +2.0\}$  or  $\{-3.0, 0.0, +3.0\}$ . We also tested our trained models on Sen’s dataset [23] and Tursun’s dataset [27, 28].

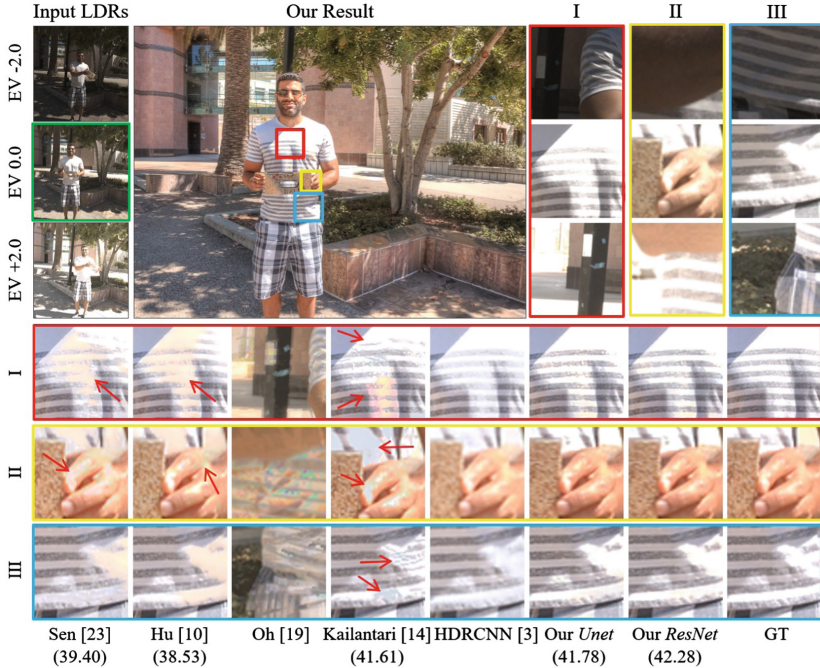
#### 4.1 Data Preparation

To focus on handling foreground motions, we first align the background using simple homography transformation, which does not introduce artifacts and distortions. This makes the learning more effective than directly trained without background alignment. Comparison and discussion are provided in Sect. 5.4.

#### 4.2 Data Augmentation and Patch Generation

The dataset was split into 74 training examples and 15 testing examples by [14]. For the purpose of efficient training, instead of feeding the original full-size image into our model, we crop the images into  $256 \times 256$  patches with a stride of 64, which produces around 19000 patches. We then perform data augmentation (flipping and rotation), further increasing the training data by 8 times.

In fact, a large portion of these patches contain only background regions, and exhibit little foreground motions. To keep the training focused on foreground motions, we detect large motion patches by thresholding the structural similarity between different exposure shots, and replicate these patches in the training set.



**Fig. 3.** Comparison against several state-of-the-art methods. In the upper half of the figure, the left column shows in the input LDRs, the middle is our tonemapped HDR result, and the last three columns show three zoomed-in LDR regions marked in the HDR image. The lower half compares the zoomed-in HDR regions of our results against others. The numbers in brackets at the bottom indicate the PSNR of the tonemapped images. Images are obtained from the Kalantari’s test set [14].

## 5 Experiments and Results

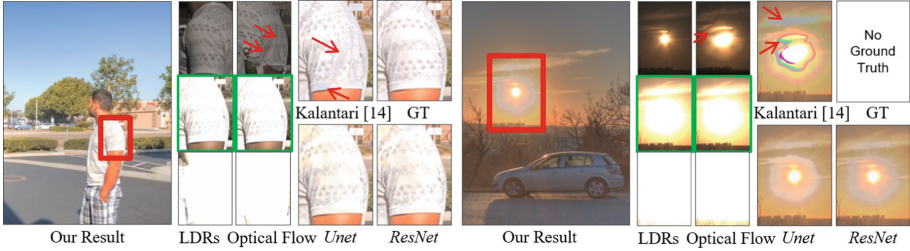
### 5.1 Implementation Details

We first perform radiometric calibration and map the input LDRs to HDR domain. Each of the resulted radiance maps is channel-wise concatenated with the LDR image respectively, and then separately fed into different encoders. After 2 layers, all feature maps are then concatenated channel-wise for merging.

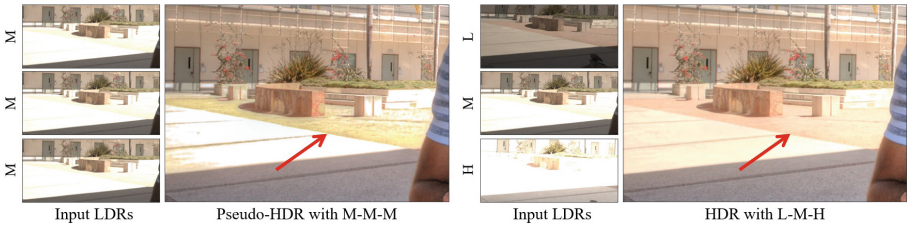
The encoding layers are convolution layers with a stride of 2, while the decoding layers are deconvolution layers kernels with a stride of 1/2. The output of the last deconvolution layer is connected to a flat-convolution layer to produce the final HDR. All layers use  $5 \times 5$  kernels, and are followed by batch normalization (except the first layer and the output layer) and leaky ReLU (encoding layers) or ReLU (decoding layers). The channel numbers are doubled each layer from 64 to 512 during encoding and halved from 512 to 64 during decoding.

For *Unet* structure,  $256 \times 256$  input patches are passed through 8 encoding layers to produce a  $1 \times 1 \times 512$  block, followed by 8 decoding layers plus an





**Fig. 4.** Comparison against flow-based method [14]. Images are obtained from the Kalantari’s dataset [14] and Tursun’s dataset [27,28].



**Fig. 5.** Example of hallucination. The left is generated using only medium exposure shot, and the right is generated using low, medium and high exposure shots. Images are obtained from the Kalantari’s dataset [14].

output layer to produce a  $256 \times 256$  HDR patch. Our *ResNet* is different only in that after 3 encoding layers, the  $32 \times 32 \times 256$  block is passed through 9 residual blocks with  $3 \times 3$  kernels, followed by 3 decoding layers and an output layer.

## 5.2 Running Time

We report running time comparison with other methods in Table 1. Although our network is trained with GPU, other conventional optimization methods are optimized with CPU. For fair comparison, we evaluated all methods under CPU environment, on a PC with i7-4790K (4.0 GHz) and 32 GB RAM. We tested all methods using 3 LDR images of size  $896 \times 1408$  as input. Note that the optical flow alignment used in [14] takes 59.4s on average. When run with GPU (Titan X Pascal), our *Unet* and *ResNet* take 0.225 s and 0.239 s respectively.

## 5.3 Evaluation and Comparison

We perform quantitative and qualitative evaluations, and compare results with the state-of-the-art methods, including two patch-based methods [10,23], motion rejection method [19], the flow-based method with CNN merger [14], and the single image HDR imaging [3]. For all methods, we used the codes provided by the authors. Note that all the HDR images are displayed after tonemapping using *Photomatix* [20], which is different from the tonemapper used in training.

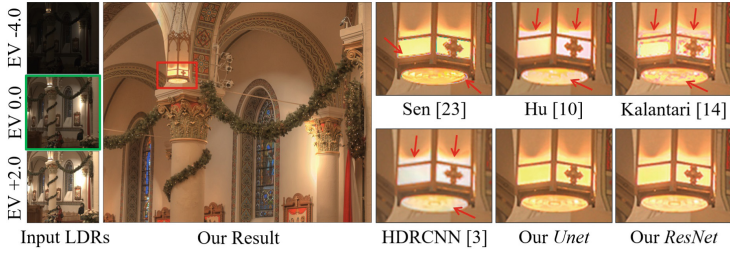


**Fig. 6.** Comparison of hallucinated details. Our network hallucinates the missing trunk texture, while others may fail. Images are obtained from the Kalantari’s dataset [14].

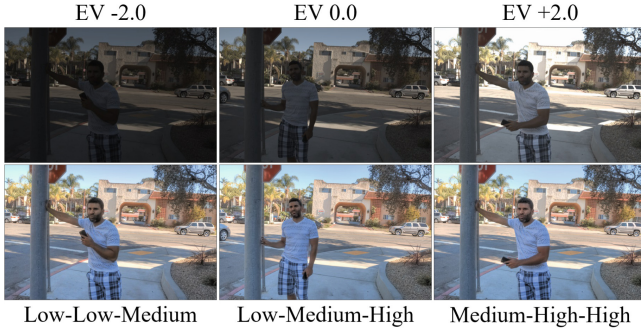
**Quantitative Comparison.** We compute the PSNR and SSIM scores between the generated HDR and the ground truth HDR, both before and after tonemapping using  $\mu$ -law. We also compute the HDR-VDP-2 [18], a metric specifically designed for measuring the visual quality of HDR images. For the two parameters used to compute the HDR-VDP-2 scores, we set the diagonal display size to 24 in., and the viewing distance to 0.5 m. We did not compare with [19] and [3] quantitatively, since the former is optimized for more than 5 LDR inputs and the latter produces unbounded HDR results.

Table 2 shows quantitative comparison of our networks against the state-of-the-art methods. Note that all results are calculated on the Kalantari’s test set [14]. While [14] results in slightly higher PSNR scores, our methods result in comparable SSIM scores and slightly higher HDR-VDP-2 scores. Besides, *ResNet* seems to yield higher scores than *Unet*.

**Qualitative Comparison.** Figure 3 compares the testing results against state-of-the-art methods. In regions with no object motions, all methods produce decent results. However, when large object motion is present in saturated regions, [10, 14, 23] tend to produce unsightly artifacts. Flow-based method [14] also produces geometric distortions. Because Oh’s method [19] uses rank minimization, which generally requires more inputs, it results in ghosting artifacts when applied with 3 inputs. Since HDRCNN [3] estimates the HDR image using only one single reference LDR image, it does not suffer from object motions, but tends to produce less sharp results and fail in large saturated regions, as shown



**Fig. 7.** Comparison of highlight regions. Examples come from the Sen’s dataset [23].



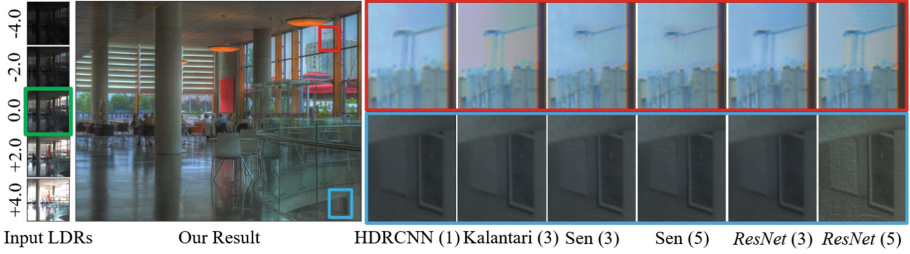
**Fig. 8.** Results with different reference images. The first row shows three LDR inputs, and the second row shows the corresponding HDR results with reference to each input.

in Fig. 1. Our two networks produce comparably good results, free of obvious artifacts and distortions. In general, *ResNet* seems to consistently outperform *Unet*.

**Comparison Against Flow-Based Method.** In addition to Figs. 1 and 3, Fig. 4 illustrates our advantages over Kalantari’s method [14], where optical flow alignment introduces severe distortions and color artifacts. Our method does not rely on erroneous optical flow, which intrinsically avoids such distortions, and is also much more efficient computationally.

**Hallucination.** One important feature of our method is the capability of hallucinating missing details that are nearly impossible to recover using conventional optimization approaches. As shown in Fig. 5, when given only the medium exposure, our network is able to properly hallucinate the grass texture in the saturated regions. When given also two other exposure shots, our network is able to incorporate the additional information such as the ground texture.

In Fig. 6, we examine the effectiveness of hallucination, by comparing our results to others with no hallucination. Hallucination can be very useful in



**Fig. 9.** Results with more input LDRs. The integers in the parentheses indicate the number of LDR images used to generate produce the HDR.

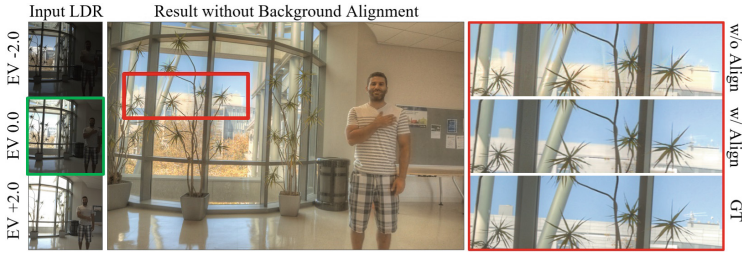


**Fig. 10.** HDR results without radiometric calibration. All examples are novel images taken using cellphones with different CRFs.

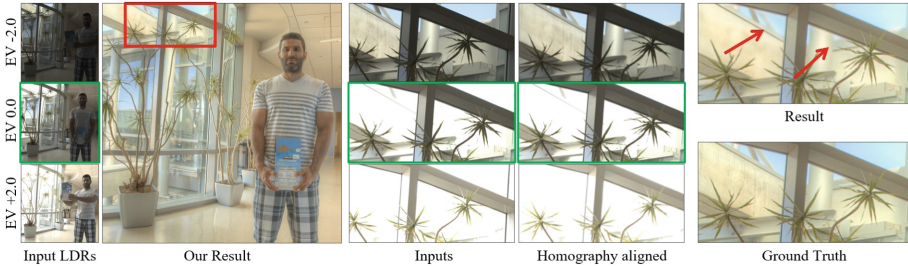
dynamic scenes, since contents in over-exposed or under-exposed regions are often missing in all LDRs due to total occlusions caused by object motions.

**Highlight.** In addition to Fig. 4, where we show that our method outperforms [14] in highlight regions, Fig. 7 compares our highlight details against others. While other methods often fail to recover details in highlight regions and introduce artifacts and distortions, our method generally works well. Specifically, Hu’s method [10] performs poorly in general at highlight regions, and other methods can only partially recover the details. Kalantari’s method [14] tends to introduce evident distortions and color artifacts as shown in Fig. 7.

**Different Reference Image.** Figure 8 illustrates another advantage of our image translation formulation: the flexibility in choosing different reference images. Currently this is achieved by re-arranging the input LDRs. For example, using only low and high exposure shots and feeding them to the network in the order of {Low-Low-Medium} will result in a pseudo-HDR image with reference to the low exposure shot. Technically, this output does not represent the accurate radiance values, but is perceptually compelling and similar to real HDR images.



**Fig. 11.** This example illustrates the effect of background alignment.



**Fig. 12.** Blurry results caused by parallax effects, which cannot be resolved by homography transformation.

Our framework may be extended to directly output multiple HDR images with different reference images, if trained in such a fashion, although we do not have appropriate datasets to corroborate this.

**More Input LDRs.** Our framework can potentially be extended for supporting more than 3 input LDRs. This is useful, because more LDRs capture more contents and improve the robustness. Although we do not have a suitable dataset to fully explore this, we decided to conduct a brief experiment using Sen’s dataset [23]. We used their produced HDR images as ground truth for training, which are yet to be perfect to be used as ground truth, but sufficient for our purpose of testing such extensibility. Using this dataset, we tested our framework using 5 LDR inputs. Figure 9 compares our results with others. Interestingly, while Sen’s [23] results using 5 inputs do not seem to be clearly better than those using 3 inputs, in our results, the details in saturated and under-exposed regions are markedly improved by using more input LDRs.

**Cellphone Example.** We also tested our model on novel cellphone images for proof of practicality, shown in Fig. 10. Our network produces good results in various kinds of settings. The input images were captured using different cellphones with different camera response functions. It is worth noting that when producing

these pseudo-HDR examples, we did not perform radiometric calibration. This again demonstrates the robustness of our network.

#### 5.4 Discussion on Background Alignment

In all our experiments and comparisons, since we are focused on handling large foreground motions, we align the backgrounds of the LDR inputs using homography transformation. Without background alignment, we found that our network tends to produce blurry edges where background is largely misaligned, as shown in Fig. 11. This can be due to the confusion caused by the background motion, which CNN is generally weak at dealing with. However, such issues can be easily resolved using simple homography transformation that almost perfectly aligns the background in most cases. Recall that in practice, the LDR inputs can be captured in a burst within a split second using nowadays handheld devices.

Nevertheless, homography is not always perfect. One particular case where homography may not produce perfect alignment is the existence of parallax effects in saturated regions. The final HDR output may be blurry. See Fig. 12.

## 6 Conclusion and Future Work

In this paper, we demonstrate that the problem of HDR imaging can be formulated as an image translation problem and tackled using deep CNNs. We conducted extensive quantitative and qualitative experiments to show that our non-flow-based CNN approach outperforms the state-of-the-arts, especially in the presence of large foreground motions. In particular, our simple translation network intrinsically avoids distortions and artifacts produced by erroneous optical flow alignment, and is computationally much more efficient. Furthermore, our network can hallucinate plausible details in largely saturated regions with large foreground motions, and recovers highlight regions better than other methods. Our system can also be easily extended with more inputs, and with different reference images, not limited to the medium exposure LDR. It is also robust across different inputs, including images that are not radiometrically calibrated.

While our advantages are clear, it is yet to be a perfect solution. We also observe challenges of recovering massive saturated regions with minimal number of input LDRs. In the future, we would attempt to incorporate high-level knowledge to facilitate such recovery, and devise a more powerful solution.

**Acknowledgement.** This work was supported in part by Tencent Youtu.

## References

1. Debevec, P.E., Malik, J.: Recovering high dynamic range radiance maps from photographs. In: Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques, SIGGRAPH 1997, pp. 369–378. ACM Press/Addison-Wesley Publishing Co., New York (1997). <https://doi.org/10.1145/258734.258884>
2. Dosovitskiy, A., et al.: FlowNet: learning optical flow with convolutional networks. In: IEEE ICCV (2015). <http://lmb.informatik.uni-freiburg.de/Publications/2015/DFIB15>
3. Eilertsen, G., Kronander, J., Denes, G., Mantiuk, R., Unger, J.: HDR image reconstruction from a single exposure using deep CNNs. ACM TOG **36**(6), 178 (2017)
4. Gallo, O., Gelfandz, N., Chen, W.C., Tico, M., Pulli, K.: Artifact-free high dynamic range imaging. In: 2009 IEEE International Conference on Computational Photography (ICCP), pp. 1–7, April 2009. <https://doi.org/10.1109/ICCPHOT.2009.5559003>
5. Gallo, O., Sen, P.: Stack-based algorithms for HDR capture and reconstruction. In: Dufaux, F., Callet, P.L., Mantiuk, R.K., Mrak, M. (eds.) High Dynamic Range Video, pp. 85–119. Academic Press (2016). <https://doi.org/10.1016/B978-0-08-100412-8.00003-6>
6. Grossberg, M.D., Nayar, S.K.: Determining the camera response from images: what is knowable? IEEE Trans. Pattern Anal. Mach. Intell. **25**(11), 1455–1467 (2003). <https://doi.org/10.1109/TPAMI.2003.1240119>
7. He, K., Zhang, X., Ren, S., Sun, J.: Deep Residual Learning for Image Recognition. CoRR abs/1512.03385 (2015). <http://arxiv.org/abs/1512.03385>
8. Heide, F., et al.: FlexISP: a flexible camera image processing framework. ACM TOG **33**(6), 231 (2014)
9. Heo, Y.S., Lee, K.M., Lee, S.U., Moon, Y., Cha, J.: Ghost-free high dynamic range imaging. In: Kimmel, R., Klette, R., Sugimoto, A. (eds.) ACCV 2010. LNCS, vol. 6495, pp. 486–500. Springer, Heidelberg (2011). [https://doi.org/10.1007/978-3-642-19282-1\\_39](https://doi.org/10.1007/978-3-642-19282-1_39)
10. Hu, J., Gallo, O., Pulli, K., Sun, X.: HDR deghosting: how to deal with saturation? In: IEEE CVPR (2013)
11. Isola, P., Zhu, J.Y., Zhou, T., Efros, A.A.: Image-to-image translation with conditional adversarial networks. In: IEEE CVPR (2017)
12. Jacobs, K., Loscos, C., Ward, G.: Automatic high-dynamic range image generation for dynamic scenes. IEEE Comput. Graph. Appl. **28**(2), 84–93 (2008). <https://doi.org/10.1109/MCG.2008.23>
13. Johnson, J., Alahi, A., Fei-Fei, L.: Perceptual losses for real-time style transfer and super-resolution (2016)
14. Kalantari, N.K., Ramamoorthi, R.: Deep high dynamic range imaging of dynamic scenes. ACM TOG **36**(4), 1–14 (2017)
15. Kang, S.B., Uyttendaele, M., Winder, S., Szeliski, R.: High dynamic range video. ACM TOG **22**(3), 319–325 (2003). <https://doi.org/10.1145/882262.882270>
16. Khan, E.A., Akyuz, A.O., Reinhard, E.: Ghost removal in high dynamic range images. In: 2006 International Conference on Image Processing, pp. 2005–2008, October 2006. <https://doi.org/10.1109/ICIP.2006.312892>
17. Mann, S., Picard, R.W.: On being ‘undigital’ with digital cameras: extending dynamic range by combining differently exposed pictures. In: Proceedings of Imaging Science and Technology, pp. 442–448 (1995)

18. Mantiuk, R., Kim, K.J., Rempel, A.G., Heidrich, W.: HDR-VDP-2: a calibrated visual metric for visibility and quality predictions in all luminance conditions. *ACM TOG* **30**(4), 40:1–40:14 (2011). <https://doi.org/10.1145/2010324.1964935>
19. Oh, T.H., Lee, J.Y., Tai, Y.W., Kweon, I.S.: Robust high dynamic range imaging by rank minimization. *IEEE Trans. Pattern Anal. Mach. Intell.* **37**(6), 1219–1232 (2015). <https://doi.org/10.1109/TPAMI.2014.2361338>
20. Photomatix: Photomatix (2017). <https://www.hdrsoft.com>
21. Raman, S., Chaudhuri, S.: Reconstruction of high contrast images for dynamic scenes. *Vis. Comput.* **27**(12), 1099–1114 (2011). <https://doi.org/10.1007/s00371-011-0653-0>
22. Ronneberger, O., Fischer, P., Brox, T.: U-net: convolutional networks for biomedical image segmentation. In: Navab, N., Hornegger, J., Wells, W.M., Frangi, A.F. (eds.) *MICCAI 2015*. LNCS, vol. 9351, pp. 234–241. Springer, Cham (2015). [https://doi.org/10.1007/978-3-319-24574-4\\_28](https://doi.org/10.1007/978-3-319-24574-4_28)
23. Sen, P., Kalantari, N.K., Yaesoubi, M., Darabi, S., Goldman, D.B., Shechtman, E.: Robust patch-based HDR reconstruction of dynamic scenes. *ACM TOG* **31**(6), 203:1–203:11 (2012)
24. Serrano, A., Heide, F., Gutierrez, D., Wetzstein, G., Masia, B.: Convolutional sparse coding for high dynamic range imaging. *Comput. Graph. Forum* **35**(2), 153–163 (2016)
25. Tocci, M.D., Kiser, C., Tocci, N., Sen, P.: A versatile HDR video production system. *ACM TOG* **30**(4), 41:1–41:10 (2011). <https://doi.org/10.1145/2010324.1964936>
26. Tomaszewska, A., Mantiuk, R.: Image registration for multi-exposure high dynamic range image acquisition. In: *International Conference in Central Europe on Computer Graphics and Visualization, WSCG 2007* (2007). [http://wscg.zcu.cz/wscg2007/Papers\\_2007/full/B13-full.pdf](http://wscg.zcu.cz/wscg2007/Papers_2007/full/B13-full.pdf)
27. Tursun, O.T., Akyüz, A.O., Erdem, A., Erdem, E.: The state of the art in HDR deghosting: a survey and evaluation. *Comput. Graph. Forum* **34**(2), 683–707 (2015). <https://doi.org/10.1111/cgf.12593>
28. Tursun, O.T., Akyüz, A.O., Erdem, A., Erdem, E.: An objective deghosting quality metric for HDR images. *Comput. Graph. Forum* **35**(2), 139–152 (2016). <https://doi.org/10.1111/cgf.12818>
29. Zhu, J.Y., Park, T., Isola, P., Efros, A.A.: Unpaired image-to-image translation using cycle-consistent adversarial networks. In: *IEEE ICCV* (2017)
30. Zhu, S., Liu, S., Loy, C.C., Tang, X.: Deep cascaded bi-network for face hallucination. In: Leibe, B., Matas, J., Sebe, N., Welling, M. (eds.) *ECCV 2016*. LNCS, vol. 9909, pp. 614–630. Springer, Cham (2016). [https://doi.org/10.1007/978-3-319-46454-1\\_37](https://doi.org/10.1007/978-3-319-46454-1_37)
31. Zimmer, H., Bruhn, A., Weickert, J.: Freehand HDR imaging of moving scenes with simultaneous resolution enhancement. *Comput. Graph. Forum* **30**(2), 405–414 (2011). <https://doi.org/10.1111/j.1467-8659.2011.01870.x>