Multi-view Image Fusion

Marc Comino Trinidad[1] Ricardo Martin Brualla[2] Florian Kainz[2] Janne Kontkanen[2]
[1]Polytechnic University of Catalonia, [2]Google

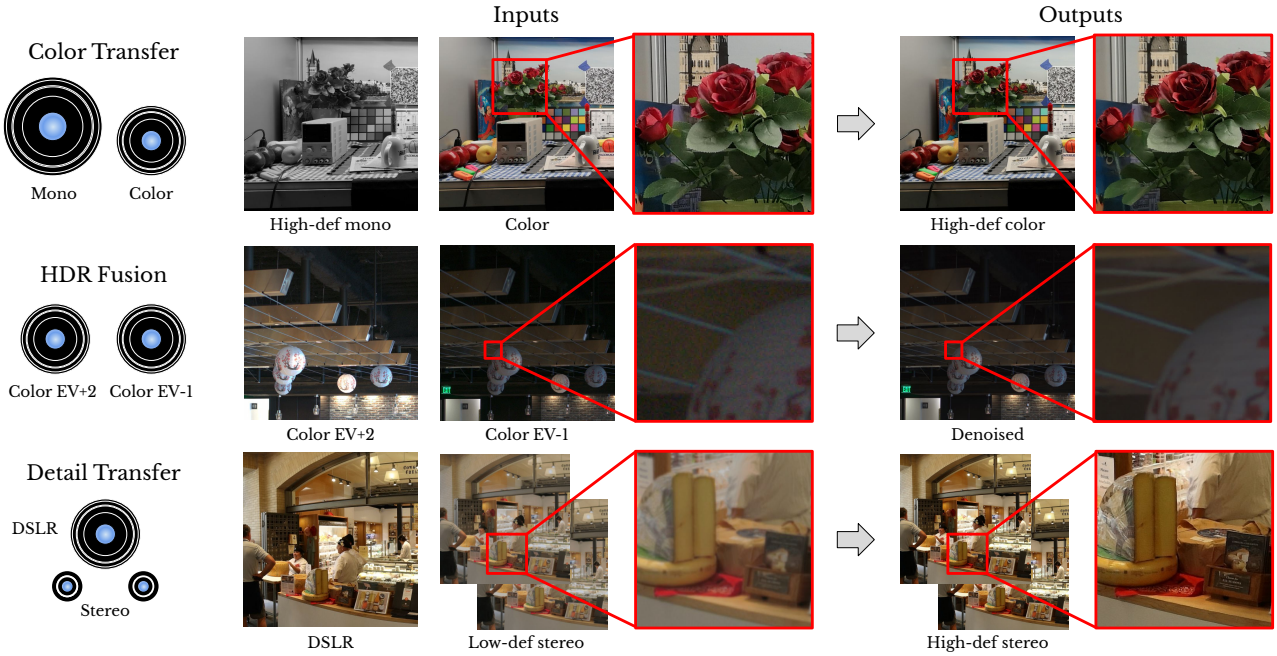


Figure 1: We present a method for multi-view image fusion that is a applicable to a variety of scenarios: a higher resolution monochrome image is colorized with a second color image (top row), two color images with different exposures are fused into an HDR lower-noise image (middle row), and a high quality DSLR image is warped to the lower quality stereo views captured by a VR-camera (bottom row).

Abstract

We present an end-to-end learned system for fusing multiple misaligned photographs of the same scene into a chosen target view. We demonstrate three use cases: 1) color transfer for inferring color for a monochrome view, 2) HDR fusion for merging misaligned bracketed exposures, and 3) detail transfer for reprojecting a high definition image to the point of view of an affordable VR180-camera. While the system can be trained end-to-end, it consists of three distinct steps: feature extraction, image warping and fus
We present a novel cascaded feature extraction method that enables us to synergetically learn optical flow at different resolution levels. We show that this significantly improves the network's ability to learn large disparities. Finally, we demonstrate that our alignment architecture outperforms a state-of-the art optical flow network on the image warping task when both systems are trained in an identical manner.

1. Introduction

In this paper we focus on the problem of fusing multiple misaligned photographs into a chosen target view. Multi-view image fusion has become increasingly relevant with the recent influx of multi-camera mobile devices.

The form factor of these devices constrains the size of lenses and sensors, and this limits their light capturing ability. Cameras with larger lens apertures and larger pixels capture more photons per pixel, and thus show less prominent photon shot noise. This is the reason mobile cameras ' ' :n lagging behind large DSLR systems in quality.

4101

In recent years the use of computational photography has narrowed the gap significantly [11, 17, 19], but the fundamentals have not changed: more light means better images.

Lately it has become common to fit 2, 3 or even 5 cameras [1, 6] into a single mobile device. The use of multiple cameras significantly improves the light gathering ability of the device. At the minimum two cameras capture twice the light of a single camera, but often it is possible to do better by recording different aspects of the scene with each camera

and then fusing the results to get the best of both worlds.

One can envision a number of applications that fall into this class, for example, fusing infrared and color images [33], HDR fusion using bracketed exposures [28], or fusing wide-angle and telephoto views for super-resolution within the central region of the wide-angle image. In this paper we show an end-to-end learned system that is suitable for a number of multi-view fusion applications. We demonstrate its effectiveness in three compelling multi-camera designs:

Color transfer: Monochrome cameras, such as those available in some smart phones [1, 3] capture roughly three times the number of photons compared to the cameras with color mosaic sensors and do not exhibit artifacts introduced by the mosaic. We explore fusing together a color image and monochrome image from slightly different view points to combine the desirable aspects of both cameras.

HDR fusion: We explore an HDR camera design where two cameras take photographs simultaneously but with different exposure settings. We show that fusing the images reduces noise and increases the dynamic range.

Detail transfer: We explore a novel architecture for building a high-quality VR180[2] camera, where a high-resolution image taken by a DSLR camera is warped to the points of view of a cheaper VR180 camera with a field of view close to 180 degrees and a lens separation that matches the human interpupillary distance (IPD). The sizes of the the lenses and bodies of DSLR cameras make it difficult to record VR180 with a pair of DSLRs and achieve a small enough IPD; our design sidesteps this issue.

Our system can be trained end-to-end but it consists of three conceptual stages: feature extraction, warping and fusion. We use a novel cascaded feature pyramid that enables synergetic learning of image alignment across different scales. We show that this architecture has a dramatic impact on learning alignment over large disparities. Instead of training the network to predict optical flow and using that for alignment, we employ the idea of task oriented flow [36] to optimize directly for our use cases since this has proven to produce better results.

We demonstrate the performance of our system with an ablation study and compare it with a state of the art optical flow network [32]. We also compare our HDR fusion technique against Kalantari et al. [23], obtaining comparable results. Finally, we provide a large number of high resolution examples in the supplementary material.

To summarize, the main contributions of this work are: 1) A novel end-to-end CNN architecture for merging information from multiple misaligned images. 2) An image warping module that employs a cascaded feature pyramid to learn optical flow on multiple resolution levels simultaneously. We show that this produces better results than state-of-the-art optical flow for multi-view fusion. 3) A demonstration of the proposed architecture in three different scenarios: Color transfer, HDR fusion, and Detail transfer.

2. Related Work

2.1. High-Dynamic Range Imaging

The seminal work of Devebec and Malik [14] presented a model of a camera's pixel response that allows fusing multiple exposures into an HDR image. Although they assumed a static camera and scene, the technique has recently been introduced to mobile cameras, where a stack of frames is fused to generate an HDR-composite [11, 17, 19]. This works the best if the misalignment between the frames is moderate, which is not the case in some of our applications (we show this in the supplementary material).

Kalantari and Ramamoorthi [23] use a neural network to generate HDR images from exposure stacks of dynamic scenes and corresponding precomputed flow fields. Wu et al [35] propose a similar technique that does not require computing optical flow. Others have focused on burst image fusion by using either recurrent networks [18] or permutation invariant networks [8]. In contrast, our proposed method jointly estimates a warp and fuses the different images to generate a high-quality composite.

2.2. Image Colorization

There is a large amount of literature on single image colorization [21, 37]. Most of the methods presented attempt to generate artificial but plausible colors for grayscale images.

Jeon et al. [22] study stereo matching between a color and a monochrome image in order to compute pixel disparity. They convert the monochrome image to YUV (luminance/chroma) format and populate the chroma (U and V) channels with information from the color input, using the previously computed disparity.

Wang et al. [33] propose colorizing infrared and ultraviolet flash images in order to obtain low-noise pictures in low-light conditions. However, their alignment is based on optical flow [9], and their neural network also needs to learn to account for misregistration artifacts, whereas our network aligns and colorizes at the same time.

2.3. VR Imaging

For virtual reality applications one would ideally capture a complete light field video of a scene. Multiple camera designs have been proposed towards this end, including rings [9] or spheres [26] of outward-facing cameras, or planar camera arrays [34]. Many of these systems do not directly produce stereo views that match the human interpupillary distance, but rely on view interpolation to generate novel views of the scene using computational methods.

Using our proposed method for multi-lens fusion, we envision creating a VR camera where we use detail transfer to project a high quality DSLR image into the viewpoints of a VR camera that captures images with the baseline that

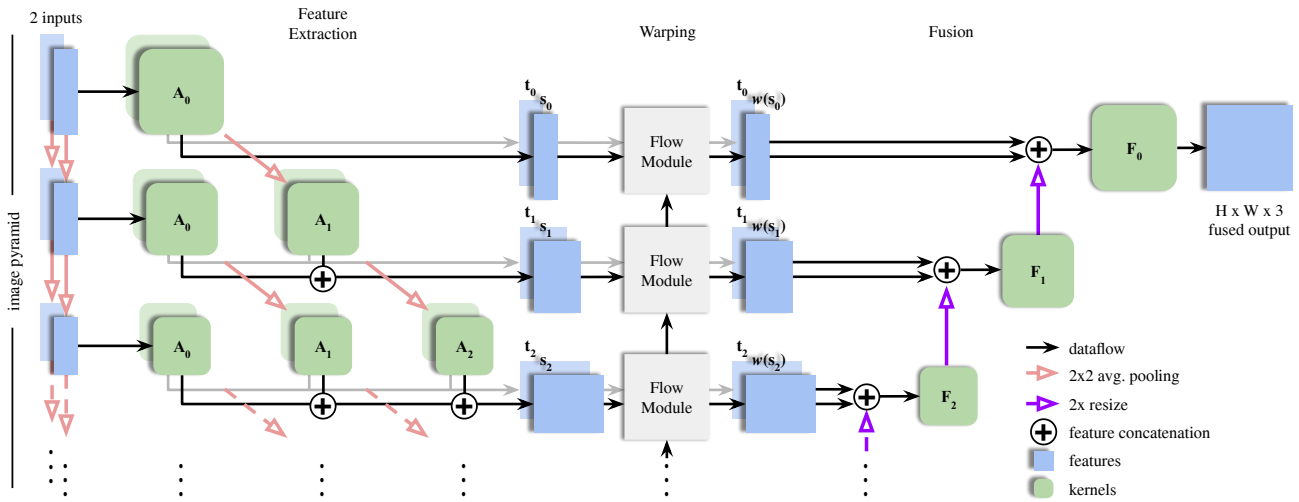


Figure 2: Our architecture takes inspiration from U-Net [30] with an encoder (feature extraction, Section 3.1) on the left and a decoder (fusion, Section 3.3) on the right. Since U-Net cannot efficiently align images, an additional warping module is inserted in the middle (Section 3.2). The green blocks A_n and F_k are kernels whereas the blue blocks represent features. Blocks A_n for $n = 0, 1, 2$ are feature extraction kernels that are sequentially applied to each level of the image pyramid. For each level k, we concatenate the features obtained by applying A_0 on the current level, A_1 on the previous level and A_2 on the level before the previous one, yielding features s_k for the source image and t_k for the target. Thus for all levels except the two finest ones, we have the same amount of feature channels ($2^4 + 2^5 + 2^6$). This allows sharing the flow prediction module for these levels. The source features s_k are warped to the target t_k yielding $w(s_k)$. These aligned features are then concatenated and fused with the information from the coarser pyramid levels to produce the fused output.

matches the human IPD. This is similar in spirit to the work by Sawhney et al. [31] where a hybrid system with a low-quality and a high-quality camera is used to record stereoscopic footage using conventional algorithms.

2.4. Optical Flow

The performance of optical flow techniques has improved dramatically in recent years according to the Sintel benchmark [12]. Fischer et al. [15] introduced FlowNet and used large quantities of synthetic examples as training data. More recent approaches borrow many concepts from traditional optical flow techniques, like coarse-to-fine refinement and residual estimation [27, 32]. Ren et al. [29] extend this idea to temporal flow, and propose computing the flow for a frame in a video sequence by using the estimates for previous frames.

3. PixelFusionNet

We introduce *PixelFusionNet*, a novel end-to-end multi-view image fusion network. The network takes as input two or more images, misaligned in time and/or space, and produces a fused result that matches the point of view of the first input. The network consists of three modules: feature extraction, warping and fusion. These are explained next. A diagram of the architecture is shown in Figure 2.

3.1. Feature Extraction

Our feature extraction architecture is motivated by the observation that optical flow over large disparities is difficult to learn from moderately sized multi-view datasets. One problem is that large disparities are solved on coarse pyramid levels where only a small number of pixels are available for learning. We are interested in processing multi-megapixel images. We typically use $N = 8$ or 9 pyramid levels and train with on crops of 1536×1536 pixels. Thus the coarsest level has only 6^2 or 12^2 pixels, which is a large disadvantage compared to the finest level filters that are learned from more than 2 million pixels per image.

Intuitively, optical flow prediction should be learnable in a scale-agnostic manner: a large disparity in a down-scaled image should look the same as a small disparity at the original resolution. In order to exploit this, we design our flow prediction module (Section 3.2) to share weights among all except two finest levels of the pyramid, which allows synergetic learning on multiple pyramid levels.

To share the flow prediction weights on multiple pyramid levels we use a novel cascaded feature extraction architecture that ensures that the meaning of filters at each shared level is the same. We start by building an image pyramid and extract features from it using the cascaded arrangement shown in Figure 2. Each block A_n for $n = 0, 1, 2$ represents two 3×3 convolutions with 2^{n+4} filters each (we denote the finest pyramid level with zero). The blocks are repeated for all the pyramid levels as shown in the figure. Note that the extracted features are of same size for every level $k \geq 2$. This is in stark contrast to the traditional encoder architecture [30] and other flow prediction methods where the number of filters grows with every down-sampling [16, 20, 32].

3.2. Warping

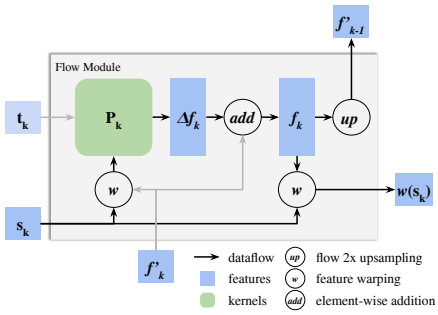


Figure 3: The image warping module that is repeatedly applied starting from the coarsest level $k = N$ towards the finest level $k = 0$ to estimate optical flow. w refers to the warping operation, f'_k is the initial flow estimate ($f'_N = 0$), P_k is the learnable residual flow prediction module, Δf_k is the predicted residual flow and f_k is the refined flow at level k. See Section 3.2 for details.

Our image warping module follows the residual flow prediction idea used in SpyNet [27] and PWC-Net [32], with the caveat that weights are shared across most of the levels. For each pyramid level k, an initial flow prediction f'_k is obtained from level $k+1$ by bi-linear up-sampling (at the coarsest level, $f'_N = 0$). Next the image features at level k are warped using this initial estimate. Then the warped features and the target image features are fed into the learned residual flow prediction network P_k, which only predicts a small correction Δf_k to improve the initial estimate. The refined flow f_k is then upsampled to obtain f'_{k-1} and the process repeats until we reach the finest level $k = 0$. The only learned component in this module is the residual flow prediction network P_k.

Our residual flow prediction network P_k is a serial application of five 2d-convolutions: 3×3×32, 3×3×64, 1×1×64, 1×1×16 and 1×1×2. All except the last layer use ReLU-activation. This module can be small because it is only expected to make small residual corrections. If predictions at level k are accurate, level $k+1$ will only ever need correction vectors within the interval [-1,1] × [-1, 1]. The combined receptive field of the above is 5 × 5 to allow for slightly larger corrections.

Note that while the structure of the warping module is similar to SpyNet and PWC-Net, there are two key differences. 1) Weight sharing: Residual flow prediction modules P_k, for $k \geq 2$ use shared weights and are learned simultaneously on all resolution levels. 2) End-to-end training: Instead of training to minimize the loss against the ground truth optical flow we train to produce image warps by penalizing the difference between the warped image and the target. Thus the network computes a Task-Oriented Flow [36] (see Section3.4 for the specific definition of the loss).

3.3. Fusion

Our fusion module follows the decoder side of the U-Net [30] architecture. The input to the decoder is a feature pyramid where each level is constructed by concatenating the warped source image features with the target image features and applying two 3×3 convolutions with filter sizes 2^{k+4} and ReLU activations, where k is the pyramid level. We denote these convolutions as F_k in Figure 2. Up-sampling from level $k+1$ to k is performed by nearest neighbor sampling followed by $2\times2\times2^{k+4}$ convolution. In the fusion stage each level $F_0, F_1 \ldots F_N$ uses independently learned weights.

At the finest level, F_0 is followed by $1\times1\times3$ convolution with no activation function to produce an RGB-image.

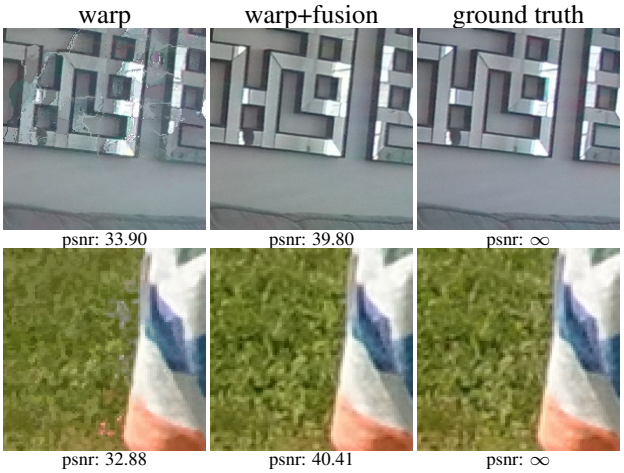


Figure 4: The advantage of adding the fusion stage demonstrated with the color-transfer application. Top: plain warping shows artifacts due to multi-layered parallax on the mirrors. Middle: full network produces pleasing results with higher PSNR. Bottom row: even in easy cases, the fusion stage produces better results than image warping. The PSNRs are computed from 3072x2560 images instead of the small crops shown. The full images can be found from the supplementary material.

3.4. Training

We use two losses for training: a reconstruction loss between the final image and the ground truth and warping loss between the intermediate warped image and the ground truth. Both are computed using the perceptual loss [38] with the pre-trained VGG-19 network. We use the layer weights given by [13]. We have found that the warping loss is the key to generating good results, as otherwise the network might fall into a local minimum where it tries to guess the output without image warping. This is particularly true for applications such as HDR fusion, where relatively good results may be obtained by just denoising the target view.

We also employ a regularization loss on the residual flows. The loss is equal to the L^2 norm of Δf_k for each pyramid level, with weight $\lambda_r = 5e^{-3}$. This is to encourage small residual predictions that are within the receptive field of the residual flow prediction network, P_k.

For efficiency, we compute the warping loss from the

(a) target image (b) full sharing (c) partial sharing (d) no sharing (e) PWC-Net (f) source image

Figure 5: Image warping experiment: Our warping module compared to the state of the art optical flow network. The example shown here demonstrates a challenging case with 350 pixels of disparity. The source image on the right has been warped into the target image on the left using four different models: 1) our approach with weight sharing across all levels 2) our best approach. 3) our approach using no weight sharing across levels 4) PWC-Net. All four approaches have been trained with our dataset using VGG-loss and the same settings.

Metric	No sharing	Partial sharing	Full sharing	PWC-Net
L_1	0.0144	**0.0121**	0.0129	0.0208
VGG	23.10	**18.04**	19.40	31.0

Table 1: Image warping experiment: Mean L_1 and VGG-losses over the test set on three different variants of our alignment architecture + PWC-Net. Model `partial_sharing` performs the best on average. On visual inspection, it outperforms the other models especially on large disparities (see Figure 5).

second finest resolution. We did not find a multi-resolution loss as in PWC-Net or SpyNet useful in our experiments. For all our training we use learning rate $1e^{-4}$, batch size 1 and the Adam optimizer. All in all, our network has 37 times fewer weights than the one of [32].

3.5. Image Warping Experiments

In this section we focus on the warping component of our system and optimize only for the warping loss. To demonstrate the effect of weight sharing, we show an ablation study using three different variants of our architecture: `full_sharing`, `partial_sharing` and `no_sharing`. `full_sharing` and `no_sharing` represent different ways of omitting our cascaded architecture.

In `full_sharing` we share weights across all levels, i.e. we have no independently learned levels at all. This caps the number of extracted filters to 16 to control memory consumption at the finest resolution.

In `no_sharing` we do not share any weights and we use a typical encoder architecture where the number of filters increases with every down-sampling.

Variant `partial_sharing` refers to our proposed architecture that shares weights across all except the two finest levels.

In addition to the ablation study, we compare against a publicly available implementation [4] of PWC-Net, a state of the art optical flow network [32]. To make sure that the receptive field is big enough for our use case, we increased the pyramid depth of PWC-Net from 6 to 8, using 264 and 332 filters respectively.

The results are shown in Figure 5 and Table 1. All models were trained with the same data, the warping loss and the settings explained in Section 3.4.

4. Applications

4.1. Color transfer

In our first experiment we transfer color from an RGB-camera to the point of view of a monochrome target. This task has practical value since monochrome cameras tend to produce higher-quality images. A monochrome sensor does not need a color filter mosaic and thus can capture a larger number of photons. In addition monochrome images are sharper as there is no need for demosaicing. For these reasons, several cellphone manufacturers have released smartphones with additional monochrome cameras [1, 3].

To fuse the images we warp the color image to the monochrome target. While the fusion could also be done the other way around, we chose to warp the color image because this ensures that the high-definition monochrome pixels are not blurred by re-sampling or other stages.

Our proof-of-concept experiment uses a dataset captured using a Yi Horizon stereo camera with fisheye lenses. We chose this camera as it captures synchronized images by default. The baseline between the two lenses is 6.4 cm, so warping is harder than in the cell phone use case where the distance between lenses is usually much smaller.

We converted the left side images to monochrome, and the network was tasked to predict the color images on that side. We entertained the idea of predicting just chroma while keeping luma fixed, but it turned out that the network learned this even when predicting RGB-images. We captured 397 images with 3200×2656 pixels. 61 were randomly selected into the test set. We did not rectify the images. We trained using random 1536×1536 crops.

Figure 6 shows results from the test set. We numerically evaluate against U-Net trained for the same task and an ablated version of our architecture where we only perform warping but no fusion. For numeric results see Table 2. PixelFusionNet outperforms the other two techniques in PSNR, VGG and SSIM metrics. Additionally, in Figure 6 we show a comparison against a larger network where PWC-Net warped images are fed into U-Net for fusion. While U-Net is capable of fixing small scale warping errors, the results with larger disparities corroborates our findings that our warping technique is better suited for the image fu-

(a) color input (b) mono input (c) overlay (d) PWC+U-Net (e) PixelFusionNet (e) full fused image

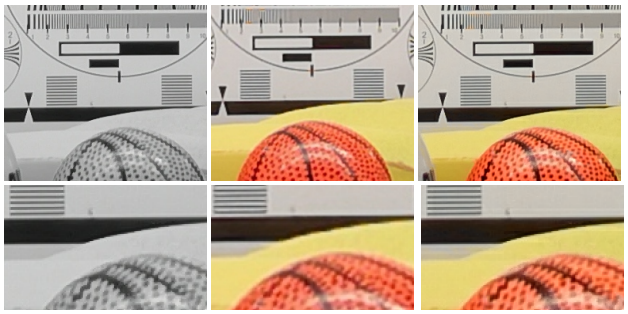
Figure 6: Color transfer: Color (a) and mono images (b) are captured simultaneously from different viewpoints. The disparity is illustrated by an overlay image (c). Color is transfered from the color image to the mono image using PWC-Net+U-Net (d) and our PixelFusionNet. The full fused image is shown in (f). For high resolution examples, see the supplementary material.

sion task (described in Section 3.5).

We also experimented with a real monochrome camera in a Huawei P20 Pro smart phone. This is shown in Figure 7. We fed images from the smart phone's RGB and mono cameras to our network. The fused color image shows definition similar to the monochrome input and has fewer demosaicing artifacts than the input color image.

4.2. HDR fusion

Merging differently exposed low dynamic range images is a widely used technique for improving the dynamic range of a photograph. Depending on the local luminance of the scene, the ideal exposure time varies. If the exposure time is too short noise will dominate the signal, whereas a long exposure may saturate the highlights. By combining differently exposed images, we can select the most suitable exposure locally within the image.



(a) mono input (b) color input (c) fused result

Figure 7: Color transfer: The Huawei P20 Pro captures (a) a mono and (b) a color image, with a crop highlighted in the second row. (c) Our fused color result (c), from the point viewpoint of the mono camera, is of higher resolution and contains fewer color artifacts.

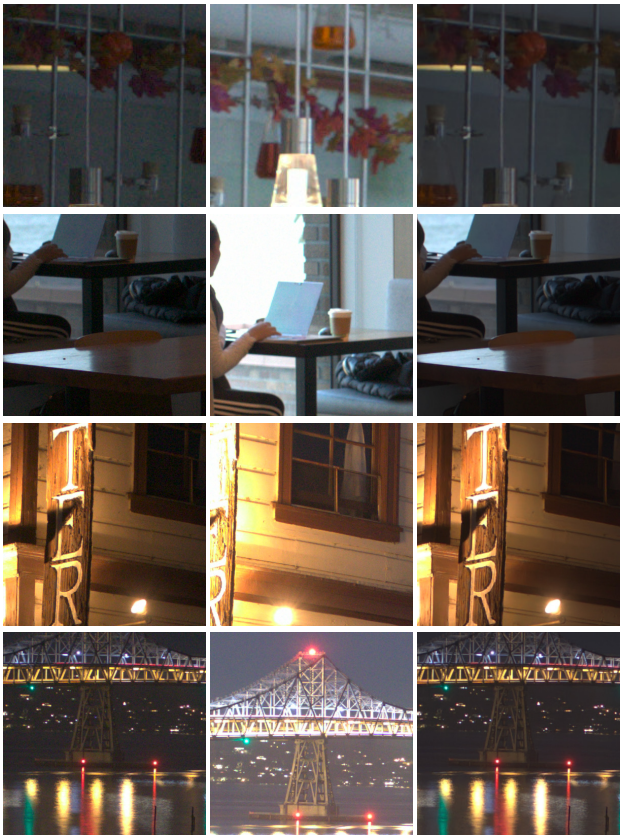
As in all our applications, we focus on the case of misalgined input images. In Sections 4.2.1 and 4.2.2 we experiment with spatial and temporal misalignement, respectively.

4.2.1 Multi-view HDR fusion

For this experiment we captured training data with a pair of Canon 5DS DSLR cameras mounted side by side. We used Tamron 45 mm f/1.8 lenses. An external switch was used to trigger both cameras at the same time.

We let the cameras run in auto-exposure mode, fixing aperture and ISO, and letting the cameras choose the exposure times. We selected an exposure bias of -2 EV on the left and +1 EV on the right side. Most of the time images were exposed 3 EV apart, but since the cameras did not see exactly the same scene the difference could be be larger or smaller. We captured 266 8868×5792 pixel image pairs, including indoor, outdoor, urban and rural subjects. We also included portraits and close objects.

We down-scaled the raw images to half resolution to reduce noise and decrease the parallax in pixels. Then we broke each image to two 2048×2048 pixel crops and added simulated photon shot noise to approximate the noise levels found in images shot with a phone camera. Then we demosaiced both the noise-corrupted versions and the original short-exposure left side image. We ensured that the same white balancing was used in left and right images.



(a) short exposure (b) long exposure (c) fused result

Figure 8: Magnified crops of the HDR fusion experiments. Note the reduced noise in (c). For full images, see the supplementary material.

With the two noise-corrupted images as inputs our network was tasked to predict a noise-free short exposure image on the left side. Note that single-image denoising and HDR fusion are very similar problems. HDR fusion just uses multiple images in hopes of achieving superior results.

We randomly selected 27 images as the test set, and as in Section 4.1 we trained using random 1536×1536 crops.

In Table 2, we compare our results numerically against U-Net, a common choice for neural image denoising [24]. We feed both short and long exposure images to U-Net. We outperform U-Net in the PSNR and VGG metrics, but it gives a slightly better SSIM score than our technique. For visual results see Figure 8 and the supplementary material.

4.2.2 Multi-frame HDR fusion

We also evaluated our method against the state of the art HDR fusion algorithm by Kalantari et al. [23]. Their input consists of three images with short, medium and long exposures. In their method all images are first brought into alignment using the flow algorithm by Liu [25], and then fused together as described in Section 2.

We trained our network using their training set and evaluated the results using the five images that they use to showcase the algorithm in their paper. In this experiment, we added an additional L^2 reconstruction loss term with $\lambda = 10$ to encourage low PSNR (Kalantari et al. optimize solely for L^2-loss).

This training set contains only 74 1500x1000 pixel images which is hardly enough to train the alignment network. In contrast to Kalantari et al., who use a fixed optical flow algorithm, our method requires more varied training data to learn warping as well as fusion.

We use a data augmentation technique similar to the one by Kalantari et al. Specifically, we apply random horizontal and vertical flips and random a number of 90 degree rotations, and we randomly shuffle the color channels. Despite this, we suspect that training with a larger data set would be helpful, but this remains as future work. We achieve quality comparable to their method as shown numerically in Table 2 and visually in the supplementary material.

4.3. Detail transfer

It is common to record stereo imagery using two cameras mounted side by side. Recently VR180 [2] has become a popular capture and delivery format for virtual reality.

In VR180 cameras such as Yi Horizon [7], the two lenses are mounted such that the baseline matches the human interpupillary distance (6-7 cm [5]). This limits the size of lenses and sensors and consequently the resolution and dynamic range of VR180 cameras are not very high.

We evaluated PixelFusionNet on the task of transferring high-quality pixel data from images captured by a Canon 5DS to the points of view of the two lenses of a Yi Horizon.

To capture training data, we mounted a Yi Horizon below a pair of Canon 5DS cameras. All three cameras were connected to an external Arduino microcontroller, which was outfitted with a shutter button. The Arduino was programmed to compensate for the cameras' different shutter delays, so that pressing the button made all three cameras take a picture at the same time.

Before we could train PixelFusionNet we learned to degrade the DSLR-images on the right side to resemble the Yi Horizon quality. We used U-Net as our degradation network, which was shown samples of DLSR images and was tasked to convert them to Yi Horizon images warped to the same viewpoint using optical flow [10].

Once the degradation network was trained, we used it to degrade all the DSLR-images on the right side. This gave us a training set for the PixelFusionNet, which was shown the original left-side DSLR image and the degraded right-side image, and tasked to predict the original right-side image.

We show results visually in Figure 9 and numerically in Table 2, where we compare against U-Net and the ablated version of PixelFusionNet without warping. The full Pix-

(a) Hi-def input (b) Lo-fi target (c) U-Net (d) Warp only (e) PixelFusionNet (f) Ground truth

Figure 9: Detail transfer experiments. With our technique the high-definition source image (a) is warped to the point of view of the low-definition target image (b), and the images are fused to create image (e). For comparison we show results produced by U-Net (c) and by learned image warping without the fusion stage (d). U-Net (c) has difficulties resolving the fine detail in the Roman numerals (top row) and in the railings (bottom row). Image warping (d) exhibits artifacts in the railing close to the camera. These are small crops from the full resolution 5289×4356 pixel images available in the supplementary material.

	Color transfer			**Spatial HDR fusion**			**Temporal HDR fusion**			**Detail transfer**		
Approach	PSNR	VGG	SSIM	PSNR	VGG	SSIM	PSNR	VGG	SSIM	PSNR	VGG	SSIM
Kalantari [23]							**42.67**	**7.45**	**0.9777**			
U-net [30]	37.12	14.48	0.9925	43.43	12.89	**0.9815**				31.96	26.59	0.8893
Warping	32.63	20.81	0.9580							29.86	27.96	0.9056
PixelFusionNet	**38.20**	**12.66**	**0.9956**	**44.23**	**11.71**	0.978	42.40	7.82	**0.9777**	**32.75**	**21.18**	**0.9246**

Table 2: The numeric evaluation of our method on three different applications and four different datasets.

elFusionNet outperforms the other two models in PSNR, VGG and SSIM metrics.

5. Limitations

Our algorithm may fail to align two images if the disparity is too large. The flow estimation searches correspondences within a local window centered at each pixel. Our network uses a receptive field of 5×5 pixels. Theoretically, the aggregate search radius over 9 pyramid levels is $\sum_{k=1}^{9} 2^k = 1022$ but we found that disparities that approach the search radius $2^9 = 512$ of the coarsest level cannot be recovered. Whether this limitation can be removed by increasing the pyramid depth remains as future work.

Moreover, disoccluded areas (parts of the target image not visible on the source image) can cause problems. Our algorithm has proven effective in inpainting these regions, but there may be examples that are too hard. Finally, computing correspondences on saturated images is an ill-posed problem. This is best demonstrated in the HDR fusion failure case shown in the supplementary material.

6. Conclusion

We focused on the problem of multi-view image fusion, and introduced *PixelFusionNet*, a novel end-to-end learnable architecture. Our model first extracts features on all input images, then warps the computed features to the reference view, and finally fuses the information from all images to generate a higher quality fused result. We have applied our approach to three challenging problems: transferring the color from one image to another taken with a higher quality monochrome sensor; using two images taken at different exposures to generate a denoised HDR result; and transferring details from a high-quality image onto a lower quality stereo pair.

Our approach does not rely on camera calibration (neither extrinsics nor intrinsics are required) and thus does not exploit epipolar constraints. However, this is also an advantage as the network can choose to warp patches from a larger area if they are usable for the task at hand. In future work we hope to evaluate our warp prediction network on established optical flow and stereo benchmarks, especially for data sets with large disparities.

7. Acknowledgements

The authors would like to thank Hugues Hoppe, Shahram Izadi, Steve Seitz, John Flynn, Jon Barron, David Gallup, Carlos Hernandez, Sameer Agarwal, Dan Erickson, Sameh Khamis and Marc Levoy for support and inspiring discussions. This work has been partially funded by the Spanish Ministry of Economy and Competitiveness and FEDER under grant TIN2017-88515-C2-1-R, and the Spanish Ministry of Education, Culture and Sports PhD grant FPU14/00725. The majority of the work was done while the first author was an intern at Google.

References

[1] HUAWEI p20 pro. https://consumer.huawei.com/en/phones/p20-pro/. 1, 2, 5

[2] Introducing vr180 cameras. https://vr.google.com/vr180/. 2, 7

[3] Nokia 9 pureview. https://www.nokia.com/phones/en_int/nokia-9-pureview/. 2, 5

[4] Optical flow prediction with tensorflow. https://github.com/philferriere/tfoptflow. 5

[5] Pupillary distance. https://en.wikipedia.org/wiki/Pupillary_distance. 7

[6] Samsung galaxy s10. https://www.samsung.com/us/mobile/galaxy-s10/camera/. 1

[7] Yi horizon vr180. https://www.yitechnology.com/180-vr-camera. 7

[8] Miika Aittala and Fredo Durand. Burst image deblurring using permutation invariant convolutional neural networks. In *The European Conference on Computer Vision (ECCV)*, September 2018. 2

[9] Robert Anderson, David Gallup, Jonathan T Barron, Janne Kontkanen, Noah Snavely, Carlos Hernández, Sameer Agarwal, and Steven M Seitz. Jump: virtual reality video. *ACM Transactions on Graphics (TOG)*, 35(6):198, 2016. 2

[10] Jonathan T Barron and Ben Poole. The fast bilateral solver. In *European Conference on Computer Vision*, pages 617–632. Springer, 2016. 7

[11] Radu Ciprian Bilcu, Adrian Burian, Aleksi Knuutila, and Markku Vehvilainen. High dynamic range imaging on mobile devices. In *2008 15th IEEE International Conference on Electronics, Circuits and Systems*, pages 1312–1315. IEEE, 2008. 1, 2

[12] Daniel J Butler, Jonas Wulff, Garrett B Stanley, and Michael J Black. A naturalistic open source movie for optical flow evaluation. In *European conference on computer vision*, pages 611–625. Springer, 2012. 3

[13] Qifeng Chen and Vladlen Koltun. Photographic image synthesis with cascaded refinement networks. In *The IEEE International Conference on Computer Vision (ICCV)*, Oct 2017. 4

[14] Paul E. Debevec and Jitendra Malik. Recovering high dynamic range radiance maps from photographs. In *Proceedings of the 24th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '97, 1997. 2

[15] Alexey Dosovitskiy, Philipp Fischer, Eddy Ilg, Philip Hausser, Caner Hazirbas, Vladimir Golkov, Patrick Van Der Smagt, Daniel Cremers, and Thomas Brox. Flownet: Learning optical flow with convolutional networks. In *Proceedings of the IEEE international conference on computer vision*, pages 2758–2766, 2015. 3

[16] Alexey Dosovitskiy, Philipp Fischer, Eddy Ilg, Philip Hausser, Caner Hazirbas, Vladimir Golkov, Patrick van der Smagt, Daniel Cremers, and Thomas Brox. Flownet: Learning optical flow with convolutional networks. In *The IEEE International Conference on Computer Vision (ICCV)*, December 2015. 3

[17] Natasha Gelfand, Andrew Adams, Sung Hee Park, and Kari Pulli. Multi-exposure imaging on mobile devices. In *Proceedings of the 18th ACM International Conference on Multimedia*, MM '10, pages 823–826, New York, NY, USA, 2010. ACM. 1, 2

[18] Clement Godard, Kevin Matzen, and Matt Uyttendaele. Deep burst denoising. In *The European Conference on Computer Vision (ECCV)*, September 2018. 2

[19] Samuel W Hasinoff, Dillon Sharlet, Ryan Geiss, Andrew Adams, Jonathan T Barron, Florian Kainz, Jiawen Chen, and Marc Levoy. Burst photography for high dynamic range and low-light imaging on mobile cameras. *ACM Transactions on Graphics (TOG)*, 35(6):192, 2016. 1, 2

[20] Eddy Ilg, Nikolaus Mayer, Tonmoy Saikia, Margret Keuper, Alexey Dosovitskiy, and Thomas Brox. Flownet 2.0: Evolution of optical flow estimation with deep networks. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 3

[21] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and Alexei A Efros. Image-to-image translation with conditional adversarial networks. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 1125–1134, 2017. 2

[22] Hae-Gon Jeon, Joon-Young Lee, Sunghoon Im, Hyowon Ha, and In So Kweon. Stereo matching with color and monochrome cameras in low-light conditions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 4086–4094, 2016. 2

[23] Nima Khademi Kalantari and Ravi Ramamoorthi. Deep high dynamic range imaging of dynamic scenes. *ACM Transactions on Graphics (Proceedings of SIGGRAPH 2017)*, 36(4), 2017. 2, 7, 8

[24] Jaakko Lehtinen, Jacob Munkberg, Jon Hasselgren, Samuli Laine, Tero Karras, Miika Aittala, and Timo Aila. Noise2noise: Learning image restoration without clean data. *arXiv preprint arXiv:1803.04189*, 2018. 7

[25] Ce Liu. Beyond pixels: Exploring new representations and applications for motion analysis, 2009. 7

[26] Ryan S. Overbeck, Daniel Erickson, Daniel Evangelakos, Matt Pharr, and Paul Debevec. A system for acquiring, processing, and rendering panoramic light field stills for virtual reality. *ACM Trans. Graph.*, 37(6):197:1–197:15, Dec. 2018. 2

[27] Anurag Ranjan and Michael J. Black. Optical flow estimation using a spatial pyramid network. In *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, July 2017. 3, 4

[28] Erik Reinhard, Wolfgang Heidrich, Paul Debevec, Sumanta Pattanaik, Greg Ward, and Karol Myszkowski. *High dynamic range imaging: acquisition, display, and image-based lighting*. Morgan Kaufmann, 2010. 2

[29] Zhile Ren, Orazio Gallo, Deqing Sun, Ming-Hsuan Yang, Erik Sudderth, and Jan Kautz. A fusion approach for multi-frame optical flow estimation. In *2019 IEEE Winter Conference on Applications of Computer Vision (WACV)*, pages 2077–2086. IEEE, 2019. 3

[30] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. U-net: Convolutional networks for biomedical image segmentation. In *International Conference on Medical image computing and computer-assisted intervention*, pages 234–241. Springer, 2015. 3, 4, 8

[31] Harpreet S. Sawhney, Yanlin Guo, Keith Hanna, Rakesh Kumar, Sean Adkins, and Samuel Zhou. Hybrid stereo camera: An ibr approach for synthesis of very high resolution stereoscopic image sequences. In *Proceedings of the 28th Annual Conference on Computer Graphics and Interactive Techniques*, SIGGRAPH '01, pages 451–460, New York, NY, USA, 2001. ACM. 3

[32] Deqing Sun, Xiaodong Yang, Ming-Yu Liu, and Jan Kautz. Pwc-net: Cnns for optical flow using pyramid, warping, and cost volume. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 8934–8943, 2018. 2, 3, 4, 5

[33] Jian Wang, Tianfan Xue, Jonathan Barron, and Jiawen Chen. Stereoscopic dark flash for low-light photography. *arXiv preprint arXiv:1901.01370*, 2019. 2

[34] Bennett Wilburn, Neel Joshi, Vaibhav Vaish, Eino-Ville Talvala, Emilio Antunez, Adam Barth, Andrew Adams, Mark Horowitz, and Marc Levoy. High performance imaging using large camera arrays. *ACM Trans. Graph.*, 24(3), July 2005. 2

[35] Shangzhe Wu, Jiarui Xu, Yu-Wing Tai, and Chi-Keung Tang. Deep high dynamic range imaging with large foreground motions. In *Proceedings of the European Conference on Computer Vision (ECCV)*, pages 117–132, 2018. 2

[36] Tianfan Xue, Baian Chen, Jiajun Wu, Donglai Wei, and William T Freeman. Video enhancement with task-oriented flow. *arXiv*, 2017. 2, 4

[37] Richard Zhang, Phillip Isola, and Alexei A Efros. Colorful image colorization. In *European conference on computer vision*, pages 649–666. Springer, 2016. 2

[38] Richard Zhang, Phillip Isola, Alexei A Efros, Eli Shechtman, and Oliver Wang. The unreasonable effectiveness of deep features as a perceptual metric. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 586–595, 2018. 4