HDR Deghosting: How to deal with Saturation?

Jun Hu
Duke University

Orazio Gallo
NVIDIA Research

Kari Pulli
NVIDIA Research

Xiaobai Sun
Duke University

Abstract

We present a novel method for aligning images in an HDR (high-dynamic-range) image stack to produce a new exposure stack where all the images are aligned and appear as if they were taken simultaneously, even in the case of highly dynamic scenes. Our method produces plausible results even where the image used as a reference is either too dark or bright to allow for an accurate registration.

1. Introduction

High-dynamic-range (HDR) imaging has started to make its way to commercial products, such as smartphones. The limited dynamic range of most imaging sensors often fails to capture the irradiance range visible to the human eye in common real-world scenes. A relatively cheap way to address this limitation is to capture a stack of differently exposed pictures of the same scene and merge them, effectively extending the captured range [18, 6]. However, because the merging process assumes that the pixels of the different images are aligned, any motion—either due to the motion of the camera or to anything moving in the scene—will cause *ghosting* artifacts (if the motion is large) or *blurring* artifacts (if the motion is small).

A common approach to address the artifacts due to the camera motion is to first register the low-dynamic-range (LDR) images, a task complicated by the dramatic changes in brightness across the stack, since most registration algorithms rely on the *brightness constancy* assumption [31]. Ward [27] and Jacobs *et al.* [12] address the brightness changes by binarizing each exposure and determining the optimal translation and rotation, respectively. Tzimiropoulos *et al.* [26] compute the gradient map for each exposure and find a similarity transformation in the Fourier domain. Tomaszewska and Mantiuk [25] used SIFT features to estimate a global homography. However, even assuming a perfectly static scene, such rigid transformations would be accurate only for planar scenes, or scenes where all the objects are far from the sensor's plane.

Much research has focused on additional corrections for subjects moving in the scene, under the assumption that the

Figure 1. Our method takes as an input an exposure stack of a dynamic scene captured with a hand-held device (left column, note the dramatic chances in the scene); it selects a reference image and, for each of the other images in the stack, synthesizes an image that looks as if it was taken at the same time as the reference, only with different exposure settings (middle column). These images can then be fused into a single image showing more details (right). Our approach allows gathering data from the images in the stack even for regions that are severely under- or over-exposed in the reference, a main limitation of many state-of-the-art approaches.

camera is static, or that a global registration of the background can be performed. Gallo *et al.* [7] model the exposure change and determine patches that might contain moving objects by counting the pixels that deviate from the predicted behavior. Raman and Chaudhuri [23] follow a similar idea, but they model the intensity change and detect the motion in irregular patches obtained by grouping pixels into super-pixels. These algorithms pay for the reduction of motion artifacts with a potentially reduced dynamic range, as they drop data that does not follow the registration of the background.

Some algorithms incorporate the de-ghosting process in the weighting used to merge the pixels. Jacobs *et al.* [12] detect pixels that would cause ghosting based on the variance and entropy across the exposure stack. Khan *et al.*

[14] use kernel density estimators to compute the probability that a pixel belongs to the background and weight the pixel based on the computed probability. Heo *et al*. [10] use a weight that emphasizes well-exposed pixels and a second weight that enforces consistency across spatial and exposure domains. Zhang and Cham [29] propose to weight the pixel using local gradients across the exposure stack as a measure of consistency. While computationally efficient, these approaches have the drawback that they downweight or completely ignore pixels of moving objects except, possibly, in one of the images. At the same time they may mix in inconsistent pixels, even if with a reduced weight.

More sophisticated methods attempt to establish dense correspondences between the reference image and the other images in the stack. However, standard optical flow algorithms [2] rely on the brightness constancy assumption, which is always violated, by construction, in the case of exposure stacks. Kang *et al*. [13] boost the image intensity to compensate for this and use a standard optical flow to refine the correspondence mapping initialized by a global registration. Zimmer *et al*. [30] propose to compute the optical flow in the gradient domain which is assumed to remain constant as exposures vary. Our earlier work [11] uses the method by HaCohen *et al*. [9] to find dense correspondences; wrong correspondences are then detected and corrected by pasting small patches transformed with local homographies. The recent method by Sen *et al*. [24] [1] converts each image into a linear space inverting the camera response function, and selects an image as the reference for the final HDR image. Using a variant of PatchMatch, they reconstruct an HDR image which maximizes the similarity with the reference image at the pixel level while minimizing the bidirectional similarity metric with the remaining images.

In general, methods dealing with non-rigid scene motion fall in one of two categories, each with its limitations:

- Algorithms that do not define a reference image and incorporate de-ghosting in the definition of the pixel weights. These can only produce good results when moving objects occupy different regions of the scene in each image of the stack; for the same reason, these approaches tend to require larger stacks, to optimize the chances of capturing the background more often.

- Approaches requiring the definition of a reference image. Pixels that are saturated or severely underexposed in the reference image are generally impossible to match: the only pixels with valuable information about the irradiance in those areas are therefore discarded (either because they are detected as outliers, or because they are not matched successfully).

---

[1] Sen *et al*. [24] developed their method concurrently to ours; the paper and code became available only after our original submission, but for completeness we still include comparisons against it in the final version of this paper.

The method we propose falls in the second category. However, while it capitalizes on the benefits of selecting a reference image (producing a consistent image [7]) it also enables us to recover from the other images the regions that contain clipped pixels (either too dark or too bright) in the reference image.

In a nutshell, from each source image S in the stack we attempt to build a new image that looks as if it was taken at the same time as the reference R, but with the exposure settings of S. For the areas where R provides sufficient detail, the process is driven by the reference image to ensure consistency. For the remaining areas, we use other constraints, as reliable direct registration becomes impossible, and we rely mostly on the information in the source image. To get consistent results even when parts of the scene are moving, we ensure that the boundaries of the saturated regions are consistent with both R and S.

Our contribution is a novel method for generating a registered stack from a set of mis-aligned images of dynamic scenes, similar to Hu *et al*. [11] and Sen *et al*. [24]. However, as opposed to their work, our algorithm can be applied to generic non-linearized exposure stacks, and is also capable of dealing with large saturated regions in the reference image, even under large camera motion or scene object displacements. Besides, our method propagates both intensity and gradient information in the reconstruction process, so we can preserve more detail from the exposure stacks.

2. Method

Our algorithm works by first selecting the image with the highest number of well-exposed pixels to be the *reference* image R [13, 7]. Then, for each *source* image S in the stack, it synthesizes a new image L (the *latent* image) that looks like the reference image R, only exposed like S. In particular, the resulting latent image L has two important properties. First, where the reference R is properly exposed, L has image content that is geometrically compatible with R. In Figure 1, where the reference R is the middle exposure, this means for instance that the arms of the woman in the latent images L must appear in the same location as they appear in R. Second, if the reference R has areas that are either too dark or too bright to perform a reliable registration, the resulting L must have content from S that could plausibly appear there: for example, for areas that are saturated in the reference, such as some regions outside the window in Figure 1, we just need to find content from the source S that matches the neighboring areas (which we can reliably register) and whose luminance could plausibly match the bright pixels in the reference: the latent image L simply cannot be "too dark" there. If the reference had been the darkest image (top row in Figure 1), the areas posing these difficulties would have been the dark areas, where details are lost due to clipping.
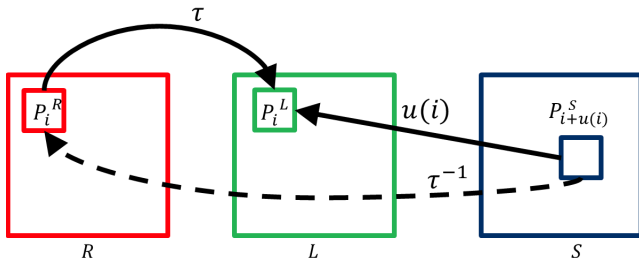
Figure 2. For each source image S in the stack, we want to synthesize the latent image L we would have if we had captured it at the same time as the reference image R, but with the same exposure settings as used to capture S. τ is an intensity mapping function accounting for how the pixel values change under the exposure change. P_i is the $p \times p$ patch centered at pixel i, and the superscript $\{R, L, S\}$ indicates the image it is from. $u(i)$ maps pixel locations from L (and R) to S.

The same process is applied to all the images in the stack, one at a time, so below we will focus on a pair of images. Assuming that we have images $1, ..., 5$ (ordered by exposure time); if the reference image is 3, we first register 2 and 4 to the reference 3, then 2 acts as the reference for 1, and 4 acts as the reference for 5.

Figure 2 illustrates our process in more detail. The reference R is on the left (red), the source S is on the right (blue), and we want to create a latent image L in the center (green) so the shapes of objects in L look like they do in R, except that they have the luminance range of S. We first initialize L by applying a color mapping function τ to R, where τ is initialized using the intensity histograms of the images [8], and is later refined as L is updated. We then find dense correspondences between L and S at the patch level using the Generalized PatchMatch algorithm [3]. If the reference patch P_i^R is not clipped, that is, it is mostly mid-tones and does not contain too dark or bright pixels, PatchMatch looks for a match from S. However, if P_i^R is clipped, neither the color mapping τ, nor direct registration is reliable. In this case we modify the PatchMatch to find a patch P_i^S that could plausibly match P_i^R: pixels in P_i^S should match the pixels in P_i^R that are not clipped, and the rest of the pixels in P_i^S would clip under the current τ. Note, however, that those pixels don't necessarily clip in S, allowing us to bring in more detail to L than is available in R. As we progress, the intensity mapping function τ is updated and refined based on the dense correspondence. To avoid a bad local minimum and to better synthesize clipped areas, these processes are executed iteratively using a coarse-to-fine schedule. We now proceed to explain the details of the whole system.

2.1. Two-picture Synthesis Algorithm

We wish to synthesize the latent image L that looks just as if R was taken using the exposure setting of S: in other

words, L should be consistent with R everywhere in geometry. To account for this, we need to define a *radiance consistency* measure C_r between two images. To maximize the applicability of our algorithm, we do not want to limit its scope to RAW (linear) images. Image signal processors (ISP) apply various non-linear transformations to the almost-linear pixel values; these highly non-linear transformations are usually much more sophisticated than simple gamma compressions, and they sometimes even depend on the image content [11, 15], making it difficult or even impossible to invert the transformations. Hence, instead of linearizing the input images, we take inspiration from the energy definition by Darabi *et al.* [5], but we account for a generic intensity mapping function τ:

$$C_r(L, R, \tau) = \sum_{i \in \Omega} \left(d(L, \tau(R)) + \alpha \, d(\nabla L, \nabla \tau(R)) \right),$$
(1)

where, for clarity, we omitted the dependency of R and L from the pixel location i. Ω is the image domain and $d(x, y) = \|x - y\|^2$. For every pixel i in either image, we extract six channels: the three *RGB* components and the three corresponding gradients. The parameter α balances the color and gradient (texture) consistencies. In addition to boosting the details of the texture [1, 21], using gradients helps to compensate for exposure changes [30]. The intensity mapping function τ describes how the *RGB* values change from the reference to the source image.

Minimizing the cost of Eq. 1 is an ill-posed problem, so we need additional constraints to better define L. We can define a term structurally very similar to Eq. 1, that encourages *texture consistency* between L and the source image S:

$$C_t(S, L, u) = \hspace{4cm} (2)$$
$$\frac{1}{p^2} \sum_{i \in \Omega} \left(d(P_i^L, P_{i+u(i)}^S) + \alpha \, d(\nabla P_i^L, \nabla P_{i+u(i)}^S) \right),$$

where P_i^S is a $p \times p$ patch centered at i in image S (same for P_i^L and L) and $u(i)$ maps patches in L to the corresponding patches in S, see Figure 2. If α is set to zero, this term resembles the coherence term defined by Wexler *et al.* [28] and Kopf *et al.* [16]. We operate in the *RGB* color space and only search over translations, which makes the updates of L faster but does not lower the quality of our results, given the expected changes in an exposure stack.

Now we can combine the two energy functions and find the desired image:

$$L^* = \arg \min_{L, \tau, u} \left(C_r(L, R, \tau) + C_t(S, L, u) \right). \quad (3)$$

2.1.1 Optimization Algorithm

We propose to decompose the optimization problem in Eq. 3 into three relatively simple sub-optimizations, and then iter-

ate between them until convergence. For the coarsest level, we initialize τ using the intensity histograms of the images [8] to minimize the effect of misalignment between the images, initialize $L = \tau(R)$, and apply the Generalized PatchMatch on S and L to initialize u.

In the first step, given the existing L, we optimize for u; note that u only appears in C_t, Eq. 3. C_t can be minimized globally with respect to u, as the latter is independent for each pixel, see Eq. 2. The optimal solution can therefore be reduced to finding the nearest-neighbor patches in S for each patch P_i^L. Instead of a complete search, we use Generalized PatchMatch [3].

In the second step, given the existing u and τ, we seek to find a solution L that minimizes Eq. 3; note that this equation is quadratic in L and it is therefore equivalent to the following quadratic function:

$$L^* = \arg\min_L (\, d(L,T) + \alpha \, d(\nabla L, \nabla T) \,) \qquad (4)$$

where, using Eqs. 1 and 2, and summing over the pixels in the patches rather than over than over the patches themselves, Eq. 2, we define the auxiliary image T as

$$T(i) = \frac{1}{2}\tau(R(i)) + \frac{1}{2p^2} \sum_{j \in n(i)} S(i + u(j)), \qquad (5)$$

where $n(i)$ is a $p \times p$ window centered at i. Basically, T is the weighted average of the colors of all the similar pixels in S and the patch in $\tau(R)$, while ∇T denotes the weighted average of the gradients. Eq. 4 is a Screened Poisson equation, which can be optimized efficiently in the Fourier domain [4]. The square function $d(x,y)$ we defined is computationally efficient, but the result is very sensitive to outliers. To avoid the effects of outliers, we add two weighting terms:

$$T(i) = \frac{1}{s}[w_\tau(i)\tau(R(i)) + \frac{1}{p^2} \sum_{j \in n(i)} w_u(j)S(i + u(j))], \qquad (6)$$

where $w_\tau(i)$ and $w_u(i)$ reflect the confidence of the intensity mapping function $\tau(\cdot)$ and the geometric mapping $u(\cdot)$ for pixel i, respectively, and s is the normalization factor $w_\tau(i) + \frac{1}{p^2} \sum_{j \in n(i)} w_u(j)$. Note that with two additional weighting terms, if a matching pixel is in a useful range in both R and S we combine the information. If one of the two images is bad, we use the other image. Finally, if both images are bad, we use only the reference.

The intensity mapping function τ, which describes how the *RGB* values change from the reference to the source image, cannot be accurate across the whole range, due to saturation and under-exposure. For example, if S was captured with a shorter exposure time (darker) than R, and if the top of the range in the domain of R is saturated, τ will be flat in that area, thus not providing any relevant information; all the useful information for registration and HDR image
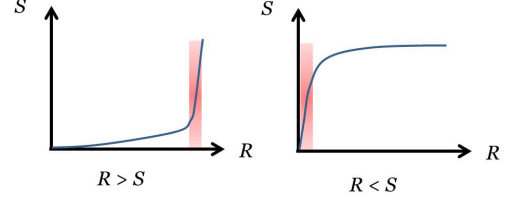
creation is in S. The opposite may be true when S was captured with a longer exposure time, see inset, where red bands show the range in which the mapping τ is not reliable. We choose w_τ to reflect the quality of $\tau(R(i))$: $w_\tau(i)$ is ϵ (a small constant) if $R(i)$ is severely over- or under-exposed, but if the quality of pixel i is good, $w_\tau(i)$ is 1.

The weight function $w_u(i)$ indicates the confidence in the mapping $u(i)$. HaCohen *et al*. [9] define this confidence using the local consistency ratio of $u(i)$, but this may fail for the regions where over- or under-exposure causes the texture to be weak. Instead, inspired by Wexler *et al*. [28], we define:

$$w_u(i) = \begin{cases} \exp\left(-\dfrac{d\big(\tau(P_i^R), \, P_{i+u(i)}^S\big)}{2\sigma_1^2}\right) & \text{if } R \text{ is not clipped} \\[2ex] \exp\left(-\dfrac{d\big(P_i^R, \, \tau^{-1}\big(P_{i+u(i)}^S\big)\big)}{2\sigma_2^2}\right) & \text{if } R \text{ is clipped} \end{cases}$$

$$(7)$$

Intuitively, we normally want to use pixels from S when they are consistent with R (the first case in Eq. 7). However, consider an area that is saturated in R and assume that we are working with an S that is darker, and therefore better exposed. In such regions, $\tau(P_i^R)$ is not reliable and we want to relax the requirement that patches from S have to match, or we would reject all the patches in that area. On the other hand, if a patch in S is so dark that it wouldn't possibly become saturated in R we also don't want to allow its use. Basically, by applying τ^{-1} to $P_{i+u(i)}^S$ first (the second case in Eq. 7), we say that if this patch from S would saturate we are still willing to use it. In this way, the clipped areas of R in L can be reasonably synthesized using the information from S. σ_1 and σ_2 are data-dependent parameters controlling the smoothness of the induced error surface and we compute them as Wexler *et al*. [28].

In the third and last step, given the existing L, we need to re-estimate the intensity mapping function (IMF) τ (Eq. 1) [17, 8, 22, 9, 11]. We follow the parametric models of Ha-Cohen *et al*. [9] and Hu *et al*. [11],

$$\tau_c = \arg\min_\tau \sum_{i \in \Omega} ||\tau(R_c(i)) - L_c(i)||_1, \qquad (8)$$

with $\tau_c'(\cdot) \geq 0$, $\tau_c(\cdot) \in [0,1]$, and $c \in \{r,g,b\}$, but with two main differences: first, instead of using RANSAC to remove outliers, we use the L_1 norm, which can be optimized using iteratively reweighed least squares (IRLS). Second,

in addition to the hard monotonicity constraint, we also require the function to be within $[0, 1]$, and be convex (or concave) if the exposure time of R is longer (or shorter) than that of S. We found that this additional constraint creates a better, generally smoother IMF.

2.1.2 Multi-scale Solution

At each of the steps described above, the objective function is guaranteed to not increase. To further enforce a better local optimum, and to speed up convergence, we use a pyramid approach. The optimization starts at the coarsest scale of a Gaussian pyramid, and the solution is propagated to finer scales. When moving from a level to a finer one, three variables need to be propagated; we transfer τ as is, and linearly interpolate the mapping u. However, we found that linear interpolation of the latent image L leads to blurry results. Therefore, we propagate the solution using the weights w_τ and w_u described above; the rationale is that each pixel of image L at a given scale should be initialized with the corresponding pixel from the reference image from the same level of the pyramid (appropriately mapped with τ) if it is within a reasonable range. Otherwise, it should be initialized using the source image S (using the mapping u derived from the previous level).

3. Results

We now compare the performance of our algorithm to state-of-the-art approaches. As mentioned in Section 1, we are only aware of four methods that attempt to address the general case of camera motion and scene changes at the same time [13, 30, 11, 24]. All the fused results were generated using the method by Mertens *et al*. [20], with the exception of Figure 6, which was tonemapped with the method by Mantiuk *et al*. [19] to allow for a fair comparison with the method by Sen *et al*.

Figures 3 and 4 show results sensibly better than Zimmer *et al*. and Kang *et al*. respectively. When the reference image is reasonably well-exposed everywhere, our method produces very similar results as Hu *et al*. However, when part of the reference is saturated, as in Figure 5, Hu *et al*. discard valuable information from the shorter exposure (first row, middle image); our method, on the other hand, successfully captures all the available information in the synthesized latent image (second row, middle image). Sen *et al*. assumes exposure stacks of RAW or linearized images. For the examples shown in Figure 5, this assumption is violated because no estimation of the camera response function was available, and their result shows visible artifacts. Figure 6 shows another case with a large saturated region. We use RAW images as the input for the algorithm by Sen *et al*. and their non-linear counterpart (first row in Figure 6) as the input to our method. Note that the halos
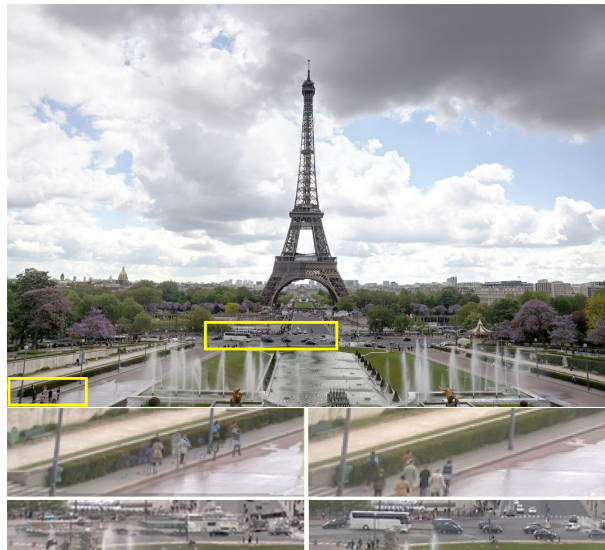

Figure 3. Comparison with Zimmer *et al*. [30]. The top image shows our result. The two bottom rows show blow-outs of two different regions of the image that are problematic for Zimmer *et al*. (left). Our algorithm does not produce artifacts (right). Images courtesy of Henning Zimmer.

in the results by Sen *et al*. are not caused by the tonemapping algorithm, rather they are artifacts of their registration algorithm. In our result (bottom, rightmost image in Figure 6) the sky is more faithful to the original images and no artifacts are introduced. As we mentioned in the previous section, we attempt to preserve as much information as possible from the exposure stack by using both the intensity and the gradients in our reconstruction.

As for any patch-based algorithm, our results are somewhat affected by the patch size p. In all our examples we used $p = 10$. However, in some situations this may be too small a neighborhood. Figure 7 shows an extreme case of a stack comprising only two images, with a region that is saturated in both images, demonstrating one of the limitations of our method. Note that any existing method would be hard-pressed to achieve decent results here because the dramatic change of exposure makes it extremely difficult to match pixels across the images; the fact that the stack contains only two pictures also constitutes a challenging situation for most state-of-the-art algorithms. Our method can register the images correctly despite selecting a reference image that has a completely saturated sky. However, since the sun is saturated in both images, our algorithm fills in the saturated sun using non-saturated pixels from S. Since this region is saturated, the algorithm technically did the right thing: it filled the region with the available information. However, most photographers would prefer the sun to be left un-touched; in this case a simple increase of the patch size to $p = 15$ solves the problem.

Figure 4. Comparison with Kang *et al*. [13]. Note the large motion of the head in the original images (left). The middle image is the result by Kang *et al*., note that the artifacts around the ears and muzzle. On the right is our result. Images courtesy of Sing Bing Kang.
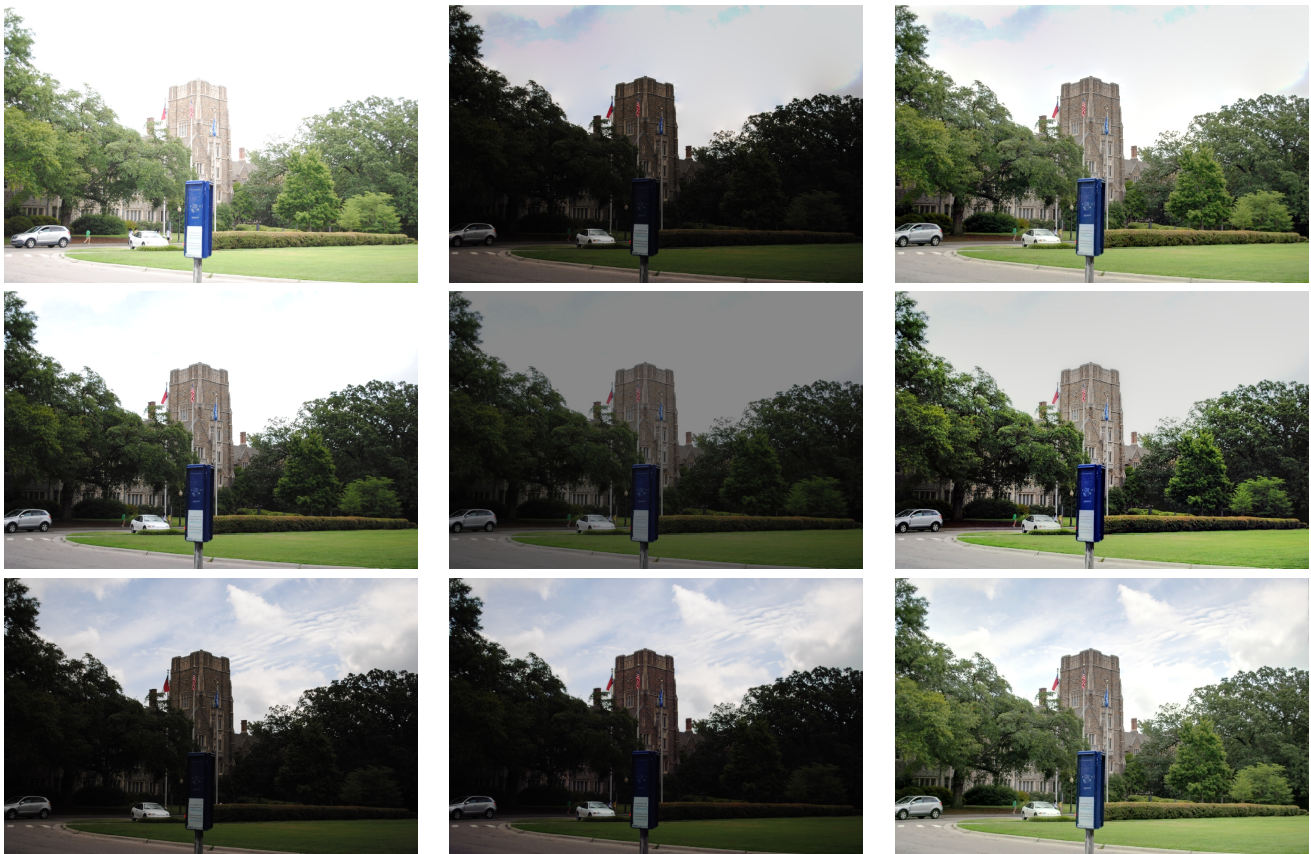

Figure 5. A comparison with Hu *et al*. [11] and Sen *et al*. [24]. The first column shows the original images. The reference, as selected by Jun *et al*., is the middle exposure. Notice that the sky is almost completely saturated, causing their algorithm to disregard useful information in the short exposure (top row, middle image), and leading to poor quality in the fusion result (top right). Sen's algorithm is designed to work on linear exposure stacks. For this non-linear stack, a reliable estimation of the camera response function would require acquiring a stack of registered images. Partially due to the non-linearity of the input images, their method fails in reconstructing the content for the saturated regions in the reference: both reconstructed HDR image (middle row, rightmost image) and the intermediate aligned shorter exposure stack (middle row, middle image) show a degraded quality. With the same reference frame our algorithm can synthesize a novel image which is completely consistent with the reference, and also captures all the details of the sky (bottom row, middle image). This directly reflects in the high quality of our exposure fusion result (bottom row, rightmost image).

Figure 6. Another comparison with Sen *et al*. [24]. The first column shows the original images in the stack, the middle exposure is selected as the reference. For the method by Sen *et al*., we first linearize the original images and use the linearized exposure stacks as the input. Their algorithm generates a plausible result (top middle and right). However, their method still suffers from various artifacts. For example, the blurred sky in the saturated region and the halo around the dome are unexpected. Note that the halo in the reconstructed shorter exposure is not caused by tone mapping but the errors in HDR reconstruction. For the tone mapped HDR image (top right), the reconstructed sky is not natural for the saturated region in the reference. Our algorithm can synthesize an image (bottom middle) that is completely consistent with the reference and also preserves as much information as possible from the whole exposure stack. Our tone mapped HDR image is plausible and virtually artifact free (bottom right).
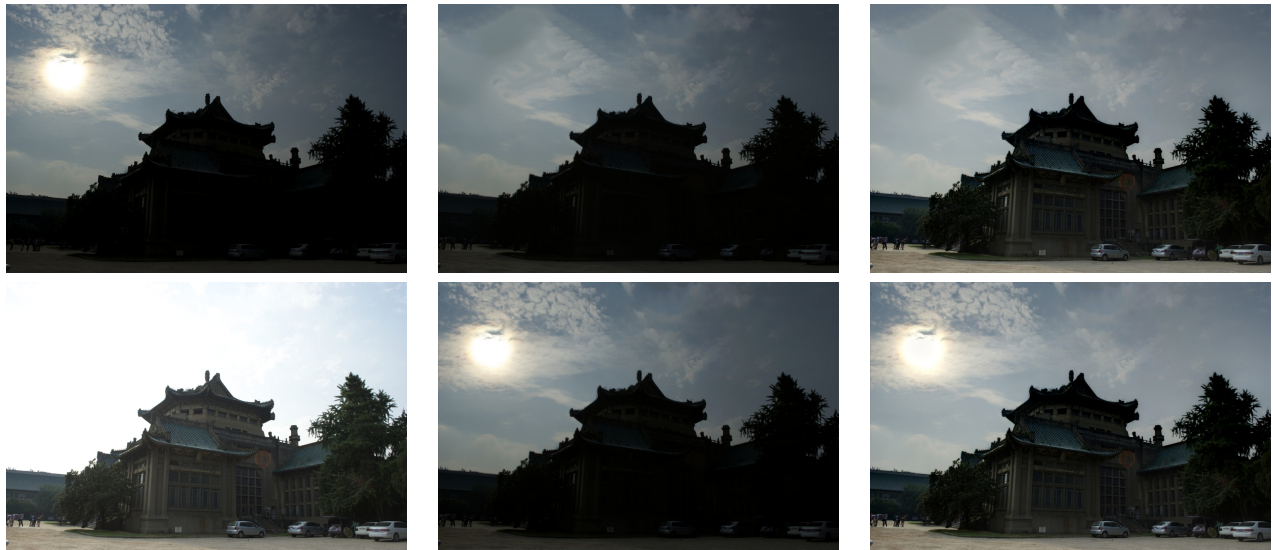

Figure 7. A very challenging 2-image stack. The original images (left) are dramatically separated in terms of exposure time: the areas that are correctly exposed in one are barely visible in the other. An interesting feature of this stack is that the region around the sun is saturated in both images. Note that the longer exposure, which we selected as the reference (left bottom), is completely saturated in the sky; our algorithm attempts to synthesize the saturated region in the source image from other pixels in the same image, thus effectively removing the sun (middle top). A larger patch size ($p = 15$) forces the algorithm to leave the sun region untouched (middle bottom). The last column shows the exposure fusion result for the standard patch size (top) and for the larger patches (bottom).

4. Conclusions

We have presented a novel method to generate a perfectly aligned stack from a set of images of a dynamic scene, captured with a hand-held camera. Four previous methods can deal with both the camera and scene object motion at the same time: Kang *et al*. [13], Zimmer *et al*. [30], Hu *et al*. [11], and Sen *et al*. [24]. We showed that our method, like [11] and [24], compares favorably with the first two; in addition we showed that our method outperforms [11] and [24] in several ways. It successfully deals with large saturated regions in the reference image, which is the most common limitation for algorithms that select a reference frame. Further, our algorithm does not require linear images; this not only allows to capitalize on the finely tuned processing that cameras perform when compressing the images to 8 bits, but it also extends the applicability of the proposed approach to a larger set of scenarios.

References

[1] A. Agarwala, M. Dontcheva, M. Agrawala, S. Drucker, A. Colburn, B. Curless, D. Salesin, and M. Cohen. Interactive digital photomontage. In *SIGGRAPH*, 2004.

[2] S. Baker and I. Matthews. Lucas-Kanade 20 Years On: A Unifying Framework. *Int. J. Computer Vision*, 2004.

[3] C. Barnes, E. Shechtman, D. B. Goldman, and A. Finkelstein. The generalized PatchMatch correspondence algorithm. In *ECCV*, 2010.

[4] P. Bhat, B. Curless, M. F. Cohen, and C. L. Zitnick. Fourier analysis of the 2d screened poisson equation for gradient domain problems. In *ECCV*, 2008.

[5] S. Darabi, E. Shechtman, C. Barnes, D. B. Goldman, and P. Sen. Image Melding: Combining Inconsistent Images using Patch-based Synthesis. In *SIGGRAPH*, 2012.

[6] P. E. Debevec and J. Malik. Recovering high dynamic range radiance maps from photographs. In *SIGGRAPH*, 1997.

[7] O. Gallo, N. Gelfand, W. Chen, M. Tico, and K. Pulli. Artifact-free high dynamic range imaging. In *ICCP*, 2009.

[8] M. Grossberg and S. Nayar. Determining the Camera Response from Images: What is Knowable? *IEEE Trans. PAMI*, 2003.

[9] Y. HaCohen, E. Shechtman, D. B. Goldman, and D. Lischinski. Non-rigid dense correspondence with applications for image enhancement. In *SIGGRAPH*, 2011.

[10] Y. S. Heo, K. M. Lee, S. U. Lee, Y. Moon, and J. Cha. Ghost-free high dynamic range imaging. In *ACCV*, 2011.

[11] J. Hu, O. Gallo, and K. Pulli. Exposure stacks of live scenes with hand-held cameras. In *ECCV*, 2012.

[12] K. Jacobs, C. Loscos, and G. Ward. Automatic high-dynamic range image generation for dynamic scenes. *IEEE CG&A*, 2008.

[13] S. B. Kang, M. Uyttendaele, S. Winder, and R. Szeliski. High dynamic range video. In *SIGGRAPH*, 2003.

[14] E. Khan, A. Akyüz, and E. Reinhard. Ghost removal in high dynamic range images. In *ICIP*, 2006.

[15] S. J. Kim, H. T. Lin, Z. Lu, S. Süsstrunk, S. Lin, and M. S. Brown. A new in-camera imaging model for color computer vision and its application. In *IEEE Trans. PAMI*, 2012.

[16] J. Kopf, W. Kienzle, S. Drucker, and S. B. Kang. Quality prediction for image completion. In *SIGGRAPH Asia*, 2012.

[17] S. Mann. Comparametric equations with practical applications in quantigraphic image processing. *IEEE Trans. Image Processiong*, 2000.

[18] S. Mann and R. Picard. Being 'undigital' with digital cameras: Extending dynamic range by combining differently exposed pictures. In *IS&T*, 1995.

[19] R. Mantiuk, K. Myszkowski, and H.-P. Seidel. A perceptual framework for contrast processing of high dynamic range images. *ACM Trans. Appl. Percept.*, 2006.

[20] T. Mertens, J. Kautz, and F. V. Reeth. Exposure fusion. In *Pacific Graphics*, 2007.

[21] P. Pérez, M. Gangnet, and A. Blake. Poisson image editing. In *SIGGRAPH*, 2003.

[22] K. Pulli, M. Tico, and Y. Xiong. Mobile Panoramic Imaging System. In *WS Embedded Computer Vision, CVPR*, 2010.

[23] S. Raman and S. Chaudhuri. Reconstruction of high contrast images for dynamic scenes. *Vis. Comput.*, 2011.

[24] P. Sen, N. K. Kalantari, M. Yaesoubi, S. Darabi, D. B. Goldman, and E. Shechtman. Robust Patch-Based HDR Reconstruction of Dynamic Scenes. In *SIGGRAPH Asia*, 2012.

[25] A. Tomaszewska and R. Mantiuk. Image registration for multi-exposure high dynamic range image acquisition. In *WSCG*, 2007.

[26] G. Tzimiropoulos, V. Argyriou, S. Zafeiriou, and T. Stathaki. Robust FFT-based scale-invariant image registration with image gradients. *IEEE Trans. PAMI*, 2010.

[27] G. Ward. Fast, robust image registration for compositing high-dynamic range photographs from handheld exposures. *Journal of Graphics Tools*, 2003.

[28] Y. Wexler, E. Shechtman, and M. Irani. Space-time completion of video. *IEEE Trans. PAMI*, 2007.

[29] W. Zhang and W.-K. Cham. Reference-guided exposure fusion in dynamic scenes. *J. Vis. Commun. Image Represent.*, 2012.

[30] H. Zimmer, A. Bruhn, and J. Weickert. Freehand HDR imaging of moving scenes with simultaneous resolution enhancement. *Computer Graphics Forum*, 2011.

[31] B. Zitov and J. Flusser. Image registration methods: a survey. *Image and Vision Computing*, 2003.