Non Local Exposure Fusion

Cristian Felipe Ocampo-Blandon, Yann Gousseau

Non Local Exposure Fusion

Cristian F. Ocampo-Blandon and Yann Gousseau
LTCI, CNRS, Telecom ParisTech, Universite Paris-Saclay, 75013, Paris, France

***Abstract*—Exposure fusion is an efficient method to obtain a well exposed and detailed image from a scene with high dynamic range. However, this method fails when there is camera shake and/or object motions. In this work, we tackle this issue by replacing the pixel-based fusion by a fusion between pixels having similar neighborhood (patches) in images with different exposure settings. In order to achieve this, we compare patches in the luminance domain. We show through several experiments that this procedure yields comparable or better results than the state of the art, at a reasonable computing time.**

I. Introduction and previous works

Natural scenes often exhibit luminance ranges that are beyond the capacity of most imaging sensors. In such cases, a single standard photograph cannot capture details in dark regions without producing saturated values in the brightest regions. A classical way to create a high dynamic range (HDR) image faithfully representing the luminance of a scene is to combine photographs acquired with different exposure times [1]. Once the response function of the camera is known (or when working with linear RAW images) and when the scene is static, this combination task is essentially a statistical problem [2], [3]. In order to visualize such an image on a classical display, contrast is then compressed by using a so-called *tone mapping* operator [4], [5], [6]. An alternative to this two-step procedure (HDR image creation followed by tone mapping) was proposed in [7] and is called *exposure fusion*. The main idea is to bypass the HDR creation step and to directly create a low dynamic range image (typically made of 8 bits per color channel) by fusing the input images corresponding to different acquisition times. Specific weights are used to ensure that the final result is contrasted enough, has vivid colors and avoid over and under-exposure. Using classical ideas from computer graphics [8], the images are fused in a multi-scale framework, enabling one to blend images seamlessly.

This procedure is very efficient and has yielded numerous softwares and plug-ins. It has also triggered an abundant research literature proposing variants on the original method: involving image decomposition into several components at the patch level [9], formulating the problem as a MAP estimation [10] or a maximum entropy optimization [11], or involving the bilateral filter as in classical tone mapping algorithms [12]. Nevertheless, such approaches have a common drawback: they fail when there is either camera shake or moving objects in the scene. Indeed, they all perform the fusion at pixel level and assume that images are registered and that objects are still. Of course, this is a serious limitation in practice, and several works have tackled this issue. Anti-ghosting algorithms [13], [14], [15], [16], [17] propose to perform image alignment and possibly to explicitly detect moving objects to prevent using them in the fusion. Recently, the methods from [18], [19] proposed to solve the problem by an iterative optimization procedure relying on correspondences between patches. In particular, the authors of [18] propose to create, from a reference image and a set of differently exposed images, a set of images that are all aligned and radiometrically coherent with the reference, and whose content is automatically modified (in the case of moving objects) to match the reference. This is achieved thanks to contrast prescription and patch-based content comparison between the images. As illustrated in [18], exposure fusion can then be applied to the aligned set to yield a satisfactory final image, even in the case of camera shake and object motion.

In this paper, we propose an exposure fusion method that deals both with camera shake and object motion. The basic idea is very simple and illustrated by Figure 1: instead of fusing values taken by pixels at the same spatial position, we fuse values of pixels having similar neighborhood (thereafter called a patch) in the different images. The method is therefore in the spirit of non-local restoration methods such as the non-local means [20], and also share similarities with the non-local method introduced in [21] for the creation of HDR images. We show that such a simple approach has the ability to deal with unregistered images (although a previous global image registration can be performed to accelerate the process) and with moving objects. By construction, the method also boils down to the original exposure fusion algorithm in the absence of camera and object motions. The rest of the paper is organized as follows. In Section II, the original exposure fusion algorithm is first briefly recalled and a detailed presentation of the proposed method is provided. In Section III, we show experimental results and comparisons with the recent state-of-the-art algorithm from [18].

II. Non-Local Exposure Fusion

A. Classical Exposure Fusion

The method introduced in [7] proposes to fuse a series of images $I_1, \dots, I_N$ acquired with different exposure settings (the knowledge of these is not needed). For each pixel x and index i a weight $W_i(x)$ is defined by taking into account the quality of contrast, color saturation and well-exposedness at this pixel. The idea is then to fuse the values $I_i(x)$ according to the weights. The most straightforward approach would be to define the resulting image R as

This work is partly supported by Colciencias under the grant "Progama doctoral de Becas" call 529.

$$R(x) = \sum_{i=1}^{N} W_i(x) I_i(x), \tag{1}$$

but this yields incoherences in flat regions and visible seams at slow transitions. In order to achieve seamless fusion, the blending is performed in a multi-scale framework. For each level l of a Laplacian pyramid [8], the Laplacian pyramid of the resulting image R is computed as

$$\mathcal{L}_l(R) = \sum_{i=1}^{N} \mathcal{G}_l(W_i) \mathcal{L}_l(I_i),$$

where $\mathcal{L}_l(I_i)$ is the Laplacian pyramid at level l of image i and $\mathcal{G}_l(W_i)$ is the Gaussian pyramid at level l of the weight map W_i. The final image R is then reconstructed from its Laplacian pyramid.

As explained in the introduction, this (otherwise very efficient) method fails when there is camera shake or object motions.

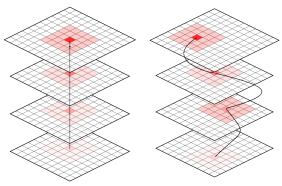
B. Non-local fusion

In order to deal with dynamic scenes we propose, for each pixel x, to replace the fusion of values at this pixel with the fusion of values at well chosen pixels $f_1(x), \dots, f_N(x)$ in the series of images. In the static case these pixels would simply be all equal to x, and in the general case, they will be defined in order to compensate motions, as we will now see.

In a way similar to the HDR creation method from [21], we define $f_i(x)$ to be the pixel having its neighborhood in I_i the closest to the one of x in I_{ref} (where $ref \in (i_1, \dots, i_N)$) a reference image to be defined later. More precisely, we define $P_i(x)$, a patch at pixel x, to be the collection of values in I_i in a $(2s+1) \times (2s+1)$ square window centered at x. We assume that we have a distance d between patches (to be defined in the next section) and then define the pixel most similar to x in I_i to be

$$f_i(x) = \arg\min_{y \in x + \mathcal{N}} \left(d(P_{ref}(x), P_i(y)) \right), \tag{2}$$

where $\mathcal{N}$ is a search window (bigger than the patch). In the (unlikely) case where the minimum is achieved for several values, one is chosen at random. If the minimum is achieved for only one value, we have in particular that $f_{ref}(x) = x$.



(a) (b)

Fig. 1: Instead of fusing values at the same position (a), we define a non-local fusion (b) relying on values from different positions, in order to account for objects' movements and camera shake.

Let us now assume for a moment that we perform the fusion at a single scale (that is, as in Formula (1)). Then the resulting image could be defined as

$$R(x) = \sum_{i=1}^{N} W_i(f_i(x)) I_i(f_i(x)).$$

Of course, as in the classical exposure fusion, defining the resulting image R this way would yield incoherences and seams. Therefore, the non-local fusion is defined, for each level l of a multi-scale pyramid, as :

$$\mathcal{L}_l(R) = \sum_{i=1}^{N} \mathcal{G}_l(\tilde{W}_i) \mathcal{L}_l(\tilde{I}_i), \tag{3}$$

where the weights are defined below and the $\tilde{I}_i$ are defined as $\tilde{I}_i(x) = I_i(f_i(x))$.

The weights $\tilde{W}_i$ are obtained by multiplying the original weights from [7] (accounting for well-exposedness, color saturation and contrast) at position $f_i(x)$ by a weight asserting the similarity between x and $f_i(x)$. Similarly to what is done in the NL-means algorithm [20] we define this similarity weight to be

$$w_i^2(x) = \exp\left(-\frac{d(P_{ref}(x), P_i(f_i(x)))^2}{\sigma^2} \right), \tag{4}$$

where σ is a parameter.

Eventually, the resulting weights is therefore

$$\tilde{W}_i(x) = w_i^1(f_i(x)) \times w_i^2(x).$$

In short, for a given pixel, the fusion is performed by using the most similar pixels in each image of the sequence and by modulating the original weights of the fusion by a similarity score.

C. Computing pixel similarity

In this section, we detail how to compute the distance between patches. For this, images acquired with different exposure settings (e.g. different exposure times) should be comparable. In this paper, we assume that we have access to the linear RAW images of the captured scene. This is not a real limitation in the common case of embedded processing. When these RAW images are not available, other scenarios are possible (such as matching all image histograms to the one of the reference image, in a way similar to [18]), but these will not be investigated in this paper.

In addition to the processed images $I_1, \dots, I_N$ (typically JPEG images after demosaicking, white balance and gamma transform), we consider the corresponding RAW (linear) images $R_1, \dots, R_N$. We assume that these have been acquired with exposure times $t_1, \dots, t_N$. The luminance images can then be computed as $L_i = g t_i (R_i - O)$, where O is the offset of the camera and g the gain (see e.g. [22]). Without loss of generality, we will assume that $g = 1$ and that the offset is known. The comparison of pixels is then carried out in the luminance domain, so that the distance d between two patches

(a) (b)

Fig. 2: Fused images performed by, (a) non-local exposure fusion (NLEF), and (b) classical exposure fusion [7].

is defined as the Sum of Squared Differences (SSD) between the values of the patches in the respective luminance images, that is,

$$d(P_i(x), P_j(x)) = \sum_{k,l=-s}^{s} (L_i(x+k) - L_j(y+k))^2. \quad (5)$$

In the case where the offset of the camera is unknown, or in order to achieve results that are more robust to spatial variability of the gain (photo response non-uniformity, PRNU) or imprecision in the exposure time, the L^2 distance can be replaced by a similarity measure that is invariant to affine transforms of the luminance. Although this aspect will not be developed in this paper, we have observed that the affine invariant distance proposed in [23] is very robust to errors in the luminance conversion.

D. Algorithm details

The complete flowchart of the algorithm is shown in Algorithm 1. A matlab code is also available at this address: [24].

Algorithm 1 Non local exposure fusion

Input: Set of images I_i, Set of raw images R_i, Reference image I_{ref}, patch size $(2s+1)^2$, search window size $(2r+1)^2$, parameter σ.
Output: Fused image I_f.

1: Linear transformation of R_i to obtain the luminance images C_i.
2: **procedure** SEARCH AND FUSION
3: **for** each pixel x **do**
4: Extract search windows $\mathcal{N}_i$ in C_i around x
5: **for** each $\{\mathcal{N}_i\}_{i=1..N}$ **do**
6: Search for the patch $P(f_i(x), C_i)$ most similar to $P(x, C_{ref})$.
7: Compute the exposure weights $w_i^1(x)$ and the similarity weights $w_i^2(x)$.
8: **end for**
9: **end for**
10: Compute the merged weight map $\tilde{W}_i = w_i^1 \cdot w_i^2$ and its Gaussian Pyramid $\{\tilde{W}_i\}_n$
11: Compute the Laplacian Pyramid $\{L_i\}_n$ for $\tilde{I}_i$
12: $\{L'(x)\}_n \leftarrow$ Fuse $\{\tilde{W}_i\}_n$ and $\{L_i\}_n$, for all levels n of the pyramid.
13: **end procedure**
14: Collapse output pyramid: $I_f \leftarrow$ collapse($\{L'\}_n$)

First, a reference image is chosen. For this, we follow the same strategy as in [18] and choose the image having the least number of saturated pixels. Second, and for each pixel in the reference, its best match in image i is chosen using Formula (2) and the similarity measure defined by (5). In order to accelerate the algorithm, the best match is found using the PatchMatch algorithm [25], yielding, for each patch of the reference image, an approximate nearest neighbor (ANN) in each image I_i. The PatchMatch algorithm is run with its default parameters, which in particular implies that the search window $\mathcal{N}$ is the entire image.

The corresponding value of the final image is obtained by fusing the values of the best matches using Formula (3). In this formula, the weights $\tilde{W}_i$ are obtained by multiplying the original weight of [7] by the similarity weight defined by Formula (4). The original weights from [7] depend on three exponents that we all choose to be equal to 1. The parameter σ of the similarity measure (4) is chosen as the quantile of level 5% of the distribution of the distances between patches (separately for each couple L_{ref}, L_i). We empirically found this choice to provide a reasonable compromise in order to simultaneously obtain a result similar to the original algorithm from [7] and efficiently discard moving objects.

A last point that we did not explain is what to do with saturated pixels. We could have chosen a strategy similar to the one of [18]: use only the non-saturated pixels when comparing patches. However, this fails as soon as the saturated region is wider than the patch, which is very common in practice. In order to get more robust results, we use the same strategy as in [21]. For each connected region Ω of saturated pixels, we detect the region from the other images that is the most similar to Ω on its immediate neighborhood (a ring outside Ω). This region is then blended into the reference using Poisson editing [26].

III. Experiments

In all experiments (see Figure 3), we use a search window equal to the full image. The patch size is set to $s = 1$ (that is 3×3 patches).

First, as a sanity check, we compare the result of our method, that we call NLEF (non-local exposure fusion) with the result of the original exposure fusion [7] on a static scene with a moving object, from the database [27], [28].

As we can see in Figure 2, the results are very similar on static parts (everything except one object on the table), but no ghost is produced when using our method.

Then, we provide three comparisons between our method and the recent method from [18], using the code kindly provided by the authors. Since exposure fusion does not aim at producing a real HDR image (a faithful representation of the luminance values), we only provide visual comparisons. Both methods rely on PatchMatch to get patch correspondences, and we use the same parameters for this algorithm in all experiments and for the two approaches (full search over the complete image and patch size of $s = 1$). The first two examples, corresponding to Figures 3(a) and 3(c), are acquired with a Canon 400D and Canon 7D camera, respectively, for which the offsets are 256 and 2046. The example corresponding to Figure 3(e) is from the publicly available database [27] and the offset was evaluated to 0. Images of these experiments, as well as other comparisons, can be seen on the dedicated website [24].

In Figure 3(a), one can see that our method produces more accurate colors. However, the color errors from [18] could probably be solved by taking into account the luminance computed from linear images. Second, one observes that both methods produce accurate details and no ghosts, with occasional artifacts at different positions, as can be checked in the other examples from Figure 3(e). However, our approach is both faster and much simpler, and in particular involves no complex optimization procedure.

IV. Conclusions

In this work, we have introduced a non-local exposure fusion yielding good results in the presence of camera shake and objects' motions. There are several ways this work could be continued. First, some aspects of the algorithm could be further developed. For instance, the saturated parts are handled using a Poisson editing procedure, but this step could probably be integrated in the PatchMatch algorithm, at the propagation step. Second, the method is applicable to other exposure fusion schemes, such as the recent patch-wise method from [9]. Last, similar non-local approaches could be studied in different fusion schemes, for instance to perform focus stacking.

Acknowledgment

The authors thank C. Aguerrebere for her help and advice.

References

[1] P. E. Debevec and J. Malik, "Recovering high dynamic range radiance maps from photographs," in *ACM SIGGRAPH 2008 classes*. ACM, 2008, p. 31.

[2] M. Granados, B. Ajdin, M. Wand, C. Theobalt, H.-P. Seidel, and H. Lensch, "Optimal HDR reconstruction with linear digital cameras," in *Computer Vision and Pattern Recognition (CVPR), IEEE Conference on*. IEEE, 2010, pp. 215–222.

[3] C. Aguerrebere, J. Delon, Y. Gousseau, and P. Musé, "Best algorithms for HDR image generation. a study of performance bounds," *SIAM Journal on Imaging Sciences*, vol. 7, no. 1, pp. 1–34, 2014.

[4] R. Fattal, D. Lischinski, and M. Werman, "Gradient domain high dynamic range compression," in *ACM Transactions on Graphics (TOG)*, vol. 21, no. 3. ACM, 2002, pp. 249–256.

[5] F. Durand and J. Dorsey, "Fast bilateral filtering for the display of high-dynamic-range images," *ACM Transactions on Graphics (TOG)*, vol. 21, no. 3, pp. 257–266, 2002.

[6] R. Mantiuk, K. Myszkowski, and H.-P. Seidel, "A perceptual framework for contrast processing of high dynamic range images," *ACM Transactions on Applied Perception (TAP)*, vol. 3, no. 3, pp. 286–308, 2006.

[7] T. Mertens, J. Kautz, and F. Van Reeth, "Exposure fusion," in *Computer Graphics and Applications, 2007. PG'07. 15th Pacific Conference on*. IEEE, 2007, pp. 382–390.

[8] J. M. Ogden, E. H. Adelson, J. R. Bergen, and P. J. Burt, "Pyramid-based computer graphics," *RCA Engineer*, vol. 30, no. 5, pp. 4–15, 1985.

[9] K. Ma and Z. Wang, "Multi-exposure image fusion: A patch-wise approach," in *IEEE International Conference on Image Processing (ICIP)*, 2015.

[10] M. Song, D. Tao, C. Chen, J. Bu, J. Luo, and C. Zhang, "Probabilistic exposure fusion," *Image Processing, IEEE Transactions on*, vol. 21, no. 1, pp. 341–357, 2012.

[11] K. Kotwal and S. Chaudhuri, "An optimization-based approach to fusion of multi-exposure, low dynamic range images," in *Information Fusion (FUSION), 2011 Proceedings of the 14th International Conference on*. IEEE, 2011, pp. 1–7.

[12] S. Raman and S. Chaudhuri, "Bilateral filter based compositing for variable exposure photography," in *Proceedings of Eurographics*, 2009.

[13] M. Tico, N. Gelfand, and K. Pulli, "Motion-blur-free exposure fusion," in *Image Processing (ICIP), 2010 17th IEEE International Conference on*. IEEE, 2010, pp. 3321–3324.

[14] A. L. Gomez, S. Saravi, and E. A. Edirisinghe, "Multiexposure and multifocus image fusion with multidimensional camera shake compensation," *Optical Engineering*, vol. 52, no. 10, pp. 102 007–102 007, 2013.

[15] J. An, S. H. Lee, J. G. Kuk, and N. I. Cho, "A multi-exposure image fusion algorithm without ghost effect," in *Acoustics, Speech and Signal Processing (ICASSP), 2011 IEEE International Conference on*. IEEE, 2011, pp. 1565–1568.

[16] F. Pece and J. Kautz, "Bitmap movement detection: HDR for dynamic scenes," in *Visual Media Production (CVMP), 2010 Conference on*. IEEE, 2010, pp. 1–8.

[17] J. Zheng, Z. Li, Z. Zhu, S. Wu, and S. Rahardja, "Hybrid patching for a sequence of differently exposed images with moving objects," *Image Processing, IEEE Transactions on*, vol. 22, no. 12, pp. 5190–5201, 2013.

[18] J. Hu, O. Gallo, K. Pulli, and X. Sun, "HDR deghosting: How to deal with saturation?" in *2013 IEEE Conference on Computer Vision and Pattern Recognition*. IEEE, 2013, pp. 1163–1170.

[19] X. Qin, J. Shen, X. Mao, X. Li, and Y. Jia, "Robust match fusion using optimization," *IEEE Transactions on Cybernetics*, 2015.

[20] A. Buades, B. Coll, and J.-M. Morel, "A non-local algorithm for image denoising," in *Computer Vision and Pattern Recognition, 2005. IEEE Computer Society Conference on*, vol. 2. IEEE, 2005, pp. 60–65.

[21] C. Aguerrebere, J. Delon, Y. Gousseau, and P. Muse, "Simultaneous HDR image reconstruction and denoising for dynamic scenes," in *Computational Photography (ICCP), 2013 IEEE International Conference on*. IEEE, 2013, pp. 1–11.

[22] C. Aguerrebere, J. Delon, Y. Gousseau, and P. Musé, "Study of the digital camera acquisition process and statistical modeling of the sensor raw data," *Preprint HAL*.

[23] J. Delon and A. Desolneux, "Stabilization of flicker like effects in image sequences through local contrast correction," *SIAM Journal on Imaging Sciences*, pp. 703–734, 2010.

(a) Our Method - NLEF, (24 sec). (b) Fusion with [18], (143 sec).

(c) Our Method - NLEF, (18 sec). (d) Fusion with [18], (116 sec).

(e) Our Method - NLEF, (68 sec). (f) Fusion with [18], (394 sec).

Fig. 3: Sets of bracketed exposure images and their fusion with our method (NLEF) and the algorithm from [18]. The reference image is framed in a red box and the corresponding processing times are indicated.

[24] "Non local exposure fusion / dedicated website," Telecom ParisTech, 2016. [Online]. Available: http://perso.telecom-paristech.fr/∼gousseau/NLEF
[25] C. Barnes, E. Shechtman, A. Finkelstein, and D. Goldman, "Patchmatch: A randomized correspondence algorithm for structural image editing," *ACM Transactions on Graphics-TOG*, vol. 28, no. 3, p. 24, 2009.
[26] P. Pérez, M. Gangnet, and A. Blake, "Poisson image editing," in *ACM Transactions on Graphics (TOG)*, vol. 22, no. 3. ACM, 2003, pp. 313–318.
[27] "HDR imaging website, international university of sarajevo," 2016. [Online]. Available: http://projects.ius.edu.ba/ComputerGraphics/HDR/
[28] "Empa media technology," 2016. [Online]. Available: www.empamedia.ethz.ch/hdrdatabase/