

GHOST REMOVAL IN HIGH DYNAMIC RANGE IMAGES

*Erum Arif Khan*¹

Ahmet Oğuz Akyüz^{1,2*}

Erik Reinhard^{1,3}

¹University of Central Florida, Orlando, USA

²Max Planck Institute for Biological Cybernetics, Tübingen, Germany

³University of Bristol, Bristol, UK

ABSTRACT

High dynamic range images may be created by capturing multiple images of a scene with varying exposures. Images created in this manner are prone to ghosting artifacts, which appear if there is movement in the scene at the time of capture. This paper describes a novel approach to removing ghosting artifacts from high dynamic range images, without the need for explicit object detection and motion estimation. Weights are computed iteratively and then applied to pixels to determine their contribution to the final image. We use a non-parametric model for the static part of the scene, and a pixel's membership in this model determines its weight. In contrast to previous approaches, our technique does not rely on explicit object detection, tracking, or pixelwise motion estimates. Ghost-free images of different scenes demonstrate the effectiveness of our technique.

Index Terms— Image generation, Pattern recognition

1. INTRODUCTION

The illumination of typical world scenes around us varies over several orders of magnitude. Conventional sensors in image capture devices are only able to capture a limited part of this range. Instead, spectrally weighted radiance of a scene may be captured more accurately by spatially varying pixel exposures [1], using multiple imaging devices, or devices that use special sensors [2]. These devices are expensive and will not be affordable for the average consumer for some years to come [2]. Meanwhile, there exist methods of obtaining high dynamic range (HDR) images using conventional devices [3, 4]. Such techniques require the user to take several images of the same scene at different exposures, and apply a weighted average over these to compute radiance values of the scene.

Multiple exposure techniques have several disadvantages. For instance, if there is any movement in the scene while the exposures are being captured, the moving objects will appear in different locations in these exposures. Therefore, merging corresponding pixel values from different exposures to produce an HDR image will cause a ghosting effect (see Figure 3). As such, existing techniques are only useful for creating HDR images of scenes that are completely still. This is rather restricting as most scenes contain motion. Without a solution to this problem, we are unable to use multiple capture techniques to produce HDR images of scenes that have any moving objects, such as people, animals, and vehicles. This is especially problematic in natural scenes where wind causes dynamic behavior in leaves, trees, flowers, clouds, etc.

One solution to this problem is to track the movement of objects across exposures, and average pixel values according to this movement. For instance, Bogoni [5] estimates an optical flow field

between the different exposures, and warps these exposures such that all scene features are in accurate alignment. Similarly, Kang et al. [6] use gradient-based optical flow between successive frames to compute a dense motion field, which is then used to warp pixels in exposures so that the appropriate values may be averaged together to generate a ghost-free HDR image. Techniques that use motion estimation work as long as the motion estimation is accurate. Currently, there are no approaches to motion estimation that work reliably for all kinds of movement. For instance, such techniques will fail for scenes with effects such as inter-reflections, specularities, and translucency [6]. Even for simple Lambertian objects, motion estimation fails when objects appear in some exposures and not others due to occlusion [7].

One solution that avoids motion estimation altogether, assumes that ghosting occurs in regions where the dynamic range is low enough to be represented accurately by a single exposure [2]. First, regions in the image where ghosting is likely to occur are detected by computing the weighted variance of pixel values for each location in the image, and selecting regions where this variance is above a threshold. Then, for each of these regions, a single exposure is selected from the exposure sequence, and its values are used directly in the HDR image. While this technique removes easily segmentable ghosts well, it will fail to capture ghosting in regions where the object color is similar to the background. Jacobs et al. [8] address this issue by applying the threshold on a measure they derive from entropy, which is invariant to the amount of contrast in the data. Regions that are detected as possible ghost regions in this manner are again replaced with values from single exposures. This solution works well for many scenes, but fails when ghosting occurs in regions where the dynamic range is high. This locally high dynamic range may be due to the presence of features in the scene such as windows, strong highlights, or light sources.

In this work, we present a novel approach to removing ghosts from HDR images. Unlike previous techniques, the proposed approach does not require any intermediate representation, such as optical flow or explicit detection. Instead, we generate an HDR image directly from image information. Thus, our approach is not conditional on the success of some intermediate process, such as optical flow computation or object detection. We make an important deviation from the standard HDR image generation process by iteratively weighting the contribution of each pixel according to its chance of belonging to the static part of the scene, (henceforth referred to as background), as well as its chance of being correctly exposed. We use a non-parametric model of the background, which enables us to compute a pixel's membership in the model, and therefore its weight. Since the model is non-parametric, we do not impose any restrictions on the background. The only assumption we make is that the exposure sequence predominantly captures the background, so that in any local region in image space, the number of pixels that capture the

*Thanks to the i²Lab for funding.

background is significantly greater than the pixels that capture the object. Given this assumption, the neighborhood of a pixel in image space may serve as a reasonable representation of the background, and the probability of the pixel's membership in this neighborhood may serve as the weight for that pixel.

The remainder of this paper is organized as follows: We describe the standard HDR image generation process in Section 2, and explain how we improve this process to remove ghosting artifacts. In Section 3, we present images in which ghosts have been removed using our technique, and compare these with results from previous work.

2. ITERATIVE GHOST REMOVAL

A set of exposures of the same scene may be used to obtain the camera response function $g(\cdot)$, of the capturing device, which is then applied to the exposures to convert them to radiance maps. The individual radiance maps, normalized by their exposure time, are averaged together to generate an HDR image [2]:

$$E(i, j) = \frac{\sum_{r=1}^R w(Z_r(i, j)) \left(\frac{g^{-1}(Z_r(i, j))}{\Delta t_r} \right)}{\sum_{r=1}^R w(Z_r(i, j))} \quad (1)$$

where $E(i, j)$ is the radiance at location (i, j) in the image, r denotes the exposure, R is the total number of exposures, $Z_r(i, j)$ is the value of the pixel at location (i, j) in the r^{th} exposure, (ranging from 0 to 255) and Δt_r is the exposure time of the r^{th} exposure. Note that the above is a weighted average of the radiance in each exposure. $w(\cdot)$ is typically chosen to diminish the contribution of pixel values that are under- or over-exposed. This weight is small for pixel values that are close to 0 and 255. The above equation is typically evaluated thrice for each pixel, for the red, green, and blue channels [2, 8]. However, since the three channels are correlated for typical natural images, a single weight for a pixel is necessary to preserve the original information [5].

We make an important improvement to this process by computing weights that are determined not only by their probability of being correctly exposed, but also by the probability that they capture the background. Unlike the first attribute, which is determined by the pixel's value, there is no existing method to find the probability that a pixel captures part of a moving object. The object may be of any color, shape, or size, and its movement may be slow or fast, rigid or non-rigid. These features make it difficult to model the background, and to assign weights to pixels accordingly. Given a set of R exposures, each of size $I \times J$, our objective is to compute a set of $I \times J \times R$ weights, that will be used to determine the contribution of each pixel in the exposure sequence.

We equate this problem to that of finding the probability that a pixel belongs to the background. We use a non-parametric estimation scheme to determine this probability so as to impose as few restrictions on our data as possible. Such estimation schemes give a high probability of membership to elements that lie in densely populated regions of the distribution's feature space. In particular, we use the kernel density estimator to compute this probability [9, 10]. To find the probability that a vector \mathbf{x} belongs to a class F , an estimate can be computed:

$$P(\mathbf{x}|F) = M^{-1} \sum_{m=1}^M K_{\mathbf{H}}(\mathbf{x} - \mathbf{y}_m), \quad (2)$$

where M is the number of vectors in the class, \mathbf{y}_m is the m^{th} vector in the class, \mathbf{H} is a symmetric, positive definite, $d \times d$ bandwidth matrix, and

$$K_{\mathbf{H}}(\mathbf{x}) = |\mathbf{H}|^{-\frac{1}{2}} K(\mathbf{H}^{-\frac{1}{2}} \mathbf{x}), \quad (3)$$

where K is a d -variate kernel function. The d -variate Gaussian density is a common choice for the kernel K :

$$K_{\mathbf{H}}(\mathbf{x}) = |\mathbf{H}|^{-\frac{1}{2}} (2\pi)^{-\frac{d}{2}} \exp\left(-\frac{1}{2} \mathbf{x}^T \mathbf{H}^{-1} \mathbf{x}\right). \quad (4)$$

and we use this kernel function in our implementation.

We represent each pixel by a vector in a feature space such that $\mathbf{x}_{ijr} \in \mathbb{R}^5$, $i = 1, 2, \dots, I$, $j = 1, 2, \dots, J$, $r = 1, 2, \dots, R$. Three dimensions represent color, and two represent the location of the pixel in image space. We use the decorrelated $L\alpha\beta$ color space to represent each pixel's color values in the feature space, as this color space is perceptually uniform to a first approximation [11]. Our use of spatial information exploits dependencies between proximal pixels.

For a vector \mathbf{x}_{ijr} , the background is represented by a $p \times q \times s$ neighborhood, \mathbf{N} , around the vector. Thus, for each \mathbf{x}_{ijr} , $F = \{\mathbf{y}_{pqs} | (p, q, s) \in \mathbf{N}(\mathbf{x}_{ijr}), (p, q) \neq (i, j), \text{ and } s = 1, 2, \dots, R\}$, (see Figure 1). This representation of the background remains the same for all the r pixels at location (i, j) .

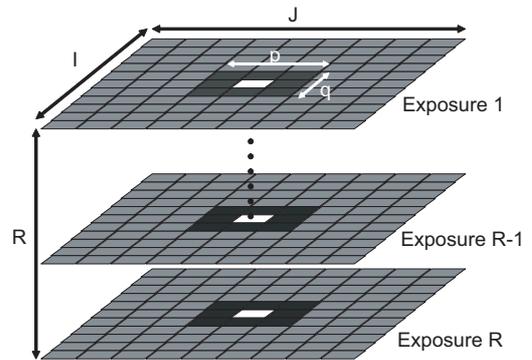


Fig. 1. The arrays represent images of the same scene taken at R different exposures. For each vector \mathbf{x}_{ijr} (the white elements shown in the centers of the arrays), \mathbf{y}_{pqs} consists of a $p \times q$ neighborhood around (i, j) in all the R exposures (shown as the dark gray region around the center elements). p and q equal three in the above diagram. Note that the representation of the background is identical for all pixels at location (i, j) .

Thus far, we have assumed that all vectors \mathbf{y}_{pqs} are equally a part of the background. In practice, we know that many of these vectors represent pixels that are under- or over-exposed, and therefore do not represent the background well. These should not be considered a part of the background. Also, some vectors represent pixels that capture the moving object, and are also not a valid part of the background. Initially, while we do not know which vectors represent the moving object, we may reduce the effect of under- and over-exposed vectors by weighting their contribution in kernel density estimation [12]. The weights assigned to each vector \mathbf{y}_{pqs} are based on a simple hat function [2] shown in Figure 2. This function is computed as follows:

$$w(Z) = 1 - \left(2 \cdot \frac{Z}{255} - 1 \right)^{12}, \quad (5)$$

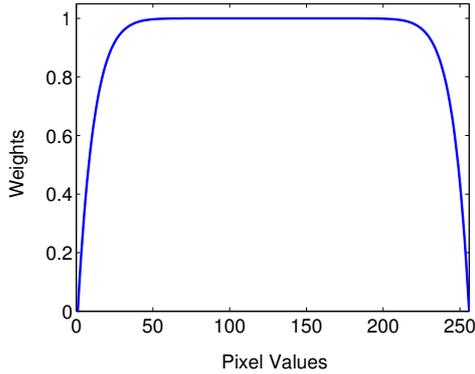


Fig. 2. Initial weights assigned to each pixel as a function of pixel value.

where Z represents pixel values.

This produces three weights for each pixel. The final weight, w_{pqs} , is an average of these three. Using these weights, the probability that a vector \mathbf{x}_{ijr} belongs to the background, now becomes:

$$P(\mathbf{x}_{ijr}|F) = \frac{\sum_{p,q,s \in N(\mathbf{x}_{ijr})} w_{pqs} K_{\mathbf{H}}(\mathbf{x}_{ijr} - \mathbf{y}_{pqs})}{\sum_{p,q,s \in N(\mathbf{x}_{ijr})} w_{pqs}} \quad (6)$$

If our assumption holds and the neighborhood around each pixel predominantly represents the background, vectors which capture the moving object will get lower probabilities than vectors that capture the background. Therefore, once the probabilities have been computed for each vector \mathbf{x}_{ijr} , these can be used as weights of the corresponding pixels in HDR image generation. An HDR image created using these weights will show diminished ghosting as compared to an image generated using the initial weights, which are determined only by absolute pixel values.

Now that we have a better set of weights for each pixel, we can repeat the above process of kernel density estimation, this time the initial weights of the vectors \mathbf{y}_{pqs} will be the ones computed in the previous iteration of kernel density estimation as shown below:

$$w_{pqs,t+1} = w(Z_s(p, q)) \cdot P(\mathbf{x}_{pqs}|F) \quad (7)$$

where $w(Z_s(p, q))$ is the initial weight for the pixel at location (p, q) with exposure s , $P(\mathbf{x}_{pqs}|F)$ is the weight of the pixel as determined by kernel density estimation, and $w_{pqs,t+1}$ is the weight that will be used in kernel density estimation in iteration $t + 1$. As before, we want to diminish the probability of pixels that are under- or over-exposed, and we multiply the newly computed weights with the initial weights obtained from the hat function before using them in density estimation. Now, even in cases where the distribution around a pixel is predominantly composed of under- or over-exposed pixels, they do not get high weights, as their initial weights are low.

3. RESULTS AND CONCLUSIONS

We used a Nikon D2X camera to capture all the exposures required to create HDR images. We used completely manual settings, doubling only the exposure time between consecutive captures of the same scene. A tripod was used to keep the camera stable during image capture. All HDR images shown in this paper have been tonemapped with the Photographic tonemapping operator [13]. Our



Fig. 3. The left column shows the second, fifth, and seventh exposure of a sequence in which people are walking across the scene. The right column shows tonemapped HDR images that are generated using (top) weights determined only by absolute pixel values, (middle) weights that were estimated with a single iteration of our algorithm, (bottom) weights that were estimated after nine iterations of our algorithm.

algorithm requires the user to specify the size of the neighborhood around a pixel, which we have kept at a constant 3×3 for all the results that we show in this section. The user is also required to specify the matrix $\mathbf{H} = \text{diag}(h_x, h_y, h_L, h_\alpha, h_\beta)$, which we kept as the identity matrix for our runs. Figure 3 shows our algorithm applied to a scene in which people are walking by. In this sequence, the amount of movement is large, and ghosting occurs in most of the image as a result. In Figure 4, ghosting is more localized, but involves the movement of a highlight (on the edge of the green book), which implies that the ghosting region has high dynamic range. Successive iterations of our algorithm remove both instances of ghosting from the HDR images.

We have compared our results with the approach described in [2], using *Photosphere*. Figure 5 shows results generated from a set of nine exposures of a flame. Due to air movement, and the high contrast of a flame with its background, this is a notoriously difficult scene to capture. The method in [2] does not correctly identify ghosting artifacts in this scene. In addition, replacing parts of a flame scene with a single exposure, as proposed in [2], would yield a low dynamic range result in an area of the image that is inherently high dynamic range. As shown, our method overcomes these limitations, and produces a plausible result.

In the future, we intend to study the application of our algorithm in creating artifact free HDR images from exposures captured with a hand-held (unstable) camera, as well as exposures that contain noise.

4. REFERENCES

- [1] Mitsunaga T. and Nayar S. K., “High dynamic range imaging: Spatially varying pixel exposures,” in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE, 2000, vol. I, pp. 472–479.
- [2] Reinhard E., Ward G., Pattanaik S., and Debevec P., *High Dy-*

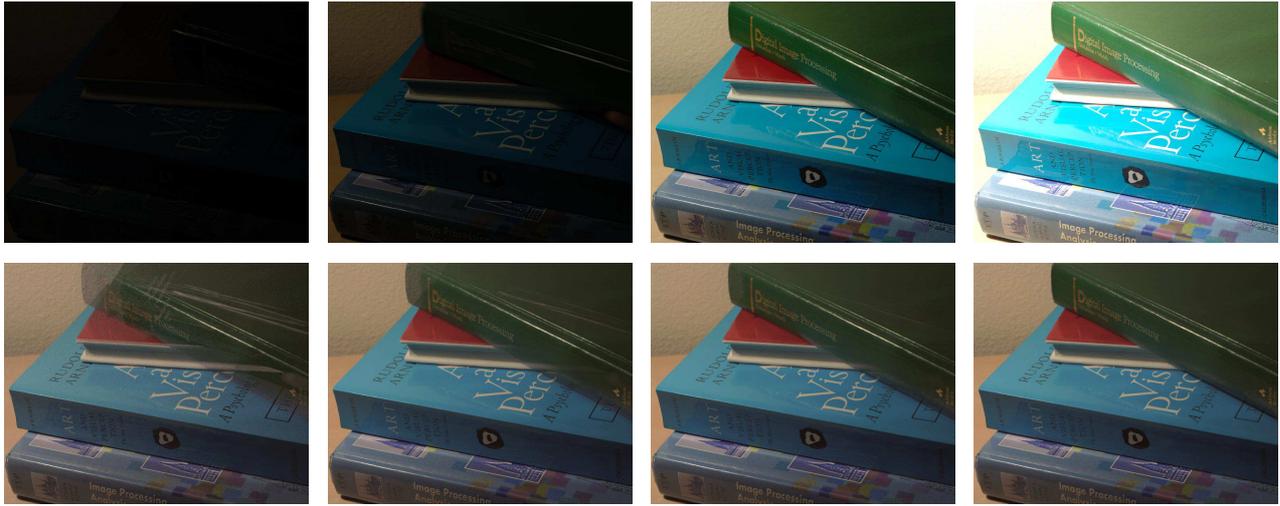


Fig. 4. The top row shows the second, fourth, sixth and eighth exposures of a sequence in which a book is being placed upon a stack of books. The bottom row (left to right) shows tonemapped HDR images that were generated using weights determined only by absolute pixel values, and weights that were estimated with one, two, and fourteen iterations of our algorithm.

Dynamic Range Imaging: Acquisition, Display, and Image-Based Lighting, Morgan Kaufmann, 2005.

- [3] Debevec P. E and Malik J., “Recovering high dynamic range radiance maps from photographs,” in *SIGGRAPH 97 Conference Proceedings*, August 1997, Annual Conference Series, pp. 369–378.
- [4] Mitsunaga T. and Nayar S. K., “Radiometric self-calibration,” in *IEEE Computer Society Conference on Computer Vision and Pattern Recognition*. IEEE, 1999, vol. I, pp. 374–380.
- [5] Bogoni L., “Extending dynamic range of monochrome and color images through fusion,” in *International Conference on Pattern Recognition*, 2000, vol. 3, pp. 7–12.
- [6] Kang S. B., Uyttendaele M., Winder S., and Szeliski R., “High dynamic range video,” in *ACM Transactions on Graphics*, 2003, vol. 22.
- [7] Wang J. Y. A. and Adelson E., “Representing moving images with layers,” in *IEEE Transactions on Image Processing*, May 1994, vol. 3, pp. 369–378.
- [8] Jacobs K., Ward G., and Loscos C., “Automatic hdri generation of dynamic environments,” in *Siggraph technical sketch: SIGGRAPH, 2005*. ACM.
- [9] Parzen E., “On estimation of a probability density and mode,” in *Annals of Math. Statistics*, 1962.
- [10] Rosenblatt M., “Remarks on some nonparametric estimates of density functions,” in *Annals of Math. Statistics*, 1956.
- [11] Ruderman D. L., Cronin T. W., and Chiao C-C., “Statistics of cone responses to natural images: Implications for visual coding,” in *Journal of the Optical Society of America A*, 1998, vol. 15, pp. 2036–2045.
- [12] Goerlich F. J., “Weighted samples, kernel density estimators and convergence,” in *Empirical Economics*, 2003, vol. 28, pp. 335–351.
- [13] Reinhard E., Stark M., Shirley P., and Ferwerda J., “Photographic tone reproduction for digital images,” in *ACM Transactions on Graphics*, 2002, vol. 21, pp. 267–276.

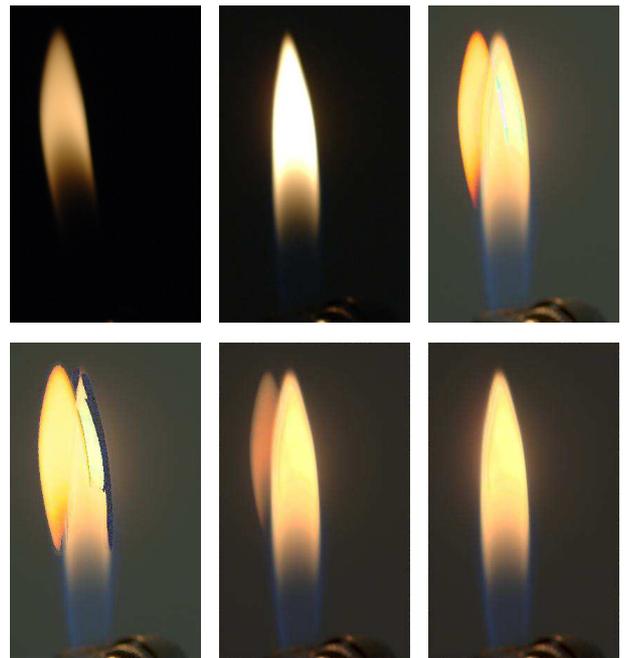


Fig. 5. The first two images in the top row show two of the nine exposures used to generate the HDR images. The third image shows the tonemapped HDR image created without using any ghost removal algorithm. The bottom row shows tonemapped HDR images that were generated using (left) the ghost removal technique as proposed by [2], (middle) two iterations of our algorithm, and (right) ten iterations of our algorithm.