Technion, Israel Institute of Technology
Computer Science Department
Geometric Image Processing Laboratory

Fusion of Differently Exposed Images

Final Project Report

Submitted by: Ron Rubinstein
Supervisor: Alexander Brook

October 2004

1 Abstract

Real-world scenes often exhibit very high dynamic ranges, which cannot be captured by a sensing device in a single shot. Nor can an imaging device successfully reproduce such a scene. To overcome this, these scenes must be represented as a series of differently exposed images, representing different sub-bands of the complete dynamic range. Unfortunately, this representation is unsuitable for many tasks, such as human interpretation or computerized analysis. In this paper we propose an effective algorithm for fusing such image sequences to a single low dynamic range image, which may be displayed on a standard device. The fusion process accumulates all the details, which initially span many images, in a single image. Our algorithm is simple, highly stable and computationally efficient. It may be applied to both gray-scale and full-color images. We analyze the method, and provide a set of parameters which enable to produce very good results in a fully-automated process. Real-world results are presented to demonstrate the method's performance.

2 Introduction

The full dynamic range of a real-world scene is generally much larger than that of the sensing devices used to capture it, as well as the imaging devices used to reproduce it. When a large dynamic range must be processed using a limited-range device, one is forced to split the dynamic range into several smaller "strips", and handle each of them separately. This process produces a sequence of images of the same scene, covering different portions of the dynamic range. When *capturing* a high dynamic range (HDR) scene, the sequence is obtained by varying the exposure settings of the sensor. When *reproducing* an HDR scene, the sequence is obtained by splitting the full range of the image into several sub-ranges, and displaying each separately. Indeed, in both cases we turn to a solution in the form of a *variable-exposure image sequence*.

The major drawback of variable-exposure image sequences is their usability. Since details span several images, a human viewer will find it difficult to study and interpret the scene. A computer program could do better, implementing specific techniques for handling such representations; however this can be a tedious task, since most visual applications are designed to handle single images containing all the details. It is therefore imperative to develop techniques for merging (fusing) such image sequences into single, more informative, low dynamic range (LDR) images, maintaining all the details at the expense of brightness accuracy.

In this project, we propose a fusion algorithm based on Laplacian pyramid representations [1, 2]. The Laplacian pyramid is an over-complete, multi-scale representation, in which each level roughly corresponds to a unique frequency band. Laplacian pyramids have several advantages over other multi-scale representations (e.g. wavelets) — they are stable under affine transformations (such as translation, rotation and scaling), and they are computationally easy. We show that a simple maximization process in the Laplacian pyramid domain, provides an effective way to produce an informative high-detail image given an aligned sequence of differently exposed images. Our method may be applied to both gray-scale as well as full-color images.

3 Background

Sequences of differently exposed images appear in many contexts. They are most often produced when capturing a real-world scene, as sensing devices may only capture a limited band of the scene's entire dynamic range at any given moment. When a larger dynamic range is required, we are compelled to capture the scene as a sequence of images, using varying exposure settings.

More recently, these sequences have been related to the evolving field of high dynamic range (HDR) imagery. HDR images are basically images in which the range of representable intensities is larger than the standard 8-bit-per-channel (256-level) range. HDR images are a natural extension of current standard image representations, however their use is limited by the fact that they cannot be accurately reproduced by any current imaging device. In fact, the typical dynamic range of an HDR image is at least *2-3 orders of magnitude* larger than that reproducible by the most capable state-of-the-art current imaging devices. Hence, in order to display an HDR image one must first substantially compress its dynamic range; this is a non-trivial task which has been extensively discussed in the literature [3]). Alternatively, the HDR image may be decomposed into a sequence of limited-range images; such a decomposition is a relatively simple task, but viewing the image this way is undoubtedly less convenient.

The goal of this project has been to develop and implement a method for fusing variable-exposure image sequences. Given such a sequence (of either gray-scale or full-color images), our program produces a single displayable LDR image of the scene. Brightness information naturally cannot be maintained in the fused image, however all scene details are preserved, allowing for convenient examination, interpretation and analysis. Possible uses for our method include

- Producing full-detail images of high dynamic range scenes using low dynamic range sensing devices
- Viewing HDR images given their decomposition to a sequence of limited-range images.

To date, the task of fusing variable-exposure image sequences to a single LDR image has not been thoroughly discussed in the literature. In most papers, the more general fusion problem is considered, where the source images may originate from different sensors, and may exhibit polarity changes (negation). For this general case, the proposed solutions are often complex, requiring statistical analysis, machine learning methods or global optimization algorithms. Our approach is unique in that it is specialized for the task of fusing variable-exposure sequences, and consequently is significantly simpler, computationally efficient, dependant on a minimal set of parameters, and fine-tuned to produce good results even with the default settings (allowing for a fully automatic process).

4 Previous Work

The field of fusing multiple images of the same scene has received much attention in the past two decades (although color fusion has received surprisingly little attention). The idea of fusing images in the Laplacian pyramid domain was first introduced by Burt and Adelson in [1, 2], and among the applications

which they mention are image stitching and focus enhancement. Burt and Kolczynski [5] continued this work, and developed a more advanced method for fusing general multi-sensor images using gradient pyramids. Their method incorporates a sophisticated selection rule in the pyramid domain, based on pixel similarity and salience measures which they introduce.

The late 90's have seen rapid advancement in wavelet decomposition theory. These advances have enabled the development of wavelet-based fusion methods, similar in nature to the basic scheme pioneered by Burt and Adelson. In 1997, Rockinger [7] introduced a method utilizing the stationary (undecimated) wavelet transform, employing a choose-max merging process similar to that applied in the Laplacian domain. The undecimated transform provided for translation-invariance, which is a desired quality of the fusion result. Zhang and Blum [8] proposed combining wavelet-based image fusion with higher-level image analysis methods. They use image segmentation and labeling techniques in order to improve the selection process in the wavelet domain.

Sharma and Pavel [6] introduce an entirely different approach. Their method is designed for fusing continuous imagery (e.g. video), obtained from multiple sensors[1]. Sharma and Pavel adopt a statistical approach based on optimal Baysian estimation. They model the intensity transformation between the sensor outputs by an affine transformation with added noise, and attempt to estimate the parameters of this transformation (for each sensor individually) using Maximum A-Posteriory (MAP) estimation. Their method operates on the continuous data, and thus is not appropriate for a fixed sequence of still images. Their initial parameter estimates are improved over time as the data becomes available.

Finally, the process of combining differently exposed images has been thoroughly discussed by Debevec and Malik [10], as well as by Mann and Picard [9]. Both develop methods for reconstructing the *true* HDR radiance map of a scene, given the set of differently exposed images. Debevec and Malik assume the exposure time for each of the images in the sequence is known; they consider the process in which these images were formed, and formulate the problem as a quadratic optimization problem. Mann and Picard propose a different method for reconstructing the radiance map, one which determines the response function without any prior data, assuming some parametric form for this function. Both works are important contributions to the theory of variable-exposure image fusion, however they are not immediately applicable when an LDR result is required.

5 Theoretical Background

5.1 Variable-Exposure Image Sequences

A *variable-exposure image sequence* is an aligned sequence of images of a common scene, representing different bands of its dynamic range. We do not assume here any specific model for the process forming the sequence, however we do require that there be no polarity reversals between the images. Figure 1 presents a sample image sequence. The leftmost image is the most under-exposed image, and contains details of the brightest objects in the scene (lamps). The rightmost

[1] For instance, an array of sensors guiding an aircraft.

image is the most over-exposed, and contains detail of the darker objects in the scene (furniture and walls).

Figure 1: A variable-exposure image sequence.

5.2 Laplacian Pyramids

The Laplacian pyramid representation was introduced by Burt and Adelson [1] in 1983, and is accepted today as a fundamental tool in image processing. It is a straightforward, intuitive band-pass decomposition which is simple to implement and computationally efficient. To implement Laplacian pyramid decomposition, one must first define two elementary scaling operations, usually referred to as *shrink* and *expand*. The *shrink* operation applies a low-pass filter to the image and downsamples it by a factor of two. The *expand* operation employs a predefined interpolation method and upsamples the image by a factor of two. Given these two operations, the Laplacian pyramid is obtained via the following two-step process:

1. Generate the *Gaussian pyramid* of the image. The Gaussian pyramid is basically a series of copies $\{G_1, G_2, \ldots, G_k\}$ of the original image I at different scales. It is obtained by setting $G_1 = I$, and iteratively applying $G_{i+1} = shrink(G_i)$.

2. Generate the *Laplacian pyramid* of the image. The Laplacian pyramid $\{L_1, L_2, \ldots, L_k\}$ is obtained by backward-processing the Gaussian pyramid, setting $L_k = G_k$ and iteratively applying $L_i = G_i - expand(G_{i+1})$.

The inverse transform, for recovering an image from its Laplacian pyramid, is computed by setting $G_k = L_k$, and iteratively computing $G_{i-1} = L_{i-1} + expand(G_i)$. The image is then given by $I = G_1$.

Assuming the low-pass filter in the *shrink* operation roughly eliminates the higher half of the image frequencies, the Gaussian pyramid is simply a series of scaled-down versions of the original image, each G_{i+1} representing the lower half of the frequencies of its predecessor G_i. For the *expand* operation, we assume a simple interpolation method (such as linear or cubic interpolation), such that the upscaling process roughly preserves the frequency composition of the image, and introduces minimal artificial high frequencies. Consequently, applying $L_i = G_i - expand(G_{i+1})$ essentially removes the lower half of the frequency spectrum from G_i, retaining only the higher frequency band. The resulting pyramid $\{L_1, L_2, \ldots, L_k\}$ is therefore a form of band-pass decomposition of the image, where L_1 represents the lowest part of the spectrum, and each L_{i+1} represents a higher frequency band than its predecessor. L_1 is often referred to as the *approximation* level in the pyramid, and the remaining L_is are the *detail* levels.

The Laplacian pyramid transform is specifically designed for capturing image details over multiple scales. It is obviously an over-complete transformation, and

as opposed to the wavelet decomposition, for instance, each band-pass level is sampled at precisely its Nyquist frequency — making it less sensitive to noise. Also, one may easily verify that the Laplacian pyramid transform is invariant under affine transformations. All these properties make the Laplacian pyramid transform a well-suited representation for the task at hand.

6 Fusing Variable-Exposure Image Sequences

We now describe the proposed fusion algorithm for variable-exposure sequences. The algorithm is based on fusion in the Laplacian pyramid domain. The core of this algorithm is a simple maximization process in the Laplacian domain, for this process there are a number of parameters which need be considered, and these are discussed. The result of the Laplacian-domain fusion is post-processed to maximize its contrast and detail visibility.

6.1 Algorithm Outline

Given the variable-exposure sequence $\{I_1, I_2, \ldots, I_n\}$, the entire fusion algorithm is outlined by the following steps:

1. Generate a Laplacian pyramid $\mathcal{L}_i$ for each of the images I_i.
2. Merge the pyramids $\{\mathcal{L}_i\}$ by taking the maximum at each pixel of the pyramid, obtaining the Laplacian pyramid representation $\mathcal{L}$ of the fusion result.
3. Reconstruct the fusion result I from its Laplacian pyramid representation.
4. Normalize the dynamic range of the result so that it resides within the range of [0,1], and apply additional post-processing techniques as necessary.

6.2 Method Details

The first step of the algorithm is to generate Laplacian pyramids for each of the input images (when color images are involved, the Laplacian pyramid is generated independently for each channel). Laplacian pyramid decomposition introduces two parameters which may affect the fusion result. First, an appropriate low-pass filter must be selected. Second, the number of levels in the pyramid should be chosen; this determines the size of the low-frequency (approximation) level. We discuss these parameters in the next section.

The second step is the actual fusion process. Here we must define some selection rule for determining the value of each pixel in the result pyramid. The selection rule may generally be different for the approximation level and for the detail levels. For the detail levels, each pixel in the result pyramid is determined by selecting the maximum among all the corresponding pixels in the input pyramids. Maximization is the most natural operation for the detail levels as we want to maximize the detail in the output image. For single-channel images, the maximum is taken in absolute value (i.e. ignoring sign). For color images, each pixel in the pyramid has three values, and in this case, the pixel with largest

norm is selected; we use the L_2 norm in our implementation. For the approximation level, different fusion approaches can be used. The most straightforward approach is averaging; since we assume this level contains little image details, averaging will not discard any valuable information. The averaging approach works well in most cases, and should usually be used. For specific cases, other selection approaches may be considered as well, in order to improve the visual quality of the result. For instance, one may apply the same maximization process on the approximation level as is done for the detail levels; this will produce a brighter result, and for some images, this approach produces nicer-looking images. Alternatively, for color images, one may consider selecting the most saturated pixel, or use a weighted combination of brightness and saturation.

Once the pyramids are merged, the fusion result is reconstructed from its pyramid representation. For the reconstruction, we implemented a standard *expand* method which multiplies the image intensity by 4 (to preserve average pixel energy in the process), upsamples it by inserting zeros between the image pixels, and smoothes the result using the same low-pass filter used for the downsampling. Due to the nature of the fusion process, the resulting image may have a dynamic range extending beyond the valid [0,1] range. Therefore the next step is to linearly compress the dynamic range of the result into the valid range. For gray-scale image, the intensity transformation is simply given by $(x-x_{min})/(x_{max}-x_{min})$. For color images, we take the minimum and maximum over all channels. After this normalization, the fusion process is essentially complete. Additional post-processing techniques may be applied at this stage, in order to improve the visual quality of the image and accentuate its details; some such techniques are provided later.

6.3 Parameters of the Fusion Process

Two parameters which affect the result of the fusion process are the choice of a smoothing filter, and the number of levels in the pyramid. It is recommended that the smoothing filter be of odd length, so that the kernel has a defined central pixel (this ensures that the filtering result is not biased); using an even-lengthed filter tends to cause artificial shadowing effects in the result. For best performance, the smoothing filter should be chosen such that the filtering process approximately removes the higher half of the frequency spectrum, maintaining the lower half. Our implementation uses periodic filtering with a Gaussian smoothing filter, and good results have been obtained using $1.2 < \sigma < 2.5$. We suggest a default value of $\sigma = 1.7$.

The number of pyramid levels also has an effect on the quality of the fusion. The number of levels should be large enough so that the approximation level contains little significant detail, or else the averaging process could cause loss of information. This may be ensured, for instance, by extending the pyramid height until the size of the approximation level reaches one pixel. However, using too many pyramid levels may have an inverse effect on the global vividness of the result and its color accuracy. A common approach is to set a minimal size for the approximation level. This approach assumes that the maximum detail size is relative to the size of the entire image (e.g. if the largest detail information is assumed to occupy no more than 25% of the image, we may set the approximation level size to approximately 2×2 pixels). Our implementation uses a default minimal size of 9×9 pixels.

6.4 Post Processing

Once the result is normalized, it may be post-processed in order to improve its detail visibility and enhance its contrast. In our implementation we use a two-step post processing stage:

1. Intensity stretching. The dynamic range of the fused image is expanded by linearly stretching the intensity values within the [0,1] range. This process saturates a predefined percentage of the pixels at the low and high intensity values. For color images, we treat all the channels as a single channel and stretch all values simultaneously. All examples in this report use the default value of 1% saturation.

2. Adaptive Histogram equalization. Following the intensity stretching, it is beneficial to apply adaptive histogram equalization on the result, in order to improve its contrast and sharpen details. This step is not essential, however visibility of the details may be further enhanced by this process. Adaptive histogram equalization individually equalizes small regions of the image, and combines them using bilinear interpolation. The local equalization step is contrast-limited in order to avoid noise amplification in homogenous regions (we use a default clip limit of 0.002, as defined by Matlab's ADAPTHISTEQ function). For color images, the result is converted to YIQ and the equalization is applied to the intensity (Y) channel.

7 Extension: Motion Compensation

The images in the input sequence may not always be fully aligned (for instance when the sequence is shot using a hand-held device). In these cases the fusion will result in a "fuzzy" or "blurry" image, since corresponding edges in the different images do not overlap, and these are accumulated in the merging process. Obviously, it is essential to align all the images prior to fusion. As part of this project, we have developed and implemented a preprocessing unit which detects the global motion between the images, and compensates for this motion.

We assume that the motion between the images may be approximated by a global translation-and-rotation model. For small movements, the *Lucas-Kanade* sub-pixel motion estimation algorithm [4] is considered highly effective. In its standard form, however, it is unsuitable for handling exposure differences; we have therefore developed a version of this algorithm which is tolerant to varying exposure conditions. We now briefly describe this algorithm and its adaptation to differently exposed images.

The basic Lucas-Kanade algorithm approximates the parameters of the global motion, by assuming a first-order linear approximation in these motion parameters (which are assumed to be small). This gives the problem the form of a simple over-specified set of linear equations (in the motion parameters), which may be easily solved by minimizing the mean squared error. The accuracy of the result is then increased by warping one of the images according to the motion estimate, and repeating the process on the new pair of images. This is the iterative form of the algorithm, and it produces a series of motion estimates which quickly converge to the correct value.

The iterative Lucas-Kanade algorithm may be improved so that it is capable of estimating larger movements. This is accomplished by the multi-resolution version of the algorithm. In its multi-resolution version, two Gaussian pyramids are constructed for the two input image. The algorithm is first applied to the coarsest (smallest) pyramid level, producing the large-scale motion estimate, and this result is used as the initial estimate for the motion estimation process of the next pyramid level. The algorithm continues along the pyramid levels, improving its estimate until it reaches the finest pyramid level. The result obtained by this level is the final output of the algorithm.

In order to adapt this well-established algorithm to the case of differently exposed images, we replace the Gaussian pyramid used in the multi-resolution version with a Laplacian pyramid. The motion estimation process is not applied to the approximation level of the pyramid but to the detail levels only, beginning with the smallest one. The Laplacian pyramid is much better suited for handling differently exposed images, since it is indifferent to global energy differences, and is focused on representing image details. Since all images describe the exact same scene, we assume the details are (more or less) common to all images. Naturally, when the exposure differences are large, this assumption may not hold. For this reason we apply the motion detection algorithm only to pairs of sequential images in the sequence. Of course, if the exposure differences are still large, details may change too drastically for the algorithm to converge; unfortunately, in this case the exact motion will be extremely hard to determine (by any algorithm), due to the lack of common image features to guide the process.

Finally, the Lucas-Kanade algorithm was refined to provide better handling of color images. Given a pair of color images, one can generally convert them to grayscale and apply L-K on the grayscale pair. This approach works well most of the time, however it is not optimal since it disregards considerable information. To increase the accuracy of the motion estimation in color images, we have implemented the option of applying the Lucas-Kanade method to all three color channels simultaneously. This is a simple generalization, which is done by replacing the over-specified set of linear constrains which was previously obtained from a single channel, with a three-fold larger set obtained from all three channels. This modification considerably increases the run-time of the algorithm, but may in some cases improve the accuracy of the motion estimation.

8 Results

We have tested our method on many sequences of images, and some sample results are presented in figures 2, 3, and 4. All results were produced automatically, using the recommended default settings. As can be seen, the resulting images contain all the details appearing in the input sequences, with no added artifacts or visible noise amplification. Color information is not entirely preserved, but its quality is acceptable.

The performance of the algorithm was also evaluated using four standard test pairs, as presented in figure 5. The left and center images in each row form a pair of input images, and the right image is the fusion result. The first two rows show the test pair 'circles'. The two circles in the second row are shifted by one pixel to the right relative to those in the first row, and this test

demonstrates the translation-invariance of the method. The next two rows show the test pair 'lines'. These two rows differ in the orientation of the lines in the input images, and are intended to demonstrate the orientation-invariance of the fusion method. The fifth row shows the test pair 'ball' (note the pattern on the middle circle). It demonstrates the behavior of the algorithm along object borders, as well as the performance of the algorithm under varying detail scales (large objects vs. delicate patterns). The last row shows the test pair 'squares', and is intended to demonstrate the algorithm behavior for overlapping objects. It should be noted that post-processing was disabled for all tests.

As can be seen from these tests, the algorithm is stable under transformations of the input, and produces no visible artifacts or edges in the result. As expected, brightness accuracy is lost during the fusion process, however all the details in the input images are maintained.

Figure 2: Image sequence and fusion result ("living room").

9 Summary and Conclusions

We have developed and implemented a simple, yet powerful method for fusing variable-exposure image sequences. The method features high stability, accuracy, and is computationally efficient. It may be applied to both gray-scale and color images. We have analyzed the fusion process, discussed its various parameters, and have presented a set of default values which allow the method

Figure 3: Image sequence and fusion result ("igloo").

Figure 4: Image sequence and fusion result ("cabin").

to consistently produce good results with no additional intervention. In addition, we have developed and implemented an effective motion compensation preprocessor which may be used for fusing non-aligned sequences. Out motion detection algorithm is capable of handling exposure changes, and can take advantage of color information for increased accuracy. Finally, some results for real-world sequences are provided, demonstrating the abilities of our method. We conclude that the method is highly effective and well-suited for fusing differently exposed images.

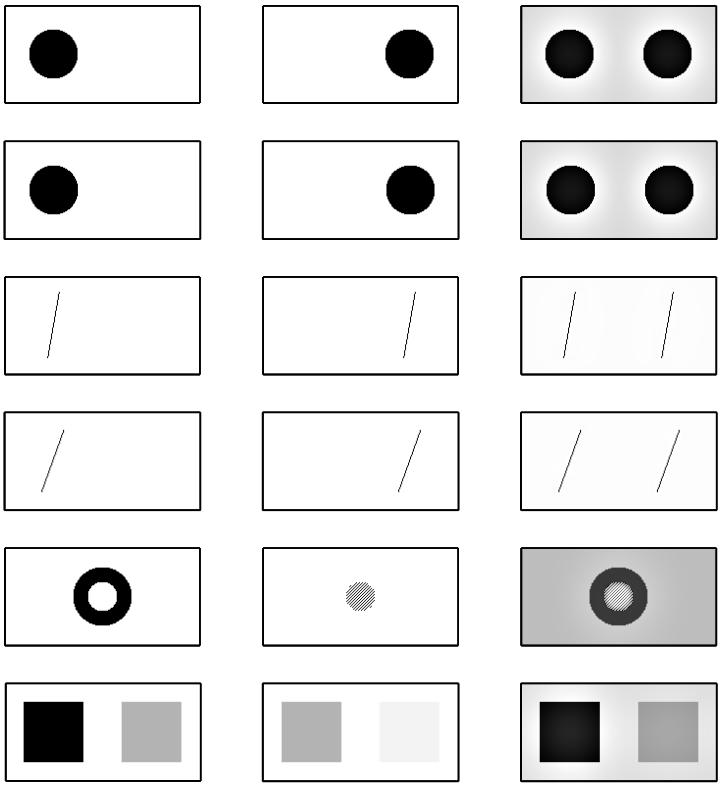

Figure 5: Standard test image pairs and their fusion results.

10 Future Work

The fusion method we have proposed works well for gray-scale images, however as previously mentioned, it may still be improved for color images. Our algorithm has not been specifically designed for interpreting color information, and consequently its color results lack accuracy and vividness. It would be desired to develop a specialized method for handling color data, in order to obtain richer, more accurate results.

As we have seen, over the past years different fusion techniques have been proposed based on various multi-resolution decompositions (wavelets, undecimated wavelets, gradient pyramids etc.). Our algorithm, which utilizes the Laplacian pyramid decomposition, could be easily adapted to utilize any of a large number of alternative decompositions, exhibiting the same *approximation-level / detail-level* structure. Each of these has its advantages and disadvantages, and their performance should be analyzed and compared.

References

[1] P.J. Burt, E.H. Adelson, "The Laplacian Pyramid as a Compact Image Code," IEEE Trans. on Communications, pp. 532–540, April 1983.

[2] E H Adelson, C H Anderson, J R Bergen, P J Burt, and J M Ogden. Pyramid methods in image processing. RCA Engineer, 29:33–41, 1984.

[3] Reinhard, E., Stark, M., Shirley, P., and Ferwerda, J. Photographic Tone Reproduction for Digital Images. Proceedings of SIGGRAPH02, pp.267-276,2002.

[4] Lucas, B. D. and Kanade, T. (1981). An iterative image registration technique with an application to stereo vision, Proceedings of the 7th International Joint Conference on Artificial Intelligence, Vancouver, pp. 674–679.

[5] P.J. Burt and R.J. Kolczynski, Enhanced Image Capture Through Fusion, in Proc. Fourth Int. Conf. on Computer Vision, May 1993, p 173182.

[6] R. Sharma and M. Pavel, "Adaptive and statistical image fusion," in SID Digest, pp. 969–972, Society for Information Display, 1996.

[7] Rockinger, O., "Image sequence fusion using a shift invariant wavelet transform", in Proc. IEEE ICIP, 1997, vol. III, pp. 288–291.

[8] Z. Zhang and R. S. Blum. Multisensor image fusion using a region-based wavelet transform approach. In Proc. of the DARPA IUW, pages 1447–1451, 1997.

[9] Mann, S., AND Picard, R. W. Being 'undigital' with digital cameras: Extending dynamic range by combining differently exposed pictures. In Proceedings of IS&T 46th annual conference (May 1995), pp. 422–428

[10] Debevec, P. E., AND Malik, J. Recovering high dynamic range radiance maps from photographs. In SIGGRAPH 97 (August 1997), pp. 369–378.