Technical Report - Computational Photography

Tassio Knop de Castro, Alexandre Chapiro
Instructor: Luiz Velho

March 17, 2011

1 Introduction

Ordinary consumer cameras are not capable of capturing the whole dynamic range visible to the human eye. For years, many researchers have worked to enhance the dynamic range of a captured scene. Since the development of High Dynamic Range Imaging in the latter 90's, it has never ceased to be a hot topic in Computational Photography.

At first, HDRI software developed by pioneer researchers (like HDRShop and Photosphere) was released, and soon the feature became available in commercial imaging products such as Photoshop. Through the past decade, several improvements were made in the technique, but the user still had to take several photographs and process them with computer software. Very recently, however, cameras with built-in HDR began to appear. The Pentax K-7, released in 2009, is said to be the first camera featuring HDRI. A more notable example is the iPhone 4 platform, which offers an HDR mode in the camera app since the release of the iOS 4.1 this year.

In the realm of HDR video, cameras with HDR sensors, like the RED Epic and the ARRI Alexa, are being widely used these days in film production. Such devices, however, are not available for the general consumer. An alternative is to use cameras that can capture sequences of frames with different exposures and apply a technique similar to the one used for still images. Fortunately, these cameras are more affordable, but methods for enhanced video are still scarce.

While HDRI techniques are very interesting from a professional point of view, at the present point in time they cannot be fully explored by consumers, since HDR images cannot be correctly displayed on normal LDR devices - such as computer monitors and projectors. Such devices require an additional tone-mapping step that decreases the dynamic range of images to a certain value in a controlled fashion in order to display HDR images. True HDR visualization devices exist, but are very uncommon and not available to the consumer. Because of these complications, other image and video improvement techniques may provide interesting alternatives to HDRI. In lieu of the traditional HDR method, there is another way to increase the quality of an image. The Exposure Fusion method merges differently exposed images using a weighted blending process. This approach has some advantages: it is less computationally expensive, does not require tone-mapping, there is no in-between HDR image, and there is no need to calibrate the camera response curve before the processing stage. Exposure Fusion will be discussed in Section 3.

This course focused on the generation of exposure-fused video on hand-held cameras, including those installed in some mobile phones.

2 Related Work

The problem of High Dynamic Range image reconstruction for still images is virtually solved [10]. Furthermore, many different solutions appeared, since the classical work by Debevec and Malik [3].

HDR video reconstruction is a natural extension of the image related problem, therefore, approaches tend to be built upon methods for still images. As it is out of the scope of this text to review all existing approaches, we refer the reader to [8], which is a nice review of the subject. High Dynamic Range video reconstruction is a more challenging task because, from the hardware side, it requires a programmable camera; and, from the software side, the data is dynamic. The earlier reference in this case is [4], where classical vision methods for motion estimation (namely, optical flow) are used to deal with the motion between frames. For a review of methods we refer to [7], where components of the HDR pipeline are presented and discussed with the main focus on video.

Regarding handheld devices, the problem of HDR reconstruction from misaligned and (possibly) blurred long-exposed photographs is treated in [5]. The authors of [1] provide a camera application with HDR mode. In fact they made available a full API for experiments with the low level aspects of the camera hardware. We have used their platform in this work, the video capture being made with a Nokia N900 smartphone.

The work of [6] introduced Exposure Fusion, a clever and simple alternative to HDR Imaging. This new area is very promising, since great results can be produced at a low cost. This method has been applied to merge photographs on mobile devices with great success [9].

3 Exposure Fusion

Exposure Fusion works as a weighted blending of photographs. There is no need for further information besides that present in the image data. This method measures some desired properties of the input images pixels and fuses them to obtain a better quality image. The result is still a low dynamic range image, but it is overall better exposed and overall more aesthetically pleasant. In this work, we used the method of [11] to align the photographs with the purpose of image registration before using them.

The input sequence of images can be considered a stack. Every pixel in every image of the stack is assigned its own weight. Our implementation of this method is based on [6].

3.1 Quality Measures

It is common that every photograph in the stack has some under or over-exposed pixels. It is important to give these pixels low weights during the averaging process, in order to reduce their effect on the final result. It is also desirable to preserve detailed regions that may contain edges or texture and are visually important to the scene and remove blurry areas, so one of our measures has to weight the contrast of a region. Another important goal is to preserve the vividness of the images, so a desaturated picture is undesirable as final result. The basic Exposure Fusion algorithm performs three measures:

- **Exposedness E(p)**

  This measure is applied to each color channel separately, then the weights are multiplied. Considering that the pixel values range from 0 to 1, it preserves the pixels not too near the boundaries, i.e, the pixels which are not under or over-exposed. We use a Gaussian curve centered at 0.5:

  $$E(p) = exp(-\frac{(p-0.5)^2}{2\sigma^2}),$$

  using $\sigma = 0.2$ in the current implementation.

- **Detail D(p)**

  A filter is applied on the grayscale version of the image to enhance detail. The D(p) measure is then the absolute value of the filter response. The current implementation uses a 3 × 3 Laplacian filter.

- **Saturation S(p)**

  The saturation is computed as a standart deviation from the R,G,B channels, at each pixel.

Finally, the resulting weight of the pixel is computed as

$$W(p) = E(p) \cdot D(p) \cdot S(p)$$

3.2 Fusion

There is a need to avoid energy losses in the process of fusion, i.e, the weights of a given pixel along all images of the stack must sum one. Therefore, we normalize W to satisfy this restriction. Even then, a simple weighted sum of the input images does not provide a good result. This is due because some discontinuities on W can generate discontinuities in the final image. An effective way to combine images is presented in [2], where a Laplacian Pyramid is used.

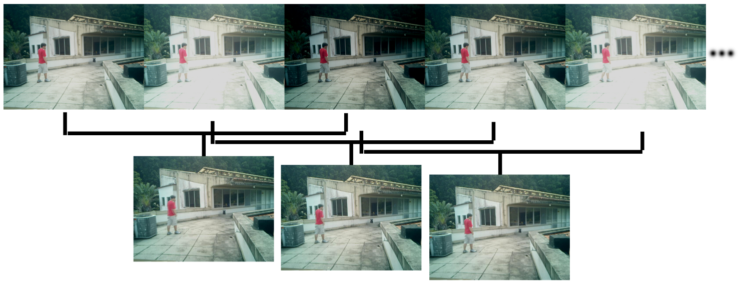
Each input image I is decomposed into a Laplacian Pyramid L(I), which is a stack of filtered and down-scaled versions of the original image. We construct a Gaussian Pyramid G(W) of W and perform the blending on each level l separately , as follows:

$$L(R)^l = \sum_{k=1}^{N} G(W_k)^l \circ L(I_k)^l,$$

where N is the amount of images in the stack, W_k and I_k are the weight maps of the k-th image and the image, respectively, and l stands for the l-th level of the pyramids. This computation is repeated for all levels. The operator $\circ$ represents an entry-wise product, that is, the (i,j) element of the product $A \circ B$ is $A(i,j) \cdot B(i,j)$. This results in a new Laplacian Pyramid L(R), which must be reconstructed via upsample and collapsed into a single image. Performing this process for the three color channels, we obtain our final result.

4 Exposure Fusion Video

The developed method is a straightforward solution based on the still image approach. For each three consecutive captures frames, we choose the first as the reference frame. The other two are then aligned to it and the common exposure fusion algorithm is performed. The whole video is generated in this way, as outlined in the following figure.



4.1 Deghosting

Although Exposure Fusion yields very interesting results when applied to still images, its application in video improvement is still imperfect. The resulting frames show a great amount of ghosting (that is, moving objects appear transparent due to averaging over several frames) and some flickering of the images' apparent brightness occurs.

A novel deghosting method was created to deal with some of the issues that surfaced in the creation of Exposure Fusion based video. The method is based on pixel regions, is at present in an initial development phase and is outlined below.

As seen in [6], applying the EF algorithm to a set of images involves a sum of pixels weighted with certain coefficients. The proposed method involves the addition of yet another coefficient that is used separately from the other three. This value is called the "deghosting" coefficient and is intended to remove areas that encompass moving objects from the final result, with the exception of such movement taken from a single frame, thus removing “ghosts”.

As in the original method, the deghosting coefficient is a floating point number assigned to each pixel in the stack of images used to create an improved video frame. This coefficient is obtained using only the grayscale values of the original images’ pixels (in the future using the Luma part of the images in the YUV colour space may be more desirable, as the currently used Nokia N900 camera-phone saves its images in this space automatically).

First of all, a gaussian low-pass filter is applied to all of the used images in order to filter out undesirable noise. Following, in order to obtain the coefficient of a certain pixel (i, j), a 3x3 region centred on this pixel is taken on both images. The size of these regions was obtained through trial and error and seems to work better than larger areas, that seem to capture too much information and unable to reliably discover movement occurrences. This size could however be larger, depending on the application. In the future, machine learning can be utilized in order to figure out what kind of mask will work best with a certain scene.

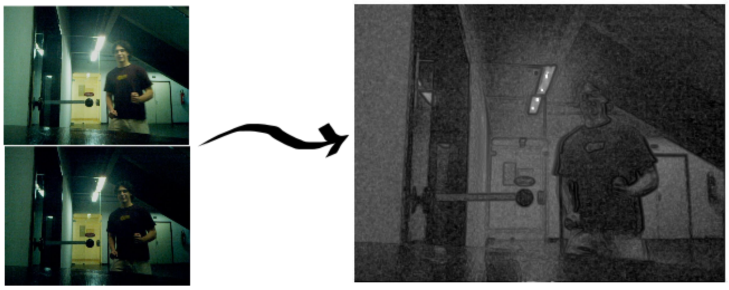
The obtained 3x3 matrices are then divided by the sum of all their elements, in order to obtain more even masks with overall element sum being 1. This makes the patches taken from the images more even, as they usually come from different exposures.

Next, one image whose movement we want to preserve must be chosen. For instance, given images 1 through N as an input, if we wish to obtain the Exposure Fusion video frame related to image number 1, we will assign a deghosting weight to pixel (i, j) for images 2 through N based on the difference between the first and the subsequent patches taken. The coefficient is taken as:

$$K(i, j) = 1 - |(\sum_{i=2}^{N} P_1 – P_i)^{0.1}|$$

The exponent coefficient was obtained from manual testing and gives a good balance between object movement detection and small differences due to the normal distinction of the differently exposed images.

The resulting coefficient can be seen below as an example. Here the numerical values are shown as grayscale.

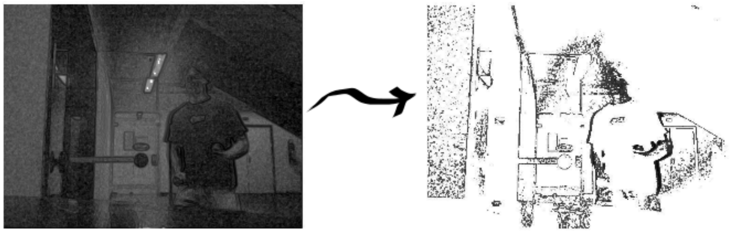


Unfortunately, the deghosting parameter also detects the normal variations due to the difference of exposure between the frames. This detection is especially strong in regions that are poorly exposed. The result of this detection is that image backgrounds are taken with very different weights in the Exposure Fusion process, which causes the resulting video to have flickering background lighting.

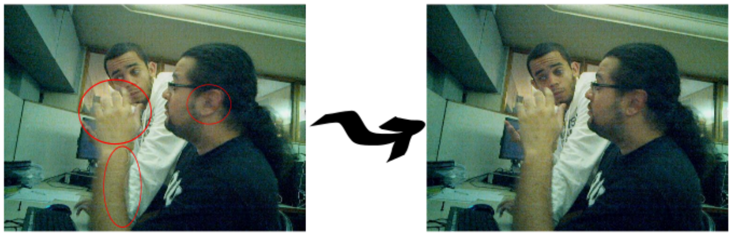
While this problem has not been completely solved yet, several measures can be taken to dramatically reduce this undesirable effect. Particularly, using the previously calculated exposure parameter, a cut-off has been placed that sets to zero the deghosting parameter values for pixels that are already poorly exposed. The reasoning behind this is that poorly exposed regions will already not be considered by the algorithm from [6], and thus need not receive any more consideration.

Another cut-off is placed on pixels that have small deghosting coefficient values. This is done because pixels where actual movement took place usually have larger deghosting coefficient values, which in turn means that the smaller values belong to background pixels and are due to the expected differences between the images.

An image of the resulting coefficient values making use of the thresholds introduced above can be seen below. This is an altered version of the image above. A few constant factors are used in the created method in order to set the cut-offs and are currently picked manually, based on testing.



Finally, the image sequence is rebuilt, making use of the deghosting coefficients created in the earlier steps during the making of a frame, to filter out the moving objects from all frames except the one currently being created. The resulting images have reduced ghosting effects, which sometimes leads to great improvements in the images' visual pleasantness. Some issues regarding the changing background lighting still persist and will be worked on in the future.



5 Results and Conclusion

In this work, we used a Nokia N900 running Maemo 5 for the capture of the input video frames and a desktop computer for the Exposure Fusion processing. Although the developed Exposure Fusion method was implemented on the N900, we preferred a desktop computer as the method was too slow on the mobile phone due to restricted memory and processing power (taking about 2 hours to process a 4 seconds video versus 15 minutes on the desktop).

A command-line tool to align photographs and a fully-functional application to perform alignment registration and exposure fusion were developed in Qt using C++.

A novel deghosting algorithm is being developed. This algorithm is based on a pixel's vicinitie's variations between frames.

5.1 Without Deghosting

The resulting videos can be found at `http://w3.impa.br/~tknop/efusion/`.

5.2 With Deghosting

The resulting videos can be found at `http://w3.impa.br/~achapiro/deghost/`.

References

[1] Andrew Adams, Eino-Ville Talvala, Sung Hee Park, David E. Jacobs, Boris Ajdin, Natasha Gelfand, Jennifer Dolson, Daniel Vaquero, Jongmin Baek, Marius Tico, Hendrik P. A. Lensch, Wojciech Matusik, Kari Pulli, Mark Horowitz, and Marc Levoy. The frankencamera: an experimental platform for computational photography. In *SIGGRAPH '10: ACM SIGGRAPH 2010 papers*, pages 1–12, New York, NY, USA, 2010. ACM.

[2] Peter J. Burt and Edward H. Adelson. A multiresolution spline with application to image mosaics, 1983.

[3] Paul E. Debevec and Jitendra Malik. Recovering high dynamic range radiance maps from photographs. In *SIGGRAPH '97: Proceedings of the 24th annual conference on Computer graphics and interactive techniques*, pages 369–378, New York, NY, USA, 1997. ACM Press/Addison-Wesley Publishing Co.

[4] Sing Bing Kang, Matthew Uyttendaele, Simon Winder, and Richard Szeliski. High dynamic range video. In *SIGGRAPH '03: ACM SIGGRAPH 2003 Papers*, pages 319–325, New York, NY, USA, 2003. ACM.

[5] Pei-Ying Lu, Tz-Huan Huang, Meng-Sung Wu, Yi-Ting Cheng, and Yung-Yu Chuang. High dynamic range image reconstruction from hand-held

cameras. *Computer Vision and Pattern Recognition, IEEE Computer Society Conference on*, 0:509–516, 2009.

[6] T. Mertens, J. Kautz, and F. Van Reeth. Exposure fusion: A simple and practical alternative to high dynamic range photography. *Computer Graphics Forum*, 28(1):161–171, 2009.

[7] Karol Myszkowski. *High Dynamic Range Video*. Morgan and Claypool Publishers, 2008.

[8] Asla M. Sa. *High Dynamic Range Image Reconstruction*. Morgan & Claypool Publishers, 2008.

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

[10] Luiz Velho. Histogram-based hdr video. In *SIGGRAPH '07: ACM SIGGRAPH 2007 posters*, page 62, New York, NY, USA, 2007. ACM.

[11] Greg Ward. Fast, robust image registration for compositing high dynamic range photographs from handheld exposures. *JOURNAL OF GRAPHICS TOOLS*, 8:17–30, 2003.