HDR Photo Compositing Using Short Exposure Bursts

Asif R. Khan, Zahid Hossain

Stanford University

(a) (b) (c)

Figure 1: *(a) Actual long exposure image under low light, typically corrupted by handshake blur. (b) Long exposure generated by naively combining multiple short burst images (c) Our proposed method for generating long expsosure by combining multiple short burst images.*

Abstract

In this project we propose to create a one-click system for taking sharp HDR photos that are compensated against camera movement experienced during long exposures. We aim to do this by capturing multiple short exposure bursts instead of a single, longer exposure. We then perform image registration, specially optimized for short burst images, to account for camera movement, and finally composite these images in such a way as to compress the dynamic range and produce an HDR photo. We also developed a method whereby one could simulate a longer exposed sharp image using a mounted camera by simply holding the camera by hand. Finally we implemented all our techniques for the NVIDIA Tegra Tablet developer platform running on Android Ice Cream Sandwich.

Keywords: Hand Shake, HDR, Short Burst, Registration, Tone Mapping

1 Introduction

Taking a well exposed photograph in low light conditions without hand-shake distortion is a challenge which can easily turn into frustration when one does not have access to a tripod. A great deal of work has been done to alleviate this problem through post-processing an image after it has been captured. Unfortunately, in post-processing there is less information available to recover the original image. Furthermore, the damage has been done and it is a difficult problem to recover the unblurred image without having to fine tune lots of unintuitive parameters, one of the most significant shortcomings of motion deconvolution algorithms. Still, this approach has been justified in the past, but now with the FCam project, there is a platform for programmable cameras allowing other avenues to correct this problem can be explored. In particular, we are now able to take multiple short burst of images along with tweaking various other camera settings to extract the most information from a scene. We posit that with a short enough exposure we could prevent hand shake distortion to ruin an image, but this still gives rise to a different set of problems to solve. First of all, each one of these short exposed images is not properly aligned with others. Secondly, the images, being shortly exposed, will likely contain lots of noise, and finally there is the problem of determining the best method to combine each of these images into a single photograph. As an added bonus, by having multiple images in hand, where each of the images could potentially have different exposure levels, we also are able to do HDR imaging and tone-mapping. The benefit here is the long exposure typically used in HDR imaging could possibly be replaced by a burst of short exposure images.

In this project we explore this concept of taking multiple images and try to solve some of aforementioned problems that come along with such an approach. Specifically, we try to achieve two distinct goals: first to create a HDR image from these short exposure images and second, to simulate a long exposure image in low light conditions while still removing hand-shake. An intrinsic part of achieving the above two goals is a robust method for registering multiple images, which we also explore in this project. We also explore various tone-mapping algorithms to represent the HDR image that is captured by the burst of images.

2 Prior Work

Years of research have been conducted towards removing motion blur distortion caused by hand-shake, with most of the work based on motion deconvolution algorithms. One of the most compelling results demonstrated so far was produced by Cho and Lee [Cho and Lee 2009]. Yet all these methods, including that of Cho and Lee's [Cho and Lee 2009], require a large number of unintuitive parameters and can be very computationally intensive. Hence, researchers have just started to come to a consensus that short burst of multiple images is probably the right way to tackle atleast the handshake problem though there is not a great deal of work reported in the literature concerning this area, as this usually requires hardware support to implement. We were able to explore this avenue because of our access to an NVIDIA Tegra development platform.

By using multiple images one would first need a robust method to register them and secondly a method to combine them into a single image. On this end, there has been extensive prior research conducted on both image registration and HDR exposure compositing.

One of the standard techniques for image registration is based on SIFT (Scale Invariant Feature Transform) as proposed by Lowe [Lowe 1999]. Though SIFT based registrations are very accurate they are very slow at the same time too and hence Bay [Bay et al. 2006] proposed a much faster method named SURF (Speeded Up Robust Feature). Both SIFT and SURF are scale and rotation invariant and still involve a lot of computation which prevents them from being real-time as yet. Later Adams [Adams et al. 2008] developed a much simpler registration technique that is robust enough to handle rotation upto a certain small degree with the goal of making it realtime for mobile platforms.

For exposure composition and tone-mapping, there exist countless methods referenced in the literature on different ways to achieve this goal. However, these methods can be partitioned into two main class of operators - global and local. In global tone-mapping, the same mapping function is applied to each pixel in the HDR representation of the image, while with local tone-mapping, this function is able to vary spatially and exploit the human visual system's sensitivity to local contrast. Due to the large scope of this area and the time constraints of this project, we decided to restrict ourselves to examining several global operators only so that we could make meaningful comparisons. Global operators also have the added advantage of potentially being implemented in the GPU using shaders as future work.

3 Image Registration

There is a handful number of image registration techniques that work decently well in practice. However, one of our main concern was to implement a robust enough technique that is also computationally less expensive since our application is based on mobile platform coupled with the fact that a quick feedback for users is typically more desirable in computational photography applications.

The image registration technqiues that we implemented in our project are discussed below:

- **SIFT**
  SIFT (Scale Invariant Feature Transform), originally proposed by Lowe [Lowe 1999], is a robust method for finding matching points between two images which are then used for computing a homography. Like the name suggests, this method is invariant to scale and orientation and hence is able to register images with large differences in orientation and scale. However it's computationally expensive compared to other simpler methods, and the fact that most of our short-burst images are misaligned only by a small amount, a technique like SIFT is often an overkill.

- **SURF**
  Bay [Bay et al. 2006] proposed an alternative technique called SURF (Speeded Up Robust Feature) that is similarly robust and invariant to scale and orientation like SIFT except it is many times faster. Yet, in our case, the small changes in position and orientation between the short-burst images renders SURF to be slightly an overkill because even simpler techniques, that we will discuss shortly, work just as well but faster by many folds.

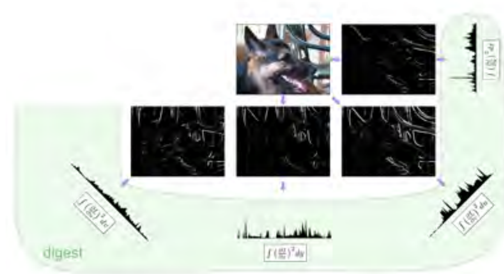


Figure 3: *Image digest similar to Adams but without k strongest corners.*

- **Viewfinder Alignment** A much simpler technique was proposed by Adams [Adams et al. 2008] that takes integrals of oriented gradients in multiple directions along with key points based on corner detection as a digest for an image. Two images are then registered based on their respective digests. The primary motivation of their work was to develop a real-time registration technique for a camera viewfinder on mobile platforms that works on relatively low resolution images and is required to register one image at a time. Unlike [Adams et al. 2008], we were required to register many high resolution images opposed to one low resolution image at a time. Yet this method was attractive to us because we did not require real-time performance.

- **Correlation of Gradients** This is a stripped down version of the above Viewfinder Alignment technique [Adams et al. 2008] where we do not consider any key points but only the vectors of gradient integrals. This method will be elaborated further shortly.

3.1 Correlation of Gradients

The Viewfinder Alignment method proposed by Adams [Adams et al. 2008] was designed to handle changes in orientation up to a certain degree because in their application a user could pan the camera around where considerable rotational distortions were expected. They even purposely allowed users to tilt and rotate the phone for an application where one could use the phone's camera as a video game controller. In our application, however, we expect very little rotational distortion due to hand shake and therefore we realize we could strip off the feature points based of off the first k strongest corners from Adams [Adams et al. 2008] method. Figure 3 illustrates this digest. This leaves us with a digest of an image that has only four vectors of gradient integrals, i.e. p_x, p_y, p_u, p_v (see [Adams et al. 2008] for details). This simple technique was further inspired by Levoy's popular iOS application SynthCam [Levoy], where an image digest, composed of only two vectors, is computed by simply integrating image intensities along rows and columns. Correlation between vectors of two images are then computed to estimate an approximate translational misalignment and this method apparently seems to work just fine. However, our method is slightly more robust because it borrows the idea of using oriented gradient instead of plain intensities from Adams [Adams et al. 2008].

However, we noticed that when our registration is off beyond a visually acceptable level there is a considerable disparity between the translation offsets estimated by the axis aligned integrals and the diagonal integrals, i.e. there would be large disparity between δ_x and

$\delta_u + \delta_v$ (See [Adams et al. 2008] to understand these variables). In most such cases we noticed that the estimates of the diagonal integrals are less reliable. Therefore to make our method more robust we do a weighted average of the estimates made by the axis aligned integrals and the diagonal integrals where more weights are put on axis aligned estimates like the following. (See [Adams et al. 2008] to understand the variables used in Equation (1)).

$$t_x = \frac{\delta_y + w(\delta_u + \delta_v)}{1 + w} \tag{1}$$

Where, $w = \exp\left(-\frac{1}{16}\left(\delta_x - (\delta_u + \delta_v)\right)^2\right)$.

Figure 2 shows output of each of the registration algorithm we implemented. We make a point that in our photography application, most of the misalignments in short burst of images occur due to translation and hence all the sub-figures, except for (a), of Figure 2 look similar. Therefore it makes more sense to use a method that captures the translational misalignment very quickly, which is precisely the method discussed in this subsection.

The image registration process is highly parallelizable and therefore we leveraged multiple cores of Tegra to perform this task in parallel using threads. Since one of the cores of Tegra is a low power CPU we used the other four in parallel and noticed that the method discussed here, Correlation of Gradients, takes about ≈ 2.5 seconds to register 16 RGB images of resolution 1280×960 each.

Finally, even though Correlation of Gradients is the registration algorithm of our choice we implemented all the algorithms mentioned in the previous section in Tegra for comparison purpose.

4 Virtual Long Exposures

One of the primary issues with casual photography that we wished to address with this project was the issue of dealing with handshake when taking long exposures. The solution we chose to pursue in this project was that of taking a burst of short exposures, registering these images, and finally combining them in an intelligent way to simulate a longer exposure. From some informal user studies, we were able to determine that capping exposure time to be around 1/50 seconds (20ms) would ensure that most blur due to handshake would not occur. However, such short exposures in low light conditions suffer greatly from noise, thus presenting us with several challenges with registration and intelligently extracting useful signal from the images.

4.1 Determining Sensor Response

Initially, we assumed that because CCD sensors produce linear output in response to light received, by simply averaging a burst of registered images at a single exposure, we would be able to compute an average pixel value at each location in the image with more quantization bits of precision than we could achieve with a single exposure. By taking this higher precision image, we believed we could apply a linear scaling factor to achieve our desired target exposure. Unfortunately, this assumption failed due to the non-linear response characteristics of the sensor, as can be seen in Section 6. Thus, we were motivated to precisely extract the Tegra tablet sensor's nonlinear response ourselves.

To this end, we implemented a method proposed by Debevec and Malik to extract such a response [Debevec and Malik 2008]. The basic premise of this approach is to assume that each pixel value given by the sensor is the result of a monotonically increasing function that is proportional to the irradiance times the exposure time, $E\Delta t$. By transforming this function into one that is parameterized $\log E\Delta t = \log E + \log \Delta t$, the inverse function $g(Z_{i,j}) =$

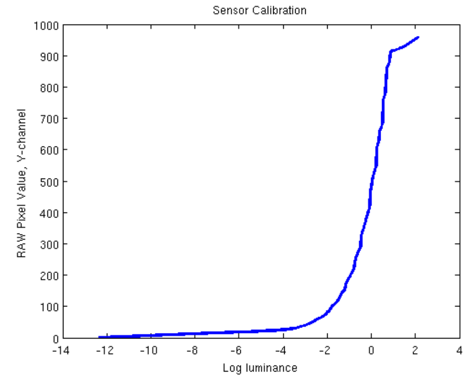


Figure 4: *Response curve extracted for the NVIDIA Tegra tablet's imaging sensor.*

$\log E_i + \log \Delta t_j$, where $Z_{i,j}$ is the pixel value at location i in the image exposed for a duration of Δt_j, can be formulated as a least squares minimization problem over a set of differently exposed images of the same scene. By taking a set of 8 images at different exposures of the same indoor scene, we were able to extract the imaging response of the NVIDIA Tegra tablet of the Y-channel in the XYZ color space. Note that the RAW data of the Tegra tablet is presented in the linear RGB colorspace, so we needed to make the appropriate conversion ourselves. As can be seen in Figure 4, this function exhibits significant non-linear characteristics, as expected.

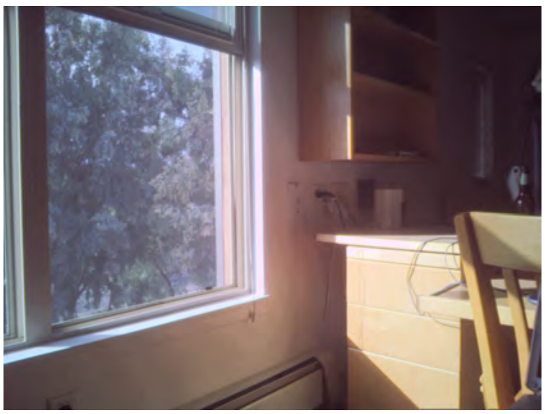
4.2 Creating the Virtual Exposure

Once we were able to extract a radiance map for each image, we took a weighted pixel average as suggested in [Debevec and Malik 2008] to calculate a radiance map for the scene using all exposures available to us. From this we were able to simply add $\log \Delta t_{long}$ to the radiance map and use the inverse mapping of g to compute the virtual exposure image's pixel values. The benefit of this approach was we were able to model the nonlinear characteristics of the sensor during a long exposure while only using short exposure bursts that were not prone to the common problem of handshake blur. We chose to use burst sizes of 16 exposures due to hardware limitations of the FCam interface, as well as memory and performance considerations.

4.3 Note on Color Calibration

In our implementation we found it necessary to work with the RAW image data provided by the NVIDIA Tegra platform since we were working in an information-limited domain. However, we were not provided with a color calibration matrix for the image sensor and were unable to access a Macbeth color checker to do an accurate calibration. We were able to approximate the color matrix at various temperatures by making use of the fact that $YUV420p$ capture already had calibrated colors. We took a picture of a Macbeth color checker on a computer monitor using both RAW capture and YUV capture, and then used least squares minimization to determine a linear mapping between the colors of the image. We assumed that the only nonlinearity applied to the YUV420p was a gamma correction of 2.2 into $sRGB$ space, but it appears there were some other non-linearities we did not model. Unfortunately, our color matrix exhibits a seemingly unnatural bias toward red colors, which manifests itself particularly in low-light situations. We believe some of the red tint seen in our later results could be mostly removed in the future with a more accurate color calibration scheme provided by NVIDIA. An example of the color calibration can be seen in Figure 5.

Without color calibration



With color calibration

Figure 5: *Effect of color calibration on the RAW images from Tegra.*

5 Tone Mapping

The next extension to our investigation on short exposure bursts was to attempt to implement several tone-mapping algorithms for generalized displaying of the framework we implemented for generating radiance maps. To facilitate increasing the dynamic range, for our implementation on the Tegra we allow for two exposures to be set, where 8 exposures are taken at one setting, and another 8 at another. This is meant to help capture short and medium exposures, and use the properties of the stack of images to extrapolate what might happen with a longer exposure. These are then used to create a radiance map that can be tone-mapped. We increase the dynamic range by varying the exposures and by virtue of the fact the minimum detectable luminance quanta becomes more granular through our averaging approach - i.e. we increase our quantization level.

The four algorithms we implemented for tone-mapping are all well-known in the literature:

- **Gamma compression**
  Here we take the dynamic range of the scene and compress it to fit in the dynamic range of our displaying device by fitting a single-parameter power curve to map the highest luminance of the scene to the highest luminance of the display.

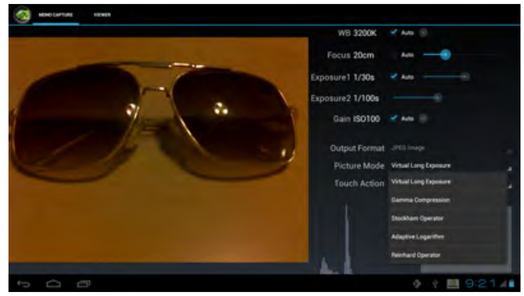


Figure 6: *An Android app implementing all of the algorithms described in this paper.*

- **Stockham operator** [Stockham 1972]
  This operator works by taking the ratio of the logarithm of the scene luminance to the logarithm of the maximum luminace.
- **Adaptive logarithmic tone-mapping** [Drago et al. 2003]
  This algorithm is an extension of Stockham's, where the logarithmic base is varied locally to increase contrast.
- **Reinhard's global operator** [Reinhard et al. 2002] This maps the luminances in a scene to the range of 0 to 1 by simply taking the ratio of the scene luminance to that of the scene luminance plus one:

$$L_d = \frac{L_w}{L_w + 1}‘ \tag{2}$$

Each of these algorithms, including virtual long exposures, were integrated into an app for the Tegra tablet. A screenshot can be seen in Figure 6.

6 Results

6.0.1 Virtual Long Exposure

Figure 7: *A 1 second night-time exposure at ISO100. Note the blurriness due to handshake with such a long exposure.*

In this section we demonstrate how we could generate a long exposed image virtually using only the burst of short exposed images. We refer reader to Section 4 for a detailed discussion on the theory of this method.

Figure 7 shows an image of the Stanford Memorial Church captured at night time with an actual exposure of 1 sec and ISO of 100 by holding the camera with hand. The blur due to handshake is clearly

Figure 8: *A single 20 ms exposure at ISO800. Note the high amount of noise and lack of observable detail. We used sixteen of these exposures to composite a virtual long exposure.*

evident and we have not been able to capture any sharper image after several attempts of trying to keep our hand as steady as possible. Figure 8 shows one of the 16 short burst images we captured using our method at a much higher ISO of 800. Note how dark and noisy the image is while there is no noticeable blur due to handshake. Next, in Figure 9 we demonstrate how taking the non-linear sensor response, as discussed in Section 4, into account after registering the images and finally compensating for ISO gain difference we are able to simulate a long exposed image that is free of handshake artifacts. One important thing to note here is how the noise level goes down, compared to one short expsoure image e.g. Figure 8, when we combined 16 such images which encourages us to take those short burst images at a much higher ISO to extract as much as light as possible while not worrying too much about noise. Despite clean of handshake artifacts in Figure 9 there are still some artifacts due to mismatched color calibration which we believe is related to our imperfect color calibration due to lack of enough information about the Tegra system.

6.0.2 HDR and Tone mapping

In this subsection we will demonstrate how HDR captures with a final tone mapping is done in our method. In the application interface a user selects two different exposures and our system captures 8 images for each one of those exposure resulting in a total of 16 captures. Details of this method is given in Section 5. Figure 10 shows one of the eight frames of the high and low expsoure captures. Note how there the sky in the high exposure image is completely washed out while there are many details present in areas near the tree and the parking lot. In the low expsoure image, on the other hand, has lots of details present in the sky but the trees and the parking lot are too dark. We take this stack of 16 images and compute a radiance map as discussed in Section 5 and finally run it over a few tone mapping algorithms which is demonstrated in Figure 11.

As shown in Figure 11, each tone mapping algorithm exibits a different set of properties though all of them seem to make the color look dull, which might be an issue with our imperfect color calibration as discussed earlier. Gamma compression introduces some unnatural looking halos. Stockham seems to do a better job of conveying the overall ambiance of the scene, but some of the colors are washed out more than the other algorithms. Adaptive logarithm looks more natural than gamma compression, but still seems to exhibit some halos around edges. Finally, Reinhard looks the most realistic among the algorithms we investigaed, even though it also has some color shift, but seems to not have many halos.

(a) Combined without registration and assuming linear sensor reponse.

(b) Combined with registration and radiance mapping.

Figure 9: *Effect of registration and taking non-linear radiance mapping while simulating long exposure images. (a) When images are combined naively without registration and taking the non-linearity of radiance transfer into account (b) An appropriate simulation of* 1 *sec exposure from* 16 *short exposure,* 20 *ms.*

7 Conclusion and Future Work

In this project we have demostrated a proof of concept of how a burst of short exposure images could be used to potentially solve perhaps one of the most annoying yet common problems in photography, i.e. motion blur due to handshake. As byproducts of short burst imaging we also showed how a longer exposure could be accurately simulated using short burst that is free of handshake artifact as well as composite a HDR image. In conclusion we realize that having multiple images of short exposure opens up a lot of interesting possiblities that are worth investigating.

As for the future work, we would like to investigate how to best extract color information at low light condition. We would also like to automate the exposure selection during HDR capture. Registration at very low light condition is unstable and we suspect there might be ways to develop better registration techniques that work well under low light.

Acknowledgements

We would like to thank Jongmin Baek, Dave Jacobs and Kari Pulli for teaching this exciting class. We would also like to thank Alejandro from NVIDIA for fixing various bugs and resolving issues in FCam in a timely manner and being very helpful in general. Finally we would also like to thank Philipp Krähenbühl for giving us tips

on OpenCV.

References

ADAMS, A., GELFAND, N., AND PULLI, K. 2008. Viewfinder alignment. In *Computer Graphics Forum*, vol. 27, Wiley Online Library, 597–606.

BAY, H., TUYTELAARS, T., AND VAN GOOL, L. 2006. Surf: Speeded up robust features. *Computer Vision–ECCV 2006*, 404–417.

CHO, S., AND LEE, S. 2009. Fast motion deblurring. In *ACM Transactions on Graphics (TOG)*, vol. 28, ACM, 145.

DEBEVEC, P., AND MALIK, J. 2008. Recovering high dynamic range radiance maps from photographs. In *ACM SIGGRAPH 2008 classes*, ACM, 31.

DRAGO, F., MYSZKOWSKI, K., ANNEN, T., AND CHIBA, N. 2003. Adaptive logarithmic mapping for displaying high contrast scenes. In *Computer Graphics Forum*, vol. 22, Wiley Online Library, 419–426.

LEVOY, M. Synthcam, [online]. `https://sites.google.com/site/marclevoy/`. Last accessed: 23 March 2012.

LOWE, D. 1999. Object recognition from local scale-invariant features. In *Computer Vision, 1999. The Proceedings of the Seventh IEEE International Conference on*, vol. 2, 1150 –1157 vol.2.

REINHARD, E., STARK, M., SHIRLEY, P., AND FERWERDA, J. 2002. Photographic tone reproduction for digital images. *ACM Transactions on Graphics 21*, 3, 267–276.

STOCKHAM, T. 1972. Image processing in the context of a visual model. *Proc. IEEE 60*, 7, 828–842.

(a) Without registration

(b) SIFT (≈ 41 secs)

(c) SURF (≈ 7.6 secs)

(d) Viewfinder Alignment (≈ 3.5 secs)

(e) Gradient Correlation (≈ 1.2 secs).

Figure 2: *Different image registration algorithms applied on 10 short burts images of a keyboard. (a) All 10 images added without any registration to simulate long exposed image by hand. For all the other subfigures: the images were registered by the algorithm as indicated by their respective captions before adding. The number in parenthesis for each algorithm shows the run-time for only the registration process. Note how each registered image looks so similar to each other justifying the usage of the fastest registration technique in most use cases of our application.*

(a) One of the 8 high exposure images after color calibration.

(b) One of teh 8 low exposure images after color calibration.

Figure 10: *Examples frames of high/low exposure images in HDR mode of our application.*

(a) Gamma compression

(b) Adaptive Logarithm

(c) Stockham operator

(d) Reinhard

Figure 11: *Different tone mapping algorithms.*