Burst Images

Sascha Scheid

RheinMain University of Applied Sciences, Wiesbaden, Germany

Figure 1: *Best teaser image you can provide . . .*

Abstract

Some informative abstract . . . Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Etiam lobortis facilisis sem. Nullam nec mi et neque pharetra sollicitudin. Praesent imperdiet mi nec ante. Donec ullamcorper, felis non sodales commodo, lectus velit ultrices augue, a dignissim nibh lectus placerat pede. Vivamus nunc nunc, molestie ut, ultricies vel, semper in, velit. Ut porttitor. Praesent in sapien. Lorem ipsum dolor sit amet, consectetuer adipiscing elit. Duis fringilla tristique neque. Sed interdum libero ut metus. Pellentesque placerat. Nam rutrum augue a leo. Morbi sed elit sit amet ante lobortis sollicitudin. Praesent blandit blandit mauris. Praesent lectus tellus, aliquet aliquam, luctus a, egestas a, turpis. Mauris lacinia lorem sit amet ipsum. Nunc quis urna dictum turpis accumsan semper.

1. Introduction

Burst photography is a technique in which a series of photographs is taken quickly in succession. This can be useful in a variety of situations, such as capturing action or movement, or to create a sense of motion. In digital photography, burst images are stored as a sequence of image files, typically in a format such as JPEG or RAW.

High Dynamic Range (HDR) is a technique used to improve the dynamic range of an image, which is the range of luminance or brightness levels that can be captured in a photograph. The dynamic range of a scene can often be greater than what a camera is able to capture in a single image, resulting in lost detail in the highlights or shadows. HDR techniques can be used to extend the range of luminance in an image, resulting in more detail and a more realistic representation of the scene.

One way that burst photography and HDR techniques can be used together is to capture a series of photographs with different exposures in a burst, and then combine the exposures into a single HDR image using software. This can be particularly useful in low-light situations, where the camera may struggle to capture a wide range of luminance levels in a single exposure. By capturing multiple exposures and combining them into an HDR image, it is possible to extend the dynamic range and capture more detail in both the highlights and shadows.

Mobile cameras can have several constraints in low light situations, including:

- Low sensitivity: Mobile cameras often have smaller sensors compared to DSLR or mirrorless cameras, which can result in lower sensitivity to light. This can make it difficult to capture usable images in low l ight conditions without using long exposures or high ISO values, which can introduce noise and other artifacts.
- Limited dynamic range: Mobile cameras can have limited dynamic range, which can make it difficult to capture both the shadow and highlight details in a scene. This can result in underexposed or overexposed areas in the image, and can be particularly challenging in low light conditions where the range of luminance values is often greater.
- Motion blur: Long exposures are often necessary in low light conditions, which can result in motion blur if the camera or the subject is moving. This can be particularly problematic for handheld shots or when photographing moving subjects.

To address these constraints, several techniques can be used to improve the performance of mobile cameras in low light conditions. These include:

- Noise reduction: Noise reduction algorithms can be used to reduce the noise introduced by high ISO values or long exposures. These algorithms can smooth out the image while preserving edges and other important details.
- High dynamic range techniques: HDR techniques, such as burst photography and computational photography, can be used to extend the dynamic range of mobile cameras and capture more detail in both the shadow and highlight areas of an image.
- Image stabilization: Image stabilization technologies, such as optical image stabilization or electronic image stabilization, can be used to reduce the effects of camera shake or subject movement during long exposures.

While there are still limitations to the capabilities of mobile cameras compared to larger, more specialized cameras, advances in technology and image processing algorithms are continually improving the quality and performance of mobile cameras in a variety of lighting conditions.

2. Related Work

The HDR+ algorithm is a computational photography technique developed by Google for use in the Google Pixel smartphone. It is designed to capture high dynamic range images using burst photography. The algorithm uses computational techniques to extend the dynamic range of the captured images, rather than relying on multiple exposures as in traditional HDR techniques, in which a rapid sequence of images at different exposures is captured. The captured images are merged to create a single HDR image. The HDR+ algorithm was developed to improve the quality and dynamic range of images captured on mobile devices, particularly in low-light conditions. The HDR+ algorithm consists of several steps, which are outlined in detail in [HSG*16]. In brief, the steps are as follows:

1. Burst capture: A sequence of raw images is captured. In contrast to the classic multi-exposure approach, each image will be underexposed and all images have the same exposure. An image bursts consists of two to eight single images.
2. Image alignment: The sharpest image of the first three images in the burst is selected as reference image and the remaining images in the burst are aligned to it. This corrects for any movement or misalignment between frames.
3. Image merging: The aligned raw images are merged to create a single intermediate raw image that will then be further processed.
4. Finishing: The merged image undergoes a set of operations including general correction, demosaicking and tone mapping. The most important operation is dynamic range compression which reduces the contrast between light and dark areas while preserving local contrast. To do so, the local tone mapping method *exposure fusion* was used, which blends together the best parts of differently exposed images. As there is only one merged image left, the ones to blend are created synthetically from the merged image.

3. Approach

- Provide more details
  - why does the git project not work?
  - overall architecture
  - how does the script work?
  - folder structure
  - configuration

[Bro16] has implemented the image processing pipeline described in section 2. To do this, he uses Halide, a programming language embedded in C++ that allows efficient image processing pipelines to be implemented with little effort. Halide itself requires LLVM to compile. Unfortunately, the project has not been actively maintained for quite some time and since new versions of Halide and LLVM have been released in the meantime, some of which were not backwards compatible, the project could not be compiled out of the box. After many hours of trial and error, the project was able to compile with version 10 of both Halide and LLVM. To get a reproducible executable version of [Bro16], a Docker image was created that contains the application and runs it on container startup. To improve efficiency, a script was created that starts processing all image bursts that are stored inside a bursts directory, which is shared with the Docker container, in parallel. Figure 2 shows the architecture described above.

4. Experiments

- Insert images
- Compare raw vs processed
- Show examples for scenarios:
  - Low light noise reduction
  - Improving contrast / dynamic range
  - Improving overall colors
- Show fails

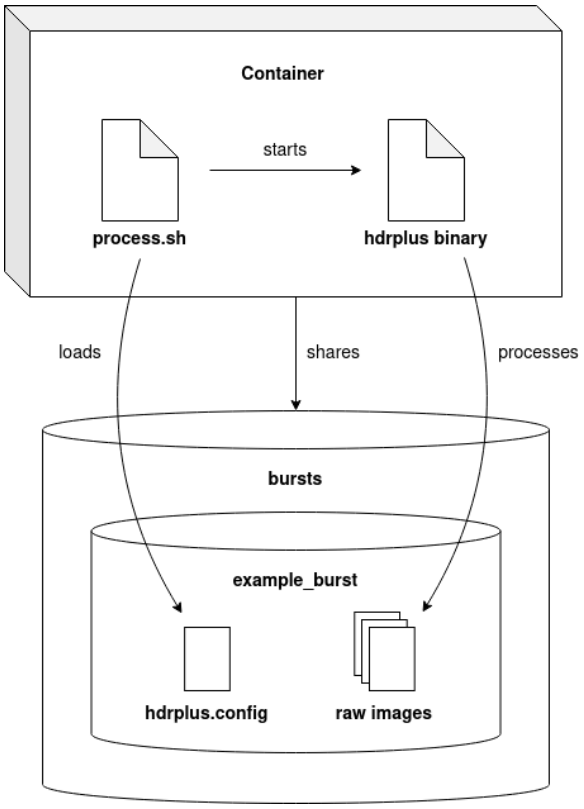

Figure 2: *Architecture of the dockerized application*

5. Conclusions

- Are there scenes that are very easy or hard to improve?
- Advantages:
  - (Revived dead open source project, yay)
  - Portable solution because Docker
  - Parallel processing of multiple bursts
  - Not as much work as manually editing bracketed bursts in photoshop
- Drawbacks:
  - Long initial compiling time when building Docker image
  - Overhead because Docker
  - Ease of use: command-line only, adjusting config by editing a file
- Future:
  - Use case: Photographer wanting HDR images, but does not want to spend hours in photoshop
  - Providing GUI:
    - Hide docker and script execution for non-nerdy users
    - Select burst folders to processes
    - Adjust gain and compress parameters by slider per folder
    - Start processing of selected folders via button click
    - Show result images when finished

6. Acknowledgements

- Cat
- Elementary school teacher
- Ex girlfriend
- Random dude at the bar who bought a round

References

[Bro16] BROOKS T.: HDR+ implementation. `https://github.com/timothybrooks/hdr-plus/`, 2016. Commit: 0ab70564493bdbcd5aca899b5885505d0c824435.

[HSG*16] HASINOFF S. W., SHARLET D., GEISS R., ADAMS A., BARRON J. T., KAINZ F., CHEN J., LEVOY M.: Burst photography for high dynamic range and low-light imaging on mobile cameras. *ACM Transactions on Graphics (Proc. SIGGRAPH Asia) 35*, 6 (2016).