Disparity Estimation and Image Fusion with Dual Camera Phone Imagery

Rose Rustowicz, roserust@stanford.edu
Department of Electrical Engineering, Stanford University

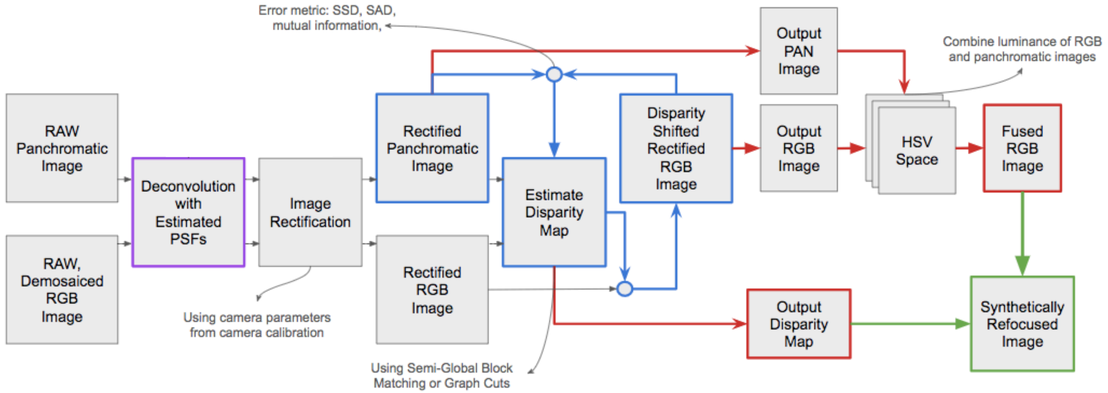
Introduction

Motivation The image signal processing (ISP) pipeline refers to the processing steps that are applied to a sensor's raw output to yield a processed image (such as illumination correction, demosaicing, image sharpening, depth estimation, etc.). ISPs for dual camera modules may integrate information from multiple sensors to construct their processed images. Heide et al. [1] showed that computational imaging and optimization methods can be used in lieu of standard ISPs to solve for a system's output processed image(s). Figure from *FlexISP*:

Objective Use computational imaging and optimization methods to implement different parts of the ISP pipeline for a dual camera phone. In particular, point spread function estimation and deconvolution, disparity map estimation, and image fusion are explored. These processes take in a stereo image pair and yield an output disparity map and fused RGB image. To explore applications of these outputs, a synthetic refocusing algorithm is also implemented.

Instrumentation The Huawei Mate 9 Pro dual camera phone was used in these experiments. The phone has two sensors — one high resolution panchromatic and one "lower" resolution RGB Bayer sensor. Raw images were captured from the phone, while processing was completed on a computer.

Technical Approach — The Overall Pipeline



PSF Estimation & Deconvolution

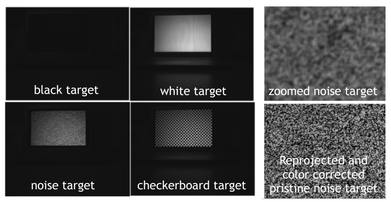
The approach in [2] is implemented to estimate the spatially-varying PSFs for both camera lenses. For the RGB Bayer sensor, the raw image is demosaiced and PSFs are estimated across the lens for each of three color channels.

PSF estimation process:

1. Map pristine noise target into image space using bilinear warping and color correction (based on calibration targets)
2. Solve for the PSFs by minimizing the following cost function:

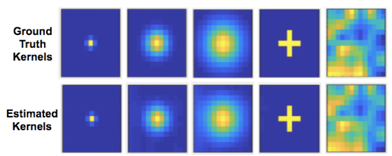
$$\underset{\mathbf{k}}{\text{minimize}}\ E(\mathbf{k}) = ||\hat{\mathbf{u}}\mathbf{k} - \hat{\mathbf{b}}||^2 + \lambda||\mathbf{k}||^2 + \mu||\nabla\mathbf{k}||^2$$

PSF implementation set-up:



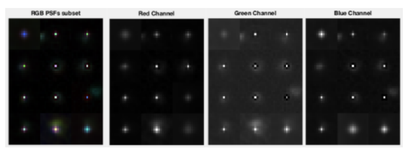
PSF implementation validation:

The procedure was validated by simulating results with known blur kernels — estimated PSFs match the ground truth



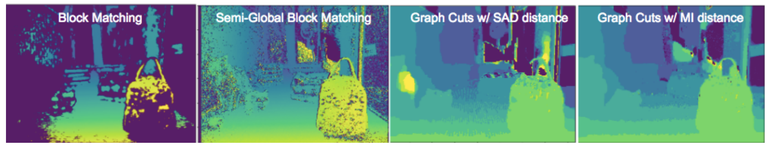
Subset of Estimated PSFs:

A small subset of normalized, estimated PSFs are shown — note the variation across different locations of the lens



Disparity Map and Image Fusion

Several disparity estimation methods were explored (block matching, semi-global block matching, and graph cuts). In particular, graph cuts is a global approach that minimizes a global energy function. Sum of absolute differences (SAD) and mutual information (MI) were used for distance metrics in graph cuts.



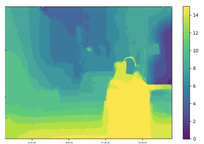
The rectified RGB image is shifted according to output disparity values at each pixel. The RGB image is transformed to hue-saturation-value (HSV) space, and the panchromatic image is fused with the value (V) image by averaging. The fused image is transformed back to the RGB color space to yield the final fused result.

Synthetic Refocusing

Access to a fused color image and disparity map may enable other applications, such as synthetic refocusing. To implement this approach, the fused image is convolved with a disparity-dependent blur kernel at each pixel location.

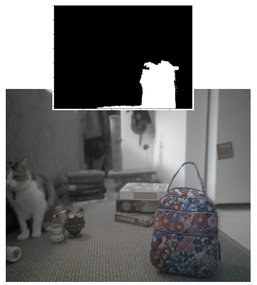
Sigma for disparity-dependent Gaussian blur kernel:

$$\sigma = C\frac{|d - d_f|}{d_f}$$



Shifted disparity map

Mid-focus (board games)



Front-focus (lunch box)

Acknowledgments and References

I would like to thank Dr. Gordon Wetzstein and Dr. Don Dansereau for their advisement throughout this project. I would also like to thank Ashwini Ramamoorthy for helpful collaboration regarding implementation of the PSF estimation technique, and to Fei Xia and Kuan Fang for discussions on using graph cuts for disparity estimation.

[1] F. Heide et al. Flexisp: A flexible camera image processing framework. *ACM Transactions of Graphics, SIG- GRAPH ASIA*, 33(6), 2014.
[2] A. Mosleh, P. et al. Camera intrinsic blur kernel estimation: A reliable framework. *IEEE Computer Vision and Pattern Recognition (CVPR)*, 2015.