

Depth and All-In-Focus Extraction from Aperture-Focal Stacks via Convex Optimization
EE 367 Final Project Proposal
Shane Barratt and Benjamin Hannel

Motivation

Modern cameras (e.g., DSLRs, smartphones) make it easy to automatically capture aperture-focal stacks, that is, multiple images of the same scene with varying apertures and focal lengths. Aperture-focal stacks contain much more information about a scene than a single image. For example, one can infer depth information, since whether or not a pixel is in focus depends on the distance from the camera to the point in the scene, while this is impossible to do with a single image without additional contextual information. Our goal in this project is to deduce, from an aperture-focal stack, the depth of each pixel in the scene, as well as an image where every pixel is in focus.

Related Work

The idea of extracting depth and an all-in-focus image from focal stacks is not new. The idea originally appeared in academic literature in 1987 (Pentland). Pentland investigated using two images with different aperture sizes to construct a depth map for mobile robots. Since then, there has been a great deal of work on extracting depth from a set of images captured at different focal lengths, but they usually use heuristics like gradient intensity to determine which part of an image is in focus (Ens, Grossman, Subbarao). Many of these techniques are slow, taking as long as 25 minutes on a top of the line processor for a single image (Suwajanakorn, Supasorn).

Project Overview

Procedural: We will take images of several static scenes with objects at various depths with a well stabilized camera, which allows us to skip the step of aligning images. We will use the Canon SDK to automate the process of collecting the aperture-focal stack. We will probably use linear f-stop spacing and logarithmic focal length spacing, but are open to experimentation. We plan to collect an abundance of images that we can subsample to investigate the robustness of the algorithm to smaller image stacks.

Approach: We will discretize depth in the scene, and model the depth of each pixel as a probability distribution over the depths. We plan to use a model that says what irradiance will be experienced at a sensor pixel given the true irradiance and depth of the scene, similar to the one described in Lecture 8. Each aperture, focal length, and depth determines a circle of confusion, which is convolved with the true irradiance. While this is a simplification, it has been observed to work well in practice. Our goal is to formulate an inverse problem, where the variables are the true irradiance of the scene and the probability of the depth of each pixel, and the observations are the aperture-focal stack. The objective function will be the reconstruction

loss plus a regularizer (e.g., total variation) on the radiance and depth. We believe that the formulation will be biconvex in the radiance and depth. This approach gives us a natural measure of the uncertainty of our depth estimates. Sharply peaked estimates are high confidence, and more diffuse ones are low confidence. We can also take the expected value of the depth estimates to get a scalar depth.

Algorithm: We will alternate between optimizing for the radiance and depth, since each problem is convex in its respective variable. We will use an ADMM-like algorithm for each problem, which we will run for a small number of iterations (not until convergence) before switching to optimizing the other variable. We hope that this approach will converge to a near global optima for the problem. If time permits, we will port our algorithm to PyTorch or Tensorflow and ultimately to an Nvidia GPU to accelerate it.

Milestones, Timeline & Goals

- (Feb 17) Capture an aperture-focal stack, possibly with the [Canon camera API](#) on a Canon DSLR.
- (Feb 21) Rescale the aperture-focal stack to account for changing magnification.
- (Mar 6) Implement the alternating convex optimization algorithm and apply it to either cropped or downsampled versions of the images we captured.
- (Mar 13) Make sure our algorithm scales to full size images, and incorporate color.
- (If time allows) Port our implementation to a GPU.

References

Ens, J., and P. Lawrence. "[An Investigation of Methods for Determining Depth from Focus.](#)" *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 15, no. 2, 1993, pp. 97–108., doi:10.1109/34.192482.

Grossmann, P. "[Depth from Focus.](#)" *Pattern Recognition Letters*, vol. 5, no. 1, 1987, pp. 63–69., doi:10.1016/0167-8655(87)90026-2.

Pentland, Alex Paul. "[A New Sense for Depth of Field.](#)" *IEEE Transactions on Pattern Analysis and Machine Intelligence*, PAMI-9, no. 4, 1987, pp. 523–531., doi:10.1109/tpami.1987.4767940.

Subbarao, Murali, and Gopal Surya. "[Depth from Defocus: A Spatial Domain Approach.](#)" *International Journal of Computer Vision*, vol. 13, no. 3, 1994, pp. 271–294., doi:10.1007/bf02028349.

Suwajanakorn, Supasorn, et al. "[Depth from Focus with Your Mobile Phone.](#)" *2015 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, 2015, doi:10.1109/cvpr.2015.7298972.