Extending the Lifetime of Broken Smartphone Cameras Through Deconvolution

`fmautner@andrew.cmu.edu`

November 22 2024

1 Summary

Smartphone camera lenses can crack at seemingly random times and for no good reason, and repairing these cameras through official channels can be prohibitively expensive[1]. These cracks result in blurry images that render the camera essentially unusable. The goal of this project is to employ the techniques presented by Heide et al. in "High-Quality Computational Imaging Through Simple Lenses" [2] to restore acceptable image quality from damaged lenses at no additional cost to the end user. Through this project, I aim to gain in-depth understanding of various deconvolution approaches and experience in implementing them for consumer applications.

2 Background

This project builds upon the foundational work presented in "High-Quality Computational Imaging Through Simple Lenses" [2], extending their techniques to address a practical consumer need. While the original paper focused on simple lenses, I plan to adapt their methodology to restore image quality from cracked smartphone camera lenses, potentially saving consumers hundreds of dollars in repair costs.

The technical scope of this project encompasses several core areas of Computational Photography, with a primary focus on accurate PSF estimation and efficient deconvolution. The PSF estimation presents a particular challenge, as the method presented in [2] is not applicable for fixed-aperture smartphone lenses. Instead, I will need to perform standard PSF calibration, but spatially based.

At the implementation level, this project will leverage powerful optimization techniques, particularly the first-order primal-dual algorithm by Chambolle and Pock [1]. While this algorithm is used in the deconvolution pipeline in [2], it has broad applications across various optimization problems in computer science. Through this work, I will gain both theoretical understanding of fundamental imaging problems and practical experience implementing convex optimization algorithms.

3 Resources

Below are the resources I expect to need for my project. I would greatly appreciate if the teaching staff could help provide the hardware items marked with an asterisk (*):

Imaging hardware:

- A simple lens (preferably Nikon F mount)*;

- An (x to F) lens mount adapter for the simple lens (if applicable)*;

- A collection of cheap, small cameras, with poor quality or damaged lenses. These can be mass produced, simple USB cameras. Their purpose is to simulate a damaged smartphone camera*.

PSF Estimation hardware:

- A Point light source*;

- A Motorized XY Linear Stage to move the light point and obtain spatial PSF calibration*.

- A Tripod or equivalent mount for a smartphone and camera to perorm PSF calibration*.

---

[1] See Apple's repair cost estimates for different IPhone models `https://support.apple.com/iphone/repair`

Software and data:

- Starter code from [2] in matlab, which I'll re-implement and extend in Python, making appropriate modifications for my particular use case.
- Original images and blur kernels from [2].

Intellectual: Please refer to the References section.

4 Goals and Deliverables

Plan to achieve (Reproduce [2] and extend to smartphone cameras):

- Implement the pipeline from [2] and test on provided data.
- Perform PSF estimation for simple lens and examine results of deconvolution.
- Extend work to simple/ damaged small cameras and smartphone cameras.
- Profile the memory and computational efficiency of my Python implementation.

Hope to achieve (Improve usability into a consumer application):

- Develop a robust and user-friendly PSF calibration process accessible to regular consumers based on a known target and camera matrix to cancel-out barrel distortion.
- Optimize implementation to allow for real-time image enhancement.
- Develop a working prototype of an IOS app to perform PSF calibration and correct images taken with a blurry lens.

5 Tentative Schedule

This schedule provides a more detailed outline of the goals and deliverables.

Week	Deliverables
Nov 22 - Nov 28	• Set up development environment[*] • Implement the proposed deconvolution pipeline of [2] in Python[*] • Evaluate results against [2] using the provided data[*]
Nov 29 - Dec 5	• Profile memory and computational efficiency of implementation[*] • Perform standard spatial PSF estimation for simple lens and small cameras[*] • Perform empirical experiments with calibrated small cameras[*]
Dec 6 - Dec 12	• Develop deconvolution-based PSF estimation with known target and camera matrix[†] • Optimize implementation to run on smartphone processors[†] • Start development of a prototype smartphone app for PSF estimation and image correction[†]
Dec 13 - Dec 16	• Finish development of prototype smartphone app[†] • Prepare presentation and video[*] • Complete final report[*]

Table 1: Project Timeline and Deliverables. Plan to Achieve: [*], Hope to Achieve: [†]

References

[1]   A. Chambolle and Thomas Pock. "A First-Order Primal-Dual Algorithm for Convex Problems with Applications to Imaging". In: *Journal of Mathematical Imaging and Vision* 40 (2011), pp. 120–145. URL: https://api.semanticscholar.org/CorpusID:261281173.

[2]   Felix Heide et al. "High-quality computational imaging through simple lenses". In: *ACM Trans. Graph.* 32.5 (Oct. 2013). ISSN: 0730-0301. DOI: 10.1145/2516971.2516974. URL: https://doi.org/10.1145/2516971.2516974.