Implementing Learning-Based Light Field View Synthesis

Thomas Lauer
UC San Diego
9500 Gilman Drive, San Diego CA
tlauer@ucsd.edu

Winston Durand
UC San Diego
9500 Gilman Drive, San Diego CA
wdurand@ucsd.edu

Abstract

Light field photography allows for the capture of images from multiple perspectives using a single camera array or array of microlenses. However these come at a tradeoff between spatial resolution and angular resolution. As the number of the microlens elements increases, the resulting lightfields have higher angular resolution which is useful for depth estimation, but will have lower spatial resolution for novel view synthesis. We implement the a paper on learning based method to synthesize views from light field camera data [1] in PyTorch.

1. Introduction

A Lytro is a common light field camera which uses microlenses to capture multiple perspectives of a scene in one light field.

Kalantari summarizes several existing methods for interpolating lightfields and their drawbacks. Many rely on high quality input images and well defined orientations between views, which are difficult to achieve with consumer light field cameras, and impossible with light fields captured by hand with cellphone cameras.

Many geometric methods novel view synthesis such as Levoy *et al*. [2] require relatively high sample counts to ensure full coverage of the 4D lightfield, as the simple linear interpolation used by Levoy do not work well if there are heavily occluded regions or missing data.

Wanner *et al*. [3] utilizes an optimization based approach, which calculates disparities using traditional computer vision as a preprocessing step. However, because the disparity estimation is independent from the loss function, it can not be optimized as part of the training process.

Kalantari *et al*. [1] propose a method to interpolate between sparsely sampled sub-apertures views. They use an 8×8 subset of the full 14×14 subaperture, since the edge pixels are often black. The four corner sub-aperture views are fed into a series of two networks, the first is used to predict disparity which is then used to warp the sample images to the final perspective. The second network then takes these warped images, along with some additional metadata, and blends them together to produce the final RGB image of the novel view. Keeping this process differentiable allows both CNNs to be trained at the same time, which lets the disparity estimator become tuned to work with the final CNN.

2. Our Proposal

Our goal is to reimplement the paper *Learning-Based View Synthesis for Light Field Cameras* by Kalantari *et al*. [1] and additionally try to apply this technique to non-Lytro camera array captures. Further, we would like to investigate rendering novel views outside of the planar quadrilateral formed by the source images. To evaluate the performance of our model, we will use the structural similarity index measure (SSIM) and peak signal to noise ratio (PSNR), which are the same metrics used by the original paper.

3. Technology

We intend to implement this using PyTorch, because it has high performance, automatically handles calculating derivatives, and we have previous experience with it. Additionally, we will likely be using OpenCV or something similar for feature extraction in the preprocessing stage. We are not expecting to get real time performance, the paper claims it took 12.3 seconds to synthesize a single image using a Matlab implementation. We will stitch the individual images into a video to help visualize the results.

4. Datasets

Our training datasets will be Lytro captures since it represents a standardized input format which is widely available, for example from Stanford at http://lightfields.stanford.edu/LF2016.html. Because Lytro images are consistently formatted, we should be able to easily test our code with many different light fields. We also want to test this approach with a camera array, and we could easily adapt our preprocessing to accept the Stanford Multi-Camera Array. We will not be considering more free form light fields, such as those captured by handheld phone cameras, because of the added complexity of different camera orientations.

5. Implementation Details

5.1. Preprocessing

Each training lightfield is split into a number of patches. To minim

5.2. Network

Our network was implemented using PyTorch, using two separate convolutional networks to perform the disparity estimation and final color assembly. Our disparity estimation network takes a $200 \times H \times W$ tensor as input, which we calculate using the preprocessing step.

6. Questions to be Answered

How does the disparity from the first CNN compare to disparity produced by traditional computer vision methods, such as OpenCV?

Will the approach outlined work when we select sampled images from different locations, instead of the corners of the sub-aperture views? Further, is the CNN learning in this fixed grid transferrable to other grid layouts?

Will this work if we try to reconstruct a novel view outside the bounds of the sampled sub-aperture views?

If learning in non-transferrable across capture grid aspect ratios, is it possible to solve this with image stretching?

References

[1] Nima Khademi Kalantari, Ting-Chun Wang, and Ravi Ramamoorthi. Learning-based view synthesis for light field cameras. *ACM Transactions on Graphics (TOG)*, 35(6):1–10, 2016.

[2] Marc Levoy and Pat Hanrahan. Light field rendering. In *Proceedings of the 23rd annual conference on Computer graphics and interactive techniques*, pages 31–42, 1996.

[3] Sven Wanner and Bastian Goldluecke. Variational light field analysis for disparity estimation and super-resolution. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 36(3):606–619, 2014.