Low-Light Light Field (LF) Restoration

Snehal Singh Tomar
Aqil K H
Subhankar Chakraborty

The Story So Far...

- A LF camera offers unique advantages such as post-capture refocusing & aperture control, but low-light conditions severely limit these capabilities
- We need to decode raw LFs captured using lenslet based plenoptic cameras
- This problem has been successfully addressed in [1], and its MATLAB code is publicly available
- Restoring LFs captured in low-light is not possible with single-frame low-light enhancement techniques designed for smartphones and DSLR cameras

[1] Dansereau, Donald G., Oscar Pizarro, and Stefan B. Williams. "Decoding, calibration and rectification for lenselet-based plenoptic cameras." Proceedings of the IEEE conference on computer vision and pattern recognition. 2013.

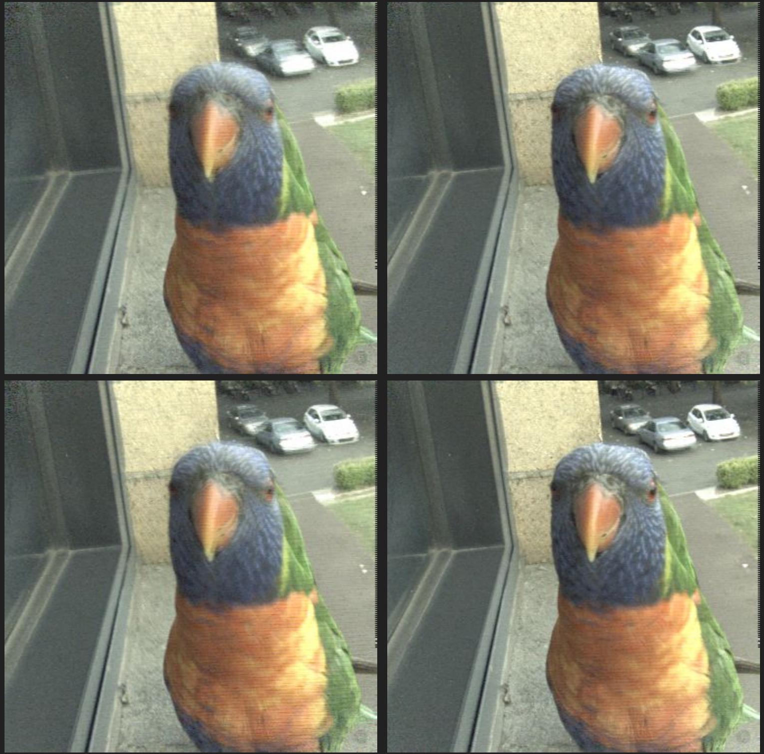
PlenoptiCam

- PlenoptiCam [2] is an open-source software for scientific light field computation
- Completely Python based and comes with a GUI
- It has the ability to calibrate an image taken by a plenoptic camera and extract sub-aperture images or synthetically focused photographs

[2] Hahne, Christopher, and Amar Aggoun. "PlenoptiCam v1.0: A light-field imaging framework." arXiv preprint arXiv:2010.11687 (2020).

Comparison of Results

LF Toolbox Decode



PlenoptiCam Decode

Comparison of Results

PlenoptiCam Decode

LF Toolbox Decode

L3FNet: Introduction

- L3FNet [3] is a deep neural network for Low-Light Light Field (L3F) restoration
- It not only performs visual enhancement of each LF view but also preserves the epipolar geometry across views
- This is achieved by adopting a two-stage architecture
    - Stage-I looks at all the LF views to encode the LF geometry
    - This encoded information is then used in Stage-II to reconstruct each LF view
- Four LFs of different scenes, with different light settings varying from optimal to extreme low-light are captured using the Lytro Illum Camera to generate a dataset

[3] Lamba, Mohit, Kranthi Kumar Rachavarapu, and Kaushik Mitra. "Harnessing multi-view perspective of light fields for low-light imaging." IEEE Transactions on Image Processing 30 (2020): 1501-1513.

We use a modified version of the below architecture

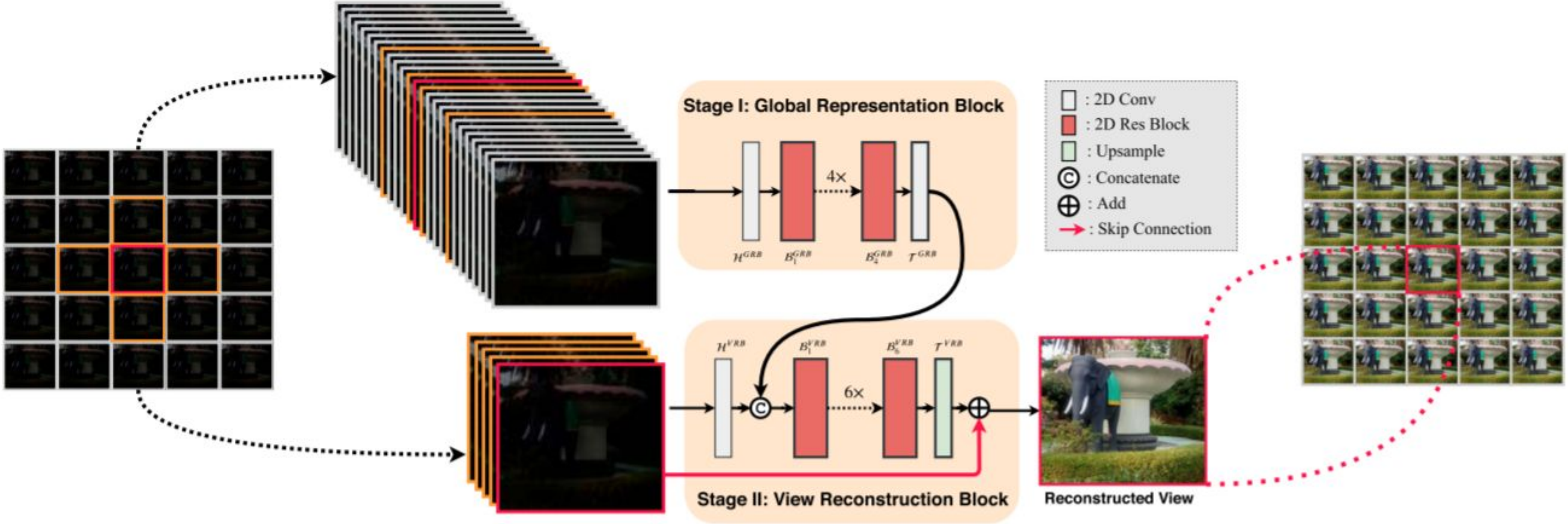


Fig. 3. L3Fnet architecture: The proposed architecture consists of a Global Representation Block (Stage-I) and a View Reconstruction block (Stage-II). Stage-I operates on the full low-light LF to obtain a latent representation that encodes the LF geometry. This latent representation is then used by Stage-II to restore each LF view.

Implementation Details

- We are using a variant of L3FNet
- Lytro Illum captures 225 views of a scene arranged in a 15×15 grid
- As the peripheral views are affected by ghosting and vignetting effect, and we were short of GPU resources, we restore only the central 3×3 views
- We train to rectify the LF captured at 1/100th of the exposure time as that of a well-lit LF. The well-lit LF is taken as the ground-truth
- The loss function is a weighted sum of L1 loss, Fourier loss, Perceptual loss and a gradient of the angular dimensions
- We train for 8000 iterations with the Adam Optimizer with a learning rate of 0.0001 and no rate decay

Results

- A view from the ground-truth image from each scene is compared to the corresponding view in the rectified L3F image
- The raw L3F images are not presented as they look almost pitch-black and it is hard to infer anything from it
- The image on the left is the rectified image after training L3FNet for 8000 iterations, and the one on the right is the ground-truth image

Restored Image

Ground-Truth Image

Restored Image

Ground-Truth Image

Restored Image

Ground-Truth Image

Restored Image

Ground-Truth Image

Future Work

- Complete the documentation of PlenoptiCam
- Perform rectification on Decoded LF images
- If time permits, see if we can throw away the PlenoptiCam completely and train a neural-network which performs both decoding and restoration of L3F data jointly

Challenges Faced/ Possible Issues

- The training process took close to 12 hours for just for 8000 iterations
- Rectifying LFs directly is a completely new domain for us
- We trained on an NVIDIA GeForce GTX 1050 Ti with 4GB of VRAM. This is pushed to the limits even on the simplified training we performed and is definitely not enough if we decide to perform rectification on decoded or raw LFs directly

Questions?