Comparing the Robustness of Different Depth Map Algorithms

Fang-Yu Lin
Dept. of Electrical Engineering
fangyuln@stanford.edu

Warren Cheng
Dept. of Electrical Engineering
wcheng90@stanford.edu

Linda Banh
Dept. of Electrical Engineering
lbanh@stanford.edu

Abstract

**This is a joint project between EE 367 and EE 368 at Stanford University.*

Accurate depth maps are critical in movie production, and augmented reality. Light field imaging processing has seen moderate traction for scene depth mapping and reconstruction. We explored three depth mapping algorithms least squares gradient (LSG) , plane sweeping, and epipolar plane strategies in order to determine their depth estimation accuracy as well as their relative computational intensities. Using light field images of simulated and real scenes, we computed their depth maps using the three methods and computed the overall time it took to compute on our machines in Matlab. Overall, plane sweeping appeared to generate the most accurate depth maps, but turned out to be very time intensive. LSG was the least computationally intensive, but produced the least accurate depth maps. The epipolar technique performed moderately well for both metrics.

1. Motivation and Objective

Depth mapping has many wide ranging applications. In movie production it is used as a means of creating accurate models of movie sets and objects for post production tasks such as rotoscoping. Being able to create accurate 3D representations of a scene also apply heavily to augmented reality (AR) applications. In order to superimpose digital content onto the real world, there must be a way to map the contours of the real world so that a digital content can accurately interact with the real world.

Current approaches use specialized time of flight (ToF), or depth, cameras in order to accomplish this. Basically, this works with a laser or infrared source emitting light at different points in a scene simultaneously. The disparity in the time of the reflected light off of surfaces is used in order to determine the appropriate depths of various objects. This approach works well for coarse representations of the world, but can be problematic for features that fall below the resolution of the depth camera. This is partially due in part to the fact that the patterns emitted by ToF cameras have poor spatial resolution at farther distances. Not to mention, it becomes very difficult to map the features of rooms and environments far away since depth cameras have range limitations due to the scattering of the reflected infrared light at greater distances. This makes meshing the outdoors difficult.

Light field imaging has many wide ranging applications such as post image capture refocusing and scene depth estimation. In recent years, it's seen more traction for use in depth mapping as seen from the research done by Disney. One of the major drawbacks however, is that light fields generally contain large amounts of data since multiple views of a particular scene must be captured at a time. This makes it computationally costly to compute for disparity and depth maps.

Therefore, through this project, we hope to improve these factors by exploring different depth map algorithms. Particularly, we will be analyzing and implementing the least squares gradient strategy proposed by Adelson and Wang in 1992[1], a plane sweeping strategy that was slightly modified from Yang's paper on occlusion depth maps[6], and lastly, an epi-polar plane compression strategy proposed by Kim et al. in 2013[5] in Disney Research, which also included a fine-to-coarse refinement method to find the optimal disparity/depth.

2. Related Works

Some work has already been done in using 4D light fields to create depth maps. For example, Bolles et al.[2] were the first to extract depth from a dense sampling of images. However, this method does not perform well with real world data that contains occlusions, varying illumination, etc. Many other works use techniques from stereo reconstruction, including plane sweeping[7]. However, again, these strategies are not necessarily robust to occlusions or other factors from real world data. Therefore, we chose the epi-polar plane strategy since Disney Research has had success in reconstructing scenes for films and the least squares gradient strategy as a baseline. Yucer and Sorkine-Hornung built on Kim et als Disney Research work by exploring a technique of using handheld video data to extract unstructured light fields (light field images captured with an unknown camera path) in order to create accurate 3D depth models of objects and segment them from their cluttered backgrounds [4].

3. Methods

3.1. Dataset

We are using the 4D Light Field Benchmark Dataset[3] and Stanford Light Field Archive. For 4D Light Field Benchmark, each 4D light field data contains a 9×9 views with a resolution of 512×512. Our quantity evaluation is based on three data from it, which are 'boxes', 'dino' and 'cotton'. For the Stanford Light Field Archive, we took the 'Lego Truck' light field for quaility evaluation, with 17×17 views and 960×1280 resolution.

3.2. Least Squares Gradient (LSG) Method

Adelson and Wang explored depth estimation using a least squares error method between light field images. Displacement of a viewpoint results in a displacement of an image patch by $d\Delta_x$ and $d\Delta_y$. This leads to the following equality:

$$L(x, y, u, v) = L(x - d\Delta_x, y - d\Delta_y, u + \Delta_x, v + \Delta_y)$$

This relationship was rearranged and redefined over all image patches as a squared error E, which will be minimized with respect to d.

$E = \int_\alpha \sum_p L(x, y, u, v) - L(x - d\Delta_x, y - d\Delta_y, u + \Delta_x, v + \Delta_y)$

$$d^* = \underset{d}{\operatorname{argmin}}\, E$$

Solving for the previous optimization problem, we arrive at the following conclusion:

$$d^* = \frac{\sum_p (L_x L_u + L_y L_v)}{\sum_p (L_x^2 + L_y^2)}$$

where d represents the displacement between the object's image across all the light field images, I_x represents the spatial derivative in the x direction, I_y represents the spatial derivative in the y direction, I_u represents the derivative with respect to the viewing position in the u direction, and I_v represents the derivative with respect to the viewing position in the v direction.

3.3. Plane Sweeping Method

In our second method, we explore plane sweeping. In our implementation, we slightly modify the method proposed by in Wang et. al's *Occlusion-aware Depth Estimation Using Light-field Cameras* [6]. Instead of performing this initial depth map method on only two views in the light field, we apply it to the entire light field.

First, 4D shearing of the light-field data is performed. This purpose of this is to refocus each light field view to the center view (demonstrated in the following formula),

$$L_d(x, y, u, v) = L(x + ud, y + vd, u, v)$$

where L_d represents the refocused light field view, L is the original light field view, and d represents the disparity. In addition, x and y are the spatial coordinates in the horizontal and vertical directions respectively, and u and v are the light field coordinates in the horizontal and vertical directions respectively.

Once the images have been re-aligned, they are stacked together and their cost volume, C, is found. (Variance is used as cost function in this implementation.)

$$\bar{L}_d(x, y, u, v) = \frac{1}{|U||V|} \sum_{u \exists U} \sum_{v \exists V} L_d(x, y, u, v)$$

$$C(x, y, d) = \frac{1}{|U||V|} \sum_{u \in U} \sum_{v \in V} (L_d(x, y, u, v) - \bar{L}_d(x, y, u, v))^2$$

$L_d(x, y, u, v)$ is the mean of $L_d(x, y, u, v)$, and U and V are the set of all possible values of u and v in the light field. $|U|$ and $|V|$ represent the number of values in each set.

Next, the cost volume is filtered with a box filter of size 3x3. The purpose of this is to create a more visually appealing result that can help remove some noise.

Lastly, after building the cost volume and filtering it, the optimal disparity d^* is found, and this is used as the disparity map.

$$d^* = \underset{d}{\operatorname{argmin}}\, C(x, y, d)$$

3.4. Epipolar-Plane and Fine-to-Coarse Refinement Method

Kim et al. extracted depth map from dense 3D light field with high resolution images. It is proposed as robust method against occlusion. Also, it is efficient with GPU since there is no global-optimization technique and all pixels can be processed in parallel. Our method in this project is a slightly simplified version from theirs. We ignore the propagation part here since we are only extracting the depth map in central view. Also, we generalized this 3D method into 4D.

First, we compute Edge Confidence C_e as

$$C_e(x,y) = \sum_{(x',y')\in N(x,y)} \parallel I(x,y) - I(x',y') \parallel$$

where I is the central view image and N denotes a 3×7 window. We set a threshold of 0.05 in level0 and 0.1 in every other level in our fine-to-coarse procedure, which we will explain later.

Second, for each pixel (x, y) in I, we sampled a set of radiance R in every different views as $R(x,y,u,v,d) = L(x + (\hat{u} - u)d, y + (\hat{v} - v)d, s, t) | s = 1..n,\ t = 1..m$ where n corresponds to number of horizontal views and m corresponds to number of vertical views. We can then compute a score of color density S as

$$S(x,y,d) = \frac{1}{R(x,y,u,v,d)} \sum_{r\in R(x,y,u,v,d)} K(r - \bar{r})$$

where K denotes a kernel $K(x) = 1- \parallel \frac{x}{h} \parallel$ when $\parallel \frac{x}{h} \parallel \leq 1$ and 0 otherwise. We set $h = 0.1$ here. Initially, $\bar{r}$ is the radiance correspond to the pixel that is computing S. To make $\bar{r}$ more robust, we will update $\bar{r}$ iteratly by mean-shift algorithm as

$$\bar{r} \leftarrow \frac{K(r - \bar{r})r}{K(r - \bar{r})}$$

Next, we will choose the disparity $d*$ that maximize score S.

$$d^* = \underset{d}{\operatorname{argmax}}\ S(x,y,d)$$

Note that we only keep the d^* value with a Depth Confidence C_d higher than $\epsilon = 0.03$. C_d can be computed as follows,

$$C_d(x,y) = C_e(x,y) \parallel S_{max} - \bar{S} \parallel$$

We will get our disparity map $D(x, y)$ and apply a median filter with a window size of 3×3 for denoising. This disparity map will be saved for our next step of fine-to-coarse. Also, we will update the disparity bound for every pixels that no d^* is assigned.

To fill-up the disparity map pixels with low C_d, we then start our fine-to-coarse procedure. We first apply a Gaussian filter on the central view image I, with a kernel size of 7×7 and a standard deviation of $\sigma = \sqrt{0.5}$. After Gaussian blurring, we down-sample the image with a factor of 0.5. We will start from computing Ce again. This loop will continue until the dimension of I is less than 10 pixels. We then upsample the disparity maps from the coarsest level to fill-up every pixels without changing the $d*$ we obtained from finer levels and combined them all as the final disparity map D.

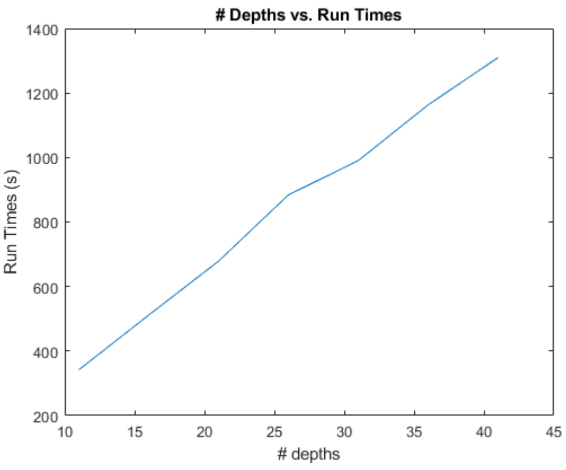


Figure 1. Depths vs. Run Times (in seconds)

4. Results

For the depth Z here, we computed through the Matlabtool Heidelberg Dataset provided. It is base on the equation

$$Z = \frac{fb}{d}$$

where f is focal lengths in pixel and b is the baseline of each adjacent views. Also, we want to emphasis on the accuracy for EPI and Fine-to-coarse method with only the level 0 depth estimation, we will show both the final result (EPI1) and level 0 result (EPI2).

5. Analysis and Discussion

5.1. LSG Method

In the LSG method, we noticed pretty accurate reconstructions of the depth map with a pretty short runtime. However, it did appear that the depth algorithm worked better on the foreground rather than the background. As you can see in cotton image, the features of the bust match the ground truth fairly well, but the background is a bit off. The algorithm does, however, appear to fair better on the background when there are textures, as you can see in the dino dataset with the grains of the wood and shadows of the dinosaur.

5.2. Plane Sweeping Method

First, some experiments were run with the 'boxes' light field from Heidelberg's dataset[3]. Using varying depths (or planes), the computation times and mean-squared error (MSE) were evaluated. As seen in Figure 1, as the number of depths increases, so does the run time. This makes sense intuitively, since if there are more depths, the algorithm will

Table 1. Depth Map Algorithm Comparisons using Heidelberg Dataset

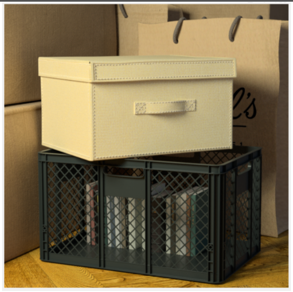
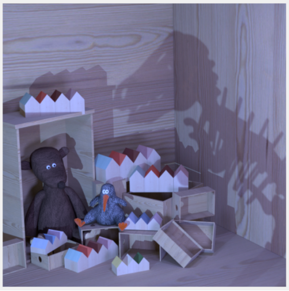
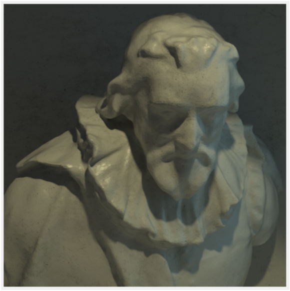
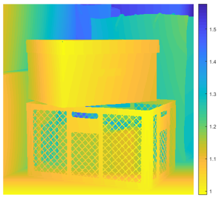
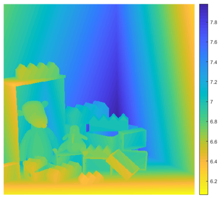
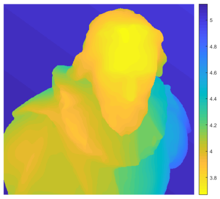
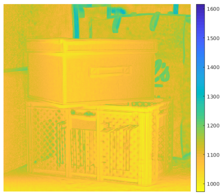
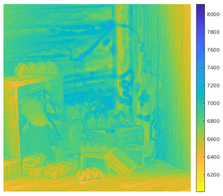
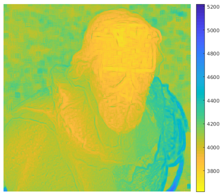
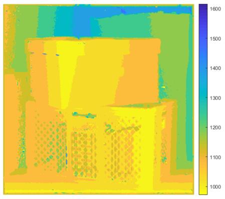
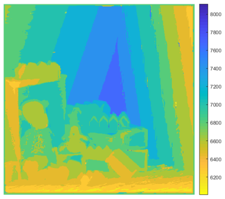
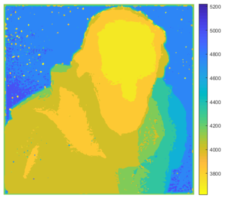
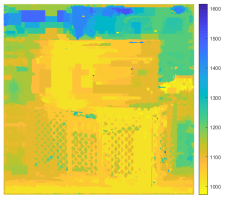
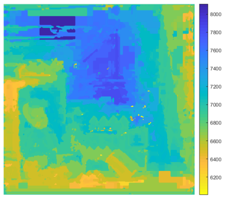
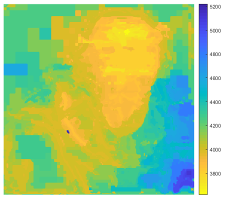
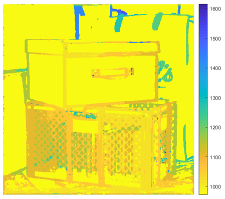
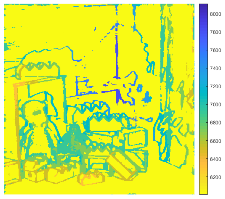
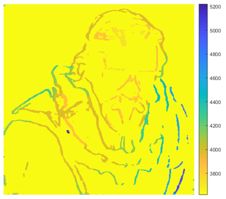
Algorithm	Boxes	Dino	Cotton
Original			
Ground Truth			
LSG			
Plane Sweeping			
EPI1			
EPI2			

Table 2. Depth Map Algorithm Error

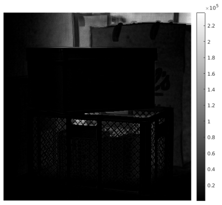
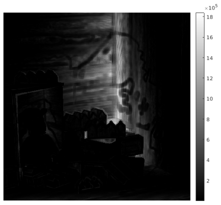
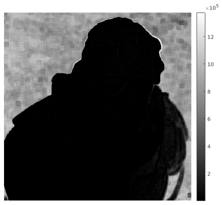
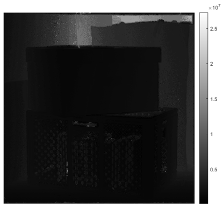
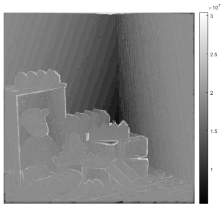
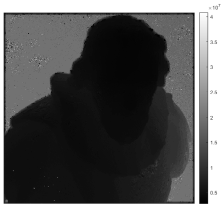
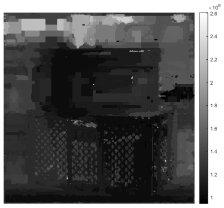
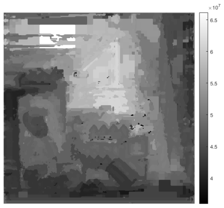
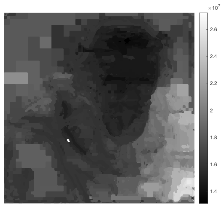
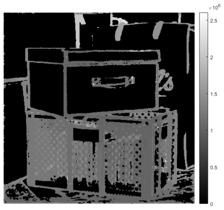
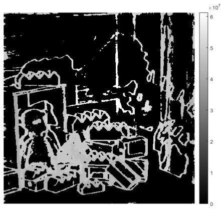
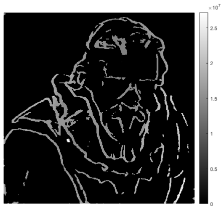
Algorithm	Boxes	Dino	Cotton
LSG			
Plane Sweeping			
EPI1			
EPI2			

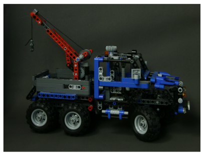
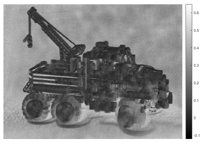
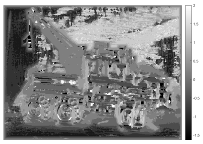
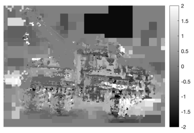
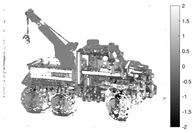
Table 3. Comparing PSNR and Run Times

Algorithm	Boxes PSNR	Boxes Runtime	Dino PSNR	Dino Runtime	Cotton PSNR	Cotton Runtime
LSG	22.1054	18.95s	26.6546	18.44s	19.3273	18.76s
Plane Sweeping	26.5306	349.14s	33.0201	322.78s	25.3360	352.01s
EPI1	25.4668	181.29s	30.6087	184.33s	20.7369	175.84s
EPI2	26.3023	-	32.9579	-	26.8590	-

have more depths to search through to find the optimal disparity. Meanwhile, looking at Figure 2 as the number of depths increases, the MSE decreases. This makes sense, since currently, this algorithm is sweeping through disparities between -2 and 2. Using these two figures, that although the mean-squared error decreases with increasing depths, it does not decrease by much since the scale is on the order of 1e-4. Therefore, the results were ran with a depth/plane number of 11 on the entire light field.

Overall, the images look visually appealing and very close to the ground truth, as shown in Table 1. In addition, looking the PSNRs in Table 3, the PSNRs are quite good and relatively high compared to other methods. In Table 2, the foreground is relatively dark (low differences from ground truth) for ’boxes’ and ’cotton’ but for ’dino’ the image is bright (larger differences from ground truth)

Table 4. Using Lytro Lego Truck from Stanford Light Field Dataset

Original	LSG	Plane Sweeping	EPI1	EPI2
				

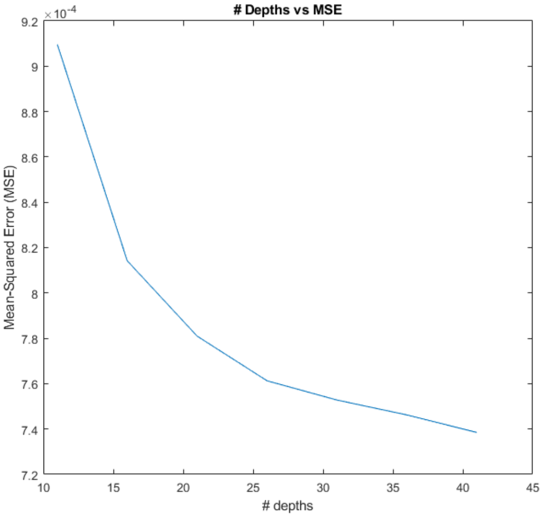


Figure 2. Depths vs. Mean-Squared Error (MSE)

overall. This may be because there are many occlusions in the 'dino' image, and thus, plane sweeping does not necessarily perform the best. However, despite its strong performance qualitatively, the run time is relatively high compared to other methods, about 3x that of the epipolar plane method and about 18x that of LSG.

Testing this out with the Lytro image from Stanford's Light Field Dataset (Table 4), it can be seen that the image is not as visually appealing. There are some artifacts around areas with more complex geometry, such as the wheels and the tubular interior of the truck. This demonstrates that the plane sweeping method may be best for smooth images and if there are complex geometries in the image, the performance decreases. In addition, the performance may break down around small occlusions (as shown around the wheels of the lego truck).

5.3. Epipolar-Plane and Fine-to-Coarse Refinement Method

From the results, we can see from Table 3 that the depth maps generated from EPI and Fine-to-coarse method have a PSNR higher than LSG method's but lower than Plane Sweeping Method's. On the other hand, the runtime is about 10 times higher than LSG but nearly half of plane sweeping. This suggest that the EPI and Fine-to-coarse method is more efficient and preserves the image quality at the same time. Similar to the LSG method, the depth estimation seems to rely on the features, which are the higher frequency in the data. The Table 2 shows that most of the error are from areas of background and with deeper area. This could be the results of the fact that when I is down-sampled to a very small dimensions, the depth bound strictly limit the the potential values of d, and leads to a uniform estimation of disparity. The general image quality looks good on the table 1. However, for the Lego truck Lytro data in Table 4, we can see that EPI1 is noisy but EPI2 is clear. This could be due to the fact that the downsampled images are highly effected by the noise or we should apply stronger Gaussian Blur to avoid aliasing.

6. Conclusion and Future Work

In conclusion, the LSG seemed to perform fairly well for the Heidelberg dataset and Stanford Light Field dataset and was the fastest algorithm. Plane sweeping performed really well for the Heidelberg dataset but poorly on the Stanford Light Field dataset. This is also seen in the epipolar plane method. Overall, if we look at run time and qualitative results from all the depth maps, LSG seemed to perform the best across diverse datasets, although its results are not perfect.

If we had more time, here is what we would focus on:

1. Test out other types of datasets, where theres more variation in illumination, fine detail, etc.

2. Test plane sweeping using different filters besides box filter (such as bilateral filtering or median filtering)

3. Use confidence values from plane sweeping to create more robust depth maps against occlusion

4. Fine-tune parameters in the epipolar method

Overall, the algorithms all performed relatively well

References

[1] E. Adelson and J. Wang. Single lens stereo with a plenoptic camera. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 14(2):99–106, 1992.

[2] R. Bolles, H. Baker, and D. Marimont. Epipolar-plane image analysis: an approach to determining structure from motion. 1987.

[3] K. Honauer, O. Johannsen, D. Kondermann, and B. Goldluecke. A dataset and evaluation methodology for depth estimation on 4d light fields. *Asian Conference on Computer Vision. Springer, Cham*, 2016.

[4] O. W. K. Yucer, A. Sorkine-Hornung and O. Sorkine-Hornung. Efficient 3d object segmentation from densely sampled light fields with applications to 3d reconstruction. *ACM Transactions on Graphics*, 35(3):1–15, 2016.

[5] C. Kim, H. Zimmer, Y. Pritch, A. Sorkin-Hornung, and M. Gross. Scene reconstruction from high spatio-angular resolution light fields. *ACM Transactions on Graphics*, 2013.

[6] T. Wang, A. Efros, and R. Ramamoorthi. Occlusion-aware depth estimation using light-field cameras. *ICCV*, 2015.

[7] S. Wanner and B. Goldlucke. Globally consistent depth labeling of 4d light fields. 2012.

7. Appendix

Overall, the work was distributed evenly amongst everyone on the team. Warren Cheng was responsible for the LSG algorithm, putting together the poster, and writing the report. Linda Banh was responsible for the plane sweeping algorithm, putting together the poster, and writing the report. Fang-Yu Lin was responsible for the epipolar plane and fine-to-coarse refinement algorithm and putting together the report.