Light Field from Micro-baseline Image Pair (Supplementary Material)

Anonymous CVPR submission

Paper ID 2231

This document provides some more results and analysis of our proposed method. We also provides a supplemental video for showing the video rendering of the synthesized views and the refocus application.

1. Iterations and Convergence Analysis

Fig.1 and Fig.2 show how the dispairty maps and the P-SNR values of the synthesized right view are updated with the iterations. Since we approximate ω_0 by a smaller value, for each iteration we underestimate d_{err} by a constant factor c. Therefore, the iteration would demonstrates a geometric convergence with a ratio of c. As shown in Fig 1 and 2. These results demonstrate the convergence of our iterative optimization.

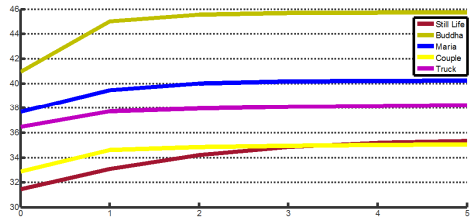

Figure 2. Convergence in terms of reconstruction PSNR.

2. Maximum Input Disparity

The proposed method is robust to the baseline between the two input images. Table 1 lists the disparity range of each data used in this work. The absolute disparity values range from smaller than half of a pixel to eight pixels.

3. 4D Light Field Quality

We provide a PSNR evaluation of the synthesized 4D lightfield for the Maria data set. The result is shown in Fig.3. It is reasonable here that, with the increasing of s-cale factor, the synthesized quality is degraded.

	Synthetic Scenes		Real-World Scenes			
	still life	buddha	maria	couple	truck	gum
Min	-3.0064	-1.083	-0.9619	-1.6720	-4.9582	-8.2001
Max	2.7609	1.558	0.4007	1.9981	3.1464	4.1479

Table 1. Disparity ranges for various data sets. Min and Max s-tands for minimum and maximum disparities of the input stereo pair.

Figure 3. PSNR evaluation of the synthesized “Maria” 4D light field.

4. Complexity Analysis

The time complexity of our method is determined by the integration for synthesizing the bands of novel view:

$$b_n^i(\mathbf{x}) = \frac{1}{2\pi}\int L(\boldsymbol{\omega})G_i(\boldsymbol{\omega})e^{i\boldsymbol{\omega}\cdot\mathbf{d}(\mathbf{k}_0)}e^{i\boldsymbol{\omega}\cdot\mathbf{x}}\,d\boldsymbol{\omega}. \quad (1)$$

We can write this integration in a matrix form. Element $b_n^i|_{xy}$ at row x column y in matrix b_n^i can be calculated by:

$$b_n^i|_{xy} = (M_x(LG_ie^{i\boldsymbol{\omega}\cdot\mathbf{d}(\mathbf{k}_0)})M_y)|_{xy}, \quad (2)$$

Where $\mathbf{d}(\mathbf{k}_0)$ is a constant with given xy, and M_x M_y are 2D inverse DFT matrices. Since we only need the element at row x column y of the calculation result, only row x in M_x and column y in M_y are required for the above computation. Therefore, the time complexity for calculating $b_n^i|_{xy}$

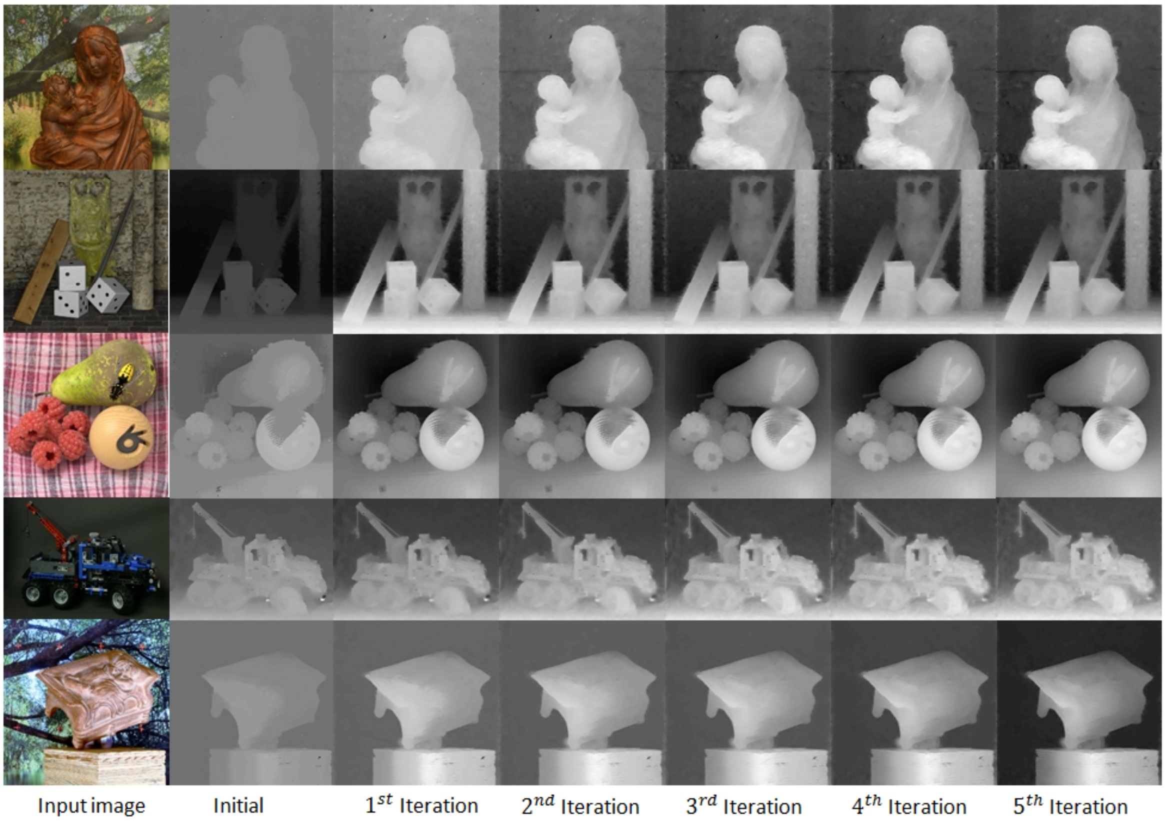

Figure 1. Depth refinement iteration result.

would be $O(n^2)$ for a image of $n \times n$ pixels,which gives a overall complexity of $O(n^4)$.

However, if $\mathbf{d}(\mathbf{k}_0)$ is along x or y axis, the complexity can be reduced. Suppose $\mathbf{d}(\mathbf{k}_0)$ is along y axis, then Eqn.2 becomes:

$$b_n^i|_{xy} = (M_x L G_i) e^{i\omega_y d(\mathbf{k}_0)} M_y)|_{xy}. \tag{3}$$

We can first compute $(M_x L G_i)$, which is a constant matrix for all xy, and then use the row x of the result with column y of M_y to calculate $b_n^i|_{xy}$. For each $b_n^i|_{xy}$, only n multiplications are needed. Therefore, the overall complexity is reduced to $O(n^3)$.

5. Full Resolution Images

Full resolution images of our zoom-in comparison (Fig.7) can be found in the 'images' folder, all in .ppm format.

Light Field from Micro-baseline Image Pair
(Supplementary Material)

Zhoutong Zhang, Yebin Liu, Qionghai Dai
Beijing Key Laboratory of Multi-dimension & Multi-scale Computational Photography (MMCP),
Tsinghua University, Beijing 100084 China

This document provides some more results and analysis of our proposed method. We also provides a supplemental video for showing the video rendering of the synthesized views and the refocus application.

1. Iterations and Convergence Analysis

Fig.1 and Fig.2 show how the dispairty maps and the P-SNR values of the synthesized right view are updated with the iterations. Since we approximate $\boldsymbol{\omega_0}$ by a smaller value, for each iteration we underestimate d_{err} by a constant factor c. Therefore, the iteration would demonstrates a geometric convergence with a ratio of c. As shown in Fig 1 and 2. These results demonstrate the convergence of our iterative optimization.

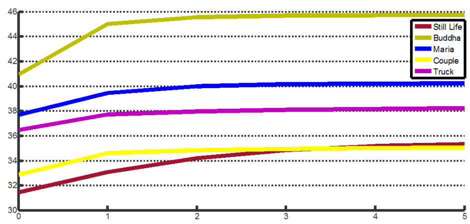

Figure 2. Convergence in terms of reconstruction PSNR.

2. Maximum Input Disparity

The proposed method is robust to the baseline between the two input images. Table 1 lists the disparity range of each data used in this work. The absolute disparity values range from smaller than half of a pixel to eight pixels.

3. 4D Light Field Quality

We provide a PSNR evaluation of the synthesized 4D lightfield for the Maria data set. The result is shown in Fig.3. It is reasonable here that, with the increasing of s-cale factor, the synthesized quality is degraded.

	Synthetic Scenes		Real-World Scenes			
	still life	buddha	maria	couple	truck	gum
Min	-3.0064	-1.083	-0.9619	-1.6720	-4.9582	-8.2001
Max	2.7609	1.558	0.4007	1.9981	3.1464	4.1479

Table 1. Disparity ranges for various data sets. Min and Max s-tands for minimum and maximum disparities of the input stereo pair.

Figure 3. PSNR evaluation of the synthesized “Maria” 4D light field.

4. Complexity Analysis

The time complexity of our method is determined by the integration for synthesizing the bands of novel view:

$$b_n^i(\mathbf{x}) = \frac{1}{2\pi} \int L(\boldsymbol{\omega}) G_i(\boldsymbol{\omega}) e^{i\boldsymbol{\omega}\cdot\mathbf{d}(\mathbf{k}_0)} e^{i\boldsymbol{\omega}\cdot\mathbf{x}} \, d\boldsymbol{\omega}. \quad (1)$$

We can write this integration in a matrix form. Element $b_n^i|_{xy}$ at row x column y in matrix b_n^i can be calculated by:

$$b_n^i|_{xy} = (M_x(LG_i e^{i\boldsymbol{\omega}\cdot\mathbf{d}(\mathbf{k}_0)})M_y)|_{xy}, \quad (2)$$

Where $\mathbf{d}(\mathbf{k}_0)$ is a constant with given xy, and M_x M_y are 2D inverse DFT matrices. Since we only need the element at row x column y of the calculation result, only row x in M_x and column y in M_y are required for the above computation. Therefore, the time complexity for calculating $b_n^i|_{xy}$

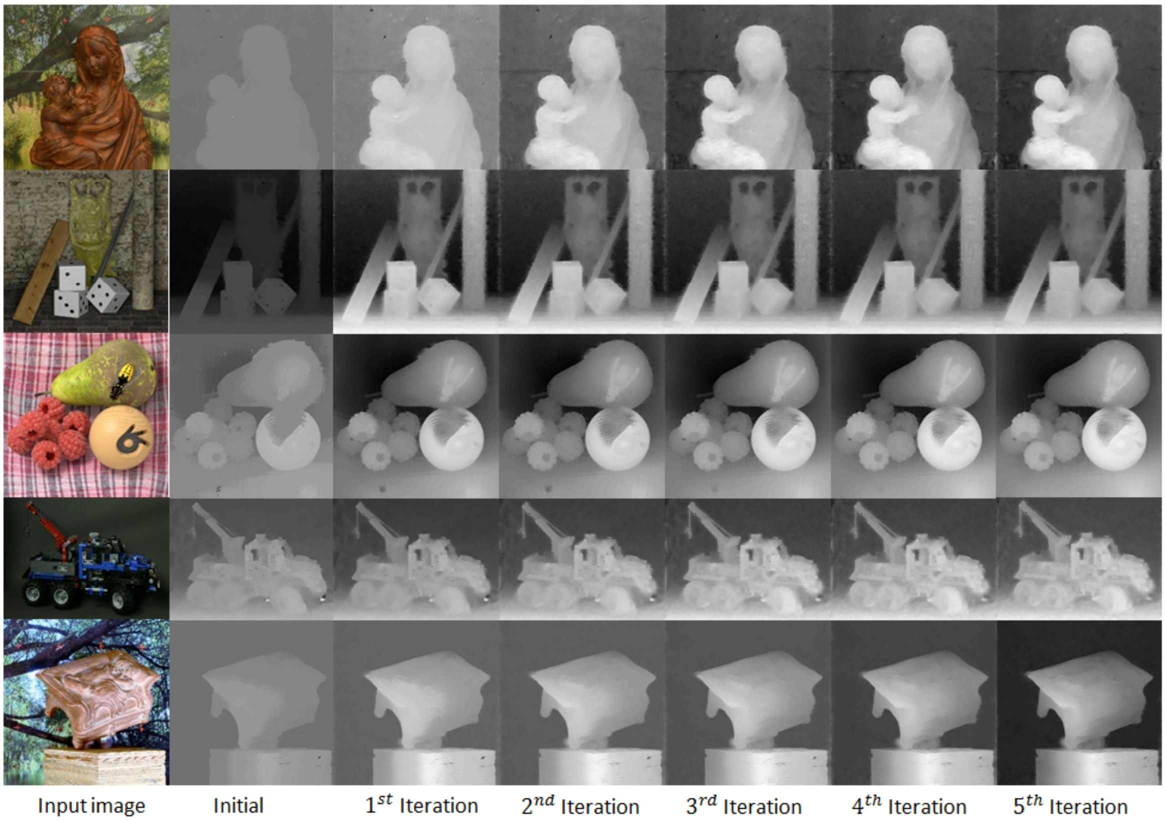

Figure 1. Depth refinement iteration result.

would be $O(n^2)$ for a image of $n \times n$ pixels,which gives a overall complexity of $O(n^4)$.

However, if $\mathbf{d}(\mathbf{k}_0)$ is along x or y axis, the complexity can be reduced. Suppose $\mathbf{d}(\mathbf{k}_0)$ is along y axis, then Eqn.2 becomes:

$$b_n^i|_{xy} = (M_x L G_i) e^{i\omega_y d(\mathbf{k}_0)} M_y)|_{xy}. \tag{3}$$

We can first compute $(M_x L G_i)$, which is a constant matrix for all xy, and then use the row x of the result with column y of M_y to calculate $b_n^i|_{xy}$. For each $b_n^i|_{xy}$, only n multiplications are needed. Therefore, the overall complexity is reduced to $O(n^3)$.

5. Full Resolution Images

Full resolution images of our zoom-in comparison (Fig.7) can be found in the 'images' folder, all in .ppm format.