Stereo Matching

Our Team

R07921052 劉兆鵬

Researcher

R06921081 張邵瑀

Programmer

R06921058 方浩宇

System Analysis

Outline

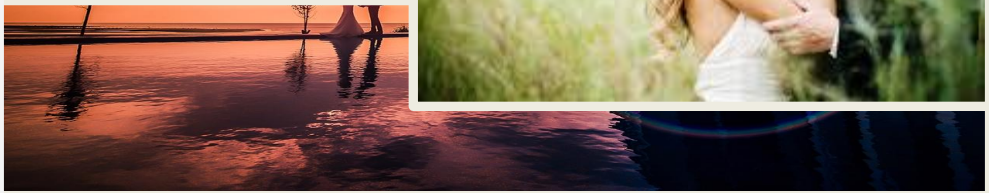
- Introduction
- Method
- Future Refinement
- Disparity map result

Introduction

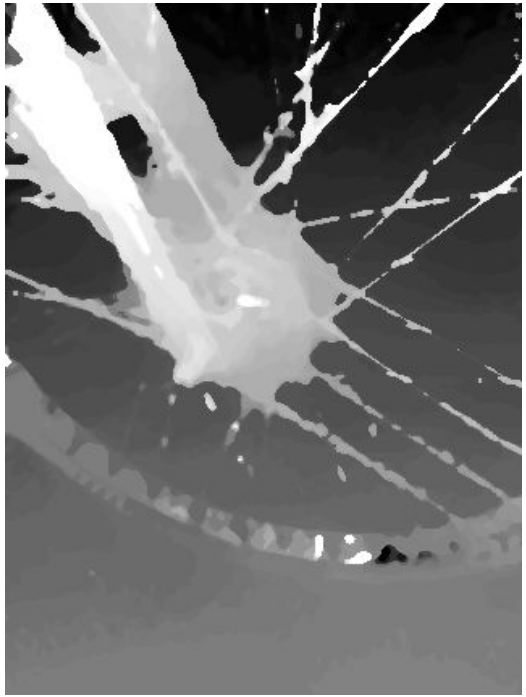


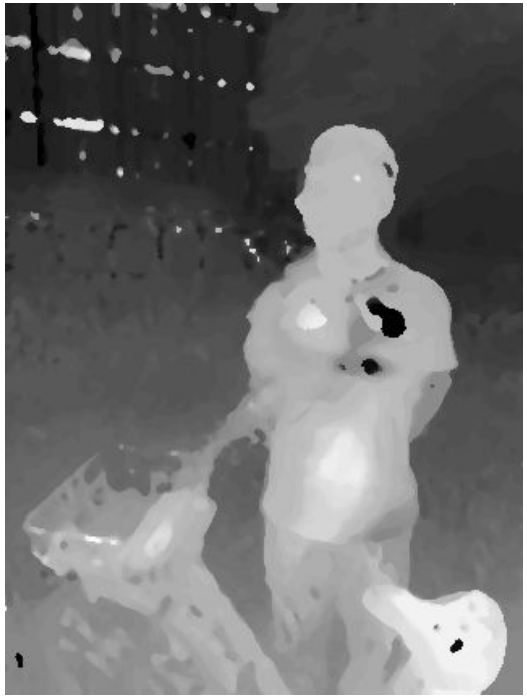




Result Preview





Method pipeline

Data preprocessing

Get max disparity

Intensity interpolation

Census Hamming distance (+ SAD)

Local voting

Bi-Layer Winner-take-all

Hole filling(?)

Weighted-median-filter

Sub-Pixel

01 The structure is similar

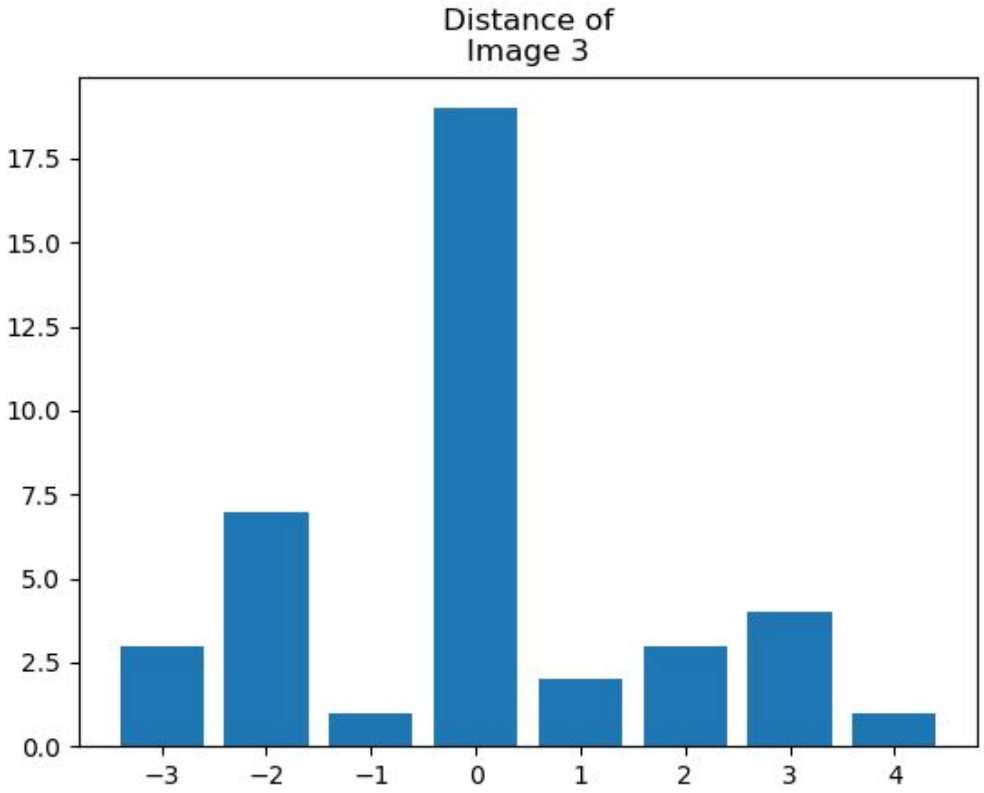
02 The focal length is too long

03 The range of intensity is too limit

→ **Intensity Interpolation**

- to increase structure information

Sub-Pixel Cont.



01

Determine
max disparity

02

Get ratio between max
disparity and SIFT

03

Scaling of interpolation

Left-Right Scanning

01 Some distance of keypoints are negative

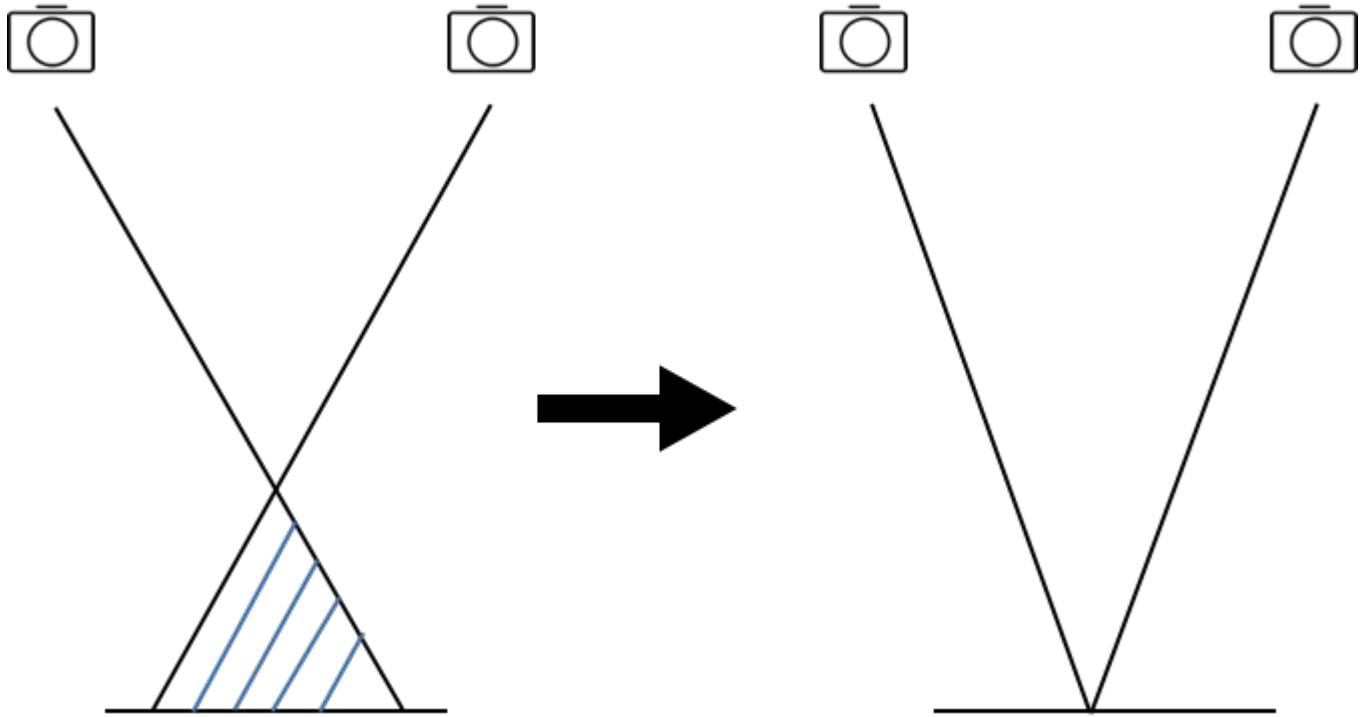
02 The infinite points are not the same

03 The images are cross together

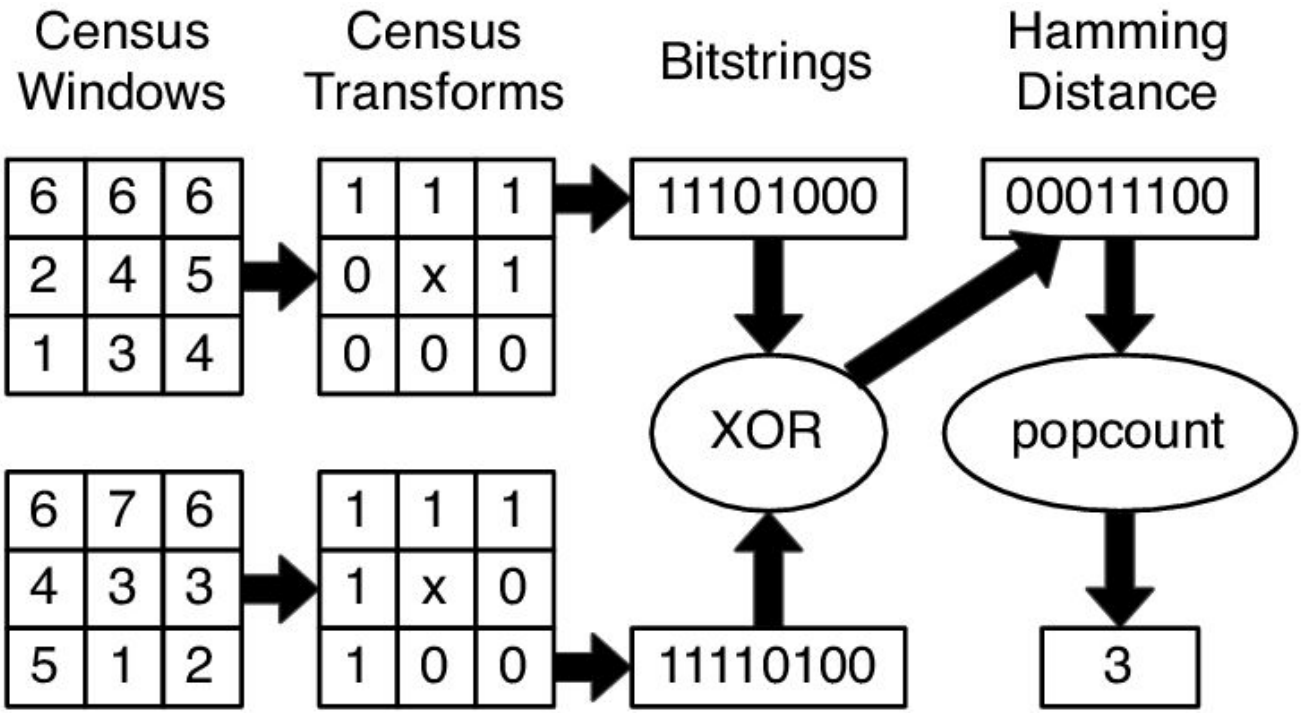
Left-Right Scanning

- **to get the cost between foreground and background**

Left-Right Scanning Cont.

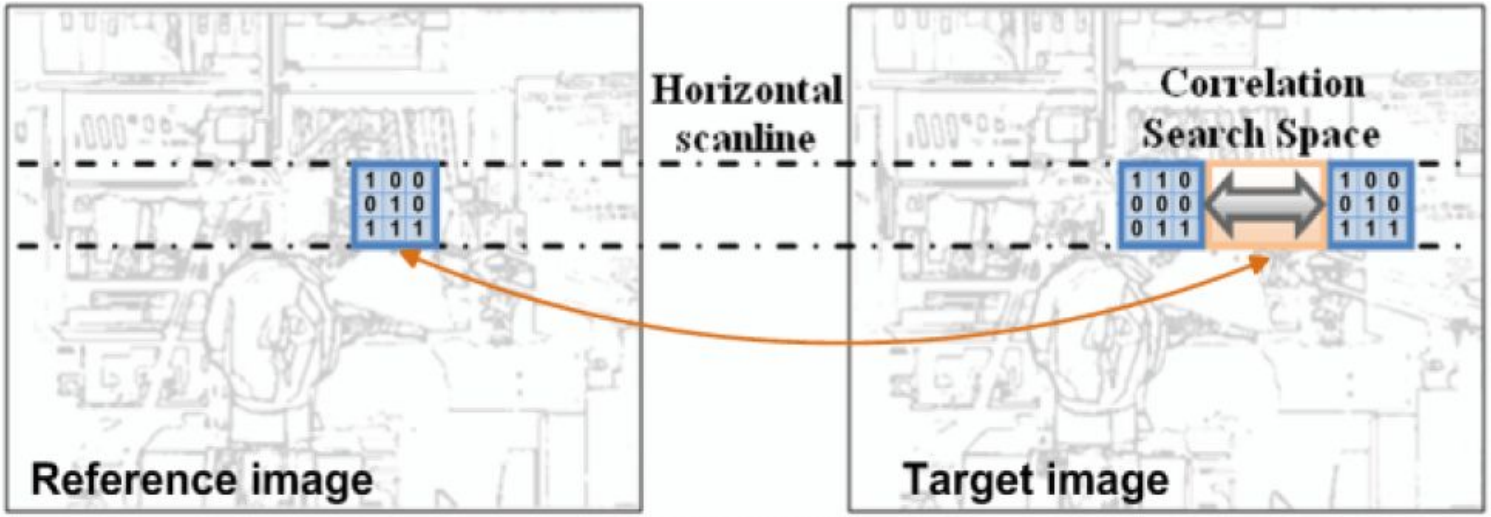


Census Transform



Reference: *Pragma based parallelization - Trading hardware efficiency for ease of use?*, ***Tobias Kenter***

Block SAD



$$SAD: \Sigma|T_i - R_i| = |1-1|+|0-1|+|0-0|+|0-0|+|1-0|+|0-0|+|1-0|+|1-1|+|1-1|$$
$$= 0+1+0+0+1+0+1+0+0=3$$

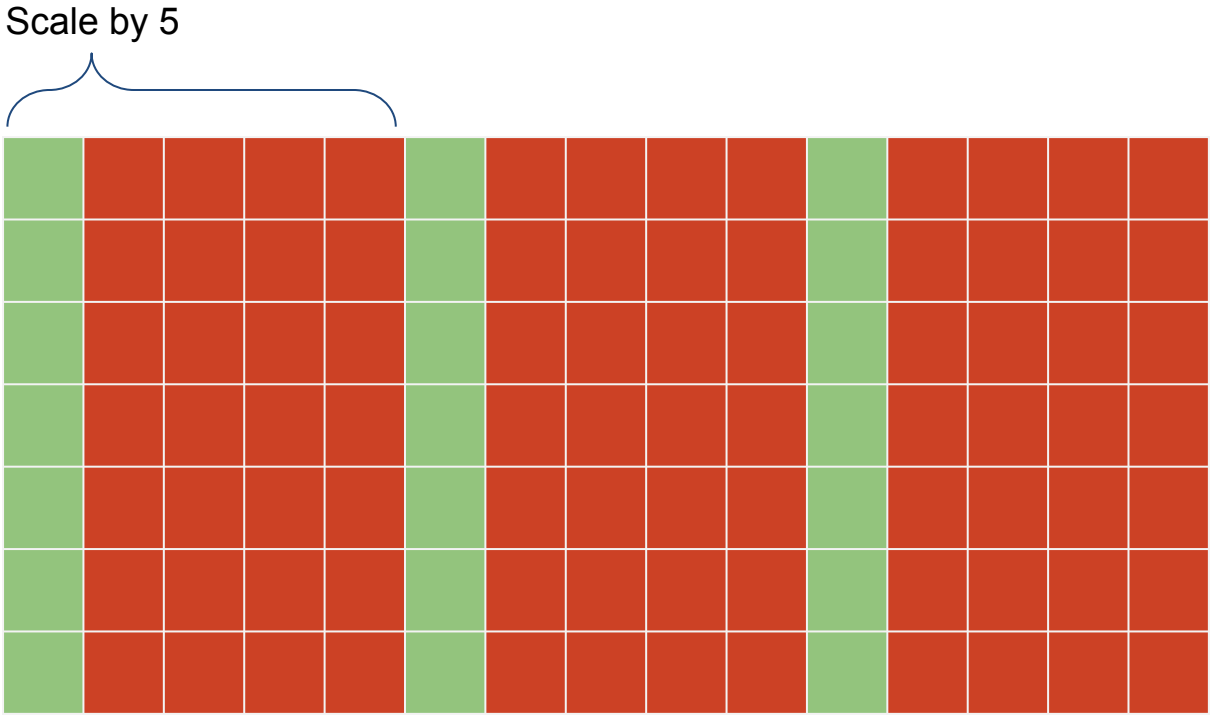
Reference: http://doi.ieeecomputersociety.org/cms/Computer.org/dl/trans/tc/2013/04/figures/ttc20130406904.gif

Bi-Layer Winner-take-all

Local voting

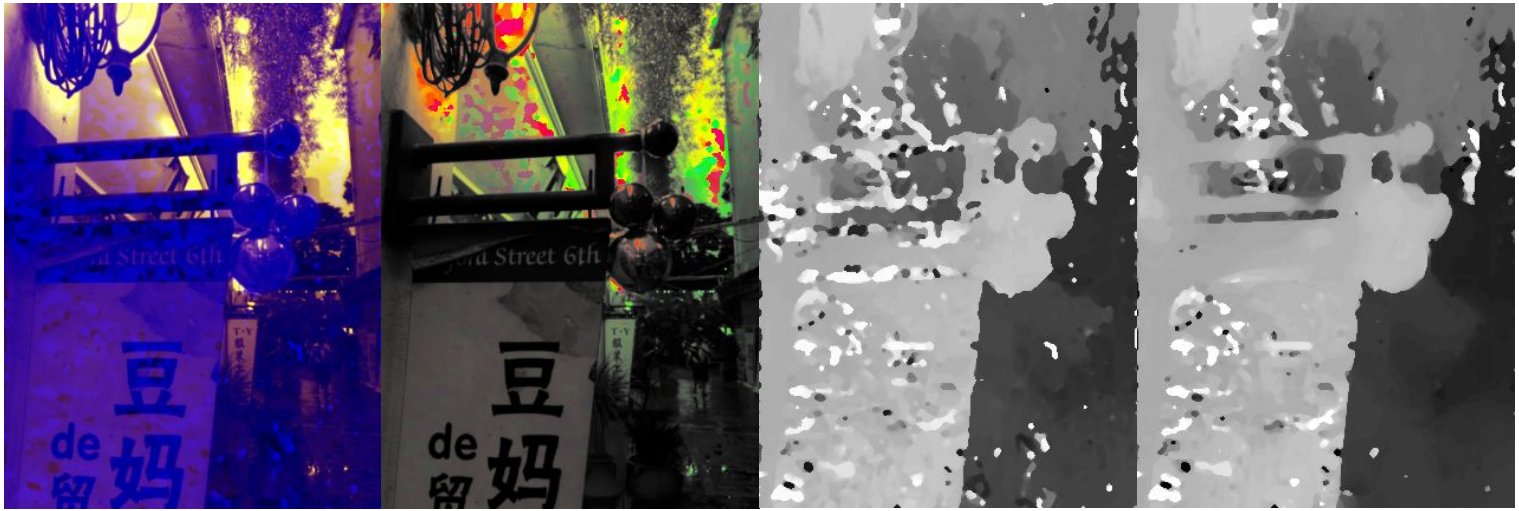
- Resize before winner-takes-all, and do voting by scale.

- Sum
- Min
- Median



Color space transformation

- Use the disparity map from winner-take-all with first channel
- Use the original one channel picture normalize square with second channel
- Use the original one channel picture with third channel
- Change color space from HSV to RGB with cv2.COLOR_HSV2RGB_FULL
- Use this as guide img to do weighted-median-filter



Future Refinement

01 Clustering each object and filling hole

02 Filling the hole of neighborhood of each disparity number

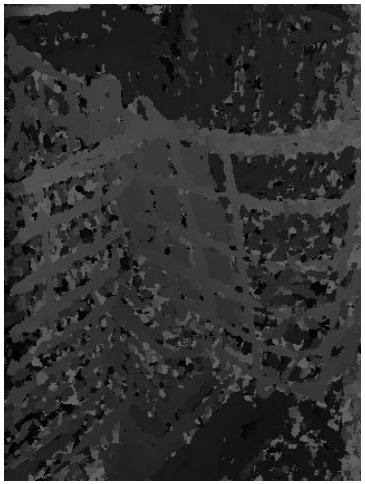
03 Hole filling by background and foreground disparity map

→ Time limitation

Future Refinement

Foreground

background



Before filling

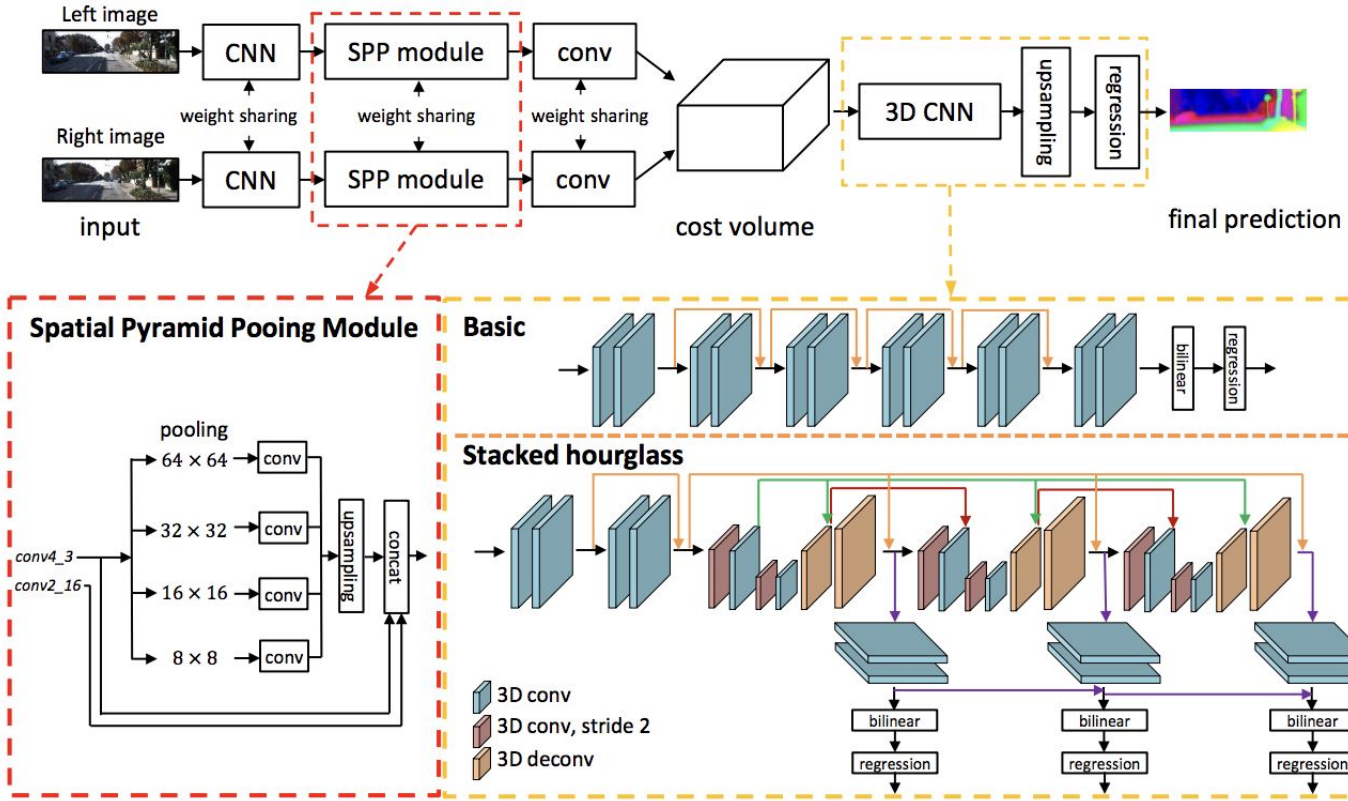
Synthetic method

01 Using PSMNet deep learning model

02 Training on SceneFlow dataset
- FlyTing3D

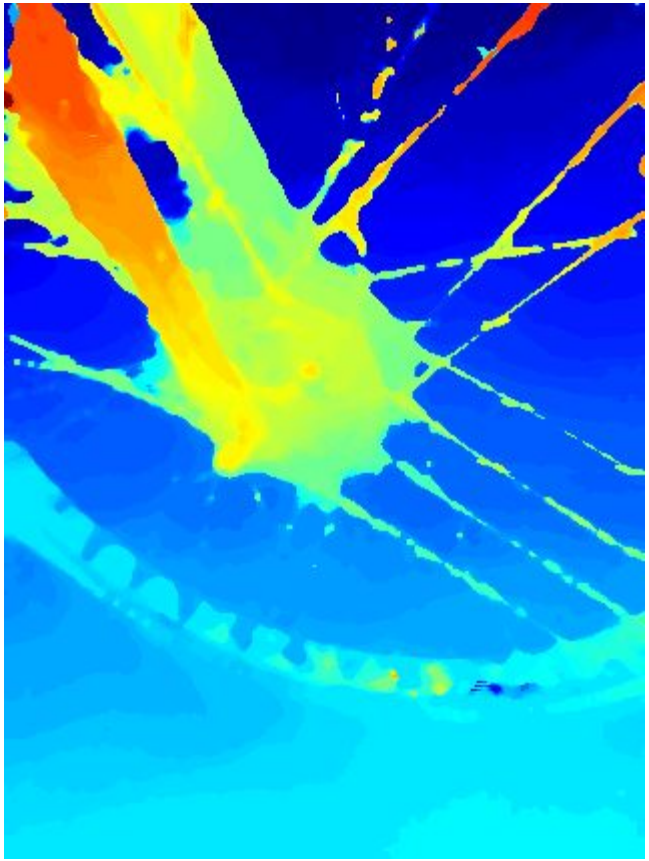
03 Fine tune on our final datasets

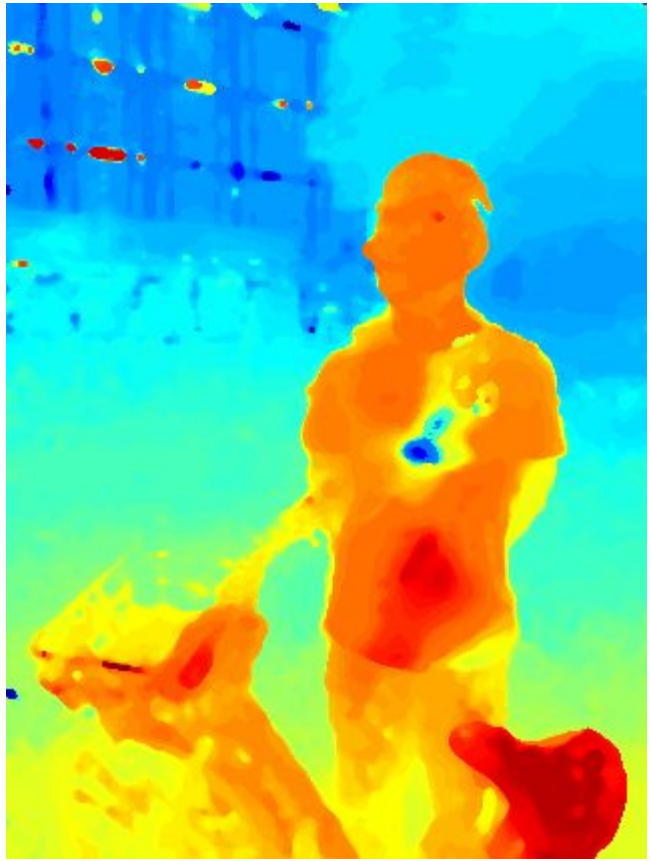
Model Structure

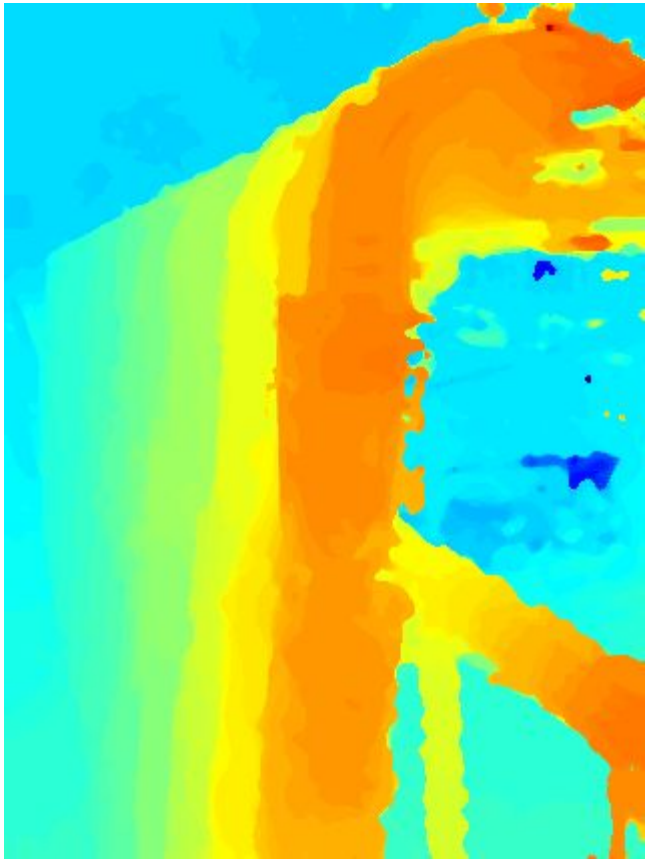


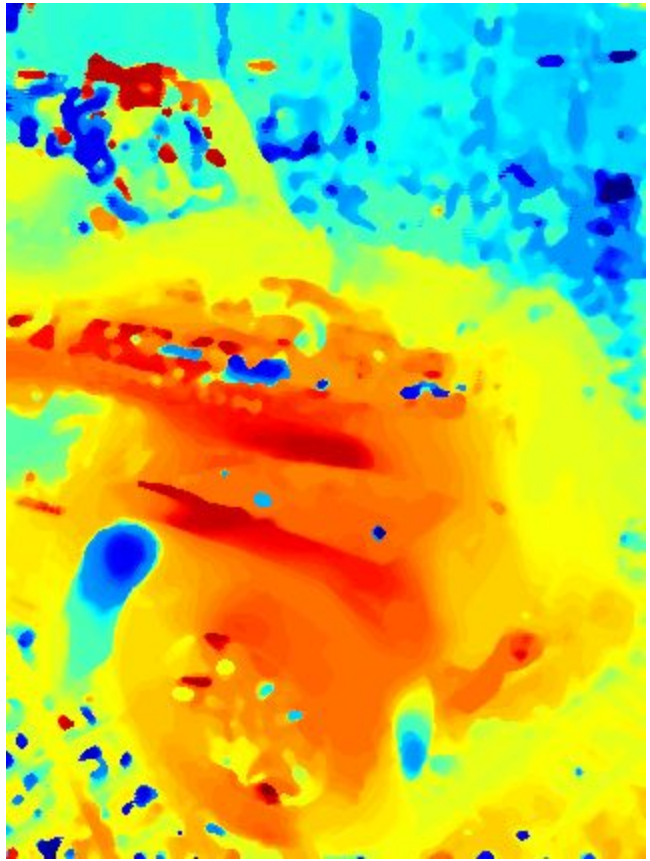
Reference: https://arxiv.org/pdf/1803.08669.pdf

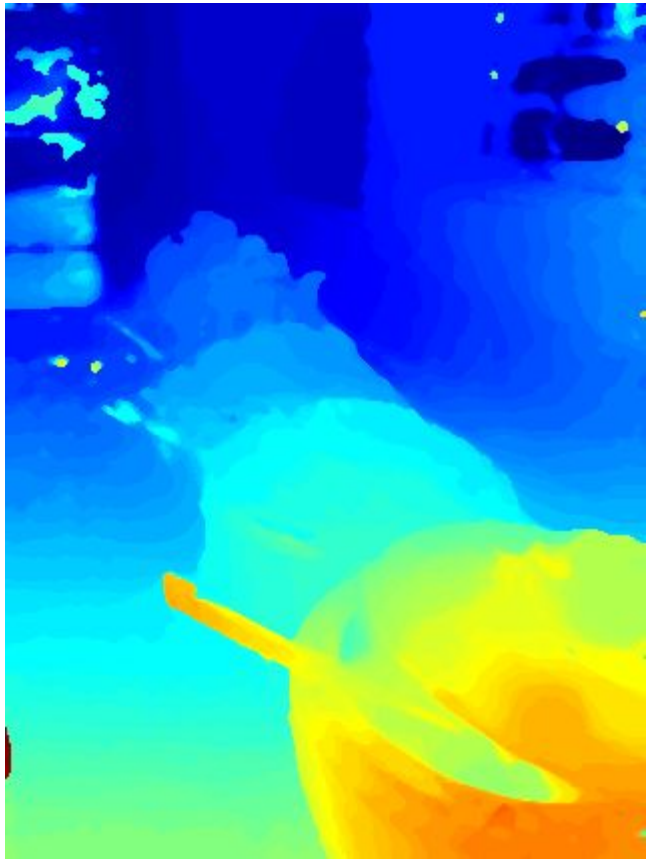
Disparity Map Result

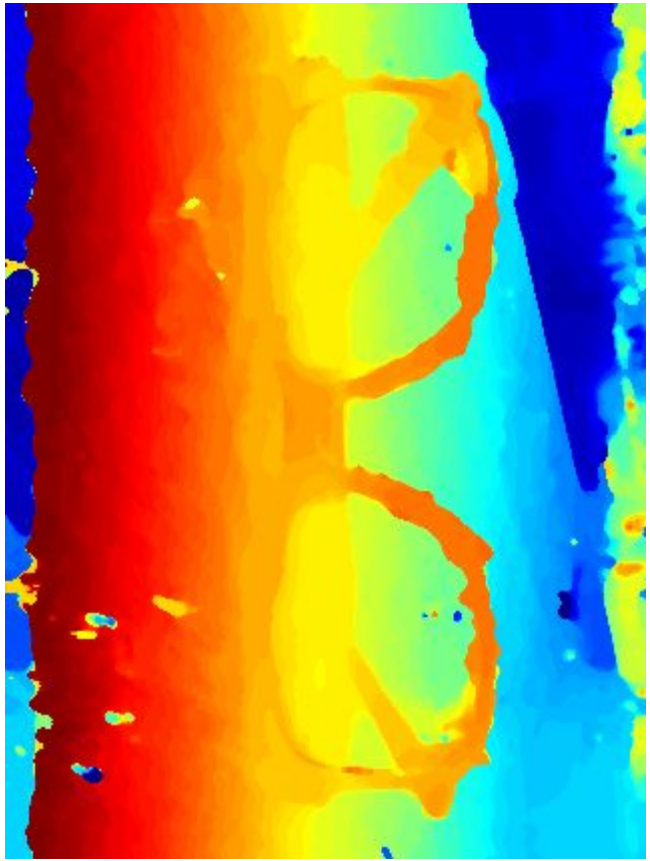


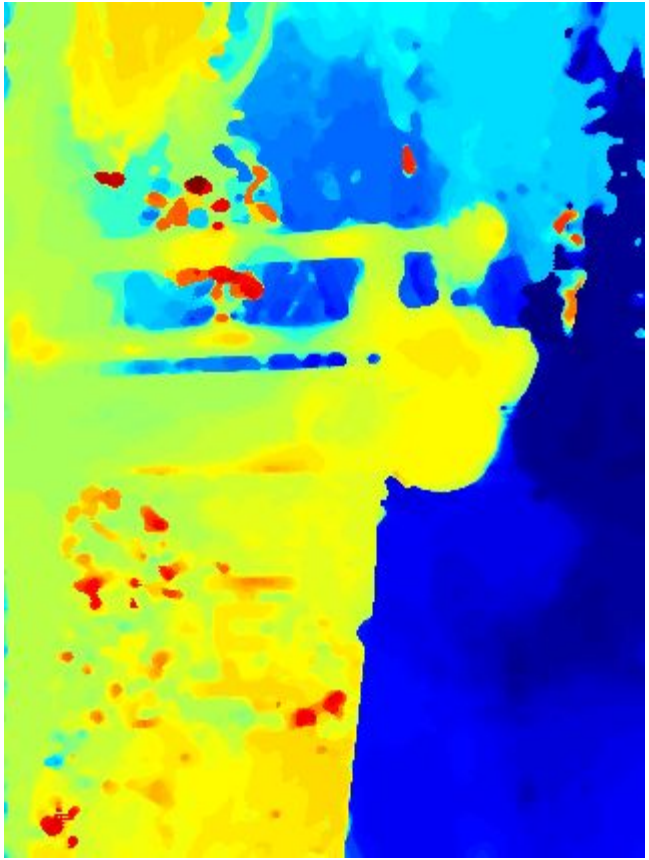


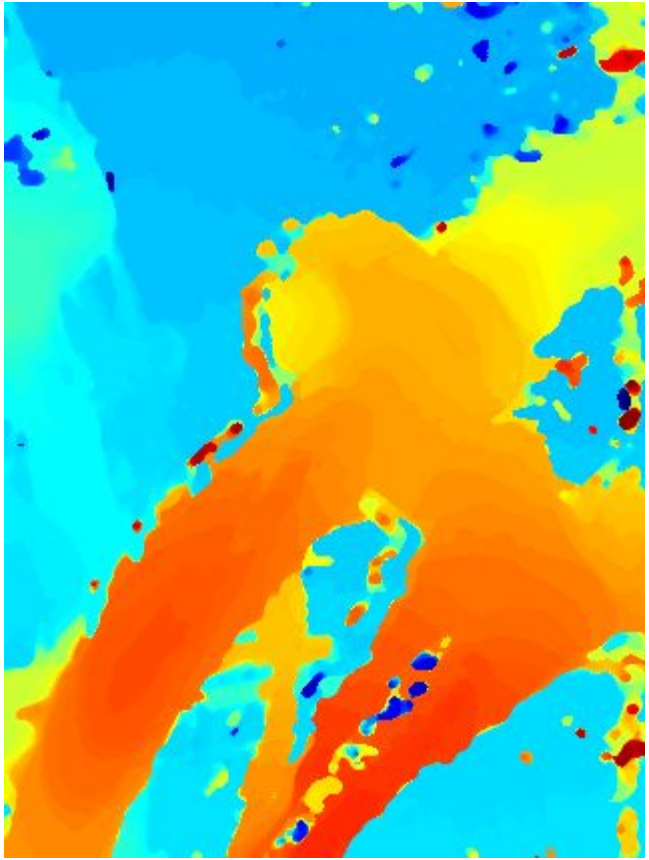


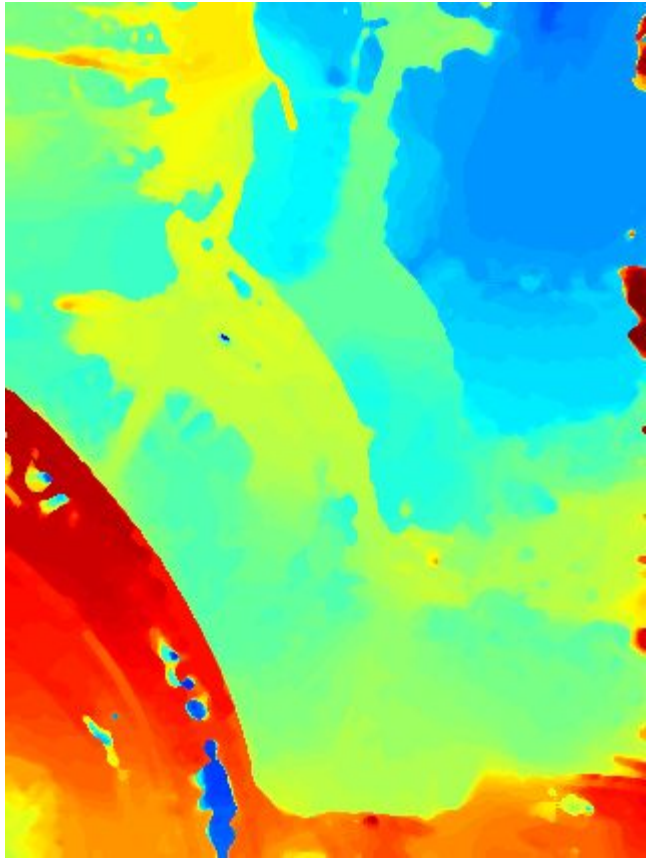


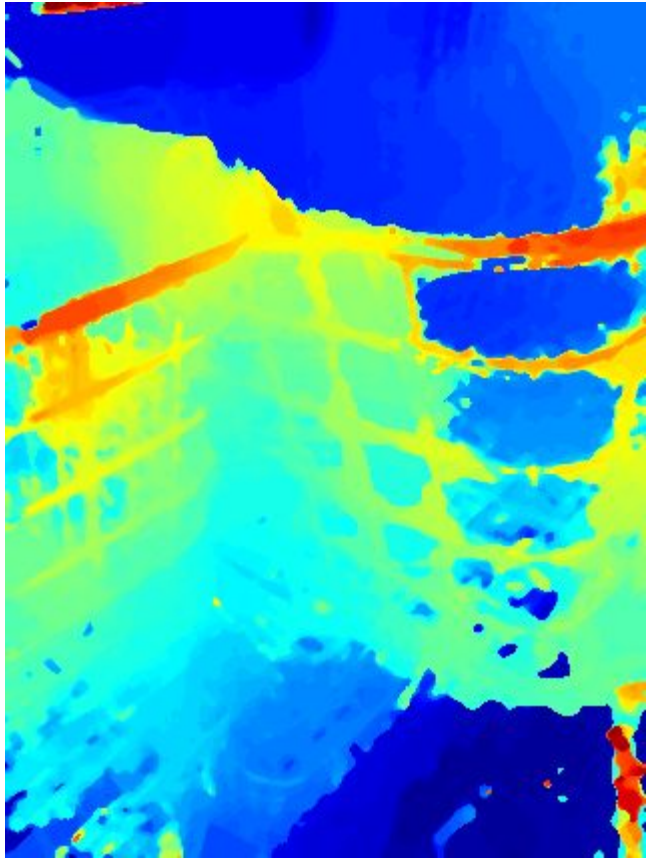


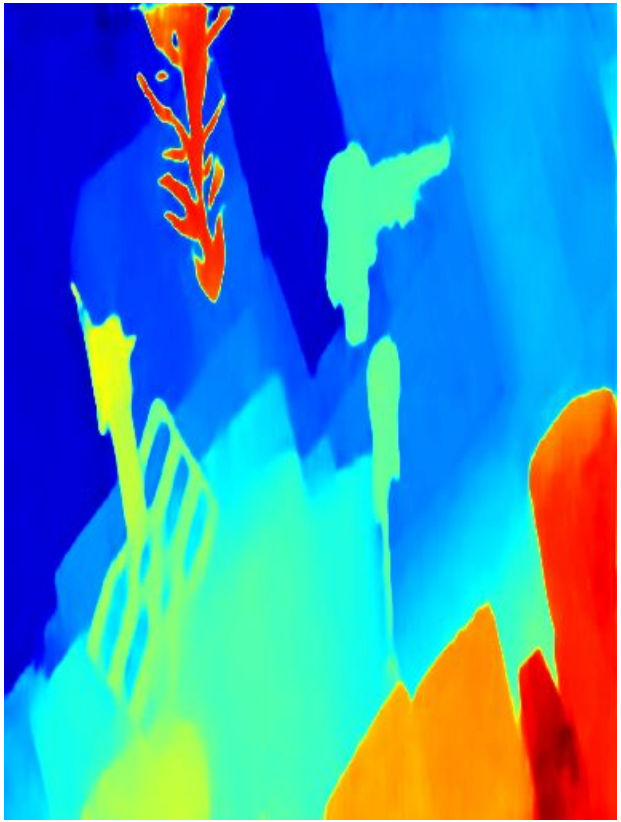




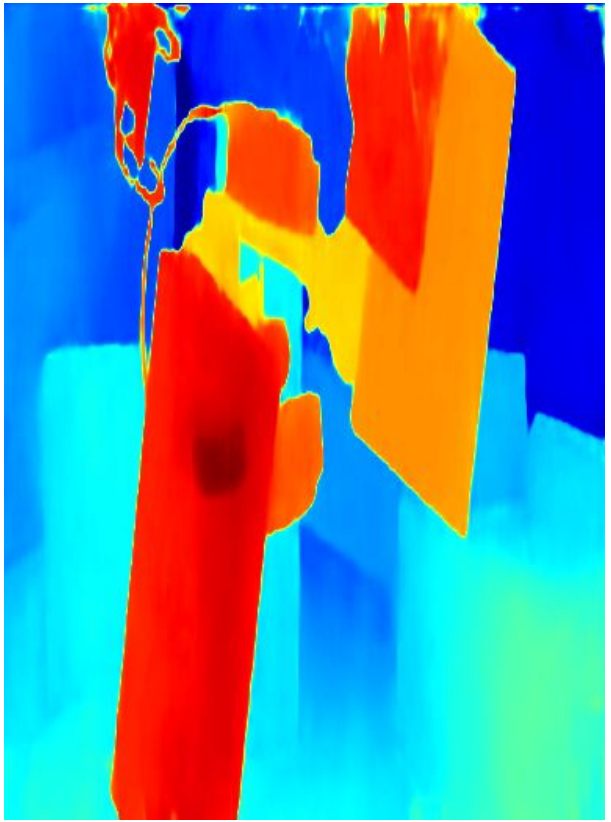




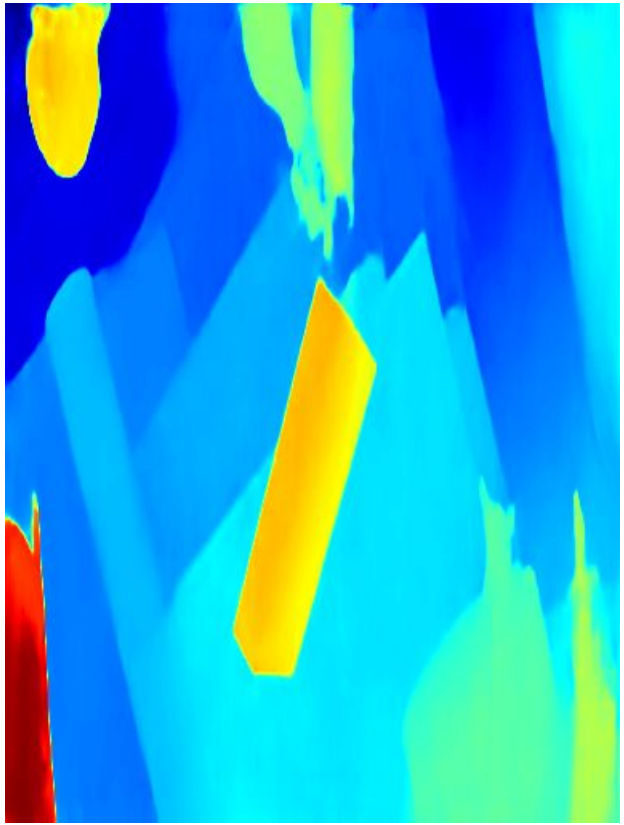




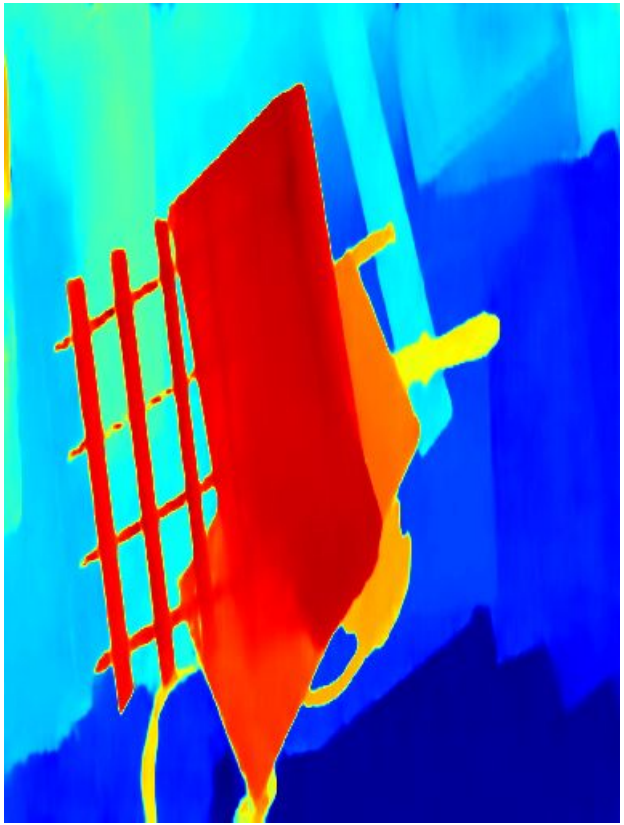
0.696



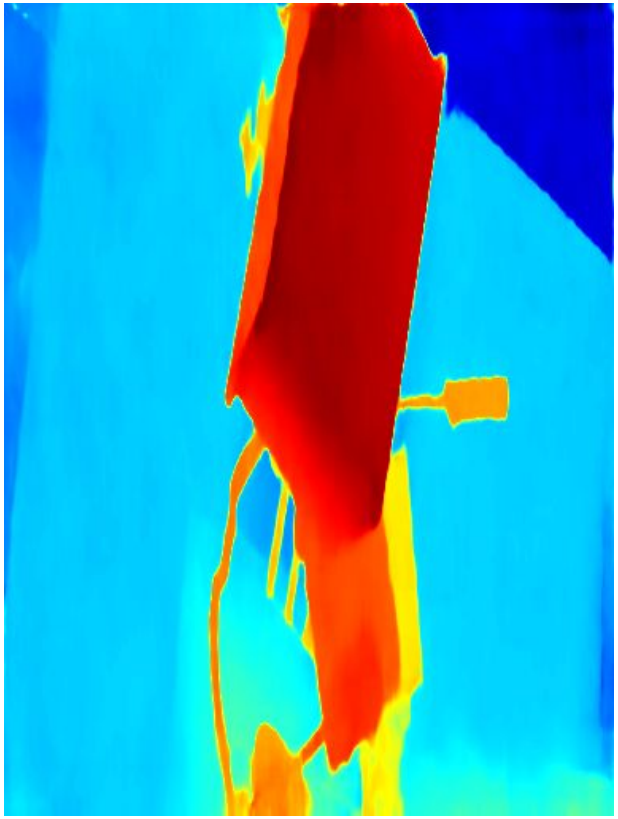
0.698



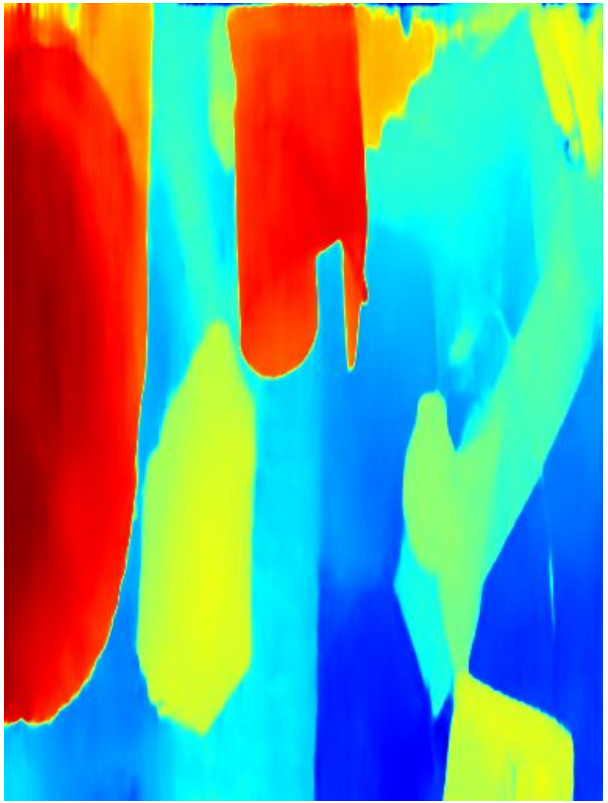
0.698



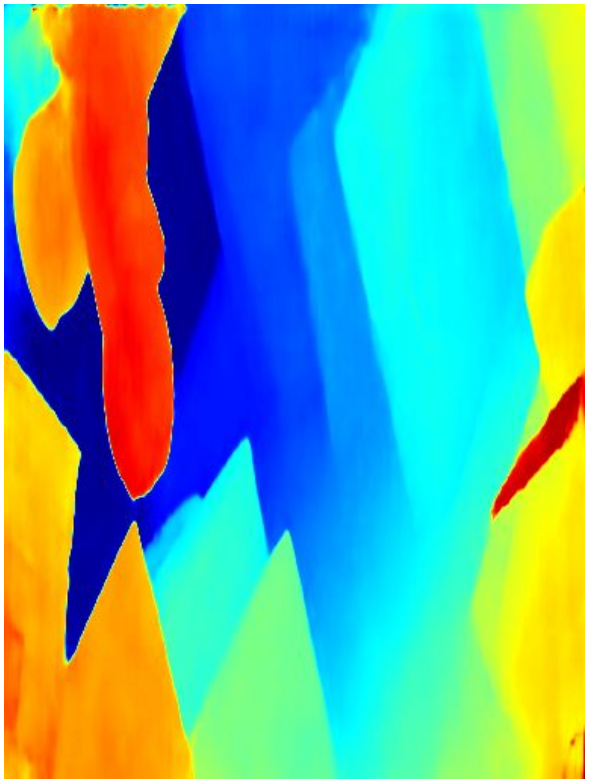
0.747



0.687

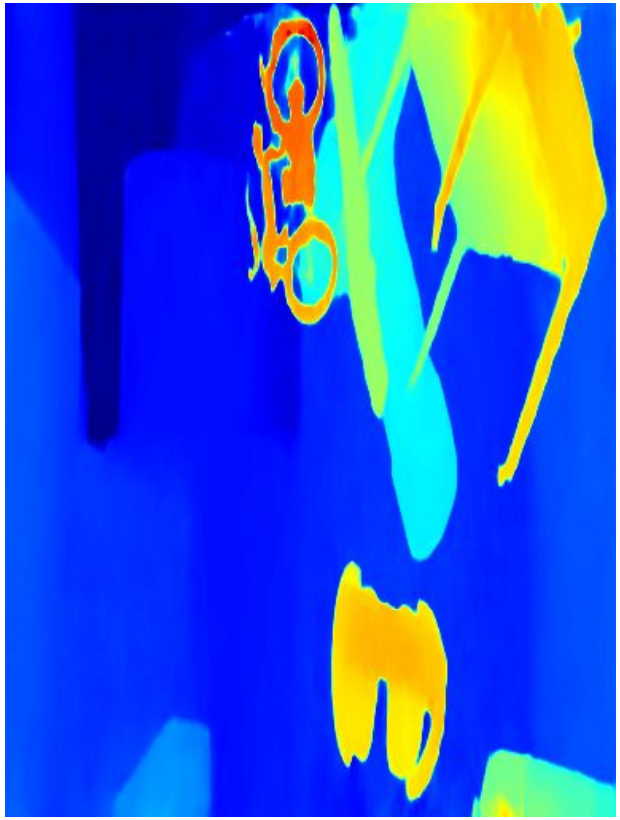


0.779



0.623

0.75



0.87

0.765

Thank You !