# Test Results

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| Test set: | Motorcycle | Terrace | Playground | Sword |
| PSNR of masked disparity map1(dB) | 25.12 | 28.18 | 25.92 | 13.70 |
| PSNR of unmasked disparity map(dB) | 20.05 | 7.28 | 11.01 | 12.30 |
| Total time consumption | 1231s | 29.25s | 42.93s | 1596s |

1: Masking indicates removing the invalid points in the ground truth from the disparity map.

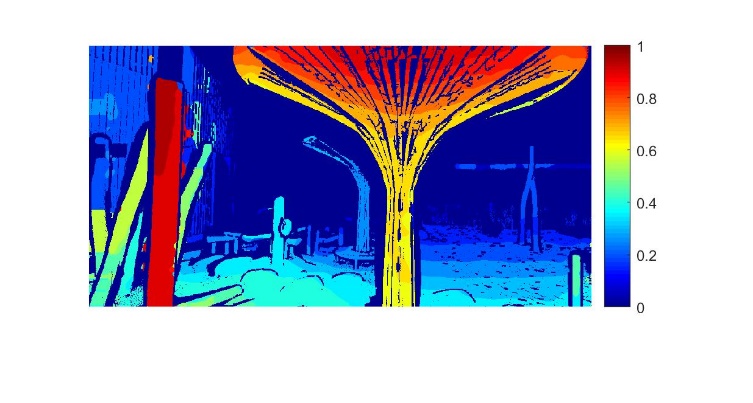
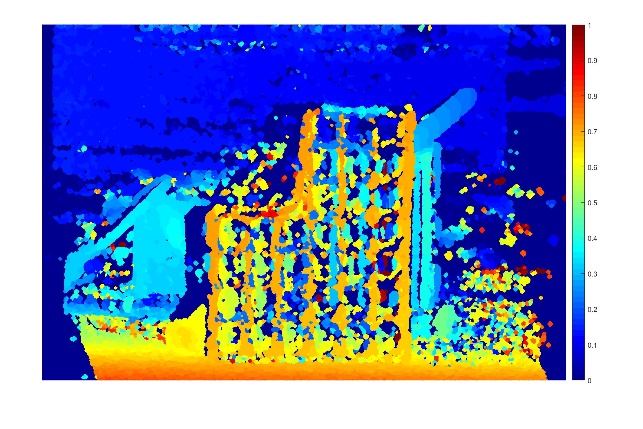
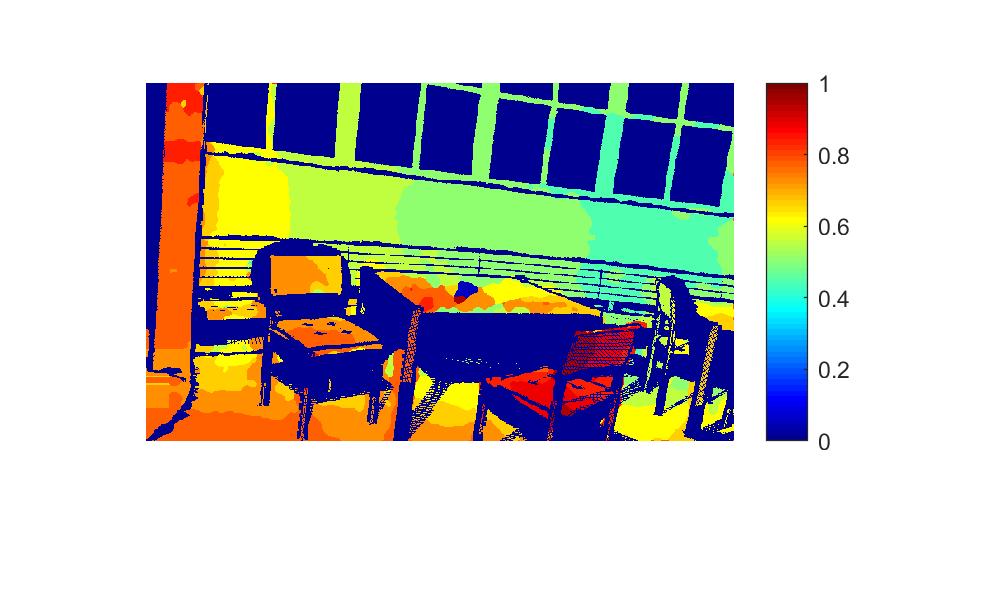
The time consumption is available in the following report.

Test environment: CPU:

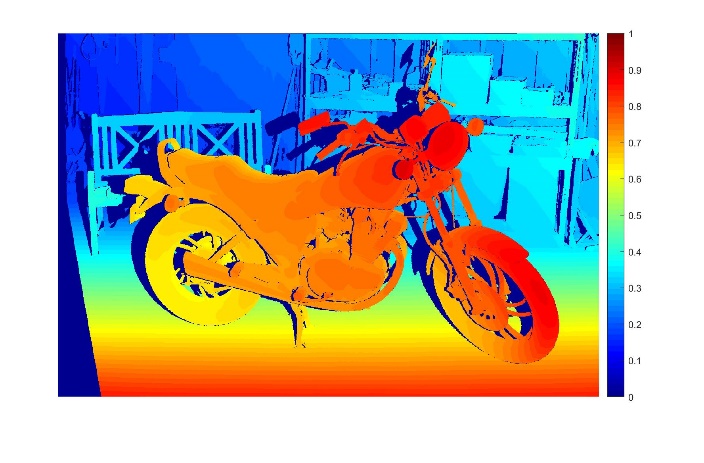
Intel [i7-6700H@2.60GHz](mailto:i7-6700H@2.60GHz)

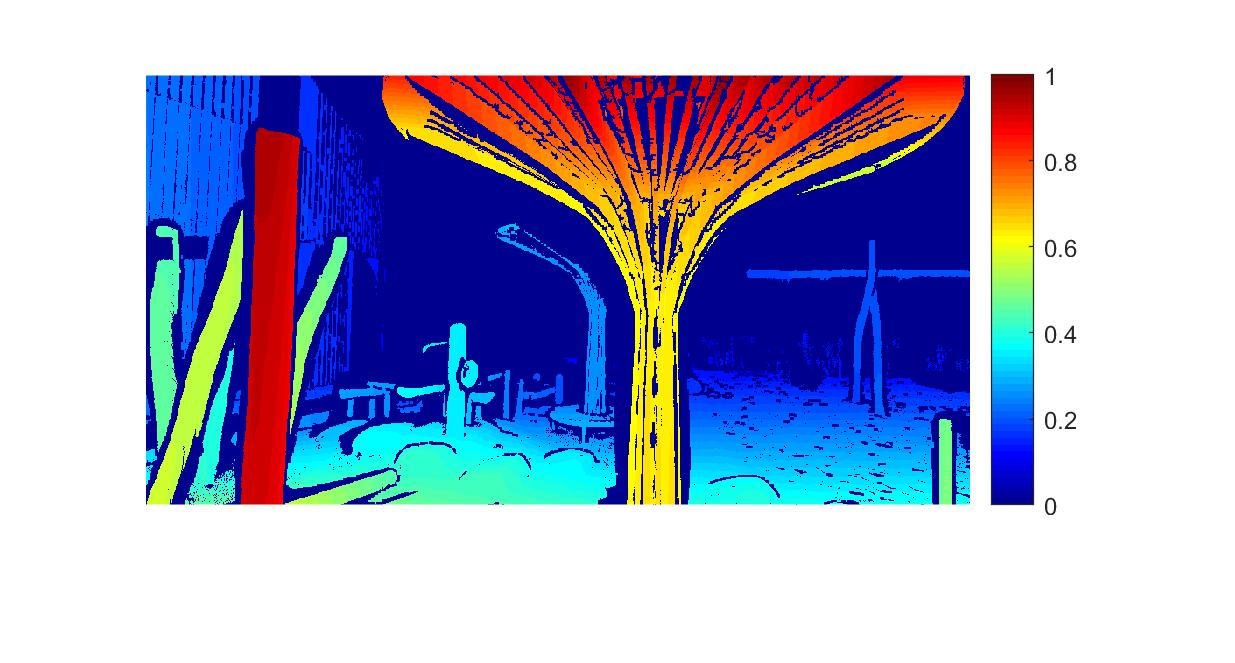
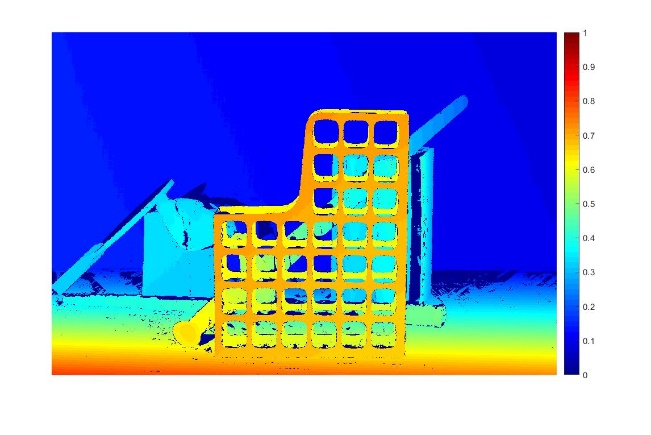
RAM:16GB

Resulting disparity maps:



Ground truth:





|  |  |
| --- | --- |
| 1 | 2 |
| 3 | 4 |

1. Motorcycle 2. Terrace

3. Sword 4. Playground

## Test set: Motorcycle:

Performance:

|  |  |
| --- | --- |
| Test set: | Motorcycle |
| PSNR of masked disparity map(dB) | 25.12 |
| PSNR of unmasked disparity map(dB) | 20.05 |

Time consumption analysis:

|  |  |
| --- | --- |
| Census Transform | 14s |
| Window Matching | 1154s |
| Refinement | 63s |
| Total: | 1231s |

Parameter setting:

|  |  |
| --- | --- |
| Algorithm: | Window matching with fixed window size and joint cost function. |
| Maximum disparity range | 280 |
| Window radius | 12 pixels |
| Refinement: | |
| Prior majority vote | 2 iterations with window radius of 6 |
| Hole\_filling:3\*r4 | 3 iterations with window radius of 4 |
| Posterior majority vote | 2 iterations with window radius of 6 |
| Background filling | Yes |

The mask file of motorcycle test set is acquired from <http://vision.middlebury.edu/stereo/data/2014/>. The mask file indicates the invalid points in the ground truth.

After removing the invalid points in the ground truth, the following disparity map is acquired.

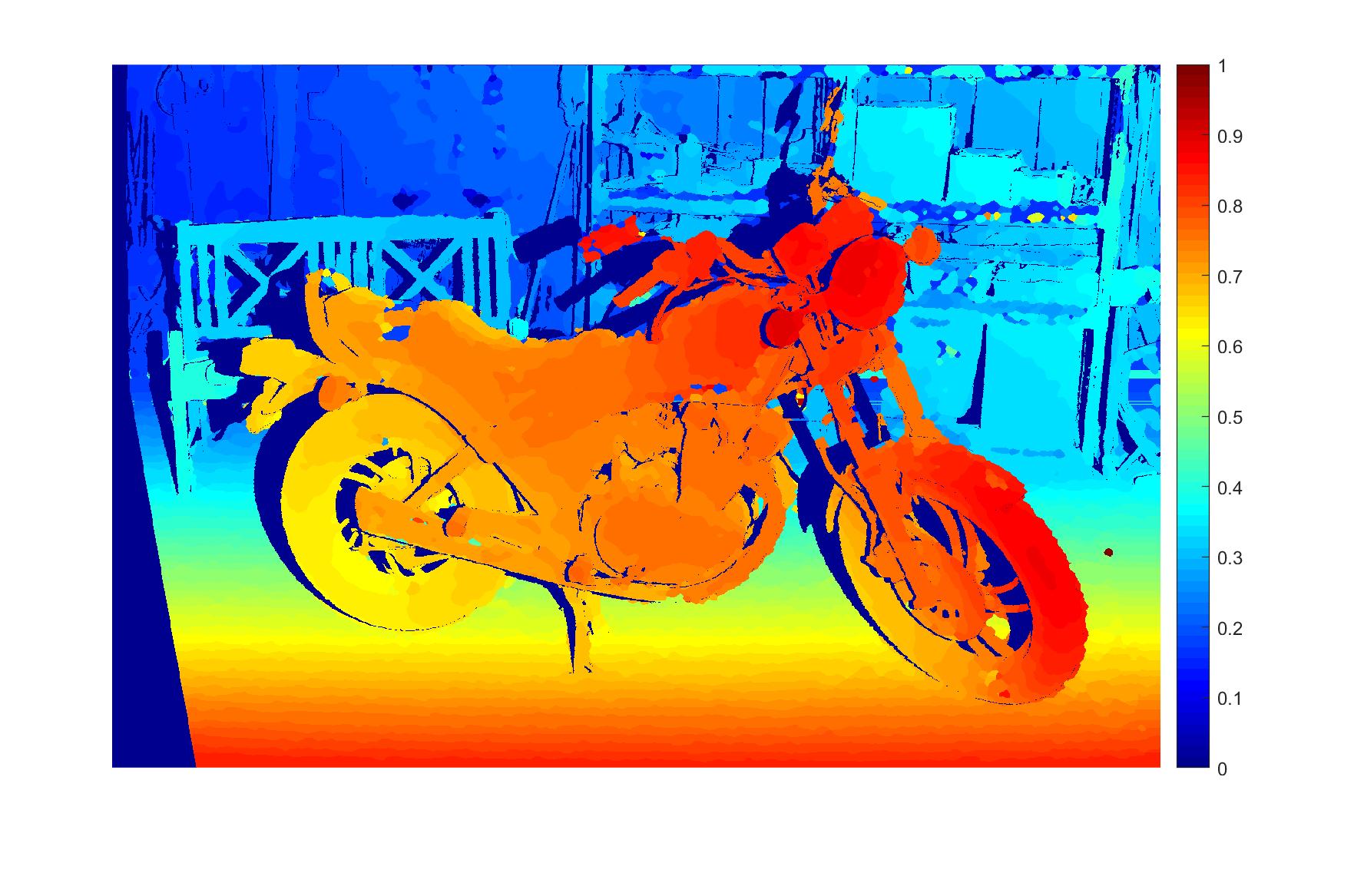


Figure.1: Masked disparity map

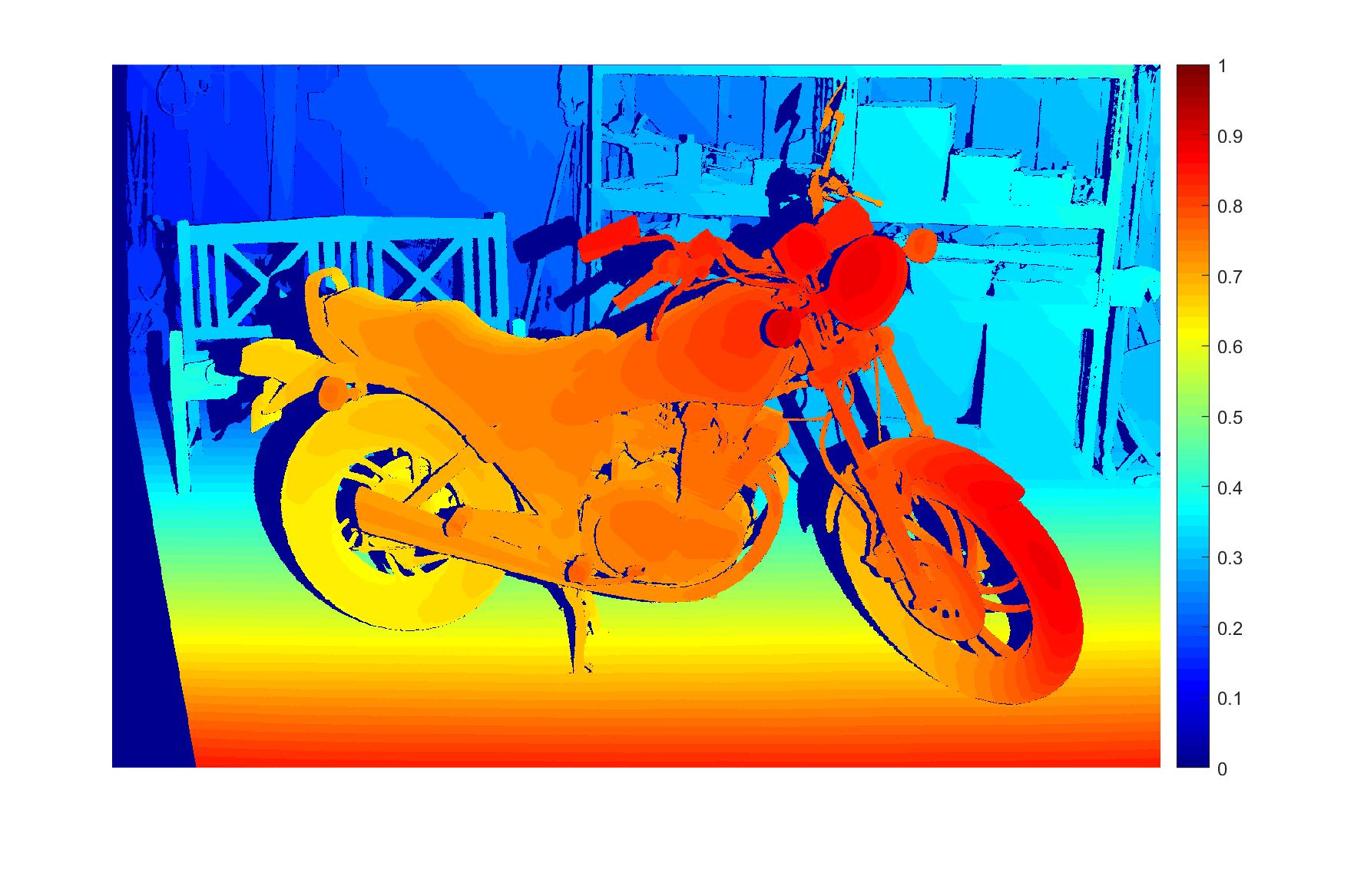


Figure.2: Masked ground truth

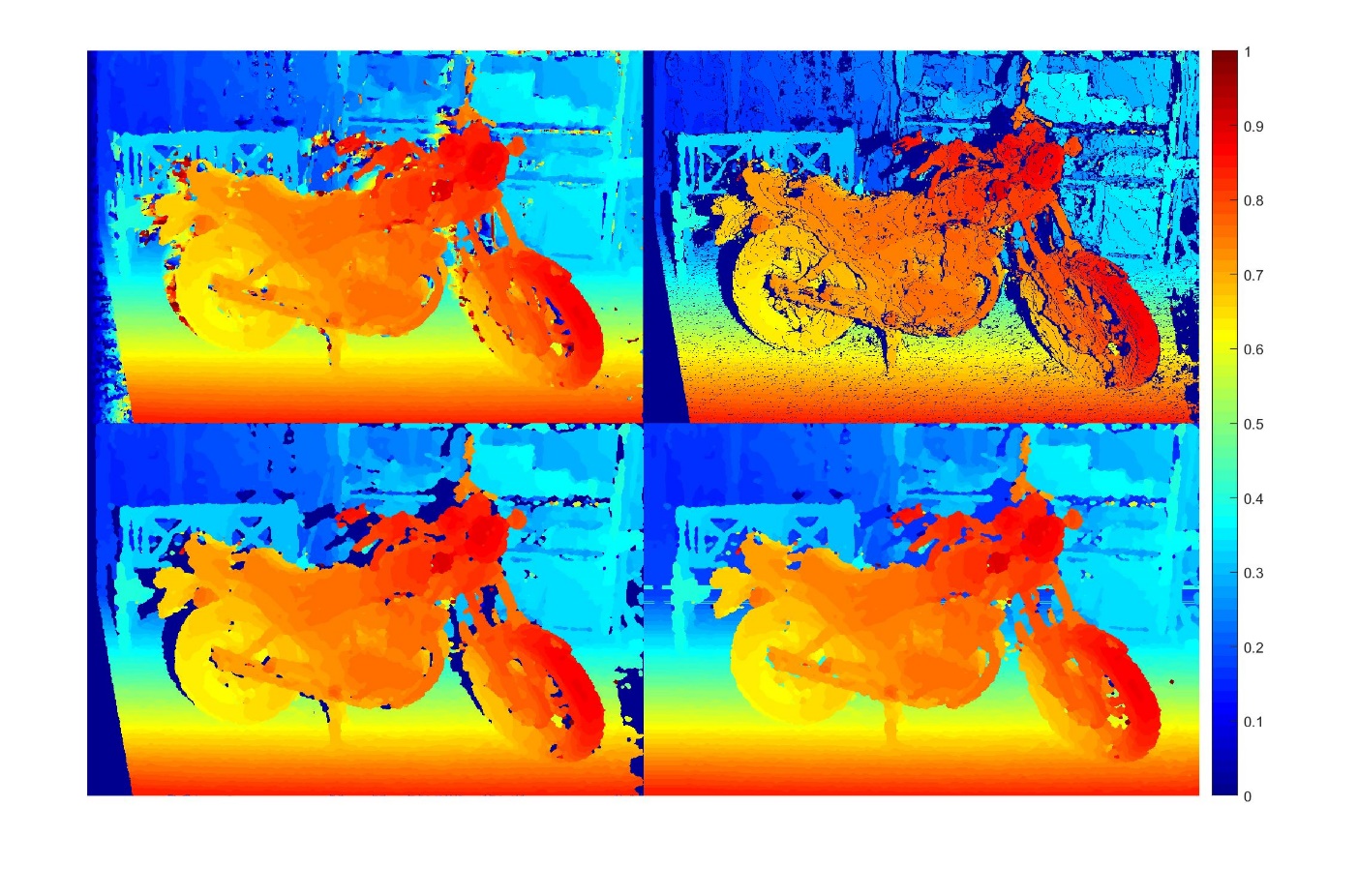


Figure.3

|  |  |
| --- | --- |
| 1 | 2 |
| 3 | 4 |

1. Original disparity map before refinement 2. Sparse disparity map after consistency check

3. Sparse disparity map after majority vote and hole filling 4. Dense disparity map after background filling

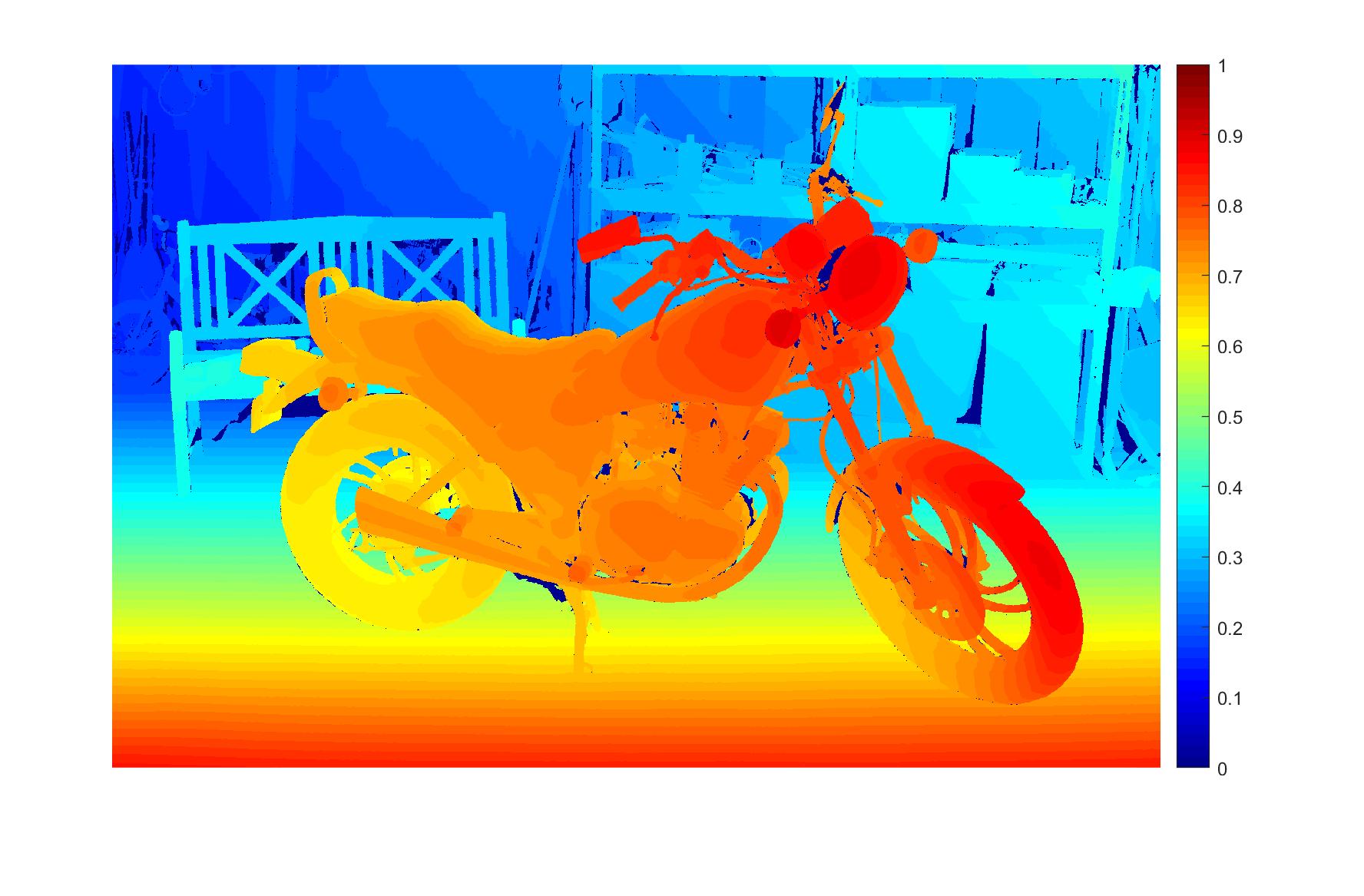


Figure.4 Ground truth

## Test set: Terrace

Performance:

|  |  |
| --- | --- |
| Test set: | Terrace |
| PSNR of masked disparity map(dB) | 28.18 |
| PSNR of unmasked disparity map(dB) | 7.28 |

Time consumption analysis:

|  |  |
| --- | --- |
| Census Transform | 0.64s |
| Window Matching | 25.13s |
| Refinement | 3.48s |
| Total: | 29.25s |

Parameter setting:

|  |  |
| --- | --- |
| Algorithm: | Window matching with adaptive window size and cost function based on Hamming distance. |
| Maximum disparity range | 30 |
| Adaptive window radius range: | 2~23 pixels |
| Penalty coefficient | Beta =25, Gamma=2 |
| Refinement: | |
| Prior majority vote | 2 iterations with window radius of 3 |
| Hole\_filling:3\*r4 | 3 iterations with window radius of 4 |
| Posterior majority vote | 2 iterations with window radius of 3 |
| Background filling | Yes |

Only a masked ground truth is provided, after removing the invalid points in the ground truth, the following disparity map is acquired.

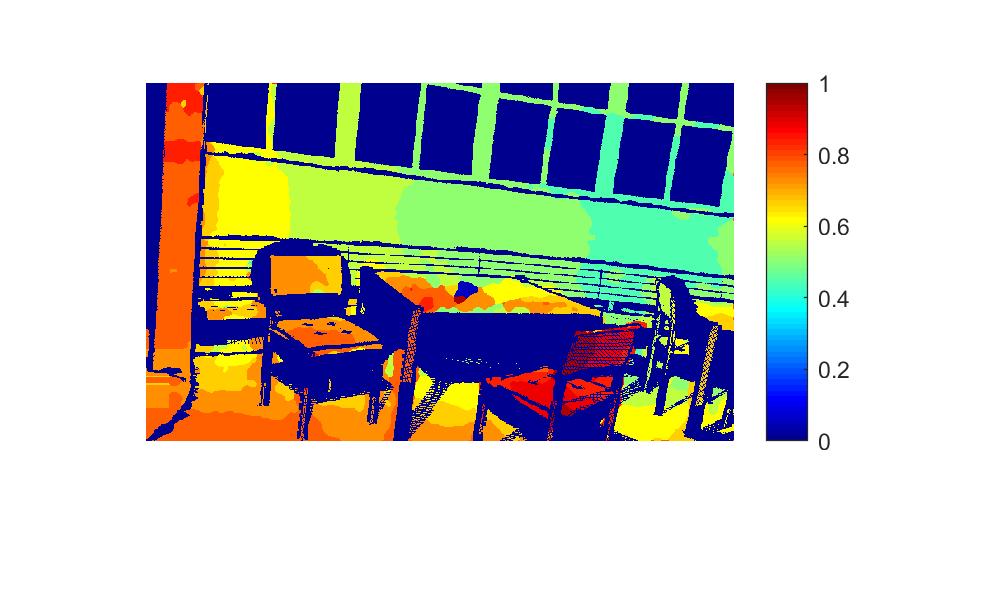


Figure.1: Masked disparity map



Figure.2 Ground truth

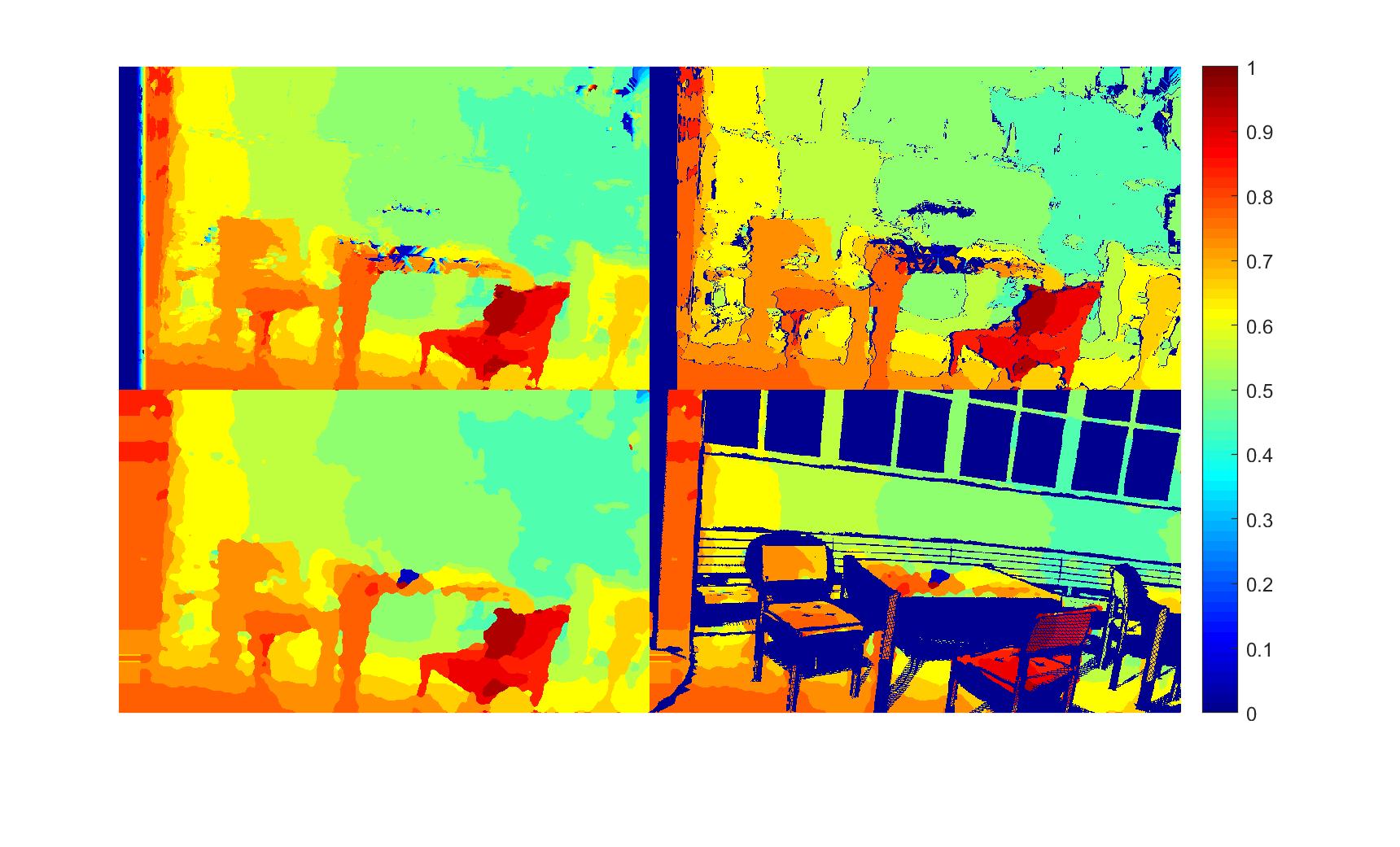


Figure.3

|  |  |
| --- | --- |
| 1 | 2 |
| 3 | 4 |

1. Original disparity map before refinement 2. Sparse disparity map after consistency check

3. Disparity map after majority vote and hole filling 4. Sparse disparity map after background filling

## Test set: Playground

Performance:

|  |  |
| --- | --- |
| Test set: | Terrace |
| PSNR of masked disparity map(dB) | 25.92 |
| PSNR of unmasked disparity map(dB) | 11.01 |

Time consumption analysis:

|  |  |
| --- | --- |
| Census Transform | 1.01s |
| Window Matching | 36.63s |
| Refinement | 5.29s |
| Total: | 42.93s |

Parameter setting:

|  |  |
| --- | --- |
| Algorithm: | Window matching with adaptive window size and cost function based on Hamming distance. |
| Maximum disparity range | 30 |
| Adaptive window radius range: | 2~35 pixels |
| Penalty coefficient | Beta =30, Gamma=2 |
| Refinement: | |
| Prior majority vote | 2 iterations with window radius of 3 |
| Hole\_filling:3\*r4 | 3 iterations with window radius of 4 |
| Posterior majority vote | 2 iterations with window radius of 3 |
| Background filling | Yes |

Only a masked ground truth is provided, after removing the invalid points in the ground truth, the following disparity map is acquired.

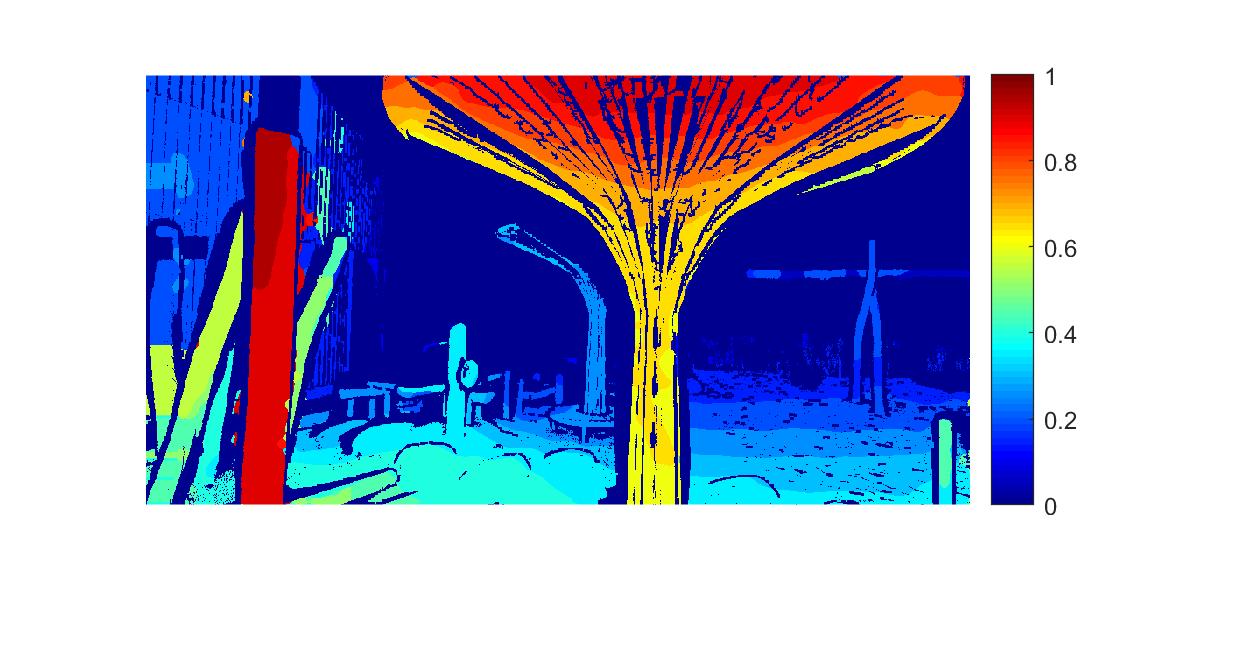


Figure.1: Masked disparity map

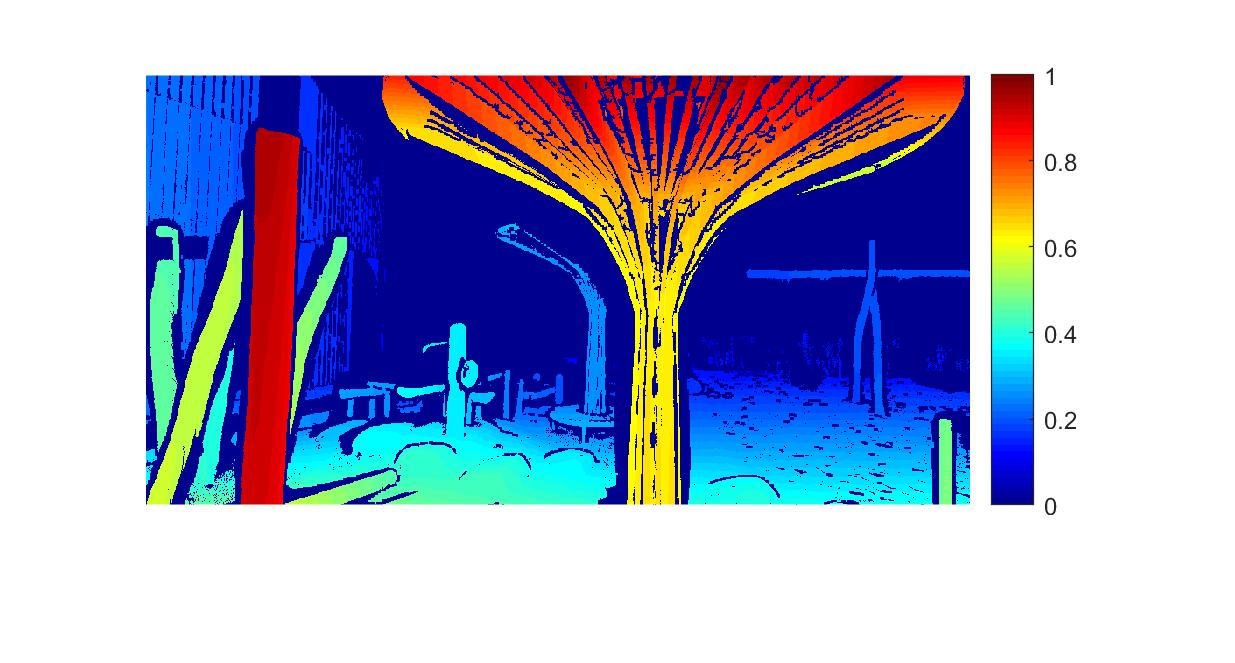


Figure.2 Ground truth

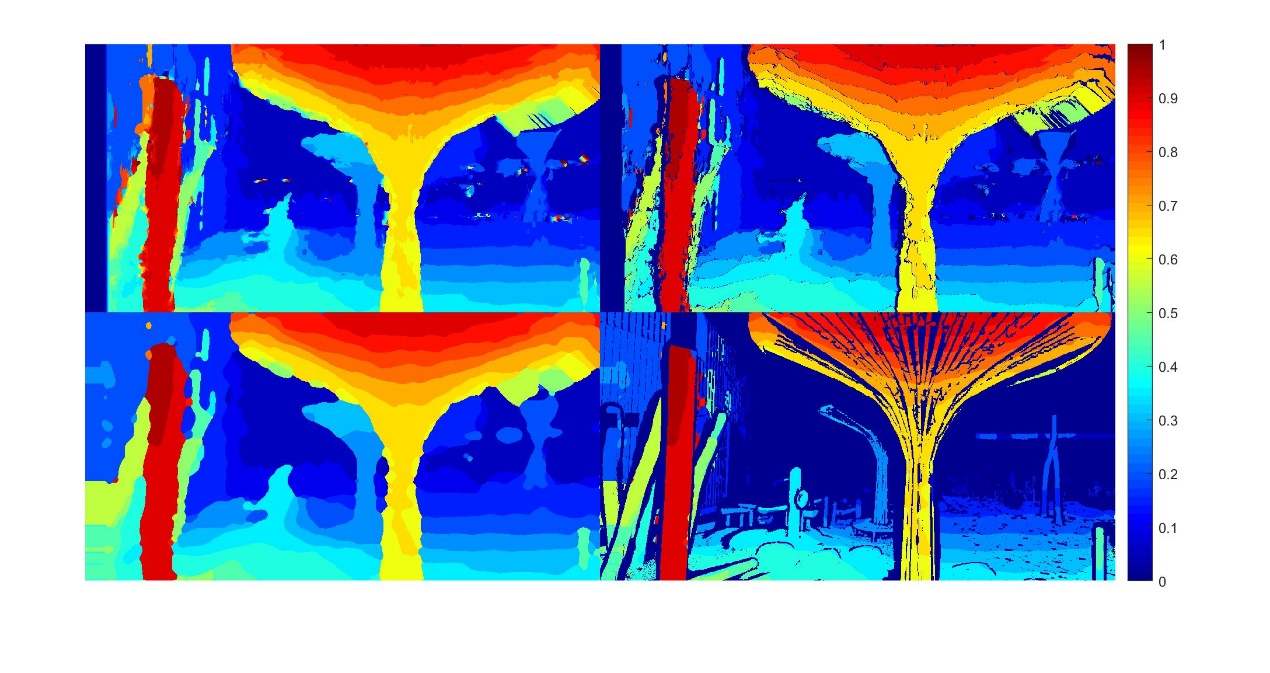


Figure.3

|  |  |
| --- | --- |
| 1 | 2 |
| 3 | 4 |

1. Original disparity map before refinement 2. Sparse disparity map after consistency check

3. Disparity map after majority vote, hole filling and background filling 4. Sparse disparity map after masking

## Test set: Sword

Performance:

|  |  |
| --- | --- |
| Test set: | Sword |
| PSNR of masked disparity map(dB) | 13.70 |
| PSNR of unmasked disparity map(dB) | 12.30 |

Time consumption analysis:

|  |  |
| --- | --- |
| Census Transform | 12.8s |
| Window Matching | 1420s |
| Refinement | 163s |
| Total: | 1595.8s/26.6min |

Parameter setting:

|  |  |
| --- | --- |
| Algorithm: | Window matching with fixed window size and joint cost function. |
| Maximum disparity range | 370 |
| Window radius | 15 |
| Refinement: | |
| Prior majority vote | 2 iterations with window radius of 3 |
| Hole\_filling:3\*r4 | 3 iterations with window radius of 4 |
| Posterior majority vote | 2 iterations with window radius of 3 |
| Background filling | Yes |

Only a masked ground truth is provided, after removing the invalid points in the ground truth, the following disparity map is acquired.

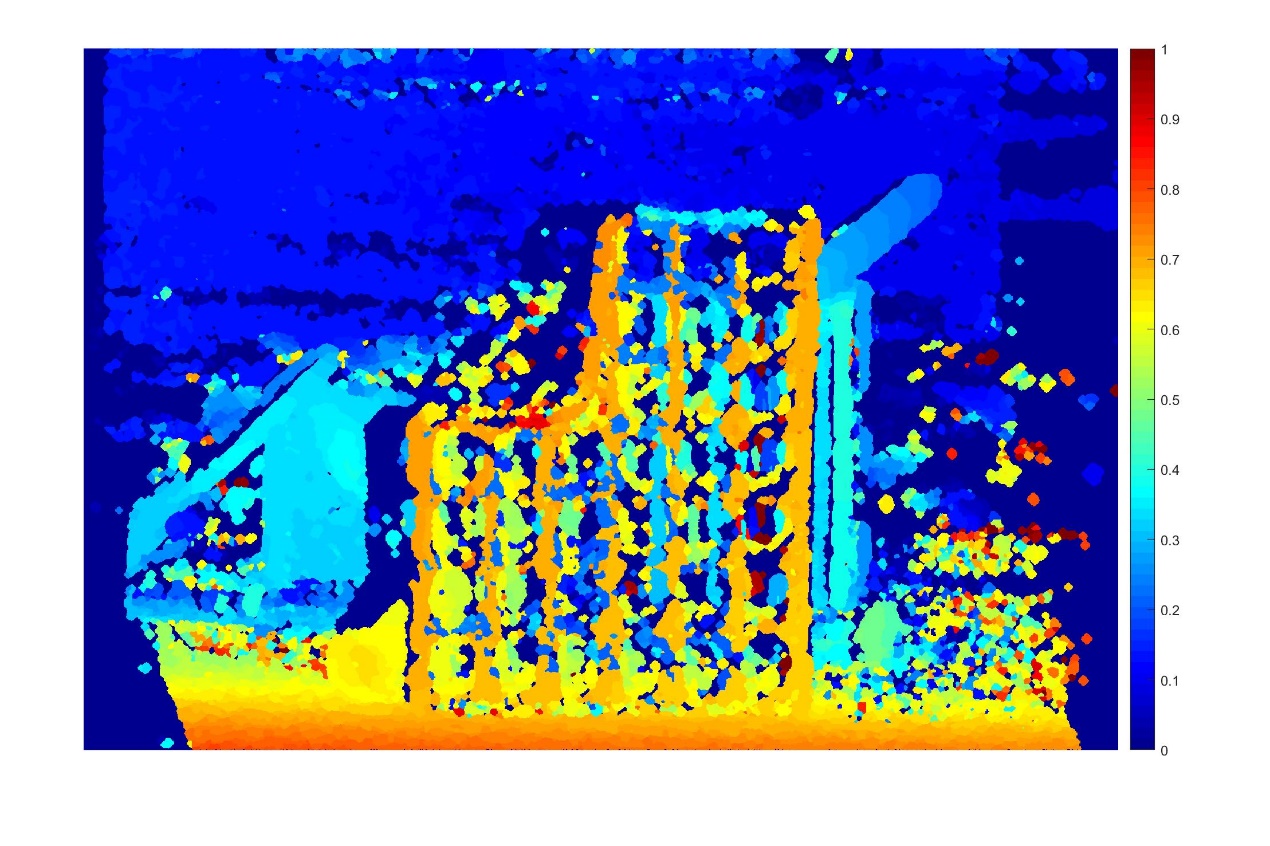


Figure.1: Masked disparity map

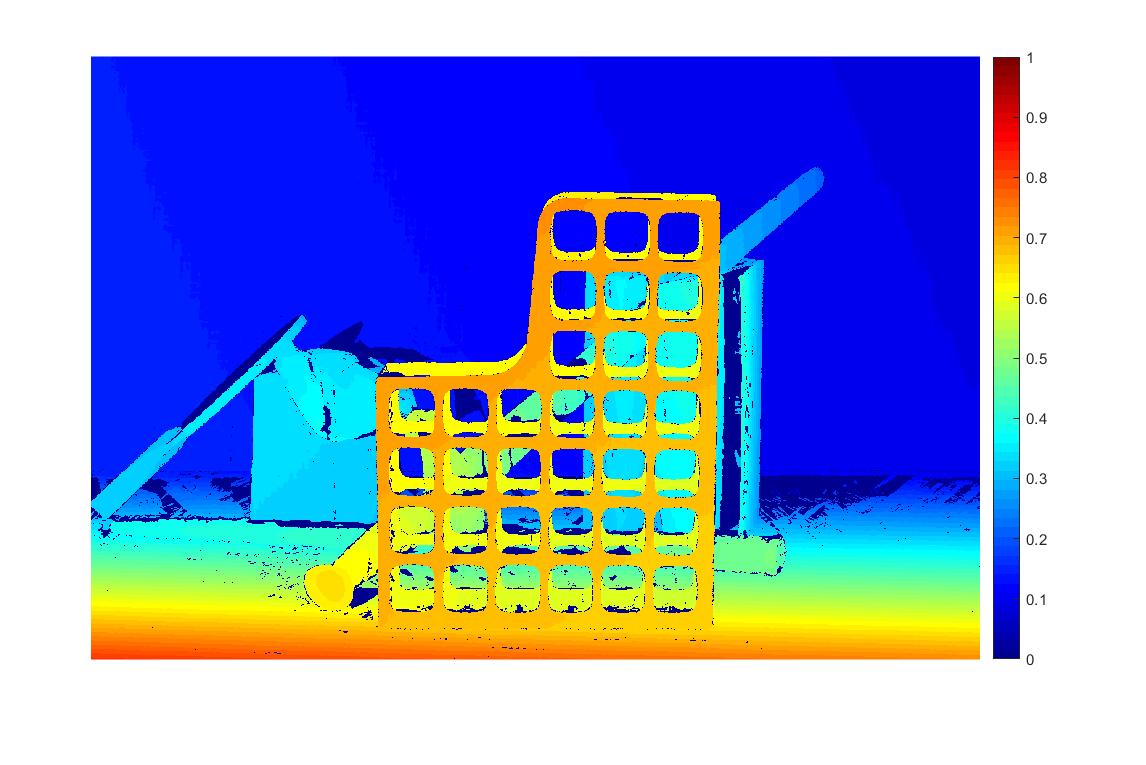


Figure.2 Ground truth

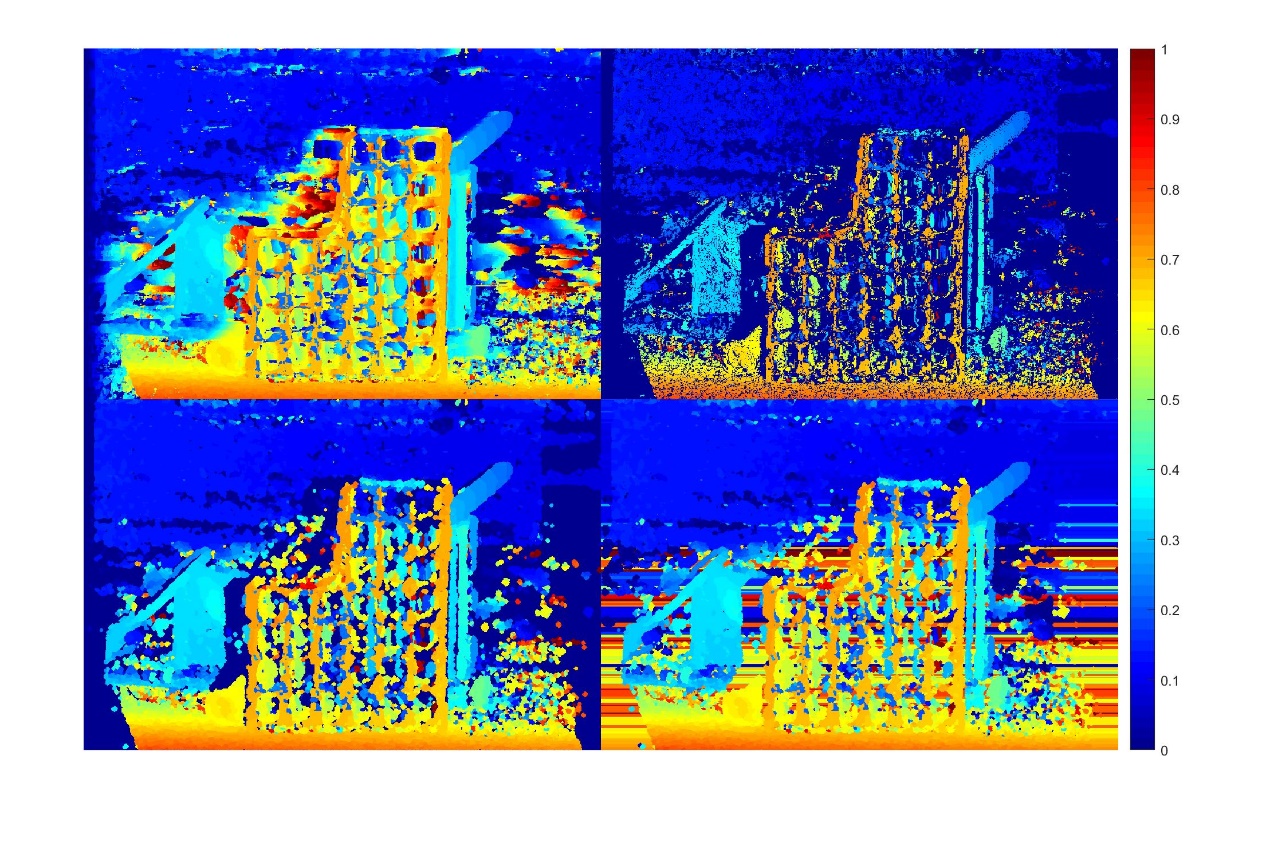


Figure.3

|  |  |
| --- | --- |
| 1 | 2 |
| 3 | 4 |

1. Original disparity map before refinement 2. Sparse disparity map after consistency check

3. Disparity map after majority vote and hole filling 4. Disparity map after bakgroundfilling