# Siamese Network & Stereo

Wenjie Luo
CSC2523

Feb 2nd, 2016

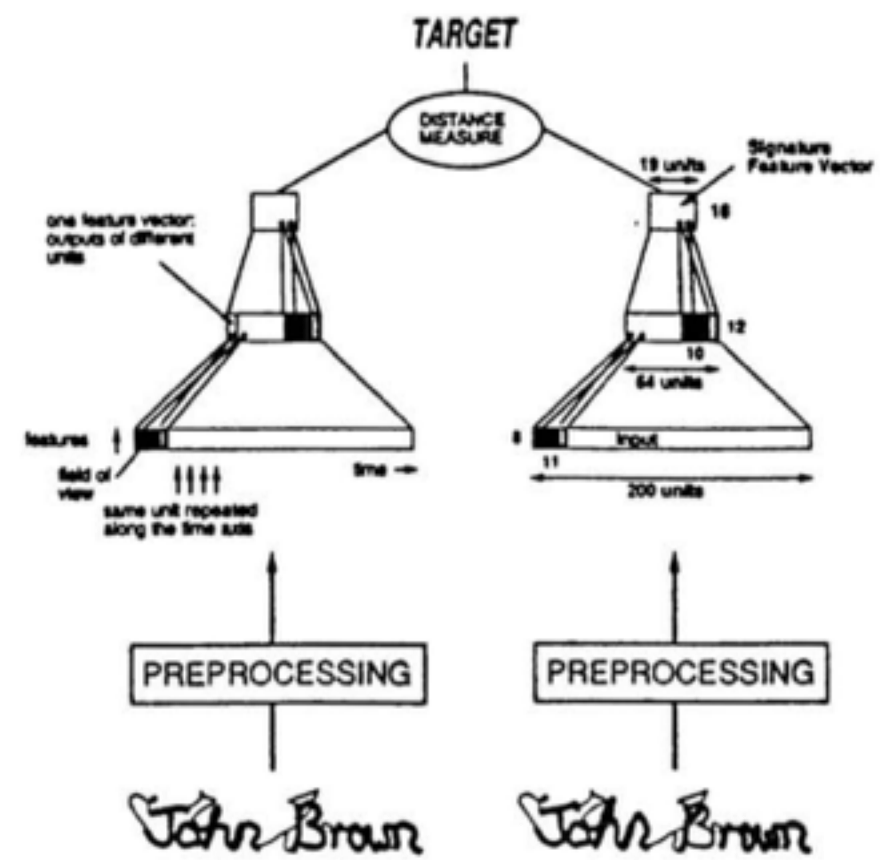
# Outline

- Siamese network
- Application: stereo
- Discussion

- Recap on CNN:
    - Input: one image
    - Output: class label, bounding box etc..
- What if?
    - Input: two images, equivalent
    - Parameter sharing?

# Siamese network

- Consists of two identical sub-networks: feature extraction
- Joined at their outputs: measure distance between feature vectors
- Date back to NIPS 1994




Source: J. Bromley et. al.

# Applications

- Face verification/recognition
- Video sequence
- *Stereo* (depth estimation)

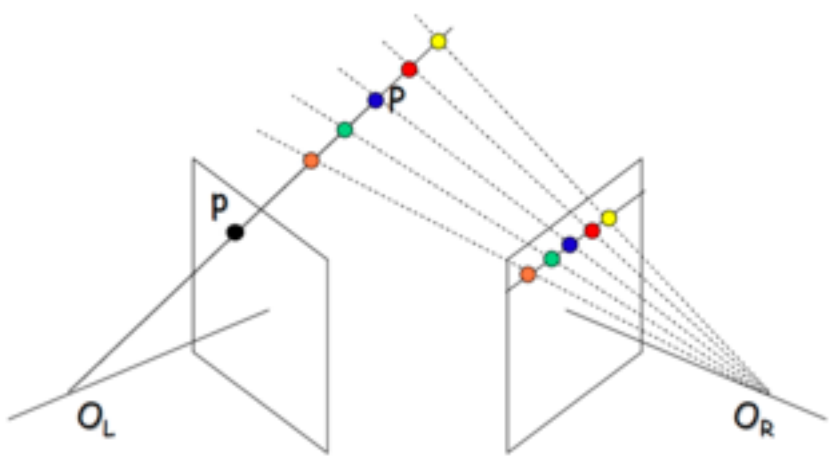
# Why depth

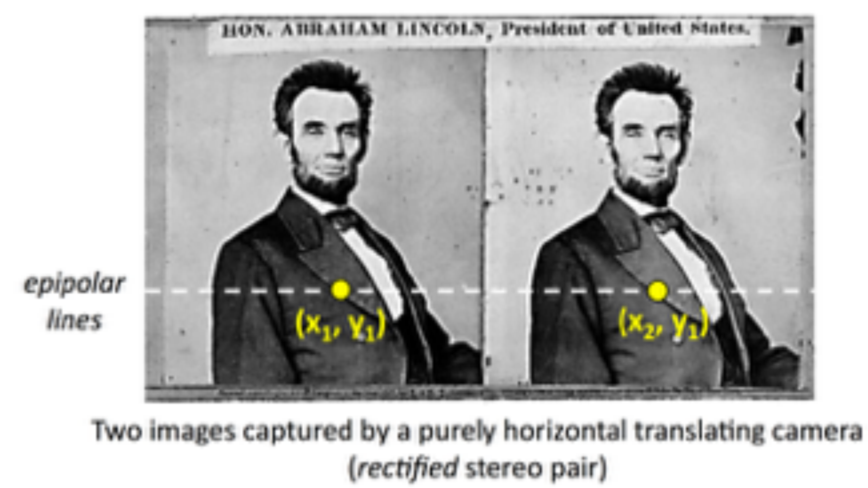
- Structure and depth are inherently ambiguous from a single view

Source: L. Lazebnik

# Stereo

- Estimate depth from stereo images.




Source: R. Urtasun

- Depth is inversely proportional to disparity.

$$Z = f\frac{B}{d}$$

Z: depth; f: focal length; B: baseline; d: disparity

# We need..

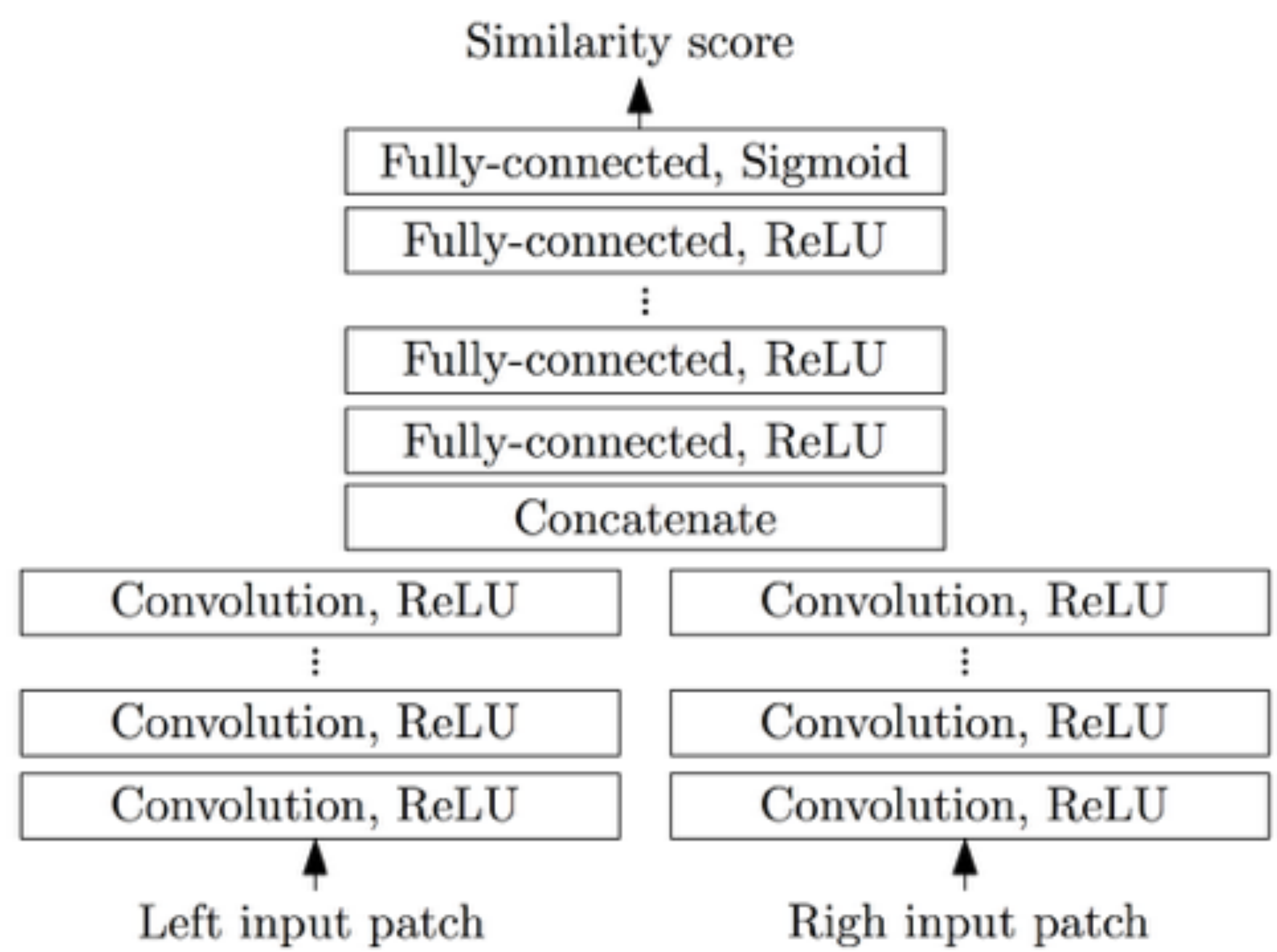
- Correspondances on image locations(Matching)
  - *Good feature*
- Refinement in practice
  - Smoothing

# Conv-Nets

- Input: two image patches
  - Equivalent
- Output: matching cost
- What architecture would you use?

# Network I

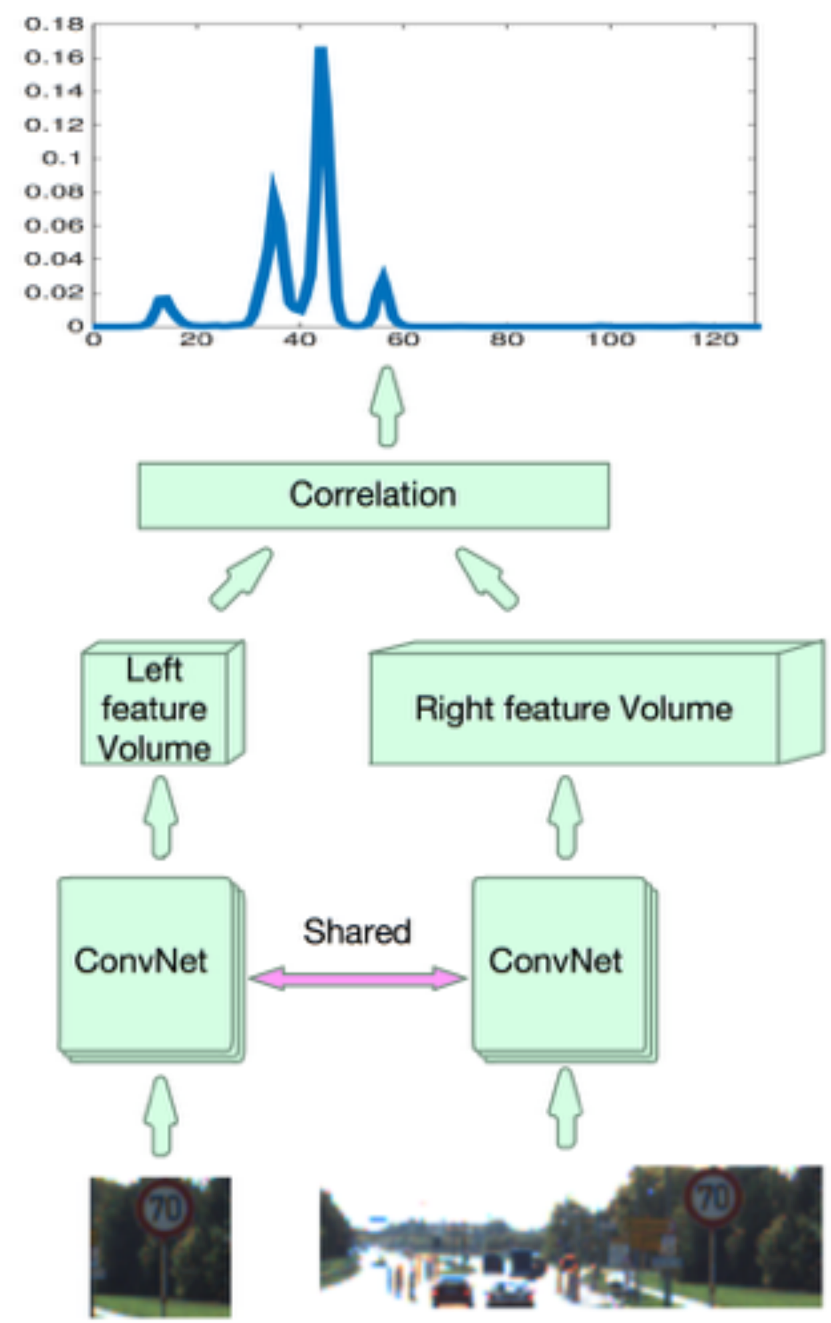
- Two stages:
  - Siamese network
  - Fully connected
- Input: small patch
- Binary prediction
- “Big” network(~600K)




Source: Zbontar & LeCun

# Network II

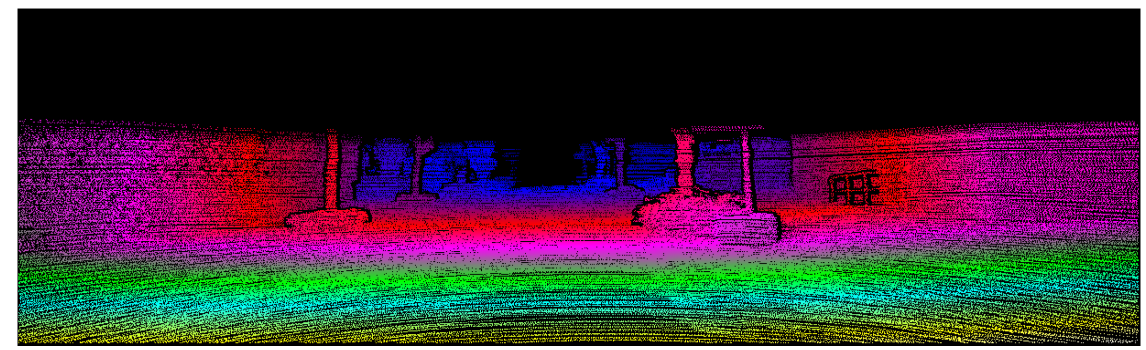
- Dot-product
- Input: full content
- Larger patch
- Log loss
- Smaller network



# KITTI 2012

Left input image

Right input image



- Gray image, outdoor/noisy, 194/195 split
- Disparity range: 256
- Saturation/Textureless(dynamic range)
- Evaluation metric

# Training

- Preprocessing
  - full image or small patch
  - data-augmentation, loading
- Siamese network
  - Gradient aggregated
- Initialization, SGD
- Batch Normalization(variance shift, works well)

# Test

- Image size: W, H; Disparity range: D
  - W * H * D: $1200 \times 370 \times 256 = 1.14 \times 10^8!$
- Computation
  - Feature shared
- Memory
  - One disparity at a time

# Smoothing

- Cost-aggregation
  - Averaging neighboring locations
- CRF
  - Semiglobal matching
- Post-processing
  - Border fixing(CNN), left-right consistency, outlier detector

# Stereo Evaluation 2012

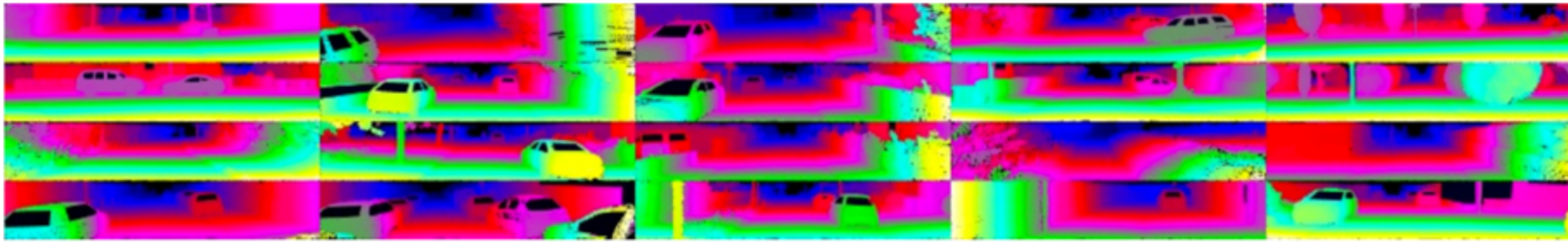


Table All Error threshold 3 pixels Evaluation area All pixels

| | Method | Setting | Code | Out-Noc | Out-All | Avg-Noc | Avg-All | Density | Runtime | Environment | Compare |
|---|---|---|---|---|---|---|---|---|---|---|---|
| 1 | Displets v2 | | code | 2.37 % | 3.09 % | 0.7 px | 0.8 px | 100.00 % | 265 s | >8 cores @ 3.0 Ghz (Matlab + C/C++) | |
| F. Guney and A. Geiger: Displets: Resolving Stereo Ambiguities using Object Knowledge. Conference on Computer Vision and Pattern Recognition (CVPR) 2015. | | | | | | | | | | | |
| 2 | MC-CNN-acrt | | code | 2.43 % | 3.63 % | 0.7 px | 0.9 px | 100.00 % | 67 s | Nvidia GTX Titan X (CUDA, Lua/Torch7) | |
| J. Zbontar and Y. LeCun: Stereo Matching by Training a Convolutional Neural Network to Compare Image Patches. Submitted to JMLR . | | | | | | | | | | | |
| 3 | Displets | | code | 2.47 % | 3.27 % | 0.7 px | 0.9 px | 100.00 % | 265 s | >8 cores @ 3.0 Ghz (Matlab + C/C++) | |
| F. Guney and A. Geiger: Displets: Resolving Stereo Ambiguities using Object Knowledge. Conference on Computer Vision and Pattern Recognition (CVPR) 2015. | | | | | | | | | | | |
| 4 | MC-CNN | | | 2.61 % | 3.84 % | 0.8 px | 1.0 px | 100.00 % | 100 s | Nvidia GTX Titan (CUDA, Lua/Torch7) | |
| J. Zbontar and Y. LeCun: Computing the Stereo Matching Cost with a Convolutional Neural Network. Conference on Computer Vision and Pattern Recognition (CVPR) 2015. | | | | | | | | | | | |
| 5 | PRSM | | code | 2.78 % | 3.00 % | 0.7 px | 0.7 px | 100.00 % | 300 s | 1 core @ 2.5 Ghz (C/C++) | |
| C. Vogel, K. Schindler and S. Roth: 3D Scene Flow Estimation with a Piecewise Rigid Scene Model. ijcv 2015. | | | | | | | | | | | |
| 6 | SPS-StFl | | | 2.83 % | 3.64 % | 0.8 px | 0.9 px | 100.00 % | 35 s | 1 core @ 3.5 Ghz (C/C++) | |
| K. Yamaguchi, D. McAllester and R. Urtasun: Efficient Joint Segmentation, Occlusion Labeling, Stereo and Flow Estimation. ECCV 2014. | | | | | | | | | | | |
| 7 | VC-SF | | | 3.05 % | 3.31 % | 0.8 px | 0.8 px | 100.00 % | 300 s | 1 core @ 2.5 Ghz (C/C++) | |
| C. Vogel, S. Roth and K. Schindler: View-Consistent 3D Scene Flow Estimation over Multiple Frames. Proceedings of European Conference on Computer Vision. Lecture Notes in, Computer Science 2014. | | | | | | | | | | | |
| 8 | Deep Embed | | | 3.10 % | 4.24 % | 0.9 px | 1.1 px | 100.00 % | 3 s | 1 core @ 2.5 Ghz (C/C++) | |
| Z. Chen, X. Sun, Y. Yu, L. Wang and C. Huang: A Deep Visual Correspondence Embedding Model for Stereo Matching Costs. ICCV 2015. | | | | | | | | | | | |
| 9 | JSOSM | | | 3.15 % | 3.94 % | 0.8 px | 0.9 px | 100.00 % | 105 s | 8 cores @ 2.5 Ghz (C/C++) | |
| Anonymous submission | | | | | | | | | | | |
| 10 | OSF | | code | 3.28 % | 4.07 % | 0.8 px | 0.9 px | 99.98 % | 50 min | 1 core @ 3.0 Ghz (Matlab + C/C++) | |
| M. Menze and A. Geiger: Object Scene Flow for Autonomous Vehicles. Conference on Computer Vision and Pattern Recognition (CVPR) 2015. | | | | | | | | | | | |

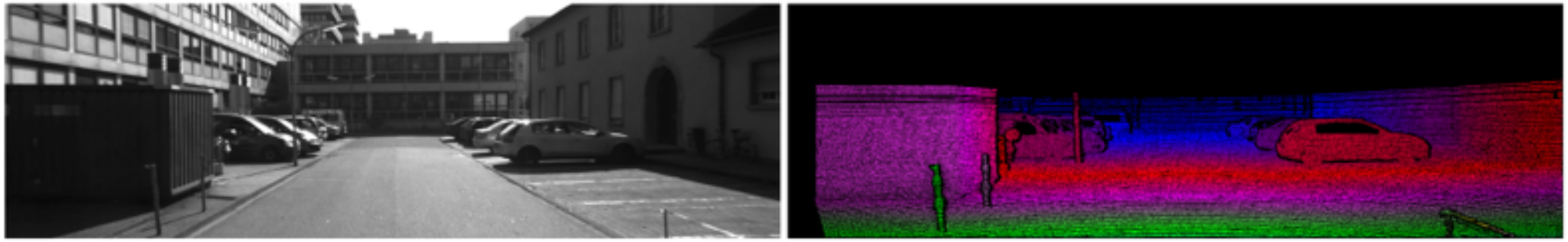
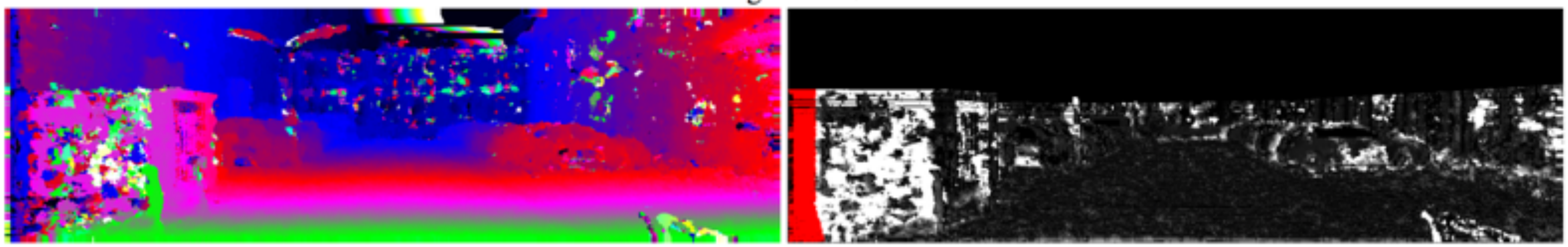
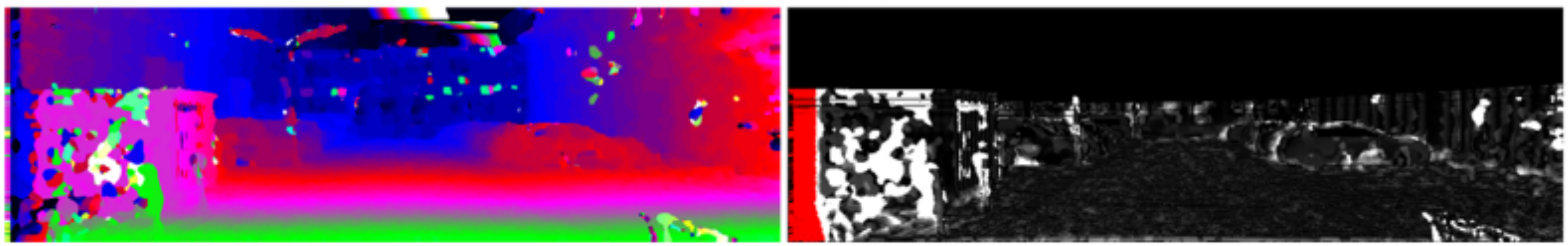


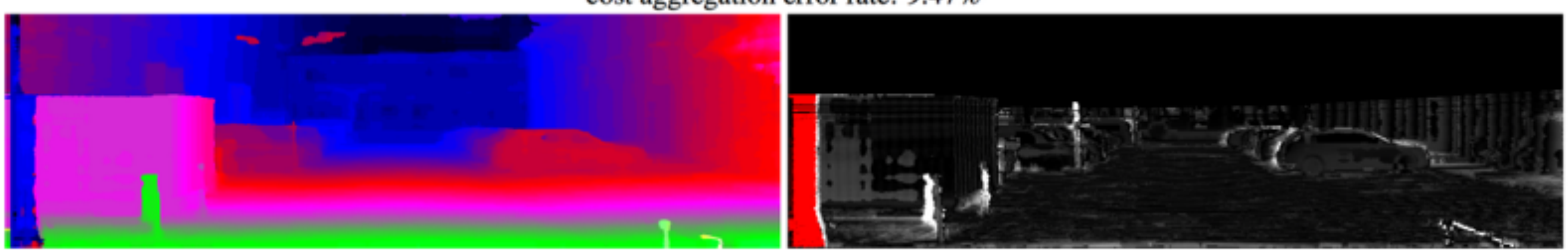
image id: 170



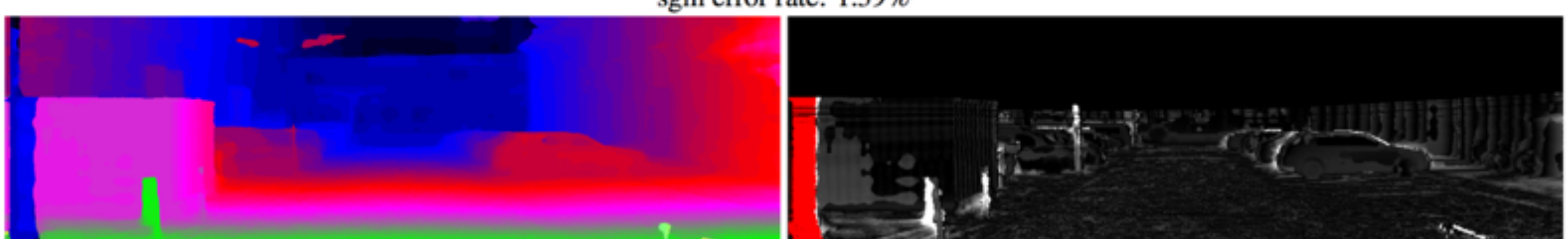
cnn error rate: 13.48%



cost aggregation error rate: 9.47%



sgm error rate: 1.39%



final error rate: 1.15%

Thank You

Q&A