Depth Estimation Using Deep Neural Networks

Mușat Bogdan-Adrian

December 2017

Problem Formulation

- Given a pair of stereo images, how to estimate the depth of the scene?

Disparity

- Disparity represents the difference of perspective created by the horizontal or vertical separation of two cameras
- The human brain processes disparity information from both eyes to estimate the depth of real world objects
- Disparity is inverse proportional with depth; given the baseline distance b between the cameras and the camera focal lenght f, the depth $\hat{d}$ from the predicted disparity d is simply $\hat{d} = bf/d$
- Objects closer to the camera have bigger disparities, whilst objects that are farther have smaller disparities

Proposed Solution

- Given a pair of rectified [1] stereo images as input, build a model which can accurately predict the disparity per-pixel (i.e. learn the Δ values with which every pixel from the left image is shifted from the right one on the x axis)
- Such a system can be modeled by a Convolutional Neural Network
- The dataset used for experiments is FlyingThings3D[2]

[1] Vertically aligned

[2] N.Mayer et al. "A Large Dataset to Train Convolutional Networks for Disparity, Optical Flow, and Scene Flow Estimation". In: *IEEE International Conference on Computer Vision and Pattern Recognition (CVPR)*. arXiv:1512.02134. 2016. URL: http://lmb.informatik.uni-freiburg.de/Publications/2016/MIFDB16.

Challenges - Textureless Areas

- Depth estimation is an ill-posed problem: a pixel in a textureless area from the left image can belong to multiple pixels in the right image

Figure 1: Example of objects with textureless areas

Challenges - Object Occlusion

- Parts of an object may be visible in an image, while being absent in the other one because of the difference of perspective

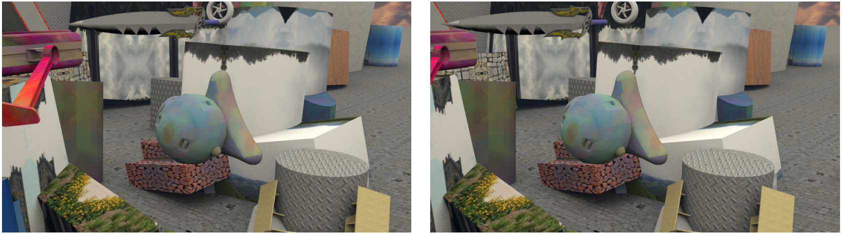

Figure 2: Occlusion in stereo images

Convolutional Neural Networks Architectures

- Supervised
  - Requires ground-truth disparity, which might be expensive to obtain
  - Based on disparity regression
  - Accurate predictions
- Unsupervised
  - Does not require any form of ground-truth
  - Based on an image reconstruction loss
  - Has artefacts around object boundaries, caused by occlusion

DispNet[3] - Overview

- Image-to-image with contractive and expanding structure

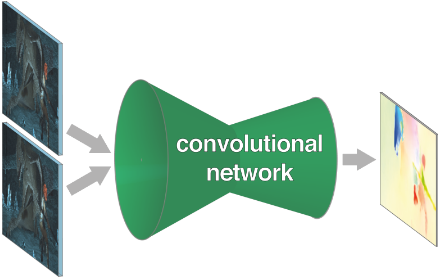

Figure 3: Hourglass structure of DispNet

[3] Nikolaus Mayer et al. "A Large Dataset to Train Convolutional Networks for Disparity, Optical Flow, and Scene Flow Estimation". In: *CoRR* abs/1512.02134 (2015). arXiv: 1512.02134. URL: http://arxiv.org/abs/1512.02134.

DispNet - Architectural Details

- Fully Convolutional Network[4]
- Concatenates the "upconvolution" results with the features from the "contractive" part to recover spatial information lost by downsampling
- Uses smooth L1 loss between the resulted disparity computed at different pyramid levels of the network and the ground-truth

$$|x|_{smooth} = \begin{cases} 0.5x^2, & if\, |x| \leq 1 \\ |x| - 0.5, & otherwise \end{cases}$$

[4] Jonathan Long, Evan Shelhamer, and Trevor Darrell. "Fully Convolutional Networks for Semantic Segmentation". In: *CoRR* abs/1411.4038 (2014). arXiv: 1411.4038. URL: http://arxiv.org/abs/1411.4038.

DispNet - Architectural Details

- The upsampling part of the network can be constructed by using transposed convolutions[5] or by using interpolation methods like nearest neighbor or bilinear followed by a convolutional layer to smoothen the results

[5] Matthew D. Zeiler, Graham W. Taylor, and Rob Fergus. "Adaptive Deconvolutional Networks for Mid and High Level Feature Learning". In: *Proceedings of the 2011 International Conference on Computer Vision*. ICCV '11. Washington, DC, USA: IEEE Computer Society, 2011, pp. 2018–2025. ISBN: 978-1-4577-1101-5. DOI: 10.1109/ICCV.2011.6126474. URL: http://dx.doi.org/10.1109/ICCV.2011.6126474.

DispNet - Results

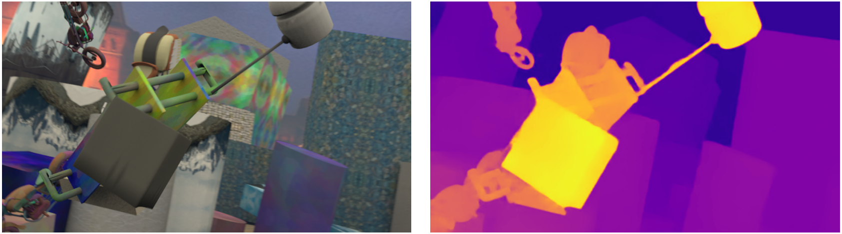

Figure 4: Supervised depth estimation

DispNet - Limitations

- Requires ground-truth data which might be expensive to obtain
- Current depth estimation hardware for collecting ground-truth data like LIDAR, Time of flight, Kinect may provide inaccurate predictions caused by environmental factors, which directly affects the network's performance

Unsupervised Monocular Network[6] - Overview

- Similar architecture as DispNet

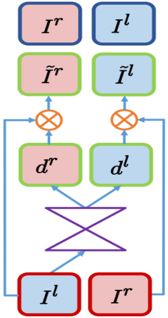

Figure 5: Unsupervised Monocular Network architecture

[6] Clément Godard, Oisin Mac Aodha, and Gabriel J. Brostow. "Unsupervised Monocular Depth Estimation with Left-Right Consistency". In: *CoRR* abs/1609.03677 (2016). arXiv: 1609.03677. URL: http://arxiv.org/abs/1609.03677.

Unsupervised Monocular Network - Architectural Details

- Shares the same hourglass structure as DispNet and works the same, up until a point
- Instead of using ground-truth disparity, the network tries to reconstruct the right image from the left one, and vice-versa
- The network tries to achieve the needed disparities as to warp the left and right image using some sampling method
- The sampling method chosen is binilinear sampling as used in Spatial Transformer Networks[7], which is fully differentiable
- During inference, only the left view image is used (monocular estimation)

[7] Max Jaderberg et al. "Spatial Transformer Networks". In: *CoRR* abs/1506.02025 (2015). arXiv: 1506.02025. URL: http://arxiv.org/abs/1506.02025.

Unsupervised Monocular Network - Architectural Details

- Uses a three term loss

$$C_s = \alpha_{ap}(C_{ap}^l + C_{ap}^r) + \alpha_{ds}(C_{ds}^l + C_{ds}^r) + \alpha_{lr}(C_{lr}^l + C_{lr}^r)$$

- Appearance Matching Loss

$$C_{ap}^l = \frac{1}{N}\sum_{i,j} \alpha \frac{1 - SSIM(I_{ij}^l, \tilde{I}_{ij}^l)}{2} + (1-\alpha)\|I_{ij}^l - \tilde{I}_{ij}^l\|$$

- Disparity Smoothness Loss

$$C_{ds}^l = \frac{1}{N}\sum_{i,j} |\partial_x d_{ij}^l| e^{-|\partial_x I_{ij}^l|} + |\partial_y d_{ij}^l| e^{-|\partial_y I_{ij}^l|}$$

- Left-Right Disparity Consistency Loss

$$C_{lr}^l = \frac{1}{N}\sum_{i,j} |d_{ij}^l - d_{ij+d_{ij}^l}^r|$$

Unsupervised Monocular Network - Results

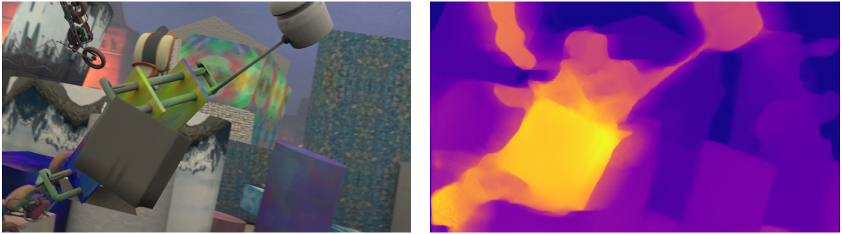

Figure 6: Unsupervised depth estimation

Unsupervised Monocular Network - Limitations

- There are visible artefacts around the object boundaries due to the pixels in the occlusion region not being visible in both images
- Because this method relies solely on image reconstruction, textureless or transparent surfaces will produce inconsistent depths
- Complex objective loss

Benchmark Evaluation

- One of the main benchmarks for depth estimation is KITTI 2015[8] and the evaluation metric used is called D1-all, which is the percentage of pixels for which the estimation error is larger than 3px and larger than 5% of the ground truth disparity at this pixel
- On this benchmark, DispNet achieves a D1-all error of 4.34%, while the unsupervised network gets a 23.81% error

[8] Moritz Menze and Andreas Geiger. "Object Scene Flow for Autonomous Vehicles". In: *Conference on Computer Vision and Pattern Recognition (CVPR)*. 2015.

Conclusions

- Unsupervised methods work more alike the brain in the sense that depth is learnt by image reconstruction enforcement
- There should be a preferance towards unsupervised models due to their lack of needing precise estimations
- For the moment, supervised networks perform much better than their unsupervised counterparts on popular benchmarks

Technical Slides - Transposed Convolutions

- The input (bottom) is specially padded and then a traditional convolutional layer is applied

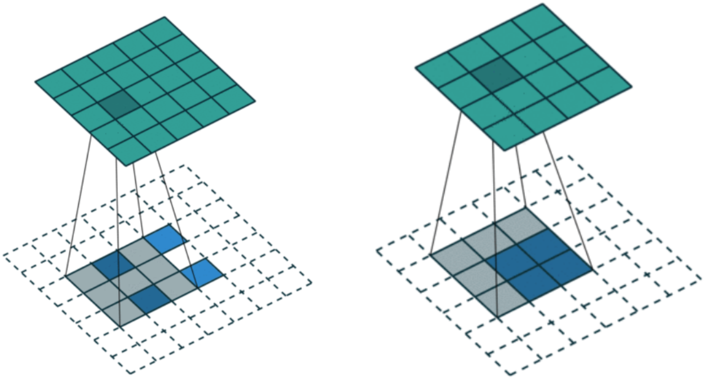

Figure 7: Transposed Convolutions

Technical Slides - Structural Similarity (SSIM)[9]

$$SSIM(x, y) = \frac{(2\mu_x\mu_y + c_1)(2\sigma_{xy} + c_2)}{(\mu_x^2 + \mu_y^2 + c_1)(\sigma_x^2 + \sigma_y^2 + c_2)}$$

- μ_x the average of x, μ_y the average of y;
- σ_x^2 the variance of x, σ_y^2 the variance of y;
- σ_{xy} the covariance of x and y;
- $c1 = (k_1L)^2, c2 = (k_2L)^2$ two variables to stabilize the division with weak denominator;
- L the dynamic range of the pixel-values (usually $2^{\#\text{bits-per-pixel}} - 1$);
- $k_1 = 0.01$ and $k_2 = 0.03$ by default.

[9] Zhou Wang et al. "Image Quality Assessment: From Error Visibility to Structural Similarity". In: *Trans. Img. Proc.* (2004).

Technical Slides - Smooth L1 Loss

$$|x|_{smooth} = \begin{cases} 0.5x^2, & if\,|x| \leq 1 \\ |x| - 0.5, & otherwise \end{cases}$$

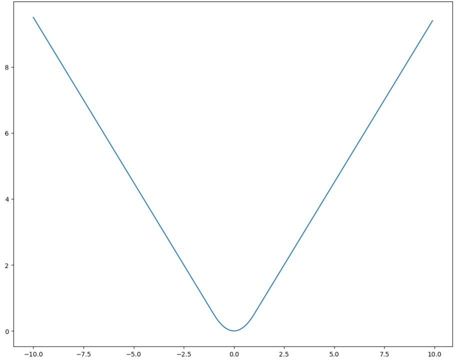

Figure 8: Plot for smooth L1