Statistical Problems in Monocular Depth Estimation

Hao Xin

April 2019

1 Introduction

Monocular depth estimation is an important problem in scene understanding, depth information from single RGB image can be useful for 3D reconstruction. One fundamental question of monocular depth learning is that whether there is indeed rich enough depth information contained in a single RGB image. If the answer is yes, then it would be possible for a well proposed model to learn depth in an unsupervised manner. Otherwise, supervised learning scheme should be preferred.

In the context of supervised learning, similar to semantic segmentation, the prediction task is pixel-wised. Thus, the problem can be seen as a pixel-wised regression problem. A lot of CNN based models have been proposed so far. Some observations from computer vision researchers shows some guideline on designing deep convolutional neural networks for depth learning are:

1. Global spatial information (usually captured by large receptive field) is beneficiary.
2. Combining imagenet pre-trained models (such as VGG and ResNet) helps improve performance.
3. Skip connections with middle level information could help with dense prediction.

2 Statistics perspectives

2.1 Modeling Part

1. No modeling.
2. FRAME (Markov random field) model.
3. CRF (Conditional random filed) model.

A big problem for random field model is to define neighborhood relation on image pixels. FRAME model has advantage by using filters to define energy function.

2.2 Convolution Operation

Large receptive field of CNN filters has been shown to be effective on capturing global features in practice. However, large filter size would bring in computation burden especially for deep network structures. Atrous convolution, also known as dilated convolution solves the computational difficulty for large filters by employ a sparse convolution with input feature maps. Is there exist other way of convolution can effectively capture global features for depth learning?

Another issue is the pooling operation combined with convolution layers. Pooling is way of down sampling output feature maps further for computation efficiency. However, this can be inappropriate for depth learning problem, for example, max-pooling could cause ignoring useful local information which could be important for depth prediction. Should we still keep pooling for convolutional neural networks?