

Unsupervised flow estimation (change this)

Joe Withers

../../2019

1 Abstract

2 Introduction

unsupervised learning of optical flow

using FlowNetS, similar approach to *ref* - extended to ?.

results show... - either comparison to other methods, or comparison to with and without occlusion constraints (or any other additions to the implementation).

3 Related literature

3.1 Traditional optical flow estimation (re-name)

In traditional gradient-based optical flow estimation, a common assumption is brightness consistency. The brightness consistency constraint, **eq**, can be used as a data driven error term in an optimization problem. However, this equation is underdetermined and therefore further constraints are required to estimate optical flow. **Horn and Schunck** impose the global smoothness of the flow as a regularization term, giving a global energy function seen in **eq**. **ref** uses a second order smoothness constraint... **include some extensions of horn and schunck**. These global energy functions can be minimized using continuous optimization methods such as gradient descent. Alternatively, these can be minimized using discrete optimization...**variation method**

3.2 Supervised CNN flow estimation

Convolutional neural networks are good at learning to extract and match important features from images... Recent works have seen CNN's used for pixel dense predictions in segmentation **ref**, and depth prediction **ref**...

In the field of optical flow, **FlowNet** introduced using CNN's to predict optical flow estimation. **FlowNet** uses convolutional layers and pooling layers to extract features and then uses up-convolutional layers to output a dense optical flow estimation. This CNN architecture is trained end-to-end in a supervised learning environment, using ground-truth optical flows. For the purpose of training with ground-truth flow labels, **FlowNet** introduces a synthetic dataset 'synthetic chairs'.

Following this, **DispNet** introduced another synthetic dataset used to train a CNN for real time disparity estimation. Extending the optical flow domain to scene flow, the CNN proposed in **DispNet** aims estimate scene flow by learning flow and disparity jointly.

Further work...

D.Teney 2016 introduces a lightweight CNN that uses signal processing principles for image contrast, phase, rotation and texture invariance...

FlowNet2.0 stacks networks to incrementally predict optical flow...

In a supervised training environment, CNN's require large labelled datasets to learn optical flow estimation. Real world datasets with ground truth optical flow labels are difficult to obtain. In the case of **FlowNet, ...**, they created synthetic datasets which they train on. Training on unrealistic datasets may impose an upper limit on the accuracy on the optical flow estimation on real data.

... attempts to...

Another approach is to use unsupervised learning.

3.3 Unsupervised CNN flow estimation

More recently we have seen works for unsupervised learning of optical flow using a CNN.

A. Ahmidi 2016...

Yu. Harley 2016 uses FlowNet Simple *ref* for its networks architecture. To train the network in an unsupervised setting, the photometric loss of the backward-warped-first-image and second image is used. This is done using a differential spatial transformer. This photometric loss is calculated for multiple levels during the upconvolutional layers. Furthermore a first order smoothness constraint on the flow.

Similarly, *Z.Ren 2017* uses *FlowNet* architecture in an unsupervised learning domain. This paper uses a differentiable bilinear warping function to calculate the photometric error of the optical flow estimation. Also, a first order smoothness constraint is imposed on the estimated flow. This method allows for end-to-end training using backpropagation since the bilinear warping function is differentiable.

Extending this, *UnFlow 2017* uses *FlowNet2.0* and considers occlusion in the loss function by estimating the flow in both directions and...

Furthermore, *UnFlow 2017* imposes second order smoothness on the flow which is shown in *...* to give better results.

Lai 2017 uses an adversarial network in contrast to the brightness constancy error from warping with the estimated flow.

These approaches, although currently not as good as the state-of-the-art supervised CNN learning have good potential for optical flow estimation going forward as they require no ground truth flow datasets. This also means that these approaches are not constrained by the implicit upper bound of performance from training on synthetic datasets.

3.4 stuff to fit in

penalty functions - charbonnier

Challenges:

-occlusion

-over-smoothing

-large displacements

-outliers

-textureless

-rotation

CNN's much faster than variation methods.

4 Implementation

4.1 unsupervised loss function

Like in [*refs*](#), we extend [*FlowNet*](#) to use a proxy flow error instead of using ground truth flows. The core of unsupervised flow estimation CNN's is a differentiable warping function [*refs*](#). Then the photometric loss between the warped second image and the ground truth first image this can be considered our data term in our loss function. Using the photometric error in optical flow estimation follows from the brightness constancy assumption [*ref*](#). In gradient and variation methods using only the brightness constancy results in an under-determined system. [*start*](#) However, training a neural network should be possible using only the photometric error. [*end*](#). Using [*eq horn and schunck*](#), we can replace the data term with our photometric error, giving [*eq*](#). This is equivalent to adding a smoothness constraint, which can be considered a regularizing term.

4.1.1 photometric loss

differentiable warping function - reference to section
Charbonnier loss

4.1.2 smoothness constraint

first order smoothness constraint
second order smoothness constraint?

4.1.3 occlusion error term?

4.2 network architecture

diagrams
introduce how loss functions are used in the architecture.

4.3 Differentiable warping

bilinear interpolation

4.3.1 Testing the warp function

using ground truth flow and comparing error of warped image

5 training

back-propagation
plots of error over epochs
tables

5.1 analysis

6 Training datasets

6.1 Flying chairs

6.2 etc

6.3 Data augmentation

7 Results

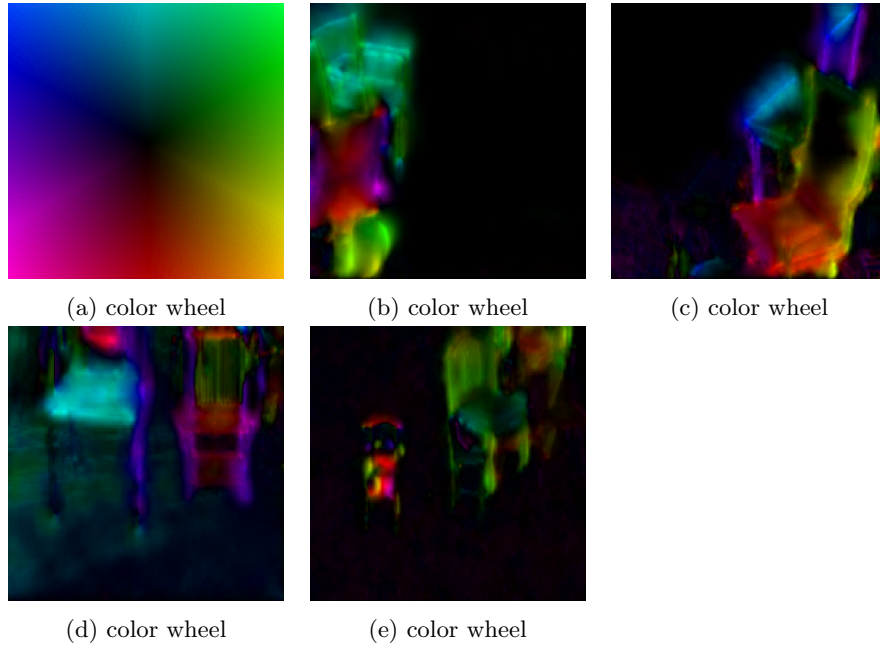


Figure 1: flow training data

qualitative - graphical results

quantitative - end-point / interpolation/ angular / etc error
tables

8 Analysis

9 Conclusion

10 Future work