Optical Flow

- Introduction

2016.08.08

Hyeongseok Kim

hskim@capp.snu.ac.kr

Optical Flow

$I(t), \{p_i\}$ $\quad$ $I(t+1)$ $\quad$ **Velocity vectors** $\{\vec{v}_i\}$

Optical Flow

$I(t), \{p_i\}$

$I(t+1)$

Velocity vectors $\{\vec{v}_i\}$

→ **Get 'Motion' information**

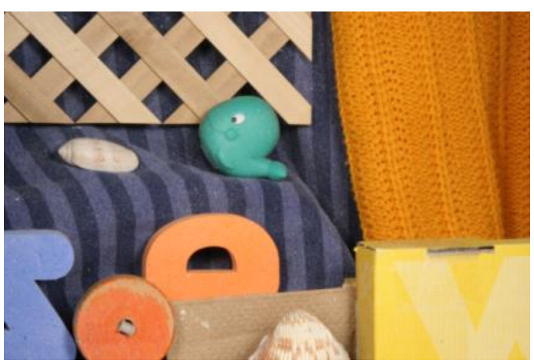
Optical Flow

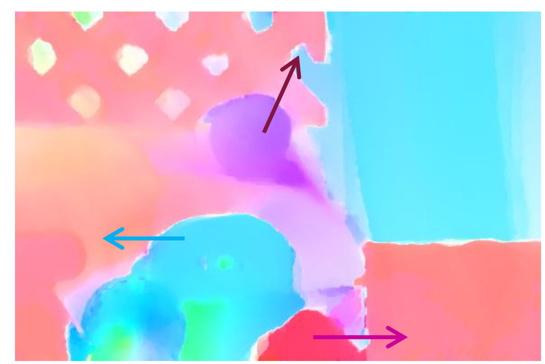
$I(t), \{p_i\}$ $\qquad$ $I(t+1)$

Velocity vectors $\{\vec{v}_i\}$

optical flow color encoding scheme

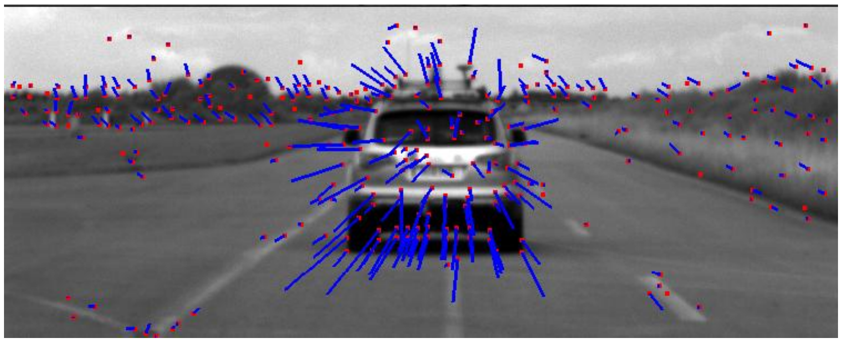
→ **Get ‘Motion’ information**

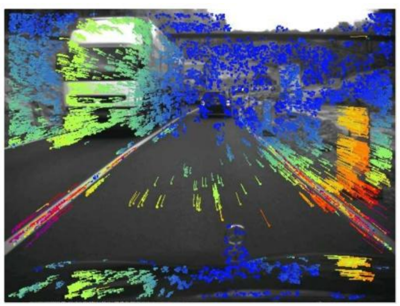


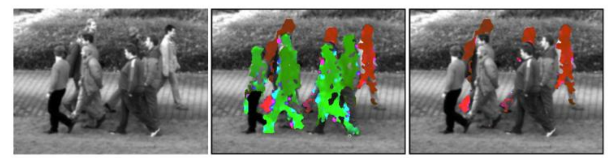


Optical Flow

→ Get 'Motion' information →







Optical Flow

Local method

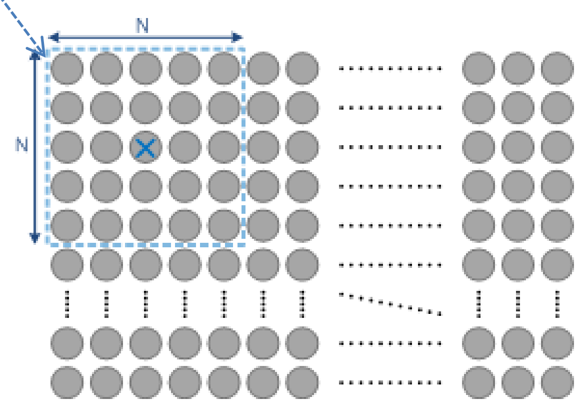
Brightness constancy
Spatial Coherence

$$I(x(t), y(t), t) = constant$$

$$\frac{\partial I}{\partial x}\frac{\partial x}{\partial t} + \frac{\partial I}{\partial y}\frac{\partial y}{\partial t} + \frac{\partial I}{\partial t} = 0$$

$$\begin{bmatrix} \boxed{\sum w I_{x_i} I_{x_i}} & \sum w I_{x_i} I_{y_i} \\ \sum w I_{x_i} I_{y_i} & \sum w I_{y_i} I_{y_i} \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} \sum w I_{x_i} I_{t_i} \\ \sum w I_{y_i} I_{t_i} \end{bmatrix}$$

ex) LK (Lucas-Kanade)



Global method

minimize Energy function

$$E(f) = \sum_{(p,q)\in\mathcal{N}} V(f_p, f_q) + \sum_{p\in\mathcal{P}} D_p(f_p)$$

ex) Horn-Schunck, TV-L1

Optical Flow

Local method

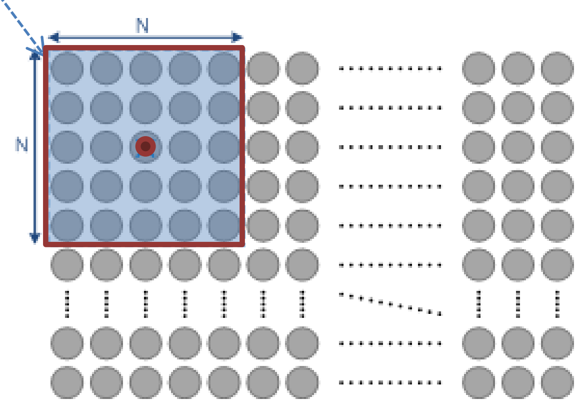
Brightness constancy
Spatial Coherence

$$I(x(t), y(t), t) = constant$$

$$\frac{\partial I}{\partial x}\frac{\partial x}{\partial t} + \frac{\partial I}{\partial y}\frac{\partial y}{\partial t} + \frac{\partial I}{\partial t} = 0$$

$$\begin{bmatrix} \boxed{\sum w I_{x_i} I_{x_i}} & \sum w I_{x_i} I_{y_i} \\ \sum w I_{x_i} I_{y_i} & \sum w I_{y_i} I_{y_i} \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = -\begin{bmatrix} \sum w I_{x_i} I_{t_i} \\ \sum w I_{y_i} I_{t_i} \end{bmatrix}$$

ex) LK (Lucas-Kanade)



Global method

minimize Energy function

$$E(f) = \sum_{(p,q)\in\mathcal{N}} V(f_p, f_q) + \sum_{p\in\mathcal{P}} D_p(f_p)$$

ex) Horn-Schunck, TV-L1

Optical Flow

Local method

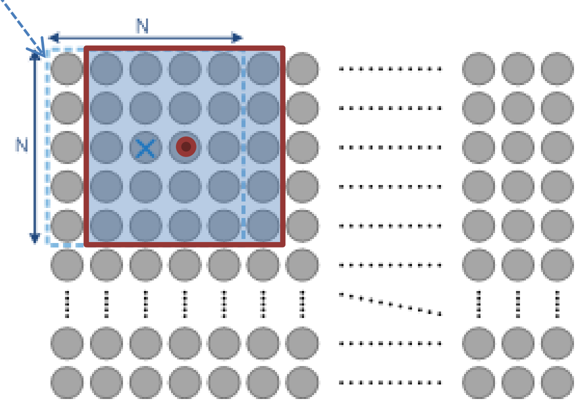
Brightness constancy
Spatial Coherence

$$\mathrm{I}(x(t), y(t), t) = constant$$

$$\frac{\partial I}{\partial x}\frac{\partial x}{\partial t} + \frac{\partial I}{\partial y}\frac{\partial y}{\partial t} + \frac{\partial I}{\partial t} = 0$$

$$\begin{bmatrix} \boxed{\sum w I_{x_i} I_{x_i}} & \sum w I_{x_i} I_{y_i} \\ \sum w I_{x_i} I_{y_i} & \sum w I_{y_i} I_{y_i} \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} \sum w I_{x_i} I_{t_i} \\ \sum w I_{y_i} I_{t_i} \end{bmatrix}$$

ex) LK (Lucas-Kanade)



Global method

minimize Energy function

$$E(f) = \sum_{(p,q)\in\mathcal{N}} V(f_p, f_q) + \sum_{p\in\mathcal{P}} D_p(f_p)$$

ex) Horn-Schunck, TV-L1

Optical Flow

Local method

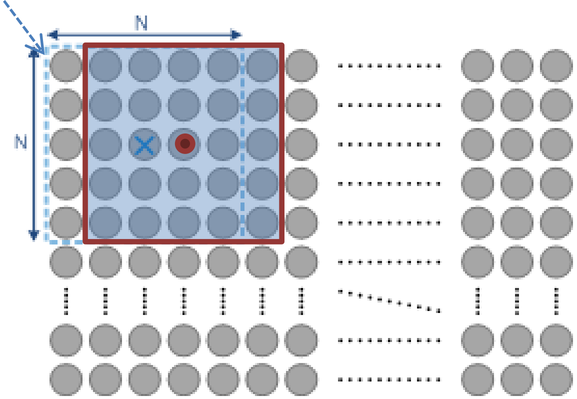
Brightness constancy
Spatial Coherence

$$I(x(t), y(t), t) = constant$$

$$\frac{\partial I}{\partial x}\frac{\partial x}{\partial t} + \frac{\partial I}{\partial y}\frac{\partial y}{\partial t} + \frac{\partial I}{\partial t} = 0$$

$$\begin{bmatrix} \sum w I_{x_i} I_{x_i} & \sum w I_{x_i} I_{y_i} \\ \sum w I_{x_i} I_{y_i} & \sum w I_{y_i} I_{y_i} \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} \sum w I_{x_i} I_{t_i} \\ \sum w I_{y_i} I_{t_i} \end{bmatrix}$$

ex) LK (Lucas-Kanade)

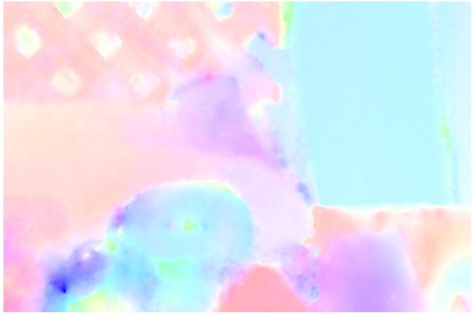


Global method

minimize Energy function

$$E(f) = \sum_{(p,q)\in\mathcal{N}} V(f_p, f_q) + \sum_{p\in\mathcal{P}} D_p(f_p)$$

ex) Horn-Schunck, TV-L1



Optical Flow

Local method

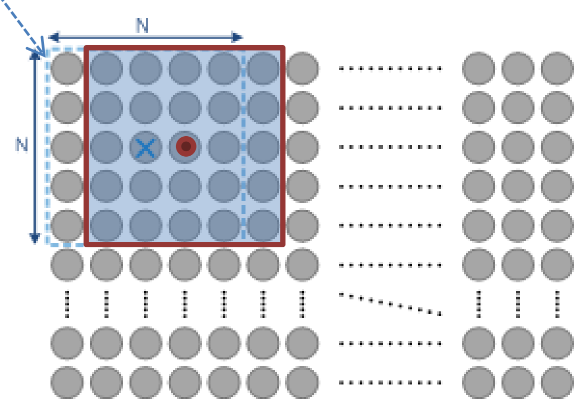
Brightness constancy
Spatial Coherence

$$I(x(t), y(t), t) = constant$$

$$\frac{\partial I}{\partial x}\frac{\partial x}{\partial t} + \frac{\partial I}{\partial y}\frac{\partial y}{\partial t} + \frac{\partial I}{\partial t} = 0$$

$$\begin{bmatrix} \boxed{\sum w I_{x_i} I_{x_i}} & \sum w I_{x_i} I_{y_i} \\ \sum w I_{x_i} I_{y_i} & \sum w I_{y_i} I_{y_i} \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} \sum w I_{x_i} I_{t_i} \\ \sum w I_{y_i} I_{t_i} \end{bmatrix}$$

ex) LK (Lucas-Kanade)

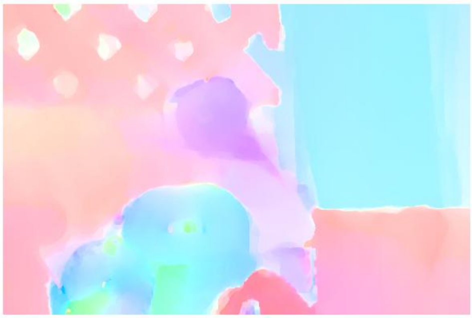


Global method

minimize Energy function

$$E(f) = \sum_{(p,q)\in\mathcal{N}} V(f_p, f_q) + \sum_{p\in\mathcal{P}} D_p(f_p)$$

ex) Horn-Schunck, TV-L1



Optical Flow

Local method

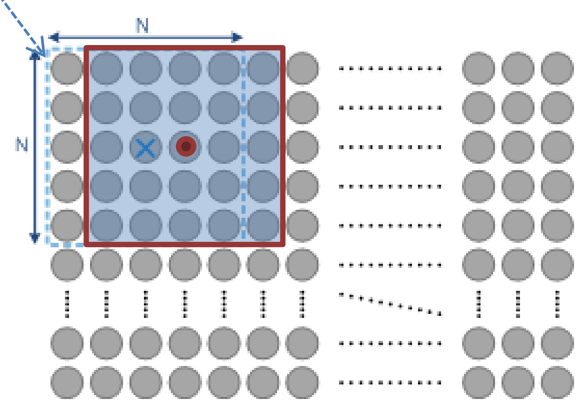
Brightness constancy
Spatial Coherence

$$\mathrm{I}(x(t), y(t), t) = constant$$

$$\frac{\partial I}{\partial x}\frac{\partial x}{\partial t} + \frac{\partial I}{\partial y}\frac{\partial y}{\partial t} + \frac{\partial I}{\partial t} = 0$$

$$\begin{bmatrix} \sum w I_{x_i} I_{x_i} & \sum w I_{x_i} I_{y_i} \\ \sum w I_{x_i} I_{y_i} & \sum w I_{y_i} I_{y_i} \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} \sum w I_{x_i} I_{t_i} \\ \sum w I_{y_i} I_{t_i} \end{bmatrix}$$

ex) LK (Lucas-Kanade)

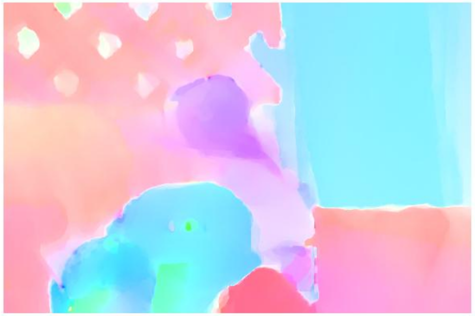


Global method

minimize Energy function

$$E(f) = \sum_{(p,q)\in\mathcal{N}} V(f_p, f_q) + \sum_{p\in\mathcal{P}} D_p(f_p)$$

ex) Horn-Schunck, TV-L1



Optical Flow

Local method

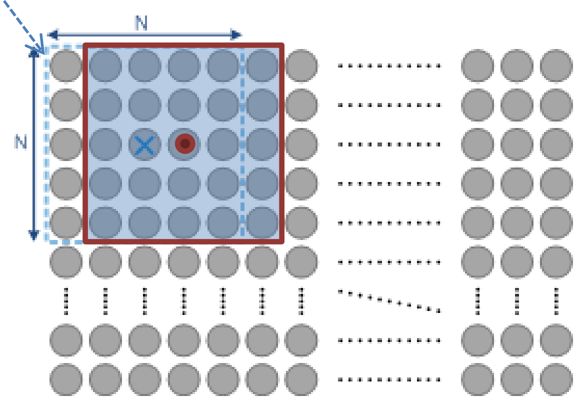
Brightness constancy
Spatial Coherence

$$I(x(t), y(t), t) = constant$$

$$\frac{\partial I}{\partial x}\frac{\partial x}{\partial t} + \frac{\partial I}{\partial y}\frac{\partial y}{\partial t} + \frac{\partial I}{\partial t} = 0$$

$$\begin{bmatrix} \boxed{\sum wI_{x_i}I_{x_i}} & \sum wI_{x_i}I_{y_i} \\ \sum wI_{x_i}I_{y_i} & \sum wI_{y_i}I_{y_i} \end{bmatrix} \begin{bmatrix} u \\ v \end{bmatrix} = - \begin{bmatrix} \sum wI_{x_i}I_{t_i} \\ \sum wI_{y_i}I_{t_i} \end{bmatrix}$$

ex) LK (Lucas-Kanade)

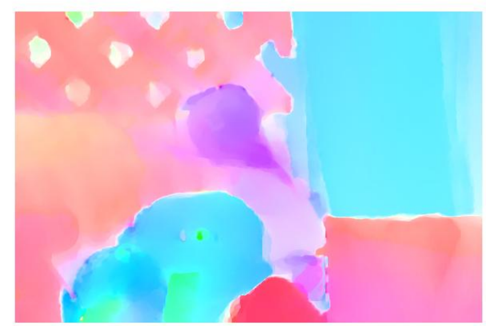


Global method

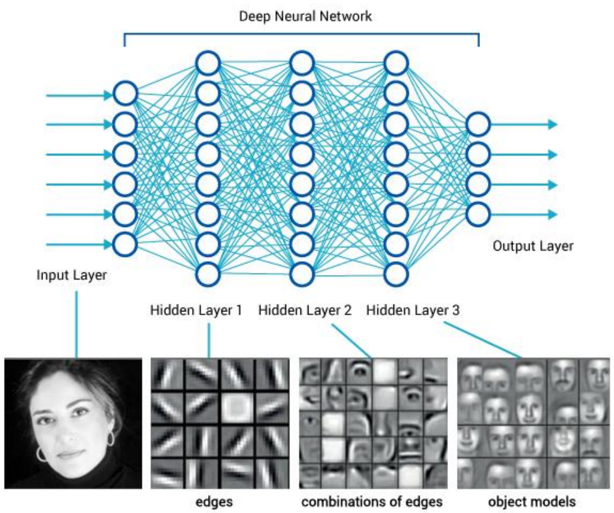
minimize Energy function

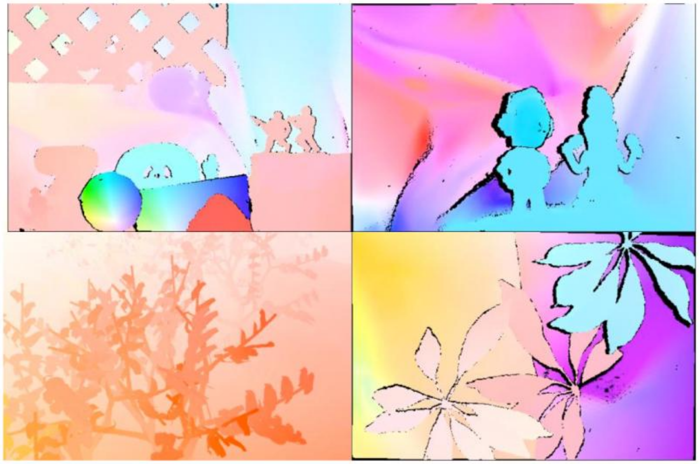
$$E(f) = \sum_{(p,q)\in\mathcal{N}} V(f_p, f_q) + \sum_{p\in\mathcal{P}} D_p(f_p)$$

ex) Horn-Schunck, TV-L1



Deep Learning for Optical Flow

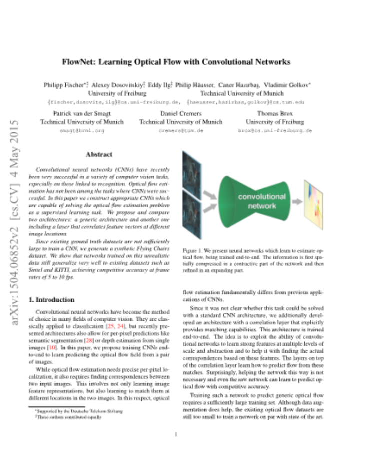




Per-pixel prediction task
- like semantic segmentation, depth prediction, keypoint prediction and edge detection

From a pair of images

Ex. DeepFlow(@ICCV 2013), EpicFlow(@CVPR 2015), FlowNet(@ICCV 2015), PathBatch(@CVPR 2016), …



arXiv:1504.06852v2 [cs.CV] 4 May 2015

FlowNet: Learning Optical Flow with Convolutional Networks

Philipp Fischer*†, Alexey Dosovitskiy†, Eddy Ilg†, Philip Häusser, Caner Hazırbaş, Vladimir Golkov*
University of Freiburg — Technical University of Munich
{fischer,dosovits,ilg}@cs.uni-freiburg.de, {haeusser,hazirbas,golkov}@cs.tum.edu

Patrick van der Smagt, Technical University of Munich, smagt@brml.org
Daniel Cremers, Technical University of Munich, cremers@tum.de
Thomas Brox, University of Freiburg, brox@cs.uni-freiburg.de

Abstract

Convolutional neural networks (CNNs) have recently been very successful in a variety of computer vision tasks, especially on those linked to recognition. Optical flow estimation has not been among the tasks where CNNs were successful. In this paper we construct appropriate CNNs which are capable of solving the optical flow estimation problem as a supervised learning task. We propose and compare two architectures: a generic architecture and another one including a layer that correlates feature vectors at different image locations.

Since existing ground truth datasets are not sufficiently large to train a CNN, we generate a synthetic Flying Chairs dataset. We show that networks trained on this unrealistic data still generalize very well to existing datasets such as Sintel and KITTI, achieving competitive accuracy at frame rates of 5 to 10 fps.

Figure 1. We present neural networks which learn to estimate optical flow, being trained end-to-end. The information is first spatially compressed in a contractive part of the network and then refined in an expanding part.

1. Introduction

Convolutional neural networks have become the method of choice in many fields of computer vision. They are classically applied to classification [25, 24], but recently presented architectures also allow for per-pixel predictions like semantic segmentation [28] or depth estimation from single images [10]. In this paper, we propose training CNNs end-to-end to learn predicting the optical flow field from a pair of images.

While optical flow estimation needs precise per-pixel localization, it also requires finding correspondences between two input images. This involves not only learning image feature representations, but also learning to match them at different locations in the two images. In this respect, optical flow estimation fundamentally differs from previous applications of CNNs.

Since it was not clear whether this task could be solved with a standard CNN architecture, we additionally developed an architecture with a correlation layer that explicitly provides matching capabilities. This architecture is trained end-to-end. The idea is to exploit the ability of convolutional networks to learn strong features at multiple levels of scale and abstraction and to help it with finding the actual correspondences based on these features. The layers on top of the correlation layer learn how to predict flow from these matches. Surprisingly, helping the network this way is not necessary and even the raw network can learn to predict optical flow with competitive accuracy.

Training such a network to predict generic optical flow requires a sufficiently large training set. Although data augmentation does help, the existing optical flow datasets are still too small to train a network on par with state of the art.

*Supported by the Deutsche Telekom Stiftung
†These authors contributed equally

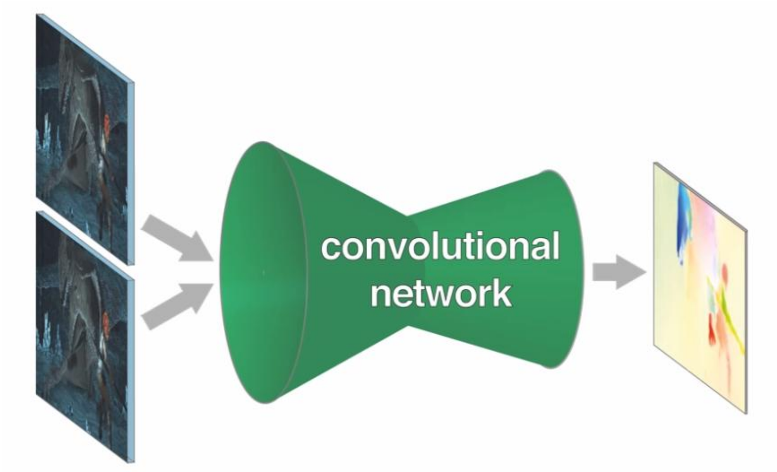
1

FlowNet

: Learning Optical Flow with Convolutional Networks

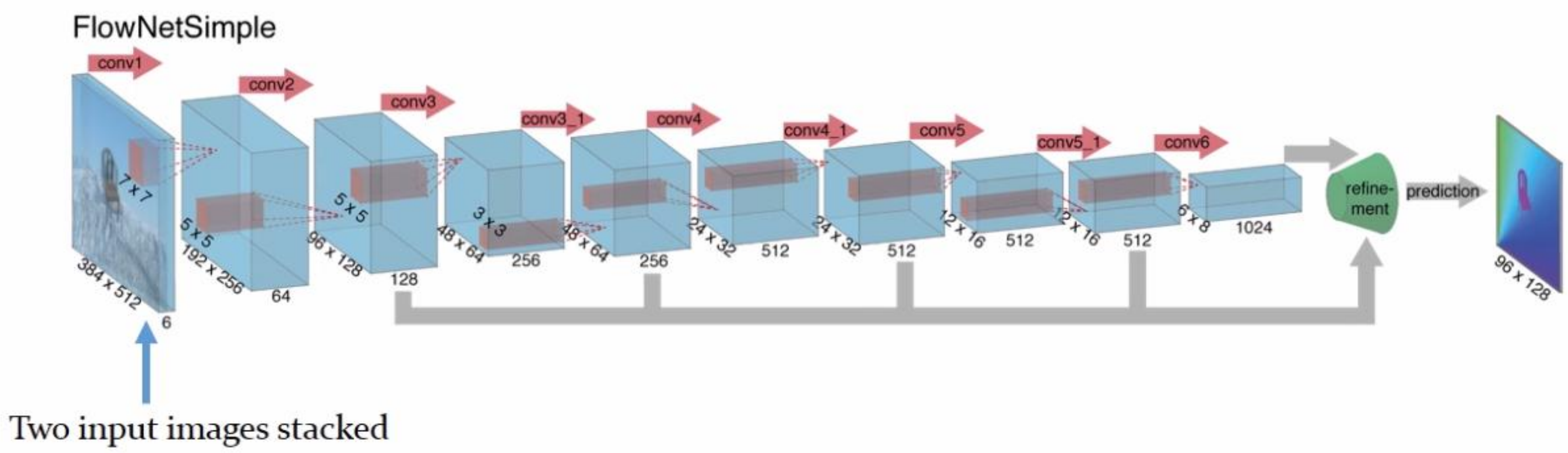
@ 2015 ICCV

FlowNet



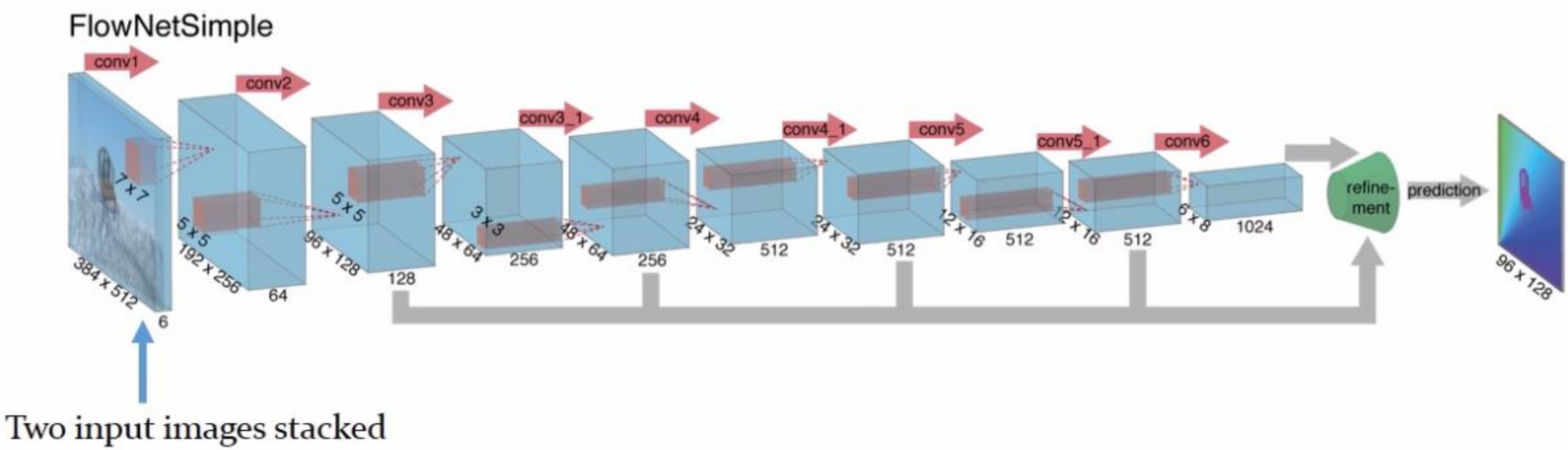
First spatially compressed in a contractive part of the network and then refined in an expanding part.

FlowNet



Stack both input images together and feed them through a network
- a rather generic architecture

FlowNet

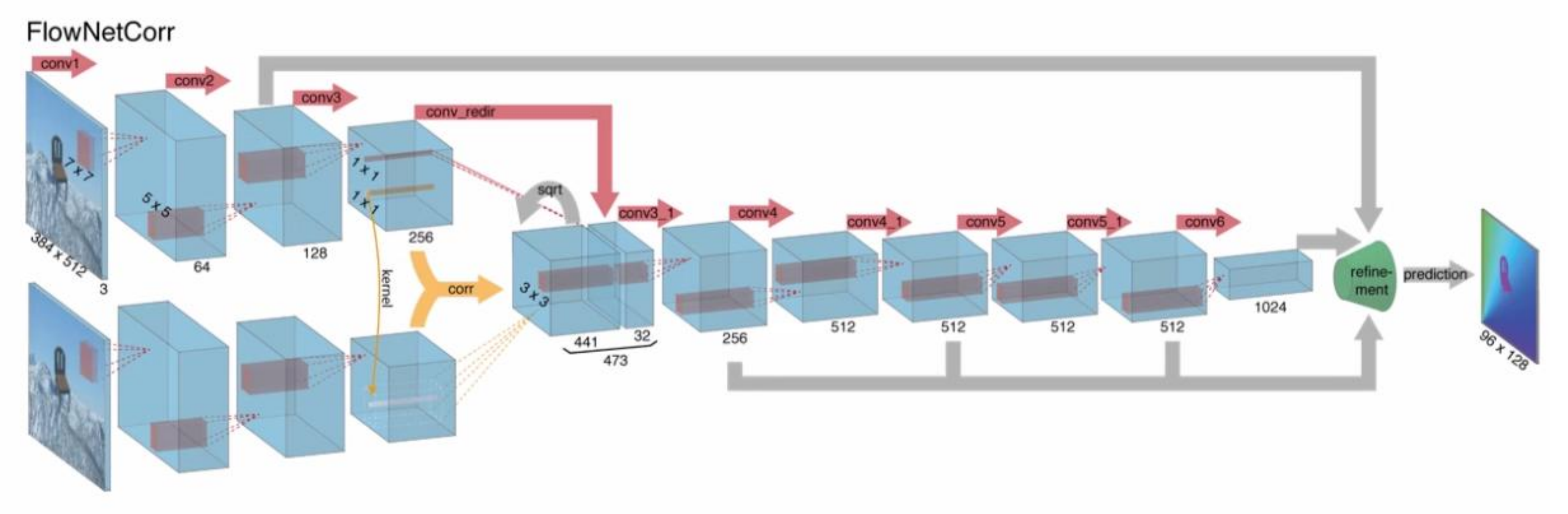


Stack both input images together and feed them through a network
- a rather generic architecture

If this network is large enough, it could learn to predict optical flow.
However, we can never be sure that a local gradient optimization like gradient descent can get the network to this point.

→Therefore, it could be beneficial to hand-design an architecture which is less generic, but may perform better with the given data and optimization techniques. → "FlowNetCorr"

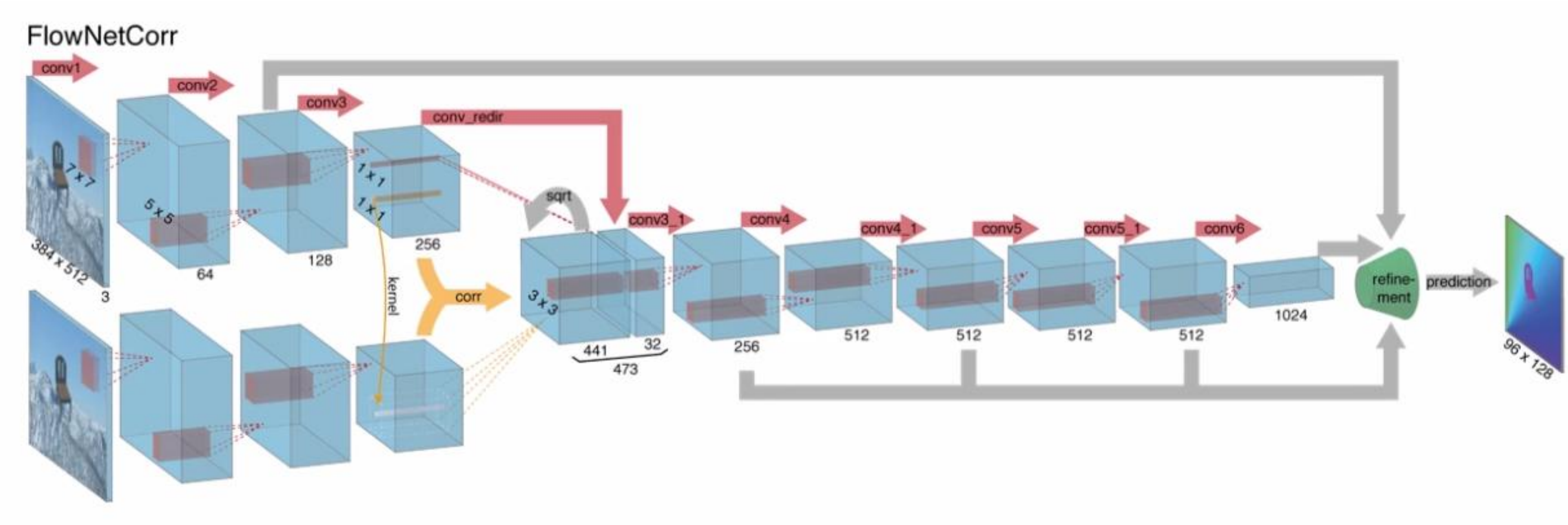
FlowNet



Two separate steps.
First produce meaningful representations of the two images separately
And then combine them on a higher level (by correlation layer)

→ Roughly resembles the standard matching approach.
when one first extracts features from patches of both images
and then compares those feature vectors.

FlowNet



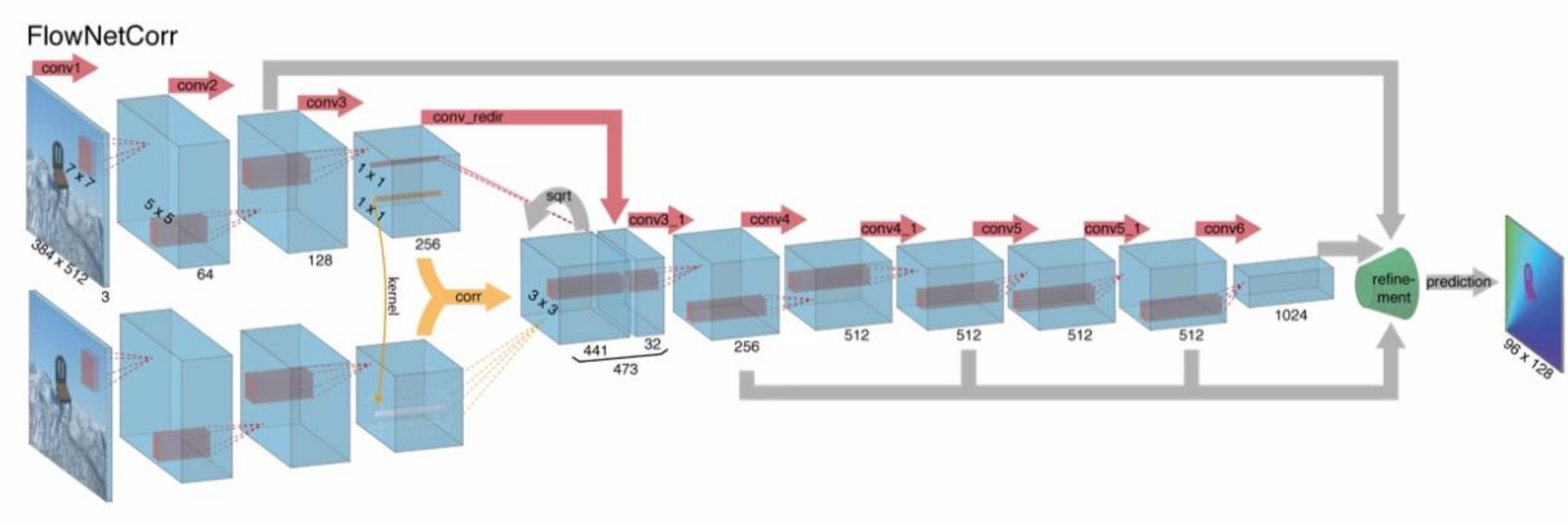
'Correlation layer' helps the network find correspondences.

Given two multi-channel feature maps
(with w, h, and c being their width, heig $\mathbf{f}_1, \mathbf{f}_2 : \mathbb{R}^2 \rightarrow \mathbb{R}^c$ r of channels),

'correlation' of two patches centered at $\mathbf{x}_1$ in the first map and $\mathbf{x}_2$ in the second map :

$$c(\mathbf{x}_1, \mathbf{x}_2) = \sum_{\mathbf{o} \in [-k,k] \times [-k,k]} \langle \mathbf{f}_1(\mathbf{x}_1 + \mathbf{o}), \mathbf{f}_2(\mathbf{x}_2 + \mathbf{o}) \rangle$$

FlowNet



'Correlation layer'

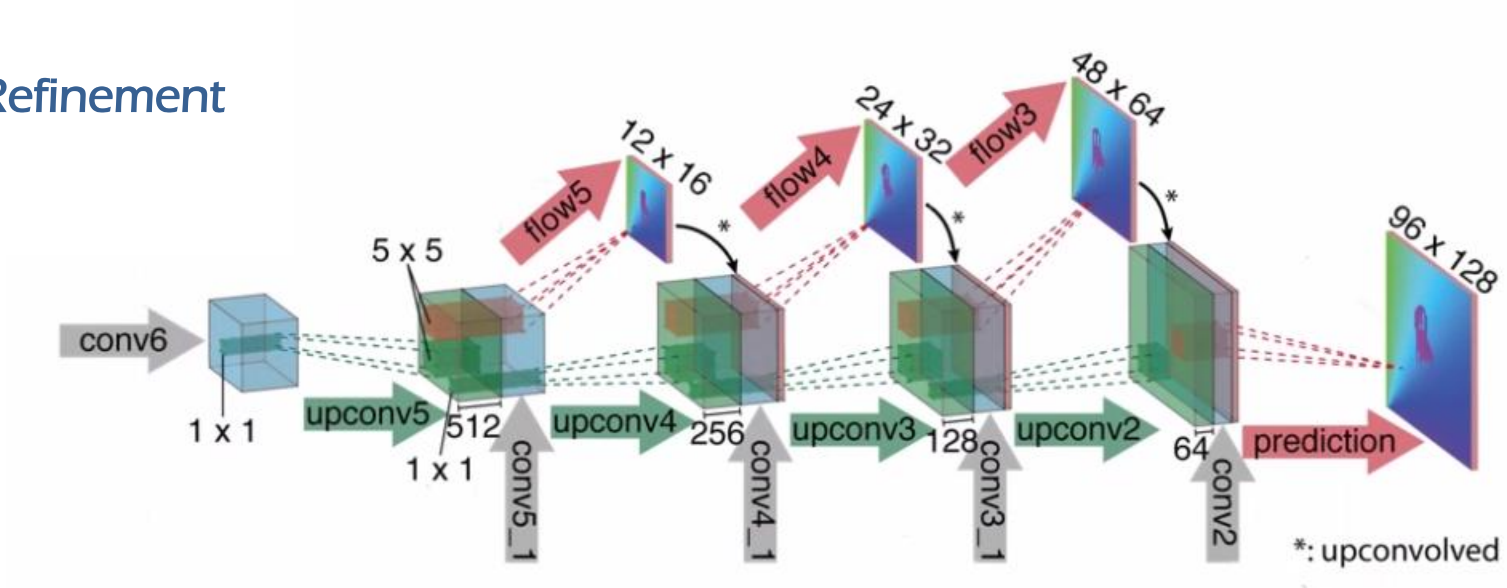
Comparing all patch combinations : $w^2 \cdot h^2$ computations → inefficient (too large)

Maximum displacement d, : $w \times h \times D^2$ computations(outputs), $D := 2d + 1$

- limit the maximum displacement for comparisons
- strides s_1 and s_2 to quantize $\mathbf{x}_1$ globally and to quantize $\mathbf{x}_2$ within the neighborhood centered around .

FlowNet

Refinement



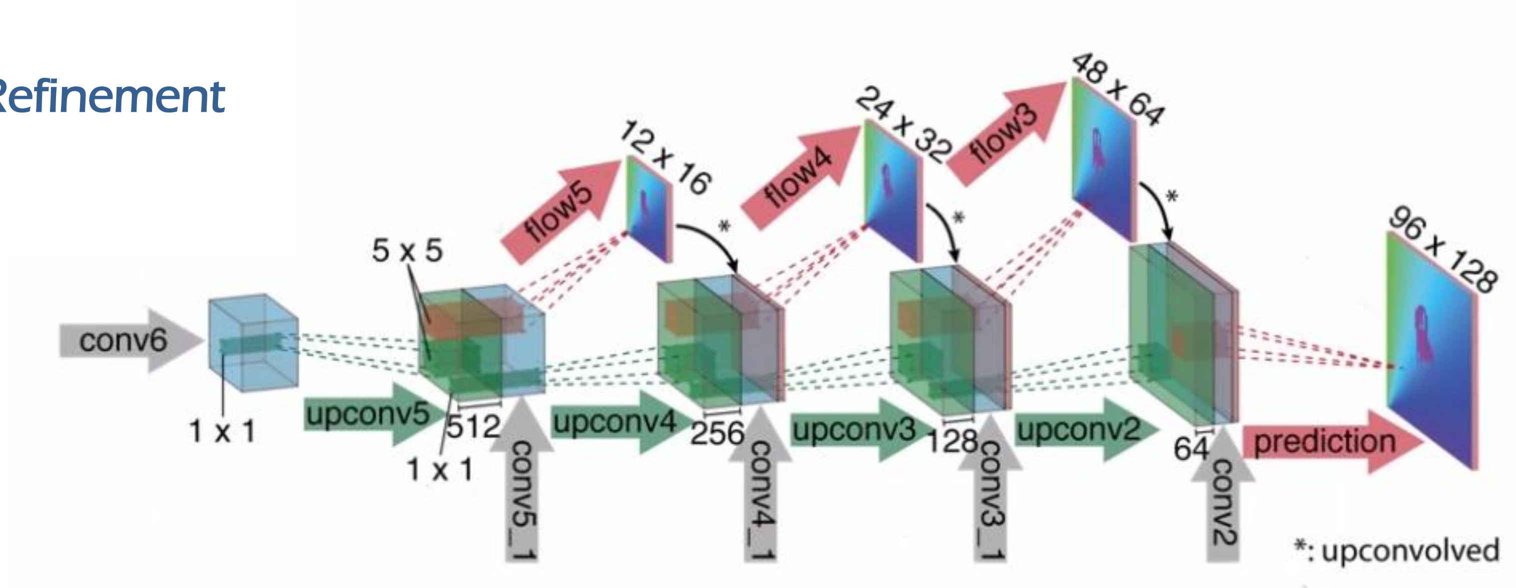
To provide dense per-pixel predictions
we need to refine the coarse pooled representation.

Apply the 'upconvolution' to feature maps,
and concatenate it with corresponding feature maps from the 'contractive' part of the network and an upsampled coarser flow prediction (if available).

→ Preserve both high-level information passed from coarser feature maps
and fine local information provided in lower layer feature maps.

FlowNet

Refinement



Repeat 4 times

- resulting in a predicted flow for which the resolution is still 4 times smaller than the input.

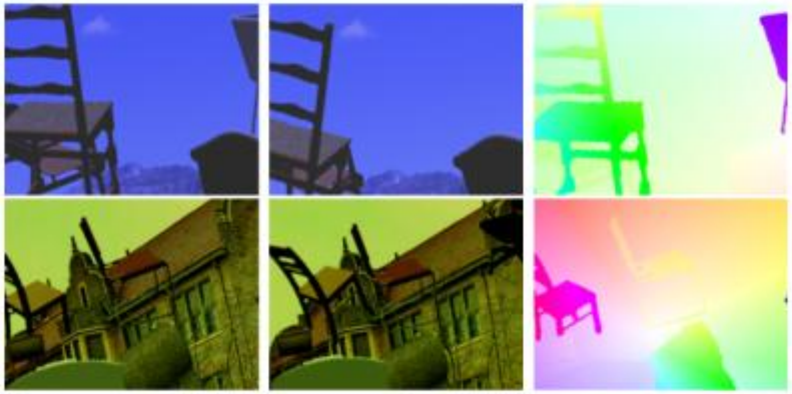
- further refinement does not significantly improve the results, compared to computationally less expensive bilinear upsampling to full image resolution.

- Alternative scheme : bilinear upsampling → variational approach

Training Data

	Frame pairs	Frames with ground truth	Ground truth density per frame
Middlebury	72	8	100%
KITTI	194	194	∽50%
Sintel	1,041	1,041	100%
Flying Chairs	22,872	22,872	100%

Generated image pair



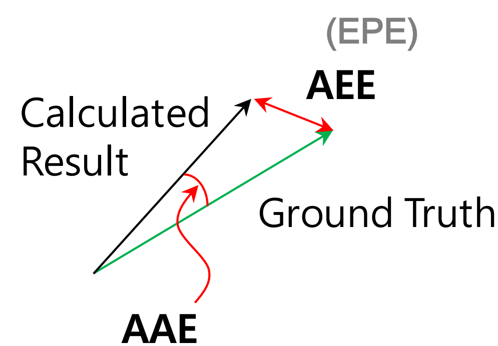
Augmented image pair

- geometric transformation
  : translation, rotation, scaling
- Gaussian noise
- brightness, contrast, gamma, color

Results

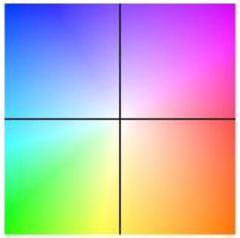
Method	Sintel Clean		Sintel Final		KITTI		Middlebury train		Middlebury test		Chairs	Time (sec)	
	train	test	train	test	train	test	AEE	AAE	AEE	AAE	test	CPU	GPU
EpicFlow [30]	2.40	4.12	3.70	6.29	3.47	3.8	0.31	3.24	0.39	3.55	2.94	16	-
DeepFlow [35]	3.31	5.38	4.56	7.21	4.58	5.8	0.21	3.04	0.42	4.22	3.53	17	-
EPPM [3]	-	6.49	-	8.38	-	9.2	-	-	0.33	3.36	-	-	0.2
LDOF [6]	4.29	7.56	6.42	9.12	13.73	12.4	0.45	4.97	0.56	4.55	3.47	65	2.5
FlowNetS	4.50	7.42	5.45	8.43	8.26	-	1.09	13.28	-	-	2.71	-	0.08
FlowNetS+v	3.66	6.45	4.76	7.67	6.50	-	0.33	3.87	-	-	2.86	-	1.05
FlowNetS+ft	(3.66)	6.96	(4.44)	7.76	7.52	9.1	0.98	15.20	-	-	3.04	-	0.08
FlowNetS+ft+v	(2.97)	6.16	(4.07)	7.22	6.07	7.6	0.32	3.84	0.47	4.58	3.03	-	1.05
FlowNetC	4.31	7.28	5.87	8.81	9.35	-	1.15	15.64	-	-	2.19	-	0.15
FlowNetC+v	3.57	6.27	5.25	8.01	7.45	-	0.34	3.92	-	-	2.61	-	1.12
FlowNetC+ft	(3.78)	6.85	(5.28)	8.51	8.79	-	0.93	12.33	-	-	2.27	-	0.15
FlowNetC+ft+v	(3.20)	6.08	(4.83)	7.88	7.31	-	0.33	3.81	0.50	4.52	2.67	-	1.12

AAE and AEE with several datasets



Results

optical flow color encoding scheme



with Sintel dataset

Video

https://youtu.be/k_wkDLJ8lJE

Thank you