WashU Vision & Learning Group

https://vlg.seas.wustl.edu/

PI: Ayan Chakrabarti (ayan@wustl.edu)

An Overview of Computer Vision & Computational Photography

CSE 591: Sep 19, 2018

INTRODUCTION

What is Computer Vision ?

Computational Systems to make sense

of the physical world by looking at

images and videos

INTRODUCTION

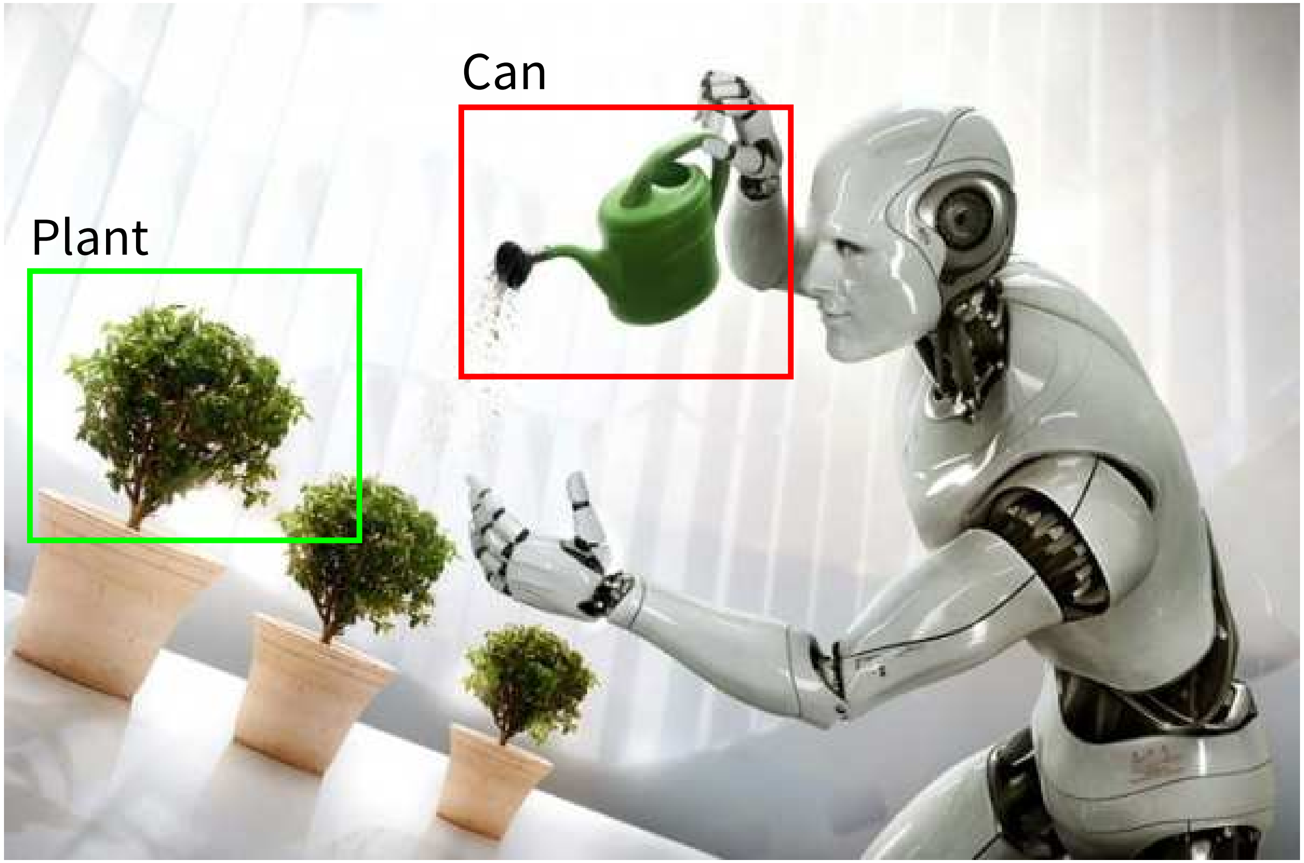
[credit: http://www.blutsbrueder-design.com]

INTRODUCTION

INTRODUCTION

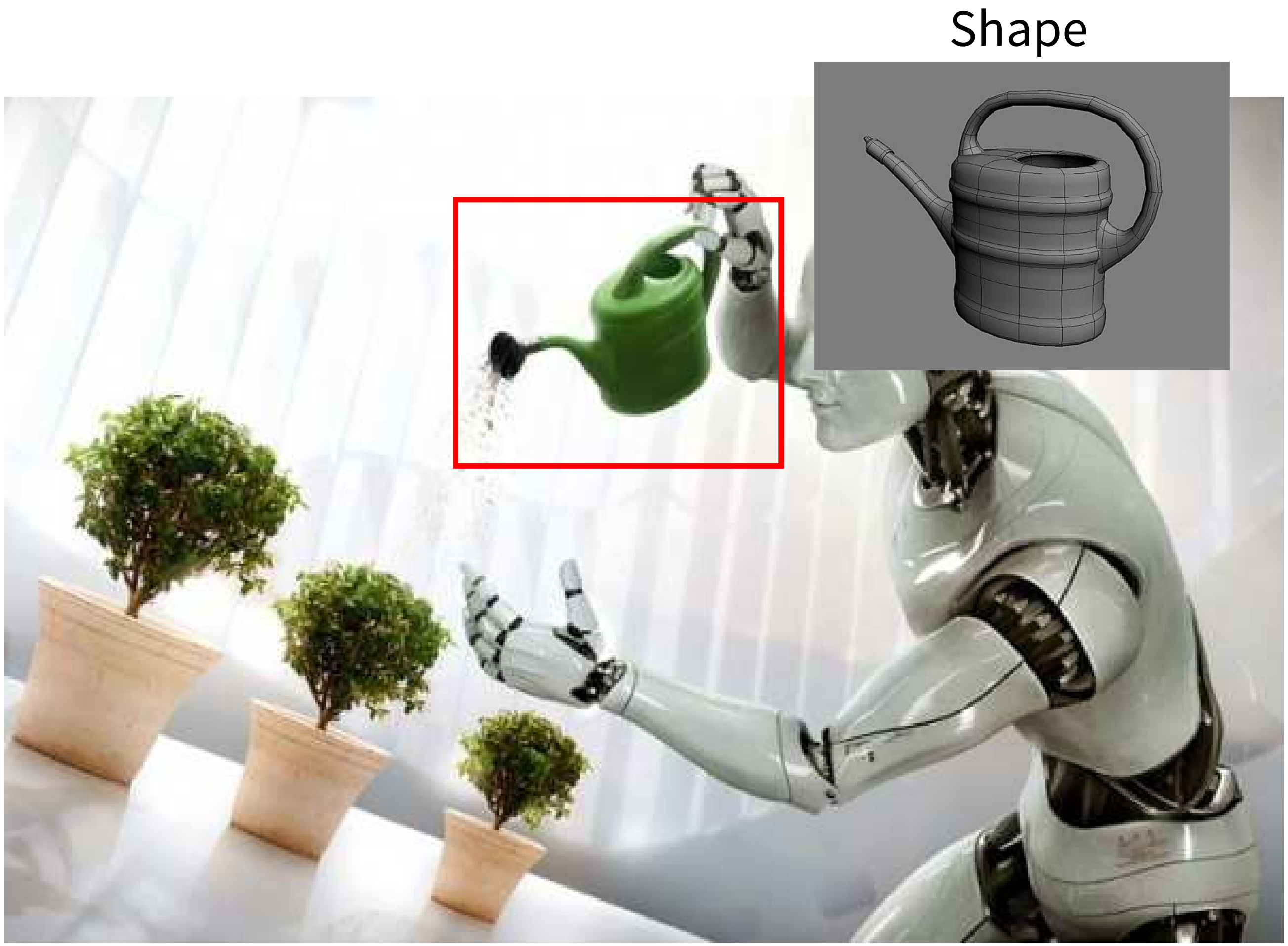
INTRODUCTION

INTRODUCTION



Recognize Objects

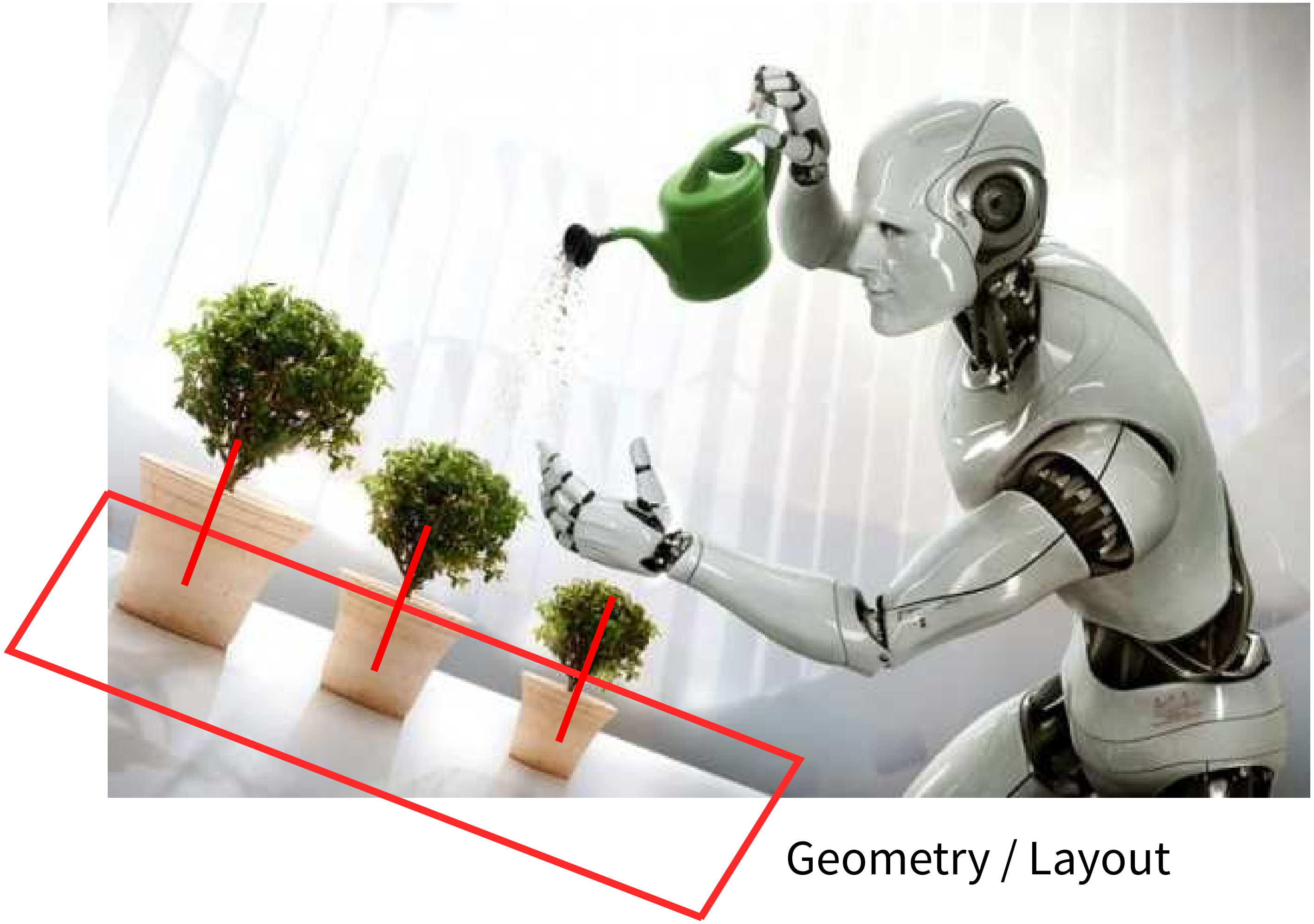
INTRODUCTION



INTRODUCTION

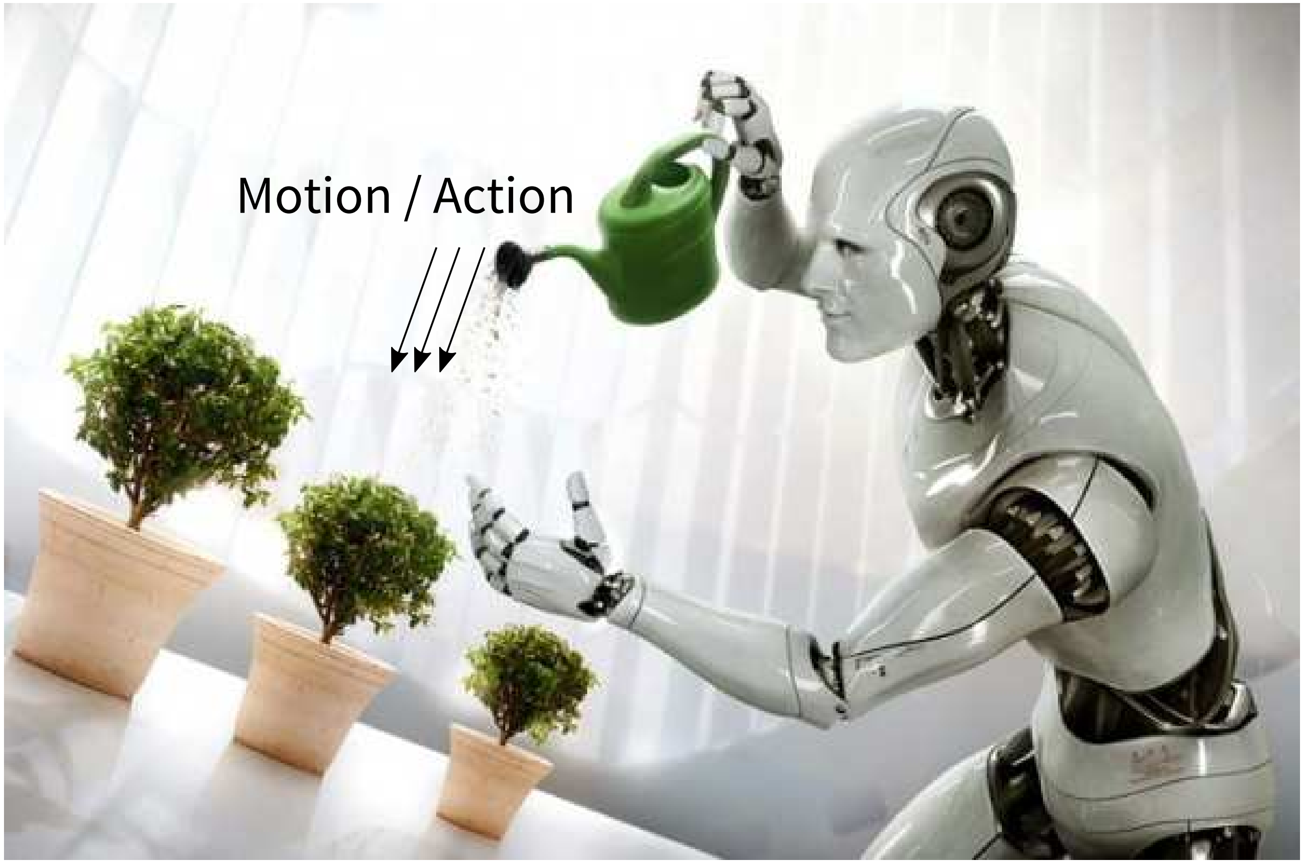
Classify Scene

INTRODUCTION



Geometry / Layout

INTRODUCTION



INTRODUCTION

Identify Materials

INTRODUCTION

Surface Properties

INTRODUCTION

Computer Vision

- Develop Algorithms that extract a description of the world from images

Computational Photography

- Think of modified cameras and acquisition setups that make this extraction easier

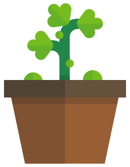
INTRODUCTION

Broad Overview of (many a) Vision Algorithm

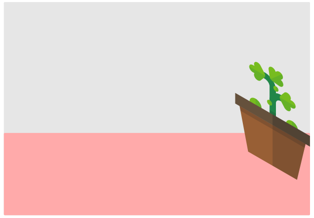
INTRODUCTION

Broad Overview of (many a) Vision Algorithm

1. Understand the Image Formation Model: Scene to Image



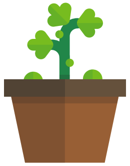
$$I = \mathcal{F}(S)$$



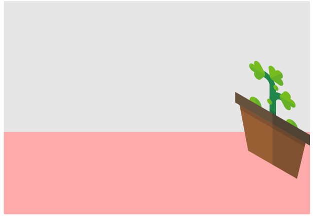
INTRODUCTION

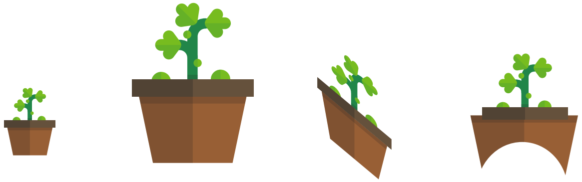
Broad Overview of (many a) Vision Algorithm

1. Understand the Image Formation Model: Scene to Image



$$I = \mathcal{F}(S)$$





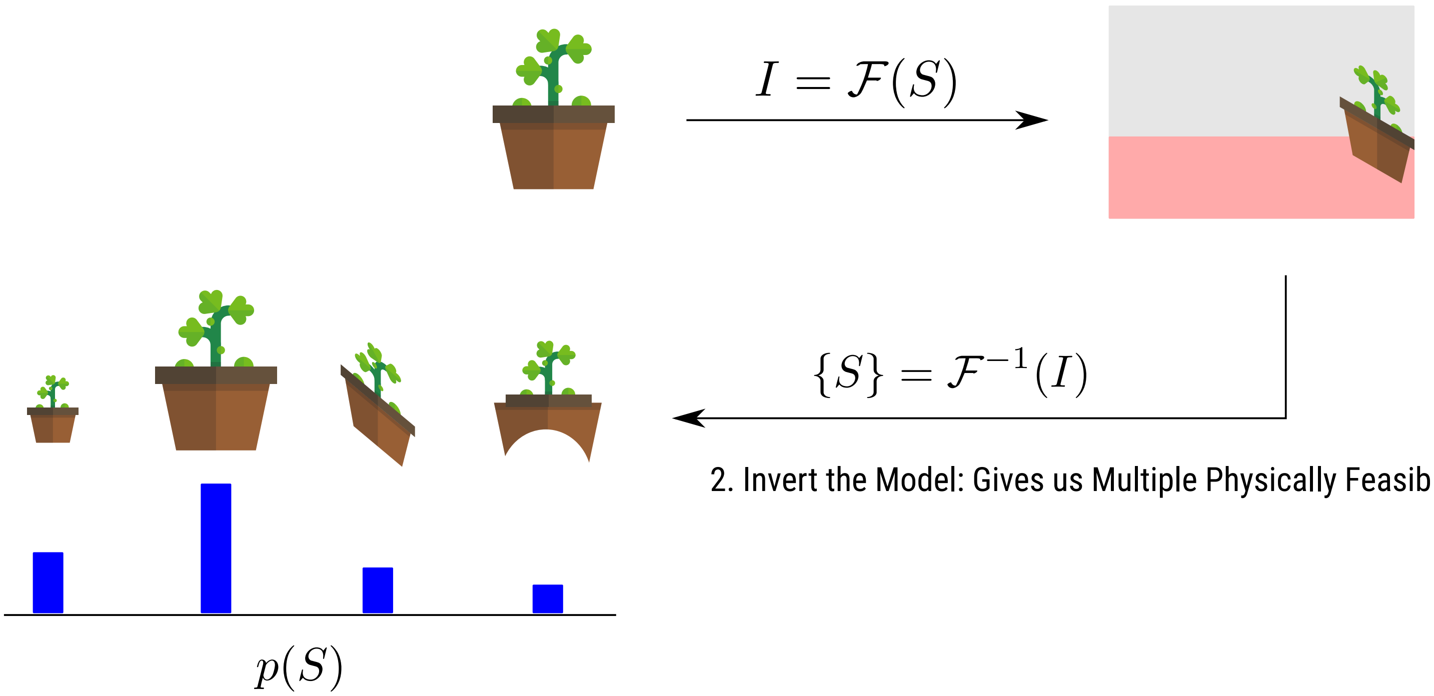
$$\{S\} = \mathcal{F}^{-1}(I)$$

2. Invert the Model: Gives us Multiple Physically Feasible Solutions

INTRODUCTION

Broad Overview of (many a) Vision Algorithm

1. Understand the Image Formation Model: Scene to Image



2. Invert the Model: Gives us Multiple Physically Feasible Solutions

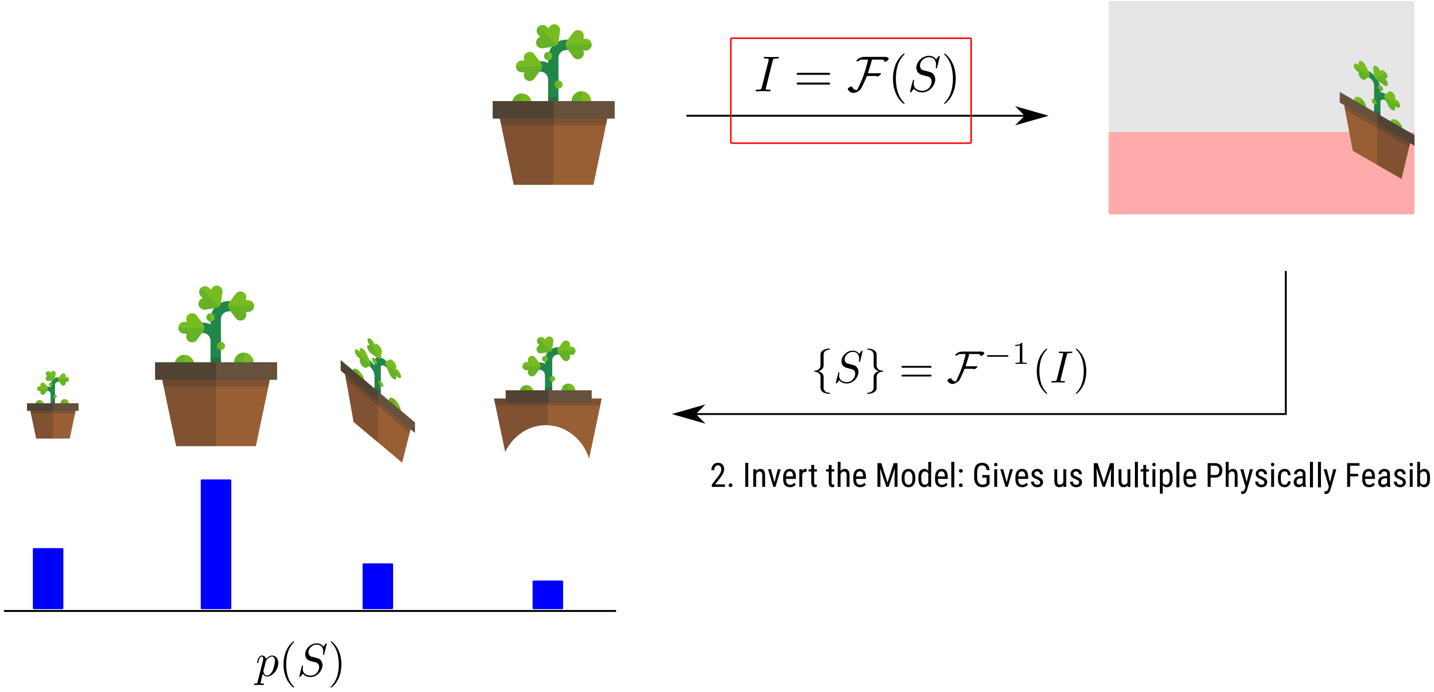
3. Learn What Natural Scenes Look Like: Use to select likely scene among those that are feasible

INTRODUCTION

Broad Overview of (many a) Vision Algorithm

1. Understand the Image Formation Model: Scene to Image

4. Computational Photography: Modify the Image Formation Model to make measurements more informative



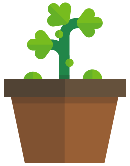
2. Invert the Model: Gives us Multiple Physically Feasible Solutions

3. Learn What Natural Scenes Look Like: Use to select likely scene among those that are feasible

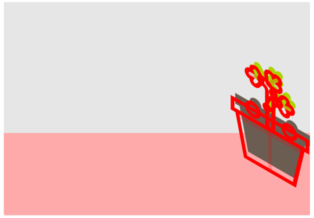
INTRODUCTION

Broad Overview of (many a) Vision Algorithm

1. Understand the Image Formation Model: Scene to Image



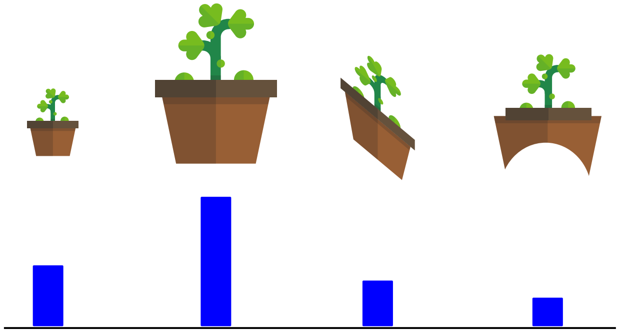
$$I = \mathcal{F}(S)$$



4. Computational Photography: Modify the Image Formation Model to make measurements more informative

$$\{S\} = \mathcal{F}^{-1}(I)$$

2. Invert the Model: Gives us Multiple Physically Feasible Solutions



$$p(S)$$

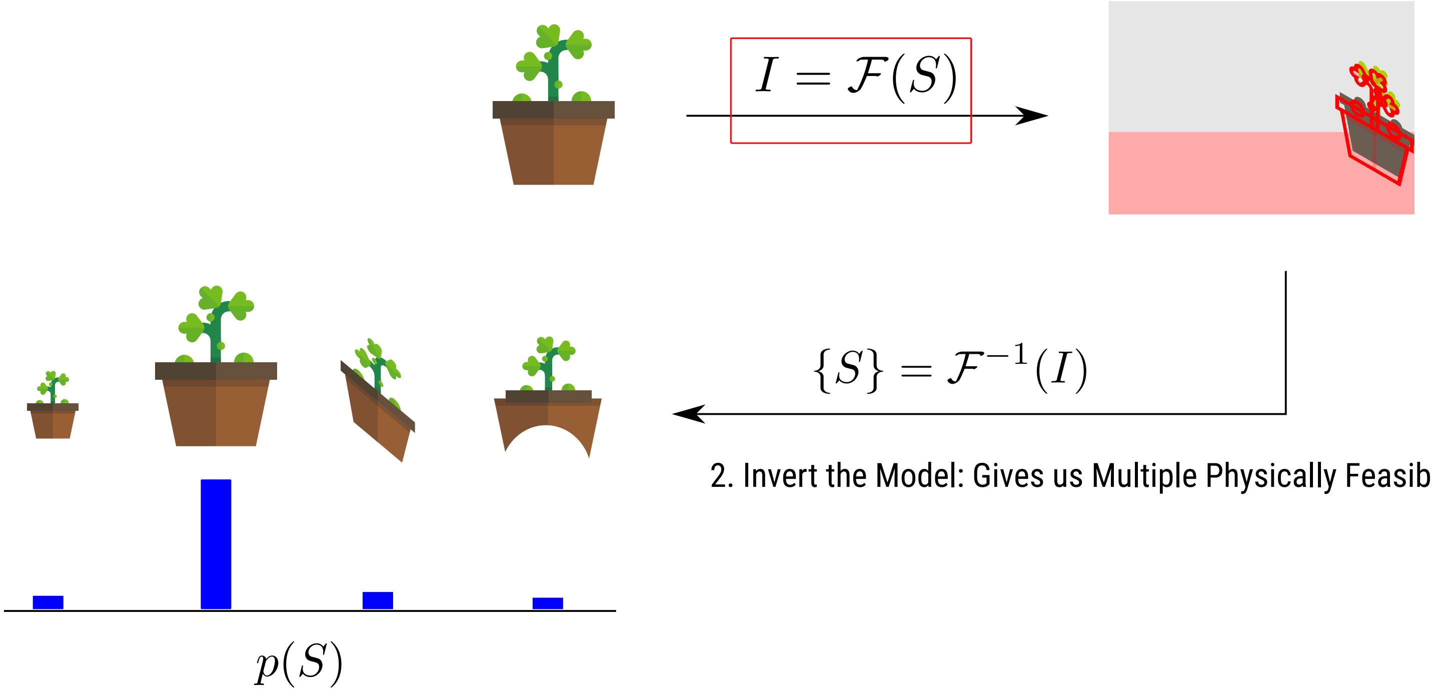
3. Learn What Natural Scenes Look Like: Use to select likely scene among those that are feasible

INTRODUCTION

Broad Overview of (many a) Vision Algorithm

1. Understand the Image Formation Model: Scene to Image

4. Computational Photography: Modify the Image Formation Model to make measurements more informative



2. Invert the Model: Gives us Multiple Physically Feasible Solutions

3. Learn What Natural Scenes Look Like: Use to select likely scene among those that are feasible

REAL-WORLD IMPACT

REAL-WORLD IMPACT

ICCP 2018 Sponsors

Gold

Qualcomm
intel

Silver

Adobe
جامعة الملك عبدالله
للعلوم والتقنية
King Abdullah University of
Science and Technology
oculus
Google

Bronze

SAMSUNG
OMRON
Snap Inc.
NEC Laboratories
America
Relentless passion for innovation
QUASAR MR
amazon

WHAT DOES VISION RESEARCH LOOK LIKE ?

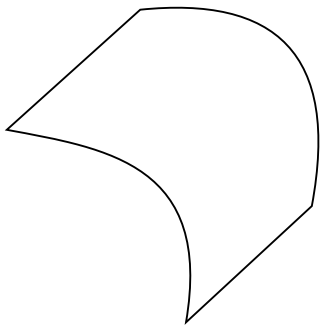
As a Grad Student working on a problem, you will have to:

- Understand the physics, geometry, optics, etc. of the setup.
- Understand to what degree the image formation process is invertible, characterize the ambiguity.
- Figure out how to the statistics of natural images could resolve this ambiguity.
- Use this to choose a model / architecture.
- Figure out how to train / learn parameters of this model.
- Develop an algorithm to use this model for actual inference.
- Make sure this is efficient and practical.

CASE STUDY: SHAPE FROM CONTROLLED LIGHTING

Photometric Stereo

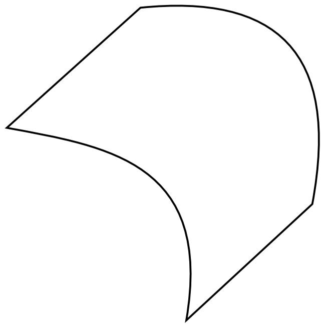
Use the fact that intensity depends on relative angle between surface normal and light source.



CASE STUDY: SHAPE FROM CONTROLLED LIGHTING

Photometric Stereo

Use the fact that intensity depends on relative angle between surface normal and light source.

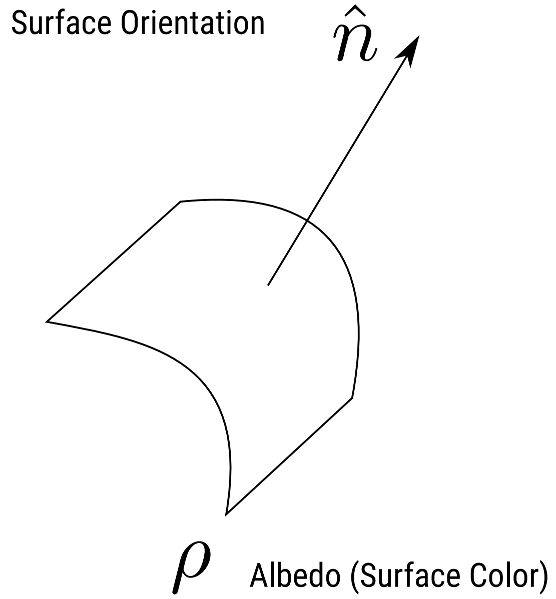


ρ Albedo (Surface Color)

CASE STUDY: SHAPE FROM CONTROLLED LIGHTING

Photometric Stereo

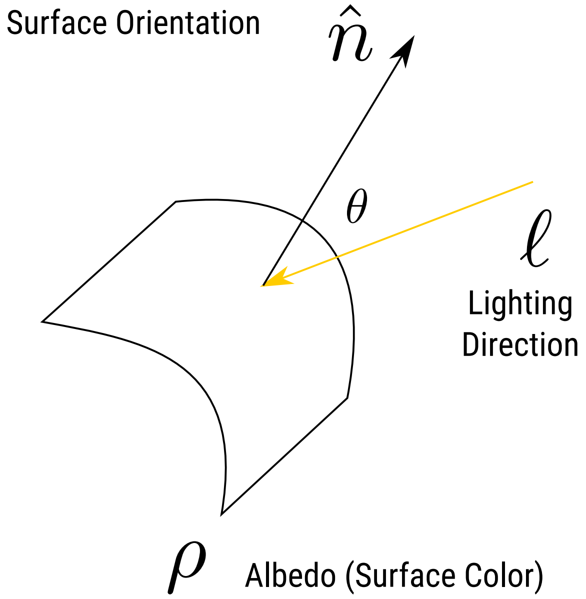
Use the fact that intensity depends on relative angle between surface normal and light source.



CASE STUDY: SHAPE FROM CONTROLLED LIGHTING

Photometric Stereo

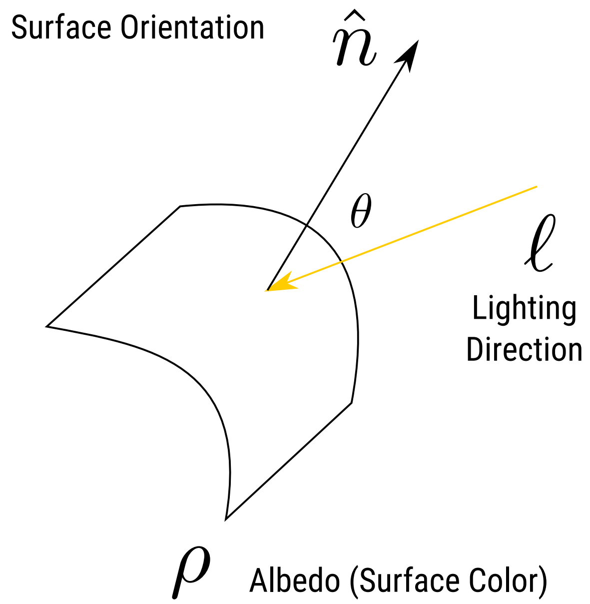
Use the fact that intensity depends on relative angle between surface normal and light source.



CASE STUDY: SHAPE FROM CONTROLLED LIGHTING

Photometric Stereo

Use the fact that intensity depends on relative angle between surface normal and light source.

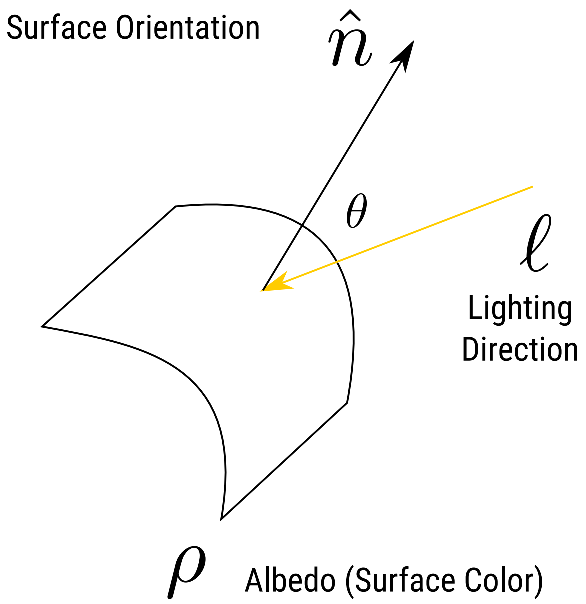


$$I = \rho \cos \theta$$

CASE STUDY: SHAPE FROM CONTROLLED LIGHTING

Photometric Stereo

Use the fact that intensity depends on relative angle between surface normal and light source.

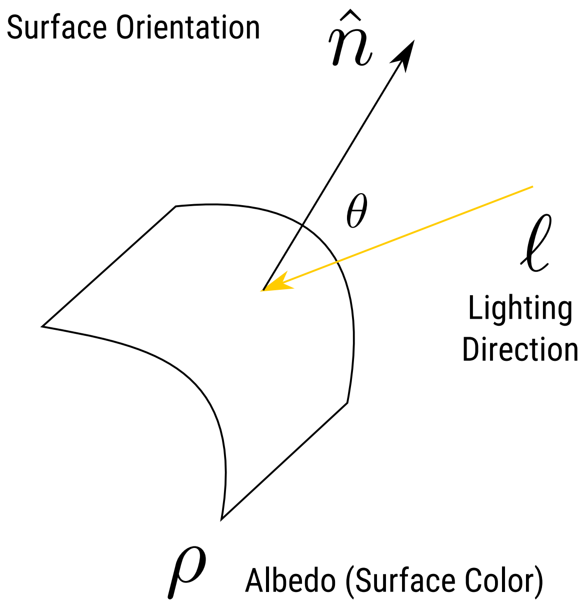


$$I = \rho \cos\theta = \rho < \hat{n}, \ell >$$

CASE STUDY: SHAPE FROM CONTROLLED LIGHTING

Photometric Stereo

Use the fact that intensity depends on relative angle between surface normal and light source.



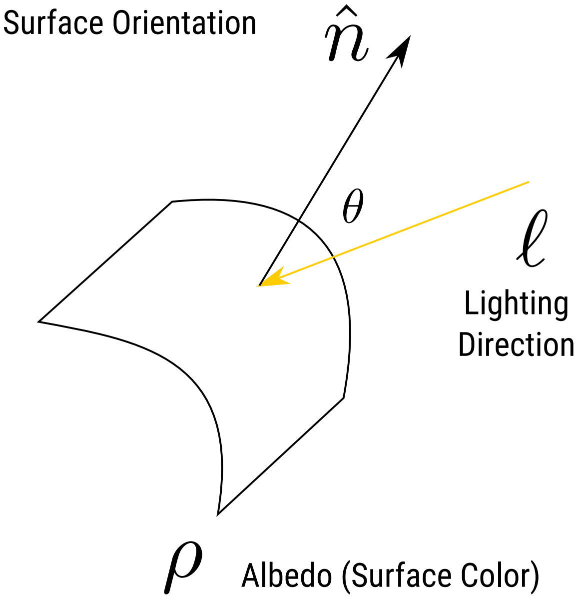
$$I = \rho \cos \theta = \rho < \hat{n}, \ell >$$

Known Lighting

CASE STUDY: SHAPE FROM CONTROLLED LIGHTING

Photometric Stereo

Use the fact that intensity depends on relative angle between surface normal and light source.



$$I = \rho \cos\theta = \rho < \hat{n}, \ell >$$

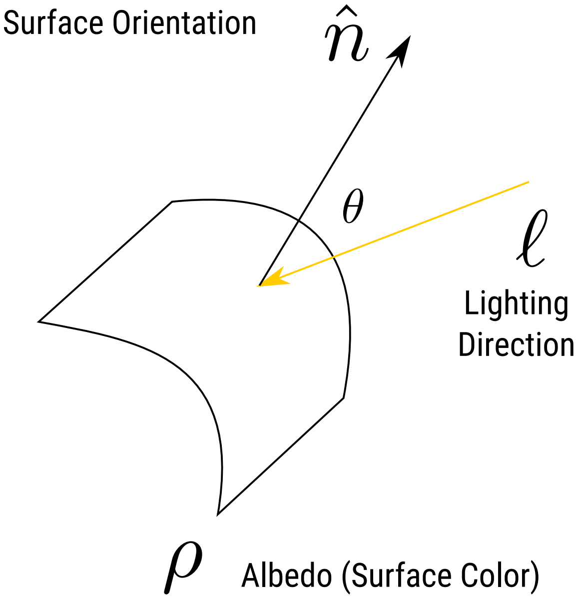
Known Lighting

One Observation: Three Unknowns

CASE STUDY: SHAPE FROM CONTROLLED LIGHTING

Photometric Stereo

Use the fact that intensity depends on relative angle between surface normal and light source.



$$I = \rho \cos \theta = \rho < \hat{n}, \ell >$$

Known Lighting

One Observation: Three Unknowns

Take multiple images with different lighting

CASE STUDY: SHAPE FROM CONTROLLED LIGHTING

Photometric Stereo

CASE STUDY: SHAPE FROM CONTROLLED LIGHTING

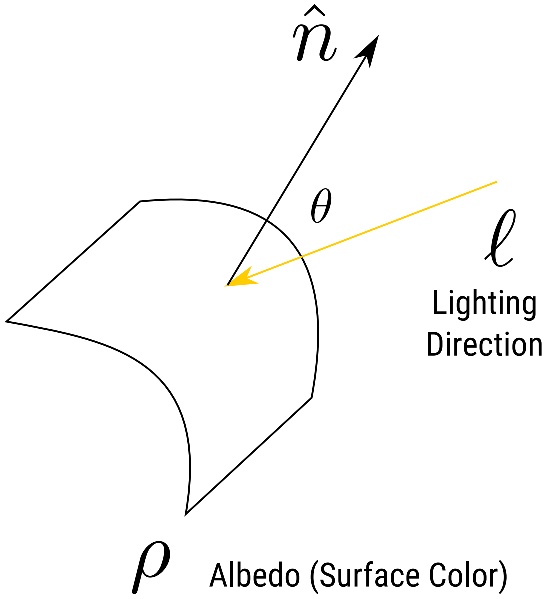
Photometric Stereo

Great, but requires you to take multiple images. What if the object is moving ?

CASE STUDY: SHAPE FROM CONTROLLED LIGHTING

RGB Photometric Stereo

Take three shots in one: use an RGB Camera

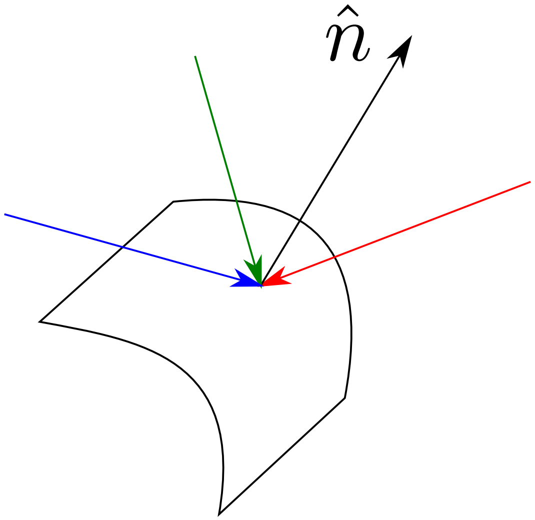


$$I = \rho < \hat{n}, \ell >$$

CASE STUDY: SHAPE FROM CONTROLLED LIGHTING

RGB Photometric Stereo

Take three shots in one: use an RGB Camera



ρ Albedo (Surface Color)

$$I_R = \rho_R < \hat{n}, \ell_R >$$

$$I_G = \rho_G < \hat{n}, \ell_G >$$

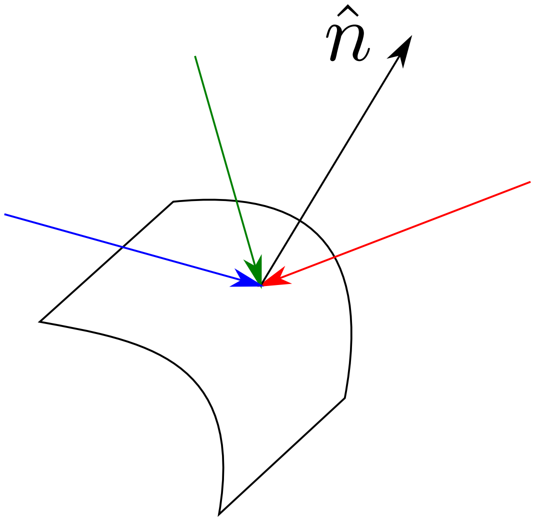
$$I_B = \rho_B < \hat{n}, \ell_B >$$

CASE STUDY: SHAPE FROM CONTROLLED LIGHTING

RGB Photometric Stereo

Take three shots in one: use an RGB Camera

But now we have extra unknowns for surface color



ρ Albedo (Surface Color)

$$I_R = \rho_R < \hat{n}, \ell_R >$$

$$I_G = \rho_G < \hat{n}, \ell_G >$$

$$I_B = \rho_B < \hat{n}, \ell_B >$$

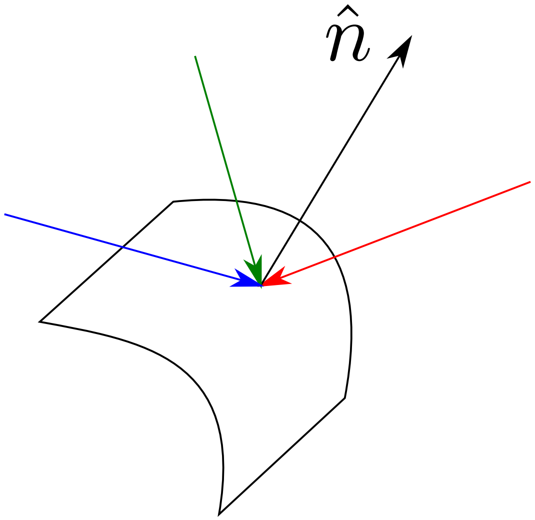
CASE STUDY: SHAPE FROM CONTROLLED LIGHTING

RGB Photometric Stereo

Take three shots in one: use an RGB Camera

But now we have extra unknowns for surface color

Solution 1: Measure albedo separately (assuming it's constant)



ρ Albedo (Surface Color)

$$I_R = \rho_R < \hat{n}, \ell_R >$$

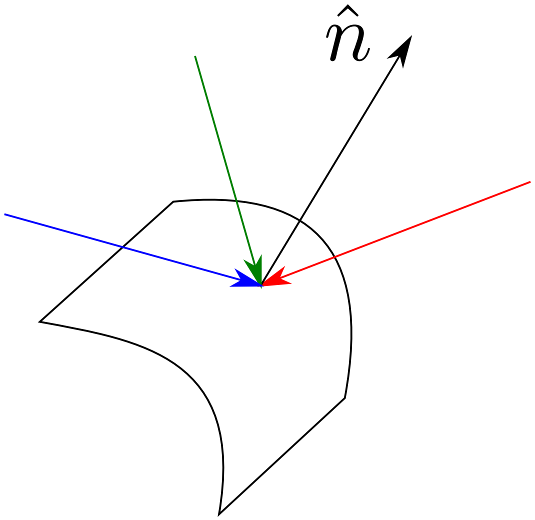
$$I_G = \rho_G < \hat{n}, \ell_G >$$

$$I_B = \rho_B < \hat{n}, \ell_B >$$

CASE STUDY: SHAPE FROM CONTROLLED LIGHTING

RGB Photometric Stereo

Take three shots in one: use an RGB Camera



ρ Albedo (Surface Color)

But now we have extra unknowns for surface color

Solution 1: Measure albedo separately (assuming it's constant)

Solution 2: Paint the object, so that albedo is known and constant

$$I_R = \rho_R < \hat{n}, \ell_R >$$

$$I_G = \rho_G < \hat{n}, \ell_G >$$

$$I_B = \rho_B < \hat{n}, \ell_B >$$

CASE STUDY: SHAPE FROM CONTROLLED LIGHTING

Single-image RGB Photometric Stereo With Spatially-varying Albedo

Ayan Chakrabarti
TTI-Chicago

Kalyan Sunkavalli
Adobe Research

Abstract

We present a single-shot system to recover surface geometry of objects with spatially-varying albedos, from images captured under a calibrated RGB photometric stereo setup—with three light directions multiplexed across different color channels in the observed RGB image. Since the problem is ill-posed point-wise, we assume that the albedo map can be modeled as piece-wise constant with a restricted number of distinct albedo values. We show that under ideal conditions, the shape of a non-degenerate local constant albedo surface patch can theoretically be recovered exactly. Moreover, we present a practical and efficient algorithm that uses this model to robustly recover shape from real images. Our method first reasons about shape locally in a dense set of patches in the observed image, producing shape distributions for every patch. These local distributions are then combined to produce a single consistent surface normal map. We demonstrate the efficacy of the approach through experiments on both synthetic renderings as well as real captured images.

from a single image of an object with unknown spatially-varying albedo under unknown natural lighting. Although impressive given the inherent ambiguities in the SFS setup, their recovered geometries are typically coarse due to the use of strong smoothness priors, and their inference algorithm is computationally expensive. This is true even when known lighting is provided as input to their algorithm, primarily because it is designed to handle arbitrary and potentially ambiguous *natural* illumination environments.

In this paper, we show that efficient and high-quality surface recovery from a single image is possible, when using a calibrated lighting environment that is specifically chosen to be directly informative about shape. Specifically, we use the RGB (or color) photometric stereo (RGB-PS) setup [3, 11, 14], where an object is illuminated by three monochromatic directional light sources, such that each of the red, green, and blue channels in the observed image is "lit" from a different direction. For natural lighting, directional diversity in color has been shown to be informative towards shape [10]. But the benefits of this lighting setup for shape recovery can be better understood by interpreting it as one that multiplexes the multiple images of classical PS into the different color channels of a single image.

CASE STUDY: SHAPE FROM CONTROLLED LIGHTING

Proposition 1. *Given noiseless observed intensities $v(p)$ at a set of locations $p \in \Omega$ on a diffuse surface patch known to have constant albedo, i.e., $\kappa(p) = \kappa_\Omega, \forall p \in \Omega$, the true surface normals $\{\hat{n}(p) : p \in \Omega\}$ and common albedo κ_Ω are uniquely determined, if:*

1. *All intensities $v(p)$ are strictly positive.*
2. *The true surface is non-degenerate in the sense that the set $\{\hat{n}(p)\hat{n}(p)^T : p \in \Omega\}$, of outer-products of the true normal vectors, span the space Sym_3 of all 3×3 symmetric matrices.*

Proof: Given κ_Ω and $\hat{n}(p)$ as the true patch albedo and normals, let κ'_Ω, $\hat{n}'(p)$ be a second solution pair that also explains the observed intensities $v(p)$ in the patch Ω. Since the observed intensities are strictly positive, this implies that the albedos $\kappa_\Omega, \kappa'_\Omega$ are strictly positive as well, and further that no point is in shadow under any of the lights, i.e. $L^T\hat{n}(p), L^T\hat{n}'(p) > 0$, $\forall p \in \Omega$. Then, since L^T is invertible, we can write

$$\begin{aligned} \text{diag}[\kappa_\Omega]L^T\hat{n}(p) &= \text{diag}[\kappa'_\Omega]L^T\hat{n}'(p) \\ \Rightarrow \hat{n}'(p) &= A\hat{n}(p), \quad \forall p \in \Omega, \end{aligned} \tag{2}$$

where we define the matrix $A = L^{-T}RL^T$, with $R = \text{diag}[\kappa'_\Omega]^{-1}\text{diag}[\kappa_\Omega]$ being a diagonal matrix whose entries

Analyze ambiguities, and show that for "most" local regions, if the albedo is constant inside the region, we can recover its shape and albedo uniquely.

CASE STUDY: SHAPE FROM CONTROLLED LIGHTING

Proposition 1. *Given noiseless observed intensities $v(p)$ at a set of locations $p \in \Omega$ on a diffuse surface patch known to have constant albedo, i.e., $\kappa(p) = \kappa_\Omega, \forall p \in \Omega$, the true surface normals $\{\hat{n}(p) : p \in \Omega\}$ and common albedo κ_Ω are uniquely determined, if:*

1. *All intensities $v(p)$ are strictly positive.*
2. *The true surface is non-degenerate in the sense that the set $\{\hat{n}(p)\hat{n}(p)^T : p \in \Omega\}$, of outer-products of the tru*
sy

Proof: (
mals, le
plains t
the obs
that the
further
i.e. L^T
invertib

where
diag$[\kappa'_\Omega$

Analyze ambiguities, and show that for "most" local regions, if the albedo is constant inside the region, we can recover its shape and albedo uniquely.

But we don't know which patches are constant albedo, and which have boundaries in them.

CASE STUDY: SHAPE FROM CONTROLLED LIGHTING

Proposition 1. *Given noiseless observed intensities $v(p)$ at a set of locations $p \in \Omega$ on a diffuse surface patch known to have constant albedo, i.e., $\kappa(p) = \kappa_\Omega, \forall p \in \Omega$, the true surface normals $\{\hat{n}(p) : p \in \Omega\}$ and common albedo κ_Ω are uniquely determined, if:*

1. *All intensities $v(p)$ are strictly positive.*
2. *The true surface is non-degenerate in the sense that the set $\{\hat{n}(p)\hat{n}(p)^T : p \in \Omega\}$, of outer-products of the* tru
   syı

Proof: (
mals, le
plains t
the obs
that the
further
i.e. L^T
invertib

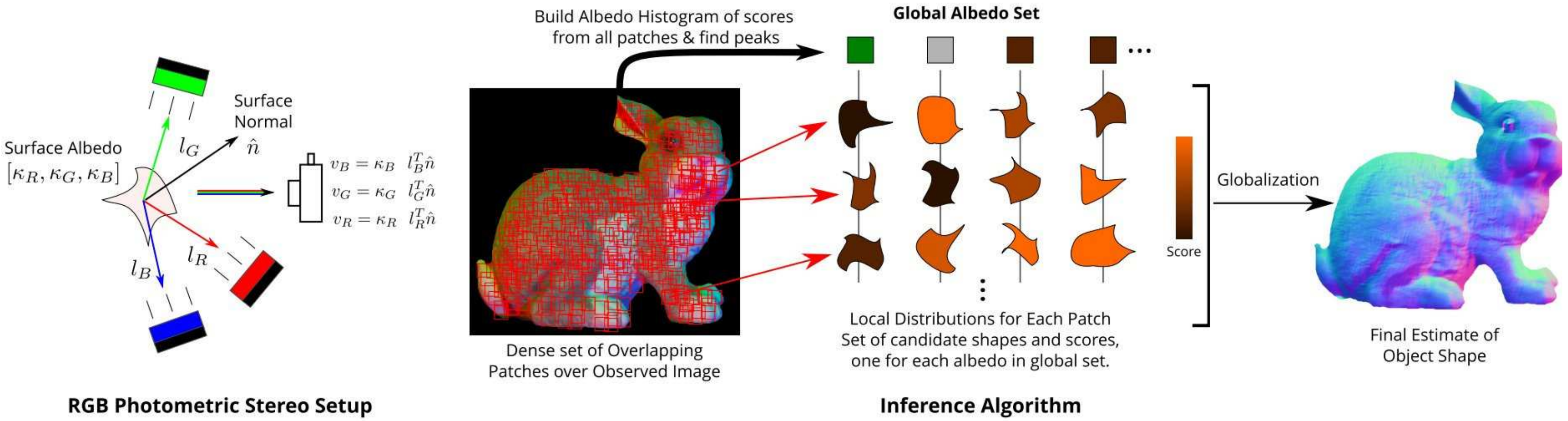
where
diag[κ'_Ω

Analyze ambiguities, and show that for "most" local regions, if the albedo is constant inside the region, we can recover its shape and albedo uniquely.

But we don't know which patches are constant albedo, and which have boundaries in them.

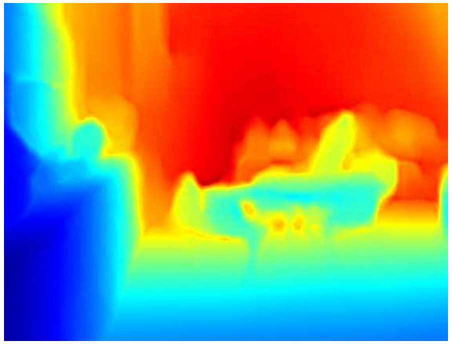
Also, uniqueness holds in idealized conditions. In reality, we'll have noise, 'non diffuse' reflection, ...

CASE STUDY: SHAPE FROM CONTROLLED LIGHTING



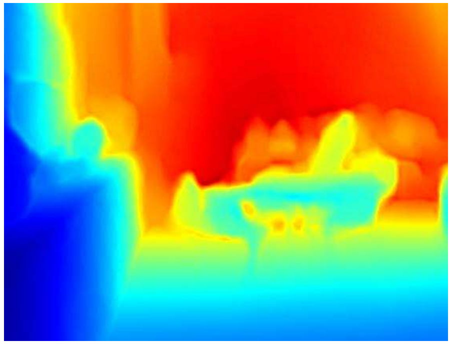
CASE STUDY: SHAPE FROM CONTROLLED LIGHTING

CASE STUDY: MONOCULAR DEPTH ESTIMATION



Depth from a Single Image

CASE STUDY: MONOCULAR DEPTH ESTIMATION

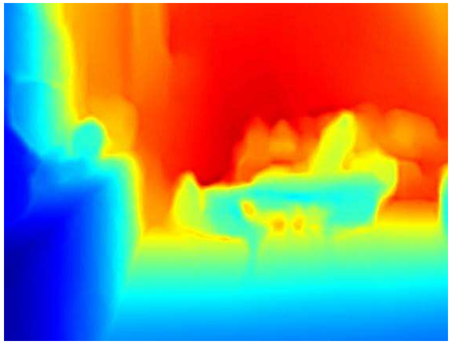


- No explicit geometric / optical cues.
- Must learn to map familiar patterns to depth.

Depth from a Single Image

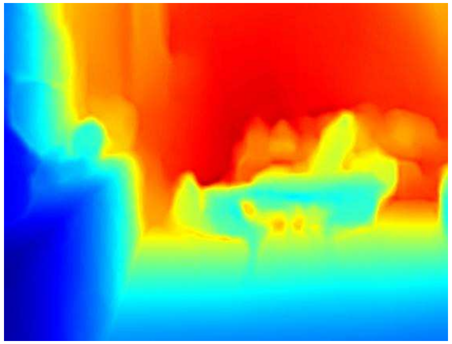
Shading.
Contours & Boundaries.
Foreshortening of regular patterns.
Scale of familiar objects.

CASE STUDY: MONOCULAR DEPTH ESTIMATION



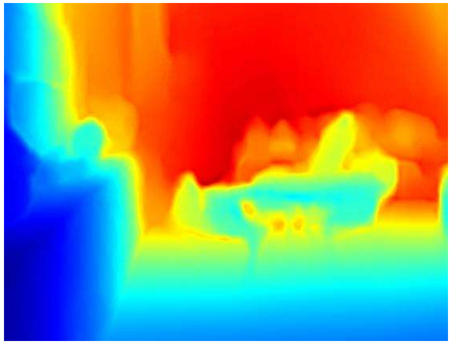
$Z(n)$

CASE STUDY: MONOCULAR DEPTH ESTIMATION



$Z(n)$ ← Lots of numbers (200k for a 500x400 image)

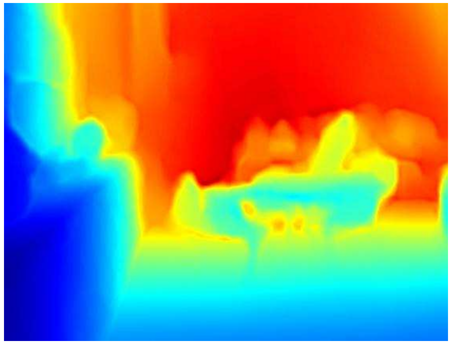
CASE STUDY: MONOCULAR DEPTH ESTIMATION



$$Z(n)$$

Estimate each Z(n) independently.

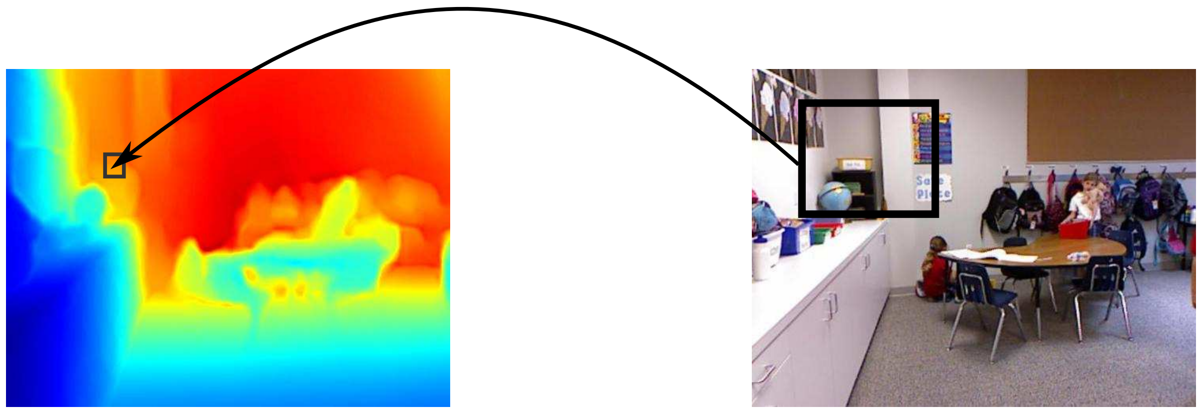
CASE STUDY: MONOCULAR DEPTH ESTIMATION



$$Z(n)$$

Estimate each Z(n) independently.

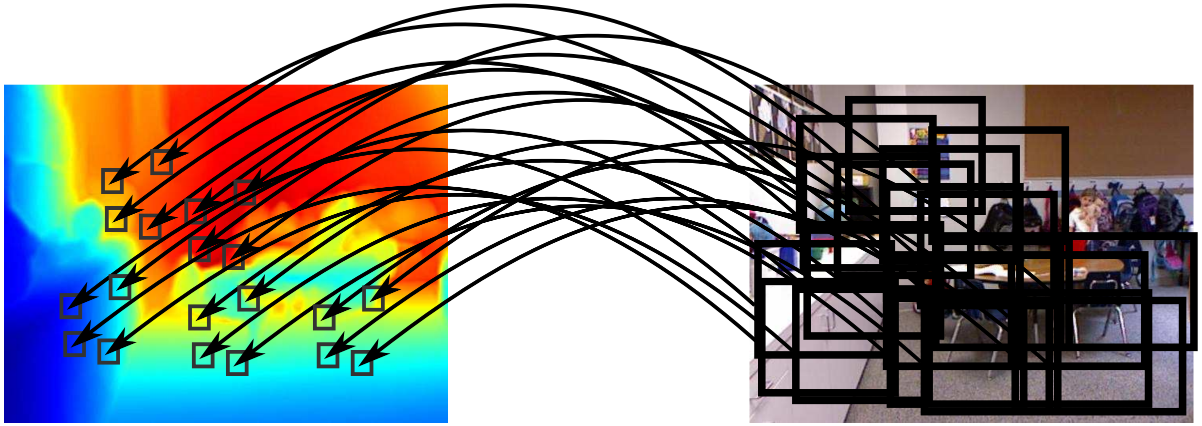
CASE STUDY: MONOCULAR DEPTH ESTIMATION



$$Z(n)$$

Estimate each Z(n) independently and *locally*.

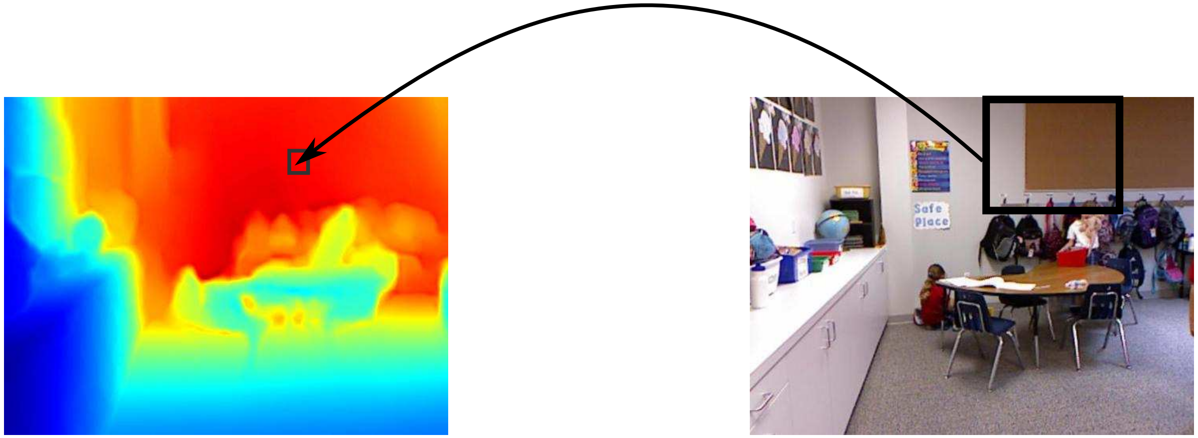
CASE STUDY: MONOCULAR DEPTH ESTIMATION



$$Z(n)$$

Estimate each Z(n) independently and *locally*.

CASE STUDY: MONOCULAR DEPTH ESTIMATION

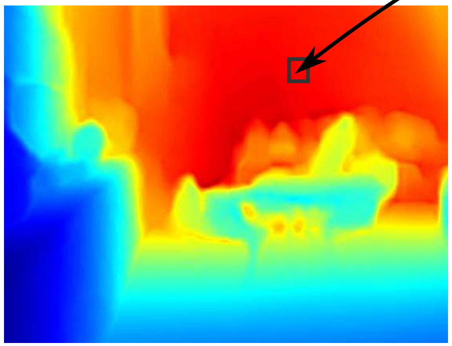


$$Z(n)$$

Problem: Local information may be ambiguous.

Estimate each Z(n) independently and *locally*.

CASE STUDY: MONOCULAR DEPTH ESTIMATION

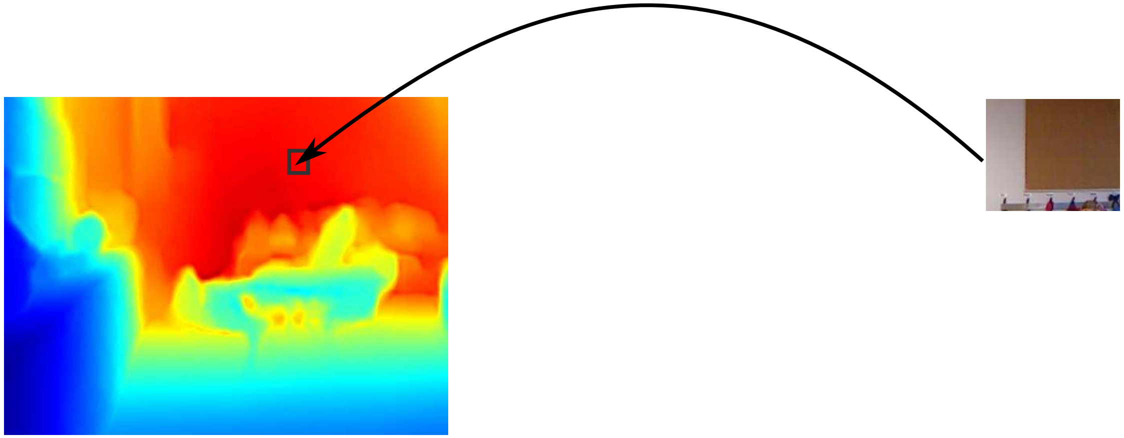


Problem: Local information may be ambiguous.

$$Z(n)$$

Estimate each Z(n) independently and *locally*.

CASE STUDY: MONOCULAR DEPTH ESTIMATION



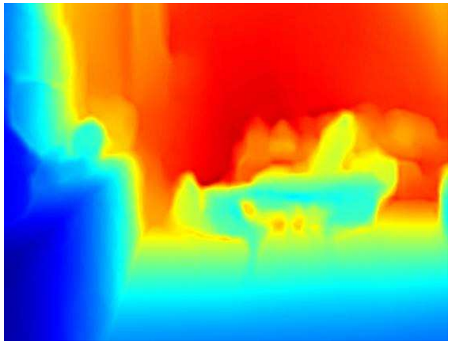
$Z(n)$

Problem: Local information may be ambiguous.

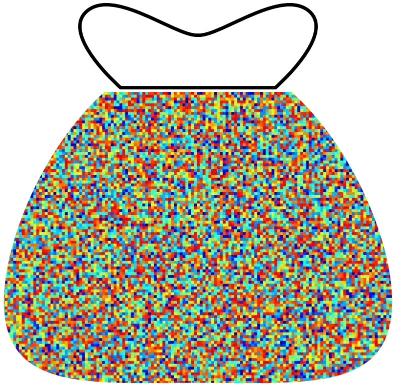
Estimate each Z(n) independently and *locally*.

Scene Maps have Structure

CASE STUDY: MONOCULAR DEPTH ESTIMATION



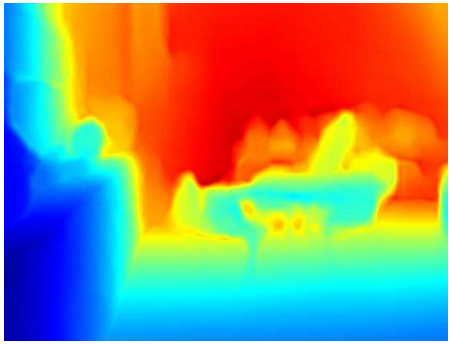
$\neq$



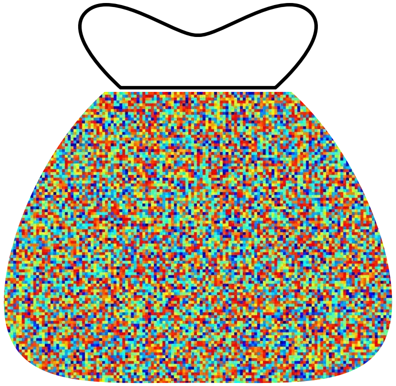
$Z(n)$

Scene Maps have Structure

CASE STUDY: MONOCULAR DEPTH ESTIMATION



$\neq$



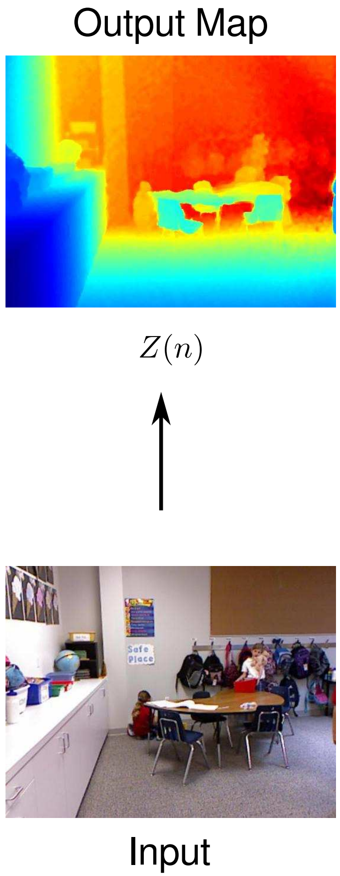
$$Z(n)$$

Build an algorithm that effectively
extracts and exploits this structure

Scene Maps have Structure

CASE STUDY: MONOCULAR DEPTH ESTIMATION

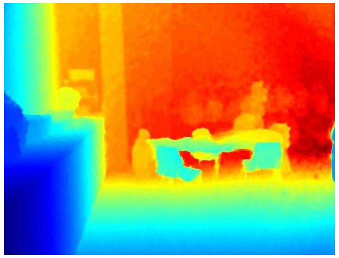
Local Estimation: Mid-level Representation



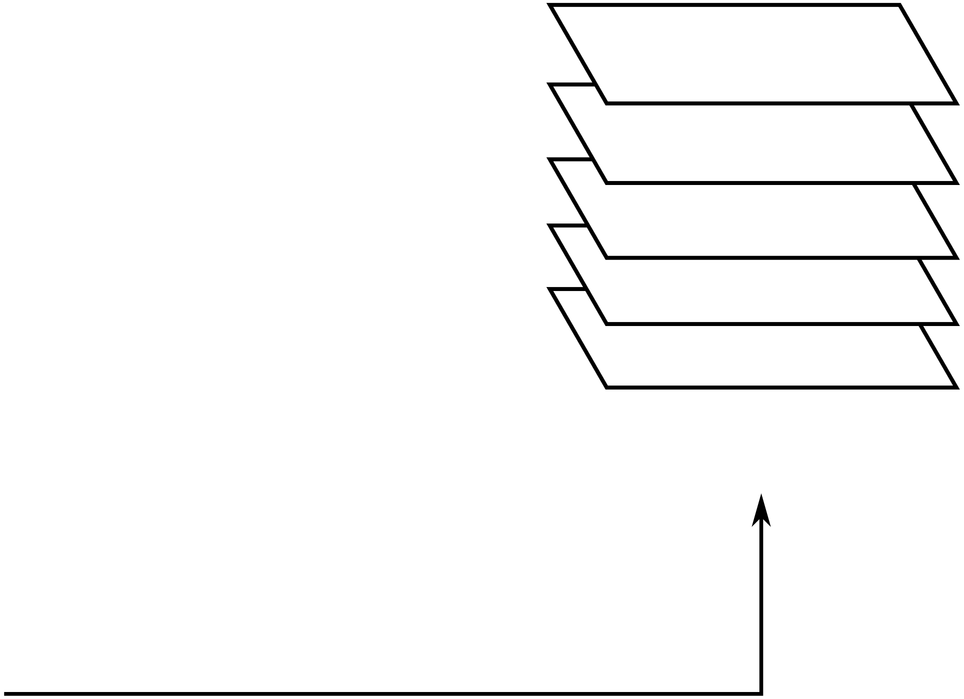
CASE STUDY: MONOCULAR DEPTH ESTIMATION

Local Estimation: Mid-level Representation

Output Map

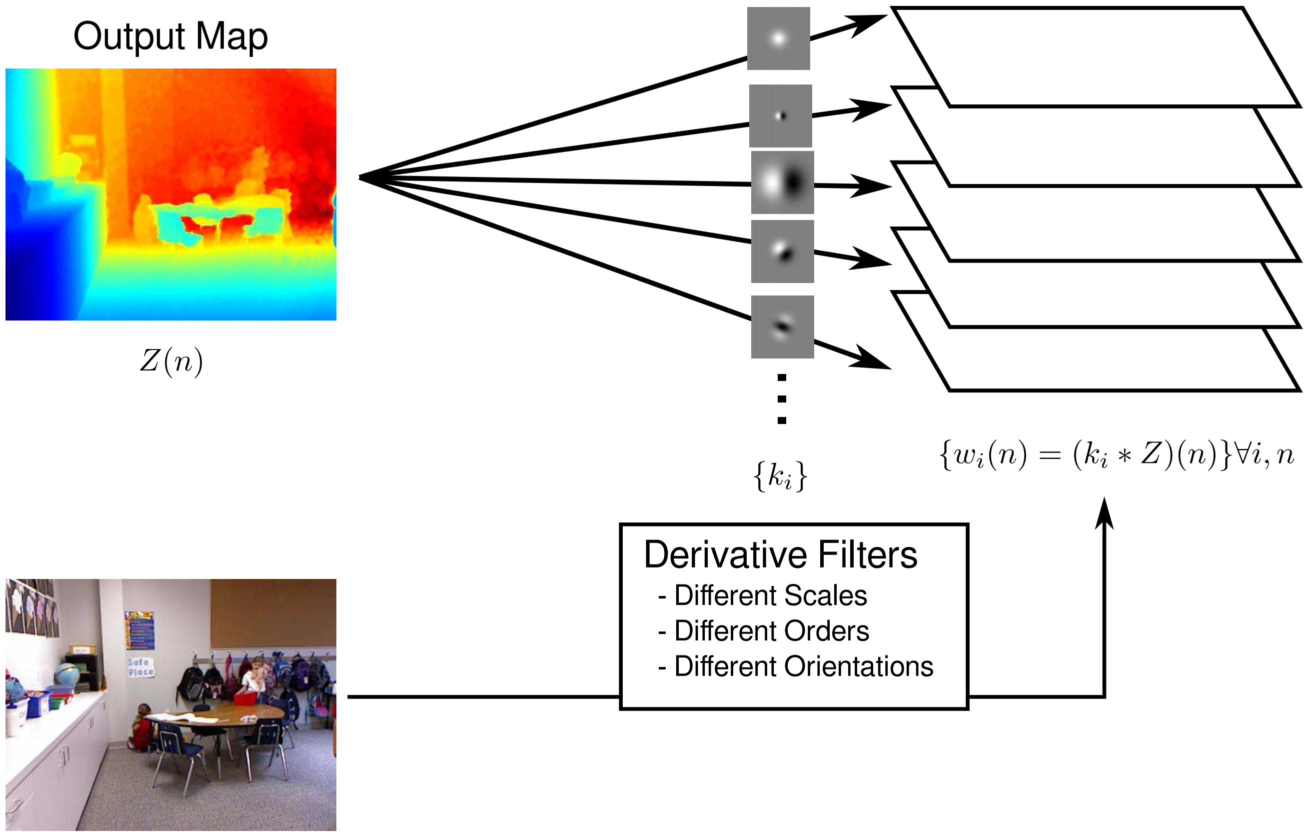


$Z(n)$



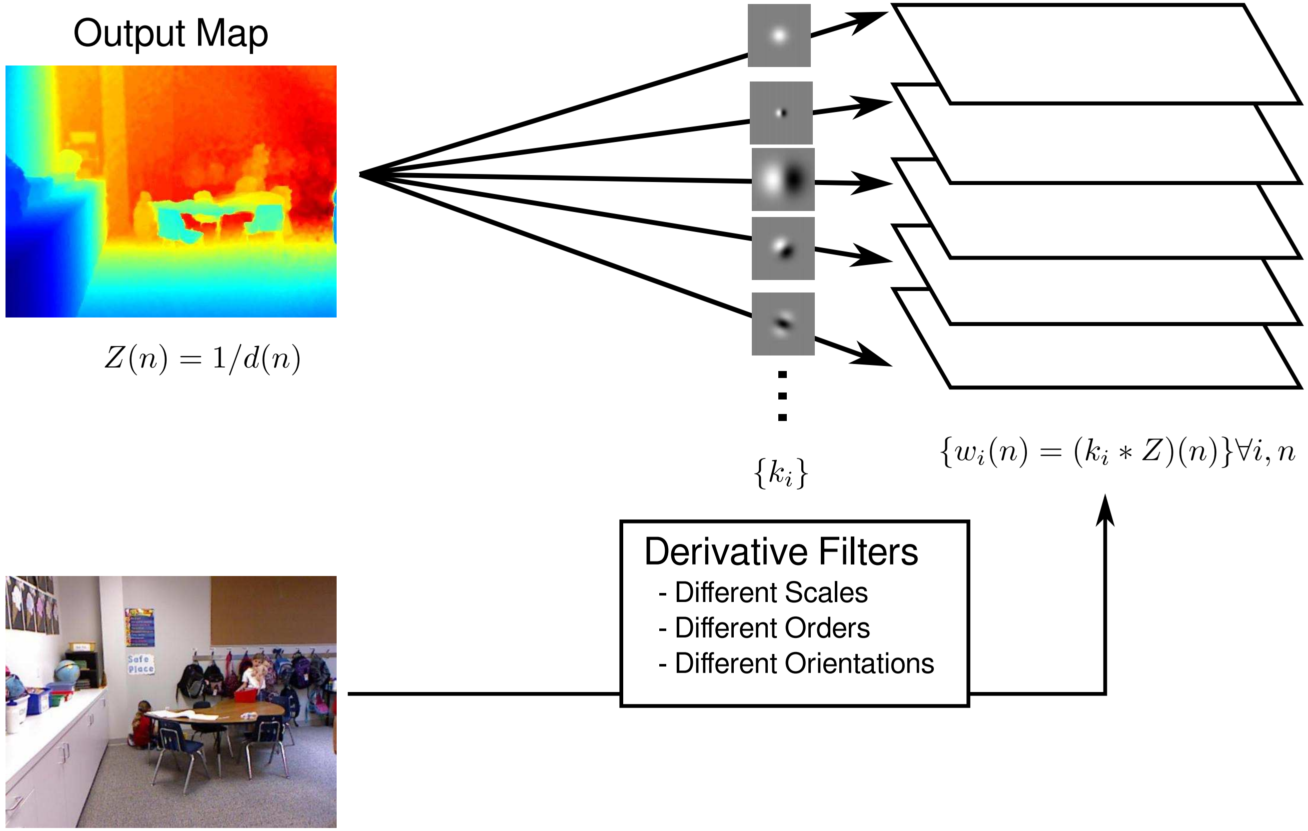
CASE STUDY: MONOCULAR DEPTH ESTIMATION

Local Estimation: Mid-level Representation



CASE STUDY: MONOCULAR DEPTH ESTIMATION

Local Estimation: Mid-level Representation



CASE STUDY: MONOCULAR DEPTH ESTIMATION

Local Estimation: Mid-level Representation

Perspective Camera

$$d(x, y) \Rightarrow (dx, dy, d)$$

World 3D Co-ordinates

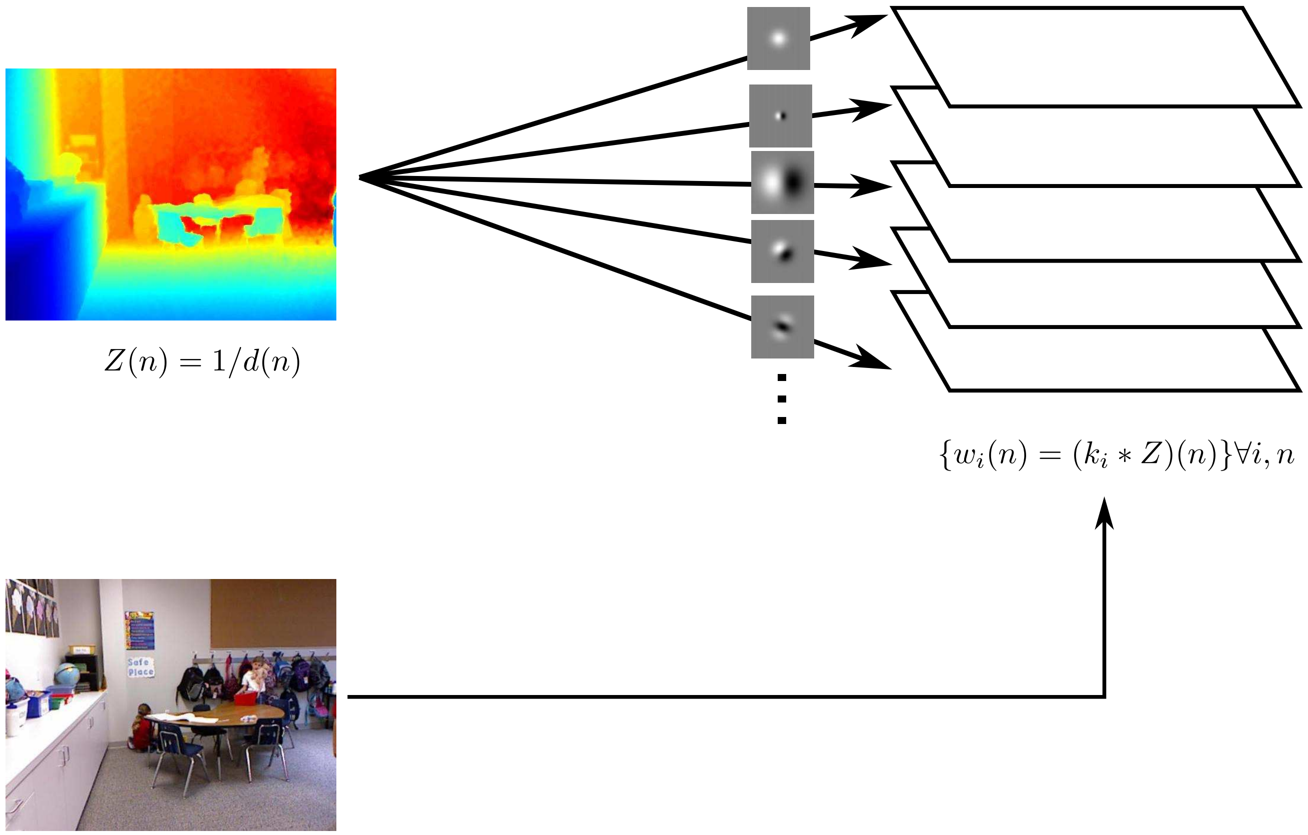
Plane Equation

$$\frac{1}{d} = \alpha x + \beta y + \gamma$$

Zeroth Derivative	=	Absolute depth
First Derivative	=	Surface orientation
Second Derivative	=	0: Planar
		Curvature, contours.

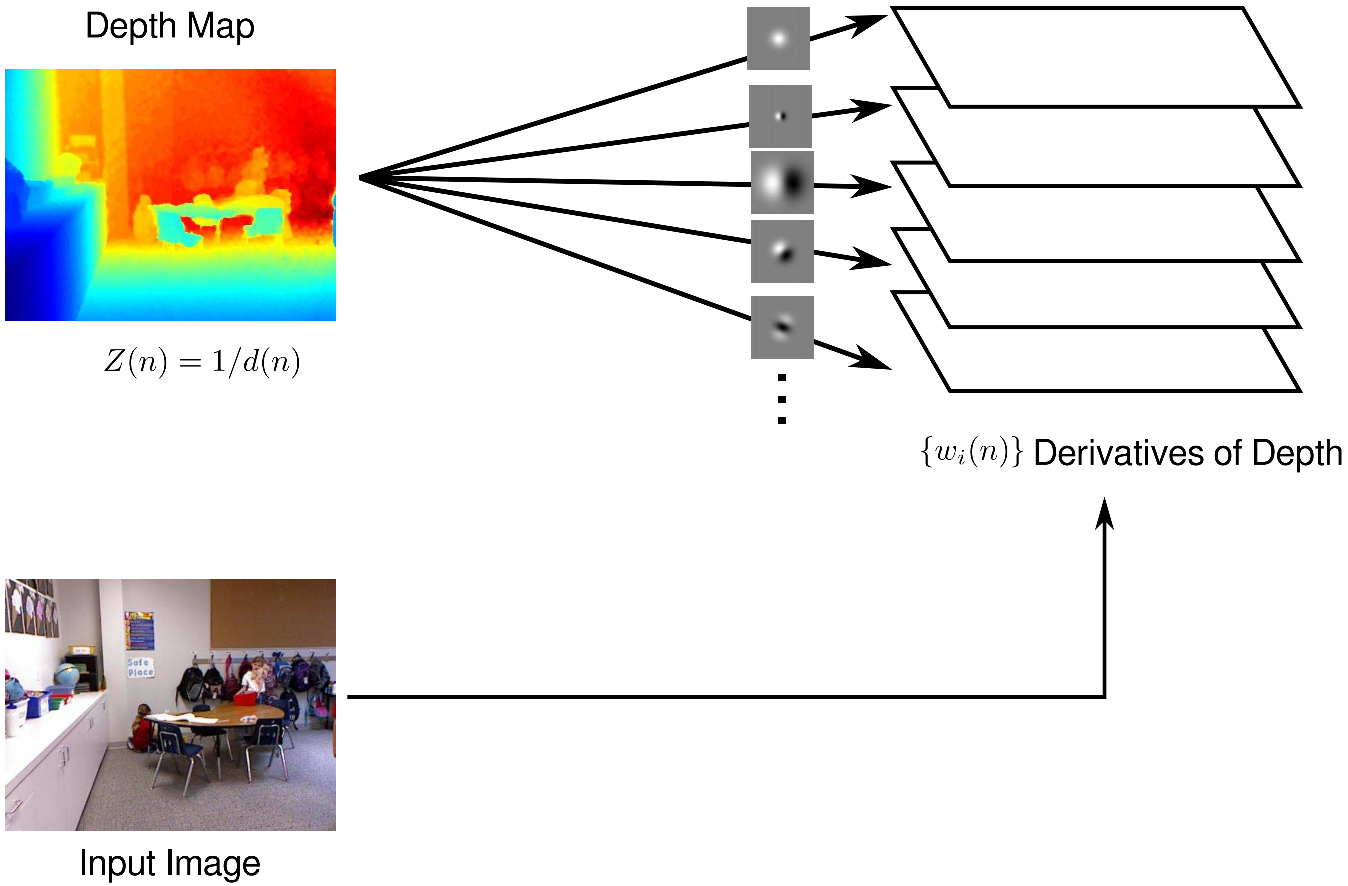
CASE STUDY: MONOCULAR DEPTH ESTIMATION

Local Estimation: Mid-level Representation



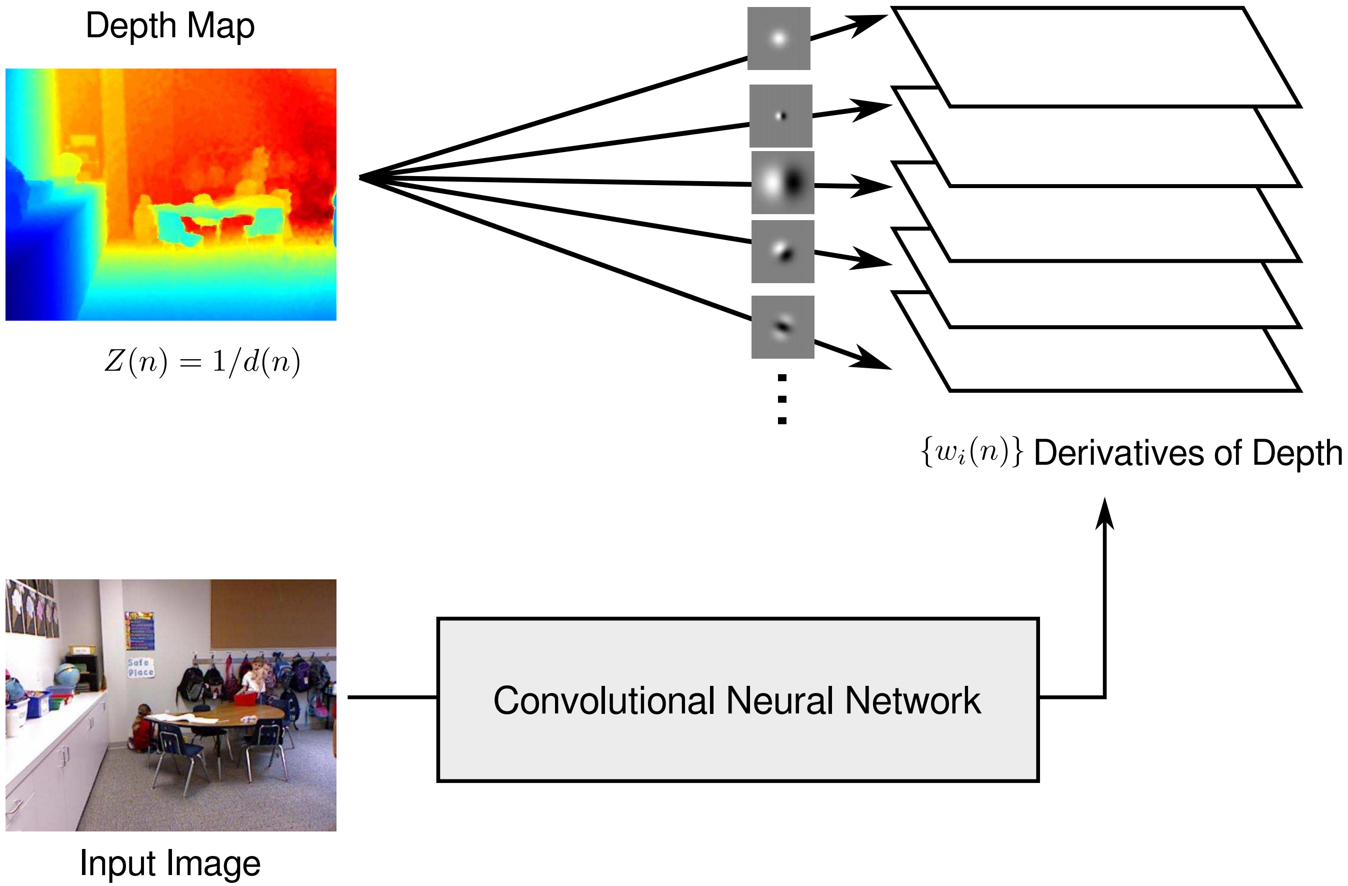
CASE STUDY: MONOCULAR DEPTH ESTIMATION

Local Estimation: Mid-level Representation



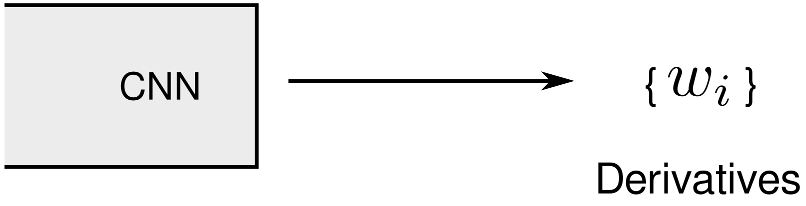
CASE STUDY: MONOCULAR DEPTH ESTIMATION

Local Estimation: Mid-level Representation



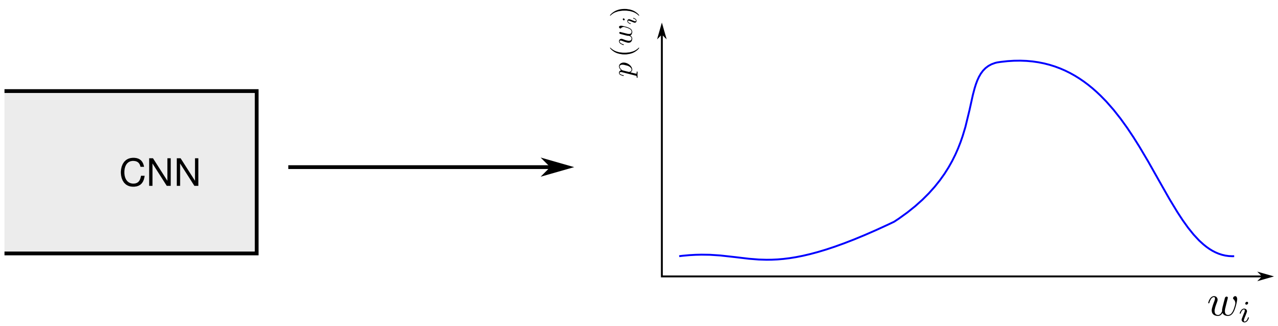
CASE STUDY: MONOCULAR DEPTH ESTIMATION

Local Estimation: Mid-level Representation



CASE STUDY: MONOCULAR DEPTH ESTIMATION

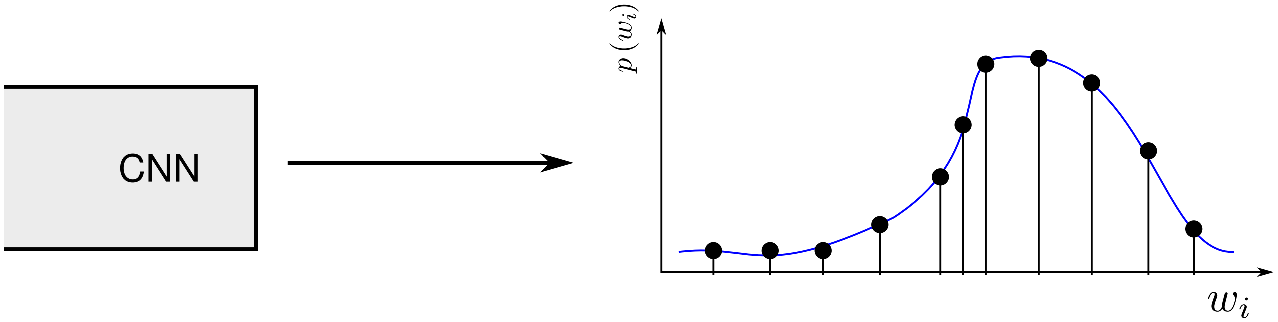
Local Estimation: Mid-level Representation



Distribution over Each Derivative

CASE STUDY: MONOCULAR DEPTH ESTIMATION

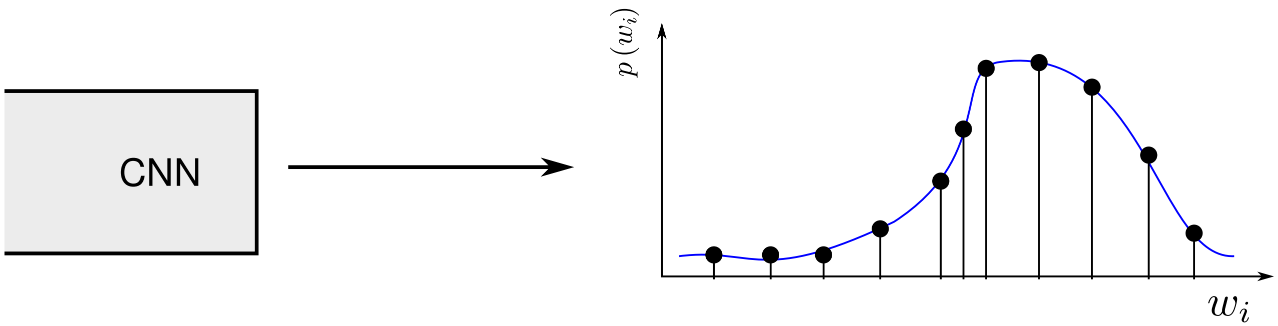
Local Estimation: Mid-level Representation



Distribution over Each Derivative

CASE STUDY: MONOCULAR DEPTH ESTIMATION

Local Estimation: Mid-level Representation



Distribution over Each Derivative

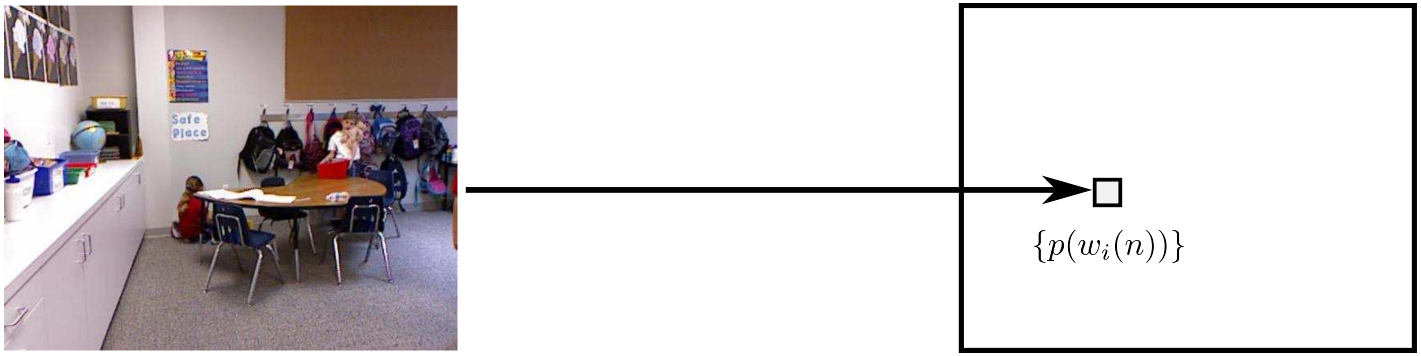
$$p(w_i(n)) = \sum_{j=1}^{M} \hat{p}_i^j(n) \frac{1}{\sqrt{2\pi}\sigma_i} \exp\left(-\frac{|w_i(n) - c_i^j|^2}{2\sigma_i^2}\right)$$

Convnet learns to predict mixing probabilities

Fixed prior to network training, using k-means on GT depth maps.

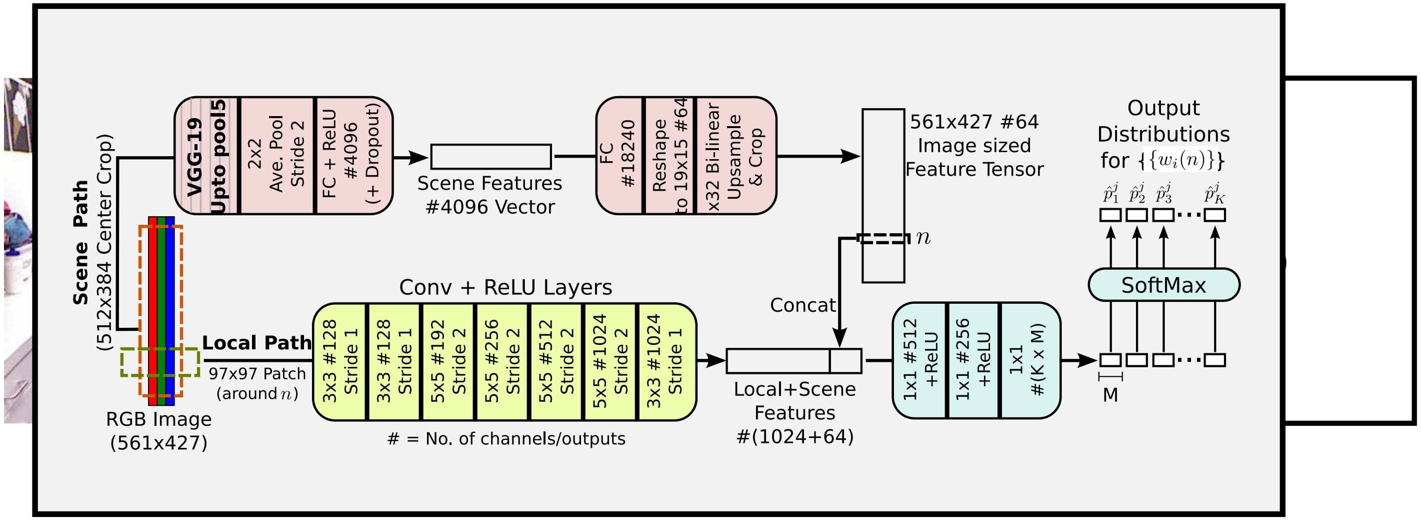
CASE STUDY: MONOCULAR DEPTH ESTIMATION

CNN



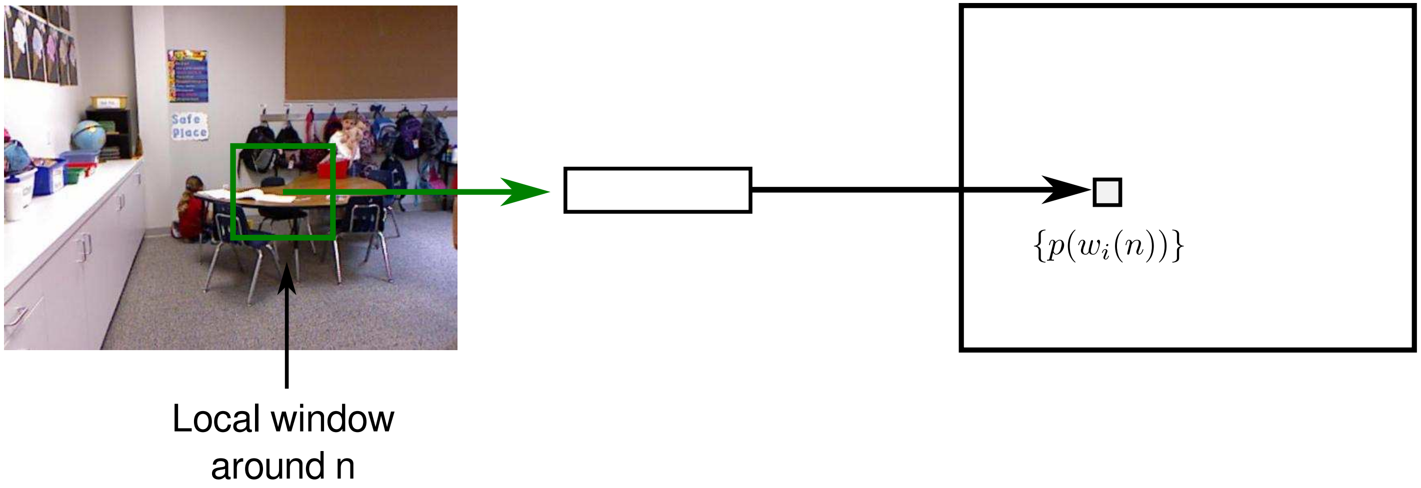
CASE STUDY: MONOCULAR DEPTH ESTIMATION

CNN



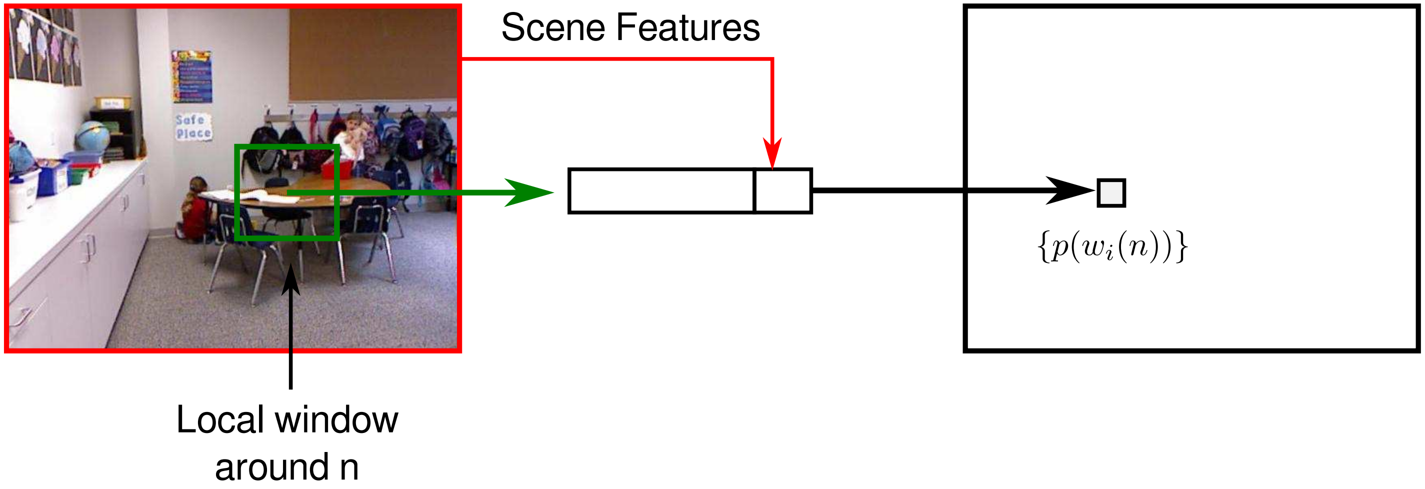
CASE STUDY: MONOCULAR DEPTH ESTIMATION

CNN



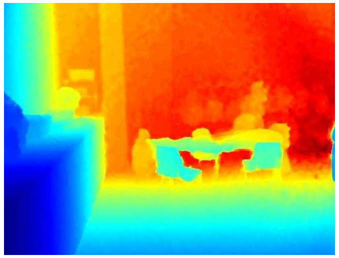
CASE STUDY: MONOCULAR DEPTH ESTIMATION

CNN



CASE STUDY: MONOCULAR DEPTH ESTIMATION

Training

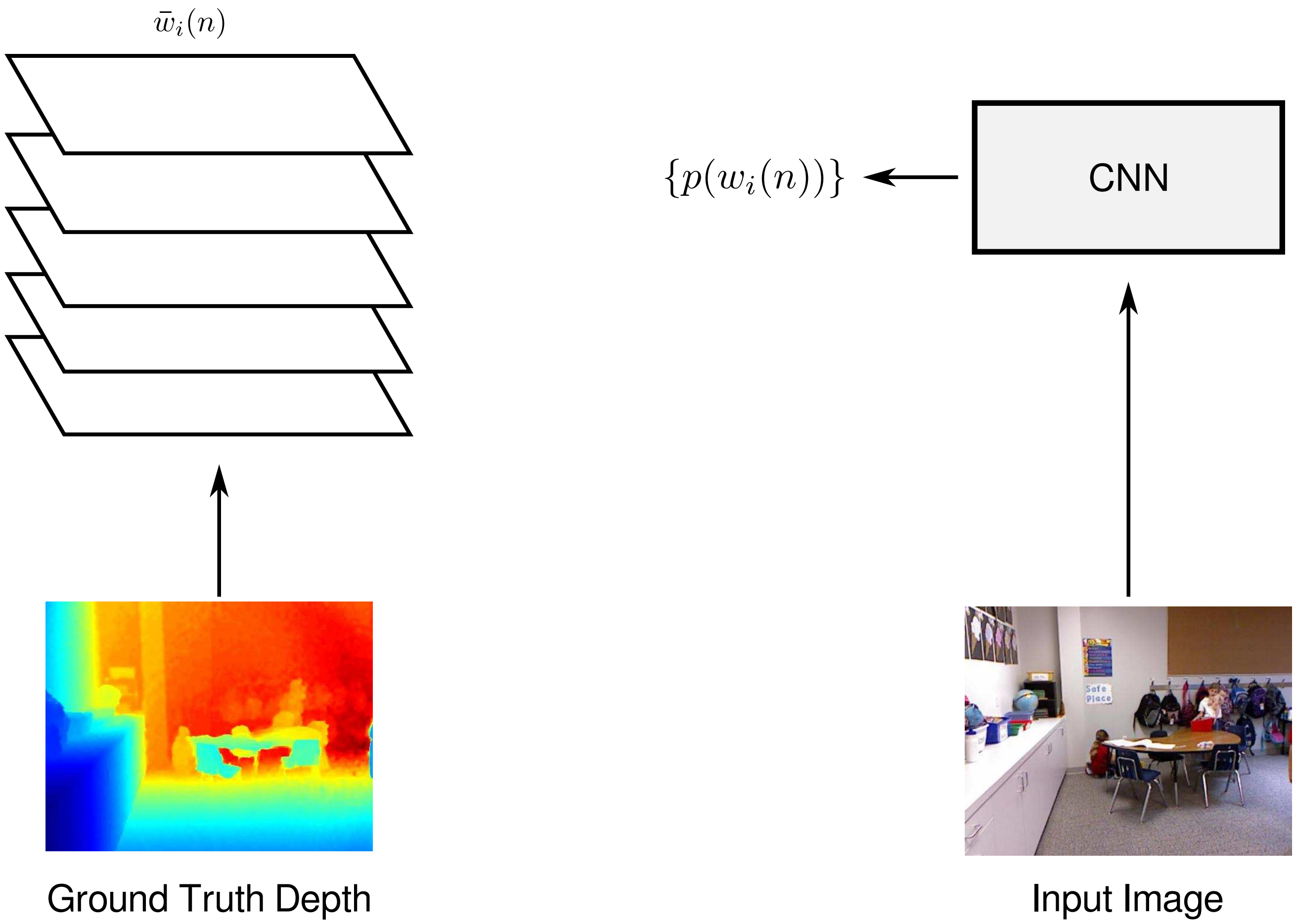


Ground Truth Depth

Input Image

CASE STUDY: MONOCULAR DEPTH ESTIMATION

Training



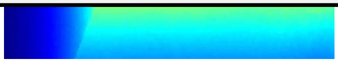
CASE STUDY: MONOCULAR DEPTH ESTIMATION

Training

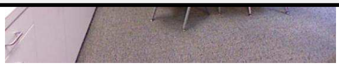
$$\hat{q}_i^j(n) \propto \exp\left(-\frac{|\bar{w}_i(n) - c_i^j|^2}{2\sigma_i^2}\right)$$

$$L = -\sum_{i,n}\sum_{j=1}^{M} \hat{q}_i^j(n)\left(\log \hat{p}_i^j(n) - \log \hat{q}_i^j(n)\right)$$

KL-Divergence



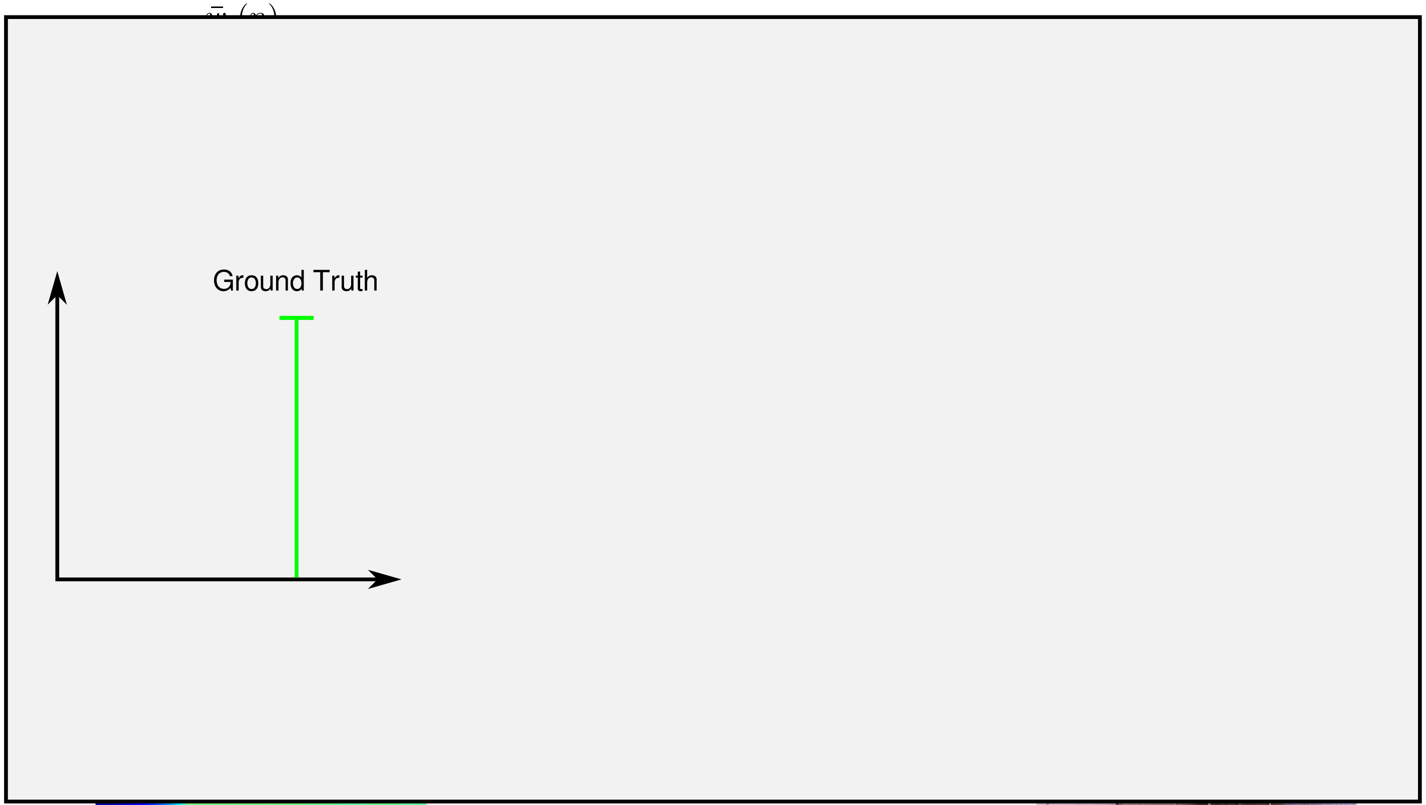
Ground Truth Depth

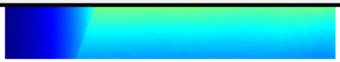


Input Image

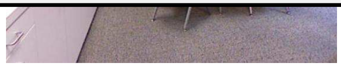
CASE STUDY: MONOCULAR DEPTH ESTIMATION

Training





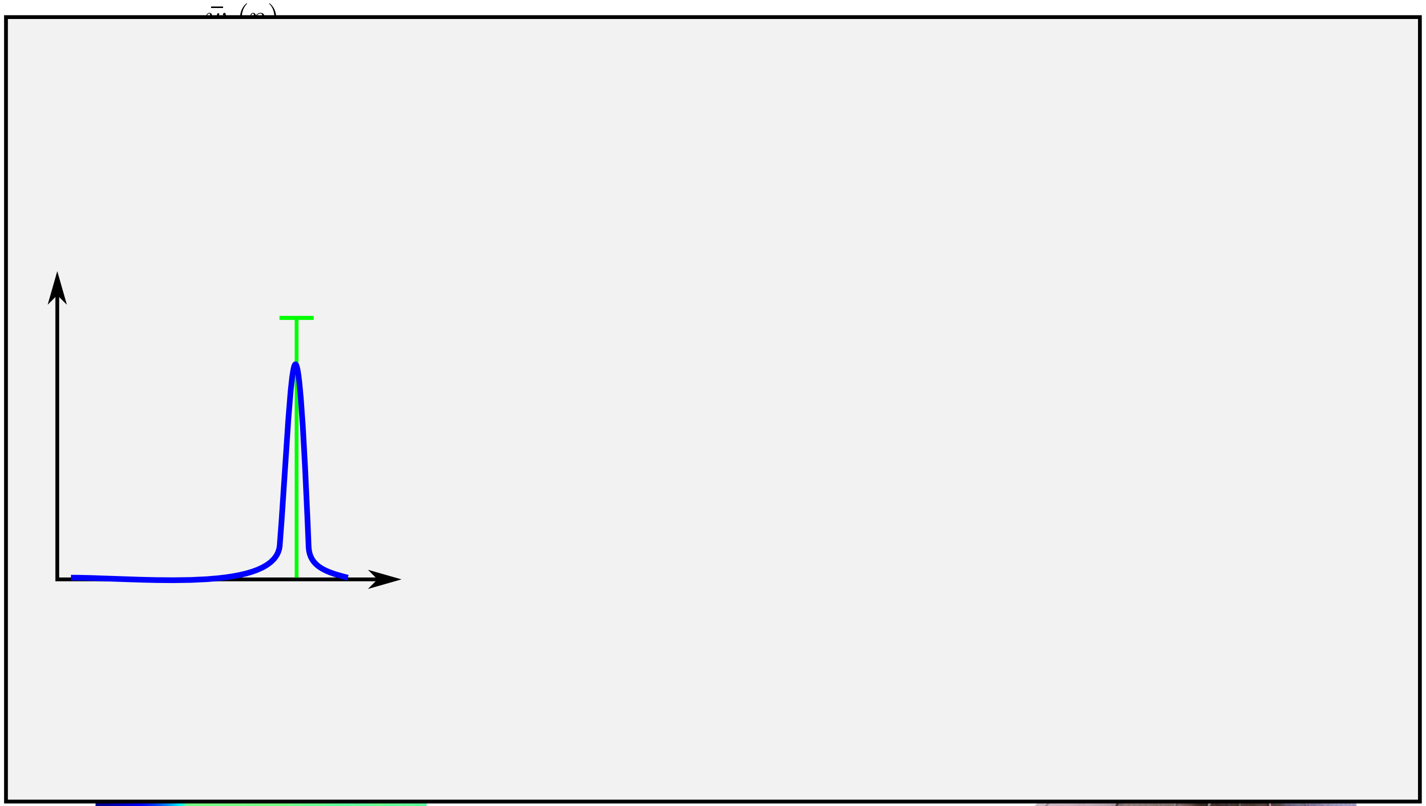
Ground Truth Depth

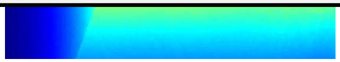


Input Image

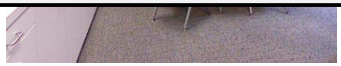
CASE STUDY: MONOCULAR DEPTH ESTIMATION

Training





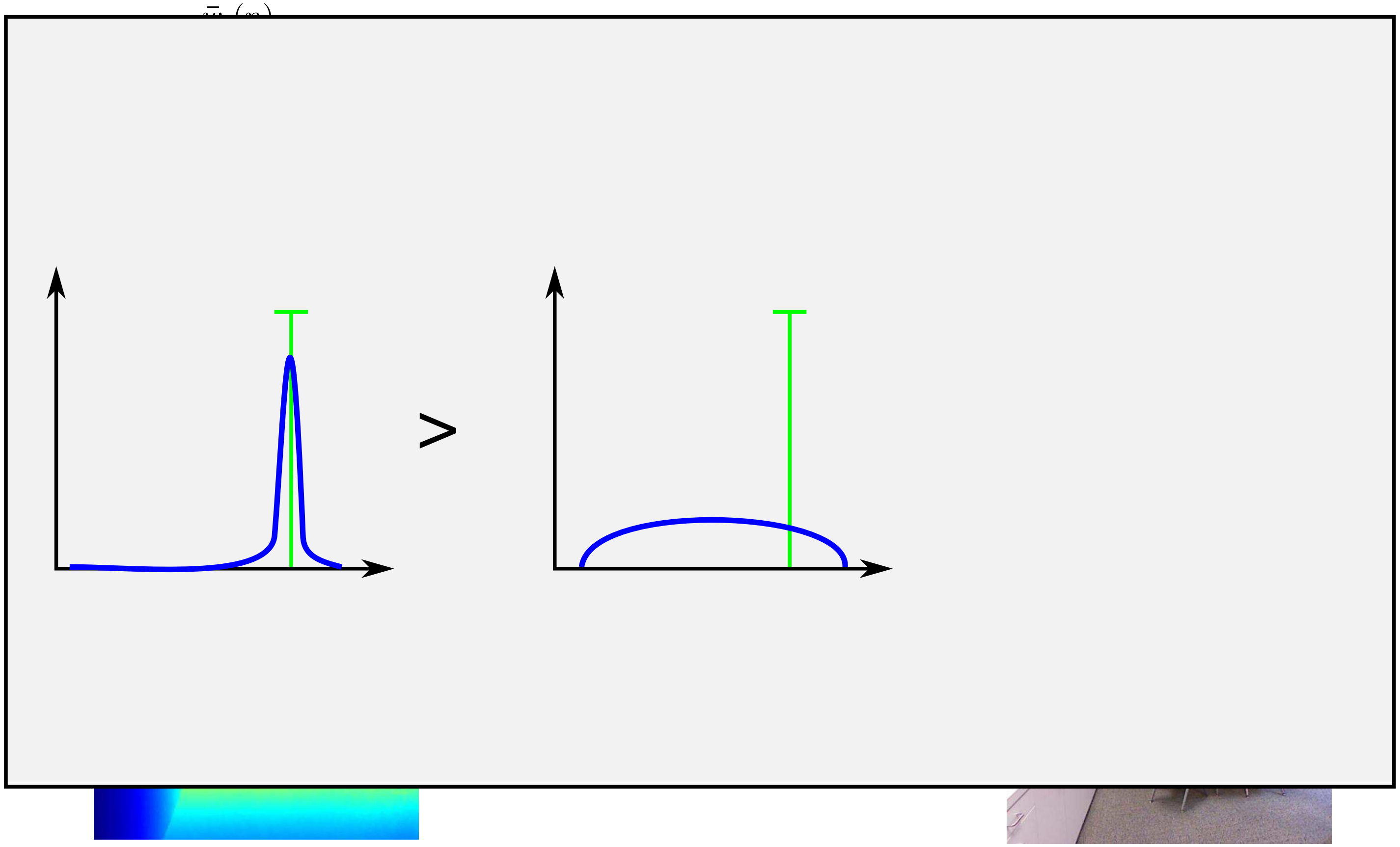
Ground Truth Depth



Input Image

CASE STUDY: MONOCULAR DEPTH ESTIMATION

Training

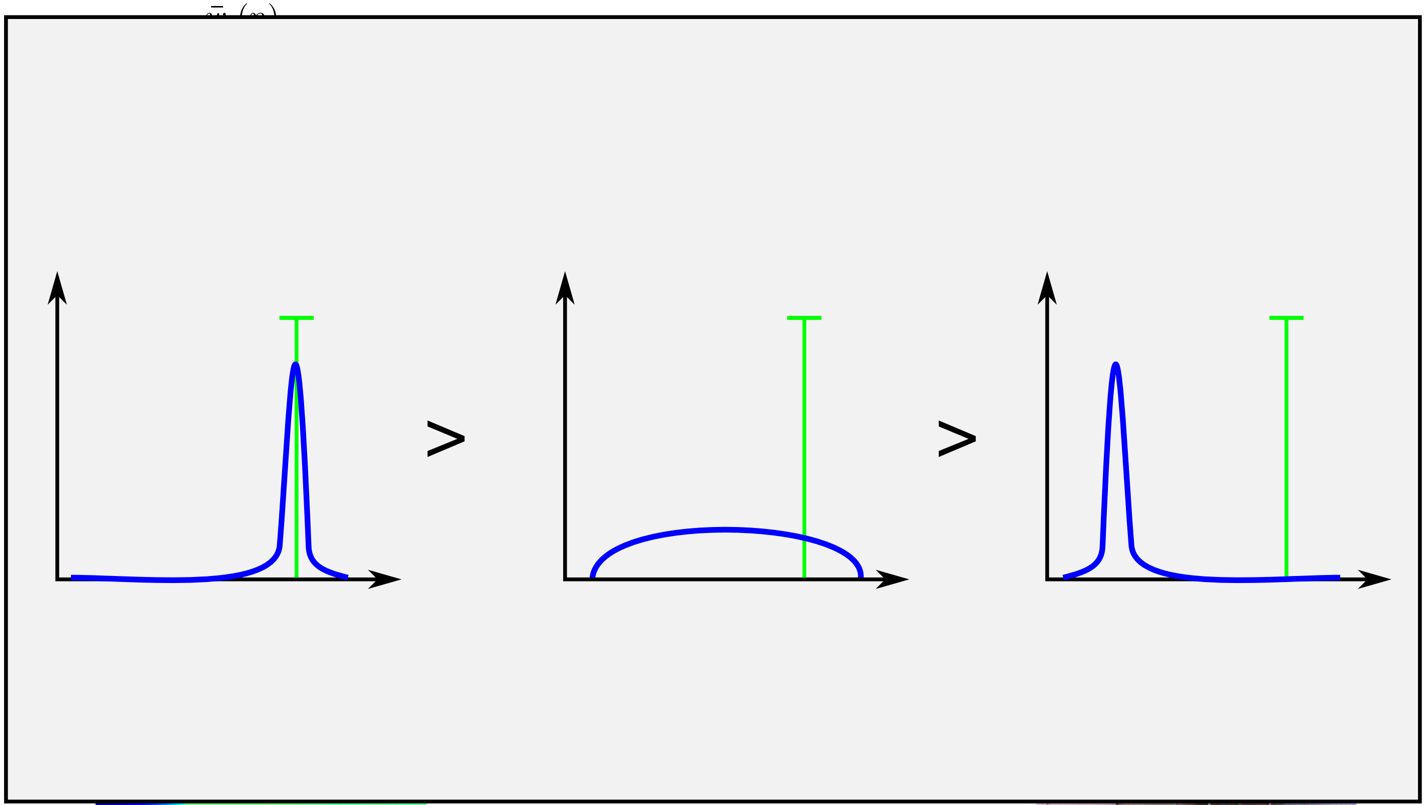


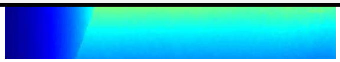
Ground Truth Depth

Input Image

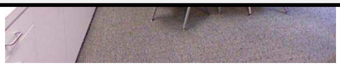
CASE STUDY: MONOCULAR DEPTH ESTIMATION

Training





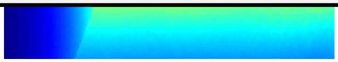
Ground Truth Depth



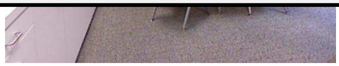
Input Image

CASE STUDY: MONOCULAR DEPTH ESTIMATION

Training



Ground Truth Depth



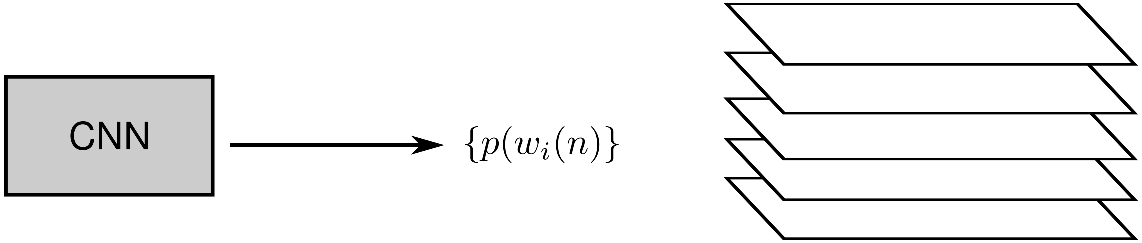
Input Image

CASE STUDY: MONOCULAR DEPTH ESTIMATION

Globalization

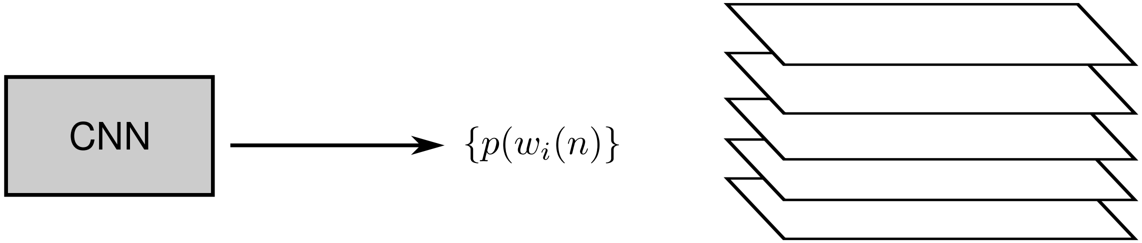
CASE STUDY: MONOCULAR DEPTH ESTIMATION

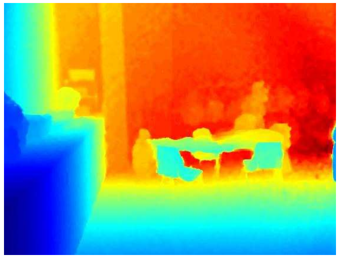
Globalization



CASE STUDY: MONOCULAR DEPTH ESTIMATION

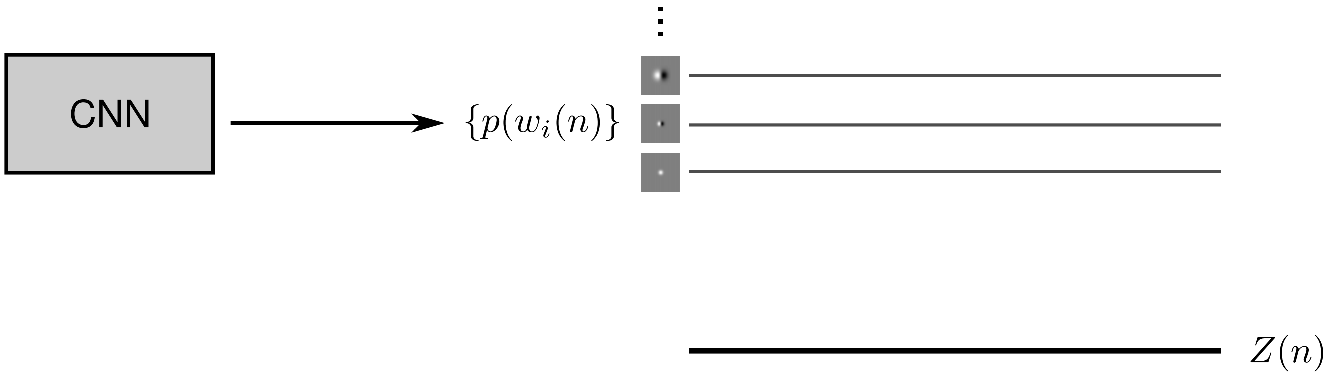
Globalization





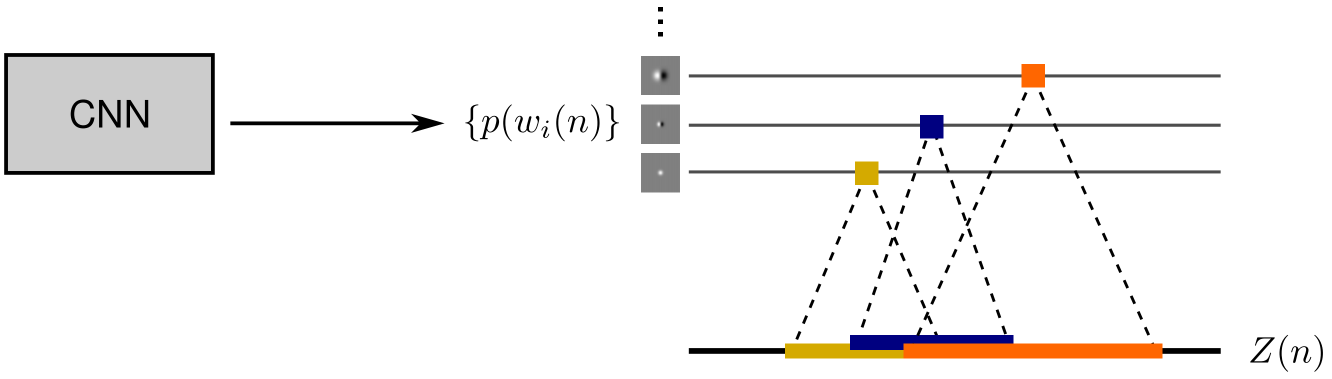
CASE STUDY: MONOCULAR DEPTH ESTIMATION

Globalization



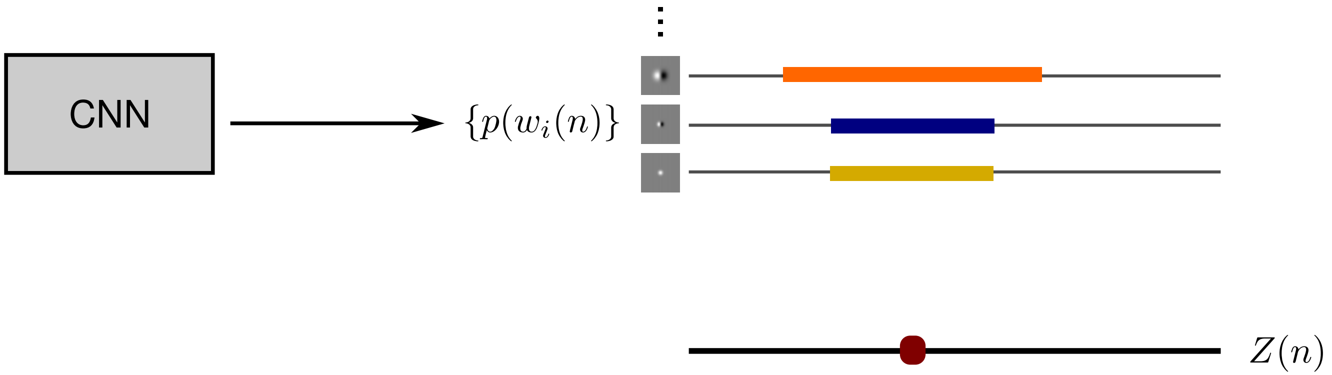
CASE STUDY: MONOCULAR DEPTH ESTIMATION

Globalization



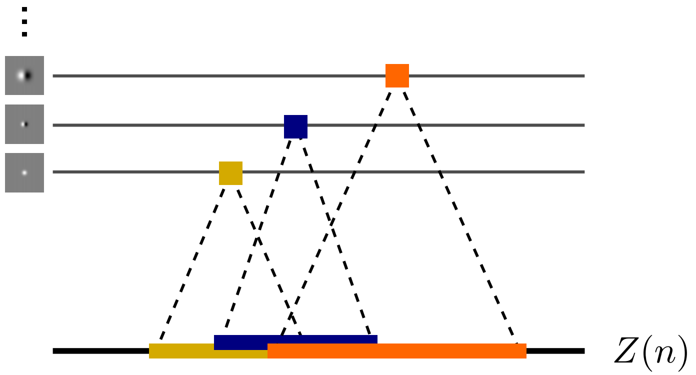
CASE STUDY: MONOCULAR DEPTH ESTIMATION

Globalization



CASE STUDY: MONOCULAR DEPTH ESTIMATION

Globalization

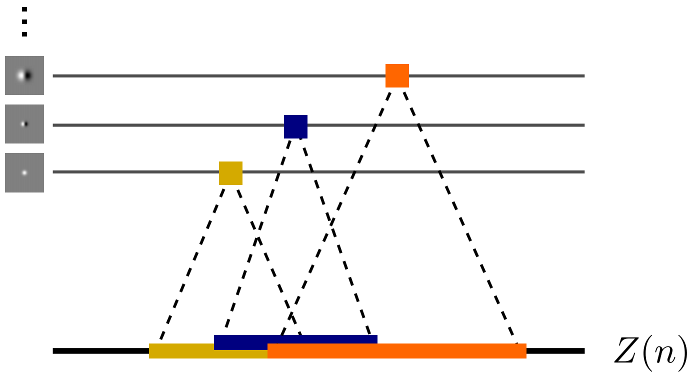


$$Z = \arg\max_{Z} \sum_{i,n} \log \underline{p_{i,n}} ((Z * k_i)(n))$$

From
CNN

CASE STUDY: MONOCULAR DEPTH ESTIMATION

Globalization



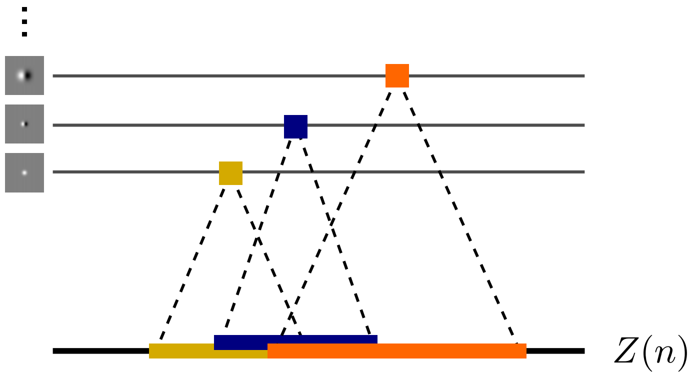
$$Z = \arg\max_{Z} \sum_{i,n} \log p_{i,n}\left(\left(Z * k_i\right)(n)\right)$$

$$Z = \arg\min_{Z} \min_{\{w_i(n)\}} - \left[\sum_{i,n} \log p_{i,n}\left(w_i(n)\right)\right]$$

Auxiliary Vars
for Derivatives

CASE STUDY: MONOCULAR DEPTH ESTIMATION

Globalization



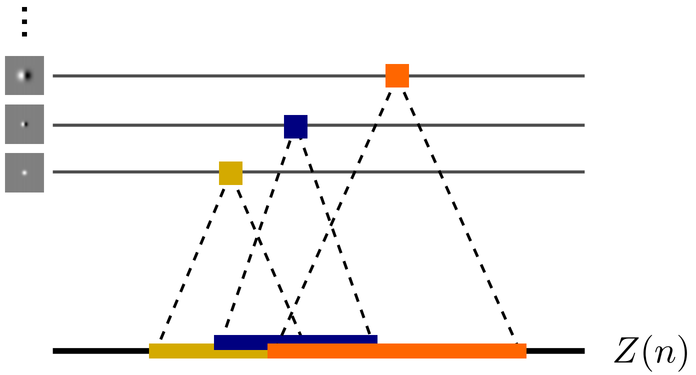
$$Z = \arg\max_{Z} \sum_{i,n} \log p_{i,n}\left((Z * k_i)(n)\right)$$

$$Z = \arg\min_{Z} \min_{\{w_i(n)\}} -\left[\sum_{i,n} \log p_{i,n}\left(w_i(n)\right)\right] + \frac{\beta}{2}\left[\sum_{i,n} \left|w_i(n) - (Z * k_i)(n)\right|^2\right]$$

Auxiliary Vars
for Derivatives

CASE STUDY: MONOCULAR DEPTH ESTIMATION

Globalization



$$Z = \arg\max_{Z} \sum_{i,n} \log p_{i,n}\left((Z * k_i)(n)\right)$$

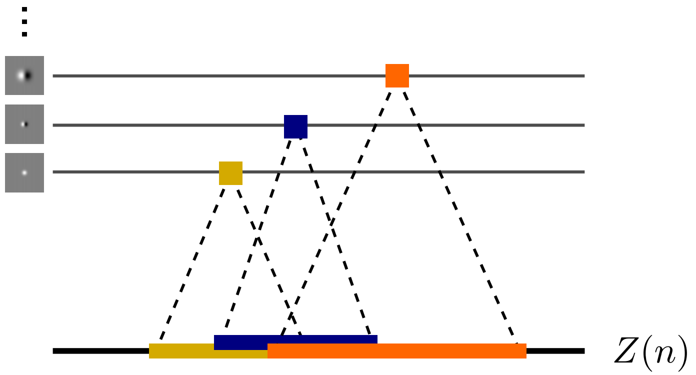
$$Z = \arg\min_{Z} \min_{\{w_i(n)\}} - \left[\sum_{i,n} \log p_{i,n}\left(w_i(n)\right)\right] + \frac{\beta}{2}\left[\sum_{i,n} \left|w_i(n) - (Z * k_i)(n)\right|^2\right]$$

Auxiliary Vars for Derivatives

Equivalent as $\beta \to \infty$

CASE STUDY: MONOCULAR DEPTH ESTIMATION

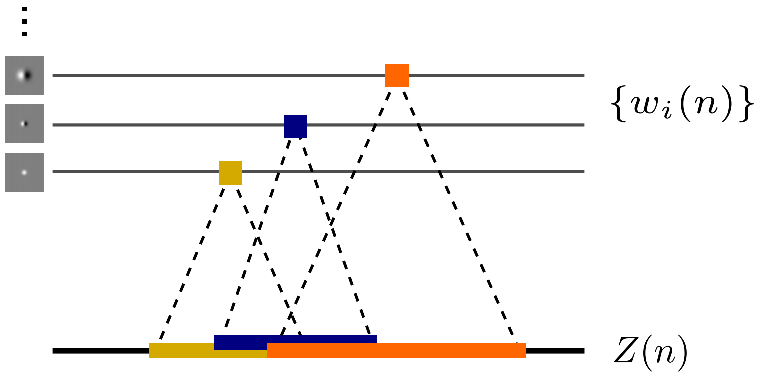
Globalization



$$Z = \arg\min_{Z} \min_{\{w_i(n)\}} -\left[\sum_{i,n} \log p_{i,n}\left(w_i(n)\right)\right] + \frac{\beta}{2}\left[\sum_{i,n} \left|w_i(n) - (Z * k_i)(n)\right|^2\right]$$

CASE STUDY: MONOCULAR DEPTH ESTIMATION

Globalization



$$Z = \arg\min_{Z} \min_{\{w_i(n)\}} - \left[\sum_{i,n} \log p_{i,n}\left(w_i(n)\right) \right] + \frac{\beta}{2} \left[\sum_{i,n} \left| w_i(n) - (Z * k_i)(n) \right|^2 \right]$$

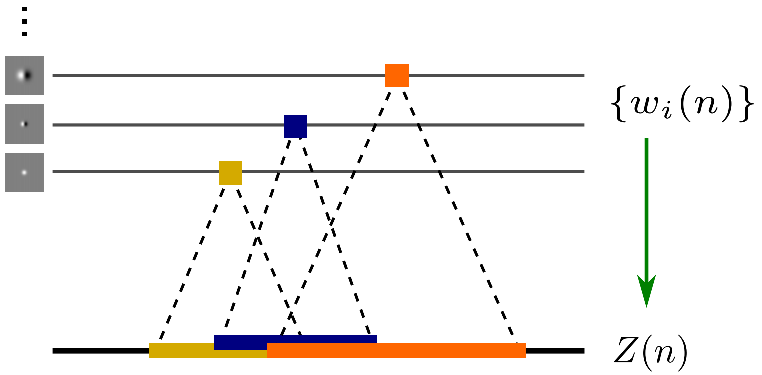
Alternatingly minimize Z and $\{w_i(n)\}$, keeping the other constant.

CASE STUDY: MONOCULAR DEPTH ESTIMATION

Globalization

Fix w, minimize wrt Z

Efficient least-squares in the Fourier-domain.



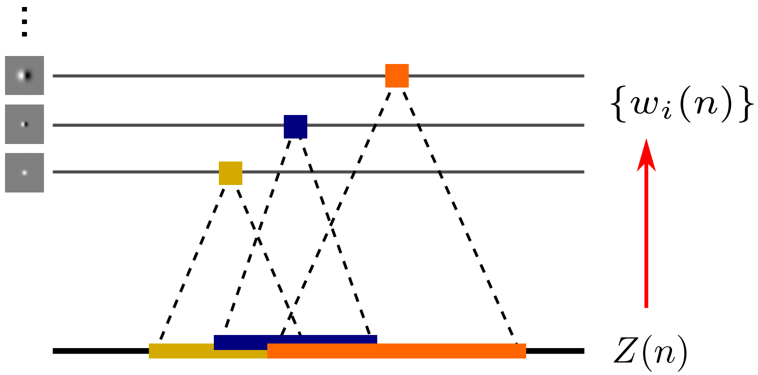
$$Z = \arg\min_{Z} \min_{\{w_i(n)\}} -\left[\sum_{i,n} \log p_{i,n}\left(w_i(n)\right)\right] + \frac{\beta}{2}\left[\sum_{i,n} \left|w_i(n) - (Z * k_i)(n)\right|^2\right]$$

CASE STUDY: MONOCULAR DEPTH ESTIMATION

Globalization

Fix Z, minimize wrt w

Independent for each $w_i(n)$



$$Z = \arg\min_{Z} \min_{\{w_i(n)\}} - \left[\sum_{i,n} \underline{\log p_{i,n}\left(w_i(n)\right)} \right] + \frac{\beta}{2} \left[\sum_{i,n} \underline{\left| w_i(n) - (Z * k_i)(n) \right|^2} \right]$$

CASE STUDY: MONOCULAR DEPTH ESTIMATION

Experimental Results

CASE STUDY: MONOCULAR DEPTH ESTIMATION

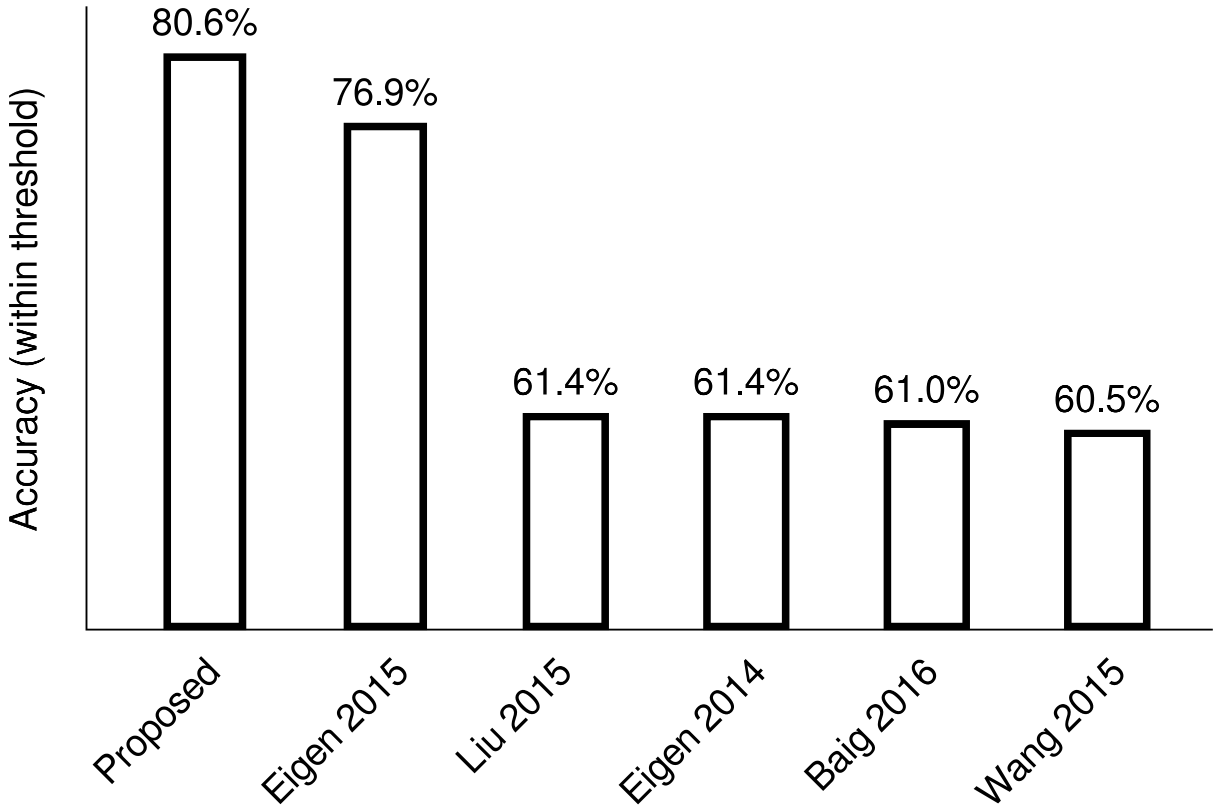
Experimental Results

NYUv2 Depth Benchmark

- Ground truth data from Kinect.
- 56,000 training pairs, 100 validation.
- 654 Test scenes.

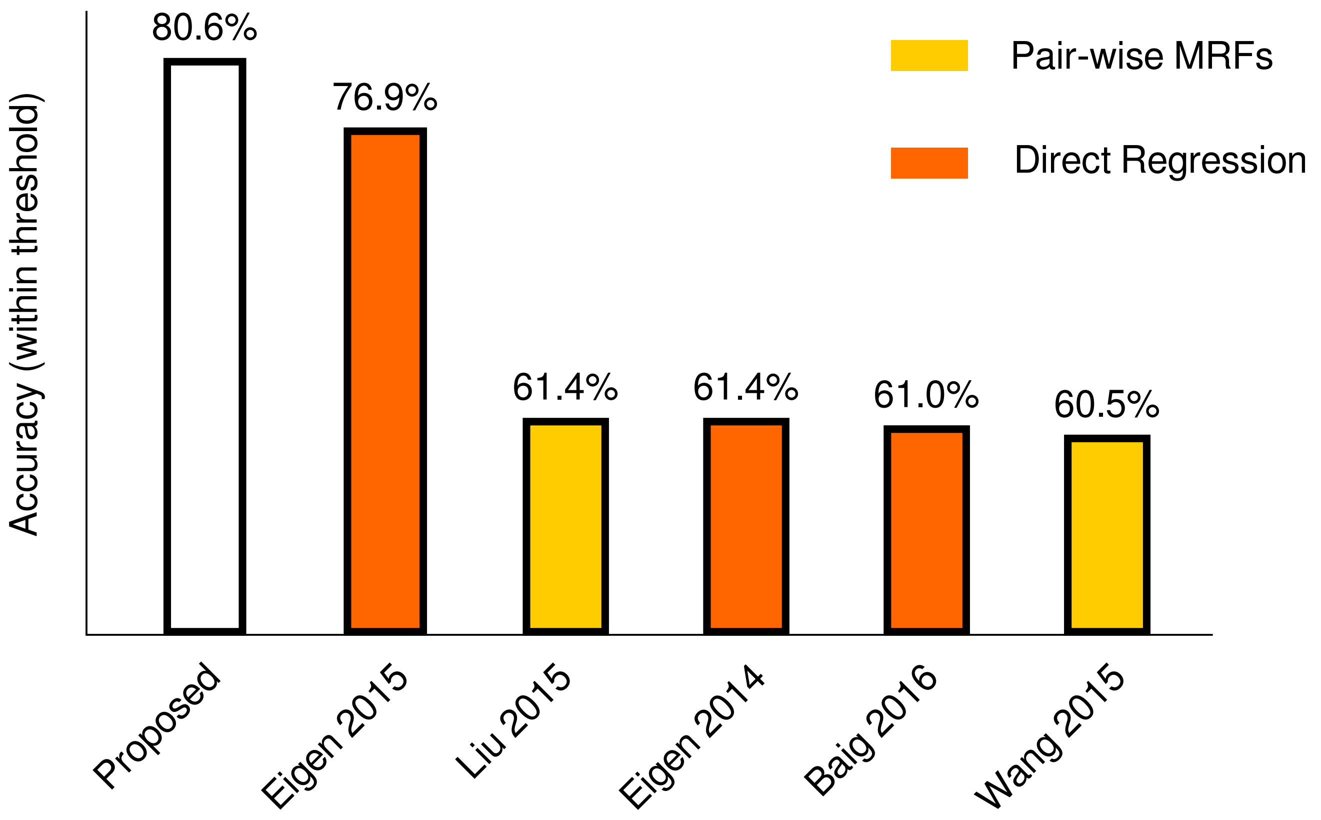
CASE STUDY: MONOCULAR DEPTH ESTIMATION

Experimental Results



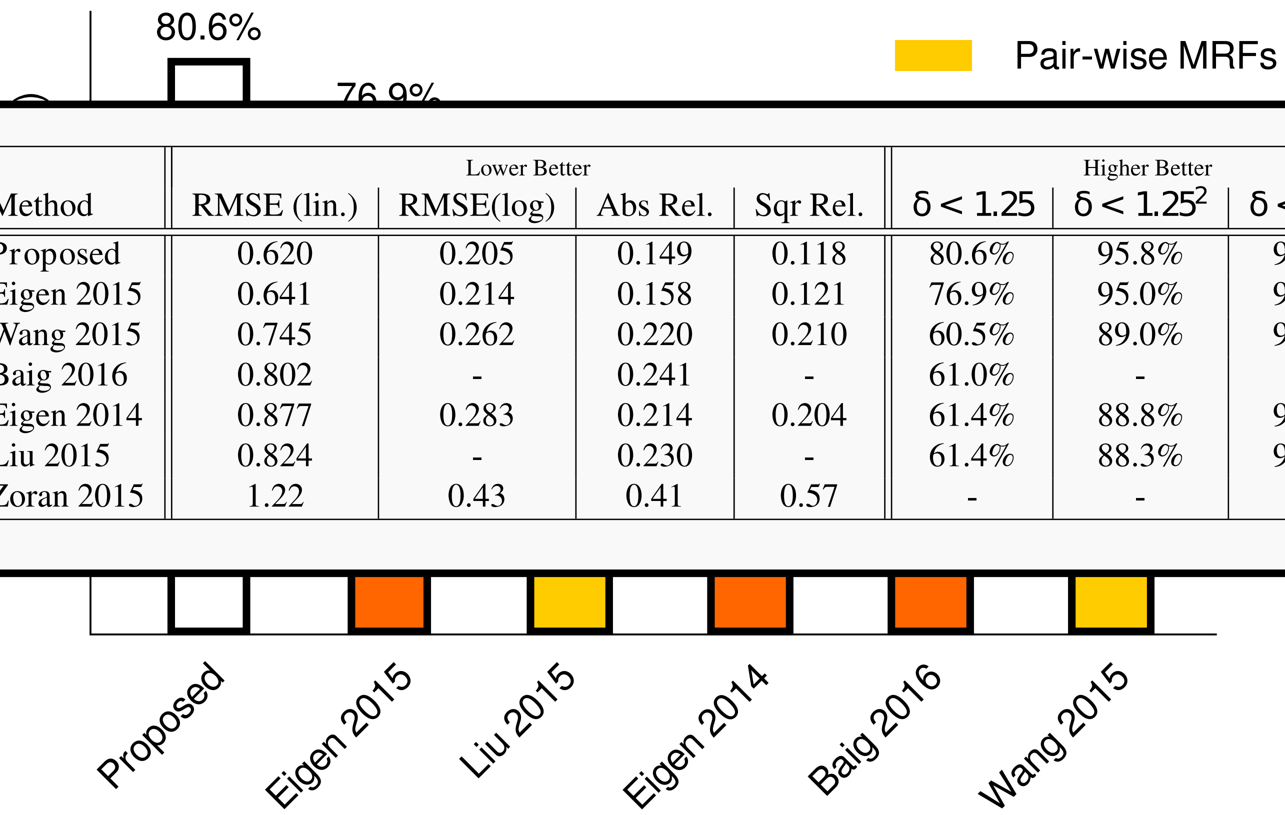
CASE STUDY: MONOCULAR DEPTH ESTIMATION

Experimental Results



CASE STUDY: MONOCULAR DEPTH ESTIMATION

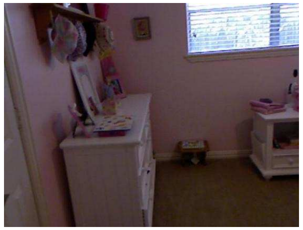
Experimental Results



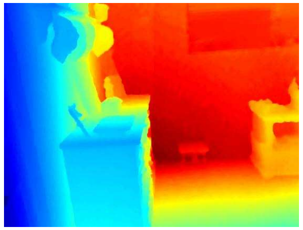
Method	RMSE (lin.)	RMSE(log)	Abs Rel.	Sqr Rel.	$\delta < 1.25$	$\delta < 1.25^2$	$\delta < 1.25^3$
	Lower Better				Higher Better		
Proposed	0.620	0.205	0.149	0.118	80.6%	95.8%	98.7%
Eigen 2015	0.641	0.214	0.158	0.121	76.9%	95.0%	98.8%
Wang 2015	0.745	0.262	0.220	0.210	60.5%	89.0%	97.0%
Baig 2016	0.802	-	0.241	-	61.0%	-	-
Eigen 2014	0.877	0.283	0.214	0.204	61.4%	88.8%	97.2%
Liu 2015	0.824	-	0.230	-	61.4%	88.3%	97.1%
Zoran 2015	1.22	0.43	0.41	0.57	-	-	-

CASE STUDY: MONOCULAR DEPTH ESTIMATION

Experimental Results



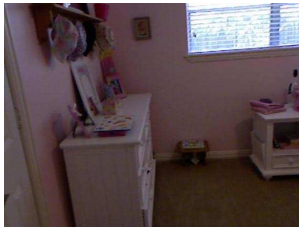
Input Image



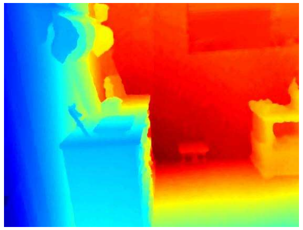
Ground Truth
Depth

CASE STUDY: MONOCULAR DEPTH ESTIMATION

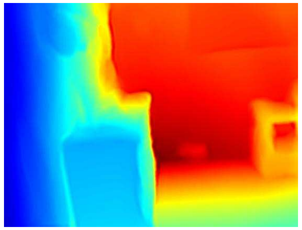
Experimental Results



Input Image



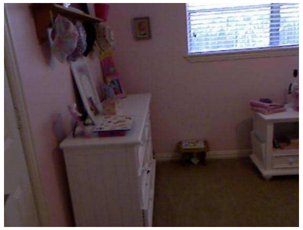
Ground Truth
Depth



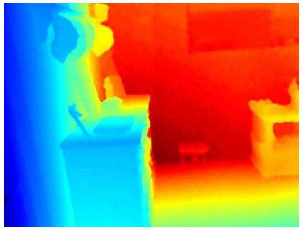
Proposed Method

CASE STUDY: MONOCULAR DEPTH ESTIMATION

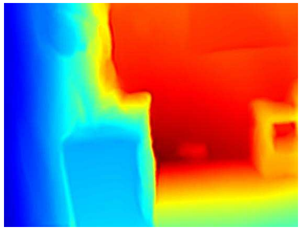
Experimental Results



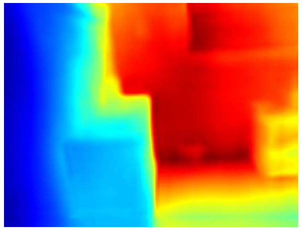
Input Image



Ground Truth Depth



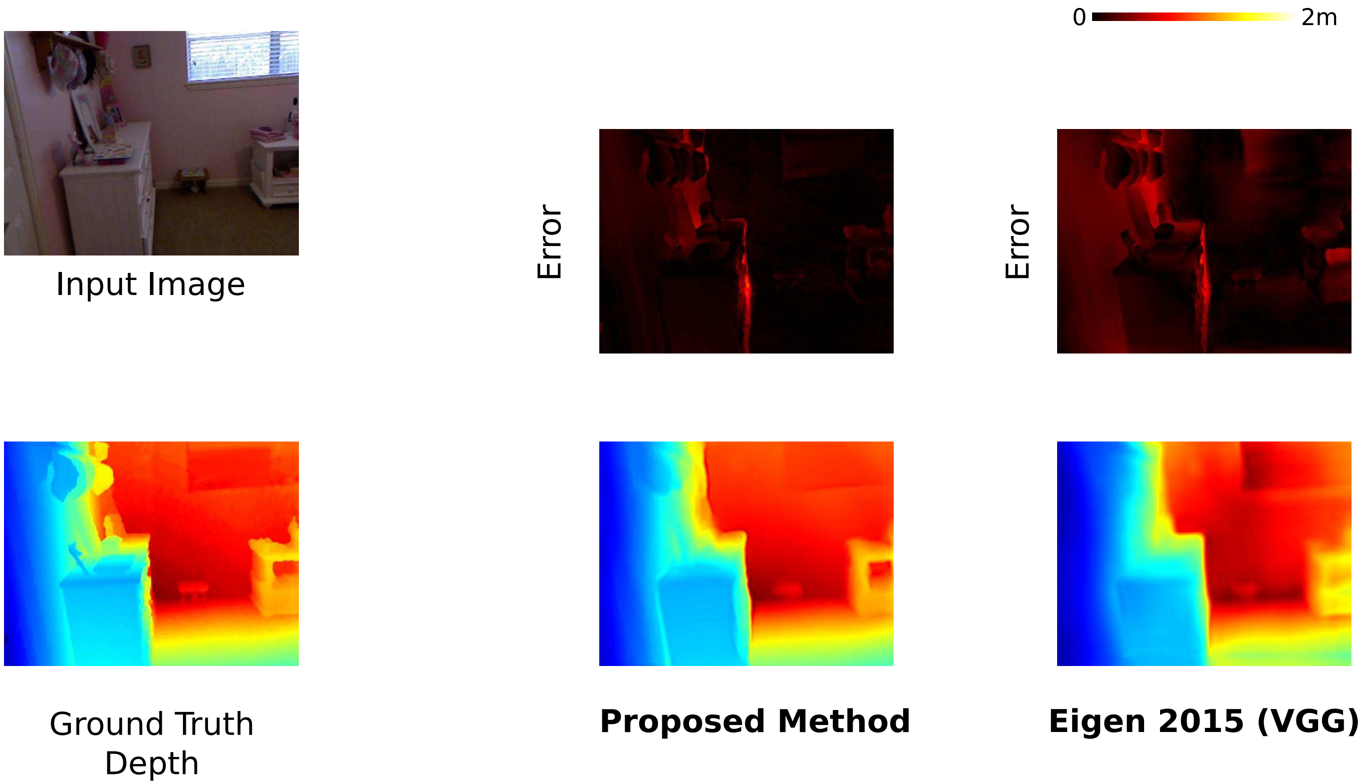
Proposed Method



Eigen 2015 (VGG)

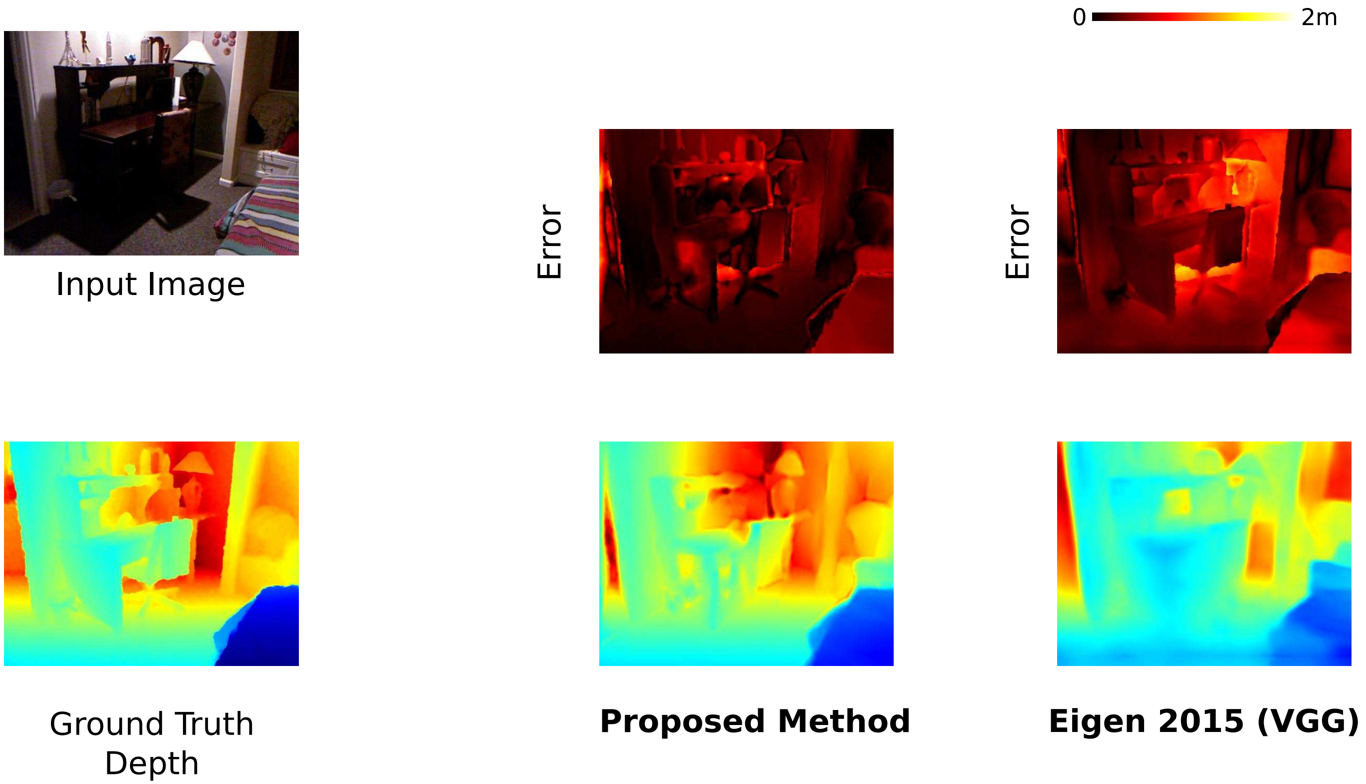
CASE STUDY: MONOCULAR DEPTH ESTIMATION

Experimental Results



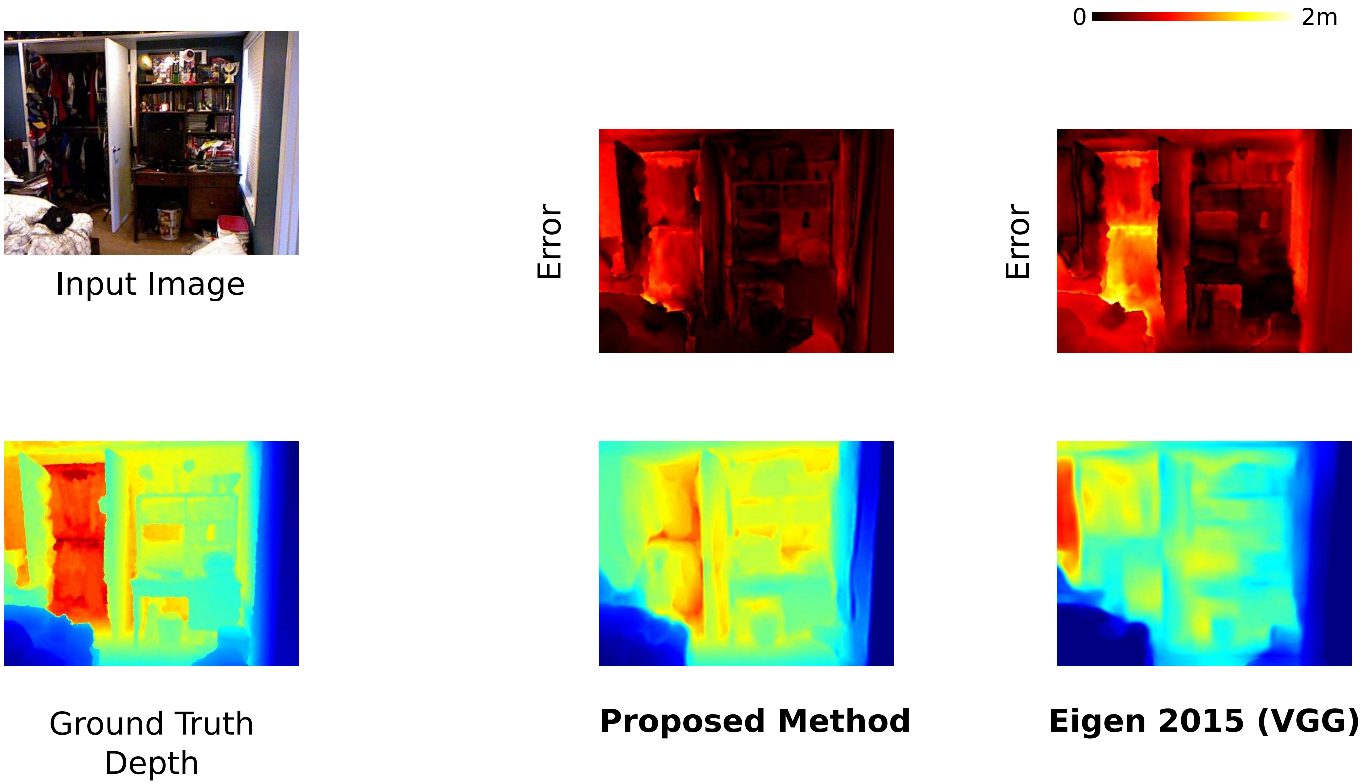
CASE STUDY: MONOCULAR DEPTH ESTIMATION

Experimental Results



CASE STUDY: MONOCULAR DEPTH ESTIMATION

Experimental Results



CASE STUDY: MONOCULAR DEPTH ESTIMATION

Beyond the Benchmark

$$Z = \arg\max_{Z} \sum_{i,n} \log p_{i,n}\left((Z * k_i)(n)\right)$$

CASE STUDY: MONOCULAR DEPTH ESTIMATION

Beyond the Benchmark

$$Z = \arg\max_{Z} \sum_{i,n} \log p_{i,n}\left(\left(Z * k_i\right)(n)\right)$$

Rich, distributional, interpretable

CASE STUDY: MONOCULAR DEPTH ESTIMATION

Beyond the Benchmark

$$Z = \arg\max_{Z} \sum_{i,n} \log p_{i,n}\left((Z * k_i)(n)\right)$$

CASE STUDY: MONOCULAR DEPTH ESTIMATION

Beyond the Benchmark

$$Z = \arg\max_Z \sum_{i,n} \log p_{i,n}\left((Z * k_i)(n)\right)$$

Other cues.

User input.

Noisy / sparse depth measurements.

CASE STUDY: MONOCULAR DEPTH ESTIMATION

Beyond the Benchmark

$$Z = \arg\max_{Z} \sum_{i,n} \log p_{i,n}\left((Z * k_i)(n)\right)$$

Other cues.

User input.

Noisy / sparse depth measurements.

Common substrate for local estimates from different cues.

CASE STUDY: MONOCULAR DEPTH ESTIMATION

Beyond the Benchmark

$$Z = \arg\max_{Z} \sum_{i,n} \log p_{i,n}\left((Z * k_i)(n)\right)$$

CASE STUDY: MONOCULAR DEPTH ESTIMATION

Beyond the Benchmark

$$Z = \arg\max_{Z} \sum_{i,n} \log p_{i,n}\left((Z * k_i)(n)\right)$$

$$P(Z(n) < \delta)$$

DISCUSSION

- Flavor of what a research project looks like.
- Look at group website for papers describing some of our other recent work.

Questions ?