Scene Understanding Tutorial: Surfaces and 3D Models

Derek Hoiem

University of Illinois

Outline

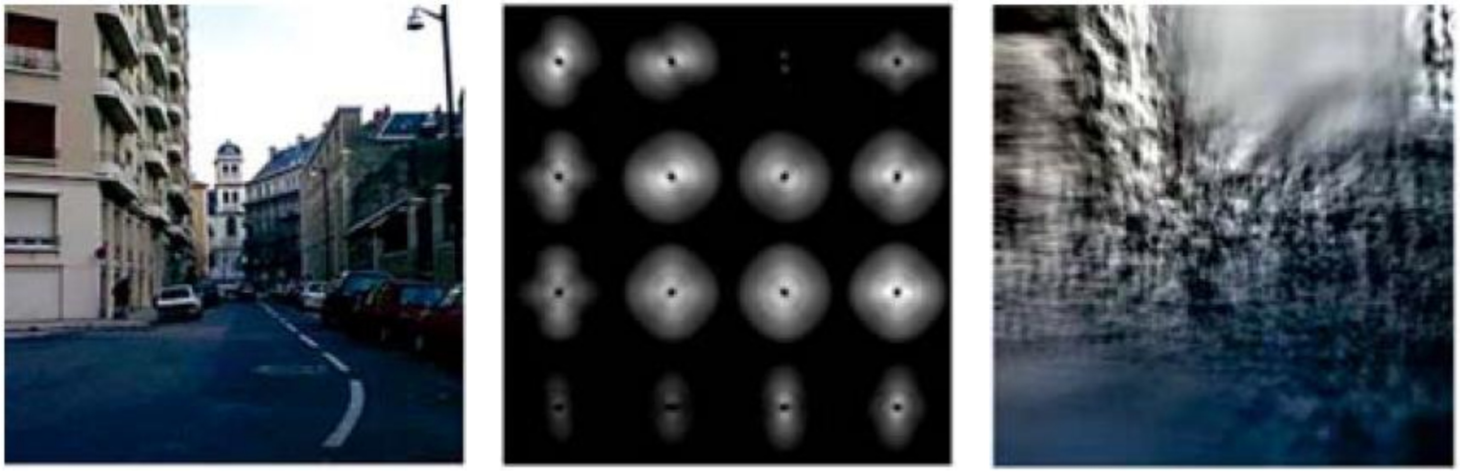
1) How do we model 3D scenes?

2) How do we recover individual geometric properties?
   – Surface orientations / materials / depth
   – Occlusion boundaries
   – Viewpoint

3) How do we infer complete 3D scenes?
   - Probabilistic model
   - Structured SVM
   - Sequential prediction

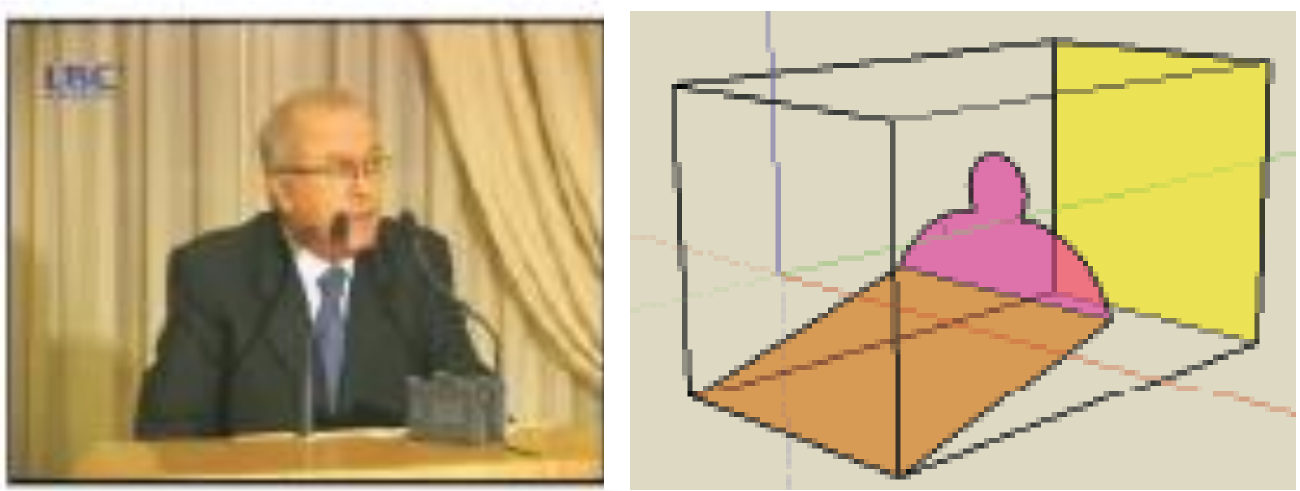
How can we model 3D scenes?

Scene-Level Geometric Description



Gist, Spatial Envelope

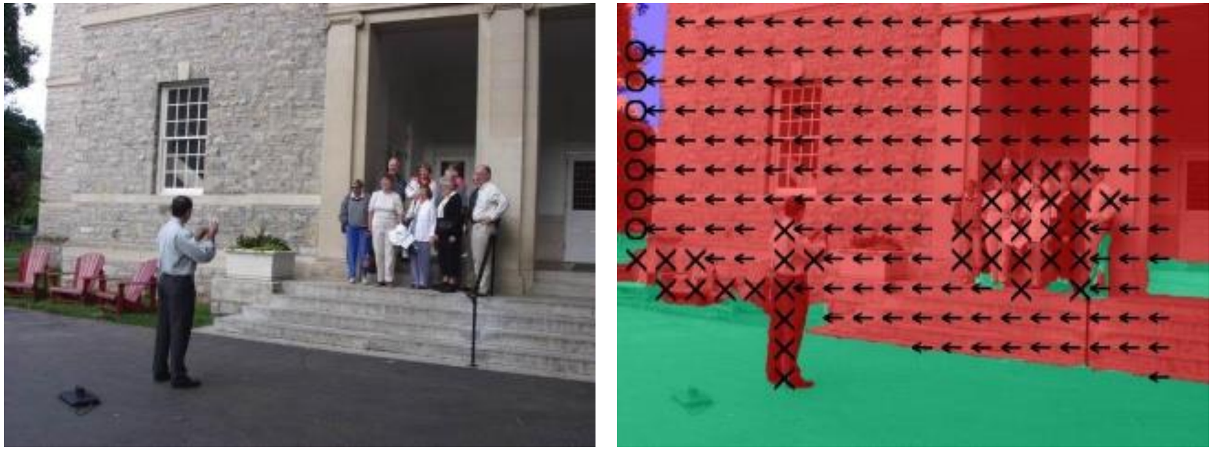
Oliva Torralba 2001, 2006



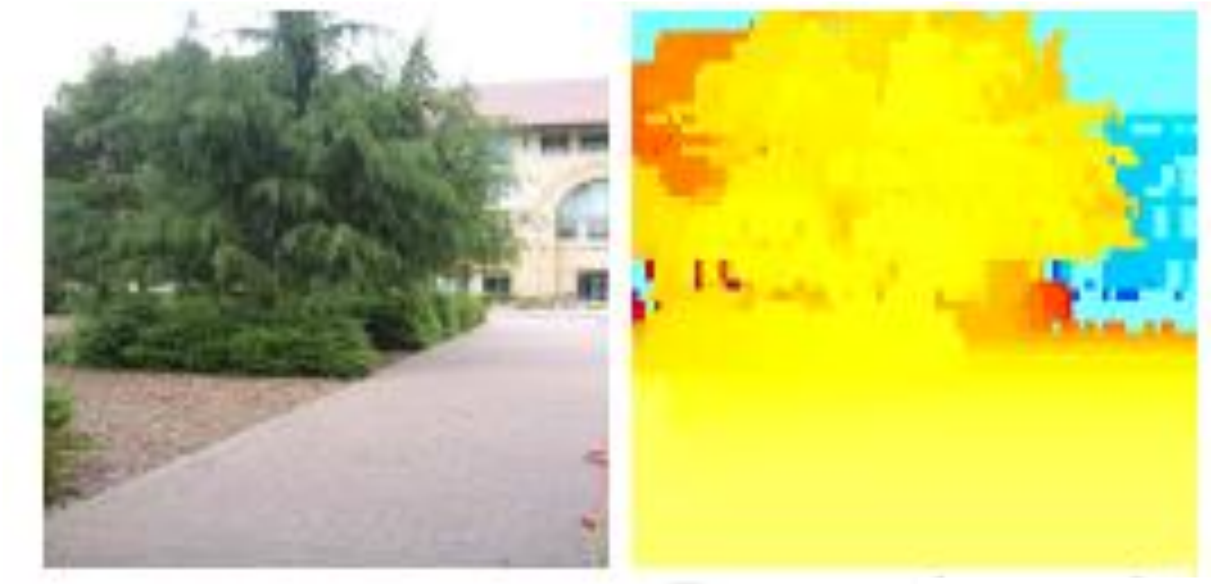
Stages

Nedovic et al. 2007

Pixel Map Geometric Description



Geometric Context Hoiem et al. 2005, 2007

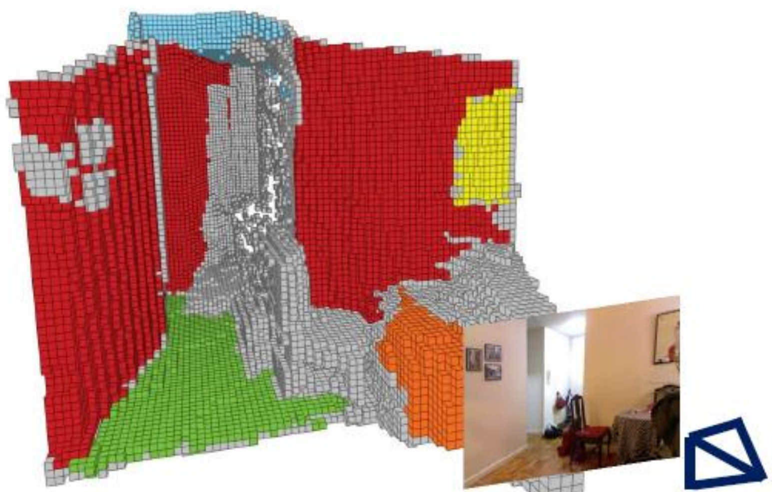


Depth Map Saxena et al. 2005, 2007

Loosely structured 3d model

Point Cloud

Agarwal et al. 2009



Voxels

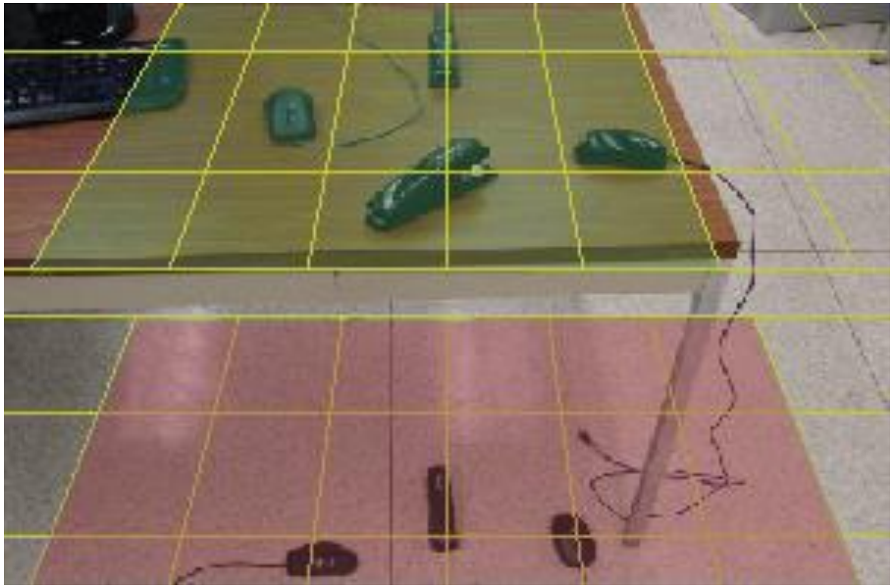
Kim et al. 2013

Structured models: supporting planes

Hoiem et al. 2006, 2008

Ground Plane

Bao et al. 2010



Multiple Support Planes

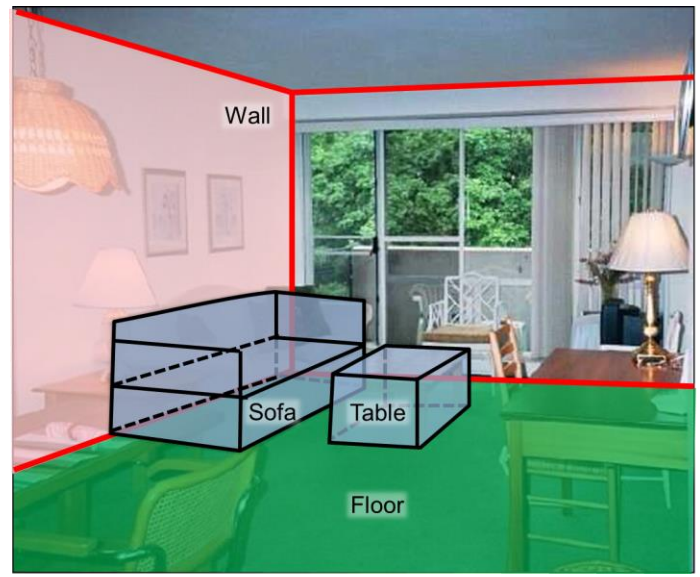
Structured models: coarse 3d

Hoiem et al. 2008

Ground Plane with Billboards

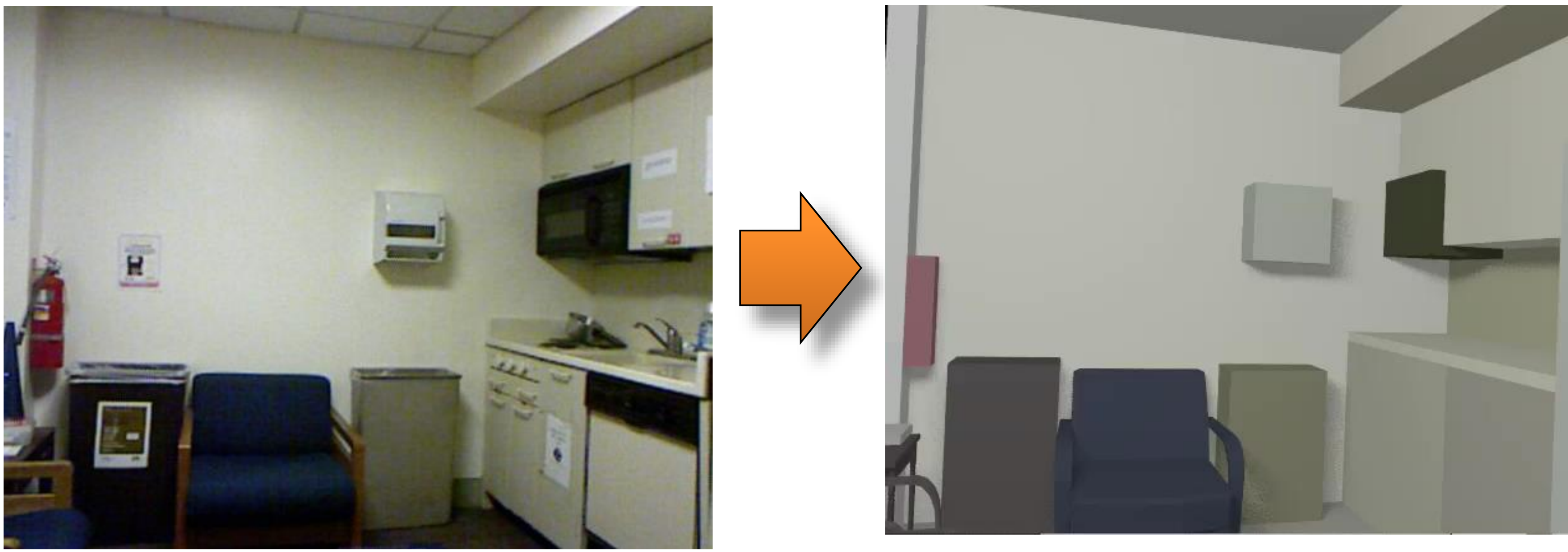
Structured models: coarse 3d

Ground Plane with Walls Lee et al. 2010



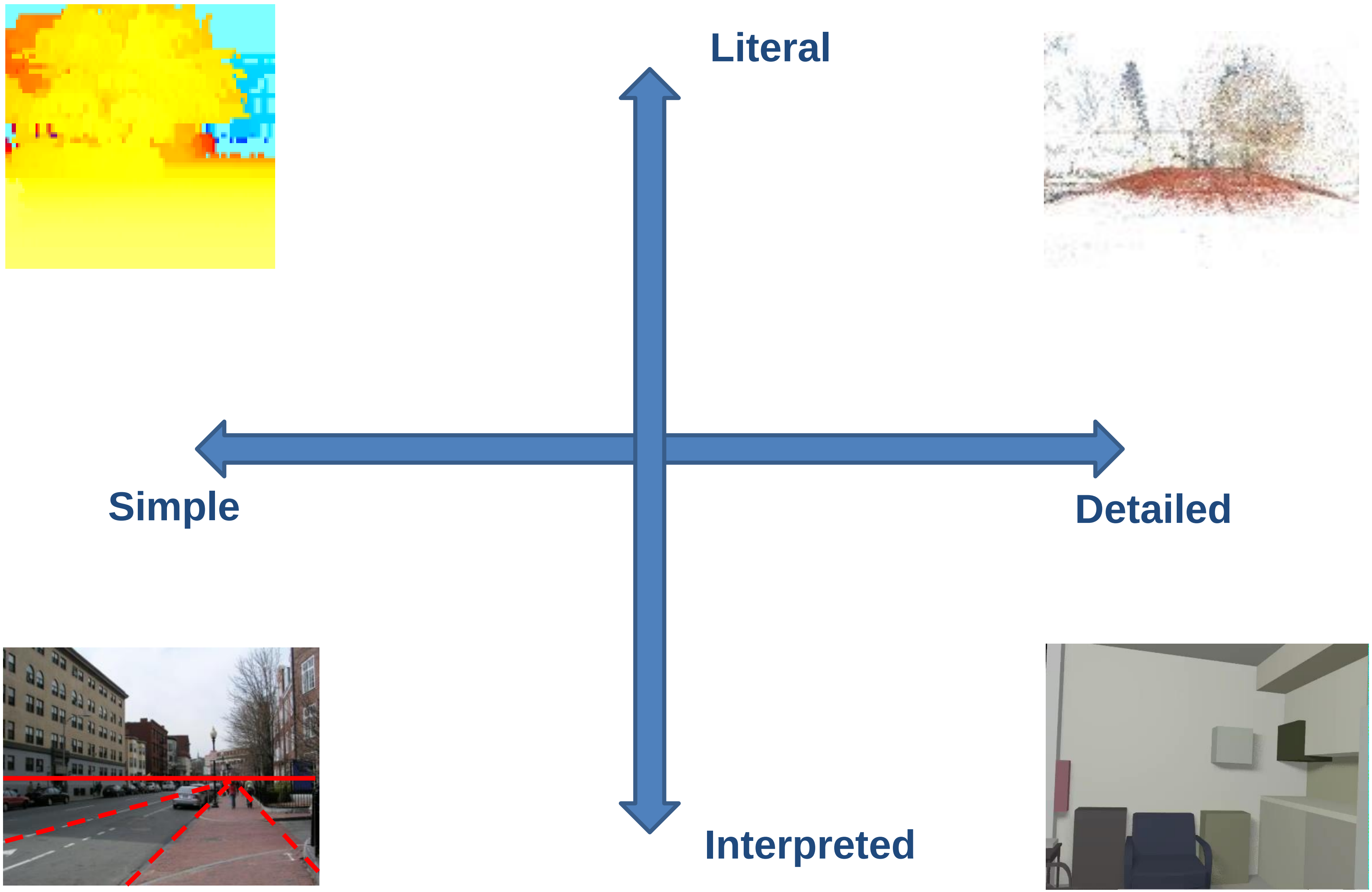
3D Box Model Hedau et al. 2009/10

Structured models: detailed 3D



Guo Hoiem (unpublished)

Representational Trade-Offs



Representational Trade-Offs

Literal

Lends to solutions derived from geometry and physics

Easily quantifiable results

Not directly useful for most interaction/understanding tasks

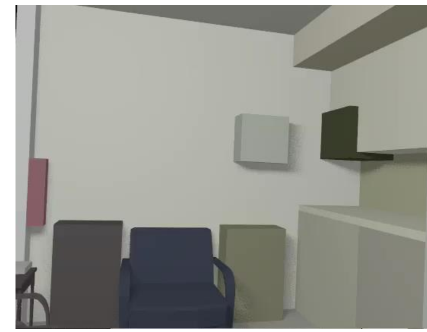
Requires hand-crafted representations and annotations

More difficult to measure error

More directly useful for high-level tasks

Interpreted

Representational Trade-Offs



Simple

Robust inference from limited cues

Incomplete scene information

Useful for general guidance and priors

Complex

Requires more sensor data for similar accuracy; important to represent uncertainty

Complete models enable high-level priors and constraints

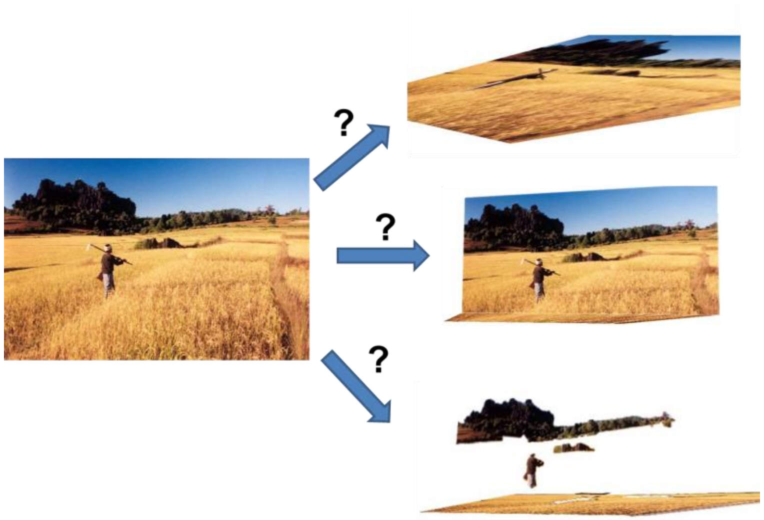
Useful for moving, grasping, understanding

How can we recover geometric properties?

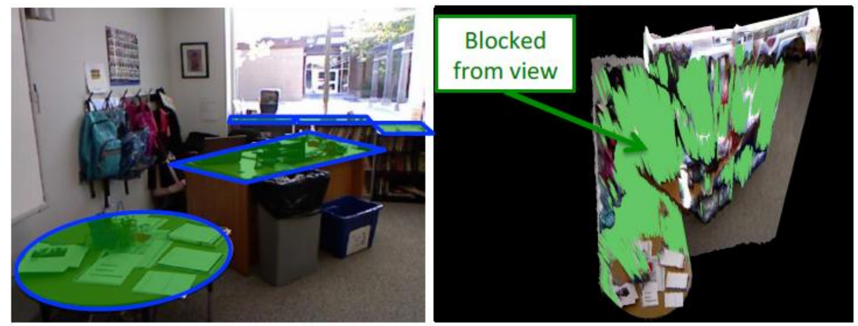
- Surface orientations, materials, depth
- Occlusion boundaries
- Viewpoint

The challenges

Ambiguity 2D projection
(loss of depth info)



Ambiguity from occlusion
(loss of 3d info)



Ambiguity in connectedness
(requires direct manipulation)

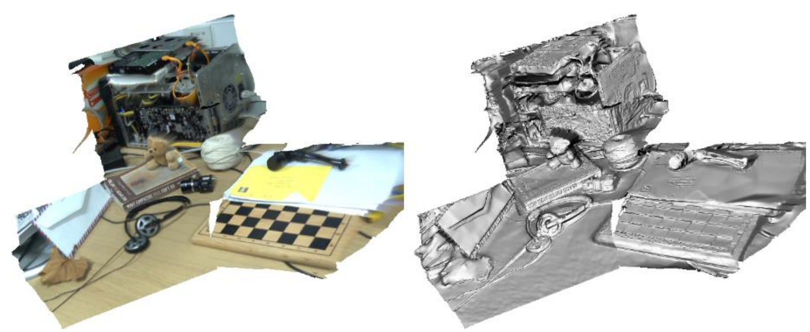
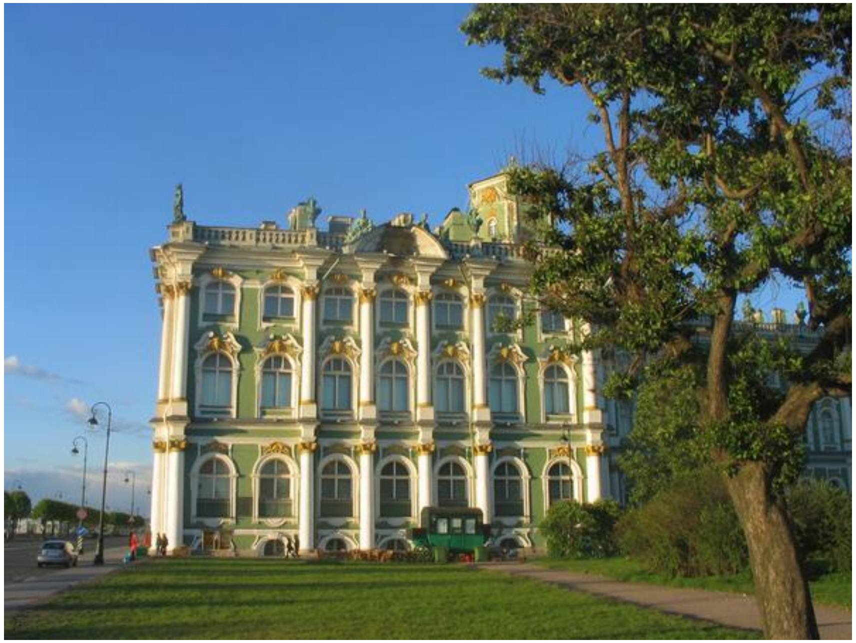


Image credit (3d surface): Newcombe Davis 2010

Geometric properties can be inferred only because our world is structured

Abstract World

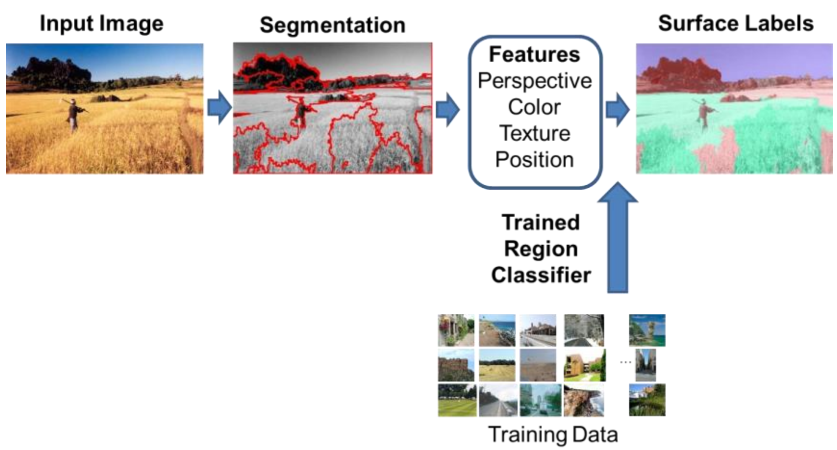


Our World

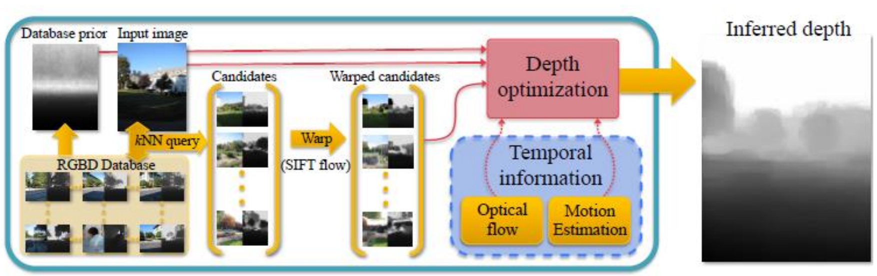
Image Credit (left): F. Cunin
and M.J. Sailor, UCSD

Recovering surface properties: two main approaches

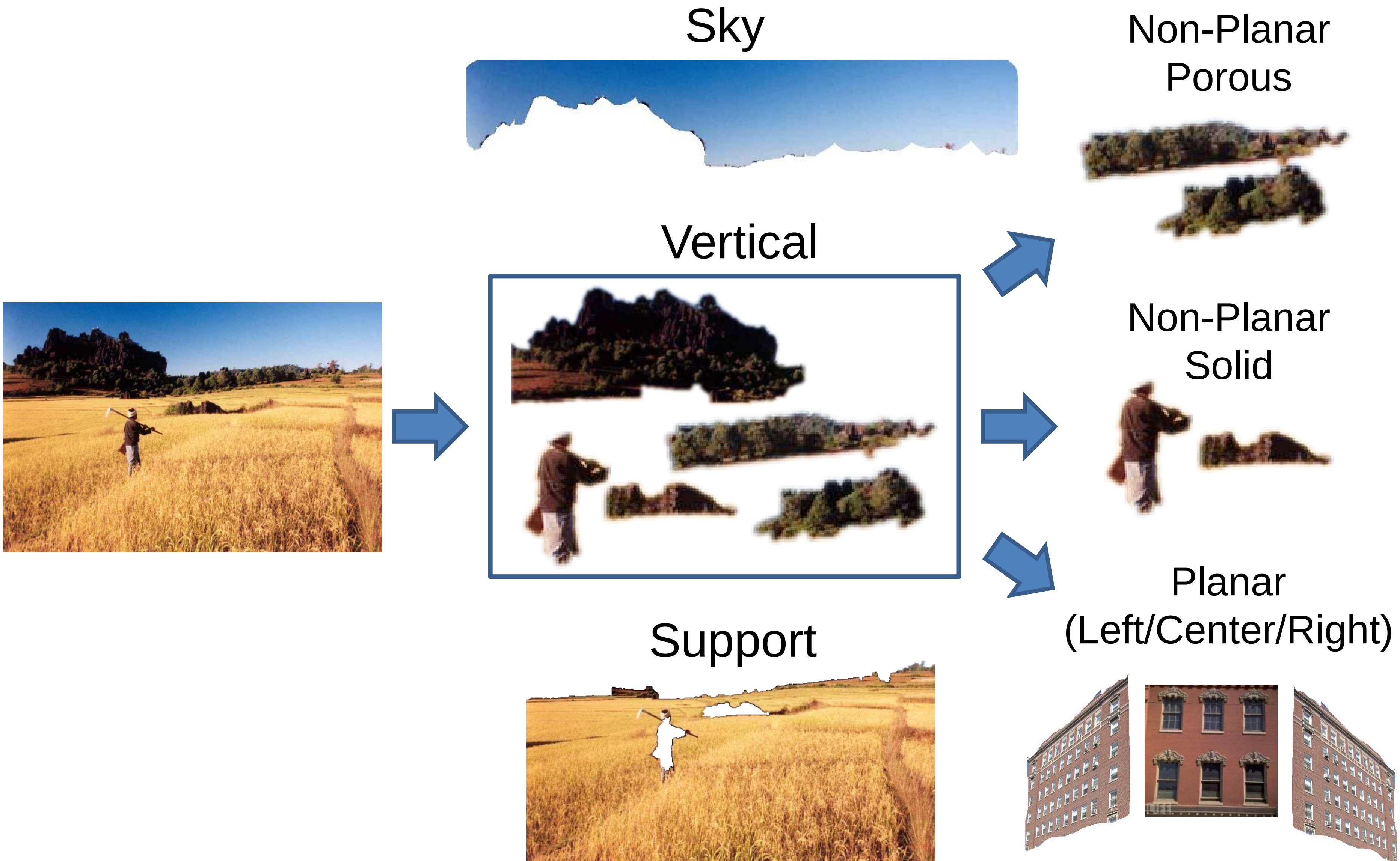
1. Train classifier/regressor



2. Transfer from patch/image matcher
   – Discussed in more detail later in tutorial

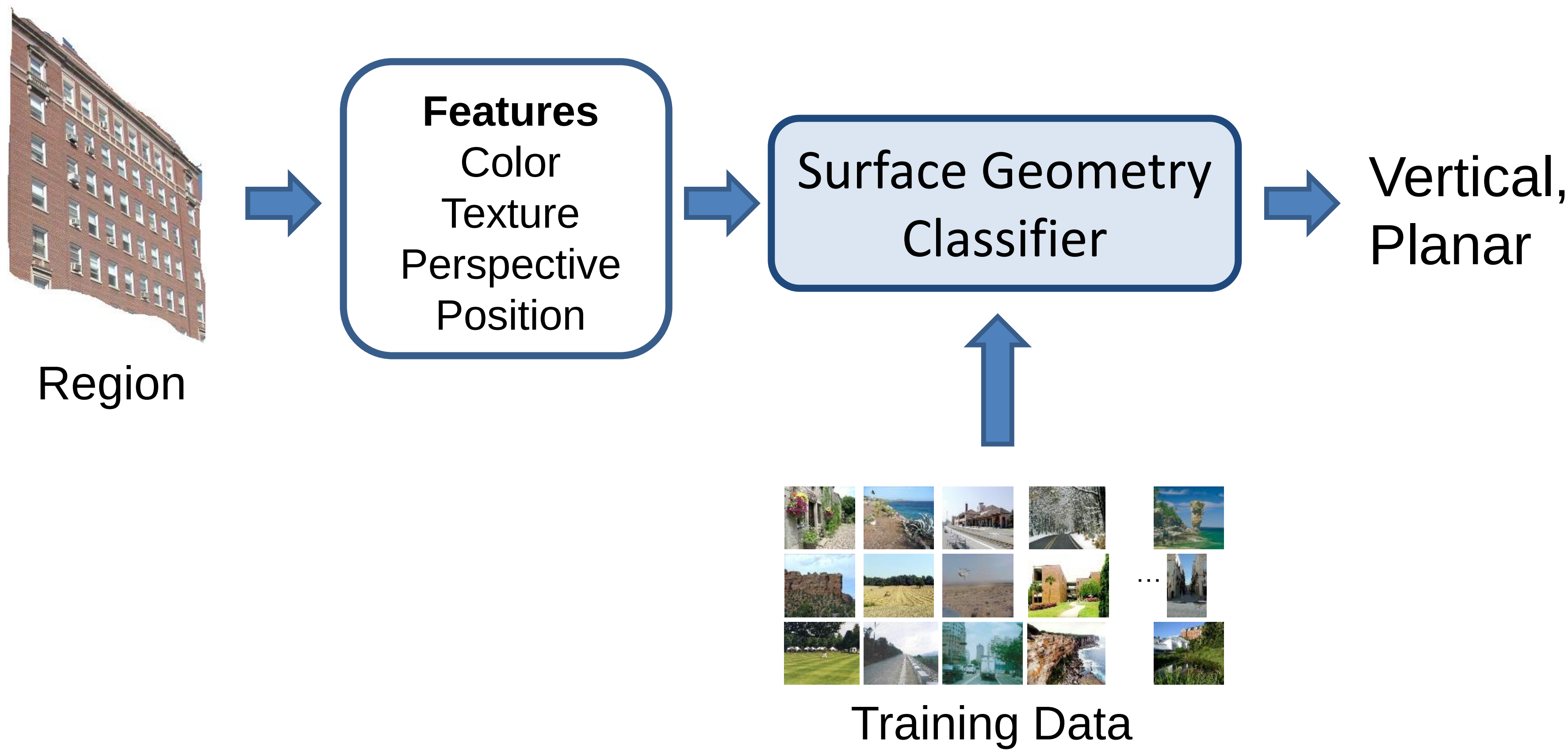


Example: describe 3D surfaces with geometric classes



Hoiem Efros Hebert, ICCV 2005, IJCV 2007

Geometry estimation as recognition



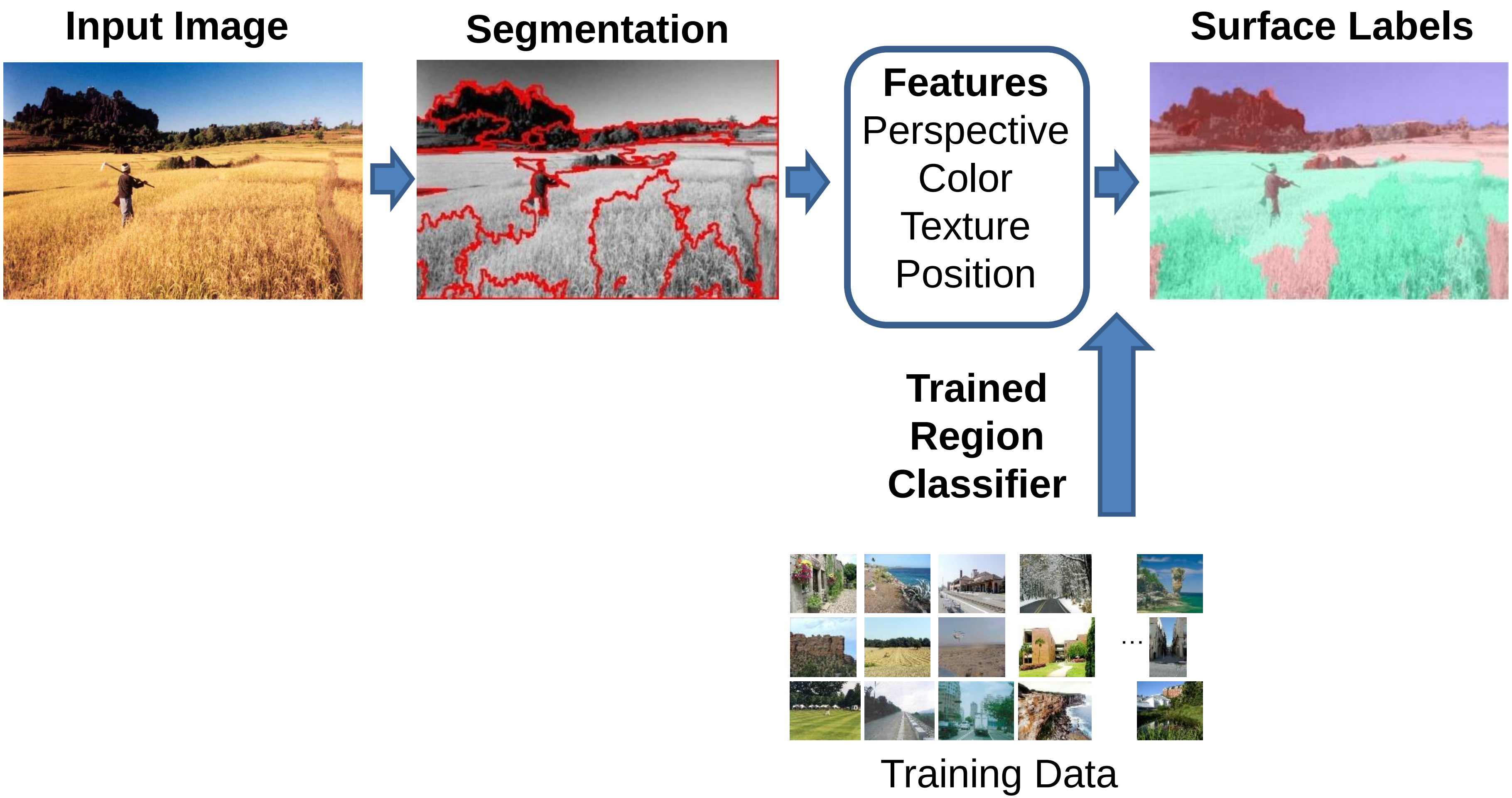
Use a variety of image cues

Vanishing points, lines

Color, texture, image location

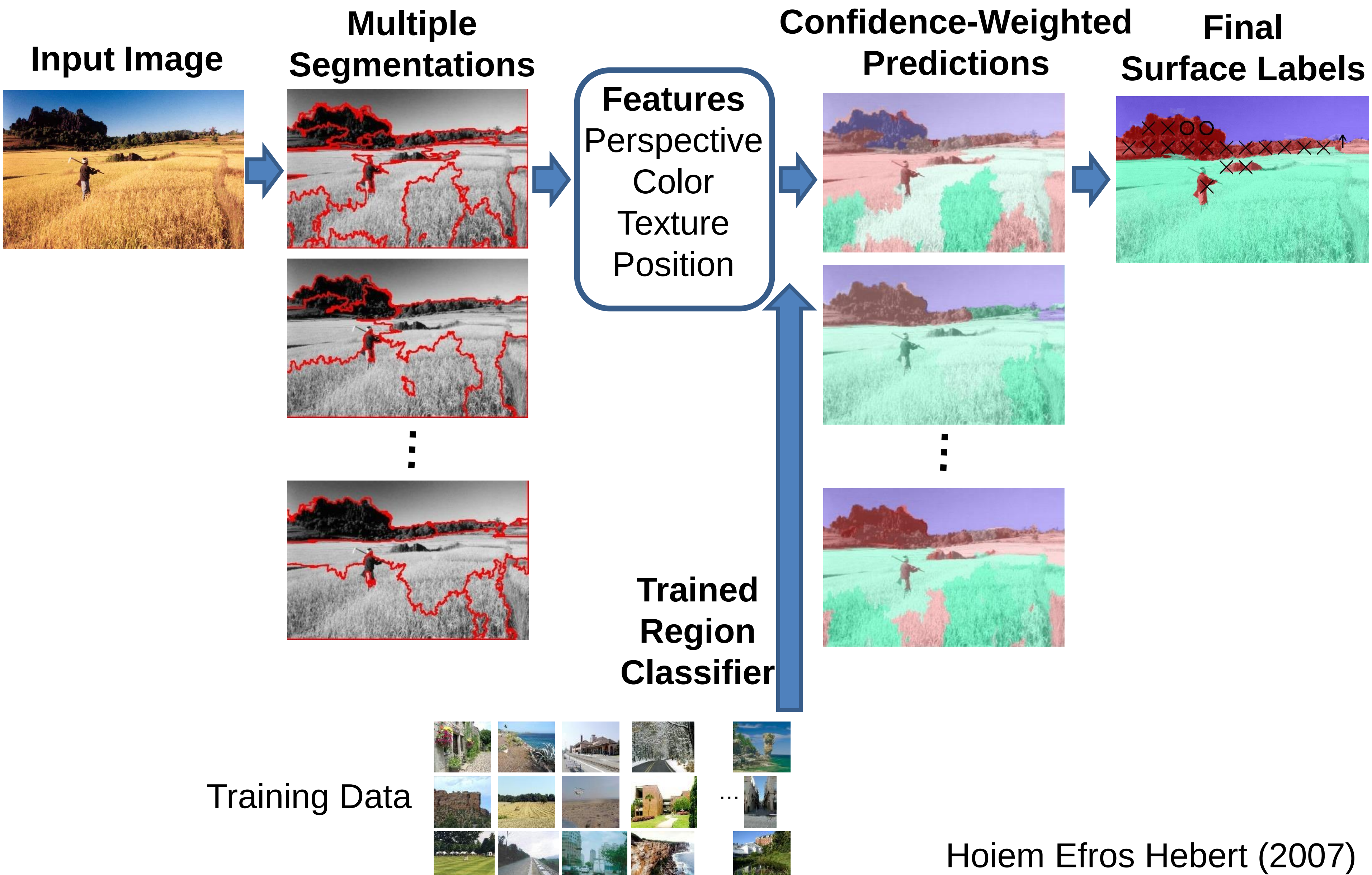
Texture gradient

Surface Layout Algorithm



Hoiem Efros Hebert (2007)

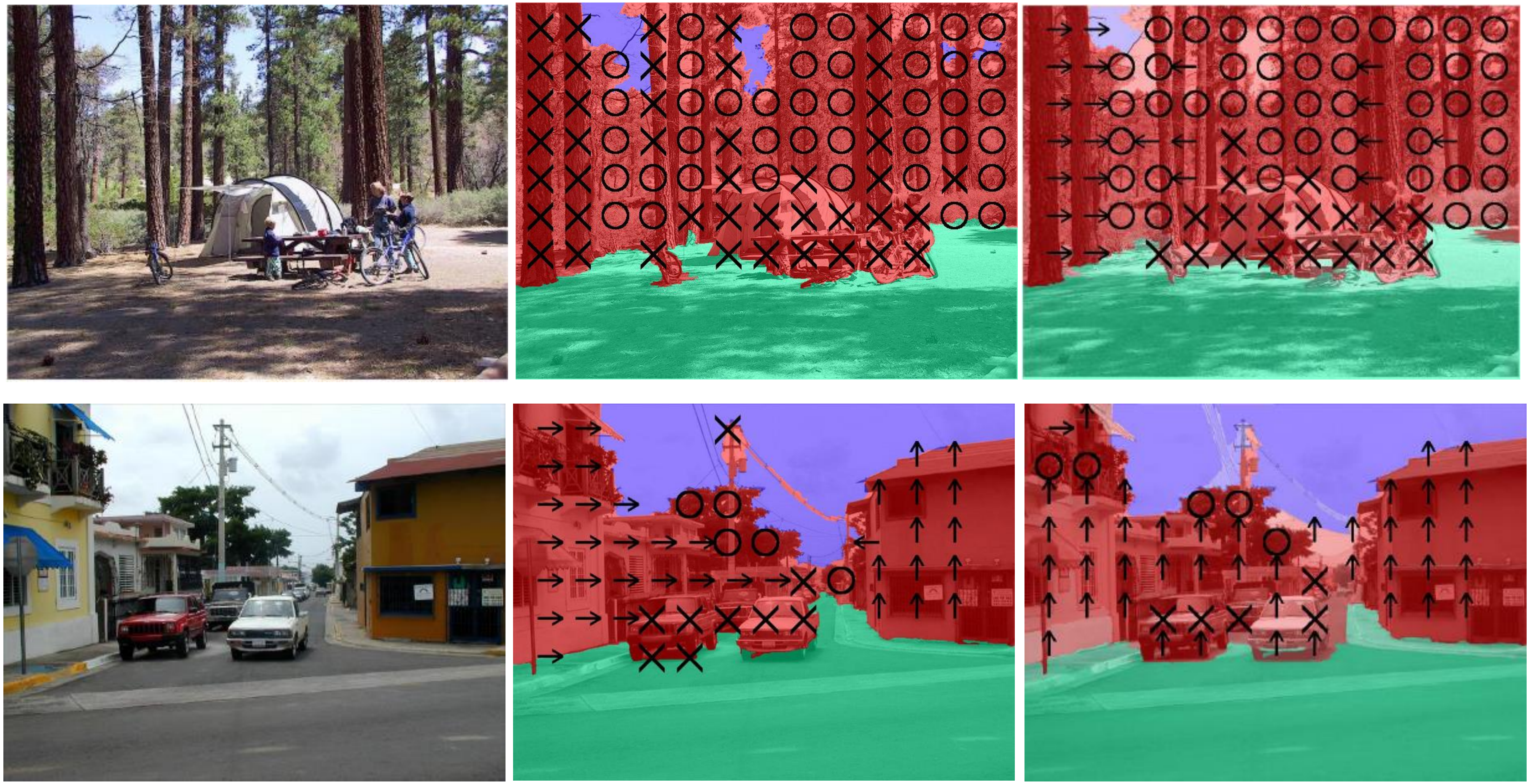
Surface Layout Algorithm



Hoiem Efros Hebert (2007)

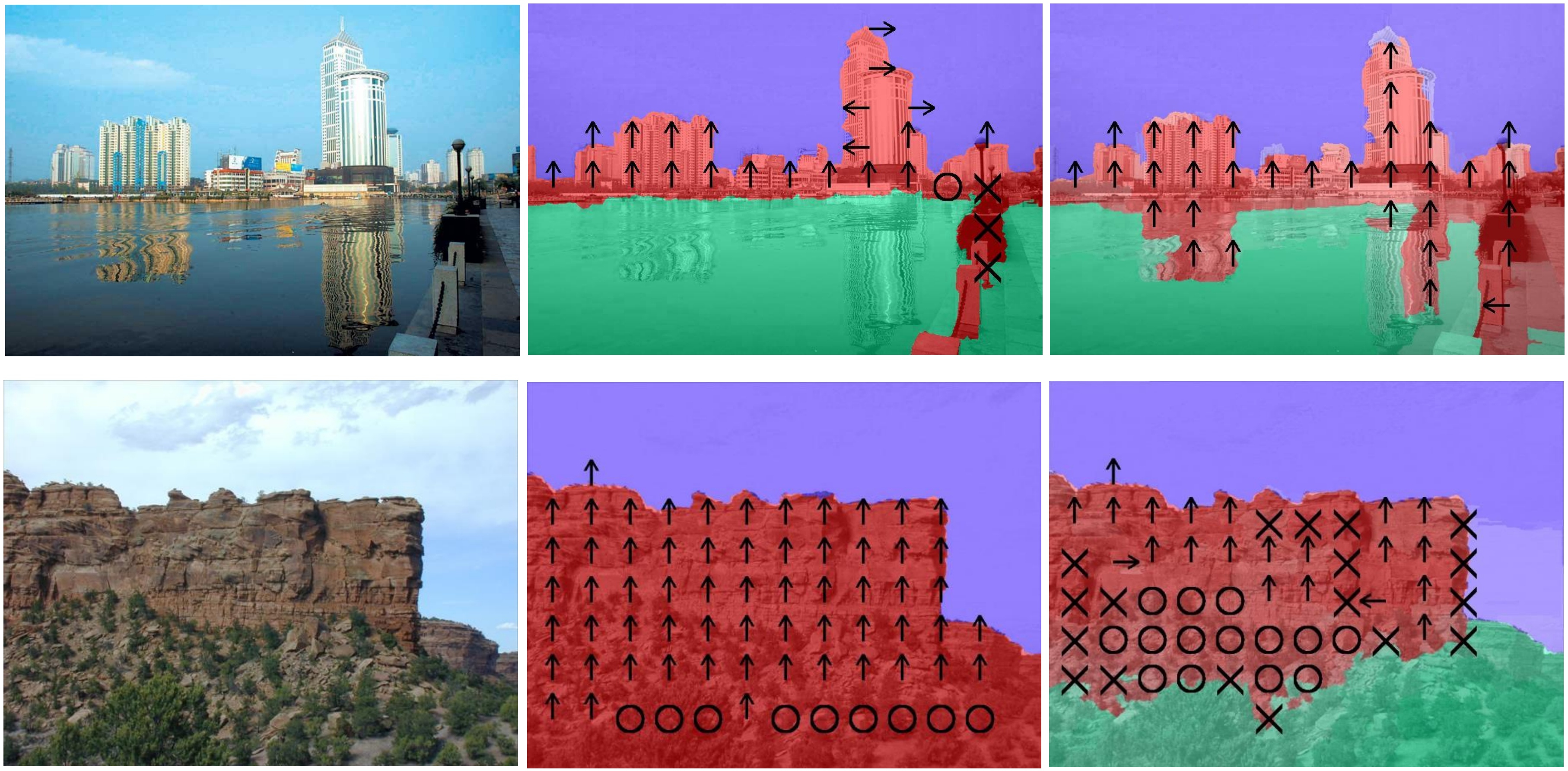
Surface Description Result

Results



Input Image | Ground Truth | Our Result

Failures: Reflections, Rare Viewpoint



Input Image Ground Truth Our Result

General framework for geometric pixel labeling

0.Input
RGB Image
Multiple Images
Video
RGBD Image

1. Split into Regions
Pixels
Square patches
Segmentation
Multiple segmentation

2. Extract features
Color
Texture
Lines (perspective)
Position
3D Normal (w/ depth)
3D Planarity (w/ depth)

3. Classify
Boost decision trees
SVM
KNN
Random forest

4. Regularize Solution
Average predictions
MRF
Fit model

5.Pixel map of labels/values
Geometric classes
Surface normals
Depth
Occlusion boundaries
Materials
Indoor surfaces
Object categories

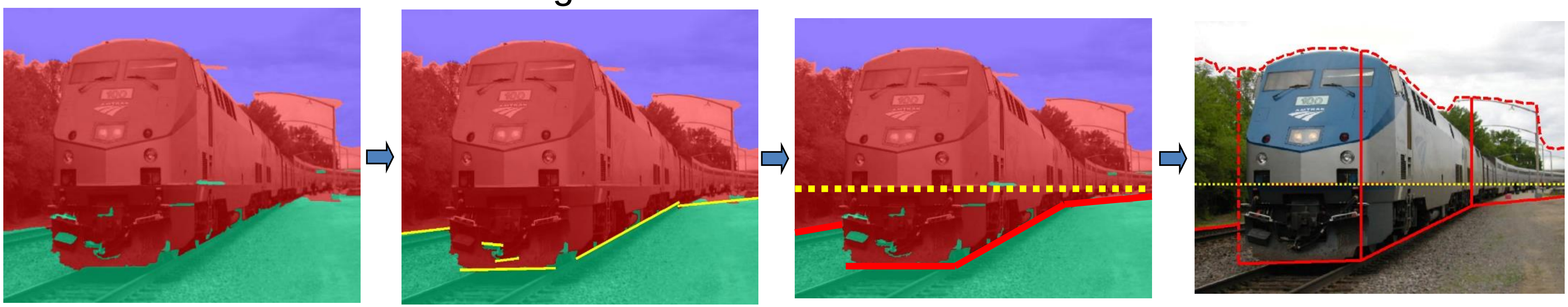
3D reconstruction from geometric context

Labeled Image

Fit Ground-Vertical Boundary with Line Segments

Form Segments into Polylines

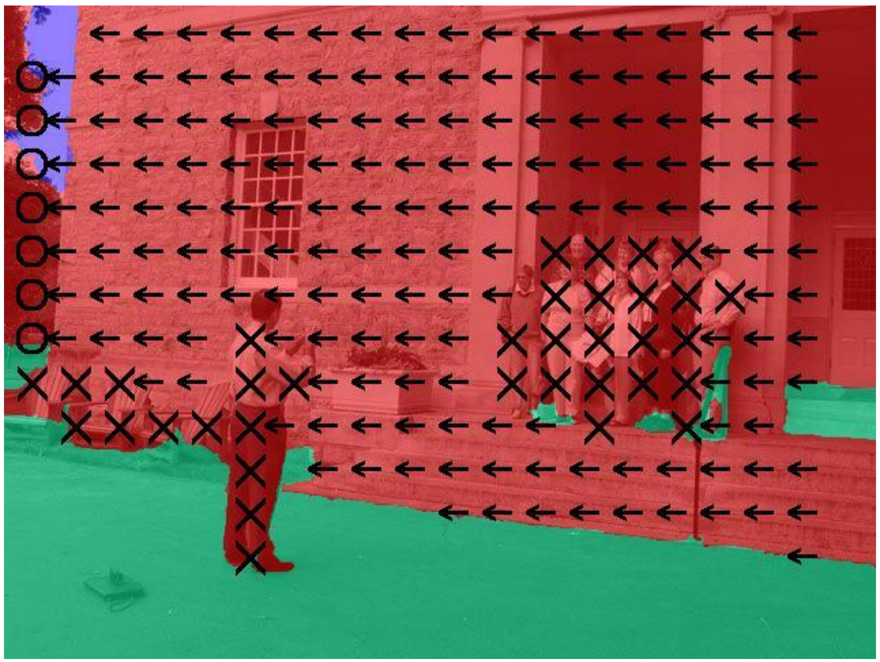
Cut and Fold



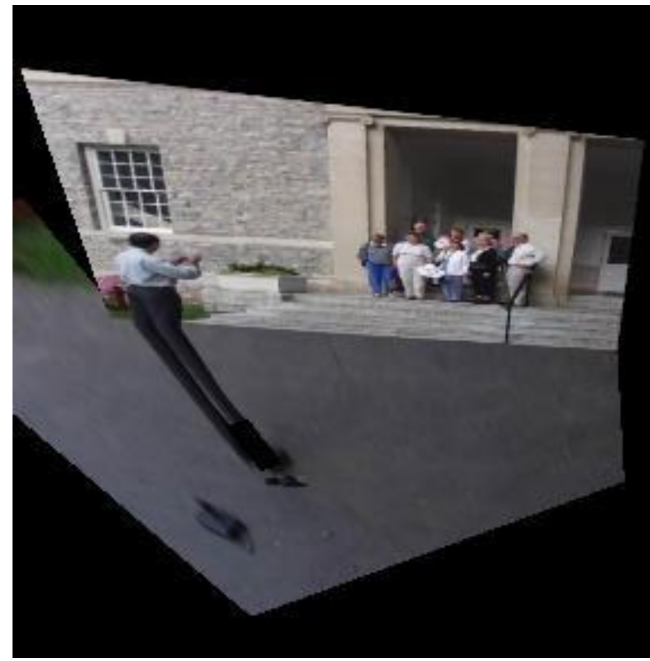
Final Pop-up Model

[Hoiem Efros Hebert 2005]

Need object/occlusion boundaries for more complex scenes

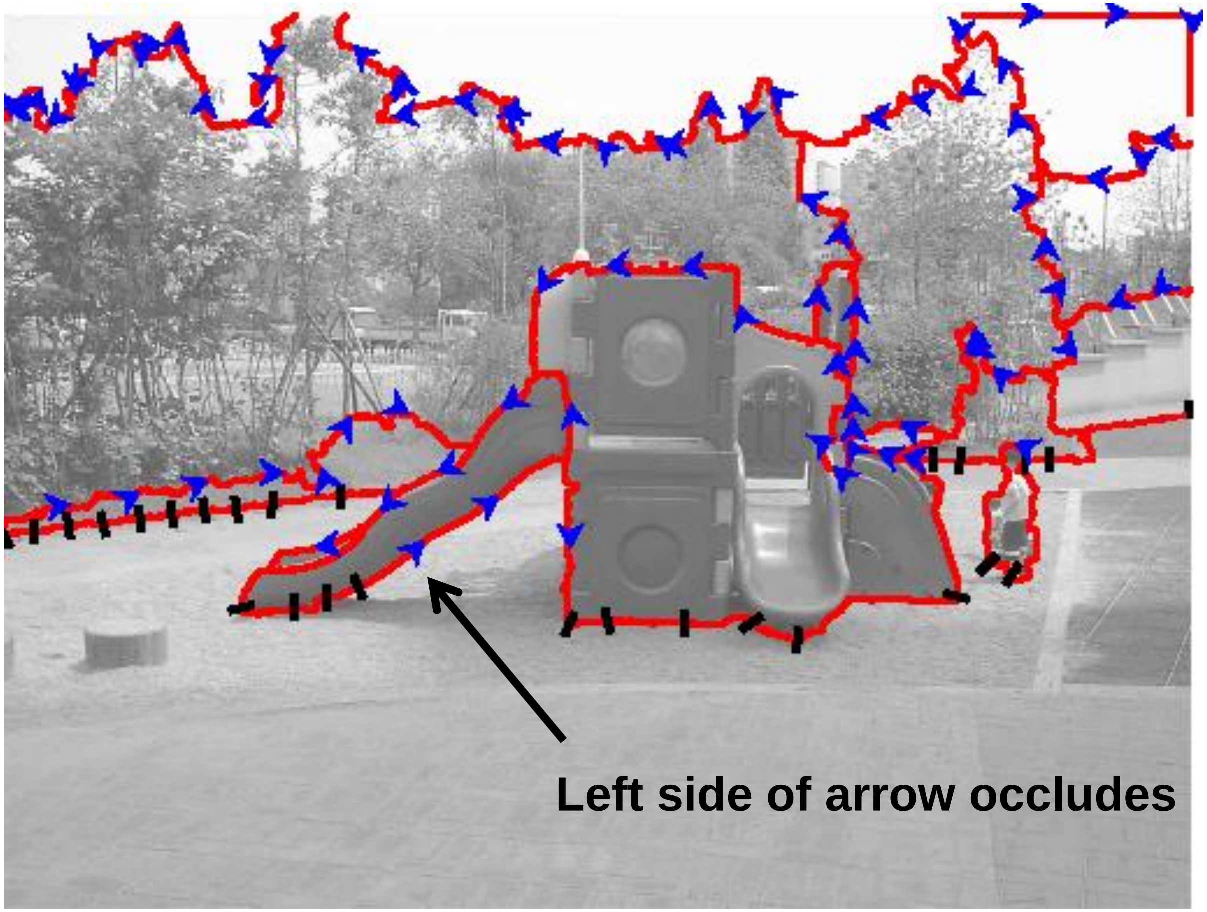


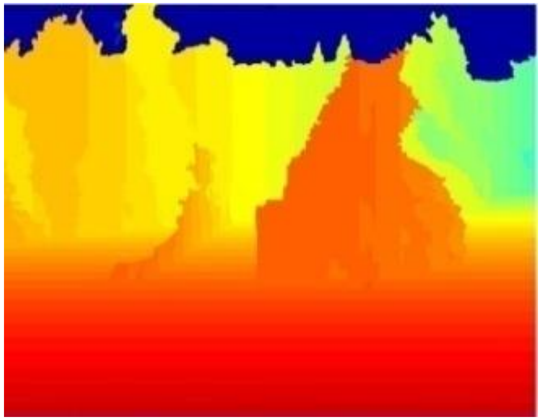
Surface Layout



3D Model

Recovering major occlusions

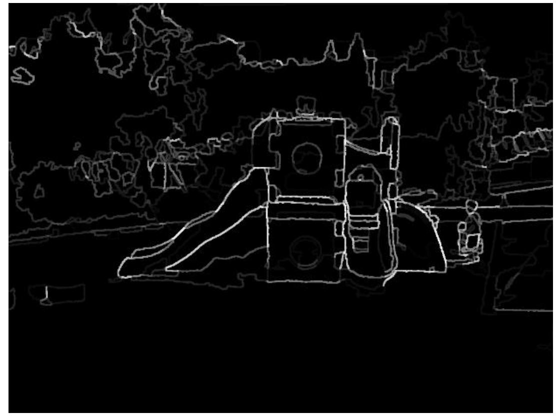




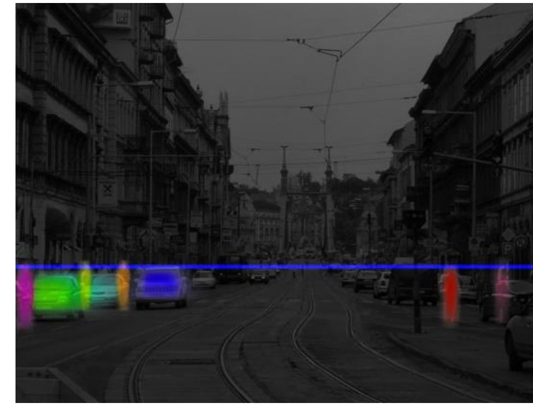
Hoiem et al., IJCV 2011

Occlusion Cues

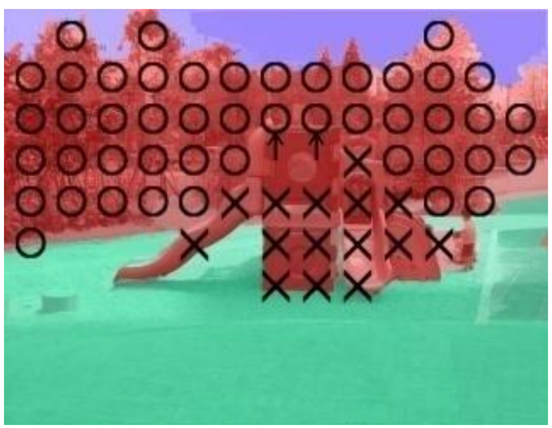
Region
color, position, shape



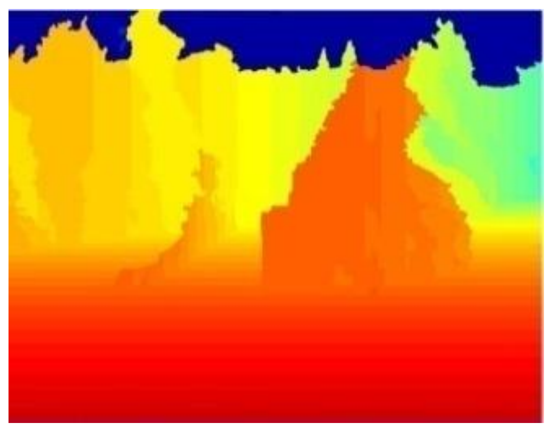
Boundary
strength, length, continuity



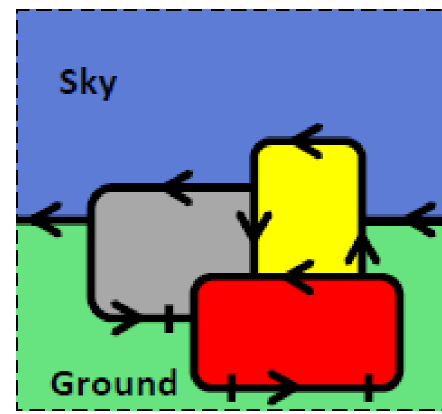
Objects



Surface Layout



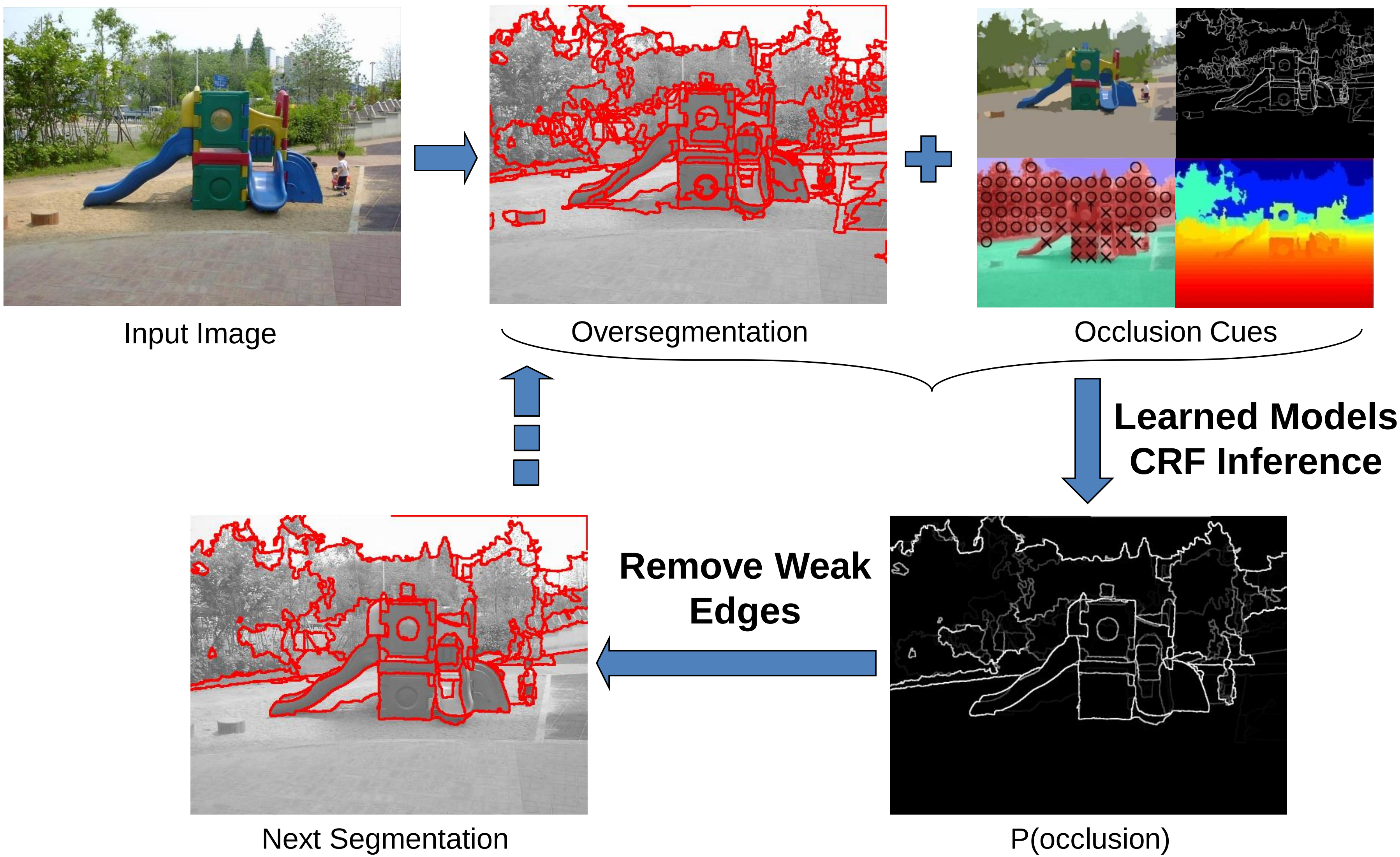
Depth



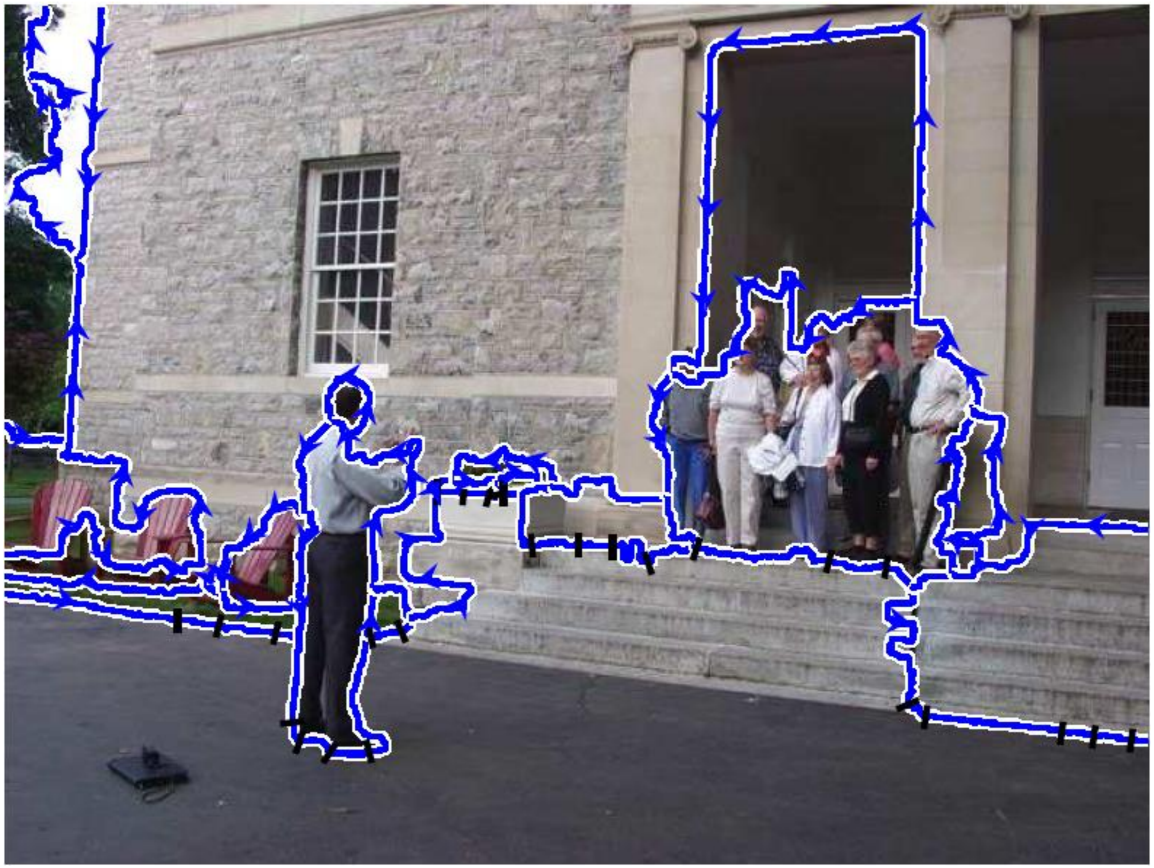
Gestalt Cues
continuity, closure,
valid junctions

Pb: Martin Fowlkes Malik '02

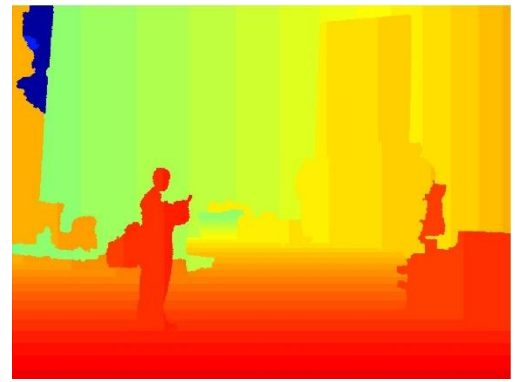
Occlusion Algorithm



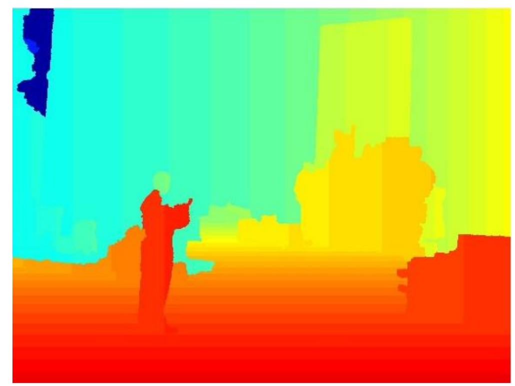
Occlusion Result



Boundaries, Foreground/Background, Contact



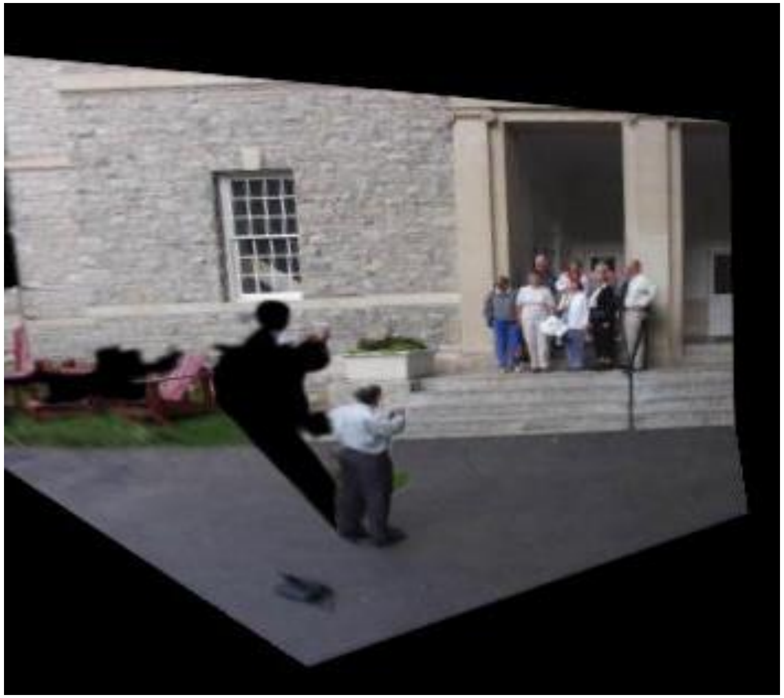
Depth (Min)



Depth (Max)

3D Model with Occlusions

3D Model without
Occlusion Reasoning



3D Model with Occlusion
Reasoning

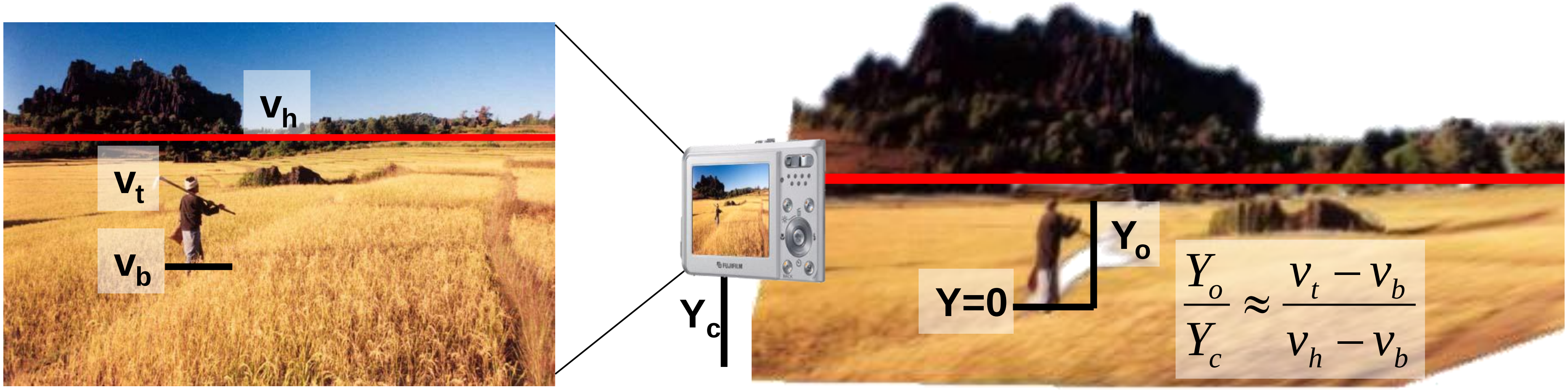
Occlusion boundary map

Input Image

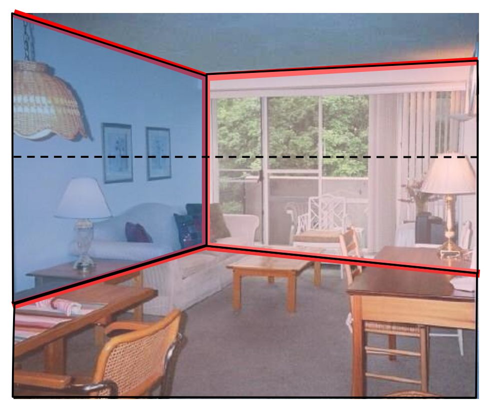
P(occlusion) boundary map

Hoiem et al., IJCV 2011

Viewpoint



Ground plane model:
horizon + height



3D box model:
vanishing points + extent

Viewpoint cues

Vanishing Points

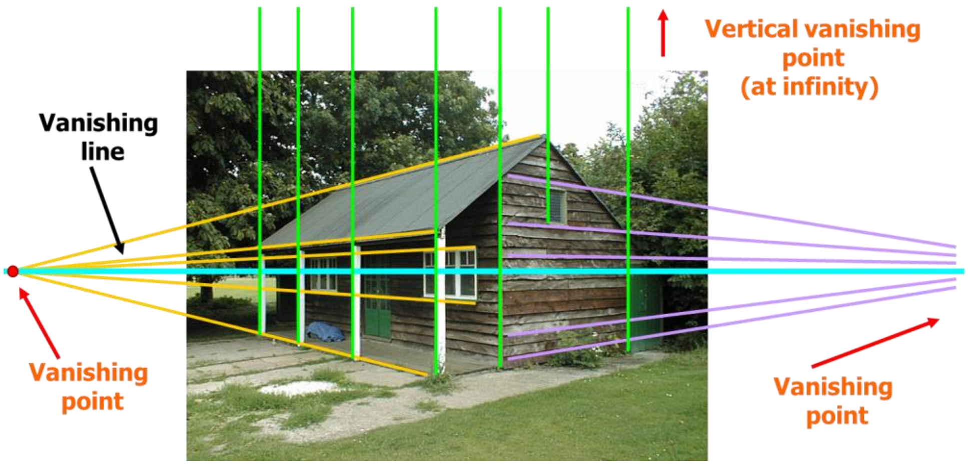
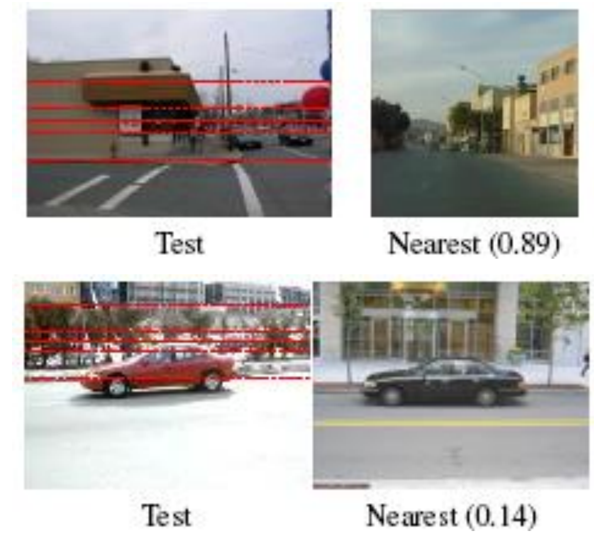
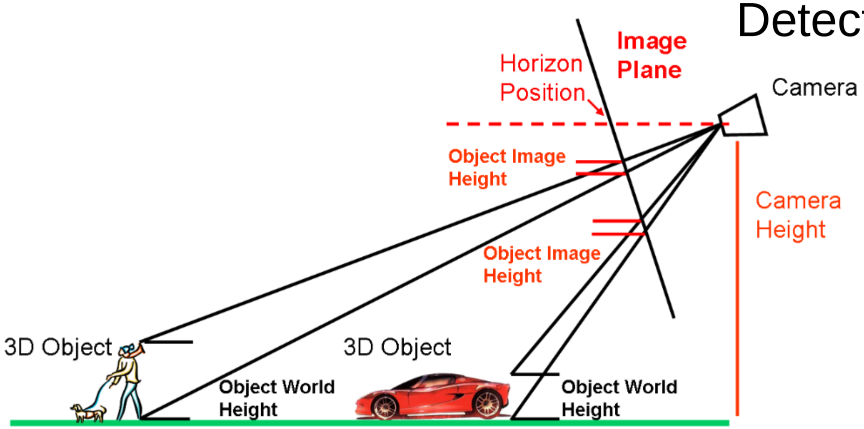
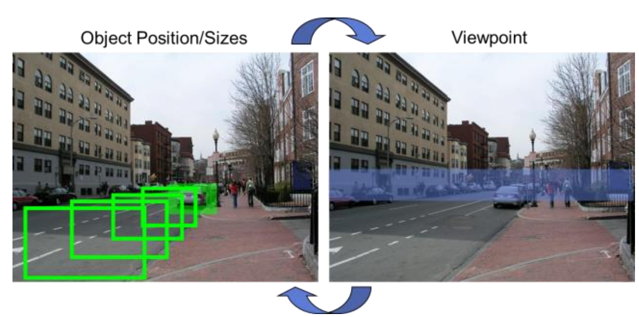


Image Texture / Transfer



Detected Objects





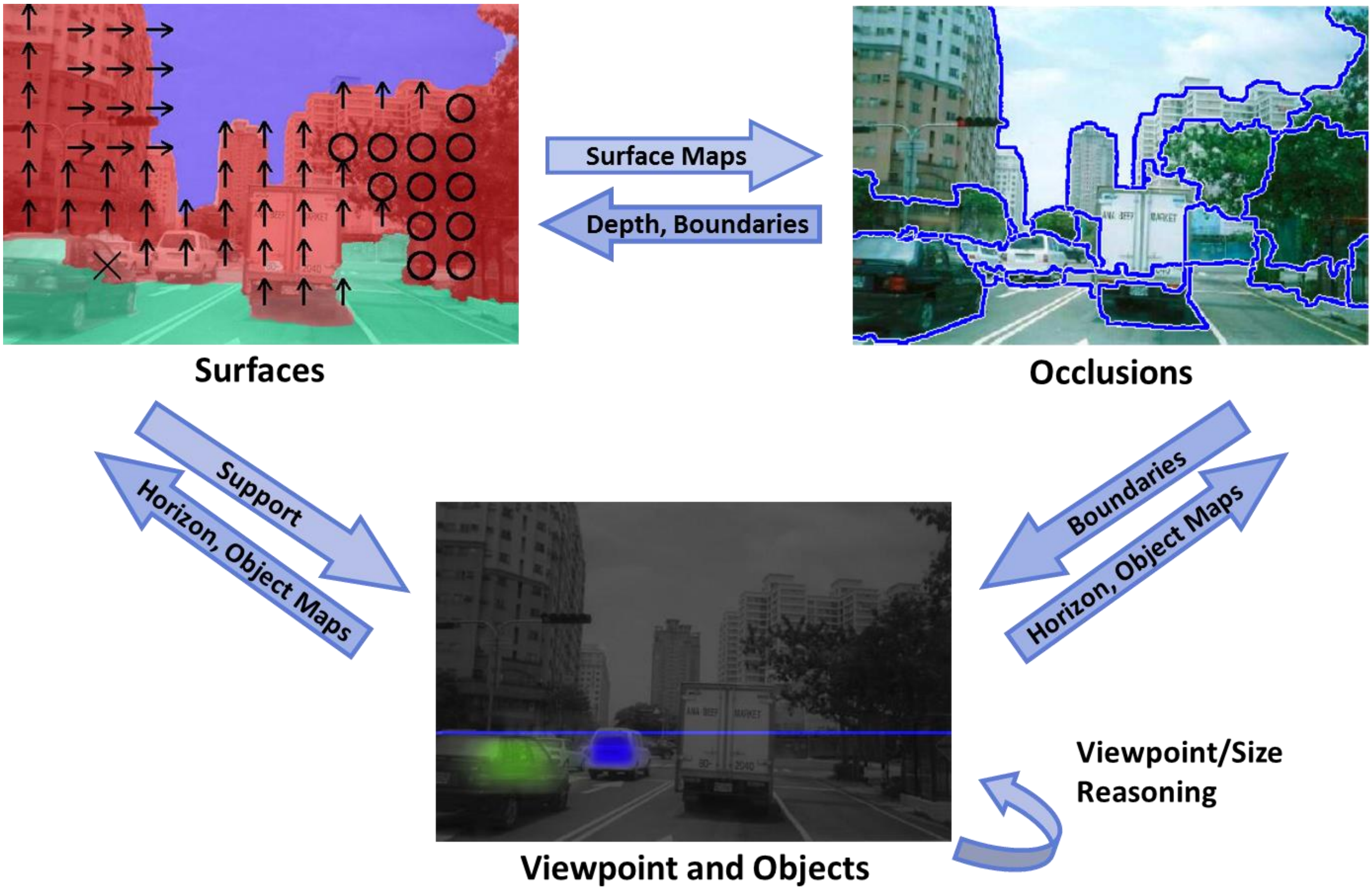
Summary of key points

- Geometric properties can be recovered by modeling (or learning) the structure of the world

- Surface, boundary, and viewpoint properties can be inferred from multiple cues

- Retain uncertain estimates, avoid early decisions

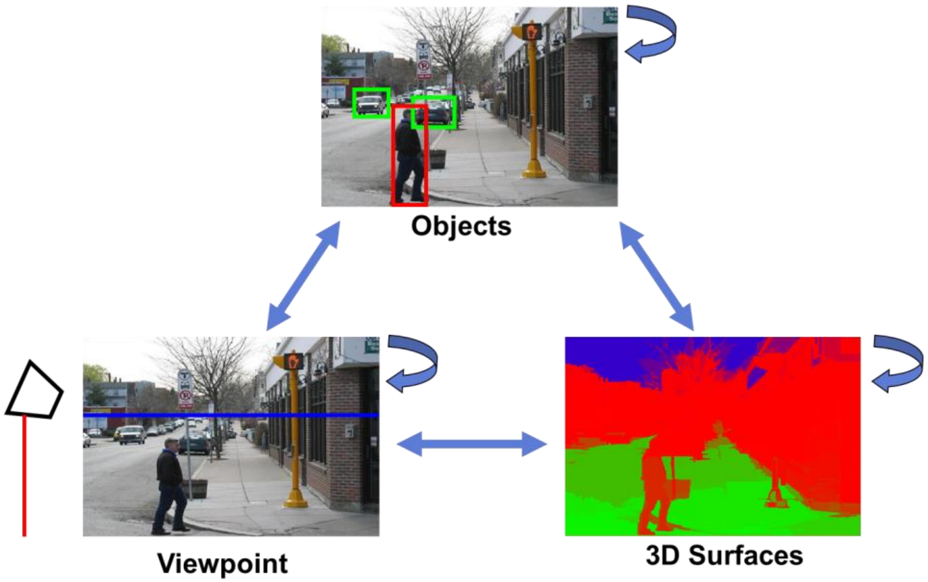
How to interpret scenes as a whole?

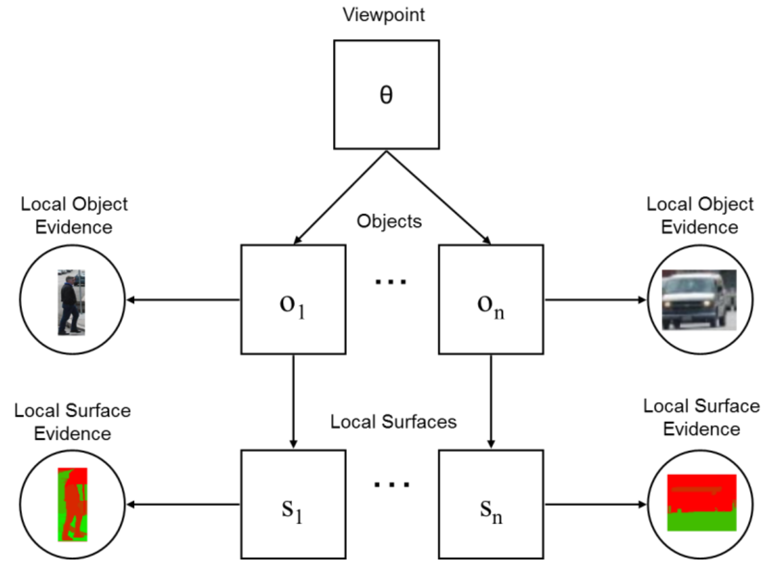


Strategies for structured prediction

- Probabilistic models
- Structured SVM
- Sequential structured prediction

Probabilistic models: example "objects in perspective"



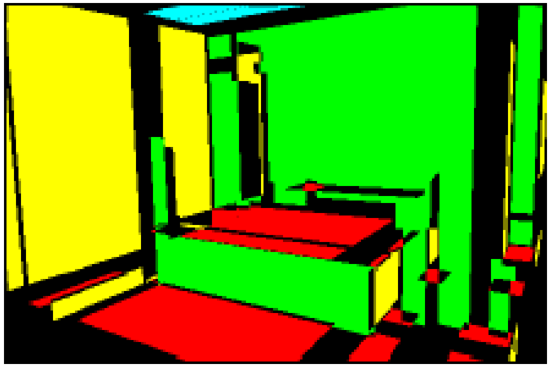


$$P(\theta, \mathbf{o}, \mathbf{g} | \mathbf{e}_g, \mathbf{e}_o) = P(\theta) \prod_i P(o_i | \theta) \frac{P(o_i | \mathbf{e}_{oi})}{P(o_i)} \frac{P(g_i | \mathbf{e}_{gi})}{P(g_i)}$$

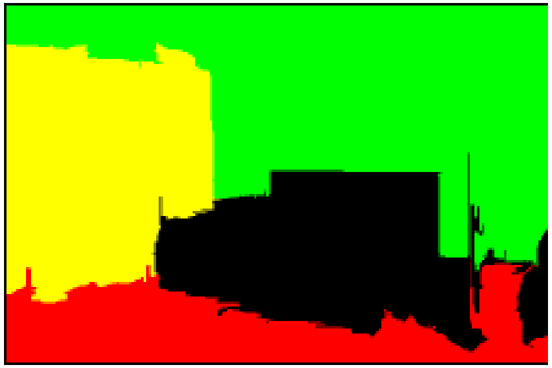
- Use when dependencies are sparse
- When dependencies form a tree, learning and inference are easy and fast
- Most likely and marginal solutions possible (depending on model)

Structured SVM

Example: Fitting a box to a room
(Schwing Urtasun 2012)



Orientation Maps (Lee et al. 2009)

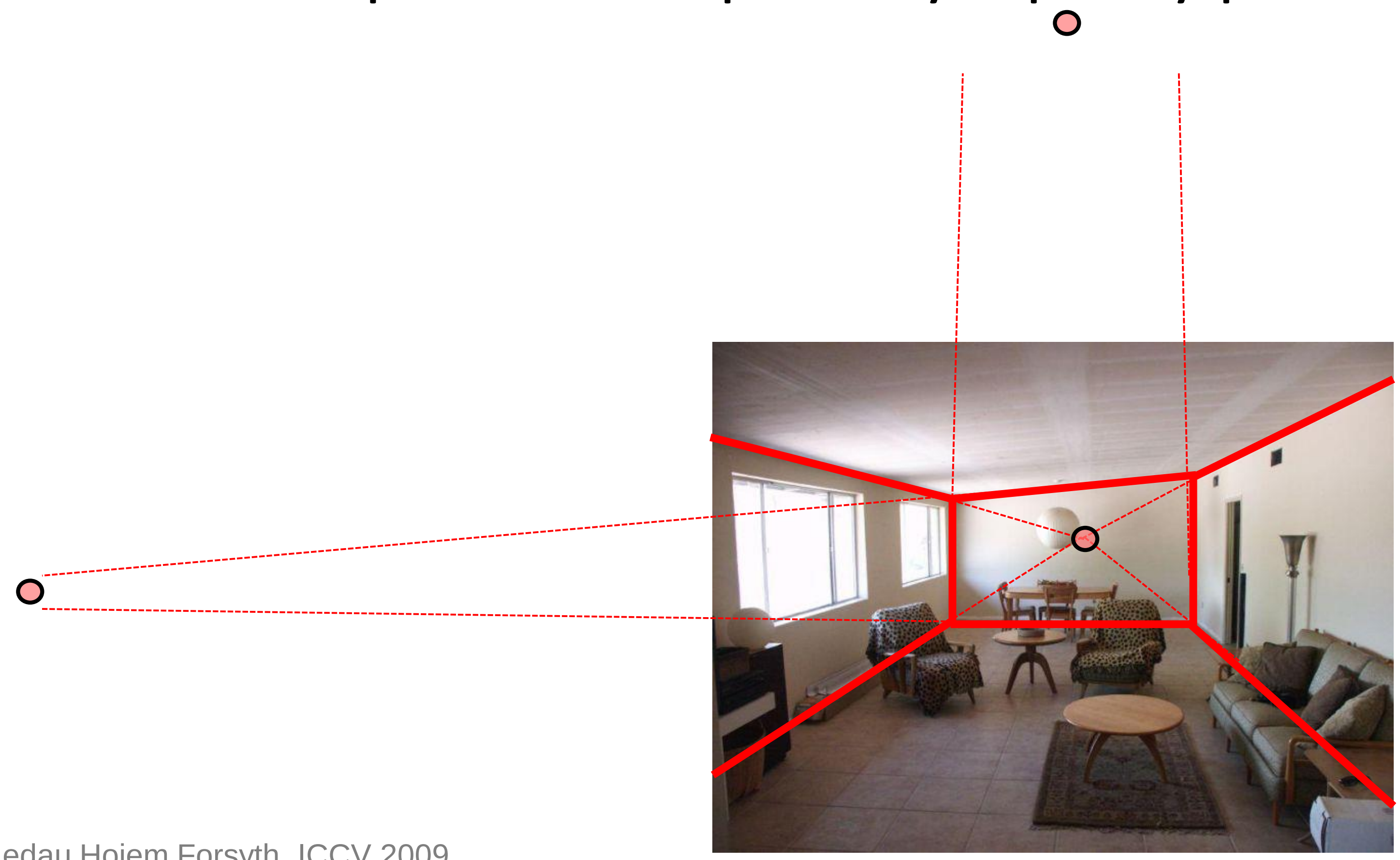


Geometric Context (Hoiem et al. 2007)

Predicted box layout

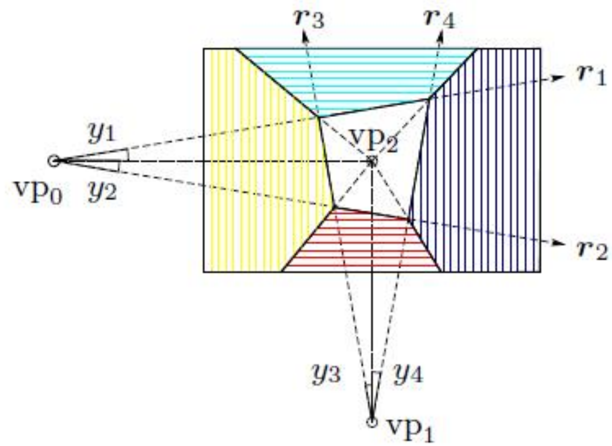
Box Layout Model

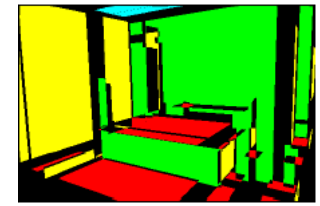
- Room is an oriented 3D box
  - Three vanishing points specify orientation
  - Two pairs of sampled rays specify position/size

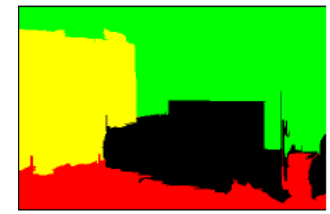


Hedau Hoiem Forsyth, ICCV 2009

SSVM example: fitting a box







Features are sum of predictions in wall/floor/ceiling regions from geometric context and orientation maps

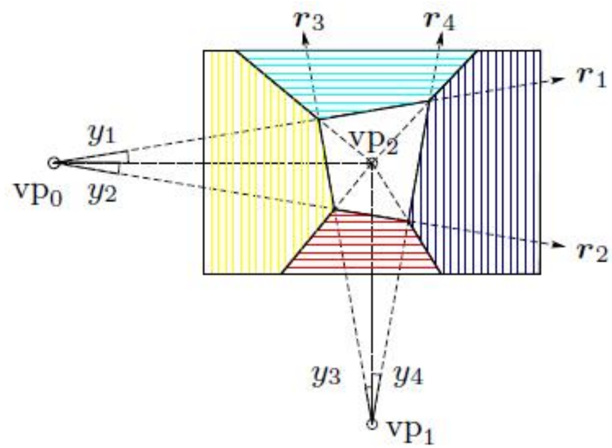
Train weights $\boldsymbol{w}$ to minimize

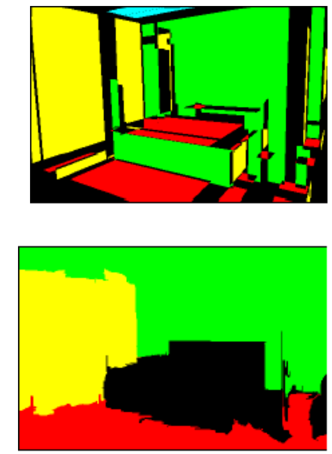
$$\min_{\boldsymbol{w}} \|\boldsymbol{w}\|^2 + C \sum_{n=1}^{l} \max_{y \in Y}\left(\Delta(y_n, y) + \boldsymbol{w}(\psi(x_n, y) - \psi(x_n, y_n)\right)$$

Loss

Margin

SSVM example: fitting a box





- Main idea: correct solution should have higher score than each other solution by a margin of that solution's loss
- Cutting plane training algorithm requires iteratively solving for "most violating constraint", so inference must be fast
- Area-sum features computed quickly with integral geometry
- Inference computed quickly (~10ms) with branch and bound

Structured SVM: comments

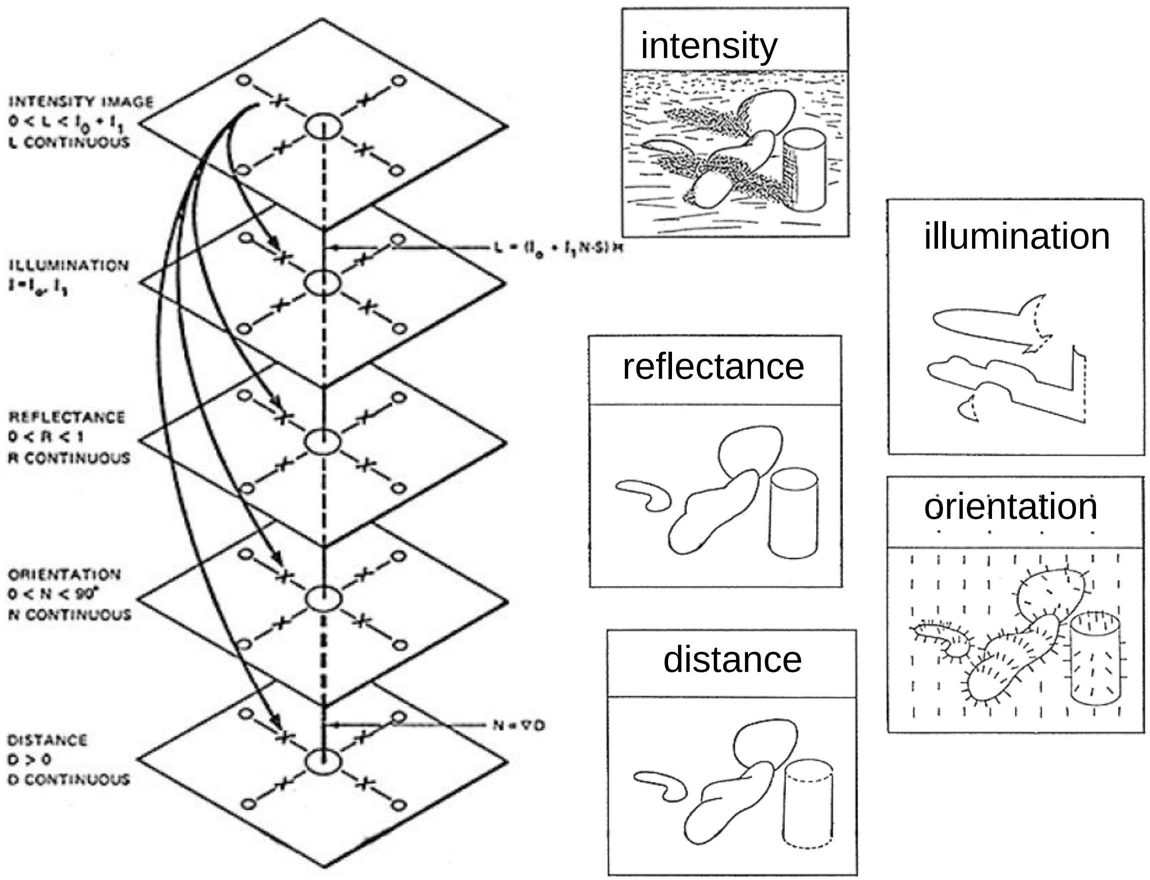
Train weights $\boldsymbol{w}$ to minimize

$$\min_{\boldsymbol{w}} \|\boldsymbol{w}\|^2 + C \sum_{n=1}^{l} \max_{y \in \mathrm{Y}} \big(\underbrace{\Delta(y_n, y)}_{\text{Loss}} + \boldsymbol{w}\underbrace{(\psi(x_n, y) - \psi(x_n, y_n))}_{\text{Margin}} \big)$$

- Most often used when predicted variables have same type (e.g., so single loss makes sense)
- Learning can be difficult when loss is complex (when loss-augmented inference is intractable)
- Often used when single solution is desired (though there are some n-best approaches cf. Batra et al.)

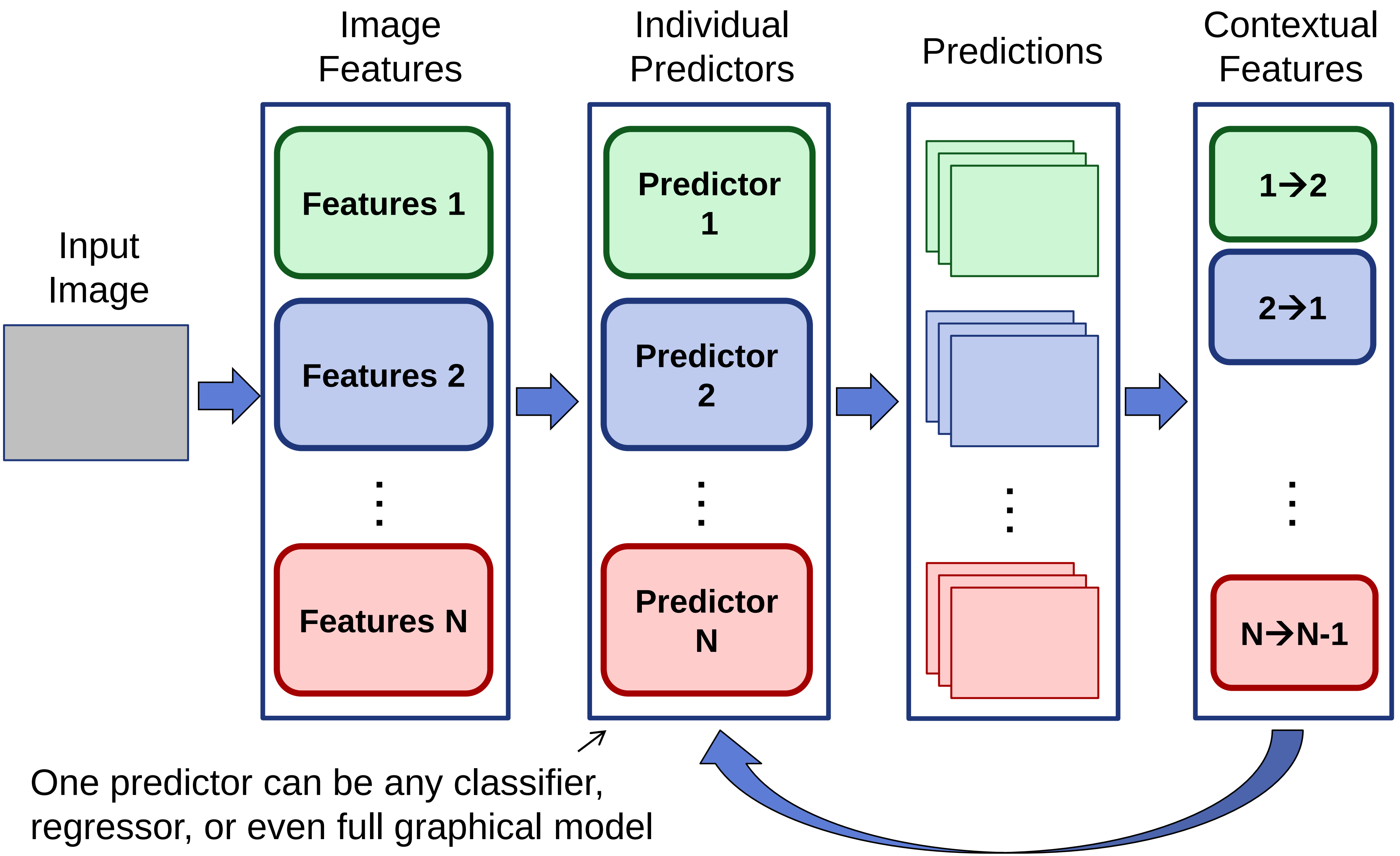
Sequential structured prediction

Iteratively predict each variable based on features and confidences of other variables



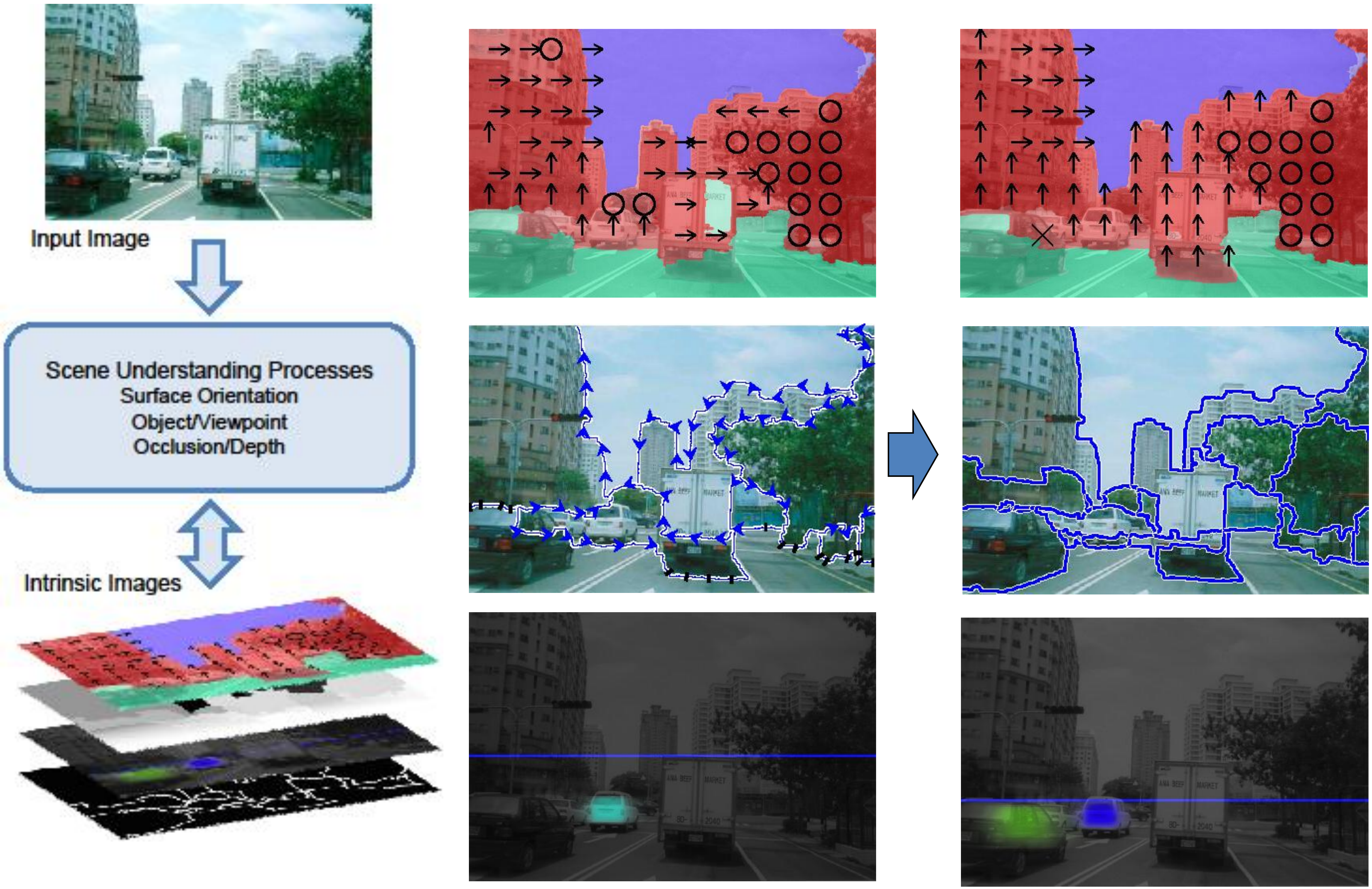
Barrow and Tenenbaum ‘78

Sequential structured prediction



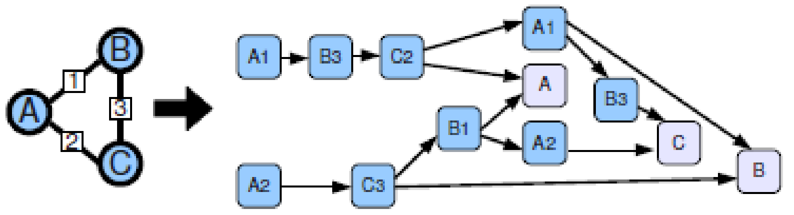
One predictor can be any classifier, regressor, or even full graphical model

Example: reasoning about surfaces, occlusion, objects, and viewpoint



Hoiem Efros Hebert 2008

Sequential prediction as belief propagation



“unrolled BP”
(Ross et al. 2011)

- Sequential prediction updates each prediction based on marginals of related predictions
  - New update function learned for each iteration (typically)
  - Classifier encodes complex functions of many variables
  - Each iteration improves likelihood of training predictions
  - Can provide guarantees on prediction loss

Tu Bai 2010: Auto-context
Ross Munoz Hebert Bagnell 2011: Message-Passing Inference Machines

What strategy to use?

- Graphical model (probabilistic or energy/SVM)
  - Dependencies are sparse and easy to model
  - Single loss or probability function makes sense
  - Want an explicit global objective function
- Sequential prediction
  - Dependencies are dense and/or complex
  - Need to make multiple predictions with different loss functions

Big open problems

- Modeling uncertainty in complex scene representations
- Developing approaches that easily adapt to different input sensors
- Cumulative scene understanding over long observations

Questions?

- Next up
  - Abhinav Gupta on “Volumetric and Functional Constraints”
  - David Fouhey on “Non-parametric approaches to 3D scene understanding”