Learning and Inference in Low Level Vision

Yair Weiss[1,2]

[1]Hebrew University of Jerusalem, [2]Canadian Institute for Advanced Research

Understanding Low-Level Vision in Human and Machine Perception

Anat Levin

Talya Meltzer

Chen Yanover

Ce Liu

Bill Freeman

Ted Adelson

Low-Level vs High Level Vision

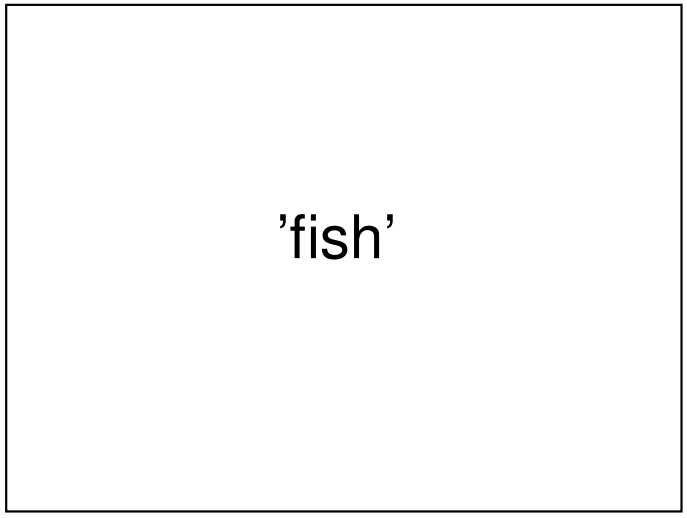
Input

Output

Low-Level vs High Level Vision

Input

Output



Low-Level vs High Level Vision

Input

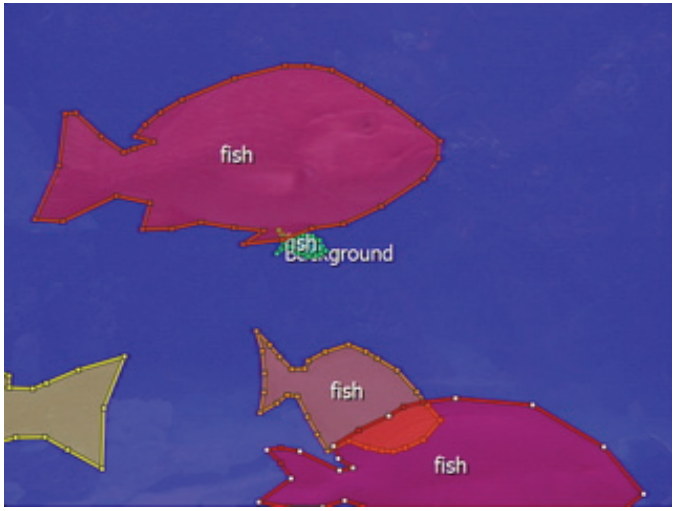
Output

Low Level Vision seeks a label at every pixel (e.g. depth, motion, figure/ground).

Low-Level vs High Level Vision

Input

Output

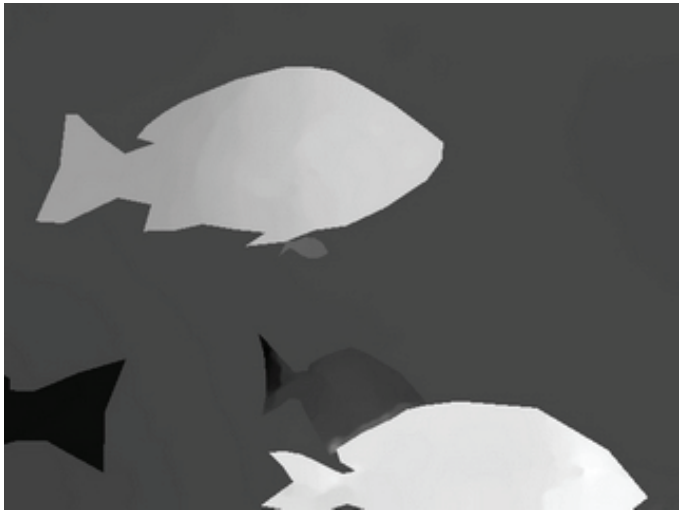


Low Level Vision seeks a label at every pixel (e.g. depth, motion, figure/ground).

Low-Level vs High Level Vision

Input

Output

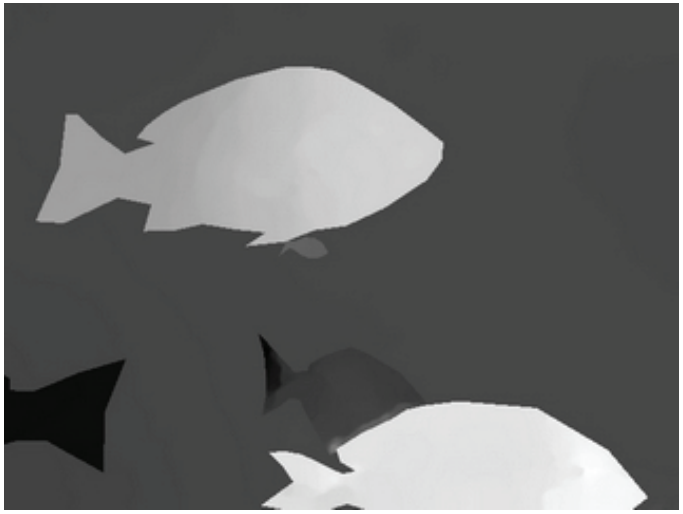


Millions of (continuous) labels vs. one discrete label.

Low-Level vs High Level Vision

Input

Output



Millions of (continuous) labels vs. one discrete label.

Outline

- Low-Level Vision in Humans
- Inference and Stereo Vision
- Learning and Segmentation
- Old vs. New Directions.

Narrow vs Fat

Perceived motion of a rotating ellipse depends on its shape (Wallach 1935,Musatti 1924)

Satellites

The ellipse is always doing the same thing. (Weiss and Adelson, 96)

Distant Satellites

Effect persists at large distances.

Competition

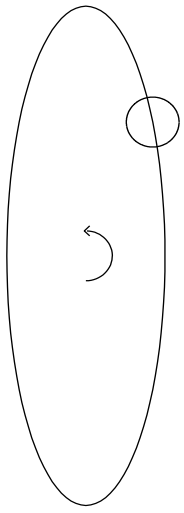
Ellipse is always doing the same thing. Closer satellites win.

Layers

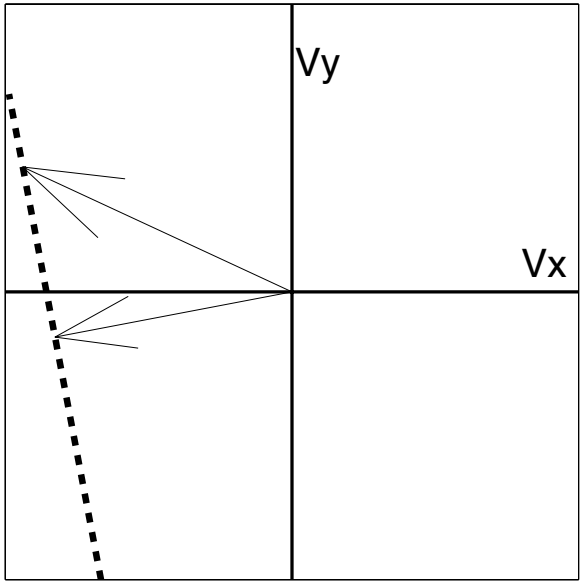
Ellipse separated from vertical dots but effected by the rotating/deforming dots.

Motion Perception is ill-posed

Local Aperture



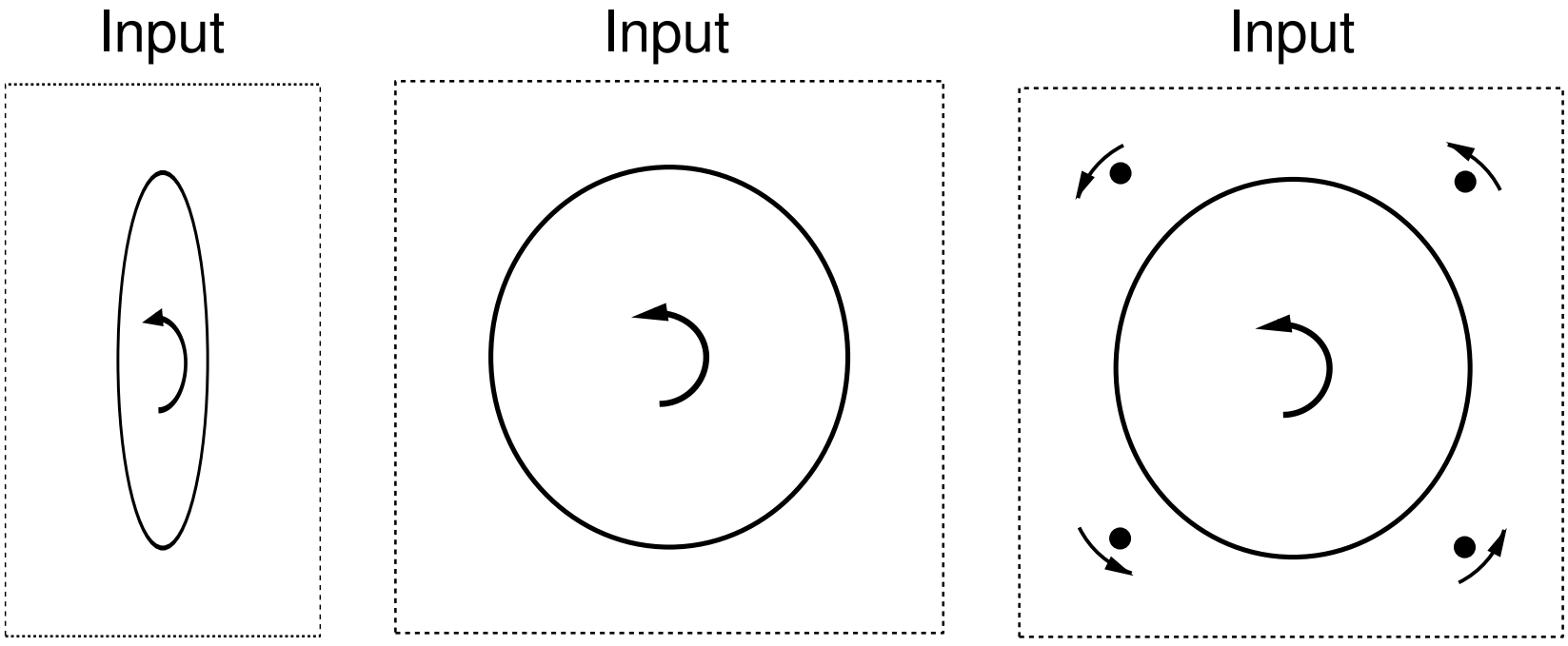
Data Term



$$E(v) = \sum_i (v_i^T a_i - b_i)^2 + \lambda \sum_{<ij>} \|v_i - v_j\|^2$$

(Hildreth 1983, Horn and Schunck 81, Grzywacz and Yuille 91)

Energy Minimization Surprisingly Predictive

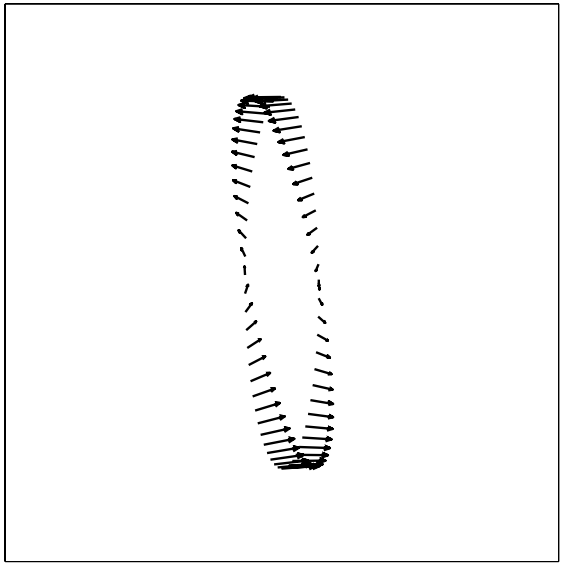


$$E(v) = \sum_i (v_i^T a_i - b_i)^2 + \lambda \sum_{<ij>} \|v_i - v_j\|^2$$

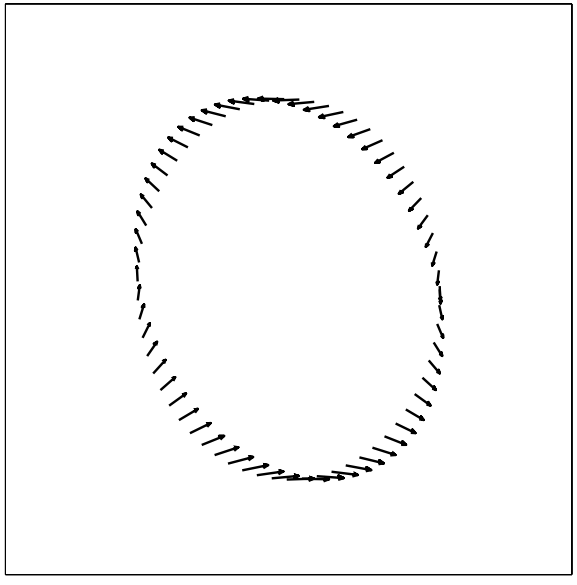
(Hildreth 1983, Horn and Schunck 81, Grzywacz and Yuille 91)

Energy Minimization Predictions

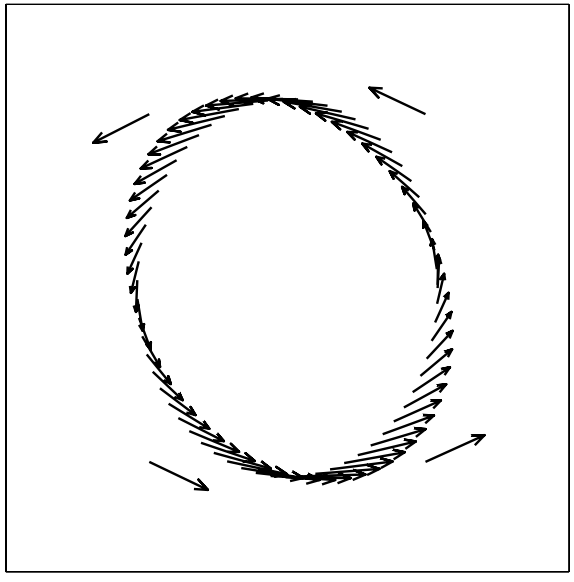
Output



Output



Output

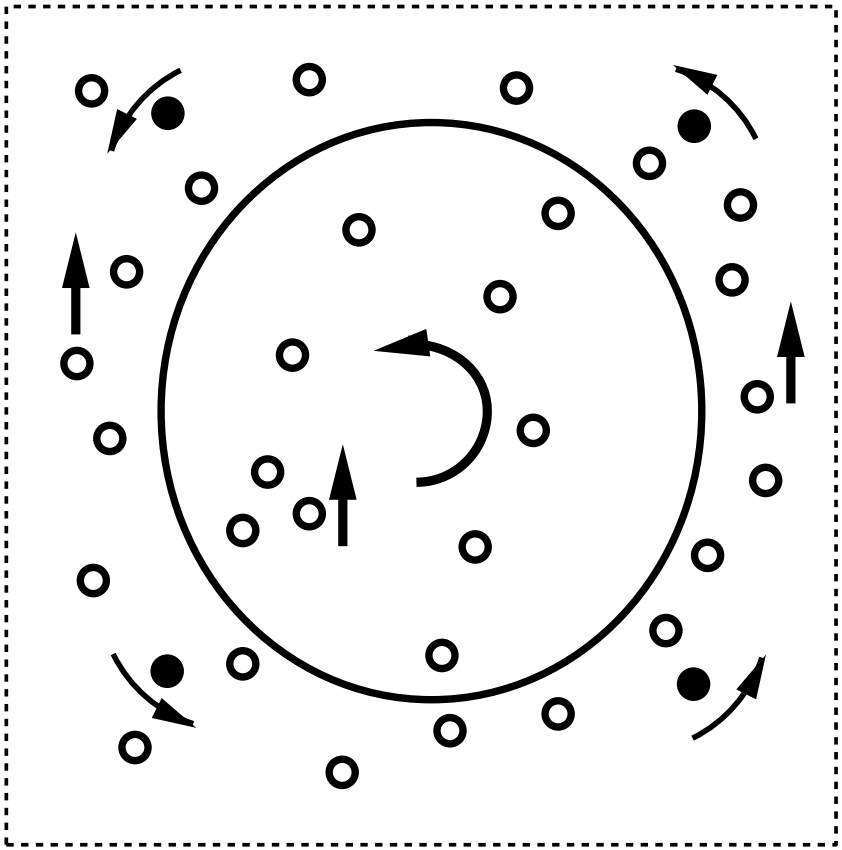


$$E(v) = \sum_i (v_i^T a_i - b_i)^2 + \lambda \sum_{<ij>} \|v_i - v_j\|^2$$

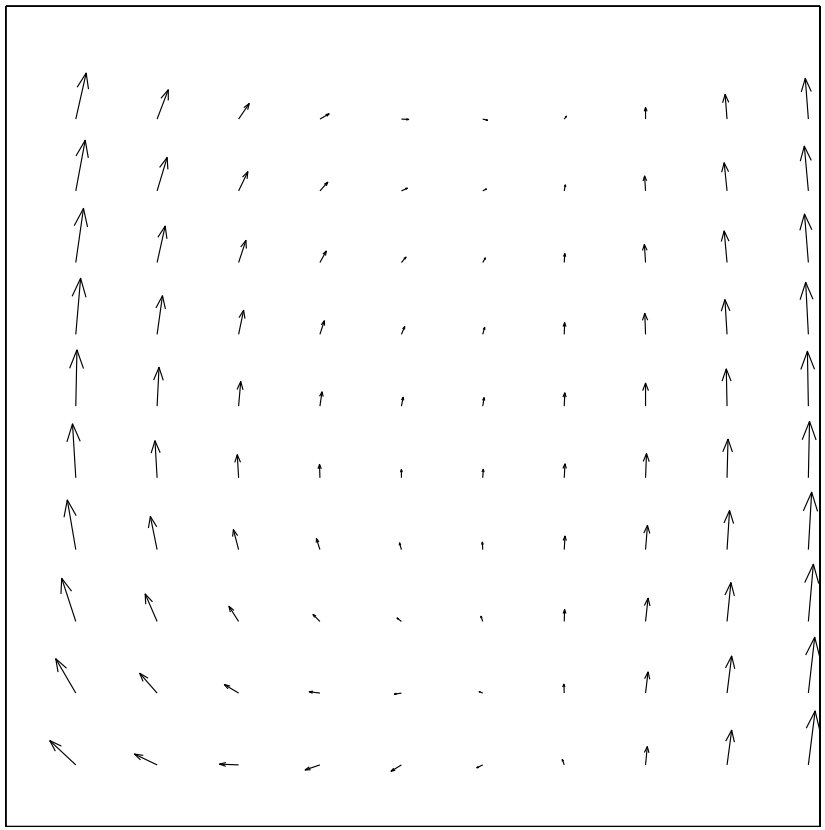
(Hildreth 1983, Horn and Schunck 81, Grzywacz and Yuille 91)

Energy Minimization Failures

Input



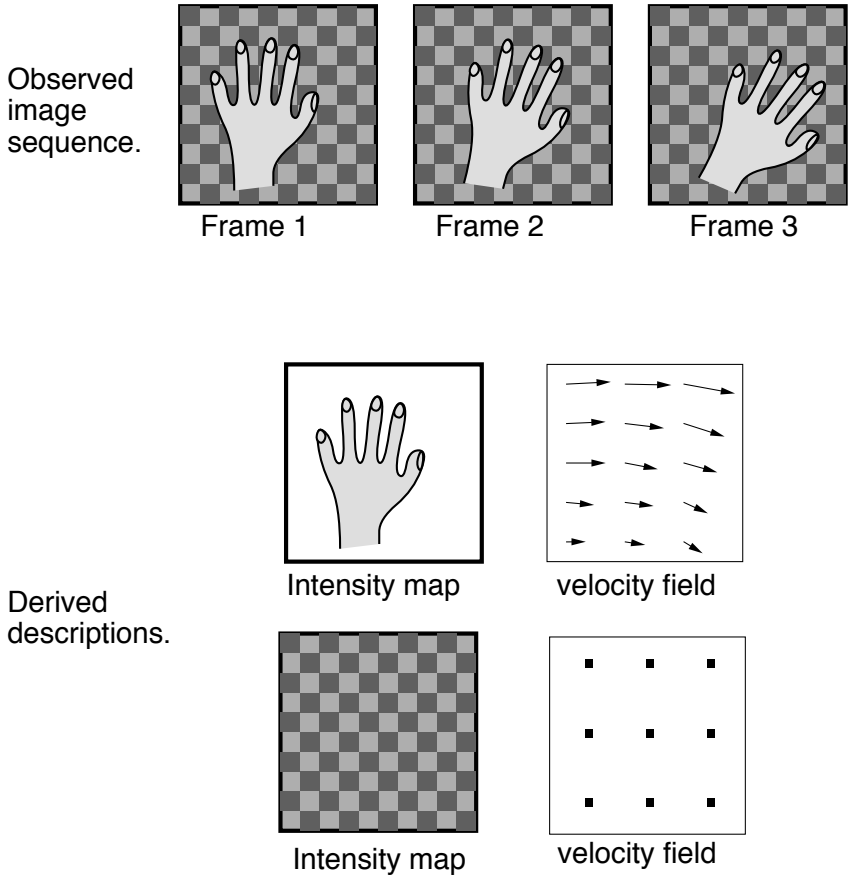
Output



$$E(v) = \sum_i (v_i^T a_i - b_i)^2 + \lambda \sum_{<ij>} \|v_i - v_j\|^2$$

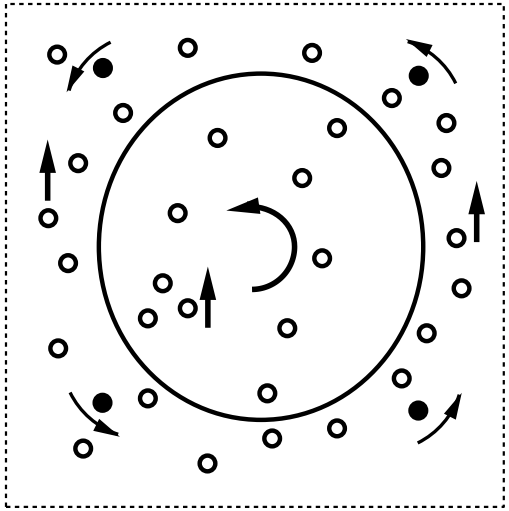
(Hildreth 1983, Horn and Schunck 81, Grzywacz and Yuille 91)

Layered Models (Wang and Adelson 93)



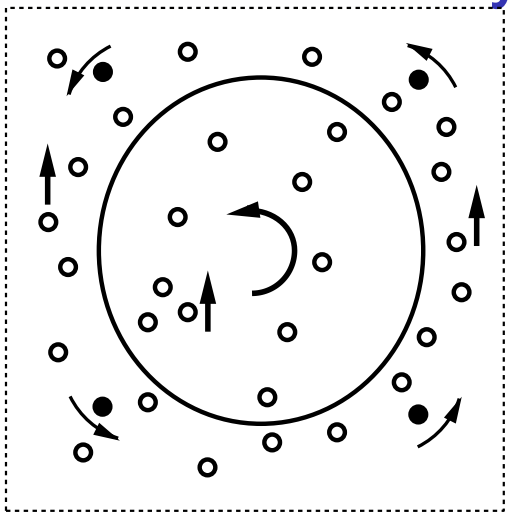
Decompose the scene into multiple layers. Calculate velocity of each layer, assignment of pixels to layers and number of layers. (Ayer and Sawhney 95, Jepson and Black 95 $\cdots$)

Energy Minimization and Layered Models



$$
\begin{aligned}
E(v_1, v_2, g_1, g_2) &= dataterm(v_1, v_2, g_1, g_2) \\
&= +\lambda\,(smooth(v_1) + smooth(v_2)) \\
&= +\lambda_2\,(smooth(g_1) + smooth(g_2)) \\
&= +\lambda_3 complexity(g)
\end{aligned}
$$

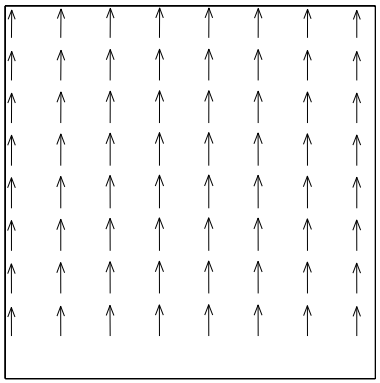
Energy Minimization and Layered Models



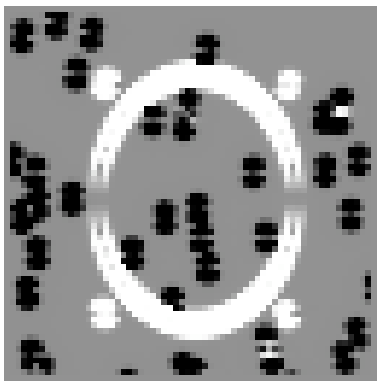
$$
\begin{aligned}
E(v_1, v_2, g) &= \sum_i g_{1i}(v_{1i}^T a_i - b_i)^2 + g_{2i}(v_{2i}^T a_i - b_i)^2 \\
&= +\lambda \sum_{<ij>} \|v_{1i} - v_{1j}\|^2 + \|v_{2i} - v_{2j}\|^2 \\
&= +\lambda_2 \sum_{<ij>} (g_{1i} - g_{1j})^2 + (g_{2i} - g_{2j})^2 \\
&= +\lambda_3 C(g)
\end{aligned}
$$

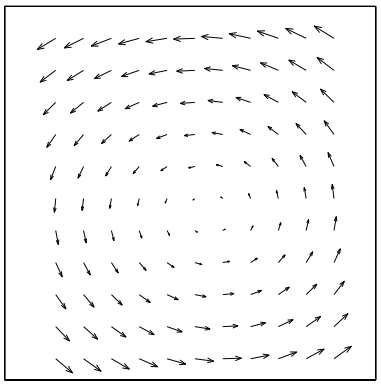
Layered Energy Minimization Models Predict Many Percepts

computed layer 1 (g_1, v_1):



computed layer2 (g_2, v_2):





"Correctly" estimates number of layers, motions and assignments. (Weiss and Adelson 98)

Layered Models for Real Sequences ?

$$
\begin{aligned}
E(v_1, v_2, g_1, g_2) &= dataterm(v_1, v_2, g_1, g_2) \\
&= +\lambda\,(smooth(v_1) + smooth(v_2)) \\
&= +\lambda_2\,(smooth(g_1) + smooth(g_2)) \\
&= +\lambda_3 complexity(g)
\end{aligned}
$$

Layered Models for Real Sequences

$$
\begin{aligned}
E(v_1, v_2, g_1, g_2) &= dataterm(v_1, v_2, g_1, g_2) \\
&= +\lambda\,(smooth(v_1) + smooth(v_2)) \\
&= +\lambda_2\,(smooth(g_1) + smooth(g_2)) \\
&= +\lambda_3 complexity(g)
\end{aligned}
$$

Can we learn the energy function?

Layered Models for Real Sequences

$$
\begin{aligned}
E(v_1, v_2, g_1, g_2) &= dataterm(v_1, v_2, g_1, g_2) \\
&= +\lambda\,(smooth(v_1) + smooth(v_2)) \\
&= +\lambda_2(smooth(g_1) + smooth(g_2)) \\
&= +\lambda_3 complexity(g)
\end{aligned}
$$

Can we optimize the energy function?

The Dream part 1

Training set:

Learning Algorithm $\Rightarrow E(v_1, v_2, g_1, g_2)$

The Dream part 2

Test sequence:

Optimization Algorithm $\Rightarrow v_1^*, v_2^*, g_1^*, g_2^*$

How to get a Ph.D. in Mathematics (according to my brother)

- Advisor proposes open problem.

How to get a Ph.D. in Mathematics (according to my brother)

- Advisor proposes open problem.
- Student fails to solve problem. Proves lemmas along the way.

How to get a Ph.D. in Mathematics (according to my brother)

- Advisor proposes open problem.
- Student fails to solve problem. Proves lemmas along the way.
- Ph.D. thesis describes the lemmas.

“Lemmas along the way”

1. For many real-world low-level vision problems, we can efficiently find the global optimum of the energy function.
2. For many real-world low-level vision problems, we can efficiently learn energy functions.

(Meltzer et al. 2005, Levin and Weiss 2006, Sontag et al 2008, Kolmogorov and Wainwright 2005, Werner 2007, Rother et al. 2007, Boros and Zabih 2007, Komodakis and Paragios 2007, Woodford et al. 2009, Kohli and Torr 2008, Kumar and Torr 2009, He and Zemel 2008, Scharstein and Pal 2008, Lan, Roth, Huttenlocher and Black 2007 . . . .)

Global Optimization in Stereo Vision

Input:

Output: $\{x_i\}$ horizontal disparity at pixel i.

Stereo by Energy Minimization

Input:

$$E(x) = \sum_i E_i(x_i) + \sum_{<ij>} E_{ij}(x_i, x_j)$$

data term $E_i(x_i)$ and smoothness term $E_{ij}(x_i, x_j)$ are non-quadratic. Optimization is NP Hard (Boykov et al. 04)

Local Optima in Stereo Vision

Input

SSD based
(early 80s)

Energy Minimization
(mid 90s)

Better energy functions or better optimization ?

Integer Programming formulation

minimize:

$$J(\{q\}) = \sum_{i,j} \sum_{x_i,x_j} q_{ij}(x_i, x_j) E_{ij}(x_i, x_j) + \sum_{i} \sum_{x_i} q_i(x_i) E_i(x_i)$$

subject to:

$$\begin{aligned} q_{ij}(x_i, x_j) &\in \{0, 1\} \\ \sum_{x_i,x_j} q_{ij}(x_i, x_j) &= 1 \\ \sum_{x_i} q_{ij}(x_i, x_j) &= q_j(x_j) \end{aligned}$$

Linear Programming relaxation

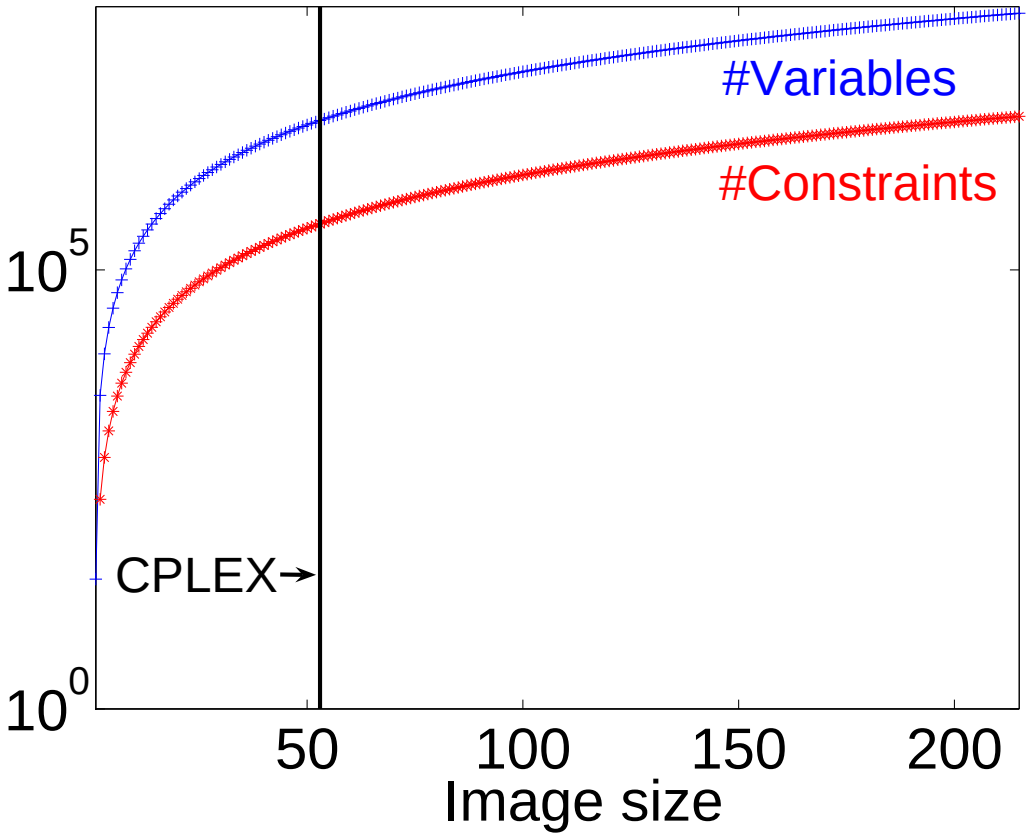
minimize:

$$J(\{q\}) = \sum_{i,j} \sum_{x_i, x_j} q_{ij}(x_i, x_j) E_{ij}(x_i, x_j) + \sum_{i} \sum_{x_i} q_i(x_i) E_i(x_i)$$

subject to:

$$\begin{aligned} q_{ij}(x_i, x_j) &\in [0, 1] \\ \sum_{x_i, x_j} q_{ij}(x_i, x_j) &= 1 \\ \sum_{x_i} q_{ij}(x_i, x_j) &= q_j(x_j) \end{aligned}$$

Off-the-shelf LP Solver?



- Millions of variables and constraints.
- Usually LP solutions fractional

LP using Belief Propagation

- Input: $\{\Psi_{ij} \propto e^{-E_{ij}}\}$
- Output: beliefs $\{b_{ij}\}$, $\{b_i\}$

Iterate:

$$m_{ij}(x_j) \leftarrow \max_{x_i} \prod_{k \neq j} m_{ki}^{\rho}(x_i) m_{ji}^{\rho-1}(x_i) \Psi_{ij}^{\rho}(x_i, x_j)$$

- $\rho = 1$,standard BP.
- $\rho < 1$ "Tree-Reweighted/Fractional/Convex" BP.

(Wainwright et al. 03, Wiegerink and Heskes 03, Weiss et al. 07)

Global Optimality

Calculate beliefs from messages:

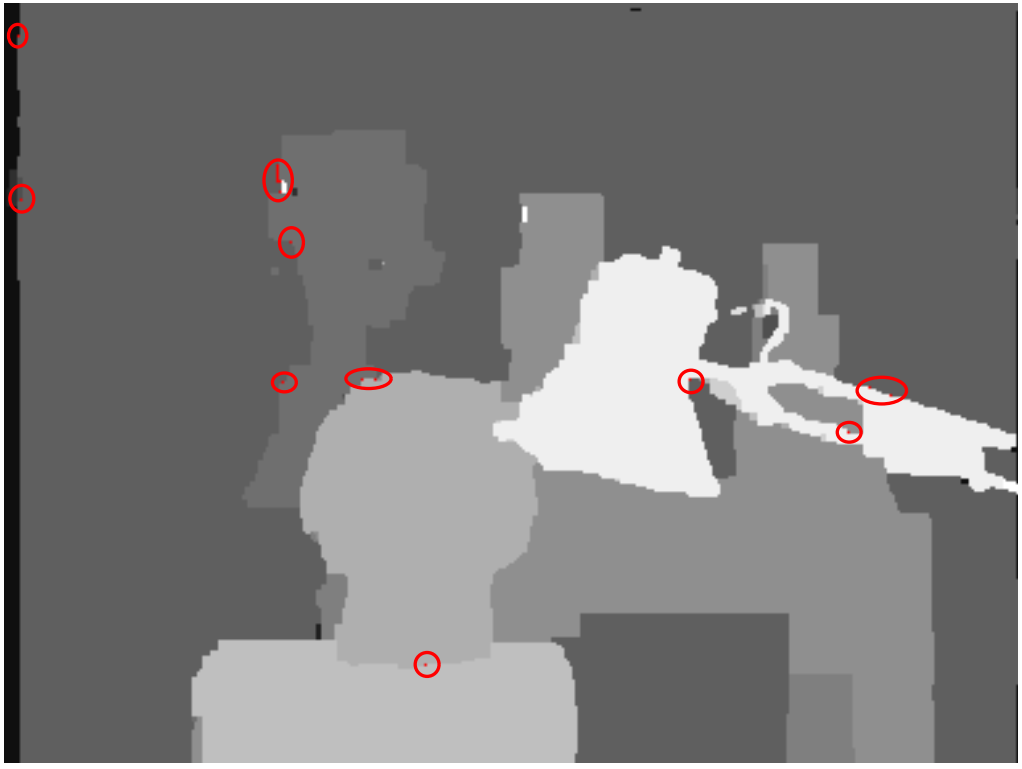
$$b_i(x_i) = \prod_j m_{ji}(x_i)$$

Ties	No Ties
$(0.4, 0.4, 0.2)$	$(0.5, 0.3, 0.2)$

Claim: For convex-BP: if the beliefs have no ties, locally maximizing the beliefs gives the *global* optimum of the energy.
Proof: equivalent to solving LP relaxation.

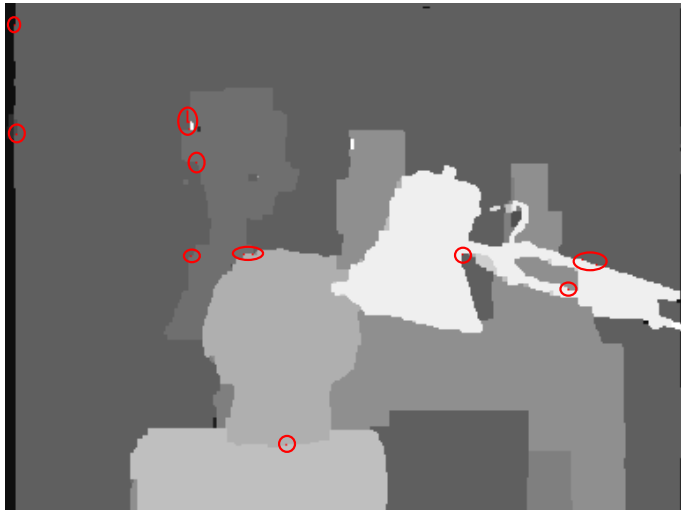
(Wainwright et al. 2003, Weiss et al. 07)

LP using Belief Propagation



- Good News: can solve full images. Faster than CPLEX on small images.
- Bad News: solution has small number of ties.

Global Optimality With Ties



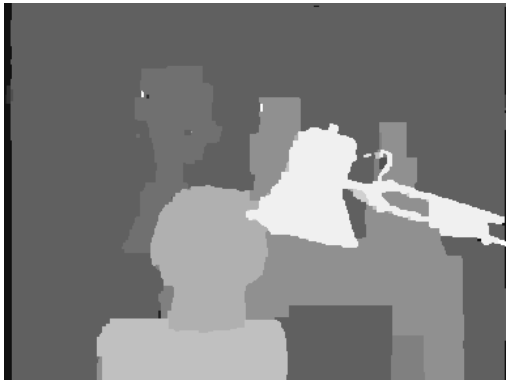
- When tied subgraph is a tree
- When beliefs on boundary of fractional subgraph are uniform
- When MAP of tied subgraph does not contradict BP beliefs

(Kolmogorov and Wainwright 05,Weiss et al. 07)

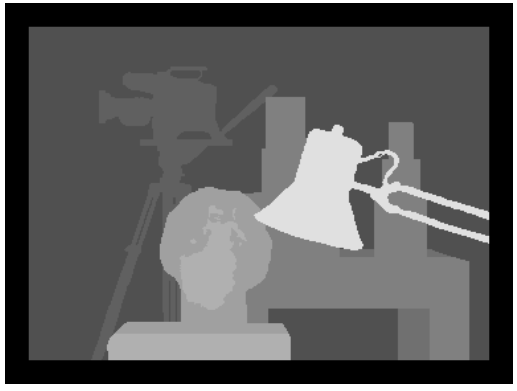
Global Optima in Stereo Vision

Input

Global Optimum



Ground Truth



For many benchmark images:

- Good News: Global optimum in approx 10 minutes.
- Bad News: Not much better than local optimum.

“Lemmas along the way”

1. For many real-world low-level vision problems, we can efficiently find the global optimum of the energy function. □
2. For many real-world low-level vision problems, we can efficiently learn energy functions.

(Meltzer et al. 2005, Levin and Weiss 2006, Sontag et al 2008, Kolmogorov and Wainwright 2005, Werner 2007, Rother et al. 2007, Boros and Zabih 2007, Komodakis and Paragios 2007, Woodford et al. 2009, Kohli and Torr 2008, Kumar and Torr 2009, He and Zemel 2008, Scharstein and Pal 2008, Lan, Roth, Huttenlocher and Black 2007 . . . .)

Learning Energy Functions for Category-Specific Segmentation

Training Set:

. . . . . .

Novel Input:

(Borenstein and Ullman, 2002)

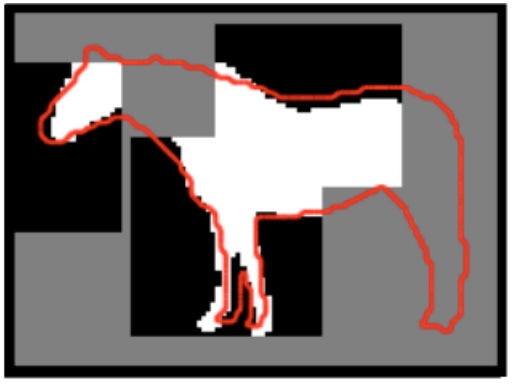
Energy Functions for Category-Specific Segmentation

Input

$$x^* = \arg\min_x \sum_i E_i(x_i) + \sum_{<ij>} E_{ij}(x_i, x_j)$$

Pairwise term

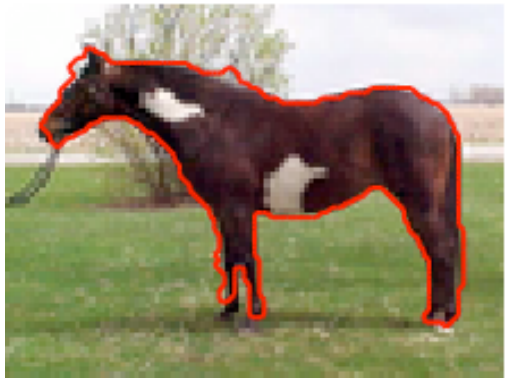
Data Term



Energy Functions for Category-Specific Segmentation

Input

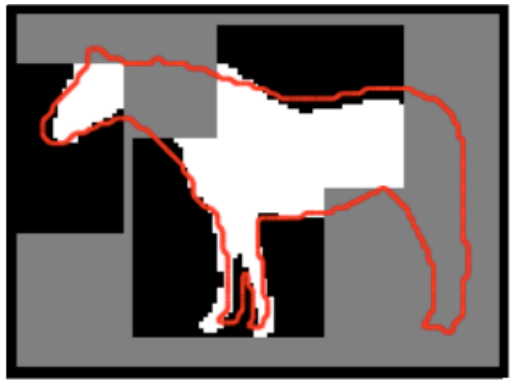
Output



$$x^* = \arg\min_x \sum_i E_i(x_i) + \sum_{<ij>} E_{ij}(x_i, x_j)$$

Pairwise term

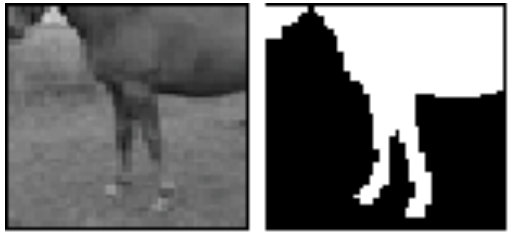
Data Term



Constructing the Data Term

Input

Fragment



$$x^* = \arg\min_x \sum_i E_i(x_i) + \sum_{<ij>} E_{ij}(x_i, x_j)$$

Search Space

Data Term

Constructing the Data Term

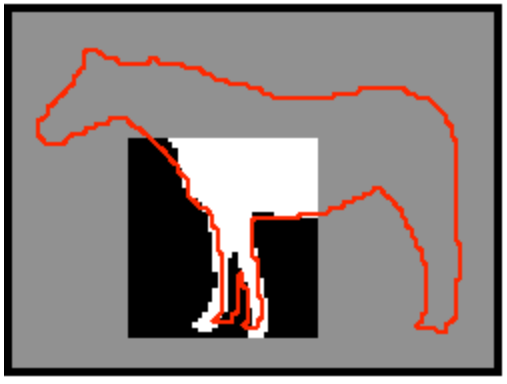
Input

Fragment

$$x^* = \arg\min_x \sum_i E_i(x_i) + \sum_{<ij>} E_{ij}(x_i, x_j)$$

Search Space

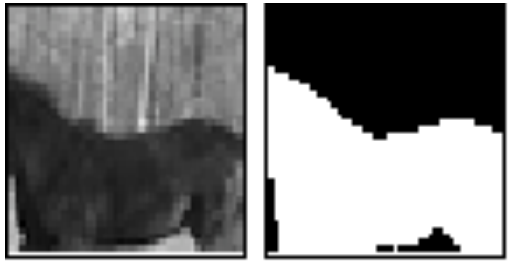
Data Term



Constructing the Data Term

Input

Fragment



$$x^* = \arg\min_x \sum_i E_i(x_i) + \sum_{<ij>} E_{ij}(x_i, x_j)$$

Search Space

Data Term

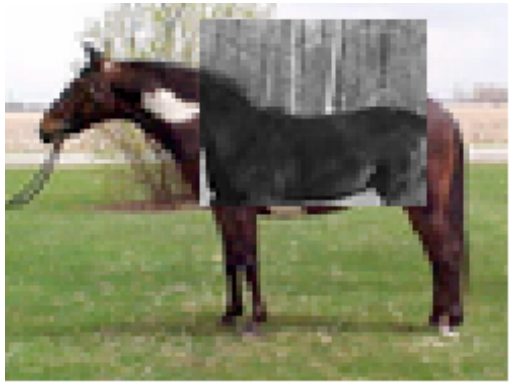
Constructing the Data Term

Input

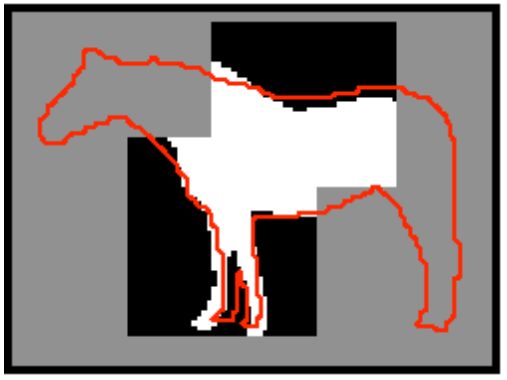
Fragment

$$x^* = \arg\min_x \sum_i E_i(x_i) + \sum_{<ij>} E_{ij}(x_i, x_j)$$

Search Space



Data Term



Constructing the Data Term

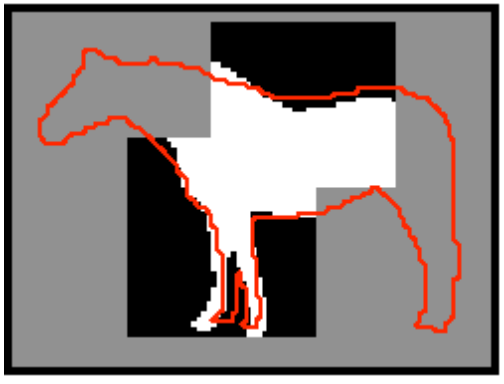
Input

Fragment

$$x^* = \arg\min_x \sum_i E_i(x_i) + \sum_{<ij>} E_{ij}(x_i, x_j)$$

Search Space

Data Term



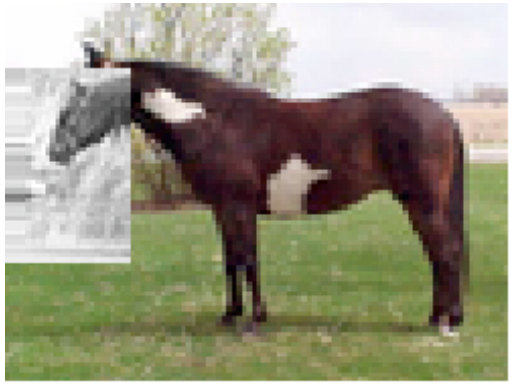
Constructing the Data Term

Input

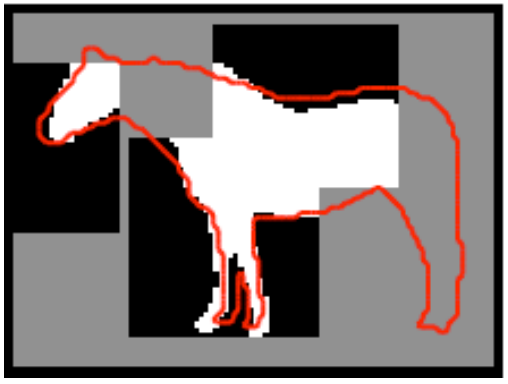
Fragment

$$x^* = \arg\min_x \sum_i E_i(x_i) + \sum_{<ij>} E_{ij}(x_i, x_j)$$

Search Space



Data Term



Learning Formulation

. . . . . .

Given training set and tens of thousands of fragments, choose a small number of fragments, thresholds and weights.

Learning Formulation

. . . . . .

Given training set and tens of thousands of fragments, choose a small number of fragments, thresholds and weights.

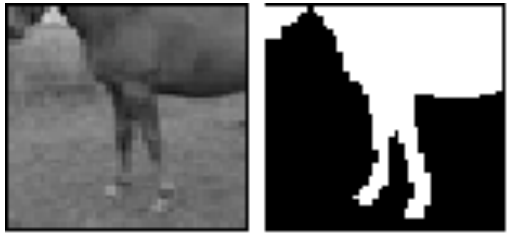
Equivalent to Feature Induction in Conditional Random Fields (Lafferty et al. 97, Lafferty et al. 2001)

Feature Induction in CRFs

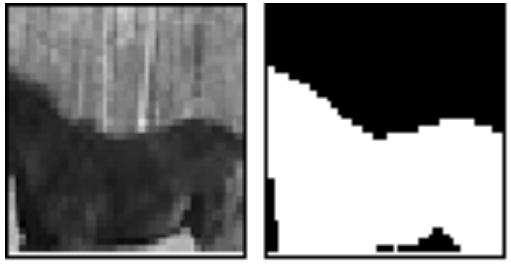
Training set:

$$E(x; I) = \sum_{<ij>} w_{ij}(I)|x_i - x_j|$$

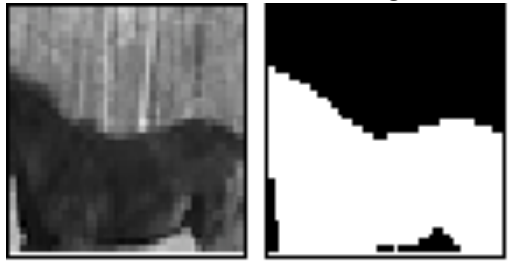
$+\lambda_1 \|x - x_{F_1}\|$



$+\lambda_2 \|x - x_{F_2}\|$



$+\lambda_3 \|x - x_{F_3}\|$



$$\lambda^* = \arg\max \log P(x; \lambda) + \textit{sparsity}$$

Calculating Conditional Likelihood Exactly is Intractable

Iteratively add fragment with highest likelihood gain:

$$\begin{aligned} \Pr(x^*; E) &= \frac{1}{Z(E)} e^{-E(x^*)} \\ &= \frac{1}{\sum_x e^{-E(x)}} e^{-E(x^*)} \end{aligned}$$

- Give low energy to desired segmentations and high energy to all other segmentations.
- “all other” : exponentially many.
- Need to evaluate likelihood gain for tens of thousands of fragments.

Bounding log Z using Belief Propagation

- Input: $\{\Psi_{ij} \propto e^{-E_{ij}}\}$
- Output: beliefs $\{b_{ij}\}$, $\{b_i\}$

Iterate:

$$m_{ij}(x_j) \leftarrow \sum_{x_i} \prod_{k \neq j} m_{ki}^{\rho}(x_i) m_{ji}^{\rho-1}(x_i) \Psi_{ij}^{\rho}(x_i, x_j)$$

- $\rho = 1$,standard BP.
- $\rho < 1$ "Tree-Reweighted/Fractional/Convex" BP.

(Wainwright et al. 03, Wiegerink and Heskes 03, Weiss et al. 07)

Bounding log Z using Belief Propagation

- Input: $\{\Psi_{ij} \propto e^{-E_{ij}}\}$
- Output: beliefs $\{b_{ij}\}$, $\{b_i\}$

Iterate:

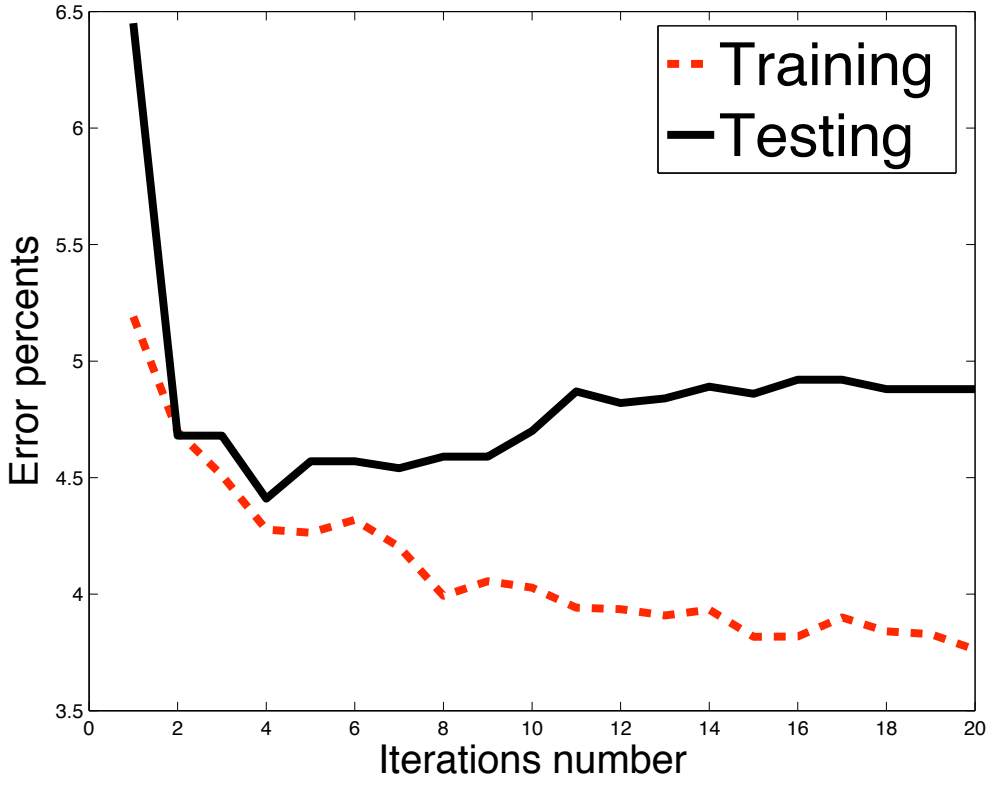
$$m_{ij}(x_j) \leftarrow \sum_{x_i} \prod_{k \neq j} m_{ki}^{\rho}(x_i) m_{ji}^{\rho-1}(x_i) \Psi_{ij}^{\rho}(x_i, x_j)$$

- $\rho = 1$,standard BP.
- $\rho < 1$ "Tree-Reweighted/Fractional/Convex" BP.

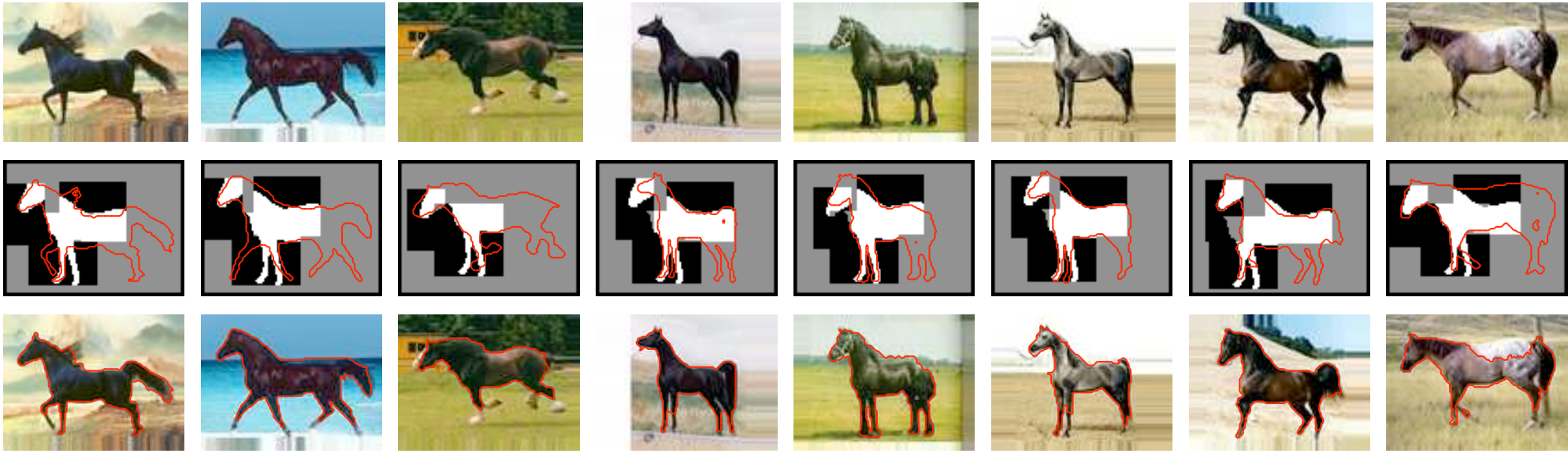
Messages can be reused for all fragments (speedup factor of 10,000)

(Wainwright et al. 03, Wiegerink and Heskes 03, Weiss et al. 07)

Learning Results



Test Set Results



“Lemmas along the way”

1. For many real-world low-level vision problems, we can efficiently find the global optimum of the energy function. □
2. For many real-world low-level vision problems, we can efficiently learn energy functions. □

(Meltzer et al. 2005, Levin and Weiss 2006, Sontag et al 2008, Kolmogorov and Wainwright 2005, Werner 2007, Rother et al. 2007, Boros and Zabih 2007, Komodakis and Paragios 2007, Woodford et al. 2009, Kohli and Torr 2008, Kumar and Torr 2009, He and Zemel 2008, Scharstein and Pal 2008, Lan, Roth, Huttenlocher and Black 2007 . . . .)

The Dream ?

Training set:

Learning Algorithm $\Rightarrow E(v_1, v_2, g_1, g_2)$

The Dream part 2

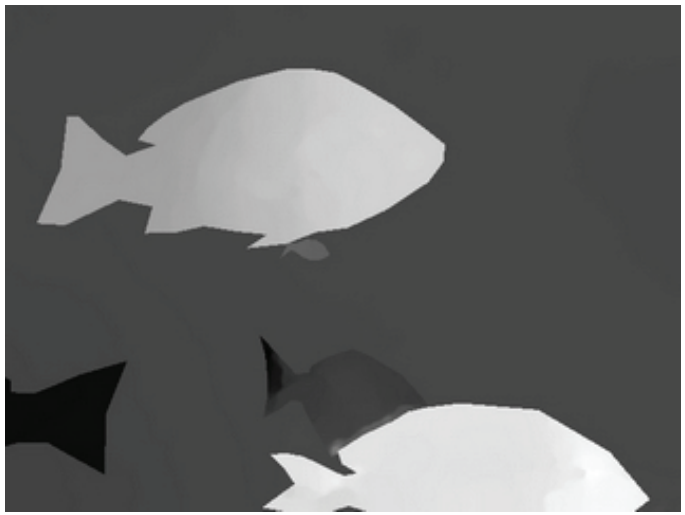
Test sequence:

Optimization Algorithm $\Rightarrow v_1^*, v_2^*, g_1^*, g_2^*$

State of the Art (CVPR 2008)

Input

Output:

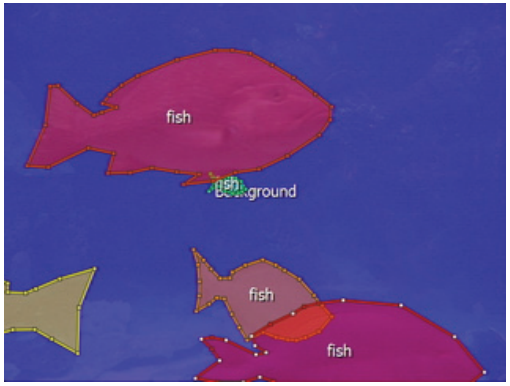


(Liu, Freeman, Adelson, Weiss 2008)

State of the Art (CVPR 2008)

Input

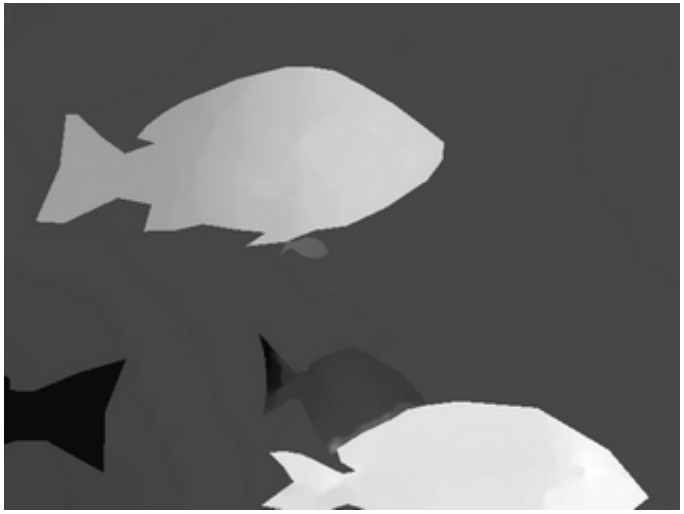
Input



Input

$\lambda_1, \lambda_2, \lambda_3 \cdots$

Output:



(Liu, Freeman, Adelson, Weiss 2008)

Outline

- Low-Level Vision in Humans
- Inference and Stereo Vision
- Learning and Segmentation
- Old vs. New Directions.

Much Emphasis on Pairwise Discrete Energies

Input:

$$E(x) = \sum_i E_i(x_i) + \sum_{<ij>} E_{ij}(x_i, x_j)$$

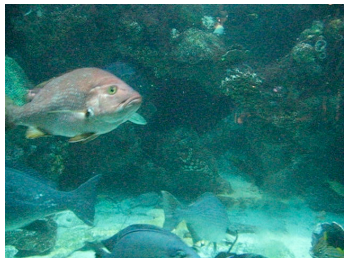
Future Directions - Inference

$$
\begin{aligned}
E(v_1, v_2, g_1, g_2) &= dataterm(v_1, v_2, g_1, g_2) \\
&= +\lambda\,(smooth(v_1) + smooth(v_2)) \\
&= +\lambda_2\,(smooth(g_1) + smooth(g_2)) \\
&= +\lambda_3 complexity(g)
\end{aligned}
$$

Can we globally optimize higher-order energy functions with discrete and continuous variables ?

Much Emphasis On Train and Test Paradigm

Training set:

Learning Algorithm $\Rightarrow E(v_1, v_2, g_1, g_2) \Rightarrow$ 

Future Directions - Learning

$$\begin{aligned} \lambda^* &= \arg\max \Pr(x;\lambda) \\ x^* &= \arg\min E(x;\lambda^*) \end{aligned}$$

Can we learn/adapt energy functions without additional training set ? (Empirical/Hierarchical/Nonparametric Bayes)

(Szeliski 89, Vasconcelos 95, Raphan and Simoncelli 08)

Conclusions

- Low-Level vision in humans consistent with highly sophisticated energy minimization.
- For many real-world problems we can:
  - *globally* optimize energy, despite NP hardness.
  - learn good energy functions, despite exponential number of constraints.
- Much work left.