Convexity Shape Prior for Segmentation

Lena Gorelick

joint work with

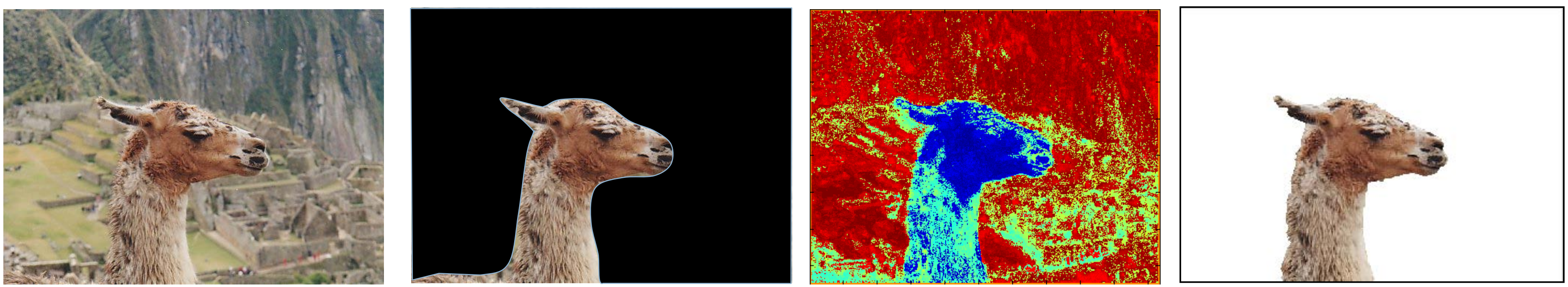


O. Veksler Y. Boykov C. Nieuwenhuis

Binary Segmentation Energy

$$E(\mathbf{x}) = \sum_{p} f_p \cdot x_p + \lambda \sum_{p,q \in \mathcal{N}} 1_{[x_p \neq x_q]}$$

Length Regularization Potts Model



$\mathbf{x} \in \{0,1\}^{\Omega}$ $\mathbf{f}$

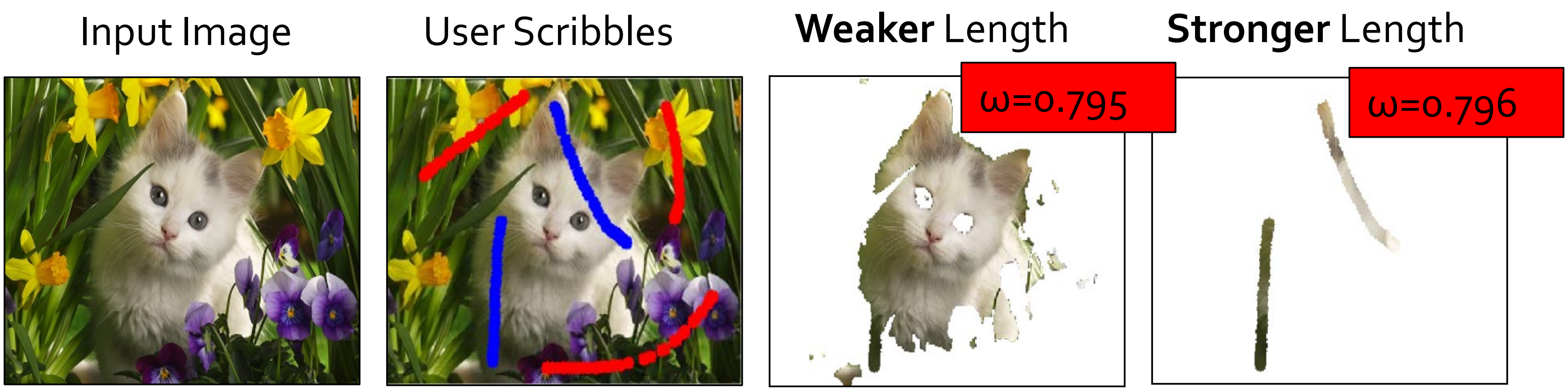
Submodular

global optimum with graphcut

(Boros & Hammer, 2002)

Length Regularization Shortcomings

$$E(\mathbf{x}) = \sum_{p} f_p \cdot x_p + \omega \sum_{p,q \in N} 1_{[x_p \neq x_q]}$$

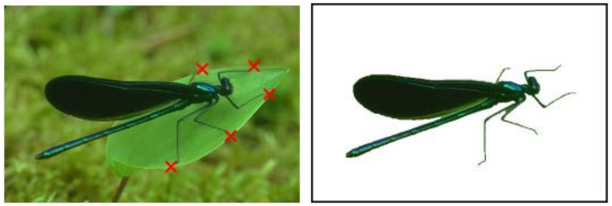


- Shrinking bias
- Sensitivity to weight of regularization

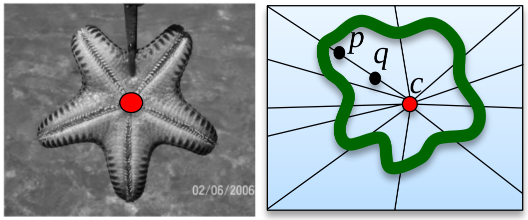
Other Regularization Models

- **Curvature**
  Schoeneman 2009, Olsson 2013, Nieuwenhuis et al. 2014

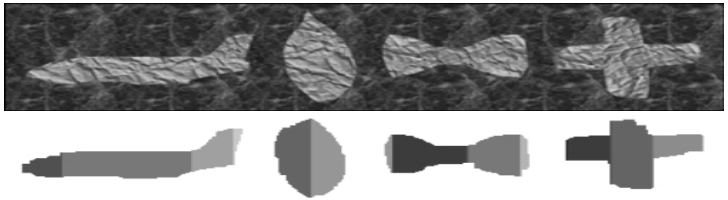
- **Connectivity Prior**
  Vicente et al.2008, Novozin et al. 2010



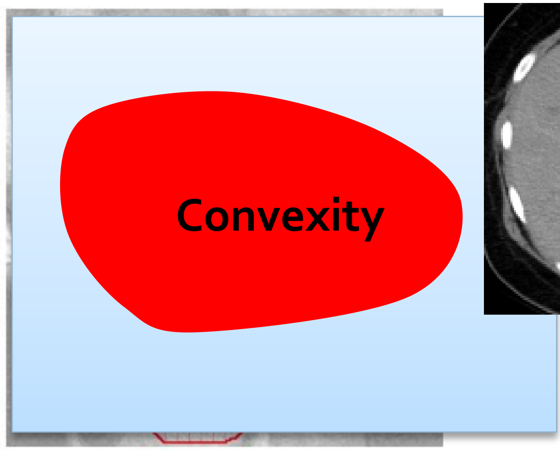
- **Star-Shape Prior**
  Veksler 2008, Gulshan et al. 2010

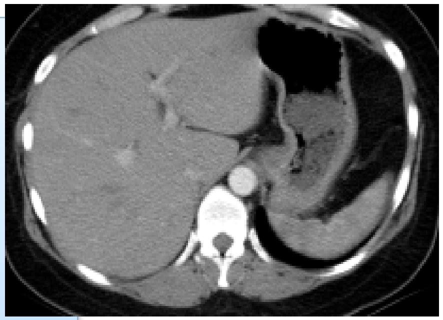


- **Part-Based Shape Prior**
  Felzenszwalb & Veksler 2010



Shape Convexity-
High Order Regularization





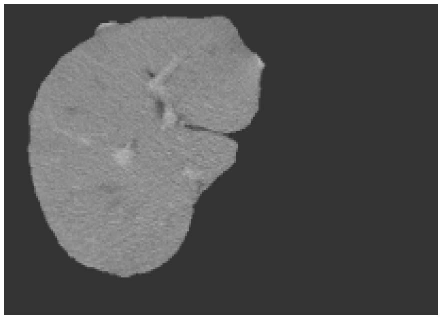


Image credit: Andrew Delong

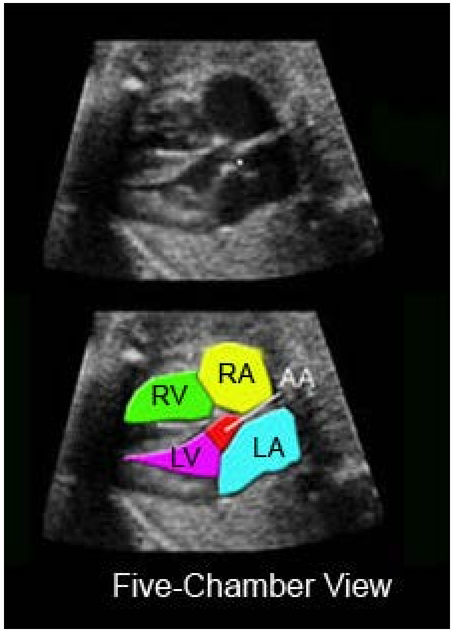
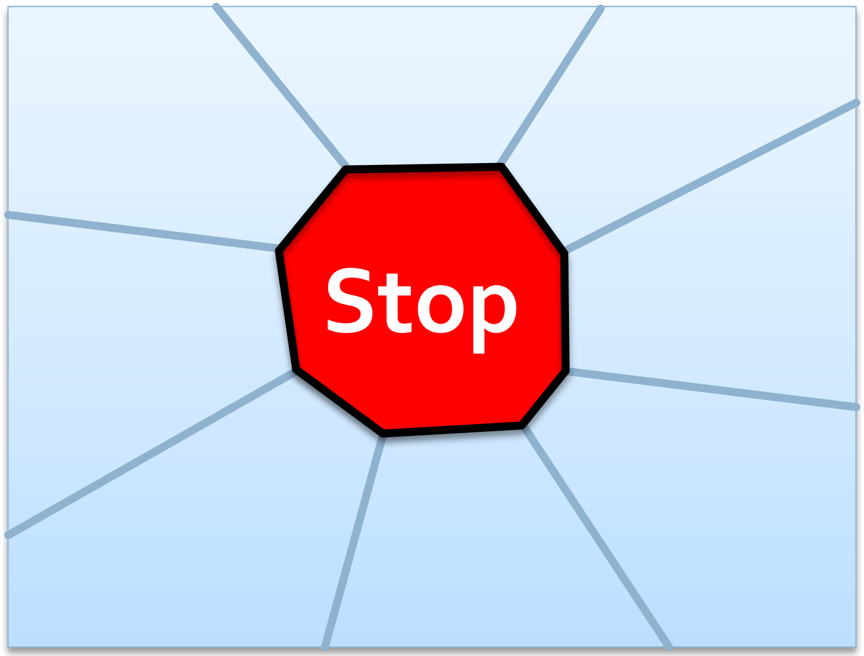


Image credit: http://www.fetal.com/

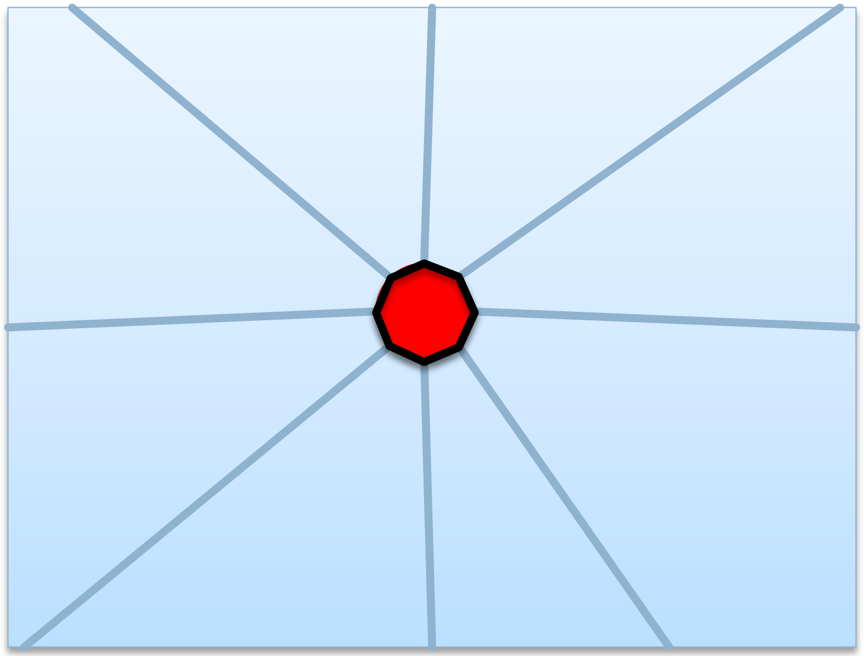
Related Work in Continuous Optimization

- **N-sided convex polygon** Strekalovskiy & Cremers 2011



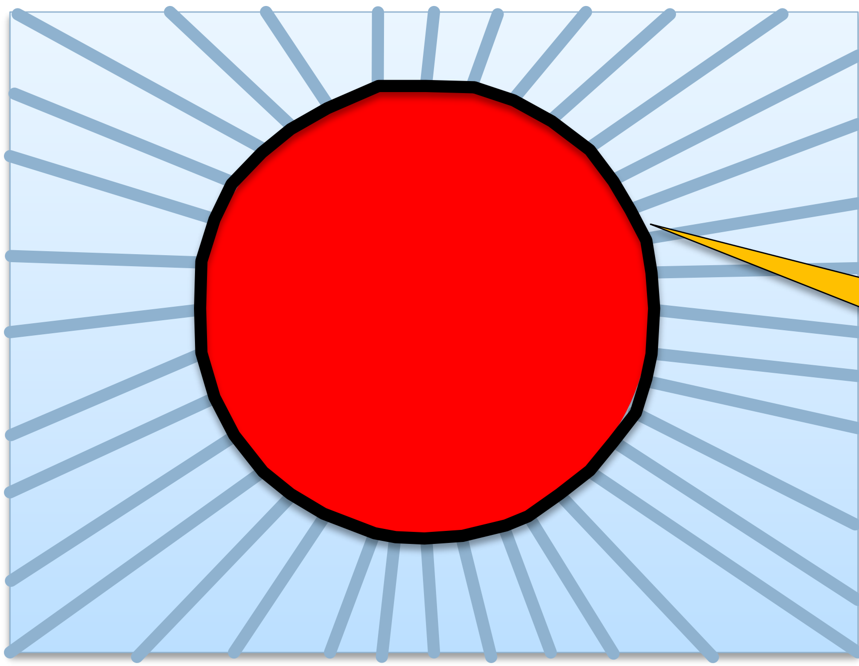
Related Work in Continuous Optimization

- **N-sided convex polygon** Strekalovskiy & Cremers 2011



Related Work in Continuous Optimization

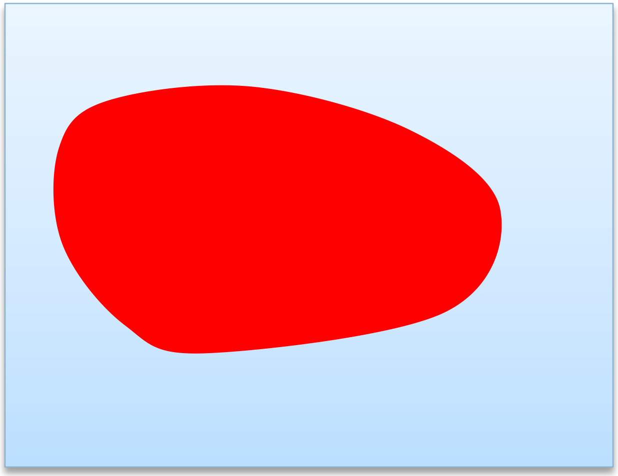
- **N-sided convex polygon** Strekalovskiy & Cremers 2011



- More Parts
- Finer Discretization of Orientation
- Expensive to Optimize

Our Approach

- We can obtain an **arbitrary convex object** for **any choice of** orientation **discretization**



- Discrete Optimization Framework
  - Efficient to optimize without GPU

Our Segmentation Energy with Convexity Shape Prior

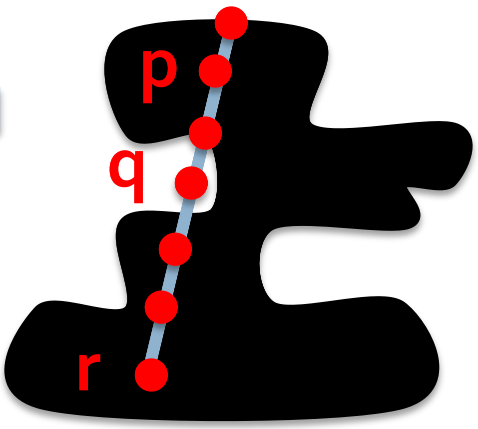
$$E(\mathbf{x}) = E^{sub}(\mathbf{x}) + E^{convexity}(\mathbf{x})$$

Discrete Segmentation Energy

Appearance
Hard Constraints
Color Separation
Length
...

Energy Formulation for Convexity Shape Prior

$$E^{convexity}(\mathbf{x})$$

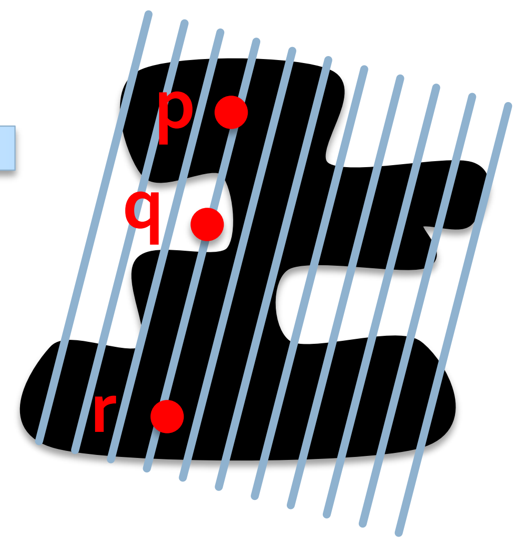


$$\varphi(\ 1\ \ 0\ \ 1\ ) = \infty$$

p q r

Energy Formulation for Convexity Shape Prior

$$E^{convexity}(\mathbf{x})$$



$$\varphi(\;1\;\;0\;\;1\;) = \infty$$

p q r

Energy Formulation for Convexity Shape Prior

$$E^{convexity}(\mathbf{x})$$

$$\varphi(\;1\;\;0\;\;1\;) = \infty$$

p q r

Energy Formulation for Convexity Shape Prior

$$E^{convexity}(\mathbf{x})$$

$$\varphi(\ 1\ \ 0\ \ 1\ ) = \infty$$

p q r

Energy Formulation for Convexity Shape Prior

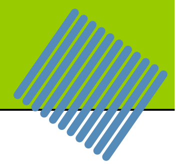
$$E^{convexity}(\mathbf{x})$$

$$\varphi(\ 1\ 0\ 1\ ) = \infty$$

p q r

Energy Formulation for Convexity Shape Prior

$$E^{convexity}(\mathbf{x}) = \sum_{l \in}$$



$$\varphi(\ 1\ \ 0\ \ 1\ ) = \omega$$

p q r

Energy Formulation for Convexity Shape Prior

- Convexity Shape Prior

O(N²) , N=|Ω|
All triplets

- Squared Curvature

Nieuwenhuis et al. CVPR14

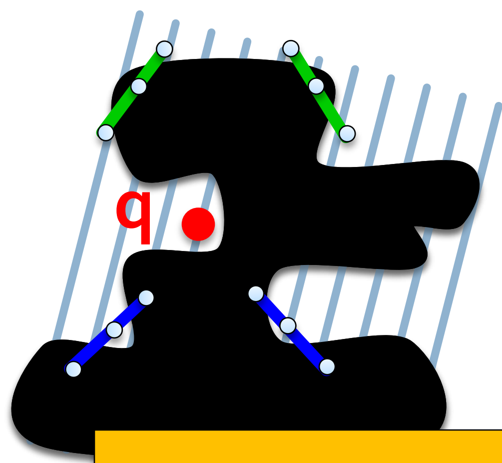
$$\varphi(\;1\;\;0\;\;1\;) = \omega_1$$

$$\varphi(\;1\;\;0\;\;1\;) = \omega$$

p q r

$$\varphi(\;0\;\;1\;\;0\;) = \omega_2$$

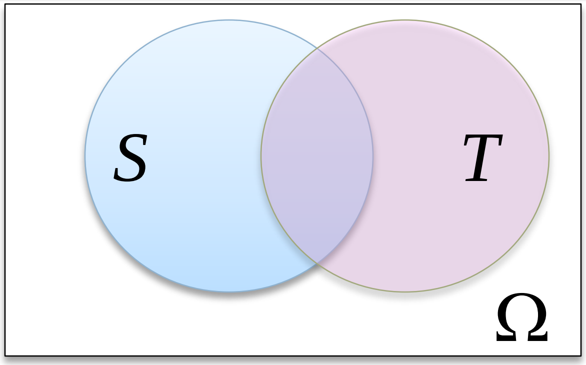
p q r



O(N) , N=|Ω|
local triplets

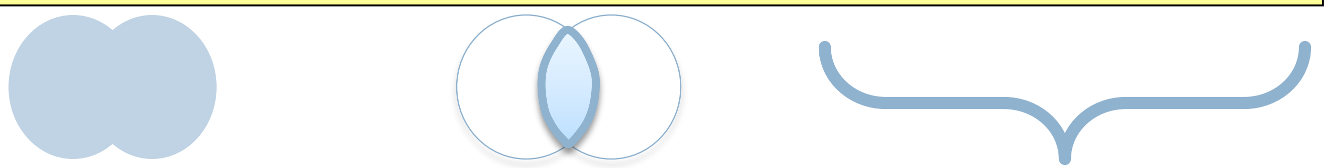
Difficulties in optimizing $E^{convexity}$

- $E^{convexity}$ is not submodular



$$E(S \vee T) + E(S \wedge T) \leq E(S) + E(T)$$

Original definition by Edmonds, 1970



∞ 0 0

- Too many triple potentials
  Naïve evaluation is expensive O(N²)

Optimization

- Trust Region Framework
  - Discrete High Order Energies
  - Binary Pairwise Energies

Gorelick et al. ECCV12, CVPR13

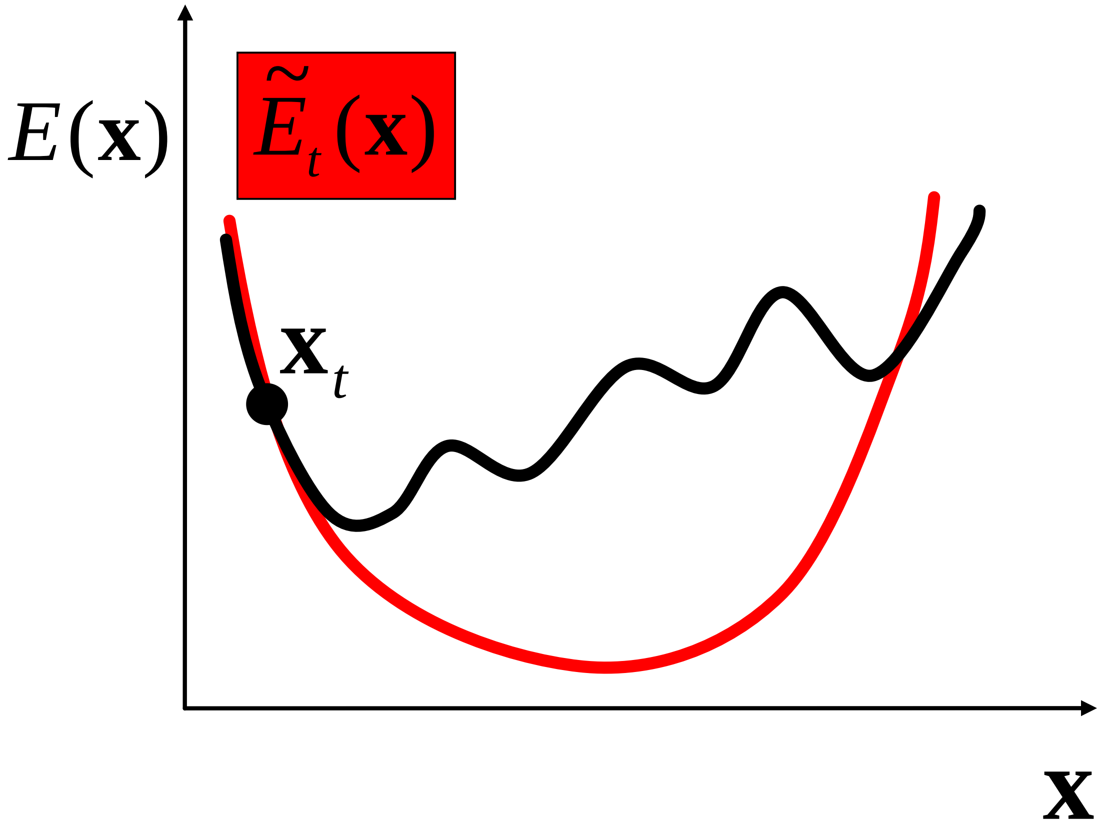
Gorelick et al. CVPR14

- Direct application – too slow!
- Dynamic Programming to speed up energy evaluation and approximation

Trust Region Overview

$$E(\mathbf{x}) = E^{sub}(\mathbf{x}) + E^{convexity}(\mathbf{x})$$

Gorelick et al. ECCV12, CVPR13, CVPR14



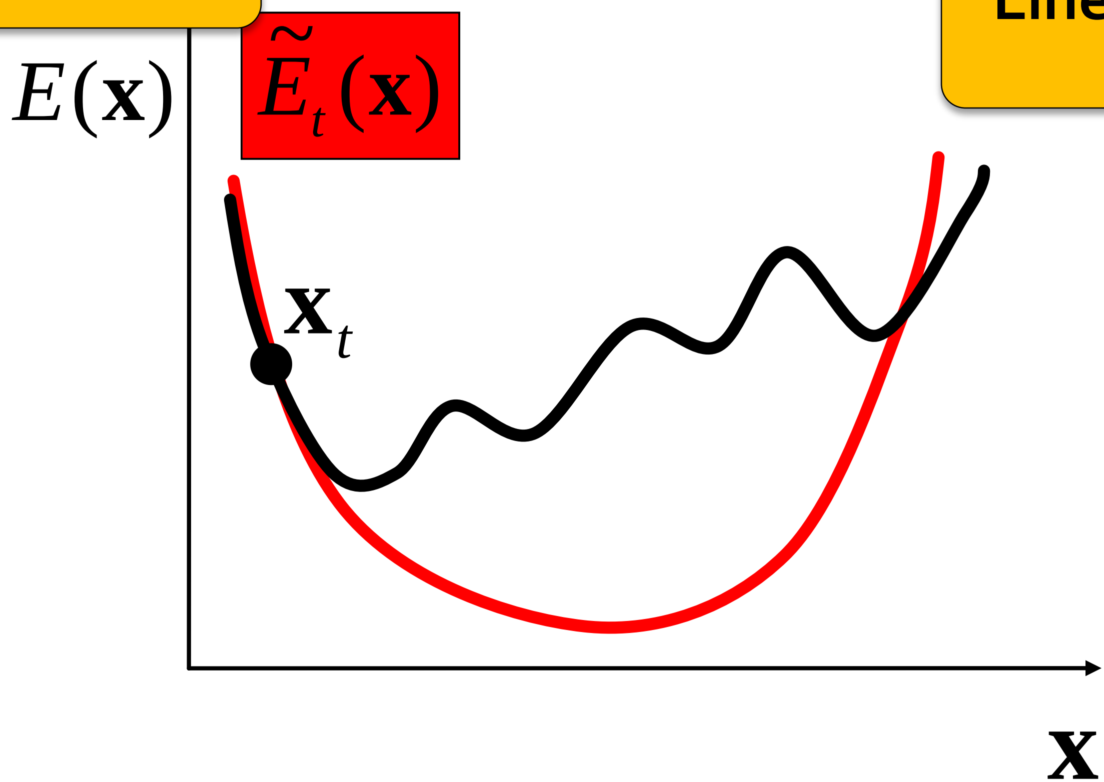
Trust Region Overview

$\tilde{E}_t(\mathbf{x}) =$

Submodular

Linear Approximation of $E^{convexity}$

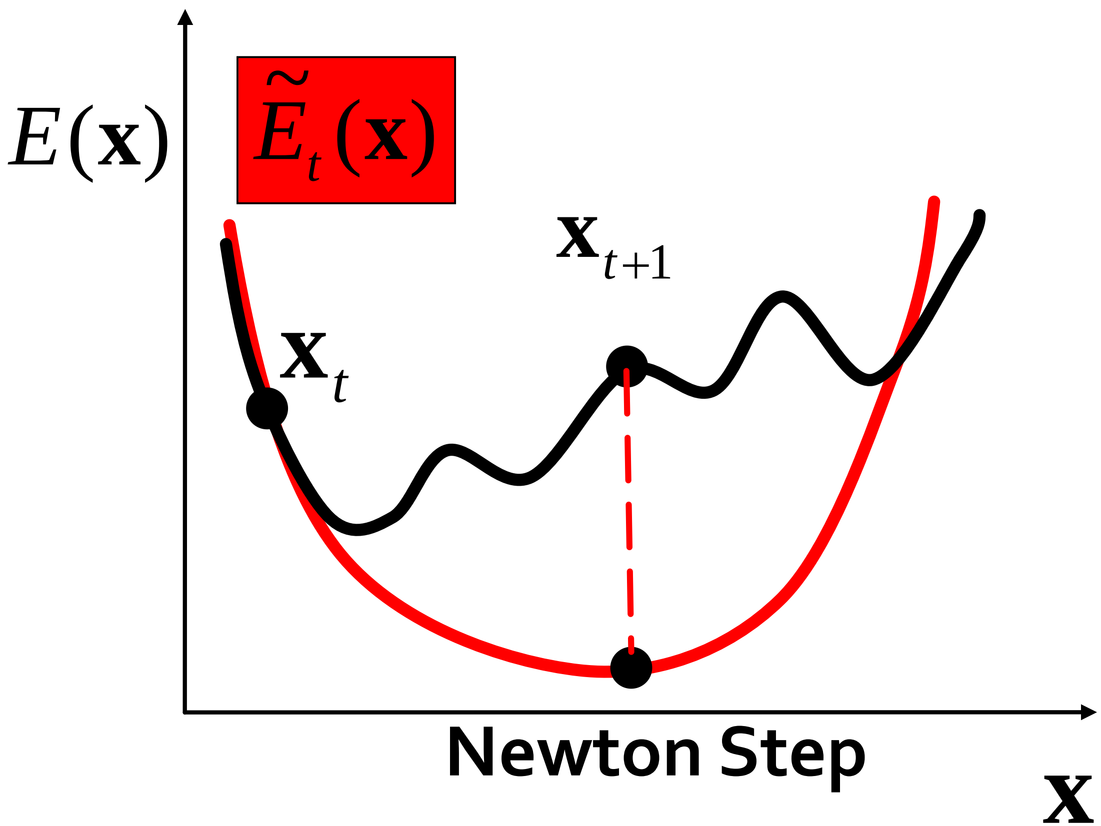
Gorelick et al.
ECCV12, CVPR13,
CVPR14



Trust Region Overview

$$\tilde{E}_t(\mathbf{x}) = E^{sub}(\mathbf{x}) + E_t^{approx}(\mathbf{x})$$

Gorelick et al.
ECCV12, CVPR13,
CVPR14

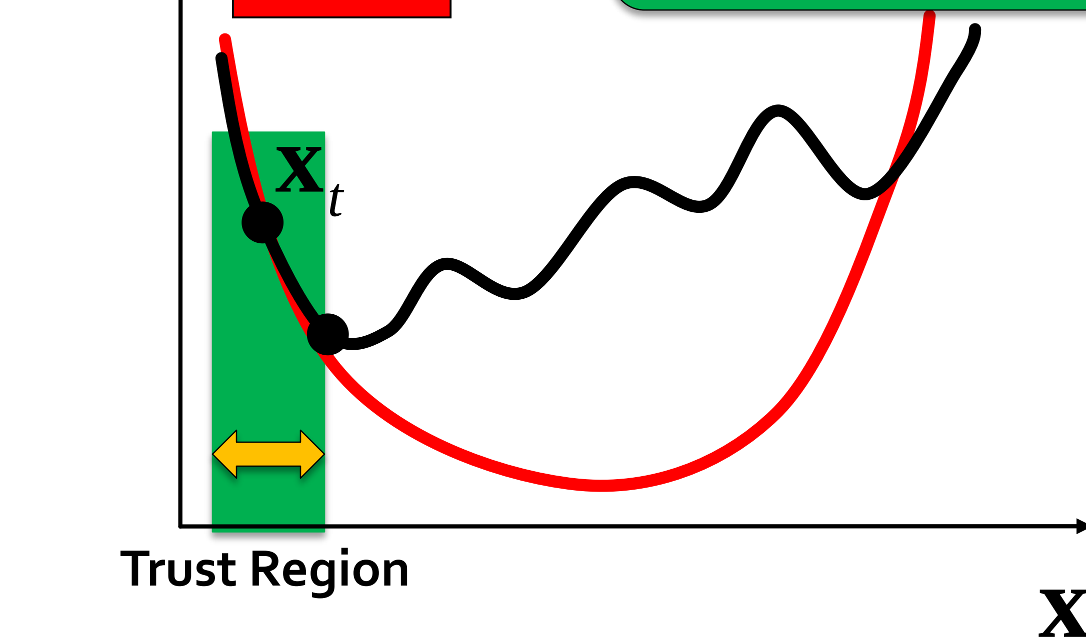


Trust Region Sub-Problem

$$\tilde{E}_t(\mathbf{x}) = E^{sub}(\mathbf{x}) + E_t^{approx}(\mathbf{x})$$

Gorelick et al. ECCV12, CVPR13, CVPR14

s.t. $\quad \| \mathbf{x} - \mathbf{x_t} \| < d_t$



Trust Region Sub-Problem

NP-hard!

Approximate TR sub-problem

$$L_t(\mathbf{x}) = E^{sub}(\mathbf{x}) + E_t^{approx}(\mathbf{x}) + \lambda_t \| \mathbf{x} - \mathbf{x}_t \|$$

Gorelick et al. ECCV12, CVPR13, CVPR14

Submodular

Unary Terms
Boykov et al. 2006

λ_t
- fixed in each iteration
- inversely related to trust region size
- adjusted based on quality of approximation

Trust Region & Dynamic Programming

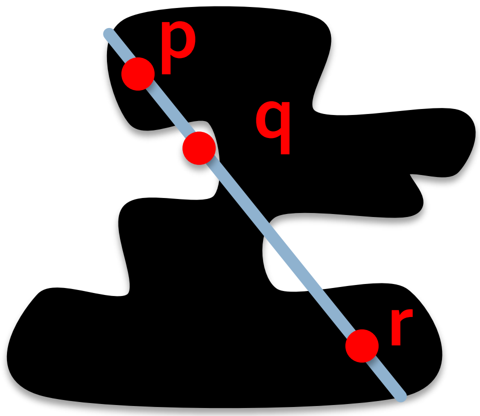
- Evaluate and approximate in each iteration $E^{convexity}$

- **Naïve computation is** **O(N²)**

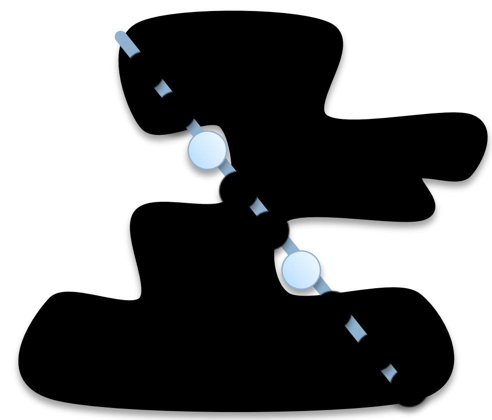
- We use **dynamic programming** **O(N)**

Evaluation of $E^{convexity}$

$$\varphi(\underset{p}{1}\ \underset{q}{0}\ \underset{r}{1}) = \omega$$

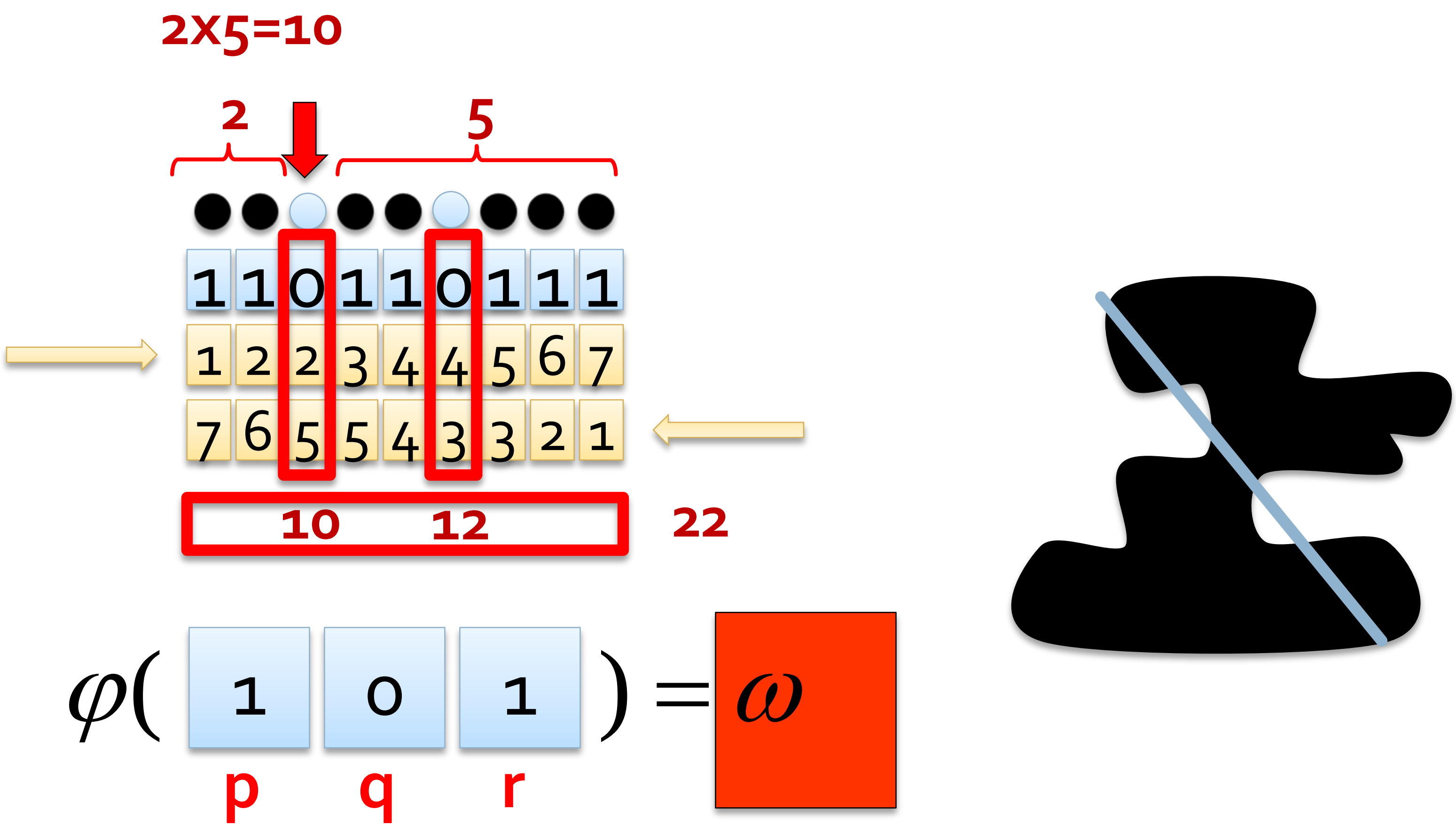


Evaluation of $E^{convexity}$



$$\varphi(\underset{p}{1}\ \underset{q}{0}\ \underset{r}{1}) = \omega$$

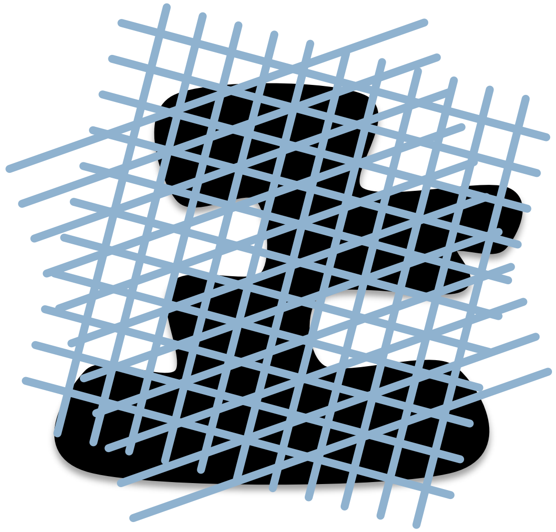
Evaluation of $E^{convexity}$



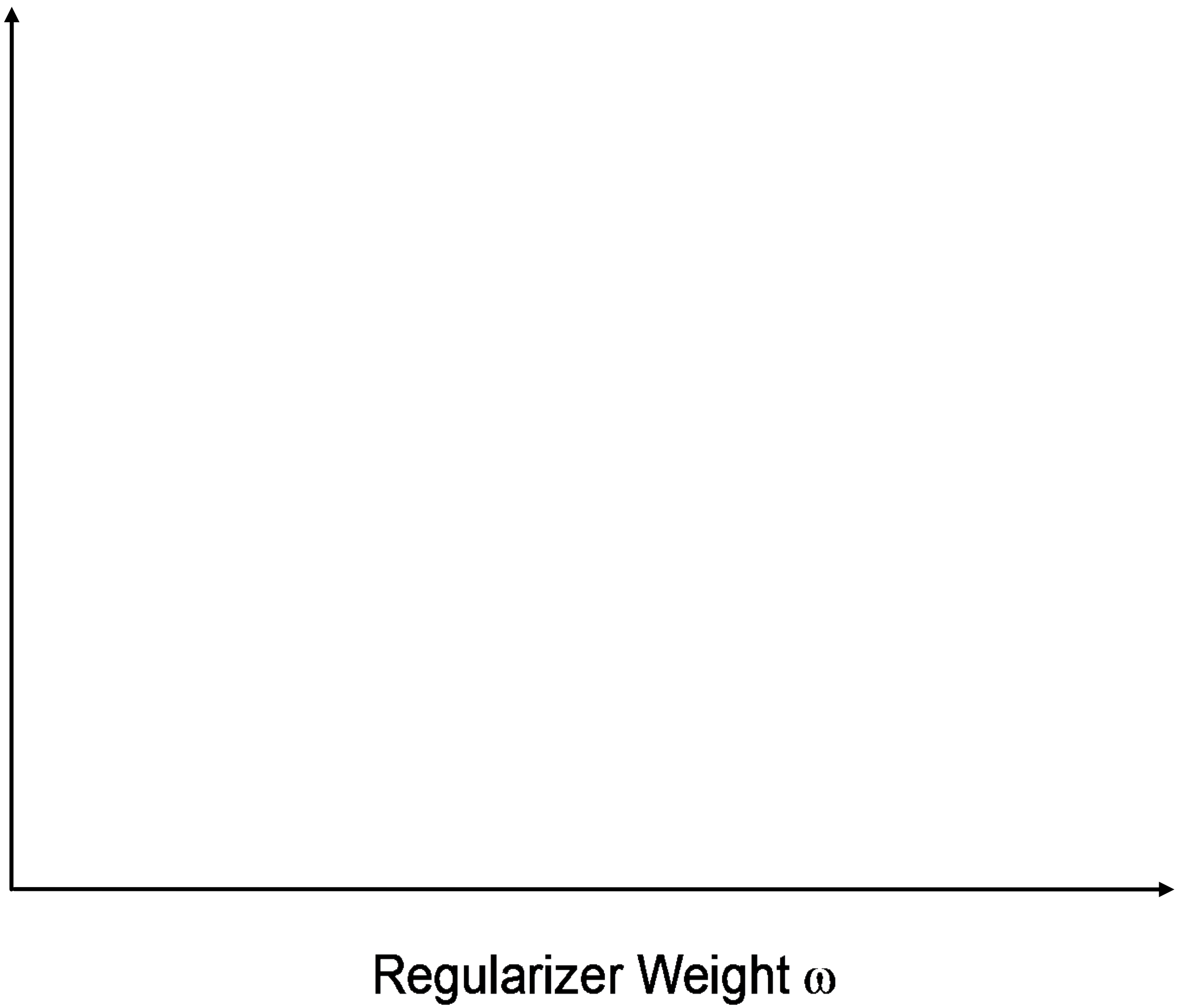
Evaluation of $E^{convexity}$

- It takes O(mN) operations to scan all lines in all orientations

N = |Ω|
m = #orientations

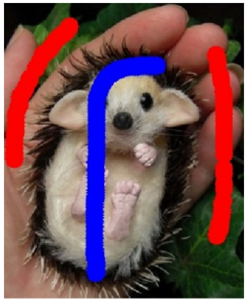


Experiments & Results
Interactive Image Segmentation



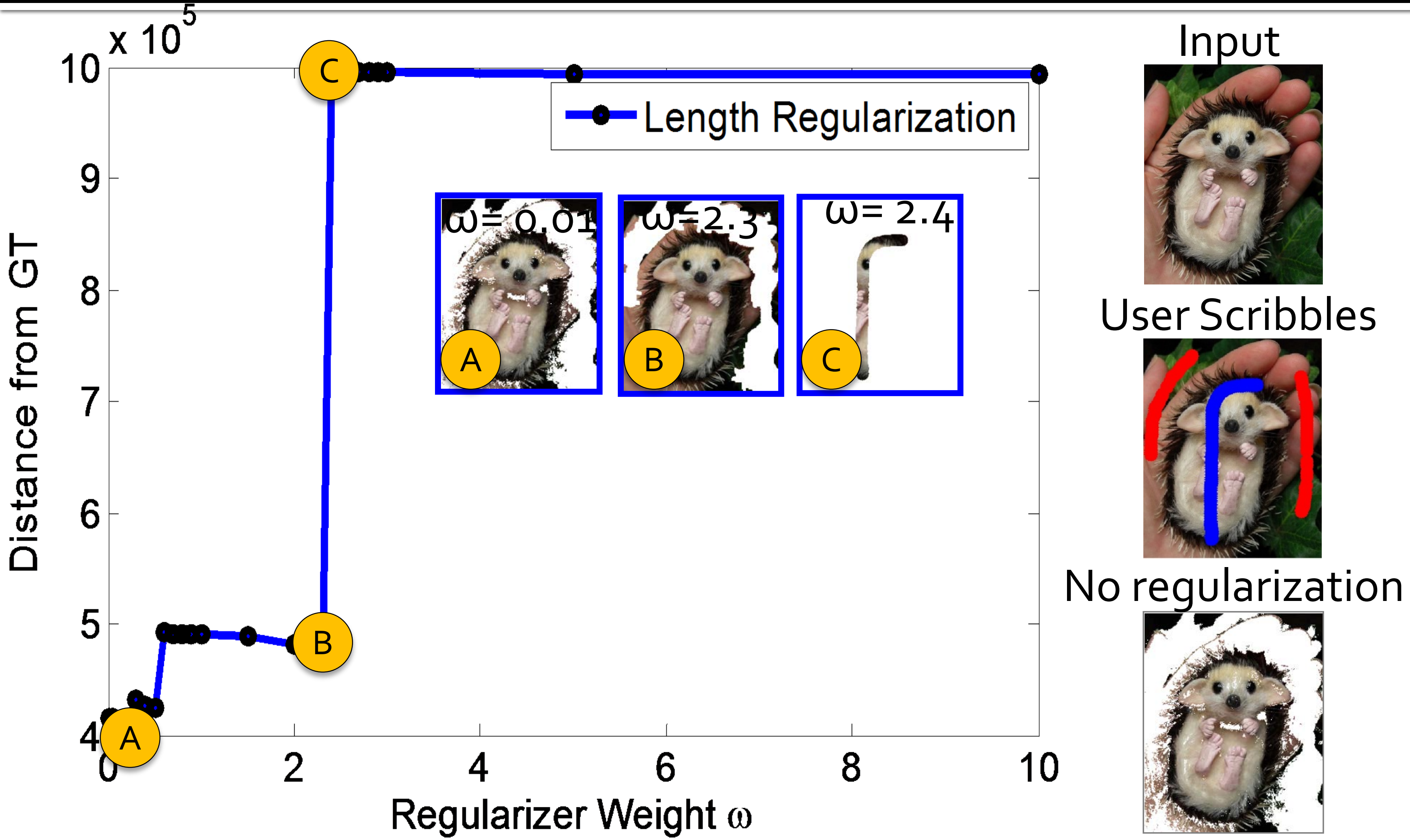
Input

User Scribbles

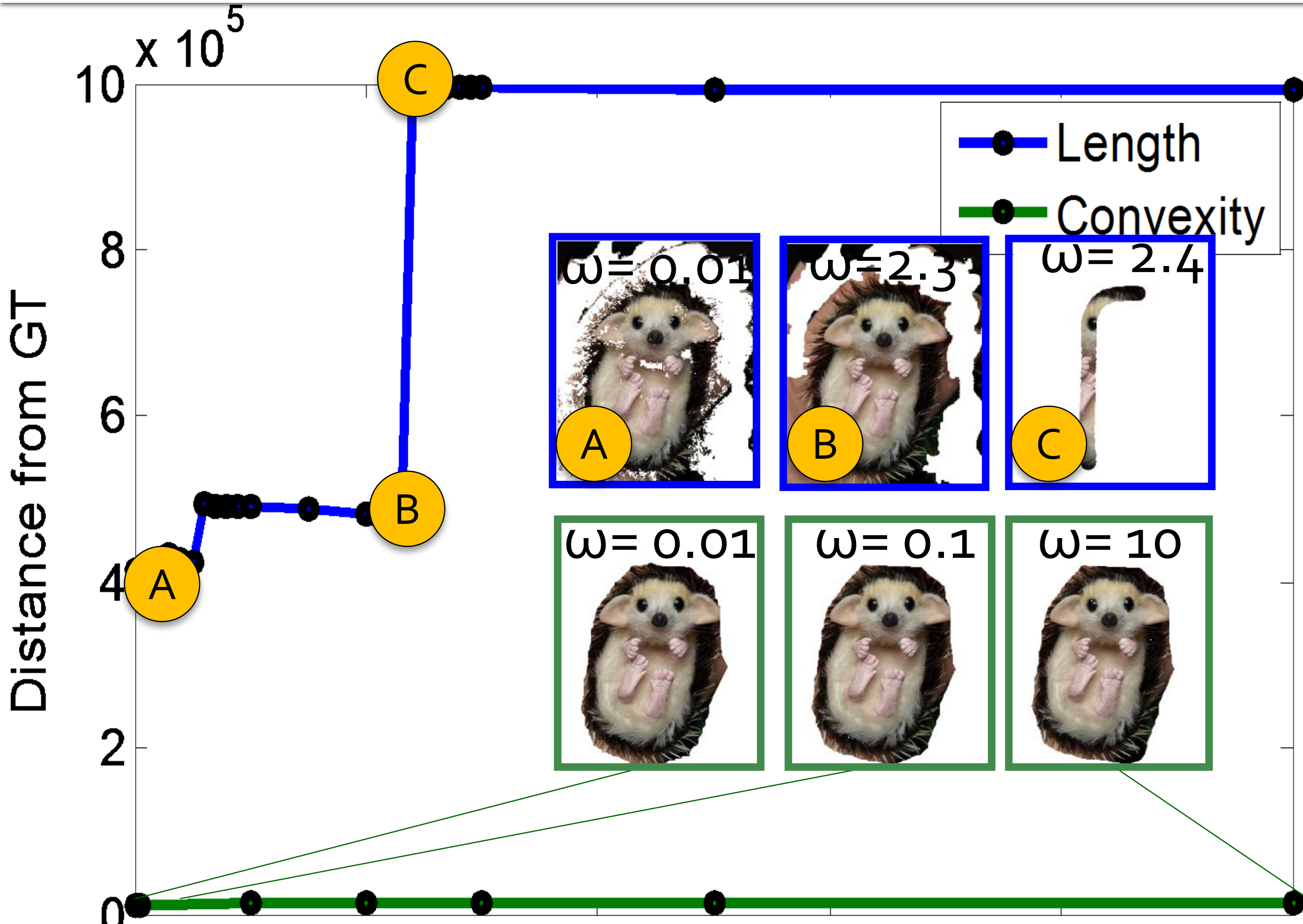


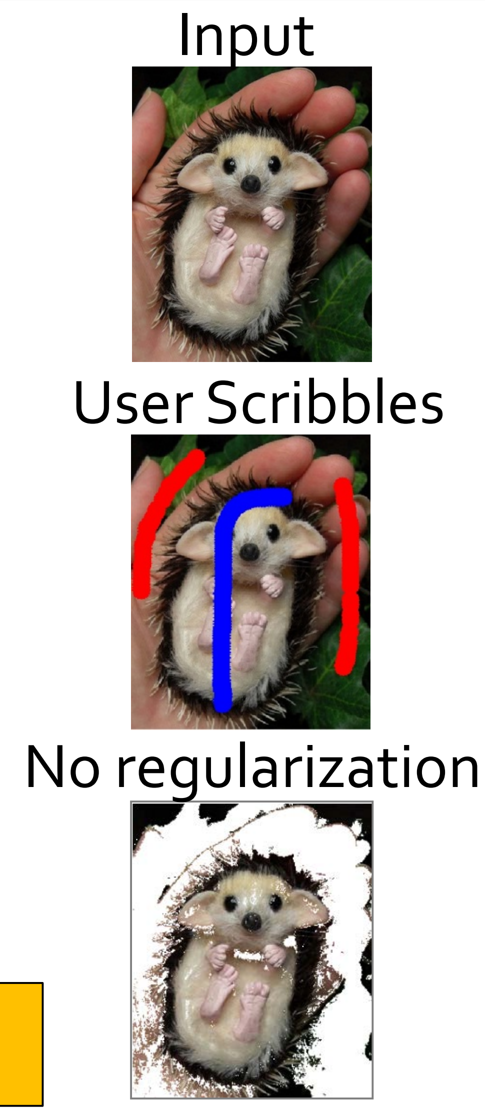
No regularization

Experiments & Results
Interactive Image Segmentation



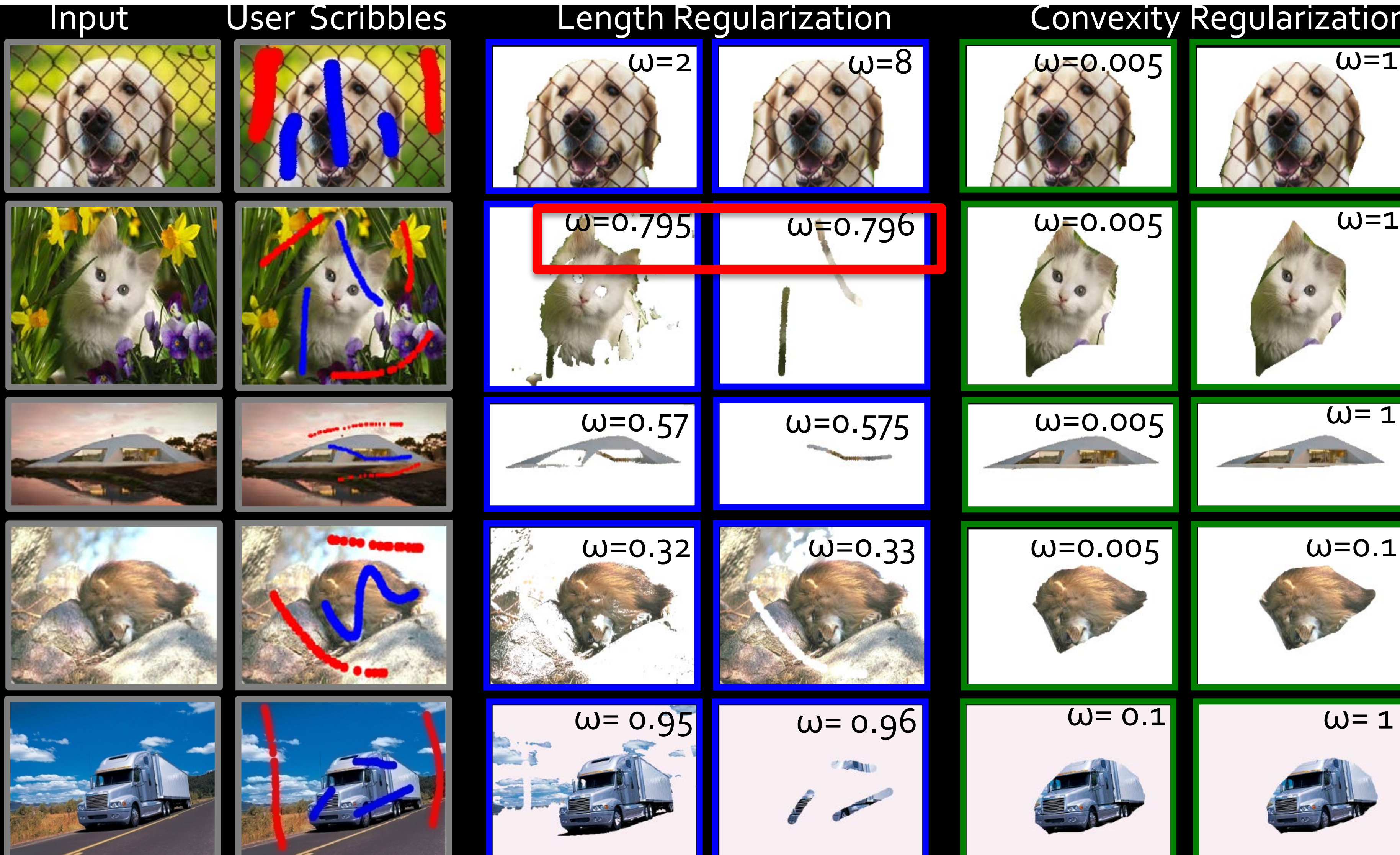
Experiments & Results
Natural Images



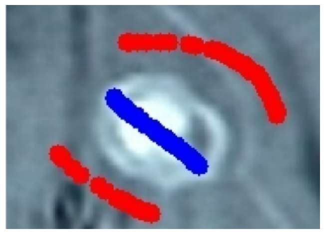
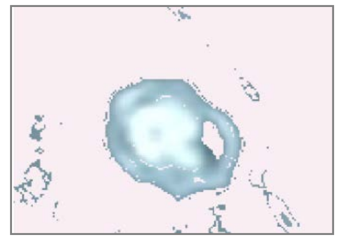
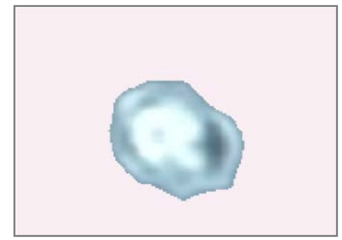
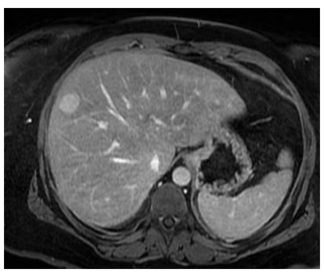
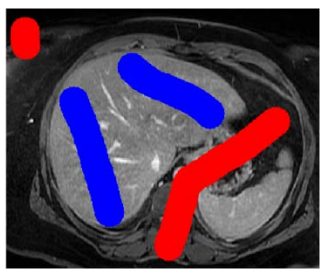
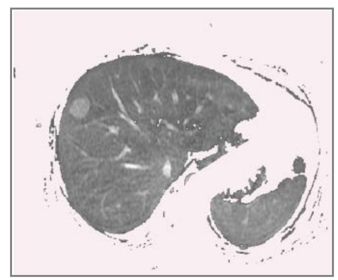
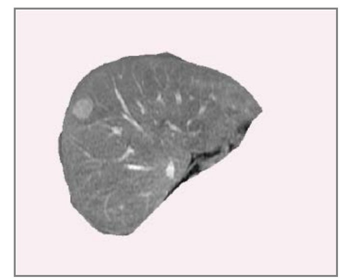


Virtually Parameter Free

Experiments & Results
Natural Images



Experiments & Results
Medical Images

	Input	User Scribbles	No regularization	Convexity	Running Time
170x126					0.9 sec.
430x360					18 sec.

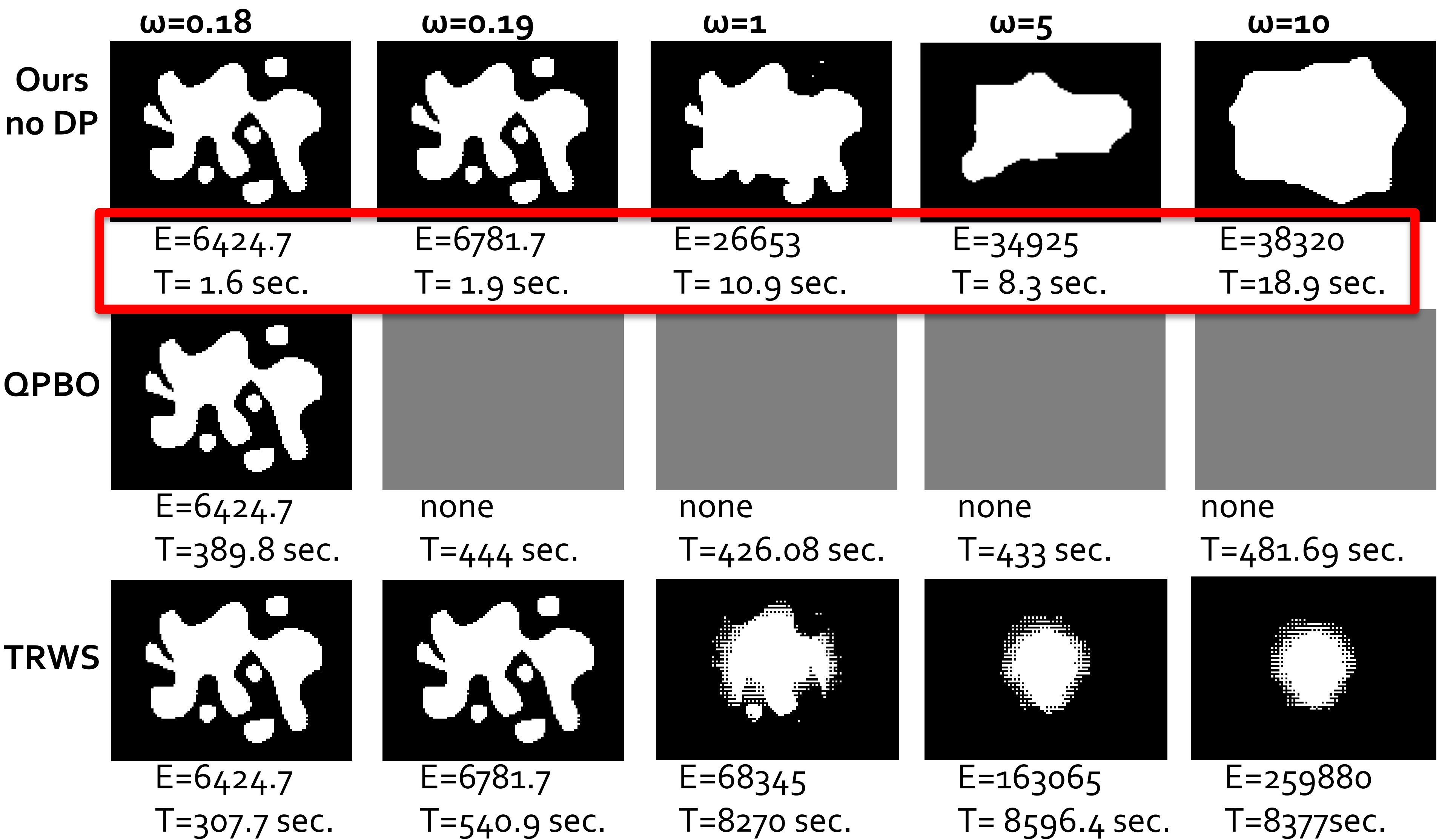
Comparison with QPBO and TRWS

- **In theory -** $E^{convexity}$
  can be optimized with QPBO and TRWS
- **In practice -** prohibitively expensive $O(N^2)$
- **Compact Model**: $O(N\sqrt{N})$ cliques

Slower than full model,
Dynamic Programming does not apply

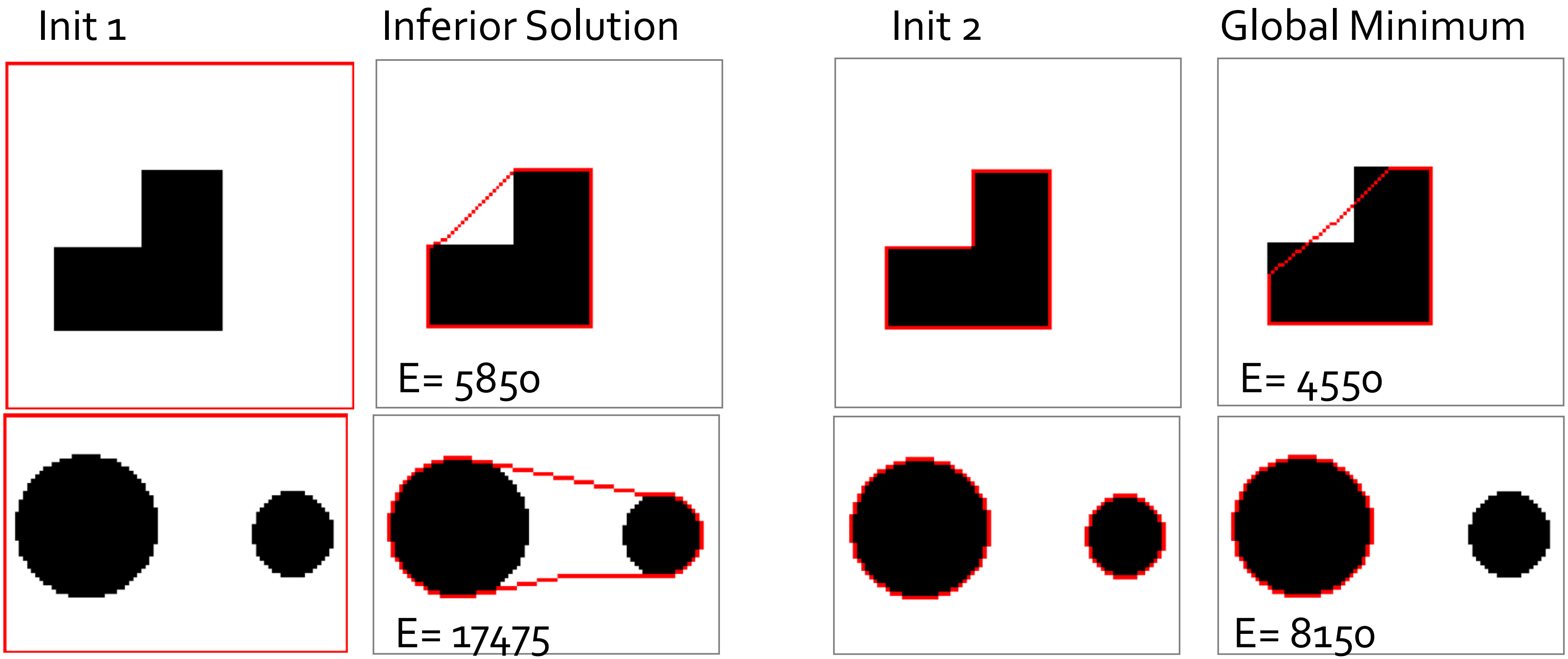
Our method w/o Dynamic Programming vs. **QPBO & TRWS**

Comparison with QPBO and TRWS for *Compact* Model



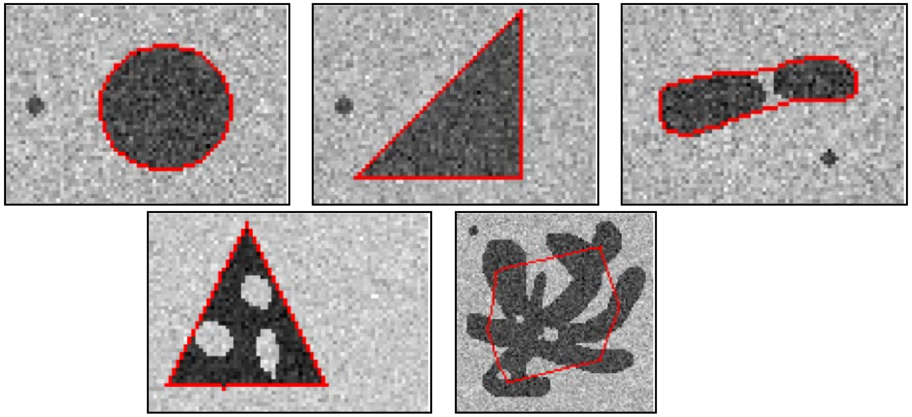
Limitations of our method

- TR is a local Iterative optimization → cannot guarantee global minimum
- Sensitivity to Initialization



Conclusions

- **Convexity shape prior** within discrete optimization framework
  - no shrinking bias
  - removes noise
  - fills in holes
  - ensures connectivity
  - preserves sharp corners



- Our model is scale invariant due to ∞ constraints
- Efficient optimization based on TR and DP

Thank you!

- Code is available online
  - **http://vision.csd.uwo.ca/code/**

- Please come by our poster tomorrow