A Directed Graph Approach to Active Contours

Adrian Barbu

Associate Professor

Department of Statistics

Florida State University

Overview

- Active Contours
  - Chan-Vese
- Related Work
- Proposed method
- Experiments:
  - Horse Segmentation
  - Liver Segmentation
- Conclusion and future work

Active Contour Energy

- Energy for a curve $c: [a, b] \rightarrow R^2$

$$E(c) = \int_a^b \left[E_{data}\big(c(t)\big) + E_{smo}\big(c(t)\big)\right] dt$$

  - Data term E_{data} encourages high gradient locations
  - Smoothness term E_{smo} encourages smooth curves
- Minimization is difficult because of self-intersections
- Level set representation
  - The closed curve is the 0-level set of a surface S
  - Extend the energy to E(S)
  - Evolve S instead of c

The Chan-Vese Algorithm

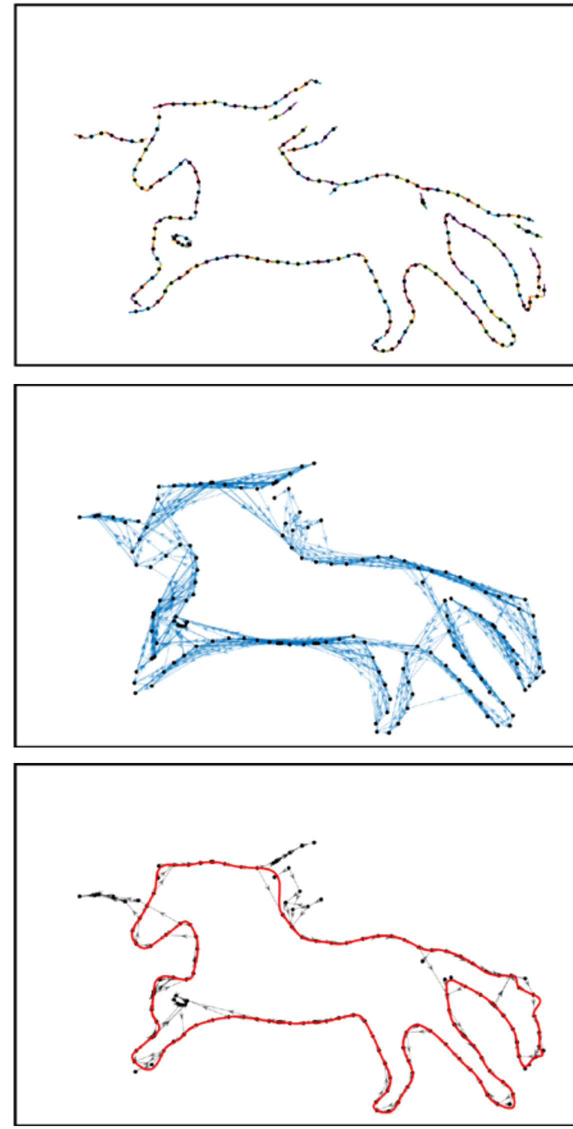
- Energy for a curve C: $[a, b] \rightarrow R^2$

$$E(C, \mu_1, \mu_2) = \int_{inside\ C} (I(x, y) - \mu_1)^2 dxdy + \int_{outside\ C} (I(x, y) - \mu_2)^2 dxdy + \mu \cdot len(C)$$

- Uses intensity information from the whole image
- Pros:
  - Less dependent on initialization
  - Robust to noise
  - Generalizes to 3D and beyond
- Cons:
  - Computationally expensive

Overview of Proposed Approach

- Minimize Active Contour Energy
  - Additive energy representation
- Construct a directed graph
  - Graph nodes = edge segments
  - Graph edges = smooth curves
  - Edge weights = partial AC energies
- Use graph optimization
  - Floyd-Warshall all-pairs shortest path
  - Obtain closed paths = segmentations



Related Work

- Minimum path algorithm for active contours (Kohen and Kimmel ’97)
  - Still used level sets for optimization
- Graph optimization (Barbu et al 2007)
  - Undirected graph
  - Open curves, no segmentation
- Linear model (Schoeneman et al, 2012)
  - Region based representation with 8/16 triangles for each pixel
    - Directed edges for interior/exterior
  - Curvature regularization
  - Optimization by linear programming

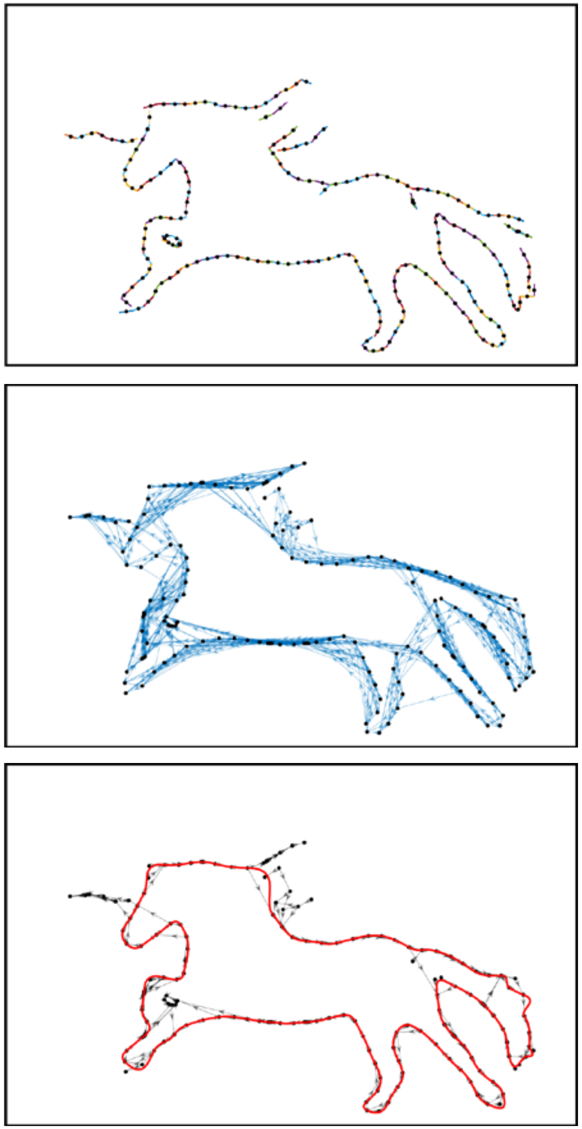
Additive Active Contour Energy

- Active contour energy depends on curve parameterization
- Want energy additive to curve concatenation
  - Use arc length parameterization
    $$c\colon [0, l\,] \to R^2, ||c'(t)|| = 1$$
  - Then if $c_1\colon [0, l_1] \to R^2$, $c_2\colon [0, l_2] \to R^2$ have arc length parametrization and $c\colon [0, l_1 + l_2] \to R^2$ is their concatenation we have
    $$E(c) = E(c_1) + E(c_2)$$
- Obtain curve energy for $c\colon [a, b] \to R^2$

$$E(c) = \int_a^b \left[E_{data}\big(c(t)\big) + E_{smo}\big(c(t)\big)\right]||c'(t)||dt$$

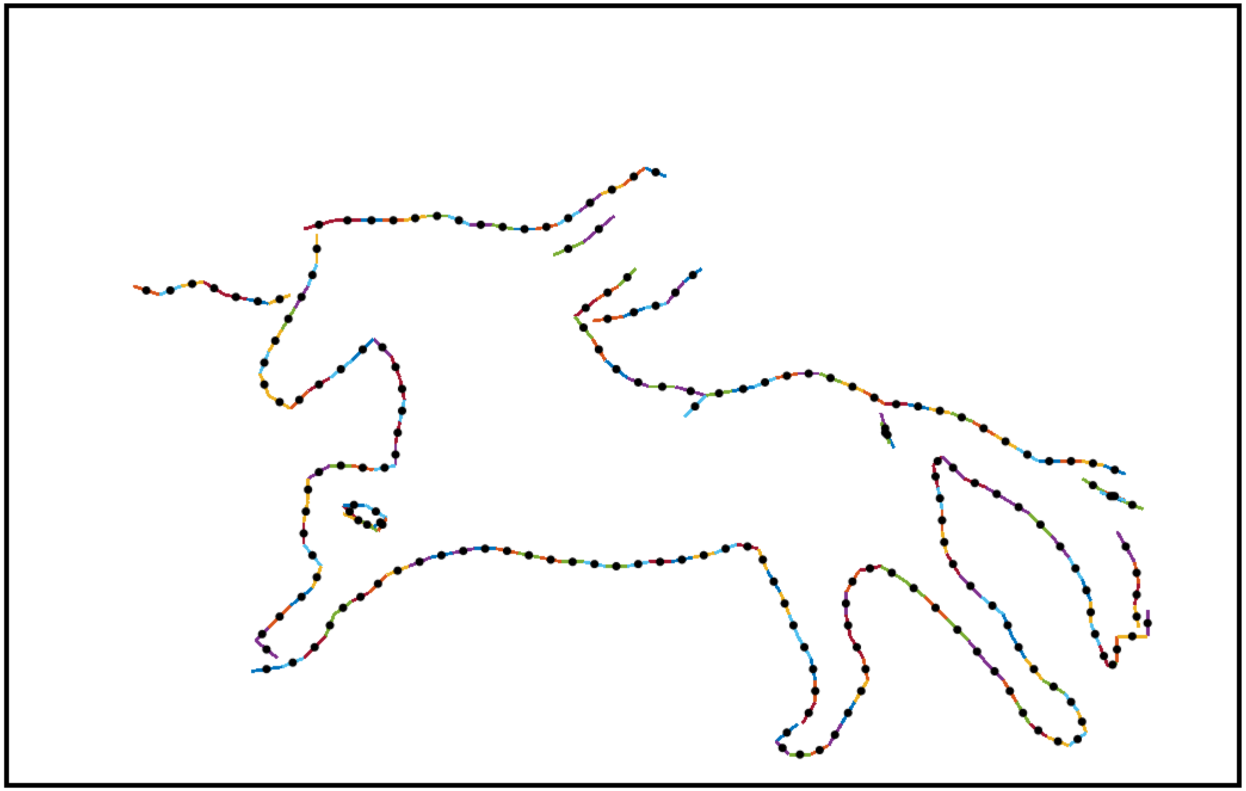
Overview of Proposed Approach

- Minimize Active Contour Energy
  - Additive energy representation
- Construct a directed graph
  - Graph nodes = edge segments
  - Graph edges = smooth curves
  - Edge weights = partial AC energies
- Use graph optimization
  - Floyd-Warshall all-pairs shortest path
  - Obtain closed paths = segmentations



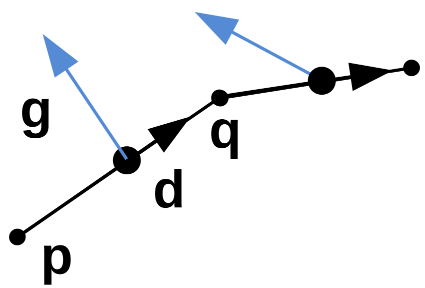
Constructing the Directed Graph

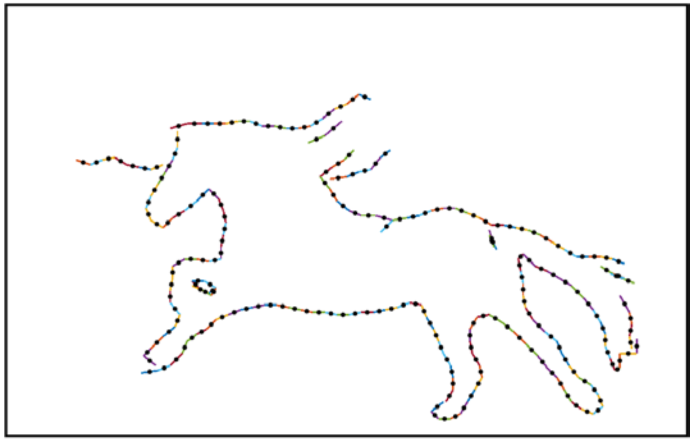
- Graph nodes:
  - Edge detection + linking
  - Cut into short segments
  - Segment centers = nodes

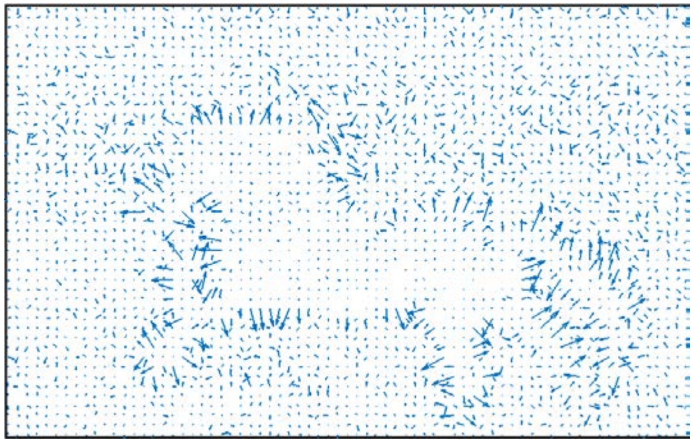


Constructing the Directed Graph

- Need a gradient field for orientation
  - Image gradient or
  - Gradient of distance transform



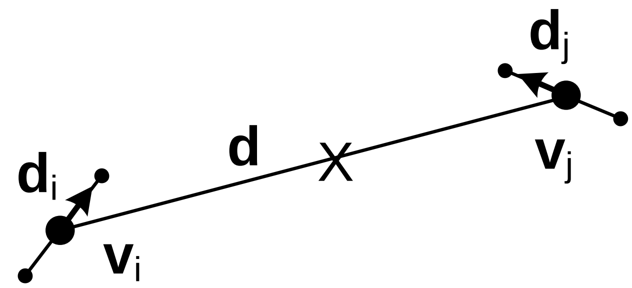


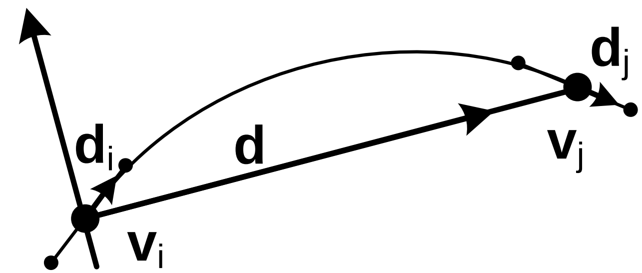


- Segment (node) orientation
  - Left hand rule
  - Gradient field is to the left of the segment

Constructing the Directed Graph

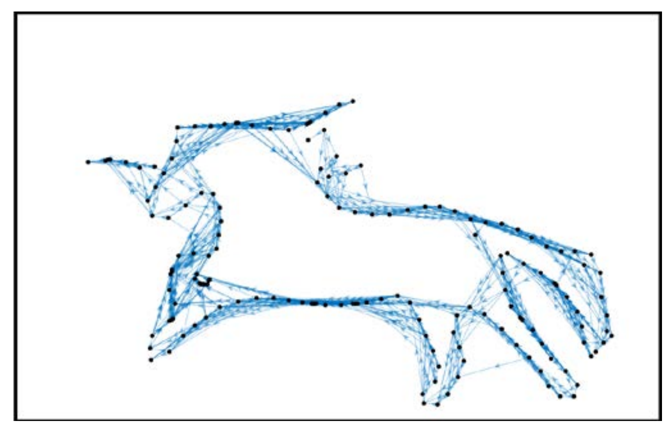
- Graph edges:
  - Segments up to a max distance
  - No edge between segments with incompatible directions
  - Edge direction = direction of segments
  - Construct smooth curve between segments as degree 3 polynomial
  - Remove edge if curve is too long



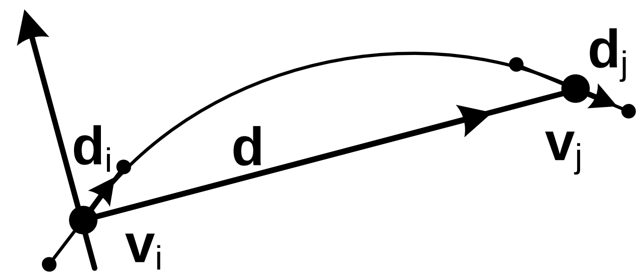


- Edge weight = curve energy

$$E(c) = \int_a^b \left[E_{data}(c(t)) + E_{smo}(c(t))\right] ||c'(t)|| dt$$



Constructing the Directed Graph



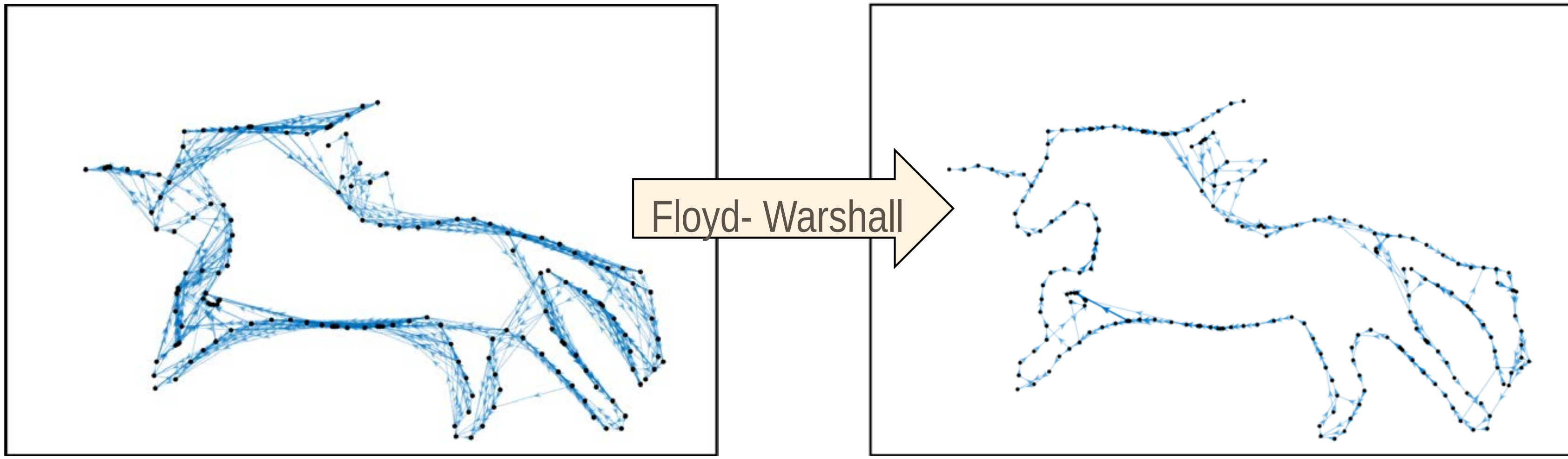
- Graph edge weight = curve energy

$$E(c) = \int_a^b \left[E_{data}(c(t)) + E_{smo}(c(t))\right] ||c'(t)|| dt$$

- Use curvature as smoothness:

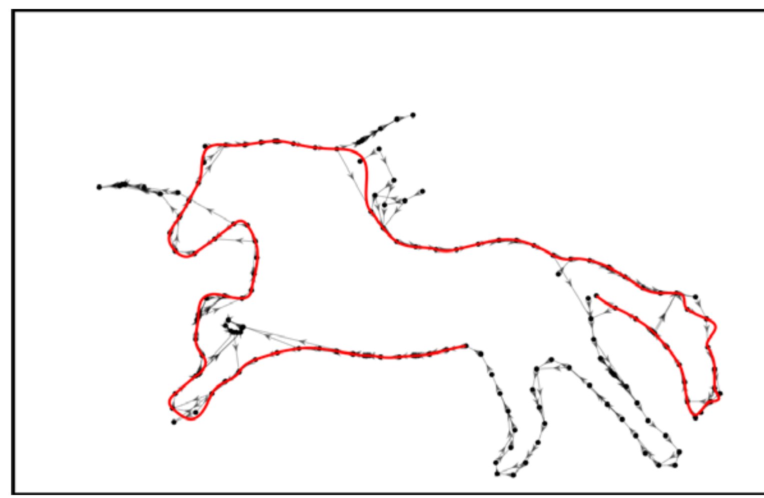
$$E_{smo}(c(t)) = |\kappa(c(t))| = \frac{|x'y - y'x|}{(x'^2 + y'^2)^{3/2}}$$

Directed Graph Optimization



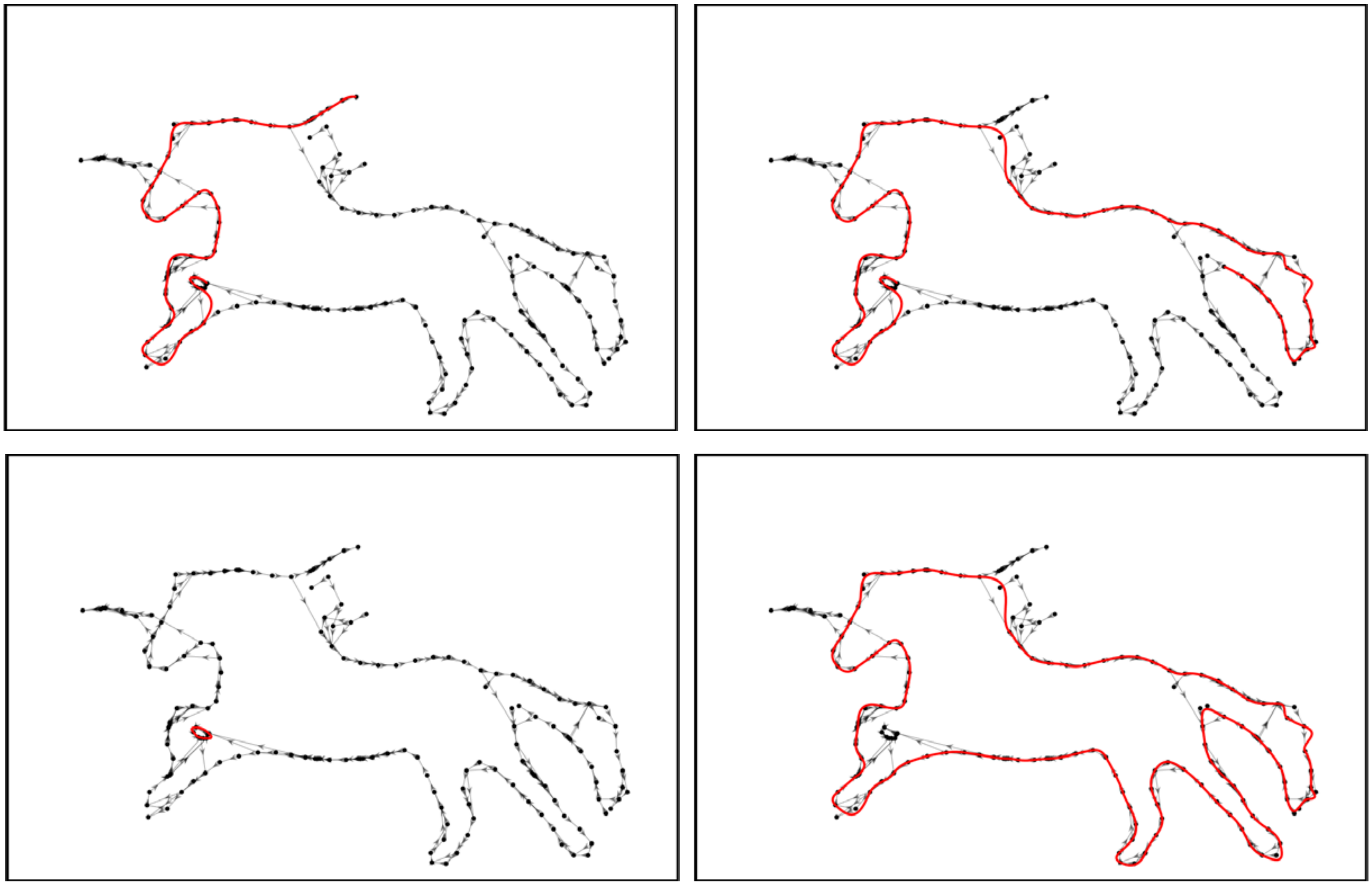
Next graph

- Use the Floyd-Warshall all-pair shortest path algorithm.
- Obtain
  - C_{ij} - minimum cost of the path from node i to j
  - N_{ij} - the next node of the minimum cost path = "Next" graph



One minimum cost path

Open and Closed Curves

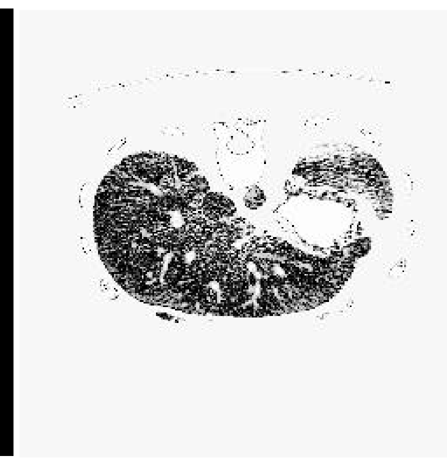


- Positive additive costs prefer shorter curves
- Normalize costs by the curve length

Datasets

- Weizmann Horse dataset
  - 328 horse images
  - Size about 300x200 pixels
  - Manual segmentations
- Liver dataset
  - 17 CT volumes
  - Manual liver segmentations
  - 11 slices each volume
    - Resized to 256x256
    - Total 187 images
  - Preprocessing:
    - CNN detection of rough liver
    - Intensity histogram
    - Liver intensity likelihood

CT slice



After preprocessing

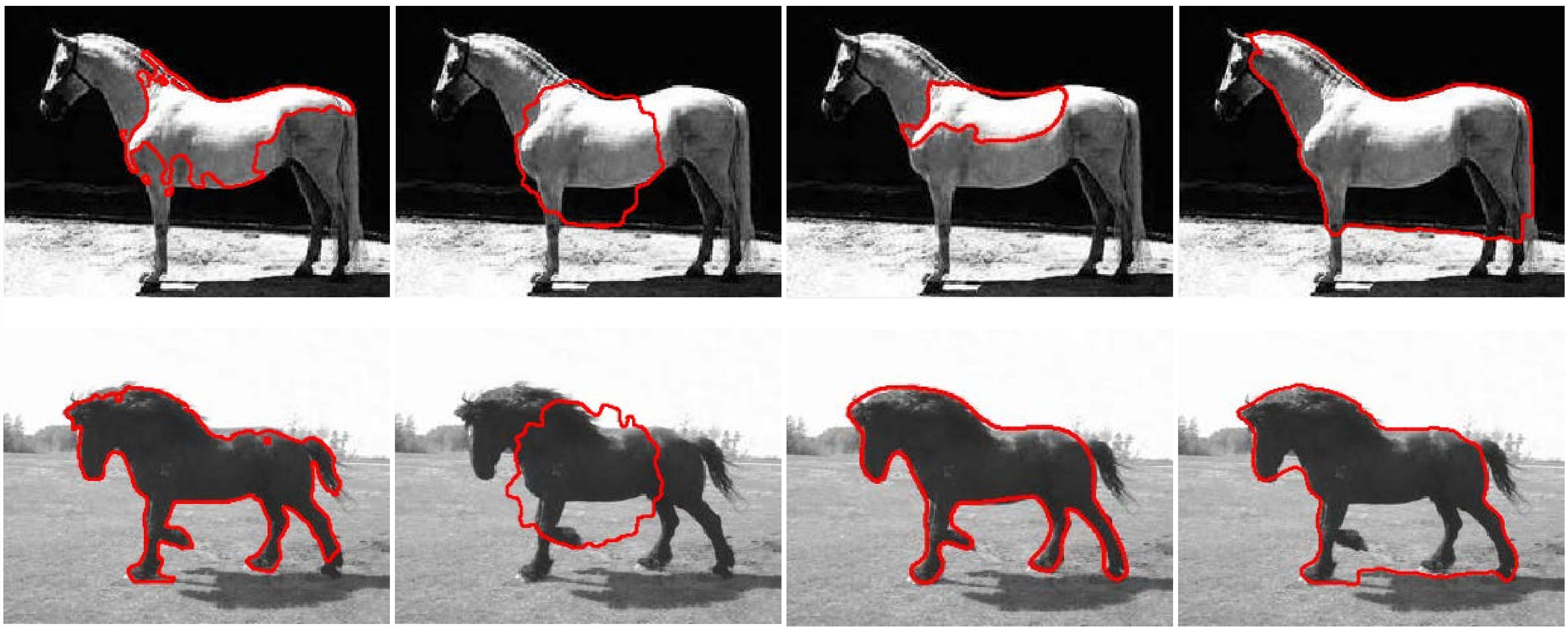
Quantitative Evaluation

- Methods:
  - Chan-Vese (Chan & Vese 2001)
  - Geodesic Active Contours (Caselles et al, 1997)
  - Ours using image gradient for edge directions
  - Ours using DT gradient for edge directions

Method	Horses	Time(s)	Livers	Time(s)
Initialization	51.49	-	83.40	-
Geodesic Active Contours	51.53	0.70	88.64	0.87
Chan-Vese	68.22	0.50	90.22	0.45
Ours w/ Image Gradient	46.76	0.59	89.46	0.03
Ours w/DT Gradient	61.81	0.70	89.99	0.02

Examples-Horses

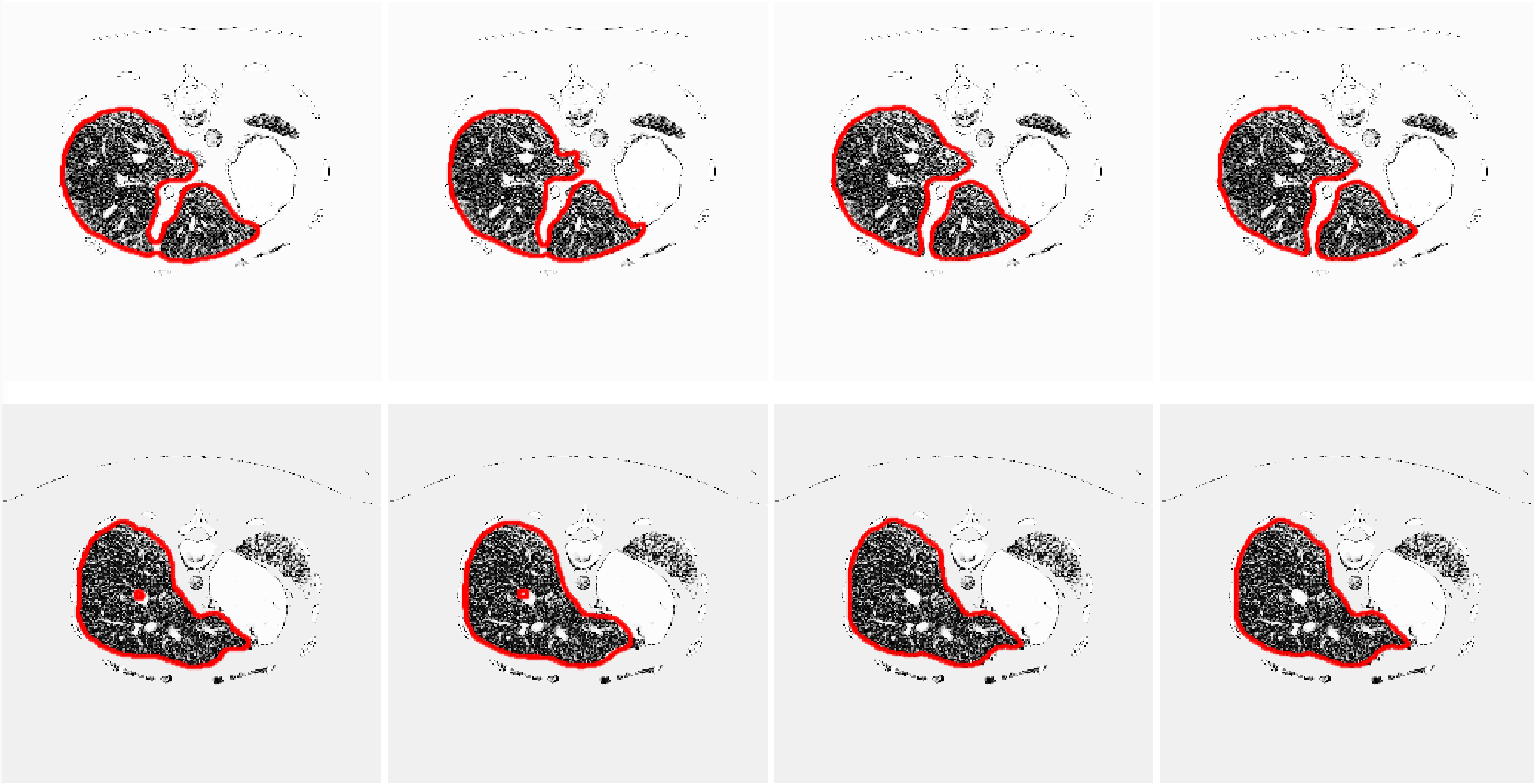
- Here distance transform (DT) is from the center pixel.



Chan-Vese | Geodesic Active Contours | Ours with image gradient | Ours with DT gradient

Examples-Livers

- DT is from a rough liver segmentation obtained by CNN



Conclusions

- A segmentation method for Active Contours
  - Additive AC energy using the arc length parametrization
  - Pieces of curves obtained as smooth polynomials
  - Directed curves to specify where the inside is.
  - Graph optimization for obtaining the result
- Pros:
  - Can impose constraints on the result
    - maximum curvature, min length, number of connected components, etc.
  - Unified treatment of open and closed curves
    - Partial segmentations
  - Does not depend on initialization
- Cons:
  - Hard to generalize to 3D