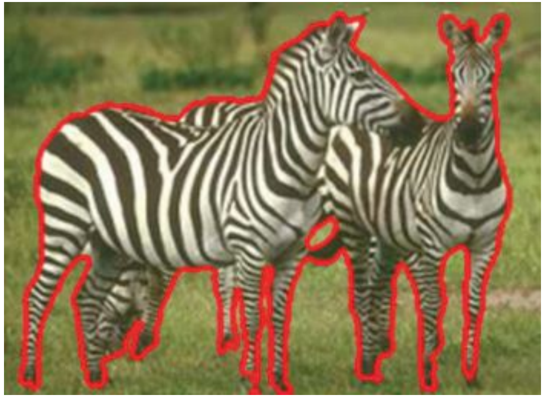
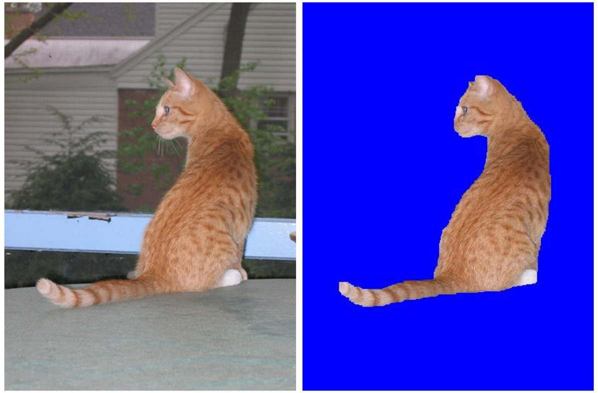
Foreground/Background Image Segmentation

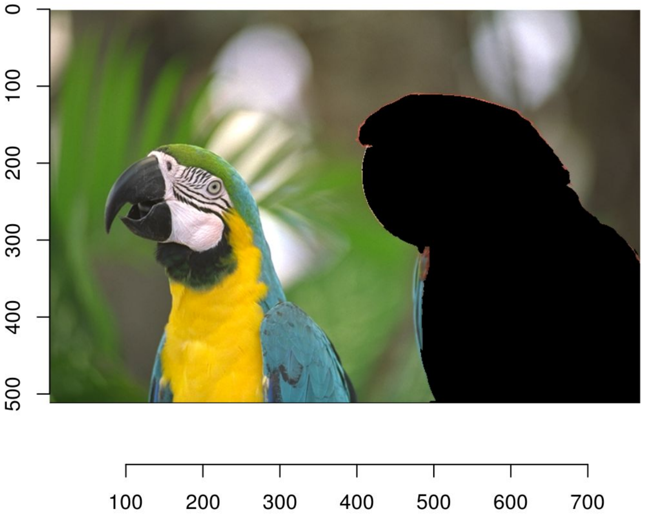
By Natalia Frumkin

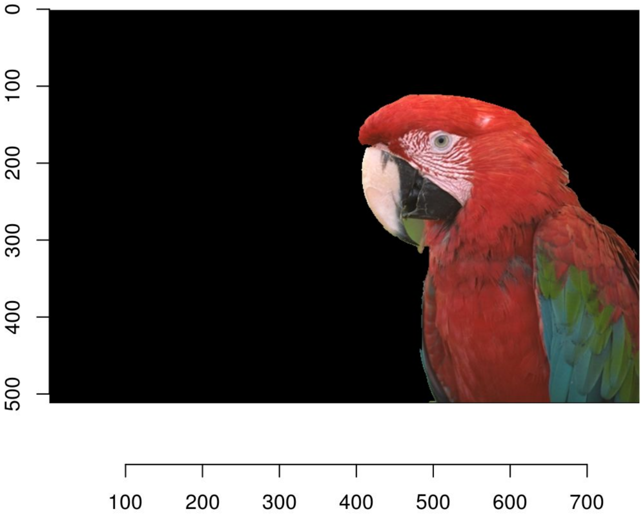




Problem Statement

Segment an image into background and foreground





Techniques

1. **Image Filtering**

2. **Max Flow Algorithms**

Otsu’s Method for Filtering

Big idea: automatic thresholding by minimizing intra-class variance

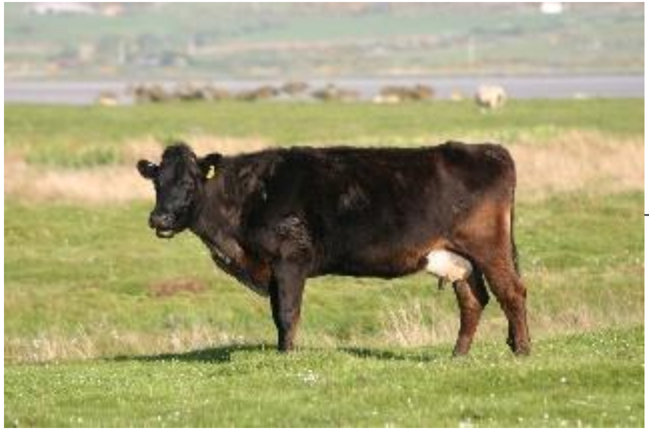
objective:

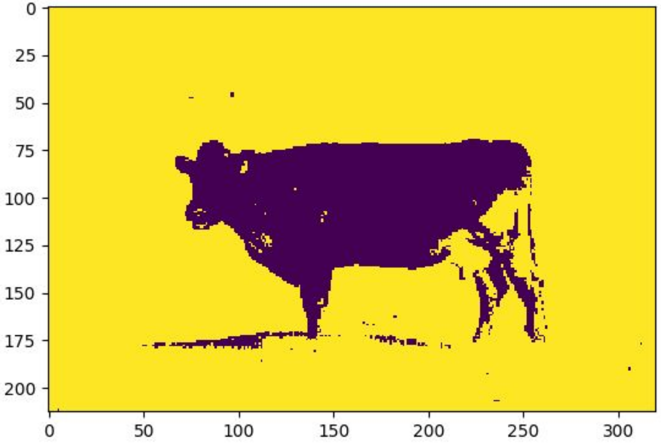
$$\sigma_w^2(t) = \omega_0(t)\sigma_0^2(t) + \omega_1(t)\sigma_1^2(t)$$

Class 0 variance

Class 1 variance

Results





Max-Flow Techniques

Energy Minimization

$$E(L) = \sum_{p \in \mathcal{P}} D_p(L_p) + \sum_{(p,q) \in \mathcal{N}} V_{p,q}(L_p, L_q).$$

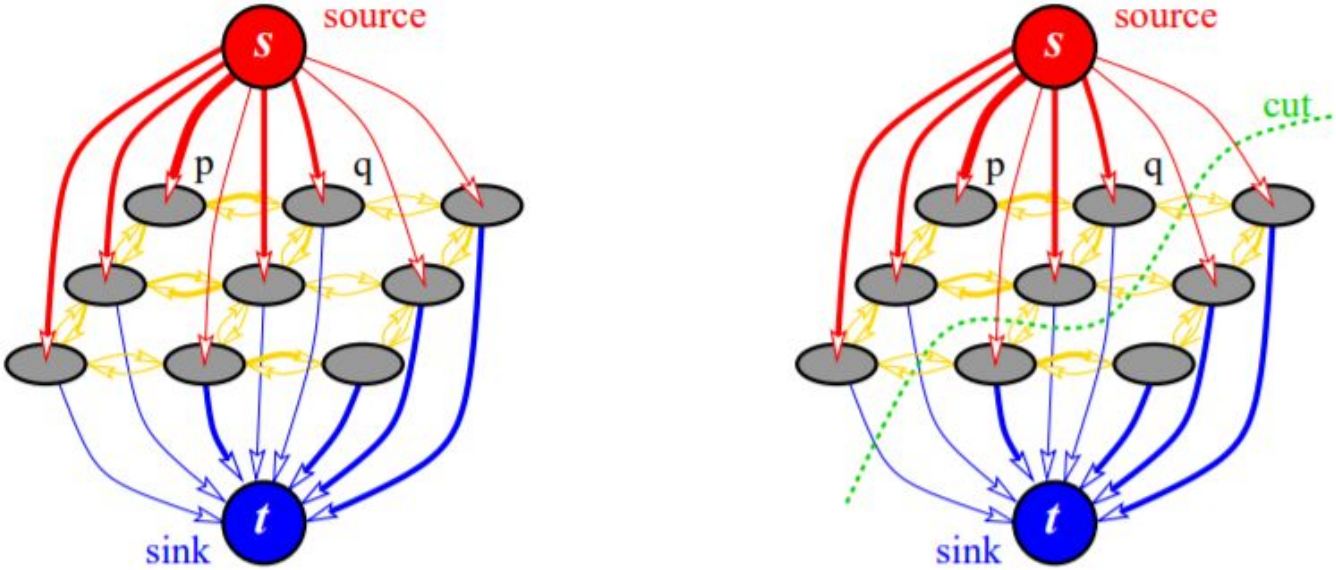
Data Penalty Term
Individual pixel intensity

Interaction Potential Term
Promotes spatial coherence between neighboring pixels

Interaction Potential Term
Promotes spatial coherence between neighboring pixels

Data Penalty Term
Individual pixel intensity

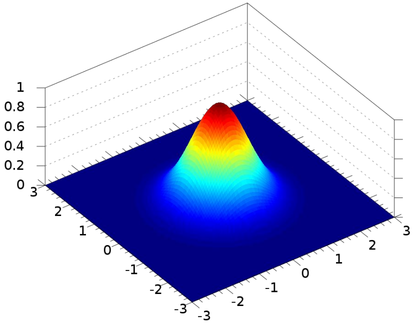
Graph Setup



(a) A graph $\mathcal{G}$

(b) A cut on $\mathcal{G}$

Graph Weights

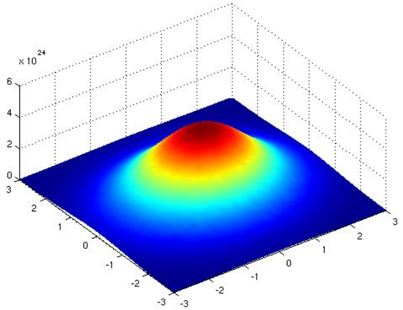


Intensity weights (Data Penalty)

distribution of individual pixels match src/sink labels

computed by:

$$exp\left(-\frac{(I_p - I_q)^2}{2\sigma^2}\right)$$

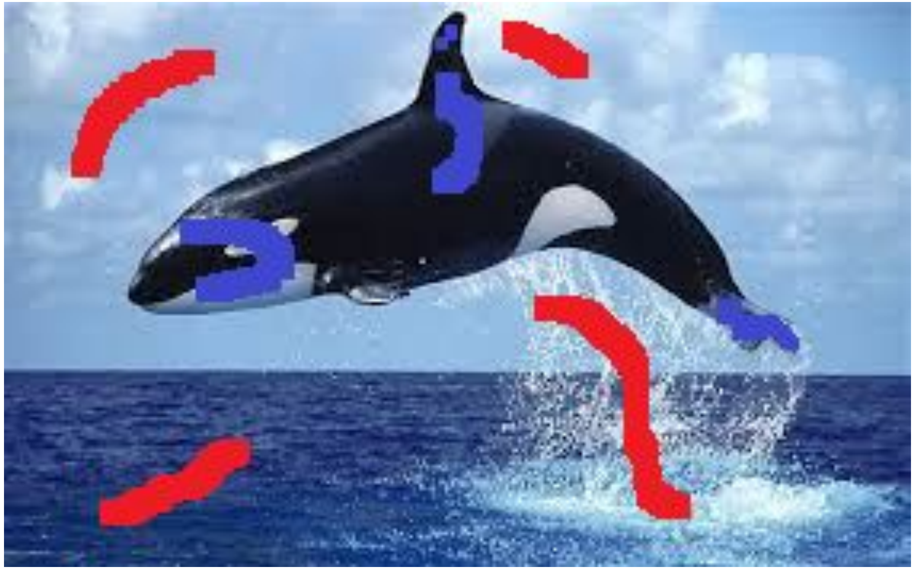


Interpixel weights (Interaction Potential)

distribution using intensity difference between neighboring pixels

computed through closed-form gaussian estimation

Annotation: Supervised Method



Ford Fulkerson Algorithm

1. Growth Phase

    Find path from source to sink using BFS

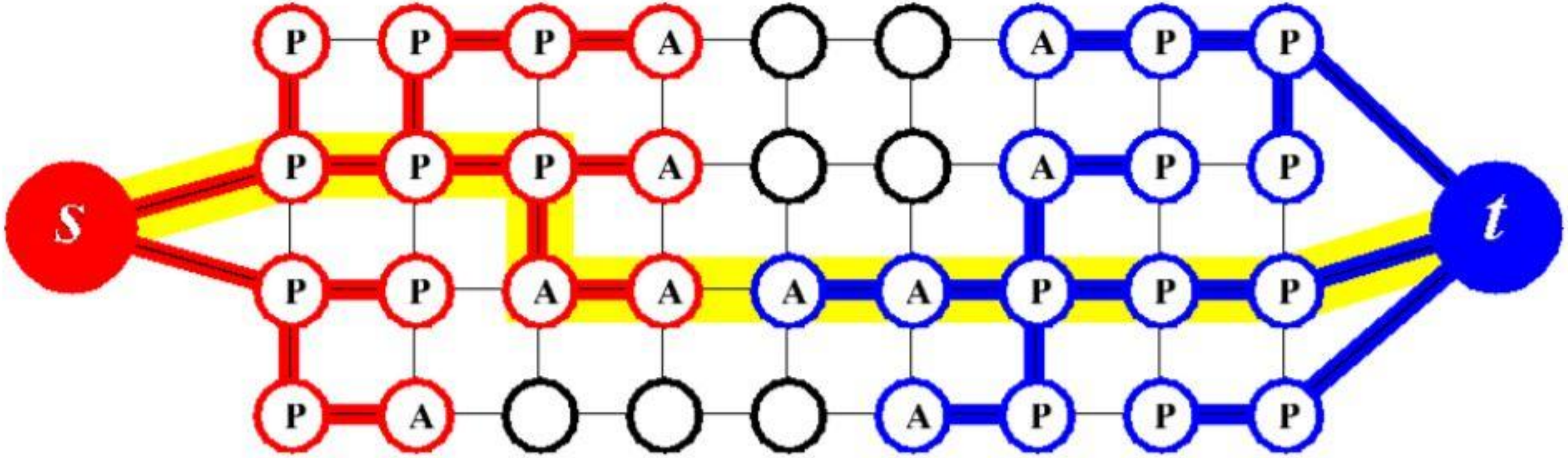
2. Augment Phase

    Push flow from s → t

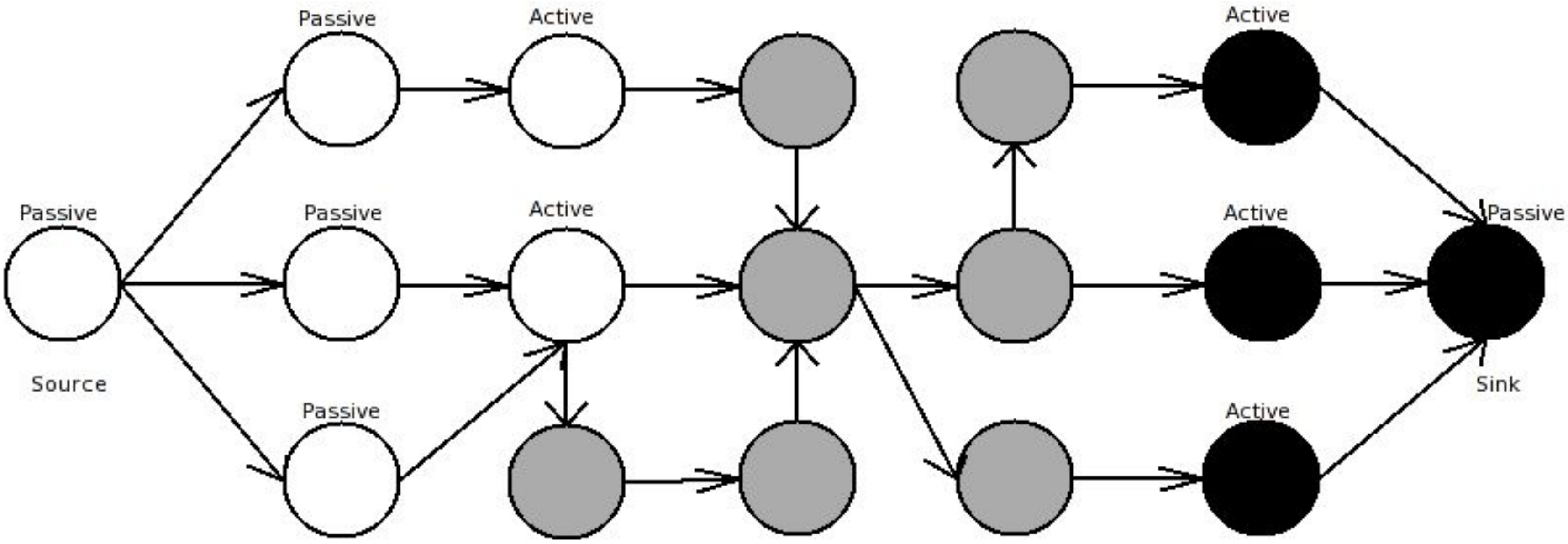
    Update residual graph and max-flow

Repeat until no paths can be found between source & sink

The Boykov-Kolmogorov Max-Flow Algorithm



The Boykov-Kolmogorov Max-Flow Algorithm



Example with labeled vertices and active/passive status

The Boykov-Kolmogorov Max-Flow Algorithm

1. Growth Phase

   Search trees S and T grow until they touch giving an s→ t path

2. Augment Phase

   Augment s → t path by pushing flow

3. Adoption Phase

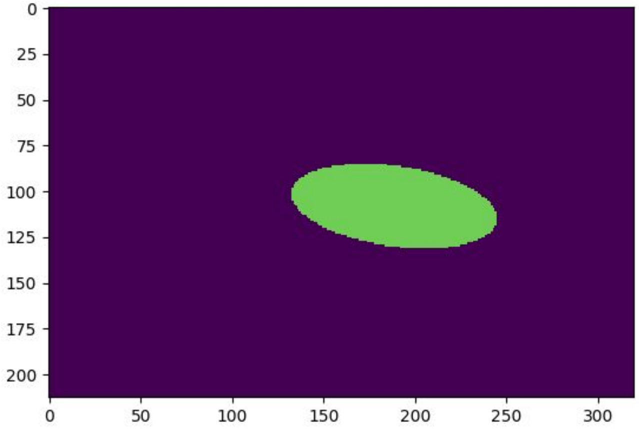
   Restore single-tree structure of sets S and T

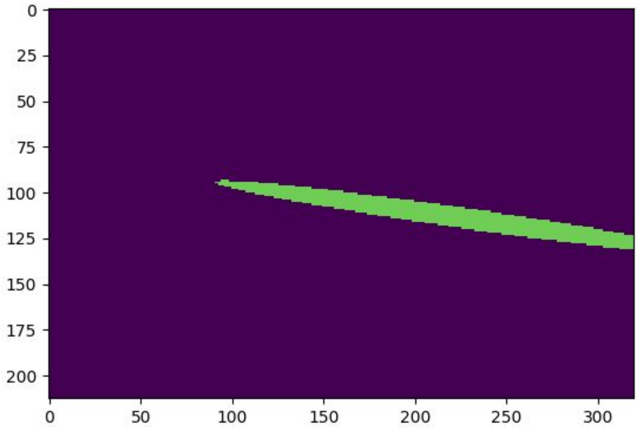
The Boykov-Kolmogorov Max-Flow Algorithm

Repeat three phases until end conditions are satisfied:

1) S & T Trees can not grow (no active nodes)
2) Trees are separated by saturated edges

Issues: Parameter Fitting





Next Steps:

- Grid search for optimal lambda and sigma
- Use Gaussian Mixture Models for weights

Runtime

Otsu’s Method

O(max(N_pixels, N_bins*N_bins) *real runtime: 0.0017 sec*

Ford-Fulkerson Method

$O(V*E^2)$ *real runtime: Too long :(*

Boykov-Kolmogorov Method

$O(V^2 * E * |C|)$ *real runtime: 10.6+0.00088 sec*

Demo!