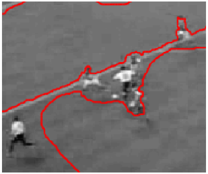
Multi-Object Tracking Through Clutter Using Graph Cuts

John Melonakos James Malcolm
Yogesh Rathi Allen Tannenbaum

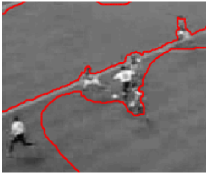
Electrical and Computer Engineering
Georgia Tech, Atlanta, GA, USA

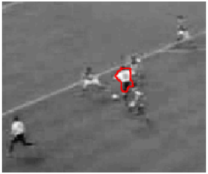
- The standard graph cut technique is a robust method for globally optimal image segmentation

- The standard graph cut technique is a robust method for globally optimal image segmentation
- Because of its global nature, it is prone to capture outlying areas similar to the object of interest.



- The standard graph cut technique is a robust method for globally optimal image segmentation
- Because of its global nature, it is prone to capture outlying areas similar to the object of interest.
- We propose a method to adaptively constrain the segmentation to a region of interest.





- Consider image segmentation as finding an energetically favorable pixel labeling

- Consider image segmentation as finding an energetically favorable pixel labeling
- Reformulate as min-cut problem

- Consider image segmentation as finding an energetically favorable pixel labeling
- Reformulate as min-cut problem
  - Fast polynomial time algorithms available

- Consider image segmentation as finding an energetically favorable pixel labeling
- Reformulate as min-cut problem
  - Fast polynomial time algorithms available
  - Globally optimal solutions

- Consider image segmentation as finding an energetically favorable pixel labeling
- Reformulate as min-cut problem
  - Fast polynomial time algorithms available
  - Globally optimal solutions
- Use multi-label graph cut technique: each object has its own label in addition to a label for background

- Standard energy formulation for some pixel labeling F:

$$E(F) = \sum_{p \in \mathcal{I}} R_p(f_p) + \lambda \sum_{(p,q) \in \mathcal{N}} B_{(p,q)}(f_p, f_q)$$

where p and q are pixels in image $\mathcal{I}$, $\mathcal{N}$ is the set of neighbor pairings, and λ adjusts the influence of smoothing

- Standard energy formulation for some pixel labeling F:

$$E(F) = \sum_{p \in \mathcal{I}} R_p(f_p) + \lambda \sum_{(p,q) \in \mathcal{N}} B_{(p,q)}(f_p, f_q)$$

  where p and q are pixels in image $\mathcal{I}$, $\mathcal{N}$ is the set of neighbor pairings, and λ adjusts the influence of smoothing

- A typical region term may be a function of a pixel's fit into a region's intensity distribution:

$$R_p(f) = -\ln P(\mathcal{I}_p | f)$$

- Standard energy formulation for some pixel labeling F:

$$E(F) = \sum_{p \in \mathcal{I}} R_p(f_p) + \lambda \sum_{(p,q) \in \mathcal{N}} B_{(p,q)}(f_p, f_q)$$

  where p and q are pixels in image $\mathcal{I}$, $\mathcal{N}$ is the set of neighbor pairings, and λ adjusts the influence of smoothing

- A typical region term may be a function of a pixel's fit into a region's intensity distribution:

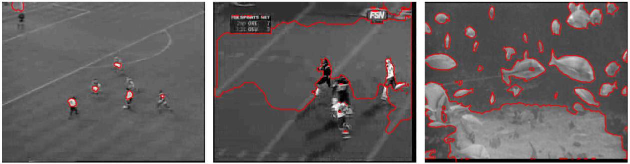
$$R_p(f) = -\ln P(\mathcal{I}_p | f)$$

- A typical boundary term may be a function of image contrast:

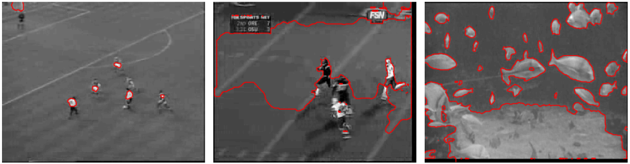
$$B_{(p,q)}(f_p, f_q) = \exp\left(\frac{-\|\mathcal{I}_p - \mathcal{I}_q\|^2}{2\sigma^2}\right) \frac{1}{\|p-q\|}$$

Outline

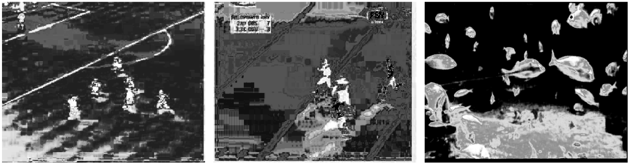
Why does the standard graph cut find the object throughout the image?



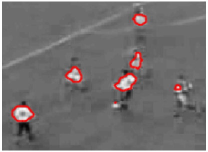
Why does the standard graph cut find the object throughout the image?

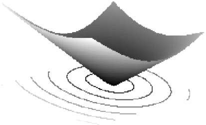


Answer: $R_p(f)$ looks at only intensity information and so evaluates to high likelihood *(white)* in several areas.

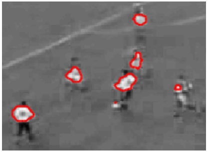


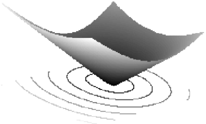
Form a basin of attraction by penalizing pixels based on their distance from the expected location





Form a basin of attraction by penalizing pixels based on their distance from the expected location





- We seek to maximize the probability of a labeling $P(f|\mathcal{I})$

$$P(f|\mathcal{I}) \propto P(\mathcal{I}|f)P(f)$$

- We seek to maximize the probability of a labeling $P(f|\mathcal{I})$

$$P(f|\mathcal{I}) \propto P(\mathcal{I}|f)P(f)$$

- Graph cuts minimizes energies, so we take the negative log-likelihood of this probability

- We seek to maximize the probability of a labeling $P(f|\mathcal{I})$

$$P(f|\mathcal{I}) \propto P(\mathcal{I}|f)P(f)$$

- Graph cuts minimizes energies, so we take the negative log-likelihood of this probability
- If we assume uniform spatial prior $P(f) = 1$, then it falls out of the negative log-likelihood of this expression

$$\begin{aligned} R_p(f) &= -\ln P(f|\mathcal{I}_p) \\ &\propto -\ln P(\mathcal{I}_p|f) - \ln P(f) \\ &= -\ln P(\mathcal{I}_p|f) \end{aligned}$$

which is the standard regional term

- In this work, we reintroduce that spatial prior claiming

$$-\ln P(f_p) \propto \phi(p)$$

where $\phi(p)$ is the distance from pixel p to the expected location

- In this work, we reintroduce that spatial prior claiming

$$-\ln P(f_p) \propto \phi(p)$$

where $\phi(p)$ is the distance from pixel p to the expected location

- Our regional term now becomes

$$\begin{aligned} R_p(f) &= -\ln P(\mathcal{I}_p|f) - \ln P(f) \\ &= -\ln P(\mathcal{I}_p|f) + \beta\phi(p) \end{aligned}$$

where we introduce β to adjust the influence of this spatial prior

- In this work, we reintroduce that spatial prior claiming

$$-\ln P(f_p) \propto \phi(p)$$

where $\phi(p)$ is the distance from pixel p to the expected location

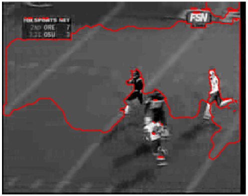
- Our regional term now becomes

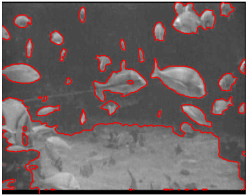
$$\begin{aligned} R_p(f) &= -\ln P(\mathcal{I}_p|f) - \ln P(f) \\ &= -\ln P(\mathcal{I}_p|f) + \beta\phi(p) \end{aligned}$$

where we introduce β to adjust the influence of this spatial prior

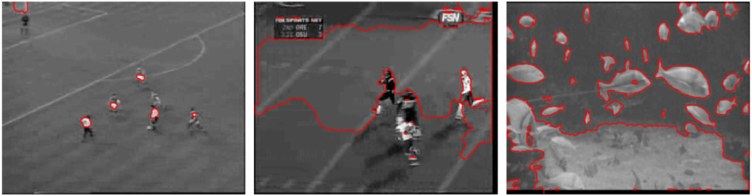
- We still consider background to have uniform spatial prior

Where we had...





Where we had...

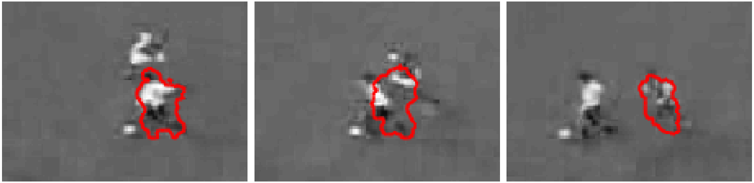


...we now have

Outline

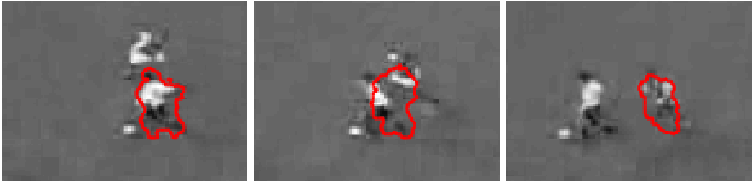
- Now we consider tracking applications.

- Now we consider tracking applications.
- Between frames, the object often moves outside the basin of attraction and we grab unintended objects



Without prediction [*play*]

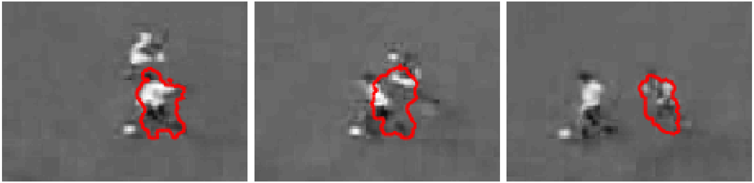
- Now we consider tracking applications.
- Between frames, the object often moves outside the basin of attraction and we grab unintended objects



Without prediction [*play*]

- Use a filter predicting the object location $\tilde{c}$ in each frame and center ϕ at this location.

- Now we consider tracking applications.
- Between frames, the object often moves outside the basin of attraction and we grab unintended objects



Without prediction [*play*]

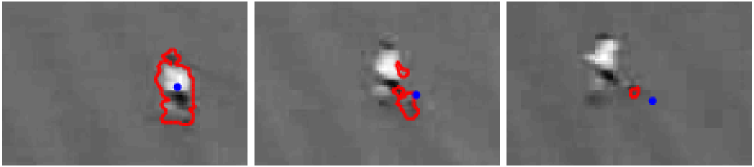
- Use a filter predicting the object location $\tilde{c}$ in each frame and center ϕ at this location.
  - For this work, we used a Kalman filter with either identity or first order linear models

Outline

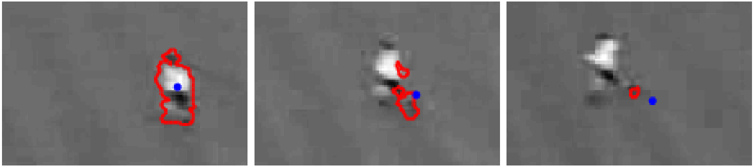
- Distance penalty constrains segmentation

- Distance penalty constrains segmentation
- Filter predicts where to center this distance penalty

- Distance penalty constrains segmentation
- Filter predicts where to center this distance penalty
- But what if the filter is wrong?



- Distance penalty constrains segmentation
- Filter predicts where to center this distance penalty
- But what if the filter is wrong?



- Solution: incorporate filter error into distance penalty construction

- Consider error as $e = ||\tilde{c} - c||$

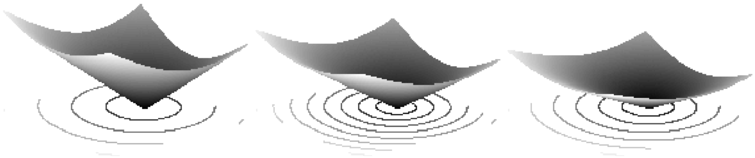
- Consider error as $e = ||\tilde{c} - c||$
  - $\tilde{c}$ is the centroid from prediction

- Consider error as $e = ||\tilde{c} - c||$
  - $\tilde{c}$ is the centroid from prediction
  - c is the centroid from segmentation

- Consider error as $e = ||\tilde{c} - c||$
    - $\tilde{c}$ is the centroid from prediction
    - c is the centroid from segmentation
- Scale distance penalty via exponential distribution:

$$\alpha(||\tilde{c} - c||) = \exp\left(-||\tilde{c} - c||^2/\rho^2\right)$$

Notice the basin of attraction widening as e increases:



Now our regional term incorporates this error feedback:

$$R_p(f) = -\ln P(\mathcal{I}_p|f) + \beta\, a(\|\tilde{c} - c\|)\phi(p)$$

Now our regional term incorporates this error feedback:

$$R_p(f) = -\ln P(\mathcal{I}_p|f) + \beta\, a(\|\tilde{c} - c\|)\phi(p)$$

Background still has a uniform spatial prior which falls out

$$R_p(bg) = -\ln P(\mathcal{I}_p|bg)$$

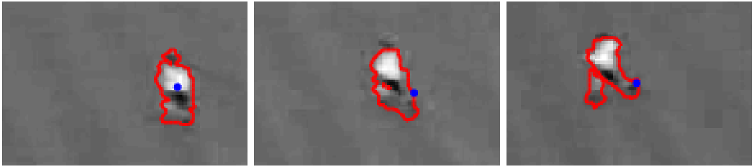
Now our regional term incorporates this error feedback:

$$R_p(f) = -\ln P(\mathcal{I}_p|f) + \beta\, a(\|\tilde{c} - c\|)\phi(p)$$

Background still has a uniform spatial prior which falls out

$$R_p(bg) = -\ln P(\mathcal{I}_p|bg)$$

We can now follow the soccer player despite inaccurate prediction



Outline

1. Introduction
2. Graph cut segmentation
3. Distance penalty
   - Spatial prior
   - Location predication
   - Adaptive scaling
4. Putting it all together
   - Algorithm
   - Results

1. Predict each object location

1. Predict each object location
2. Determine distance penalty scaling $\alpha(\|\tilde{c} - c\|)$ from previous frame

1. Predict each object location
2. Determine distance penalty scaling $\alpha(\|\tilde{c} - c\|)$ from previous frame
3. Form graph with edge weights R_p and $B_{(p,q)}$

1. Predict each object location
2. Determine distance penalty scaling $\alpha(\|\tilde{c} - c\|)$ from previous frame
3. Form graph with edge weights R_p and $B_{(p,q)}$
4. Graph cut segmentation

Outline

1. Introduction
2. Graph cut segmentation
3. Distance penalty
   - Spatial prior
   - Location predication
   - Adaptive scaling
4. Putting it all together
   - Algorithm
   - Results

Single soccer player [*play*]
Multiple soccer players [*play*]
Occlusion [*play*]
Interacting objects [*play*]
Large movements [*play*]
Rotation [*play*]
Football player [*play*]
Illumination changes [*play*]

- Distance penalty constrains segmentation

- Distance penalty constrains segmentation
- Incorporated as non-uniform spatial prior into regional term R_p

- Distance penalty constrains segmentation
- Incorporated as non-uniform spatial prior into regional term R_p
- Adaptive scale based on filter error

- Distance penalty constrains segmentation
- Incorporated as non-uniform spatial prior into regional term R_p
- Adaptive scale based on filter error

- Distance penalty constrains segmentation
- Incorporated as non-uniform spatial prior into regional term R_p
- Adaptive scale based on filter error

Questions?