Speeding up Semi-supervised On-line Boosting for Tracking

Martin Godec
Helmut Grabner
Christian Leistner
Horst Bischof

{ godec, leistner, bischof }@icg.tugraz.at
grabner@vision.ee.ethz.ch
On-line Boosting for Tracking
On-line Boosting for Tracking
Why does it fail?

Why does it fail?

Supervised Learning Algorithm for Unsupervised Problem?

Changing the Learning Strategy

• We have...
  – one reliably marked object position
  – unreliable object positions during tracking

• We want to...
  – track arbitrary/unknown objects
  – be adaptive to appearance changes
  – limit the risk of drifting

► One-shot Semi-Supervised On-line Learning
Outline

- Motivation
- Review of Semi-Supervised On-line Boosting
- Speeding up with Particle Filtering
- Evaluations and Results
- Conclusion & Outlook
Semi-Supervised Learning

Class A

Class B
Semi-Supervised Learning

Class A

Supervised Decision

Class B

Martin Godec
Stainz, 15.05.2008
Speeding Up Semi-supervised On-line Boosting for Tracking
Semi-Supervised Learning

Class A

Class B

Supervised Decision
Semi-Supervised Learning

Class A

Supervised Decision

Class B

Semi-Supervised Decision

Martin Godec
Stainz, 15.05.2008
Speeding Up Semi-supervised On-line Boosting for Tracking
Semi-Supervised Off-line Boosting

- Off-line Boosting incrementally selects the best classifier out of a pool based on a loss function
Semi-Supervised Off-line Boosting

- Off-line Boosting incrementally selects the best classifier out of a pool based on a loss function

Loss for Labeled Data
Semi-Supervised Off-line Boosting

- Off-line Boosting incrementally selects the best classifier out of a pool based on a loss function.
Semi-Supervised Off-line Boosting

- Off-line Boosting incrementally selects the best classifier out of a pool based on a loss function.

\[
h_n = \arg\min_{h_n} \left( \frac{1}{|\mathcal{X}^L|} \sum_{x \in \mathcal{X}^L} w_n(x, y) + \frac{1}{|\mathcal{X}^U|} \sum_{x \in \mathcal{X}^U} (p_n(x) - q_n(x)) \alpha_n h_n(x) \right)
\]

Loss for Labeled Data + Loss for Unlabeled Data
Semi-Supervised Off-line Boosting

- Off-line Boosting incrementally selects the best classifier out of a pool based on a loss function.

$$h_n = \arg\min_{h_n} \left( \frac{1}{|\mathcal{X}^L|} \sum_{x \in \mathcal{X}^L} w(x) \left( \sum_{x \in \mathcal{X}^U} (P_n(x) - q_n(x)) \alpha_n h_n(x) \right) \right)$$

- Loss for Labeled Data
- Loss for Unlabeled Data

How does this work for On-line Boosting?
Semi-supervised On-line Boosting (SSOB)
Semi-supervised On-line Boosting (SSOB)

Loss for Unlabeled Data
Semi-supervised On-line Boosting (SSOB)

\[ p_n(x) = w_n(x, 1) \frac{1}{|\mathcal{X}^L|} \sum_{x_i \in \mathcal{X}^+} S(x, x_i) + \frac{1}{|\mathcal{X}^U|} \sum_{x_i \in \mathcal{X}^U} S(x, x_i) e^{H_{n-1}(x_i) - H_{n-1}(x)} \]

Labeled/Unlabeled Pairs  Unlabeled Pairs
Semi-supervised On-line Boosting (SSOB)

\[
p_n(x) = w_n(x, 1) \frac{1}{|\mathcal{X}_L|} \sum_{x_i \in \mathcal{X}_+} S(x, x_i) + \frac{1}{|\mathcal{X}_U|} \sum_{x_i \in \mathcal{X}_-} S(x, x_i) - H_{n-1}(x)
\]

Labeled/Unlabeled Pairs
Unlabeled Pairs
Semi-supervised On-line Boosting (SSOB)

\[ p_n(x) = w_n(x, 1) \frac{1}{|\mathcal{X}^L|} \sum_{x_i \in \mathcal{X}^+} S(x, x_i) + \frac{1}{|\mathcal{X}^U|} \sum_{x_i \in \mathcal{X}^U} S(x, x_i) \]

Labeled/Unlabeled Pairs

Unlabeled Pairs

\[ \tilde{p}_n(x) \approx e^{-2H_{n-1}(x)} \sum_{x_i \in \mathcal{X}^+} S(x, x_i) \]
Semi-supervised On-line Boosting (SSOB)

\[
p_n(x) = w_n(x, 1) \frac{1}{|\mathcal{X}^L|} \sum_{x_i \in \mathcal{X}^+} S(x, x_i) + \frac{1}{|\mathcal{X}^U|} \sum_{x_i \in \mathcal{X}^U} S(1(x_i) - H_{n-1}(x))
\]

Labeled/Unlabeled Pairs  Unlabeled Pairs

\[
\tilde{p}_n(x) \approx e^{-2H_{n-1}(x)} \sum_{x_i \in \mathcal{X}^+} S(x, x_i) \approx e^{-H_{n-1}(x)} \sum_{x_i \in \mathcal{X}^+} S(x, x_i) \approx \frac{e^{-H_{n-1}(x)} e^{H_p(x)}}{e^{H_p(x)} + e^{-H_p(x)}}
\]

Prior Classifier

Martin Godec  Stainz, 15.05.2008  Speeding Up Semi-supervised On-line Boosting for Tracking
Semi-supervised On-line Boosting (SSOB)

\[
p_n(x) = w_n(x, 1) \frac{1}{|\mathcal{X}_L|} \sum_{x_i \in \mathcal{X}_+} S(x, x_i) + \frac{1}{|\mathcal{X}_U|} \sum_{x_i \in \mathcal{X}_-} S(x, x_i)
\]

Labeled/Unlabeled Pairs

Unlabeled Pairs

\[
\tilde{p}_n(x) \approx e^{-2H_{n-1}(x)} \sum_{x_i \in \mathcal{X}_+} S(x, x_i) \approx e^{-H_{n-1}(x)}
\]

Prior Classifier

\[
\tilde{z}_n(x) = \tilde{p}_n(x) - \tilde{q}_n(x) = \tanh(H^P(x)) - \tanh(H_{n-1}(x))
\]
Semi-supervised On-line Boosting (SSOB)

One training sample

Calculate importance and label of the sample
\[ x = \text{sign}(Z_i) \]
\[ \lambda = \text{abs}(Z_i) \]

Prior classifier

Update weight \( \alpha_1 \)

Current strong classifier \( h_{\text{Strong}} \)

Repeat for each training sample

Update weight \( \alpha_2 \)

Update weight \( \alpha_N \)
Speeding Up Semi-supervised On-line Boosting for Tracking
Open Issues?

• Speed
  – We have to evaluate two classifiers during update
  – What are the most time-consuming

• Prior Classifier
  – Can we create a strong classifier with only one sample?

• Update Strategies
  – Which patches should be presented to the classifier?
Outline

• Motivation
• Review of Semi-Supervised On-line Boosting
• Speeding up with Particle Filtering
• Evaluations and Results
• Conclusion & Outlook
Tracking Loop

- Prior classifier
- Evaluate classifier in neighborhood
- Create confidence map and analyse it
- Update classifier (tracker)
Tracking Loop

prior classifier

evaluate classifier in neighborhood
create confidence map and analyse it

update classifier (tracker)

Object Position

Martin Godec  Stainz, 15.05.2008  Speeding Up Semi-supervised On-line Boosting for Tracking
Particle Filtering

- Estimate State of a System
  - $N_p$ Particles spread over state space
  - Weighting of Particles by Evaluating their State Estimate
  - Also for non-Gaussian Processes

[Diagram taken from http://lia.deis.unibo.it/]
Regular Sampling (23fps)  
Particle-Filtering (55fps)
Outline

- Motivation
- Review of Semi-Supervised On-line Boosting
- Speeding up with Particle Filtering
- Evaluations and Results
- Conclusion & Outlook
Speedup using Particle Filtering

- Reduced evaluations per frame by 90% without decreased Performance
One-Shot Training

- One-shot Training
  - Create virtual examples
  - Simulate natural object behaviour
  - Extend the training set

<table>
<thead>
<tr>
<th></th>
<th>Simple</th>
<th>Scale</th>
<th>Rotation</th>
<th>Rot. &amp; Scale</th>
<th>Random</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tracking Rate</td>
<td>0.96</td>
<td>0.98</td>
<td>0.87</td>
<td>0.98</td>
<td>0.92</td>
</tr>
<tr>
<td>Precision</td>
<td>0.96</td>
<td>0.98</td>
<td>0.88</td>
<td>0.99</td>
<td>0.93</td>
</tr>
</tbody>
</table>
Update Patch Selection

- **Update Strategies**
  - Which samples should be presented to the learner?
  - Different Schemes have been evaluated

<table>
<thead>
<tr>
<th></th>
<th>Simple</th>
<th>Geometric</th>
<th>Random</th>
<th>Distance</th>
<th>Confidence</th>
</tr>
</thead>
<tbody>
<tr>
<td>Tracking Rate</td>
<td>0.96</td>
<td>0.98</td>
<td>0.87</td>
<td>0.98</td>
<td>0.92</td>
</tr>
<tr>
<td>Precision</td>
<td>0.96</td>
<td>0.98</td>
<td>0.88</td>
<td>0.99</td>
<td>0.93</td>
</tr>
</tbody>
</table>
Influence of Label Noise

- Influence of Label Noise
  - Manually misaligned update patches
  - Semi-Supervised Learning Algorithm is more robust than supervised one
Outline

- Motivation
- Review of Semi-Supervised On-line Boosting
- Speeding up with Particle Filtering
- Evaluations and Results
- Conclusion & Outlook
Conclusion

• Review and Evaluation of SSOB for Tracking
  – Limited Adaptivity
  – Limited Drifting
• Particle Filtering for Speedup
  – „Smarter“ Search-Space Sampling
  – Motion Information
• Initialization and Updates
  – Using virtual Samples
  – Evaluation of different schemes
Outlook

• Representation
  – Feature types

• Search-Space Sampling
  – DOF (Rotation, Scaling, Affine,...)
  – Refinement and Optimizations

• Learning Algorithm
  – Robust to Noise
  – Logit-Boost
  – Random Forests
Thank you for your attention!

QUESTIONS?