Online, semi-supervised learning for long-term interaction with object recognition systems

Alex Teichman and Sebastian Thrun

Department of Computer Science
Stanford University

July 12, 2012
What is the desired user interface for object recognition?

- Want autonomy with the option for user input.
- Online, active, semi-supervised learning...

Alex Teichman and Sebastian Thrun
Online, semi-supervised learning // RSS2012
The big picture

What is the desired user interface for object recognition?

Want autonomy with the option for user input.

Online, active, semi-supervised learning...
What is the desired user interface for object recognition?

Want autonomy with the option for user input.

Online, active, semi-supervised learning...
Static train/test framework

Table 1. Breakdown of the dataset by class. Tracks were collected from busy streets and intersections.

<table>
<thead>
<tr>
<th>Set</th>
<th>Car</th>
<th>Pedestrian</th>
<th>Bicyclist</th>
<th>Background</th>
<th>All</th>
</tr>
</thead>
<tbody>
<tr>
<td>Training</td>
<td>904</td>
<td>205</td>
<td>187</td>
<td>6585</td>
<td>7881</td>
</tr>
<tr>
<td>Testing</td>
<td>847</td>
<td>112</td>
<td>140</td>
<td>4936</td>
<td>6035</td>
</tr>
<tr>
<td>Total</td>
<td>1751</td>
<td>317</td>
<td>327</td>
<td>11,521</td>
<td>13,916</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Set</th>
<th>Car</th>
<th>Pedestrian</th>
<th>Bicyclist</th>
<th>Background</th>
<th>All</th>
</tr>
</thead>
<tbody>
<tr>
<td>Training</td>
<td>92,255</td>
<td>32,281</td>
<td>31,165</td>
<td>532,760</td>
<td>688,461</td>
</tr>
<tr>
<td>Testing</td>
<td>59,173</td>
<td>22,203</td>
<td>25,410</td>
<td>530,917</td>
<td>637,703</td>
</tr>
<tr>
<td>Total</td>
<td>151,428</td>
<td>54,484</td>
<td>56,575</td>
<td>1,063,677</td>
<td>1,326,164</td>
</tr>
</tbody>
</table>

Rigorous evaluation and comparison
Experimental setup
Occasional user interaction
Infinite unlabeled data stream

We don’t want to overfit to this framework!
### Table 1. Breakdown of the dataset by class. Tracks were collected from busy streets and intersections.

<table>
<thead>
<tr>
<th>Set</th>
<th>Car</th>
<th>Pedestrian</th>
<th>Bicyclist</th>
<th>Background</th>
<th>All</th>
</tr>
</thead>
<tbody>
<tr>
<td>Training</td>
<td>904</td>
<td>205</td>
<td>187</td>
<td>6585</td>
<td>7881</td>
</tr>
<tr>
<td>Testing</td>
<td>847</td>
<td>112</td>
<td>140</td>
<td>4936</td>
<td>6025</td>
</tr>
<tr>
<td>Total</td>
<td>1751</td>
<td>317</td>
<td>327</td>
<td>11,521</td>
<td>13,916</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Set</th>
<th>Car</th>
<th>Pedestrian</th>
<th>Bicyclist</th>
<th>Background</th>
<th>All</th>
</tr>
</thead>
<tbody>
<tr>
<td>Training</td>
<td>92,255</td>
<td>32,281</td>
<td>31,165</td>
<td>532,760</td>
<td>688,461</td>
</tr>
<tr>
<td>Testing</td>
<td>59,173</td>
<td>22,203</td>
<td>25,410</td>
<td>530,917</td>
<td>637,703</td>
</tr>
<tr>
<td>Total</td>
<td>151,428</td>
<td>54,484</td>
<td>56,575</td>
<td>1,063,677</td>
<td>1,326,164</td>
</tr>
</tbody>
</table>

- Rigorous evaluation and comparison
- Experimental setup
- Occasional user interaction
- Infinite unlabeled data stream

We don’t want to overfit to this framework!
We don’t want to overfit to this framework!

Table 1. Breakdown of the dataset by class. Tracks were collected from busy streets and intersections.

<table>
<thead>
<tr>
<th>Set</th>
<th>Car</th>
<th>Pedestrian</th>
<th>Bicyclist</th>
<th>Background</th>
<th>All</th>
</tr>
</thead>
<tbody>
<tr>
<td>Training</td>
<td>904</td>
<td>205</td>
<td>187</td>
<td>6585</td>
<td>7881</td>
</tr>
<tr>
<td>Testing</td>
<td>847</td>
<td>112</td>
<td>140</td>
<td>4936</td>
<td>6025</td>
</tr>
<tr>
<td>Total</td>
<td>1751</td>
<td>317</td>
<td>327</td>
<td>11,521</td>
<td>13,916</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Set</th>
<th>Car</th>
<th>Pedestrian</th>
<th>Bicyclist</th>
<th>Background</th>
<th>All</th>
</tr>
</thead>
<tbody>
<tr>
<td>Training</td>
<td>92,255</td>
<td>32,281</td>
<td>31,165</td>
<td>532,760</td>
<td>688,461</td>
</tr>
<tr>
<td>Testing</td>
<td>59,173</td>
<td>22,203</td>
<td>25,410</td>
<td>530,917</td>
<td>637,703</td>
</tr>
<tr>
<td>Total</td>
<td>151,428</td>
<td>54,484</td>
<td>56,575</td>
<td>1,063,677</td>
<td>1,326,164</td>
</tr>
</tbody>
</table>

- Rigorous evaluation and comparison
- Experimental setup
- Occasional user interaction
- Infinite unlabeled data stream
We don't want to overfit to this framework!

Rigorous evaluation and comparison
Experimental setup

Occasional user interaction
Infinite unlabeled data stream

Table 1. Breakdown of the dataset by class. Tracks were collected from busy streets and intersections.

<table>
<thead>
<tr>
<th>Set</th>
<th>Car</th>
<th>Pedestrian</th>
<th>Bicyclist</th>
<th>Background</th>
<th>All</th>
</tr>
</thead>
<tbody>
<tr>
<td>Training</td>
<td>904</td>
<td>205</td>
<td>187</td>
<td>6585</td>
<td>7881</td>
</tr>
<tr>
<td>Testing</td>
<td>847</td>
<td>112</td>
<td>140</td>
<td>4936</td>
<td>6025</td>
</tr>
<tr>
<td>Total</td>
<td>1751</td>
<td>317</td>
<td>327</td>
<td>11,521</td>
<td>13,916</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Set</th>
<th>Car</th>
<th>Pedestrian</th>
<th>Bicyclist</th>
<th>Background</th>
<th>All</th>
</tr>
</thead>
<tbody>
<tr>
<td>Training</td>
<td>92,255</td>
<td>32,281</td>
<td>31,165</td>
<td>532,760</td>
<td>688,461</td>
</tr>
<tr>
<td>Testing</td>
<td>59,173</td>
<td>22,203</td>
<td>25,410</td>
<td>530,917</td>
<td>637,703</td>
</tr>
<tr>
<td>Total</td>
<td>151,428</td>
<td>54,484</td>
<td>56,575</td>
<td>1,063,677</td>
<td>1,326,164</td>
</tr>
</tbody>
</table>
Static train/test framework

Table 1. Breakdown of the dataset by class. Tracks were collected from busy streets and intersections.

<table>
<thead>
<tr>
<th></th>
<th>Number of tracks</th>
<th></th>
<th>Number of frames</th>
</tr>
</thead>
<tbody>
<tr>
<td></td>
<td>Set</td>
<td>Car</td>
<td>Pedestrian</td>
</tr>
<tr>
<td></td>
<td>Training</td>
<td>904</td>
<td>205</td>
</tr>
<tr>
<td></td>
<td>Testing</td>
<td>847</td>
<td>112</td>
</tr>
<tr>
<td></td>
<td>Total</td>
<td>1751</td>
<td>317</td>
</tr>
</tbody>
</table>

- Rigorous evaluation and comparison
- Experimental setup
- Occasional user interaction
- Infinite unlabeled data stream

We don’t want to overfit to this framework!
Table 1. Breakdown of the dataset by class. Tracks were collected from busy streets and intersections.

<table>
<thead>
<tr>
<th>Set</th>
<th>Car</th>
<th>Pedestrian</th>
<th>Bicyclist</th>
<th>Background</th>
<th>All</th>
</tr>
</thead>
<tbody>
<tr>
<td>Training</td>
<td>904</td>
<td>205</td>
<td>187</td>
<td>6585</td>
<td>7881</td>
</tr>
<tr>
<td>Testing</td>
<td>847</td>
<td>112</td>
<td>140</td>
<td>4936</td>
<td>6035</td>
</tr>
<tr>
<td>Total</td>
<td>1751</td>
<td>317</td>
<td>327</td>
<td>11,521</td>
<td>13,916</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Set</th>
<th>Car</th>
<th>Pedestrian</th>
<th>Bicyclist</th>
<th>Background</th>
<th>All</th>
</tr>
</thead>
<tbody>
<tr>
<td>Training</td>
<td>92,255</td>
<td>32,281</td>
<td>31,165</td>
<td>532,760</td>
<td>688,461</td>
</tr>
<tr>
<td>Testing</td>
<td>59,173</td>
<td>22,203</td>
<td>25,410</td>
<td>530,917</td>
<td>637,703</td>
</tr>
<tr>
<td>Total</td>
<td>151,428</td>
<td>54,484</td>
<td>56,575</td>
<td>1,063,677</td>
<td>1,326,164</td>
</tr>
</tbody>
</table>

- **Rigorous evaluation and comparison**
- **Experimental setup**
- **Occasional user interaction**
- **Infinite unlabeled data stream**

We don’t want to overfit to this framework!
Table 1. Breakdown of the dataset by class. Tracks were collected from busy streets and intersections.

<table>
<thead>
<tr>
<th>Set</th>
<th>Car</th>
<th>Pedestrian</th>
<th>Bicyclist</th>
<th>Background</th>
<th>All</th>
</tr>
</thead>
<tbody>
<tr>
<td>Training</td>
<td>904</td>
<td>205</td>
<td>187</td>
<td>6585</td>
<td>7881</td>
</tr>
<tr>
<td>Testing</td>
<td>847</td>
<td>112</td>
<td>140</td>
<td>4936</td>
<td>6025</td>
</tr>
<tr>
<td>Total</td>
<td>1751</td>
<td>317</td>
<td>327</td>
<td>11,521</td>
<td>13,916</td>
</tr>
</tbody>
</table>

<table>
<thead>
<tr>
<th>Set</th>
<th>Car</th>
<th>Pedestrian</th>
<th>Bicyclist</th>
<th>Background</th>
<th>All</th>
</tr>
</thead>
<tbody>
<tr>
<td>Training</td>
<td>92,255</td>
<td>32,281</td>
<td>31,165</td>
<td>532,760</td>
<td>688,461</td>
</tr>
<tr>
<td>Testing</td>
<td>59,173</td>
<td>22,203</td>
<td>25,410</td>
<td>530,917</td>
<td>637,703</td>
</tr>
<tr>
<td>Total</td>
<td>151,428</td>
<td>54,484</td>
<td>56,575</td>
<td>1,063,677</td>
<td>1,326,164</td>
</tr>
</tbody>
</table>

- Rigorous evaluation and comparison
- Experimental setup
- Occasional user interaction
- Infinite unlabeled data stream

We don’t want to overfit to this framework!
Object recognition approaches - sliding window & tracking-by-detection

Spinello and Arras

Spinello, Stachniss, and Burgard
Object recognition approaches - semantic segmentation

Douillard et al.

Combining sliding windows and semantic segmentation: Lai et al.
Object recognition approaches - keypoint matching

Solutions in Perception Challenge

Collet et al.
Problem decomposition

Segmentation — Tracking — Track classification
Problem decomposition

Segmentation — Tracking — Track classification
Problem decomposition

Segmentation — Tracking — Track classification
Descriptors

- 29 different descriptor spaces
- $x \in \mathbb{R}^{\sim 4000}$

- Oriented bounding box size
- Spin images
- HOG descriptors computed on virtual orthographic camera images
Tracks

<table>
<thead>
<tr>
<th>Classifier</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>car</td>
<td>-16.53</td>
</tr>
<tr>
<td>pedestrian</td>
<td>-33.12</td>
</tr>
<tr>
<td>bicyclist</td>
<td>-41.31</td>
</tr>
</tbody>
</table>

363 points
Track 100 of 1477
Cloud 52 of 91
Wrap mode: on
Tracking-based semi-supervised learning

- Large, automatically-labeled background dataset is provided. Often this is easy to collect.
- Only positive examples are inducted during semi-supervised learning.
Tracking-based semi-supervised learning

- Unsupervised method given millions of additional unlabeled examples.
- Track classification accuracy is reported. (This does not include segmentation and tracking errors.)

Alex Teichman and Sebastian Thrun
Online, semi-supervised learning // RSS2012
Tracking-based semi-supervised learning

- Three hand-labeled training examples of each class + millions of unlabeled examples used to generate these results.
- Outlines are tracked objects. Track classifications are computed offline.
- White outlines are tracked objects classified as neither person, bicyclist, or car.
Offline to online

Algorithm 1 Tracking-based semi-supervised learning

\( \tau \) is a confidence threshold chosen by hand
\( S \) is a small set of seed tracks, labeled by hand
\( U \) is a large set of unlabeled tracks
\( B \) is a large set of background tracks
\( W \) is a working set, initially empty

\[ W := S \cup B \]

repeat

Train frame classifier \( C \) on \( W \)
\[ W := S \cup B \]

for \( u \in U \) do

Classify track \( u \) using \( C \)
\( c := \text{confidence}(u) \)
\( l := \text{classification}(u) \)

if \( c \geq \tau \) and \( l \neq \text{“background”} \) then

Add \( u \) to \( W \) with label \( l \)

end if

end for

until converged
Modularity

Segmentation
- Connected components
- Background subtraction

Tracking
- Kalman filters

Classification
- Boosting
- Logistic regression, stochastic gradient descent

- Discriminative segmentation and tracking
Logistic regression & stochastic gradient descent

- **Motivation**
- **Algorithm**
- **Experiments**

- Parametric
- Fast to train and evaluate
- Easy to incrementally train

\[
x \in \mathbb{R}^n, y \in \{-1, +1\}
\]

\[
P(y|x) = \frac{1}{1 + \exp(-yw^T x)}
\]

\[
\begin{align*}
\text{maximize} & \quad \prod_m P(y^{(m)}|x^{(m)}) \\
\text{minimize} & \quad \sum_{m=1}^{M} \log(1 + \exp(-y^{(m)}w^T x^{(m)}))
\end{align*}
\]

- \(M\) might be giant, or you might not have access to them all at one time.
- Stochastic gradient descent: take gradient steps using just small subsets of the data.
- ... but this fails badly if applied without thinking.

Alex Teichman and Sebastian Thrun

Online, semi-supervised learning // RSS2012
Logistic regression & stochastic gradient descent

- Parametric
- Fast to train and evaluate
- Easy to incrementally train

\[ x \in \mathbb{R}^n, y \in \{-1, +1\} \]

\[ P(y|x) = \frac{1}{1 + \exp(-yw^T x)} \]

maximize

\[ \prod_w \prod_{m} P(y^{(m)}|x^{(m)}) \]

minimize

\[ \sum_w \sum_{m=1}^{M} \log(1 + \exp(-y^{(m)}w^T x^{(m)})) \]

- \( M \) might be giant, or you might not have access to them all at one time.
- Stochastic gradient descent: take gradient steps using just small subsets of the data.
- ... but this fails badly if applied without thinking.
Logistic regression & stochastic gradient descent

- Parametric
- Fast to train and evaluate
- Easy to incrementally train

\[ x \in \mathbb{R}^n, y \in \{-1, +1\} \]

\[ P(y|x) = \frac{1}{1 + \exp(-yw^T x)} \]

\[
\begin{align*}
\text{maximize} & \quad \prod_m P(y^{(m)}|x^{(m)}) \\
\text{minimize} & \quad \sum_{m=1}^M \log(1 + \exp(-y^{(m)}w^T x^{(m)}))
\end{align*}
\]

- \( M \) might be giant, or you might not have access to them all at one time.
- Stochastic gradient descent: take gradient steps using just small subsets of the data.
- ... but this fails badly if applied without thinking.

Alex Teichman and Sebastian Thrun

Online, semi-supervised learning // RSS2012
Logistic regression & stochastic gradient descent

- Parametric
- Fast to train and evaluate
- Easy to incrementally train

\[ x \in \mathbb{R}^n, y \in \{-1, +1\} \]

\[ P(y|x) = \frac{1}{1 + \exp(-yw^T x)} \]

\[ \text{maximize} \quad \prod_{m} P(y^{(m)}|x^{(m)}) \]

\[ \text{minimize} \quad \sum_{m=1}^{M} \log(1 + \exp(-y^{(m)}w^T x^{(m)})) \]

- \( M \) might be giant, or you might not have access to them all at one time.
- Stochastic gradient descent: take gradient steps using just small subsets of the data.
- ... but this fails badly if applied without thinking.
Logistic regression & stochastic gradient descent

- Parametric
- Fast to train and evaluate
- Easy to incrementally train

\[ x \in \mathbb{R}^n, y \in \{-1, +1\} \]

\[ P(y|x) = \frac{1}{1 + \exp(-yw^T x)} \]

\[ \text{maximize} \quad \prod_m P(y^{(m)}|x^{(m)}) \]

\[ \text{minimize} \quad \sum_{m=1}^{M} \log(1 + \exp(-y^{(m)}w^T x^{(m)})) \]

- \( M \) might be giant, or you might not have access to them all at one time.
- Stochastic gradient descent: take gradient steps using just small subsets of the data.
- ... but this fails badly if applied without thinking.
Logistic regression & stochastic gradient descent

- Parametric
- Fast to train and evaluate
- Easy to incrementally train

\[ x \in \mathbb{R}^n, \ y \in \{-1, +1\} \]

\[ P(y|x) = \frac{1}{1 + \exp(-yw^T x)} \]

\[
\begin{align*}
\text{maximize } & \prod_{m} P(y^{(m)}|x^{(m)}) \\
\text{minimize } & \sum_{m=1}^{M} \log(1 + \exp(-y^{(m)}w^T x^{(m)}))
\end{align*}
\]

- \( M \) might be giant, or you might not have access to them all at one time.
- Stochastic gradient descent: take gradient steps using just small subsets of the data.
- ... but this fails badly if applied without thinking.
Logistic regression & stochastic gradient descent

- Parametric
- Fast to train and evaluate
- Easy to incrementally train

\[ x \in \mathbb{R}^n, y \in \{-1, +1\} \]
\[ P(y|x) = \frac{1}{1 + \exp(-yw^T x)} \]

maximize \( w \prod_{m} P(y^{(m)}|x^{(m)}) \)

minimize \( w \sum_{m=1}^{M} \log(1 + \exp(-y^{(m)}w^T x^{(m)})) \)

- \( M \) might be giant, or you might not have access to them all at one time.
- Stochastic gradient descent: take gradient steps using just small subsets of the data.
- ... but this fails badly if applied without thinking.
Logistic regression & stochastic gradient descent

- Parametric
- Fast to train and evaluate
- Easy to incrementally train

\[ x \in \mathbb{R}^n, y \in \{-1, +1\} \]

\[ P(y|x) = \frac{1}{1 + \exp(-yw^T x)} \]

maximize \[ \prod_m P(y^{(m)}|x^{(m)}) \]

minimize \[ \sum_{m=1}^M \log(1 + \exp(-y^{(m)}w^T x^{(m)})) \]

- \( M \) might be giant, or you might not have access to them all at one time.
- Stochastic gradient descent: take gradient steps using just small subsets of the data.
- ... but this fails badly if applied without thinking.
Linear models

\[ \log \frac{P(Y = 1|x)}{P(Y = -1|x)} \approx w^T x \]
Feature transform

\[ \mathbb{R}^{4000} \rightarrow \{0, 1\}^{400000} \]
Linear model reaches a maximum of 94.0%, fully-supervised boosting 98.7%.
Motivation

Algorithm

Experiments

Prediction stability

- Fully-supervised, looping through ~ 7M training examples.
- Can’t do semi-supervised learning if you forget about objects after not seeing them for a while!
Training buffers

- $D_S$ is the stream of examples seen so far.
- $D_C$ is a new chunk of data.
- Want to maintain $D_B$, a fixed-size buffer of examples which is representative of $D_S$.
- Resample from $D_B$ and $D_C$ proportionally, relative to how much of the total stream they represent.
Training buffers

- \(D_S\) is the stream of examples seen so far.
- \(D_C\) is a new chunk of data.
- Want to maintain \(D_B\), a fixed-size buffer of examples which is representative of \(D_S\).
- Resample from \(D_B\) and \(D_C\) proportionally, relative to how much of the total stream they represent.
Training buffers

- $D_S$ is the stream of examples seen so far.
- $D_C$ is a new chunk of data.
- Want to maintain $D_B$, a fixed-size buffer of examples which is representative of $D_S$.
- Resample from $D_B$ and $D_C$ proportionally, relative to how much of the total stream they represent.
Training buffers

- $D_S$ is the stream of examples seen so far.
- $D_C$ is a new chunk of data.
- Want to maintain $D_B$, a fixed-size buffer of examples which is representative of $D_S$.
- Resample from $D_B$ and $D_C$ proportionally, relative to how much of the total stream they represent.
Training buffers

Long periods where no cars are seen

Cars seen after the first ~6M examples

Stable and improving performance
Need to decide when to induct new tracks as positive examples of objects.
Variable confidences

Test set track results

Correct
Incorrect
Percent

Log odds

Number of test examples

Test set track results

Correct
Incorrect
Percent
Confidence threshold learning

Test set track results

Holdout set frame results

Log odds

Number of test examples

Percent

Correct
Incorrect
Percent

Alex Teichman and Sebastian Thrun
Online, semi-supervised learning // RSS2012
Algorithm sketch

- Hand-labeled training data
- x100
- Update classifier
- Auto-labeled background training data

Alex Teichman and Sebastian Thrun
Online, semi-supervised learning // RSS2012
Algorithm sketch

Hand-labeled training data → Update classifier → Update thresholds → Hand-labeled holdout data

x100

Auto-labeled background training data → Auto-labeled background holdout data

Alex Teichman and Sebastian Thrun

Online, semi-supervised learning // RSS2012
Algorithm sketch

Motivation

Algorithm

Experiments

Unlabeled data from the stream

Induct examples

Hand-labeled training data

Update classifier

Update thresholds

Auto-labeled background training data

Auto-labeled background holdout data

Hand-labeled holdout data

x100

x100

Hand-labeled

holdout
data

Unlabeled data

from the stream

Induct

examples

Update

classifier

Update

thresholds

Auto-labeled

background

training data

Auto-labeled

background

holdout data

x100 x100

Alex Teichman and Sebastian Thrun

Online, semi-supervised learning // RSS2012
Algorithm sketch

Unlabeled data from the stream

Training buffers (one per class) -> Half
Induct examples

Update classifier

Auto-labeled background training data

Hand-labeled training data

Update thresholds

Auto-labeled background holdout data

Holdout buffers (one per class) -> Half

Hand-labeled holdout data

X100

X100

Alex Teichman and Sebastian Thrun

Online, semi-supervised learning // RSS2012
Online tracking-based semi-supervised learning

- Additional hand-labeled examples can break it out of local minima.
- ~8M unique unlabeled examples.
Given lots of negative examples, recall initially drops, then recovers; overall accuracy improves.
Online tracking-based semi-supervised learning

Results after running for \(\sim 1\) week. Total hand-labeled tracks: 108, vs \(\sim 4000\) needed for good performance in fully-supervised case.

Max accuracy when training on automatically-labeled background and all hand-labeled tracks: 90.1%.
Annotating

Track Classification
- car: 9.00033
- pedestrian: -12.0312
- bicyclist: -10.858

Track 0119 / 140
Frame 0056 / 164
01279 points.

training_fms/lomita_and_santa_teresa01-11-17-2009_18-35-12.fm

Track Classification
- car: -6.72168
- pedestrian: -9.74034
- bicyclist: -12.3685

Track 0214 / 233
Frame 0014 / 46
00207 points.

training_fms/lasuen02-11-17-2009_17-07-43.fm
The holdout set can tell you where to look for incorrect examples.
Motivation
Algorithm
Experiments

Annotating

Alex Teichman and Sebastian Thrun

Online, semi-supervised learning // RSS2012
Adding classes later

Max accuracy when training on automatically-labeled background and all hand-labeled tracks: 90.8%.
Adding classes later
Causes of failure while developing this

- Memory fragmentation
- Combined training buffer rather than one per class
- Stochastic gradient constant step size
- Not weighting the hand-labeled data
Causes of failure while developing this

- Memory fragmentation
- Combined training buffer rather than one per class
  - Stochastic gradient constant step size
  - Not weighting the hand-labeled data
Causes of failure while developing this

- Memory fragmentation
- Combined training buffer rather than one per class
- Stochastic gradient constant step size
- Not weighting the hand-labeled data
Causes of failure while developing this

- Memory fragmentation
- Combined training buffer rather than one per class
- Stochastic gradient constant step size
- Not weighting the hand-labeled data
Future work: dual induction
Future work

**Segmentation**
- Connected components
- Background subtraction

**Tracking**
- Kalman filters

**Classification**
- Boosting
- Logistic regression, stochastic gradient descent

- Discriminative segmentation and tracking