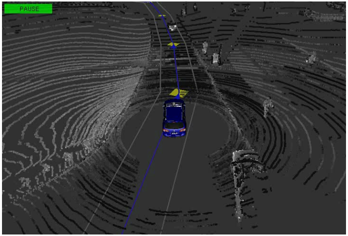
Online, semi-supervised learning for long-term interaction with object recognition systems

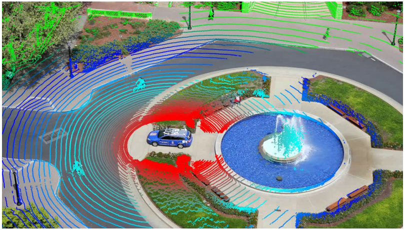
Alex Teichman and Sebastian Thrun

Department of Computer Science
Stanford University

July 12, 2012

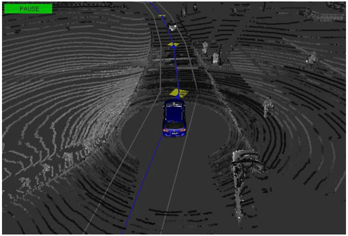
The big picture

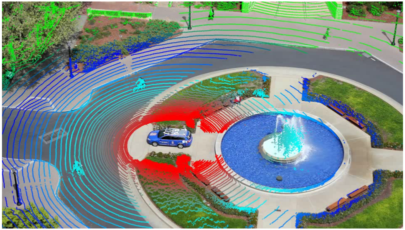




- What is the desired user interface for object recognition?
- Want autonomy with the option for user input.
- Online, active, semi-supervised learning
  ...

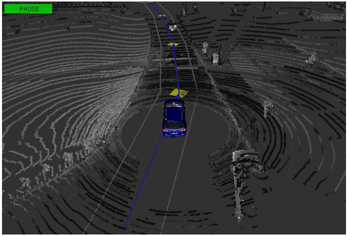
The big picture

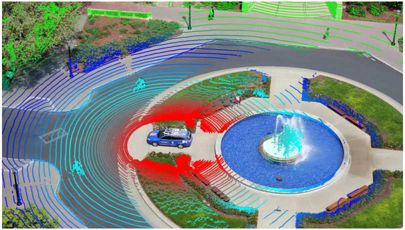




- What is the desired user interface for object recognition?
- Want autonomy with the option for user input.
- Online, active, semi-supervised learning
  ...

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
  ...

Static train/test framework

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

Number of tracks					
Set	Car	Pedestrian	Bicyclist	Background	All
Training	904	205	187	6585	7881
Testing	847	112	140	4936	6035
Total	1751	317	327	11,521	13,916
Number of frames					
Set	Car	Pedestrian	Bicyclist	Background	All
Training	92,255	32,281	31,165	532,760	688,461
Testing	59,173	22,203	25,410	530,917	637,703
Total	151,428	54,484	56,575	1,063,677	1,326,164

- Rigorous evaluation and comparison
- Experimental setup

- Occasional user interaction
- Infinite unlabeled data stream

We don't want to overfit to this framework!

Static train/test framework

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

Number of tracks					
Set	Car	Pedestrian	Bicyclist	Background	All
Training	904	205	187	6585	7881
Testing	847	112	140	4936	6035
Total	1751	317	327	11,521	13,916

Number of frames					
Set	Car	Pedestrian	Bicyclist	Background	All
Training	92,255	32,281	31,165	532,760	688,461
Testing	59,173	22,203	25,410	530,917	637,703
Total	151,428	54,484	56,575	1,063,677	1,326,164

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

We don't want to overfit to this framework!

Static train/test framework

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

Number of tracks					
Set	Car	Pedestrian	Bicyclist	Background	All
Training	904	205	187	6585	7881
Testing	847	112	140	4936	6035
Total	1751	317	327	11,521	13,916
Number of frames					
Set	Car	Pedestrian	Bicyclist	Background	All
Training	92,255	32,281	31,165	532,760	688,461
Testing	59,173	22,203	25,410	530,917	637,703
Total	151,428	54,484	56,575	1,063,677	1,326,164

- Rigorous evaluation and comparison
- Experimental setup

- Occasional user interaction
- Infinite unlabeled data stream

We don't want to overfit to this framework!

Static train/test framework

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

Number of tracks					
Set	Car	Pedestrian	Bicyclist	Background	All
Training	904	205	187	6585	7881
Testing	847	112	140	4936	6035
Total	1751	317	327	11,521	13,916
Number of frames					
Set	Car	Pedestrian	Bicyclist	Background	All
Training	92,255	32,281	31,165	532,760	688,461
Testing	59,173	22,203	25,410	530,917	637,703
Total	151,428	54,484	56,575	1,063,677	1,326,164

- Rigorous evaluation and comparison
- Experimental setup

- Occasional user interaction
- Infinite unlabeled data stream

We don't want to overfit to this framework!

Static train/test framework

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

Number of tracks					
Set	Car	Pedestrian	Bicyclist	Background	All
Training	904	205	187	6585	7881
Testing	847	112	140	4936	6035
Total	1751	317	327	11,521	13,916
Number of frames					
Set	Car	Pedestrian	Bicyclist	Background	All
Training	92,255	32,281	31,165	532,760	688,461
Testing	59,173	22,203	25,410	530,917	637,703
Total	151,428	54,484	56,575	1,063,677	1,326,164

- Rigorous evaluation and comparison
- Experimental setup

- Occasional user interaction
- Infinite unlabeled data stream

We don't want to overfit to this framework!

Static train/test framework

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

Number of tracks					
Set	Car	Pedestrian	Bicyclist	Background	All
Training	904	205	187	6585	7881
Testing	847	112	140	4936	6035
Total	1751	317	327	11,521	13,916
Number of frames					
Set	Car	Pedestrian	Bicyclist	Background	All
Training	92,255	32,281	31,165	532,760	688,461
Testing	59,173	22,203	25,410	530,917	637,703
Total	151,428	54,484	56,575	1,063,677	1,326,164

- Rigorous evaluation and comparison
- Experimental setup

- Occasional user interaction
- Infinite unlabeled data stream

We don't want to overfit to this framework!

Static train/test framework

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

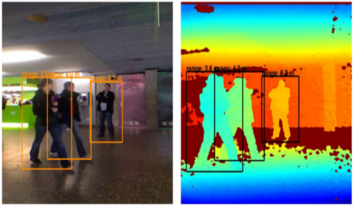
Number of tracks					
Set	Car	Pedestrian	Bicyclist	Background	All
Training	904	205	187	6585	7881
Testing	847	112	140	4936	6035
Total	1751	317	327	11,521	13,916
Number of frames					
Set	Car	Pedestrian	Bicyclist	Background	All
Training	92,255	32,281	31,165	532,760	688,461
Testing	59,173	22,203	25,410	530,917	637,703
Total	151,428	54,484	56,575	1,063,677	1,326,164

- 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

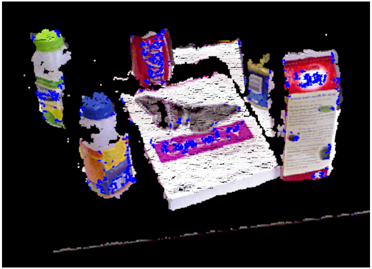
Spinello, Stachniss, and Burgard

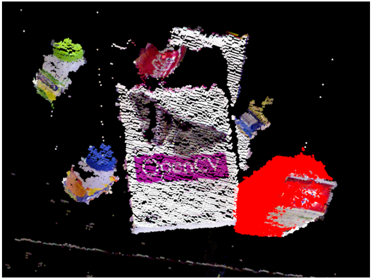
Object recognition approaches - semantic segmentation

Douillard et al.

Combining sliding windows and semantic segmentation: Lai et al.

Object recognition approaches - keypoint matching



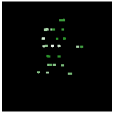


Solutions in Perception Challenge

Collet et al.

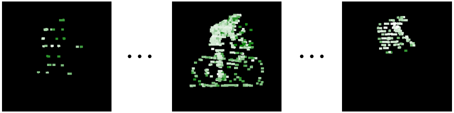
Problem decomposition

Segmentation — Tracking — Track classification



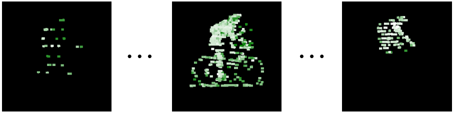
Problem decomposition

Segmentation — **Tracking** — Track classification



Problem decomposition

Segmentation — Tracking — Track classification



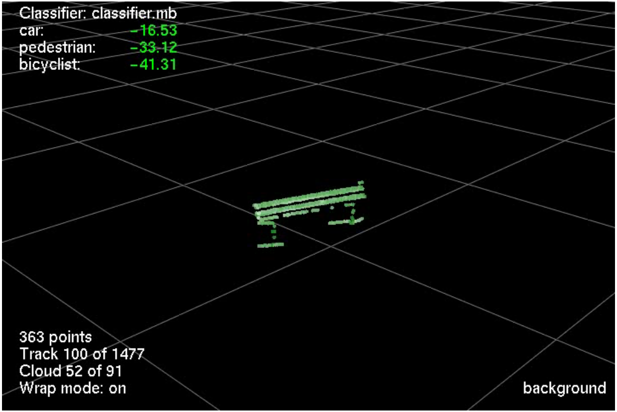
$$\begin{bmatrix} \text{car} \\ \text{ped} \\ \text{bike} \end{bmatrix} \quad \begin{bmatrix} -11.60 \\ 0.36 \\ 1.27 \end{bmatrix} \quad \cdots \quad \begin{bmatrix} -20.31 \\ -6.42 \\ 4.01 \end{bmatrix} \quad \cdots \quad \begin{bmatrix} -5.28 \\ -3.67 \\ -0.28 \end{bmatrix} \longrightarrow \begin{bmatrix} -30.98 \\ -14.68 \\ 11.42 \end{bmatrix}$$

Descriptors

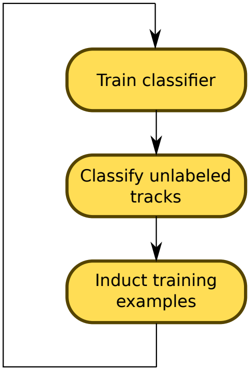
- Oriented bounding box size
- Spin images
- HOG descriptors computed on virtual orthographic camera images

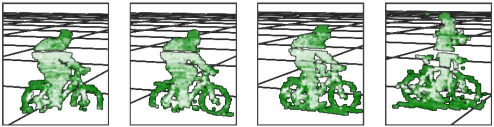
- 29 different descriptor spaces
- $x \in \mathbb{R}^{\sim 4000}$

Tracks


Classifier: classifier.mb
car: -16.53
pedestrian: -33.12
bicyclist: -41.31
363 points
Track 100 of 1477
Cloud 52 of 91
Wrap mode: on
background

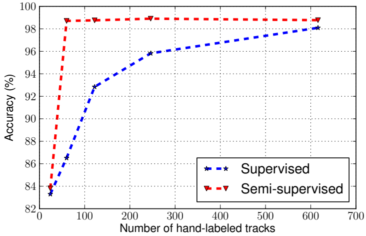
Tracking-based semi-supervised learning

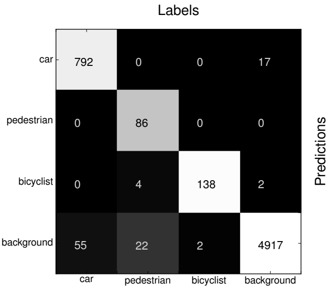




- Large, automatically-labeled background dataset is provided. Often this is easy to collect.
- Only positive examples are inducted during semi-supervised learning.

Tracking-based semi-supervised learning



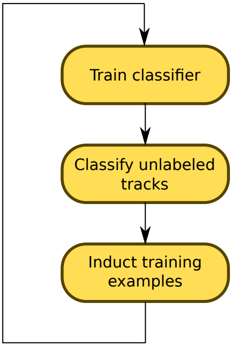


- Unsupervised method given millions of additional unlabeled examples.
- Track classification accuracy is reported. (This does not include segmentation and tracking errors.)

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

τ is a confidence threshold chosen by hand
$\mathbb{S}$ is a small set of seed tracks, labeled by hand
$\mathbb{U}$ is a large set of unlabeled tracks
$\mathbb{B}$ is a large set of background tracks
$\mathbb{W}$ is a working set, initially empty

$\mathbb{W} := \mathbb{S} \cup \mathbb{B}$
repeat
    Train frame classifier $\mathbb{C}$ on $\mathbb{W}$
    $\mathbb{W} := \mathbb{S} \cup \mathbb{B}$
    for $u \in \mathbb{U}$ **do**
        Classify track u using $\mathbb{C}$
        $c :=$ confidence(u)
        $l :=$ classification(u)
        if $c \geq \tau$ and $l \neq$ "background" **then**
            Add u to $\mathbb{W}$ with label l
        end if
    end for
until converged

Modularity

Segmentation

- Connected components
- Background subtraction

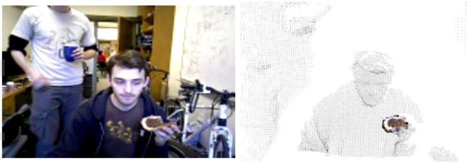
Tracking

- Kalman filters

Classification

- Boosting
- Logistic regression,
  stochastic gradient descent

- Discriminative segmentation
  and tracking



WAFR2012

Logistic regression & stochastic gradient descent

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

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

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

$$\underset{w}{\text{maximize}} \quad \prod_m \mathbb{P}(y^{(m)}|x^{(m)})$$

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

- 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

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

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

$$\underset{w}{\text{maximize}} \quad \prod_m \mathbb{P}(y^{(m)}|x^{(m)})$$

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

- 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

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

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

$$\underset{w}{\text{maximize}} \quad \prod_m \mathbb{P}(y^{(m)}|x^{(m)})$$

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

- 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

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

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

$$\underset{w}{\text{maximize}} \quad \prod_m \mathbb{P}(y^{(m)}|x^{(m)})$$

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

- 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

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

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

$$\underset{w}{\text{maximize}} \quad \prod_m \mathbb{P}(y^{(m)}|x^{(m)})$$

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

- 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

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

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

$$\underset{w}{\text{maximize}} \quad \prod_m \mathbb{P}(y^{(m)}|x^{(m)})$$

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

- 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

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

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

$$\underset{w}{\text{maximize}} \quad \prod_m \mathbb{P}(y^{(m)}|x^{(m)})$$

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

- 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

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

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

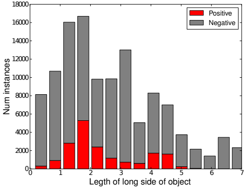
$$\underset{w}{\text{maximize}} \quad \prod_m \mathbb{P}(y^{(m)}|x^{(m)})$$

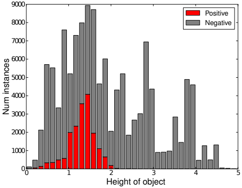
$$\underset{w}{\text{minimize}} \quad \sum_{m=1}^{M} \log(1 + \exp(-y^{(m)} w^T x^{(m)}))$$

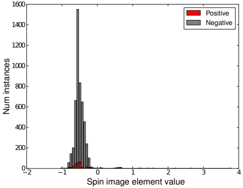
- M might be giant, or you might not have access to them all at one time.
- Stochastic gradient descent: take gradient steps using just small subsets of the data.
- ... but this fails badly if applied without thinking.

Linear models

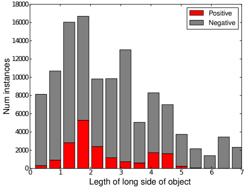
$$\log \frac{\mathbb{P}(Y=1|x)}{\mathbb{P}(Y=-1|x)} \approx w^T x$$

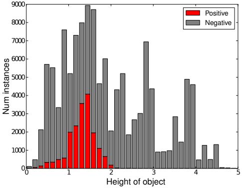


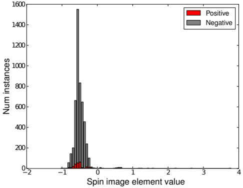


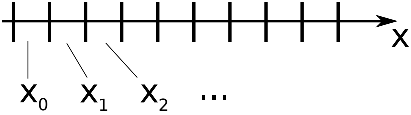


Feature transform



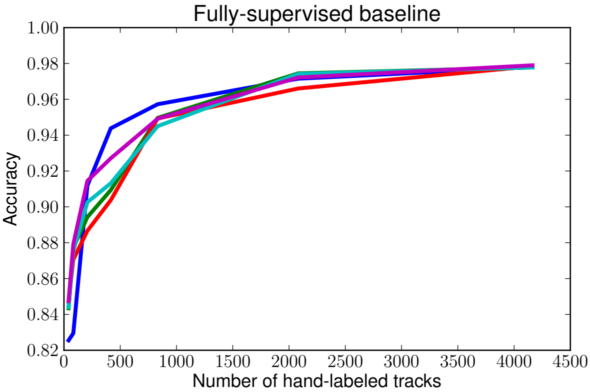






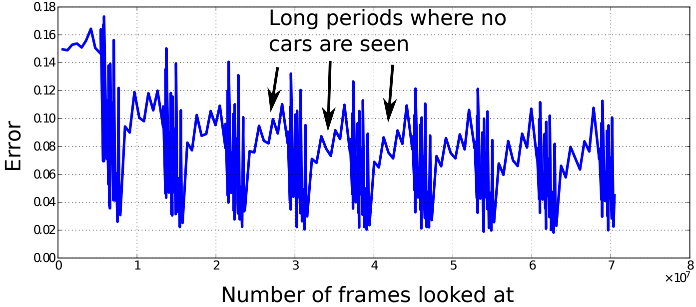
$$\mathbb{R}^{4000} \rightarrow \{0,1\}^{400000}$$

Supervised performance



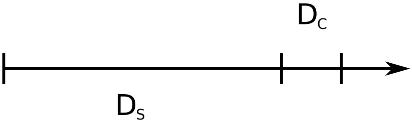
Linear model reaches a maximum of 94.0%, fully-supervised boosting 98.7%.

Prediction stability



- Fully-supervised, looping through $\sim$ 7M training examples.
- Can't do semi-supervised learning if you forget about objects after not seeing them for a while!

Training buffers

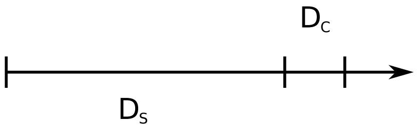


- D_S is the stream of examples seen so far.
- D_C is a new chunk of data.
- Want to maintain D_B, a fixed-size buffer of examples which is representative of D_S.
- Resample from D_B and D_C proportionally, relative to how much of the total stream they represent.

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

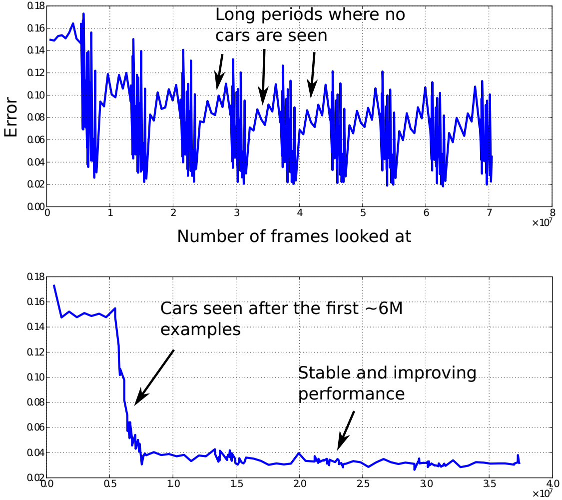


- D_S is the stream of examples seen so far.
- D_C is a new chunk of data.
- Want to maintain D_B, a fixed-size buffer of examples which is representative of D_S.
- Resample from D_B and D_C proportionally, relative to how much of the total stream they represent.

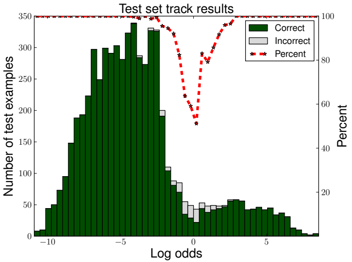
Training buffers

- 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

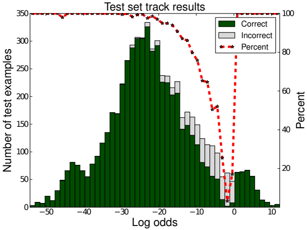


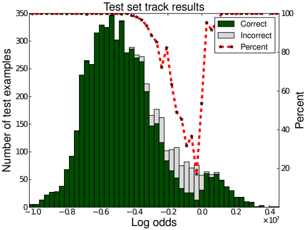
Confidence thresholds



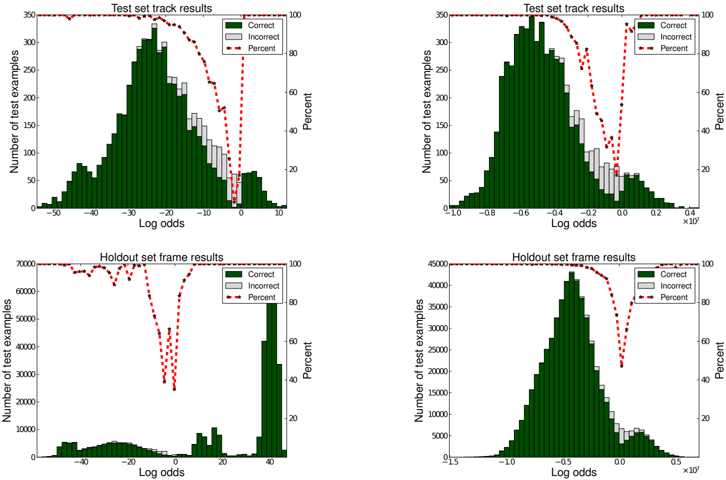
- Need to decide when to induct new tracks as positive examples of objects.

Variable confidences

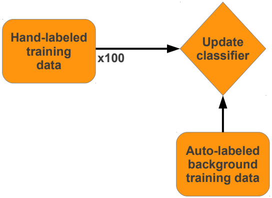




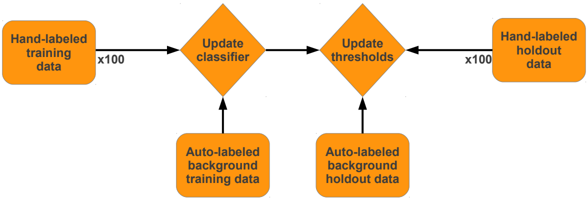
Confidence threshold learning



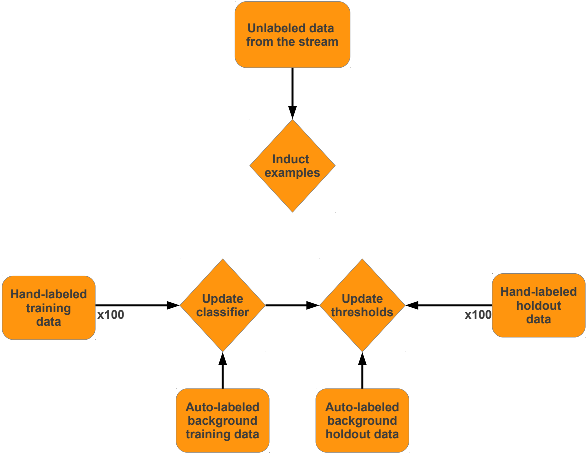
Algorithm sketch



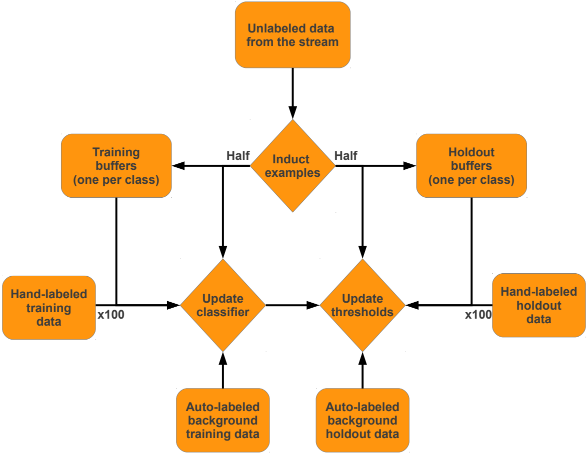
Algorithm sketch



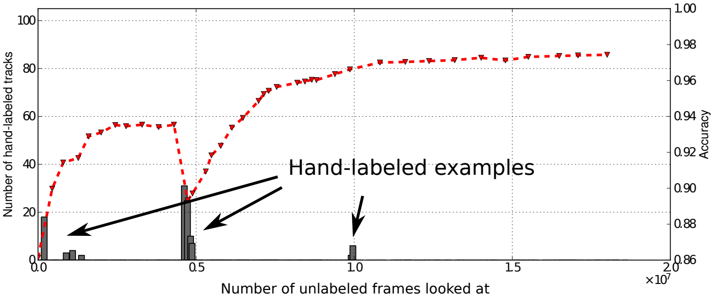
Algorithm sketch



Algorithm sketch

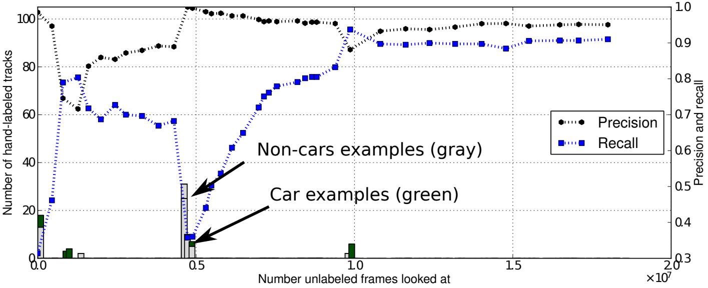


Online tracking-based semi-supervised learning



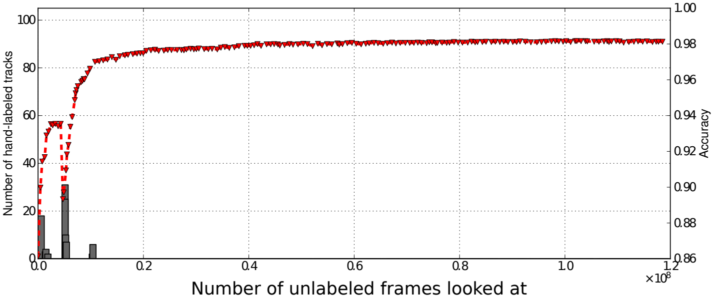
- Additional hand-labeled examples can break it out of local minima.
- ∼8M unique unlabeled examples.

Online tracking-based semi-supervised learning



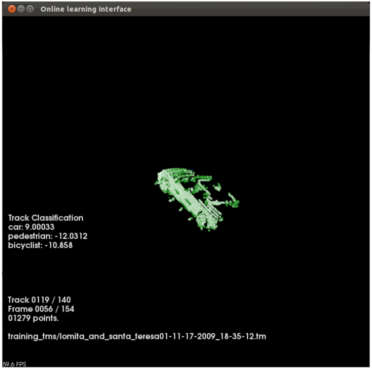
- Given lots of negative examples, recall initially drops, then recovers; overall accuracy improves.

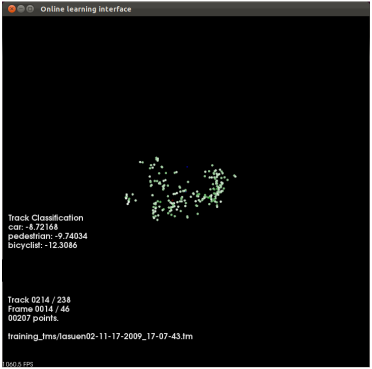
Online tracking-based semi-supervised learning



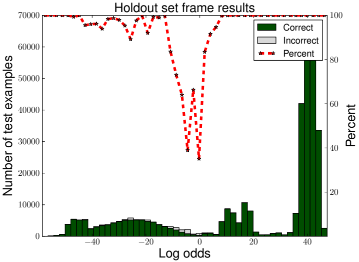
- Results after running for ∼1 week. Total hand-labeled tracks: 108, vs ∼4000 needed for good performance in fully-supervised case.
- Max accuracy when training on automatically-labeled background and all hand-labeled tracks: 90.1%.

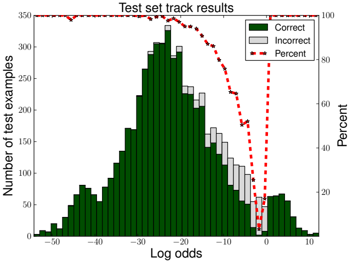
Annotating


Online learning interface
Track Classification
car: 9.00033
pedestrian: -12.0312
bicyclist: -10.858
Track 0119 / 140
Frame 0056 / 154
01279 points.
training_tms/lomita_and_santa_teresa01-11-17-2009_18-35-12.tm
59.6 FPS


Online learning interface
Track Classification
car: -8.72168
pedestrian: -9.74034
bicyclist: -12.3086
Track 0214 / 238
Frame 0014 / 46
00207 points.
training_tms/lasuen02-11-17-2009_17-07-43.tm
1060.5 FPS

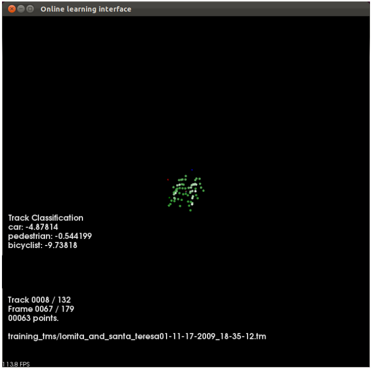
Annotating

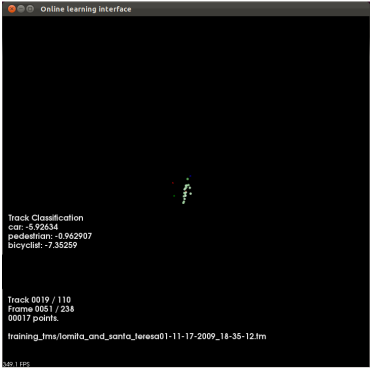




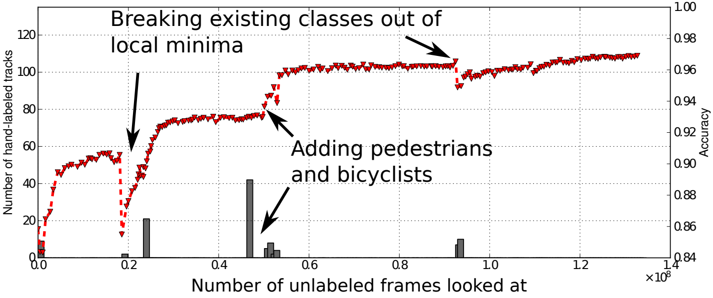
- The holdout set can tell you where to look for incorrect examples.

Annotating


Online learning interface
Track Classification
car: -4.87814
pedestrian: -0.544199
bicyclist: -9.73818
Track 0008 / 132
Frame 0067 / 179
00063 points.
training_tms/lomita_and_santa_teresa01-11-17-2009_18-35-12.tm
113.8 FPS

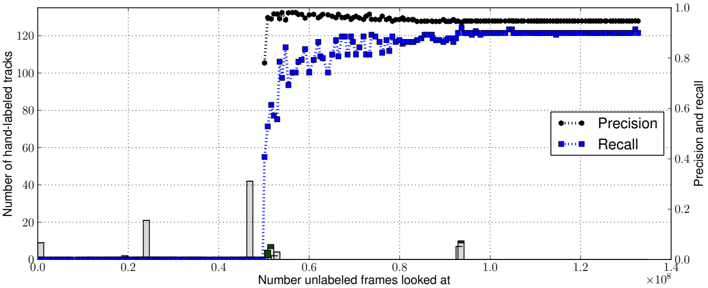

Online learning interface
Track Classification
car: -5.92634
pedestrian: -0.962907
bicyclist: -7.35259
Track 0019 / 110
Frame 0051 / 238
00017 points.
training_tms/lomita_and_santa_teresa01-11-17-2009_18-35-12.tm
349.1 FPS

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

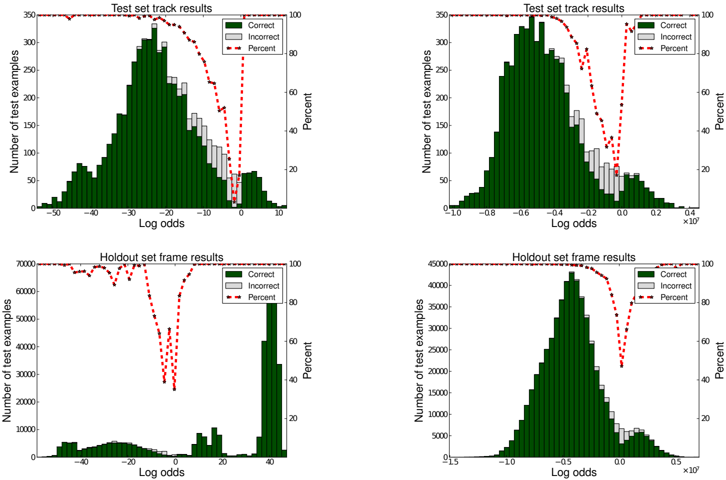
Causes of failure while developing this

- 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