Creating Better Correlation Filters

David S. Bolme

Colorado State University

CS510 - March 29, 2010

Motivation

Why do we care?

- Correlation is typical not even tried for difficult problems.
- ASEF and MOSSE provide a way to get "better" output from a correlation filter.
- Correlation is commonly used. (Taught in most beginning image processing classes.)
- Implementation is simple.
- ASEF and MOSSE are often more accurate. and much faster than state of the art detection and tracking algorithms.

Overview

- Correlation Review
- Constructing Better Filters:
  - Person Detection
- Filter Based Tracking
- Homework Tips

Correlation and Convolution

- Cross Correlation - Aligned Signal

$$(f \star h)[x] = \sum_{dx=-\infty}^{\infty} f[dx]h[x+dx]$$

- Convolution - Reversed Signal

$$(f * h)[x] = \sum_{dx=-\infty}^{\infty} f[dx]h[x-dx]$$

Convolution Theorem

- Convolution is simpler in the Fourier Domain.

$$f * h = \mathcal{F}^{-1}(\mathcal{F}(f) \odot \mathcal{F}(h)) = \mathcal{F}^{-1}(F \odot H)$$

- Convolution.

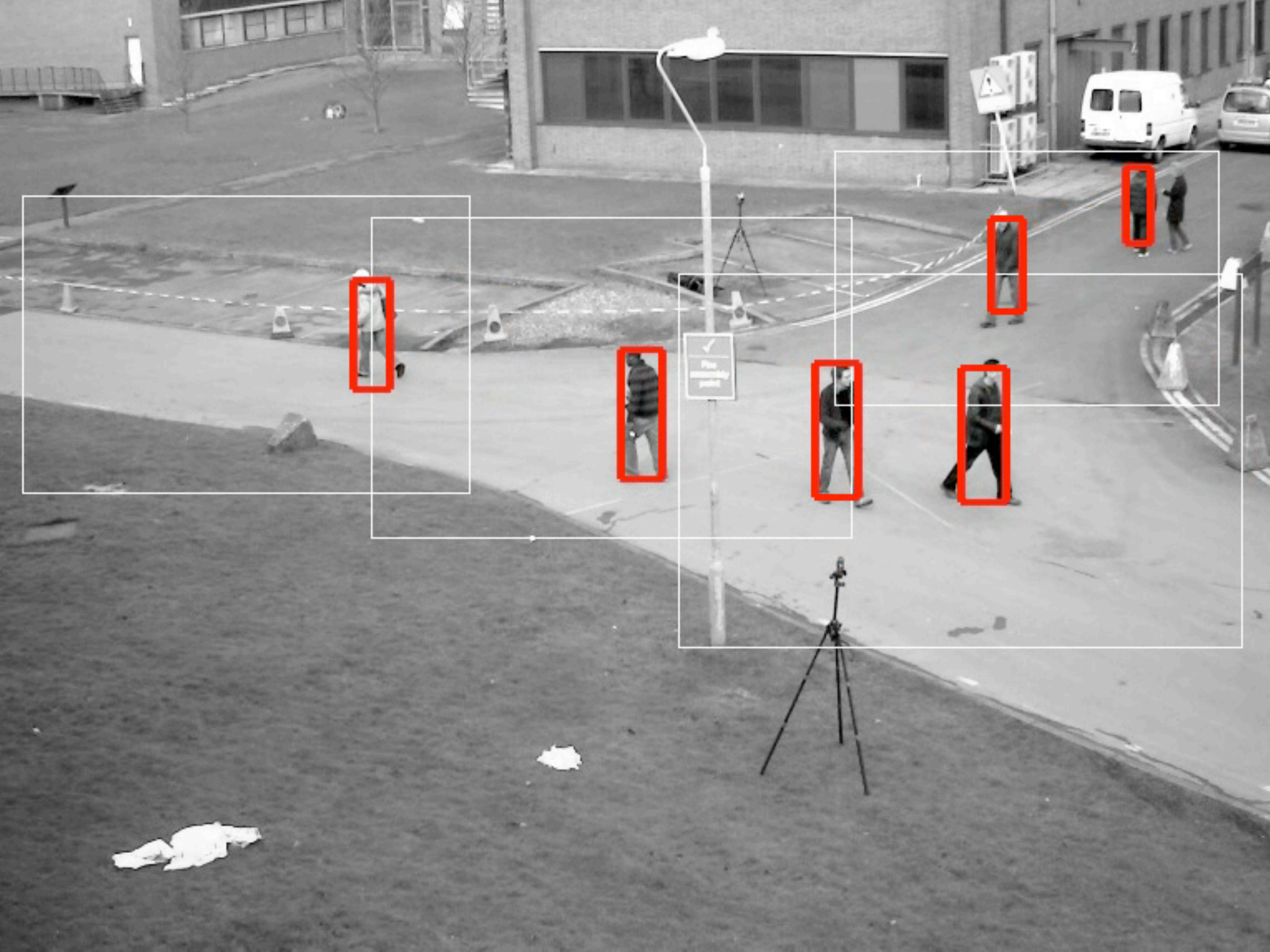
$$G = F \odot H$$

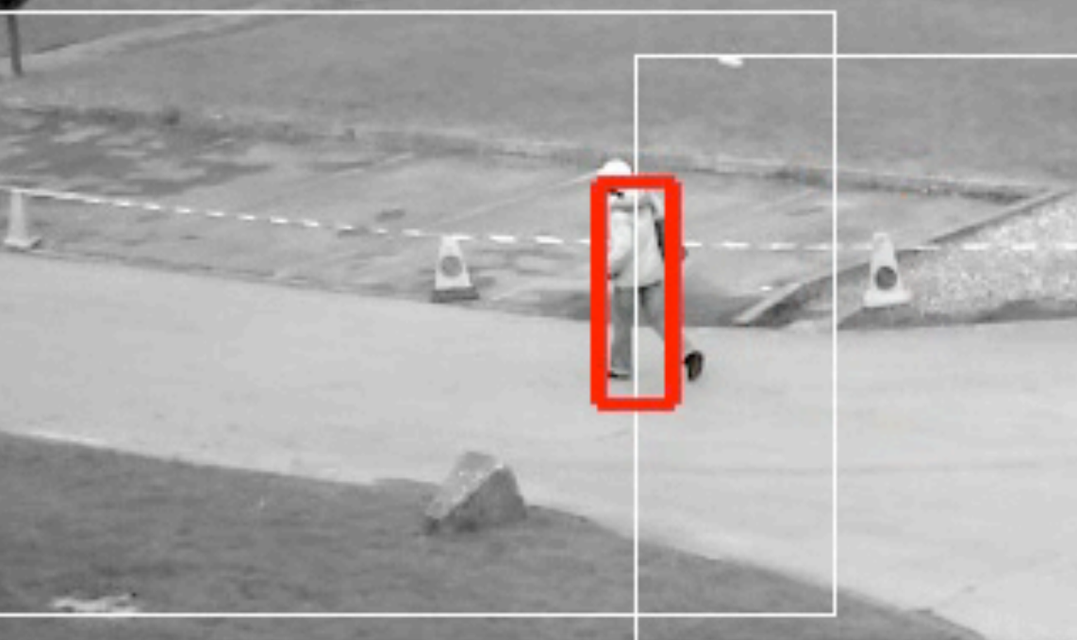
- Correlation.

$$G = F \odot H^*$$

- What does this do to an image?

$$\mathcal{F}^{-1}(\mathcal{F}(f)^*)$$



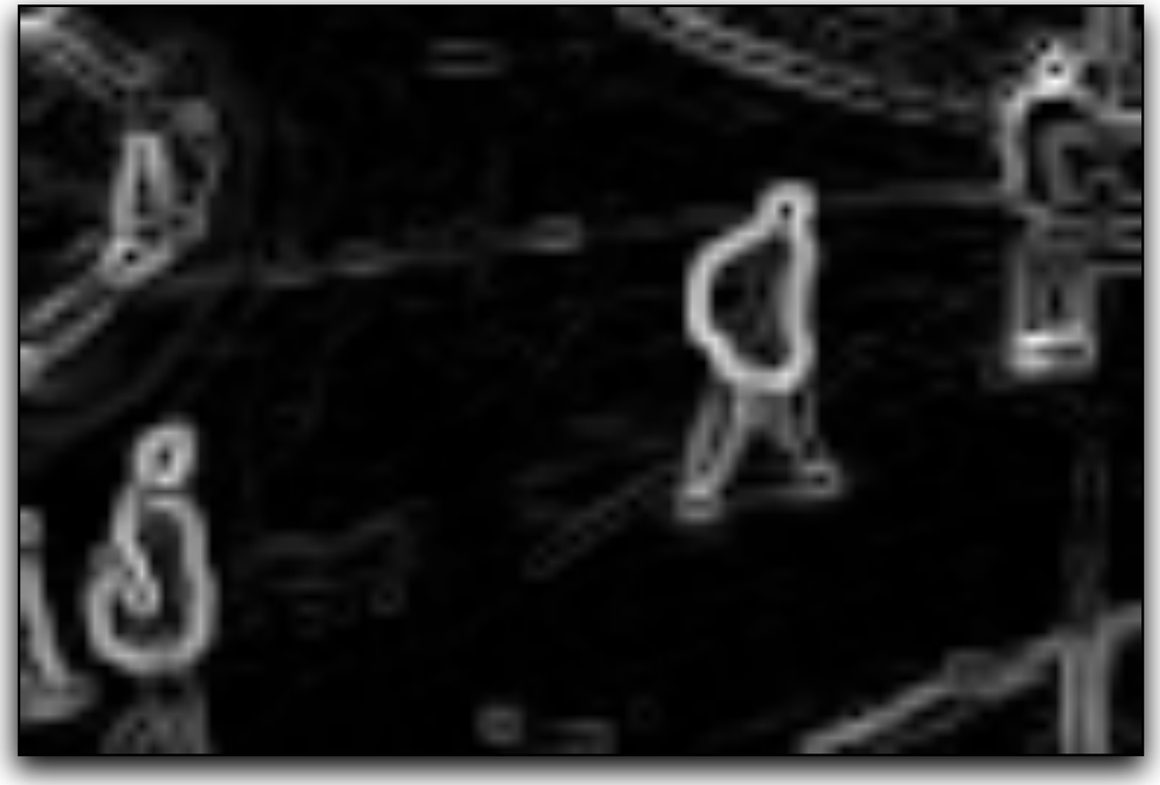


Preprocessing

The variety of clothing colors
causes problems for detection.

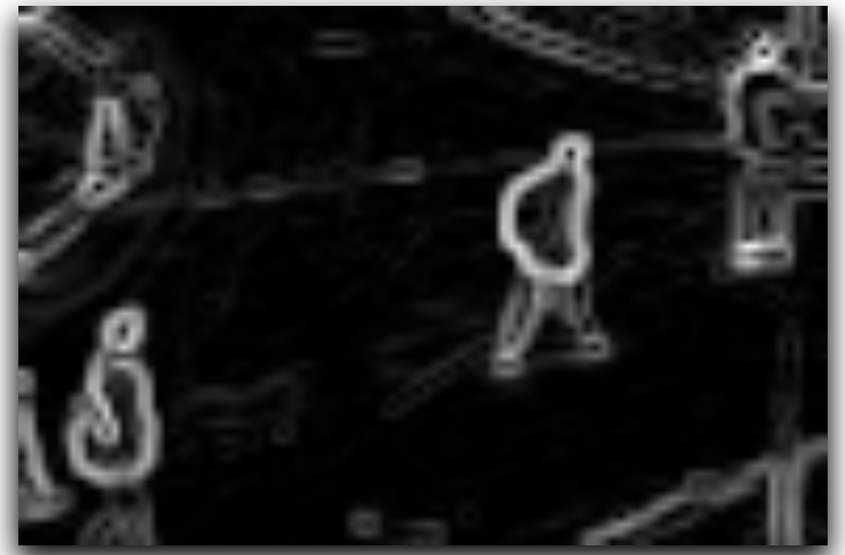
Original Frame

Gradient Magnitude



Correlation

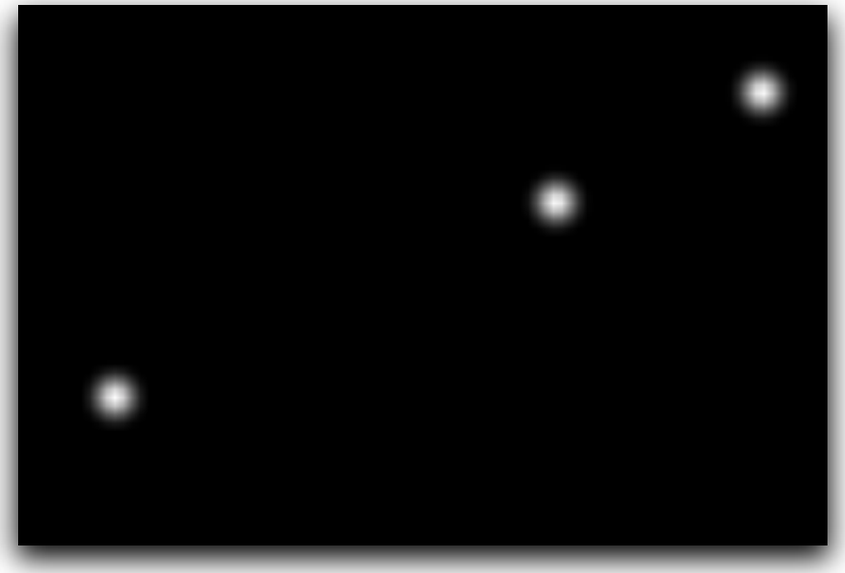
Input - f



Filter - h

?

Output - g



$$f \otimes h = g$$

$$F \odot H^* = G$$

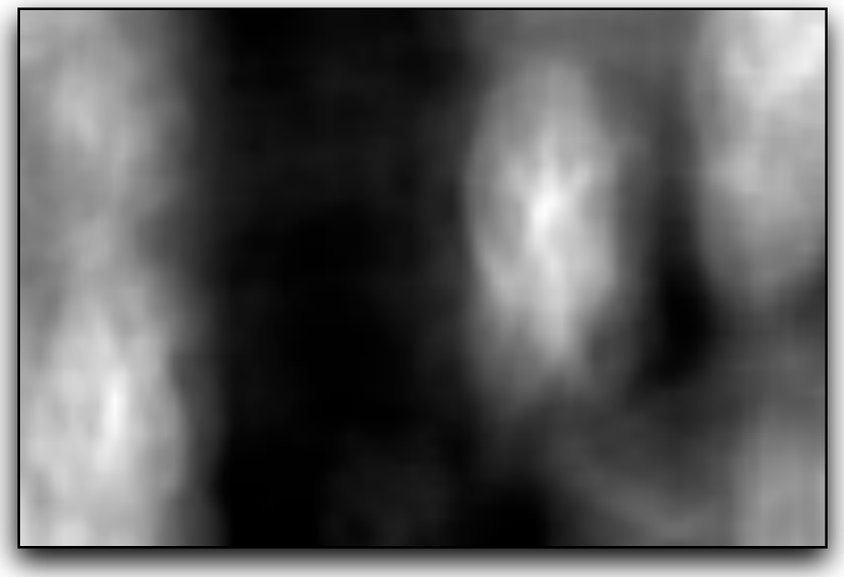
Problem: Learn a filter that maps the input to the output.

Training A Better Filter

Input

Filter

Output



Simple templates just don't work.

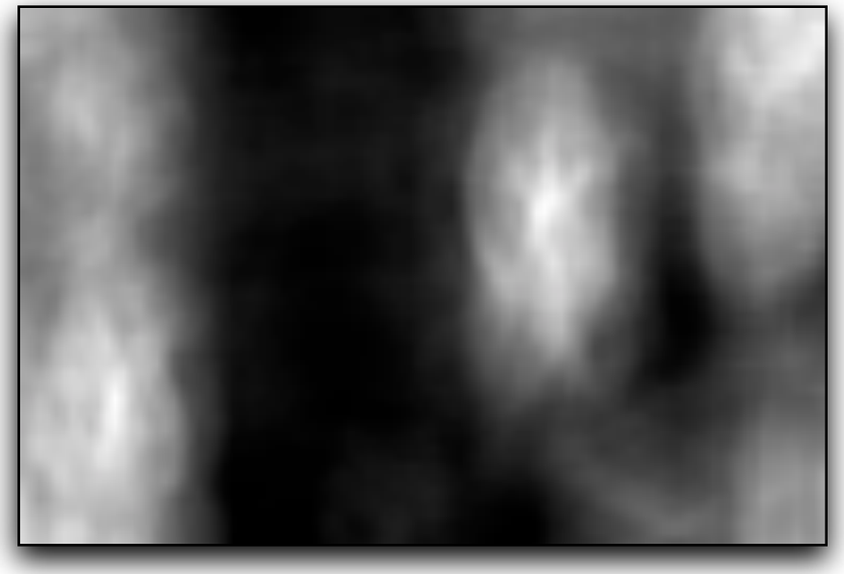
The Solution....
Average of Synthetic Exact Filters (CVPR 2009)

Exact Filters

Training Image

?

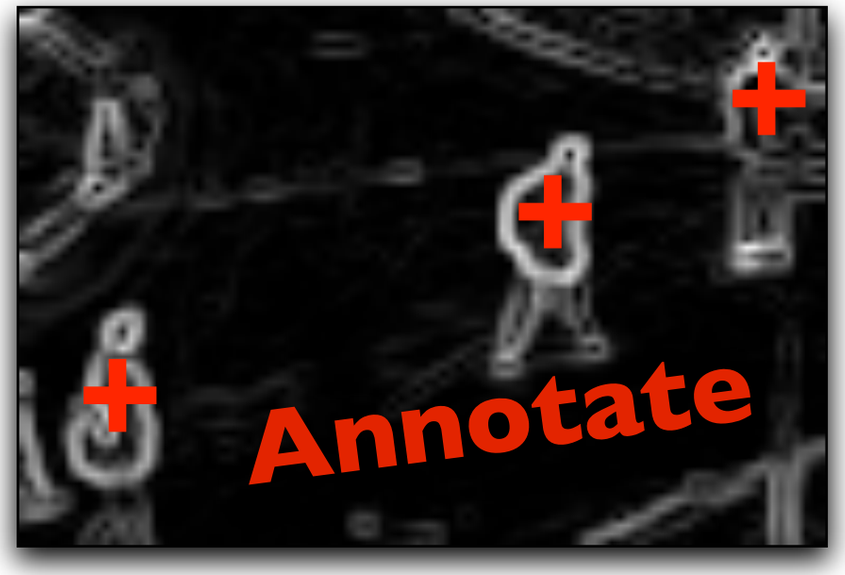
Output



$$F_i \odot H_i^* = G_i$$

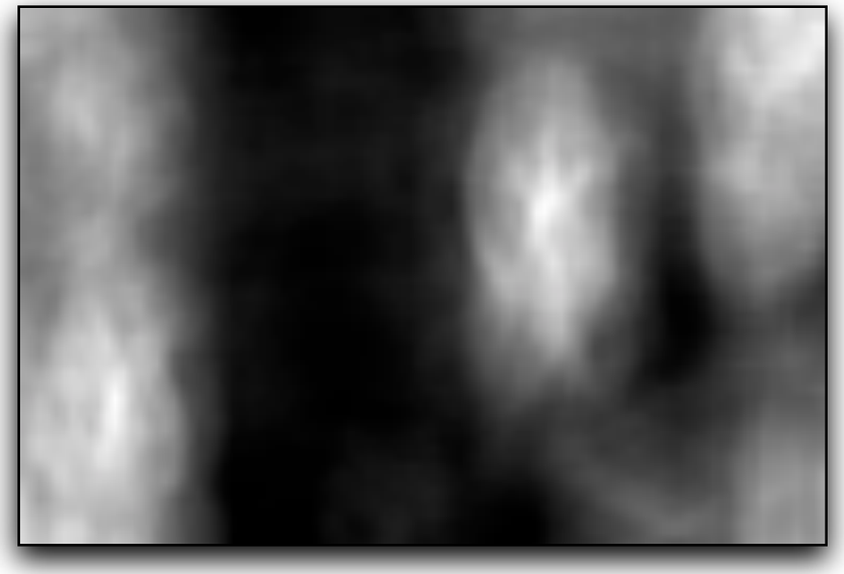
Exact Filters

Training Image



?

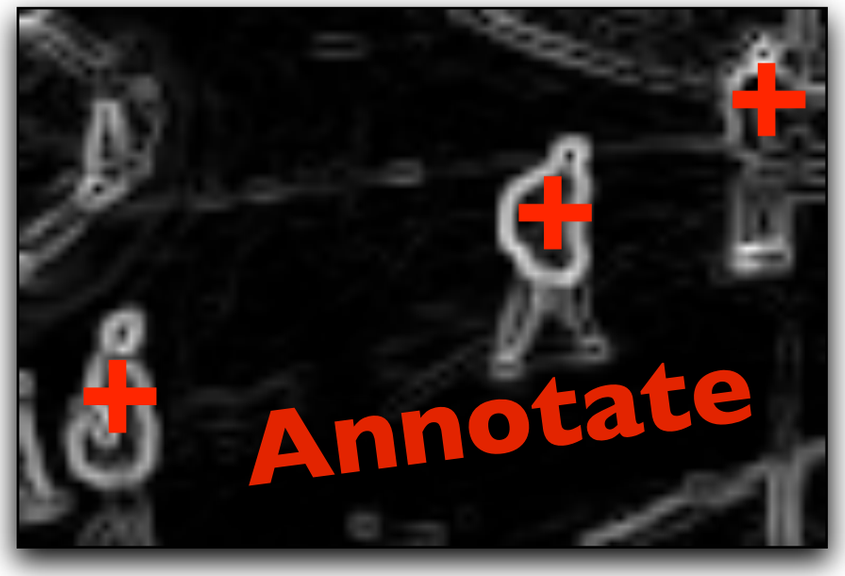
Output



$$F_i \odot H_i^* = G_i$$

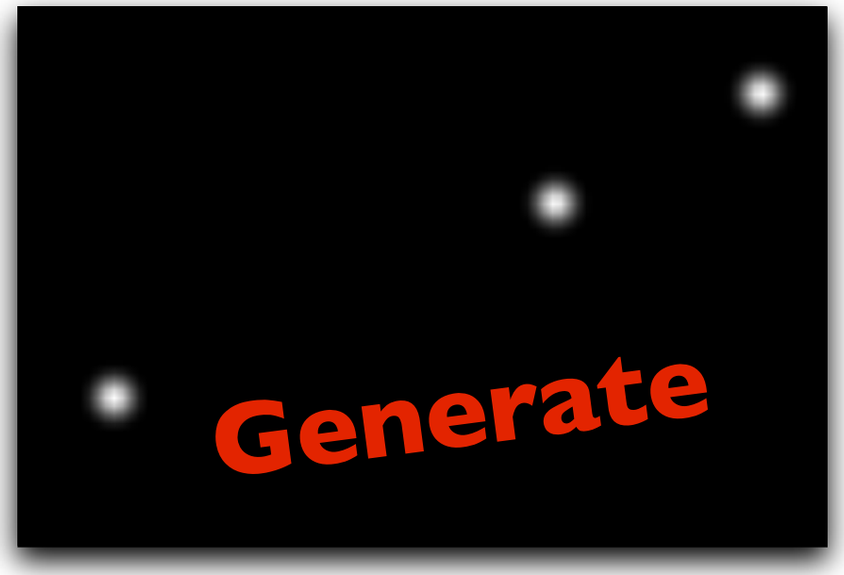
Exact Filters

Training Image



?

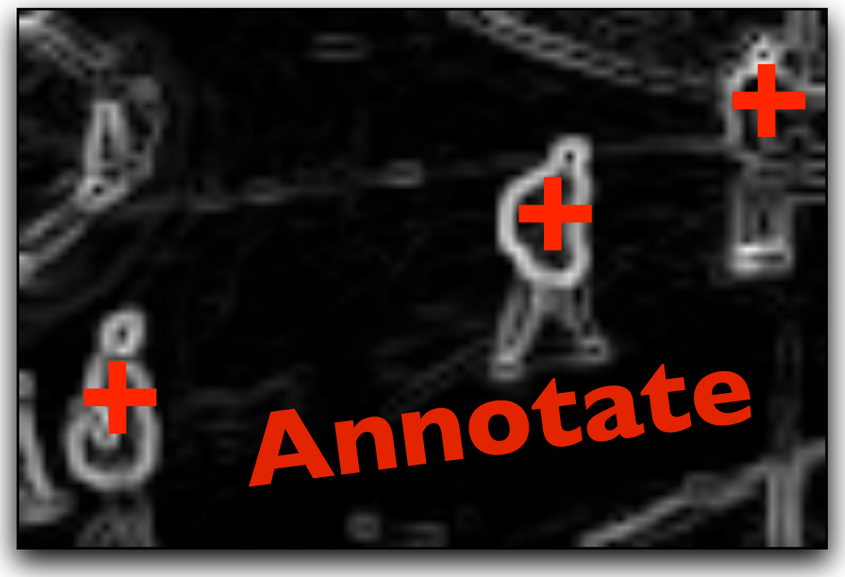
Output



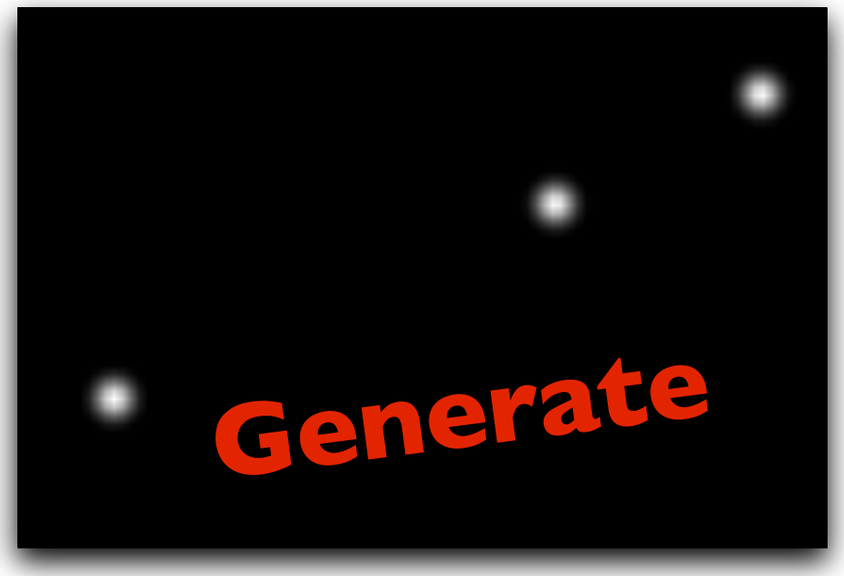
$$F_i \odot H_i^* = G_i$$

Exact Filters

Training Image



Output



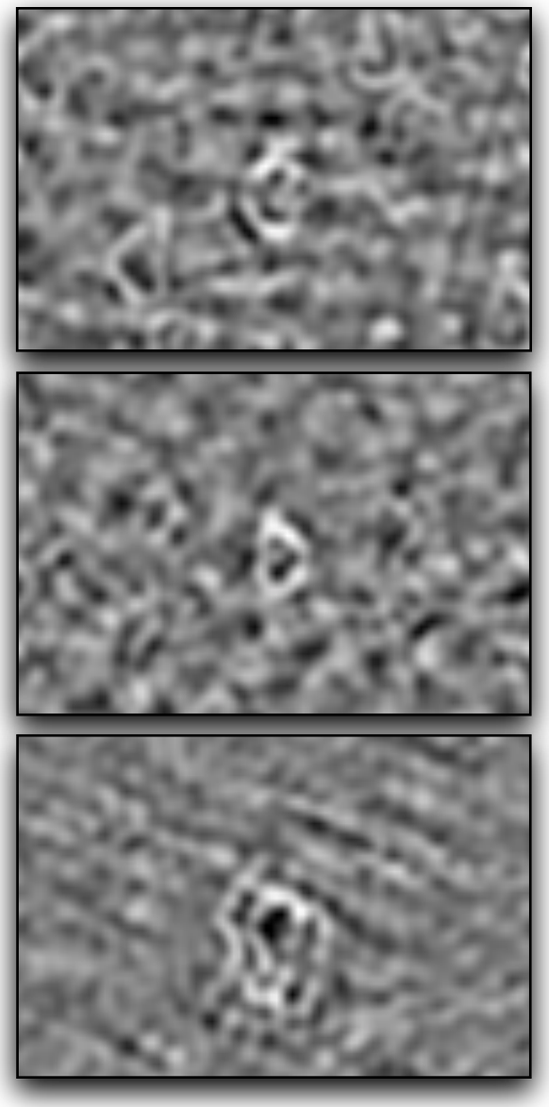
$$F_i \odot H_i^* = G_i$$

$$H_i^* = \frac{G_i}{F_i}$$

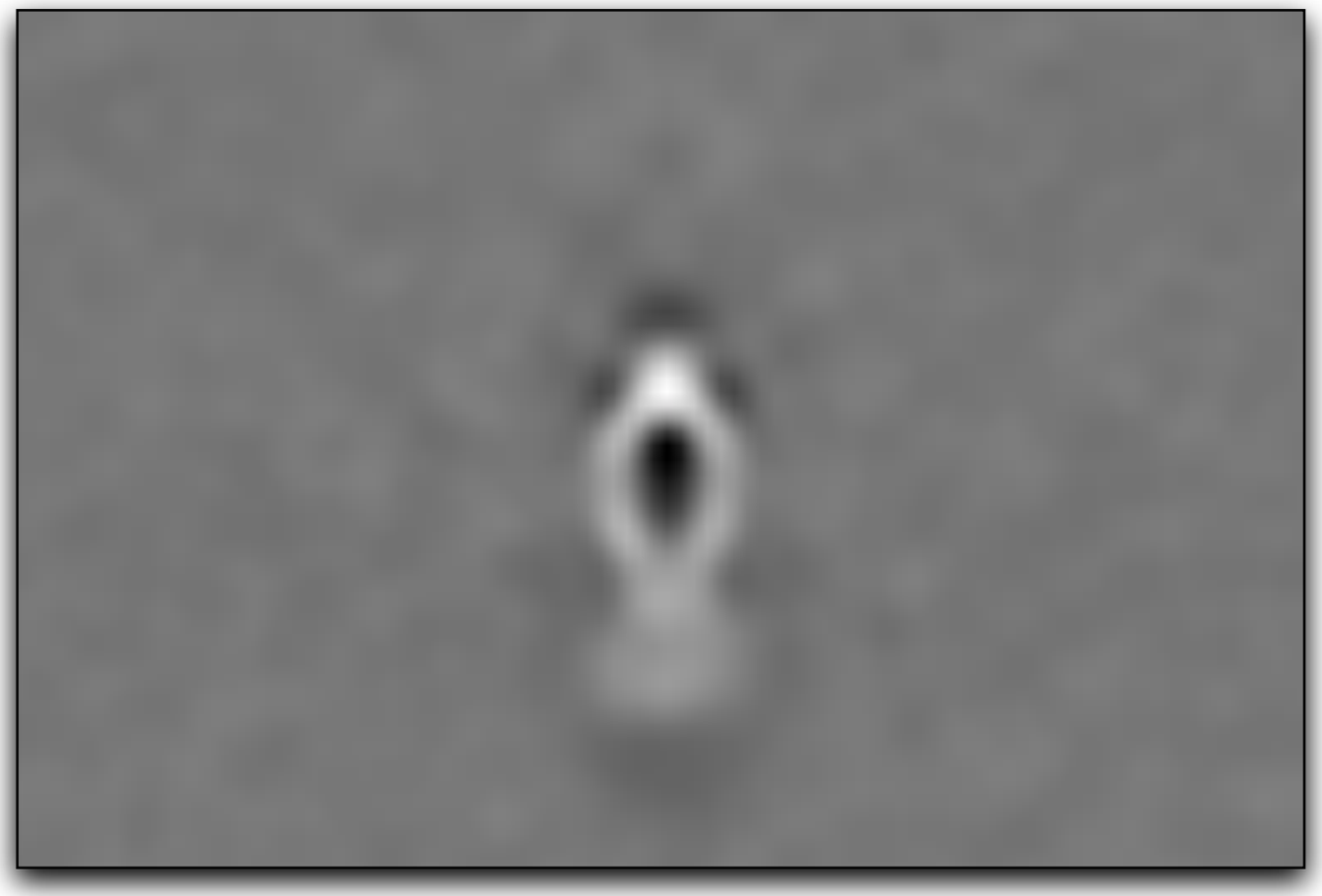
ASEF Filter

$$H^* = \frac{1}{N}\sum_{i=1}^{N}\frac{G_i}{F_i}$$

Exact Filters



AVERAGE



Original

Preprocessed

Target

Exact Filter

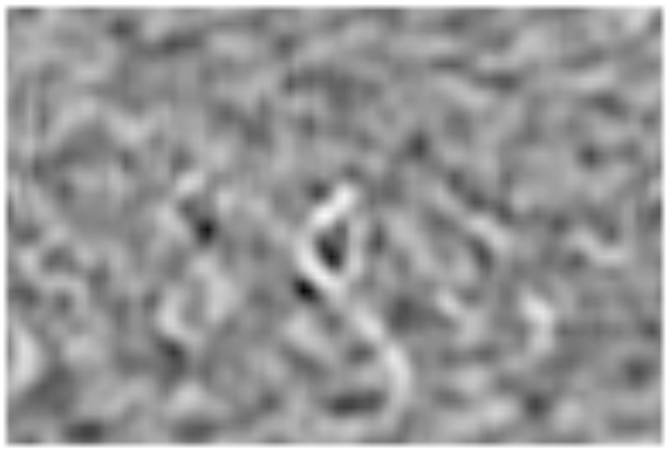
ASEF Filter

Colorado State University

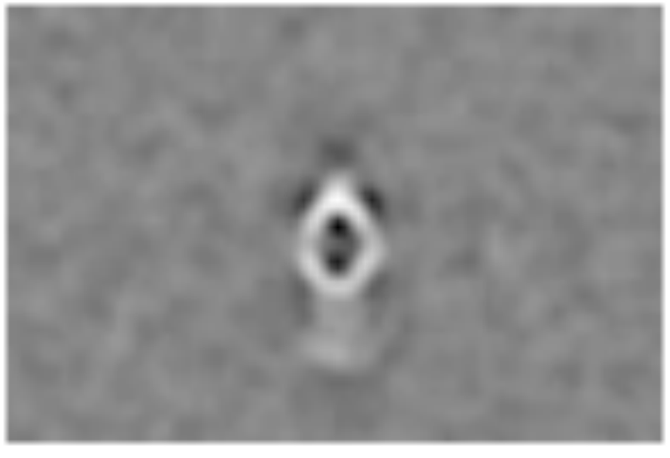
Original

Preprocessed

Target



Exact Filter



ASEF Filter

Training in realtime

Original

Preprocessed

Target

Exact Filter

ASEF Filter

Colorado State University

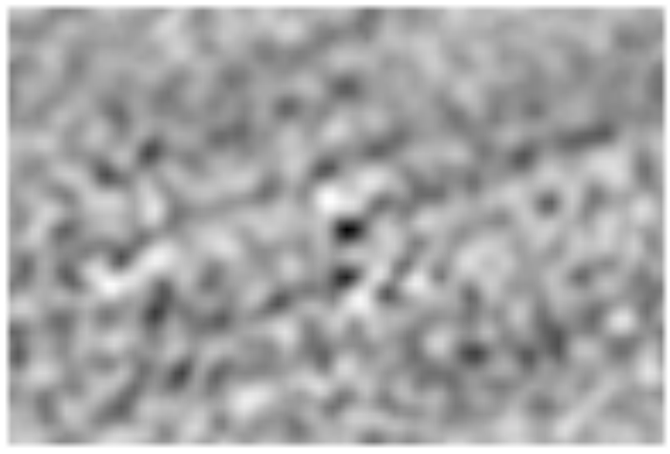
Training in realtime

Original

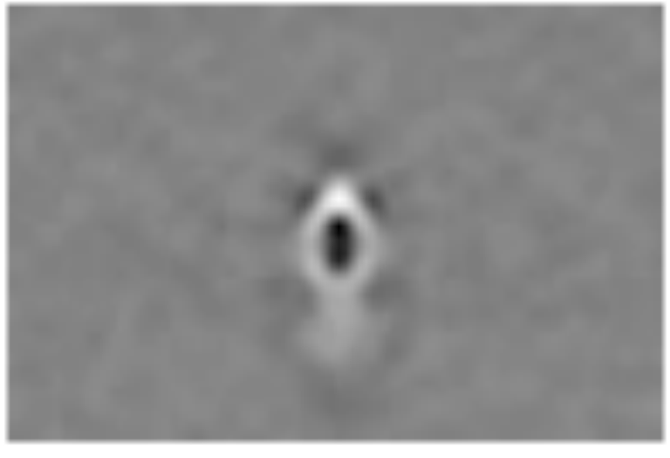
Based on 25 manually annotated frames

Preprocessed

Target



Exact Filter



ASEF Filter

Total Training Time = 12 seconds

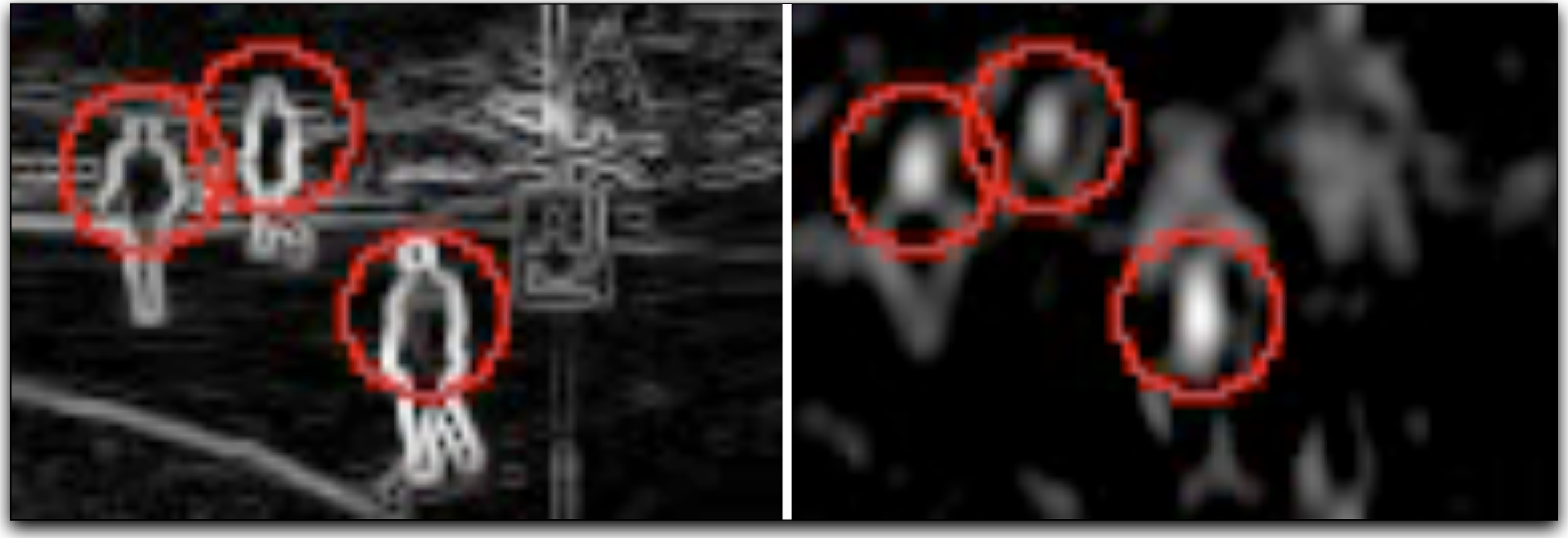
Colorado State University

Detection

Find the local maxima that exceed a given threshold.

Preprocessed Input

Correlation Output

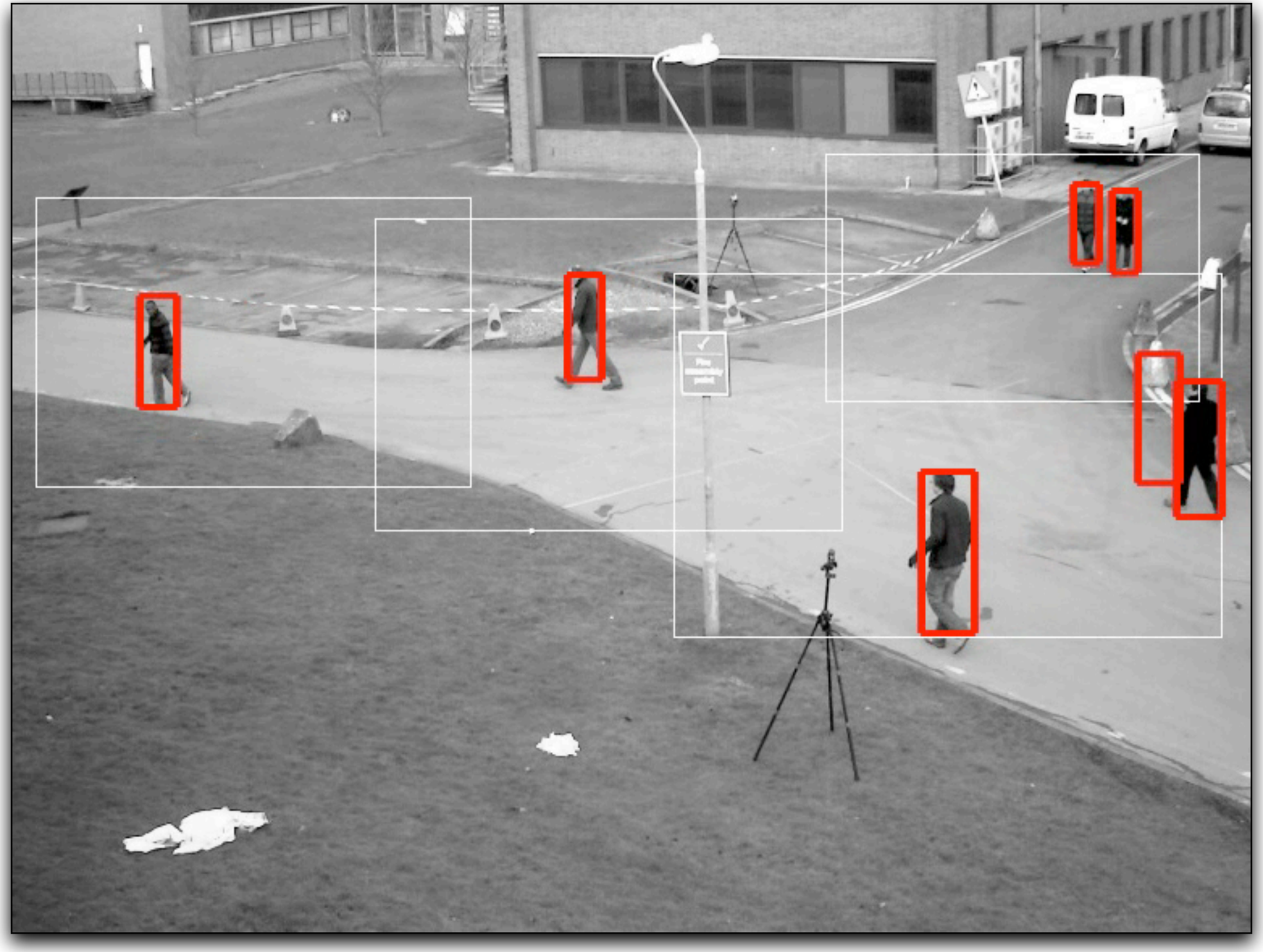


Other Algorithms

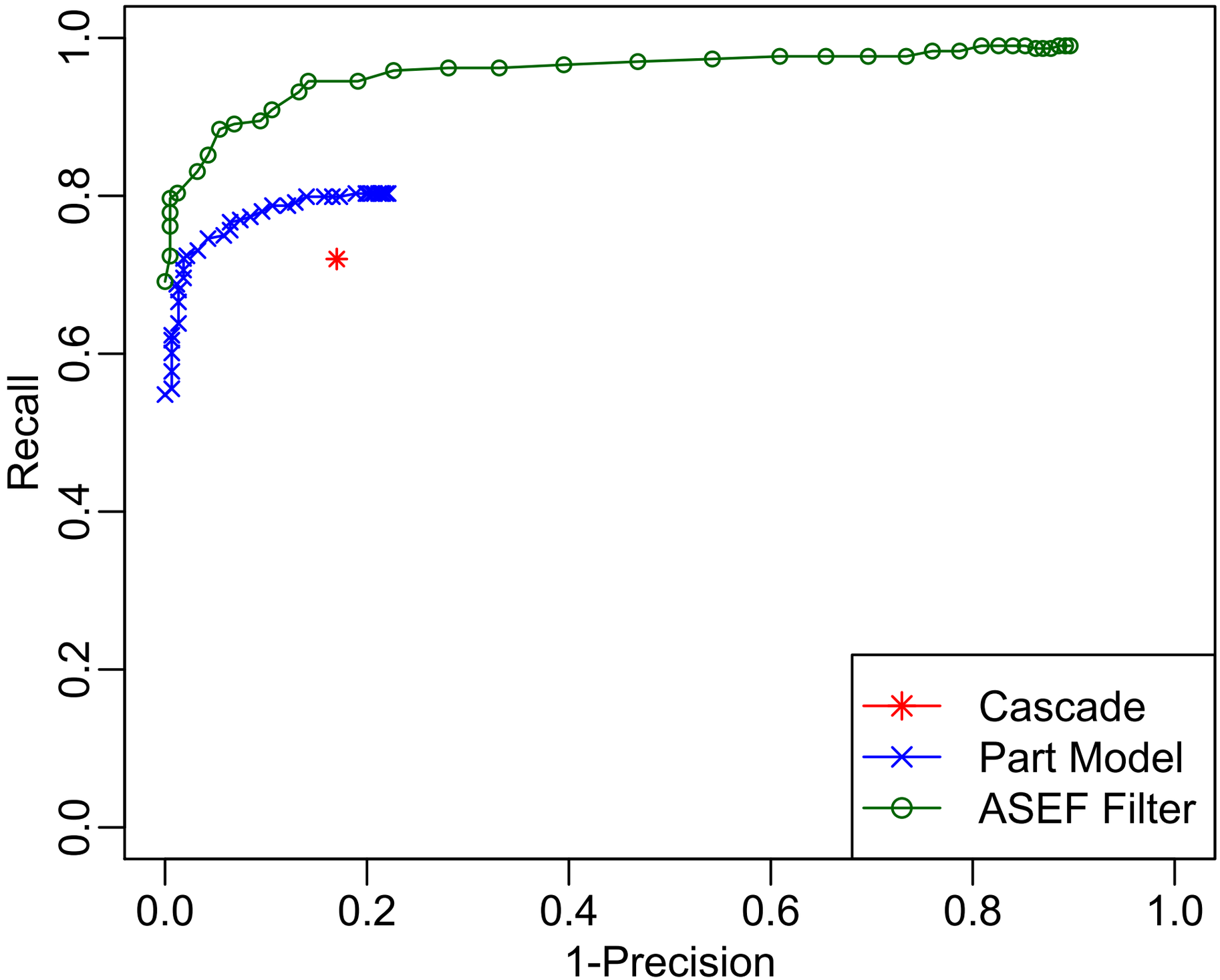
- **OpenCV Cascade Detector** - Viola and Jones. 2001.
  - Well known
  - Real time

- **Parts Based** - P. Felzenszwalb, et. al. 2008.
  - State of the Art
  - Prior Baseline

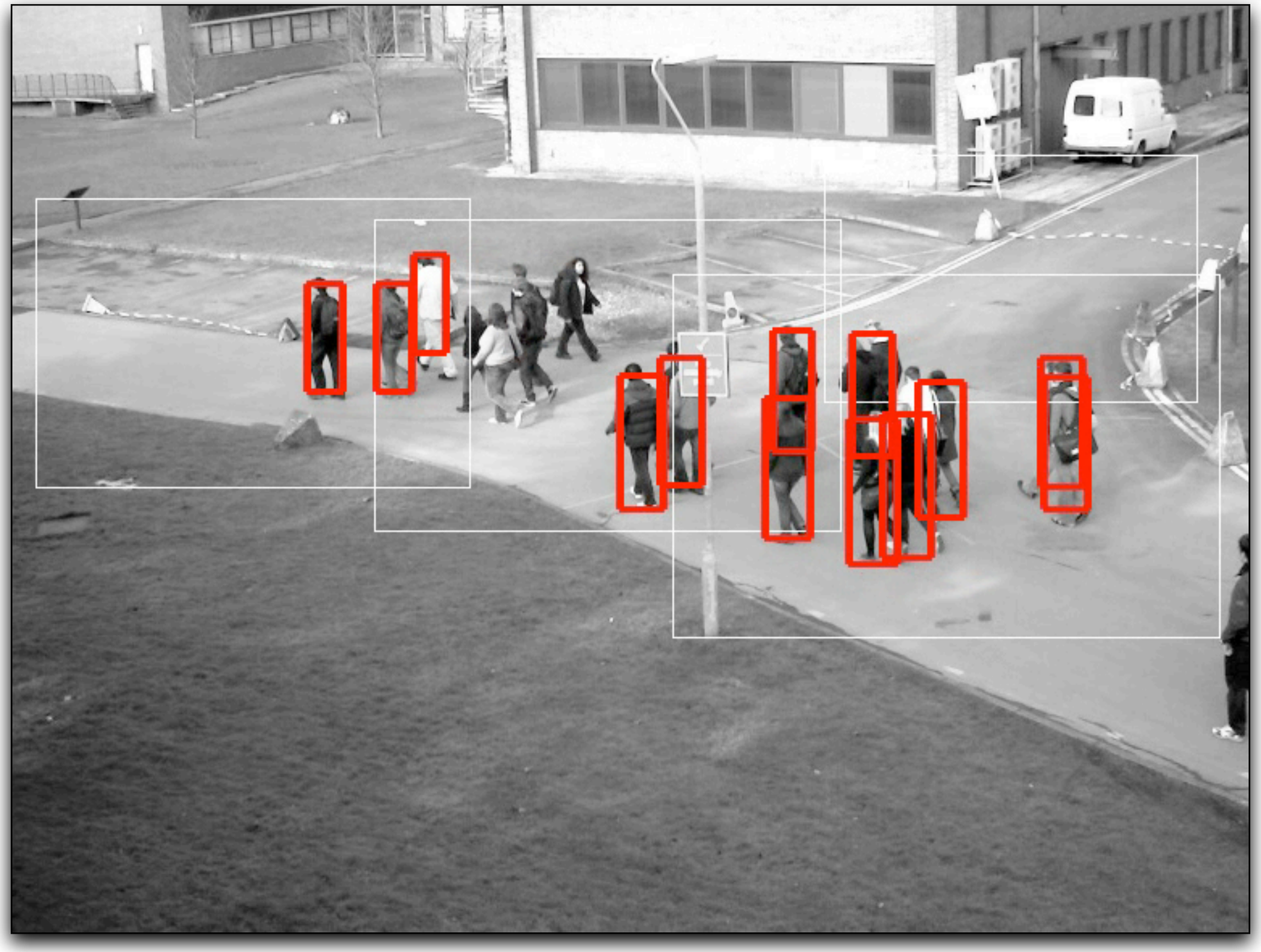
Sparse Sequence



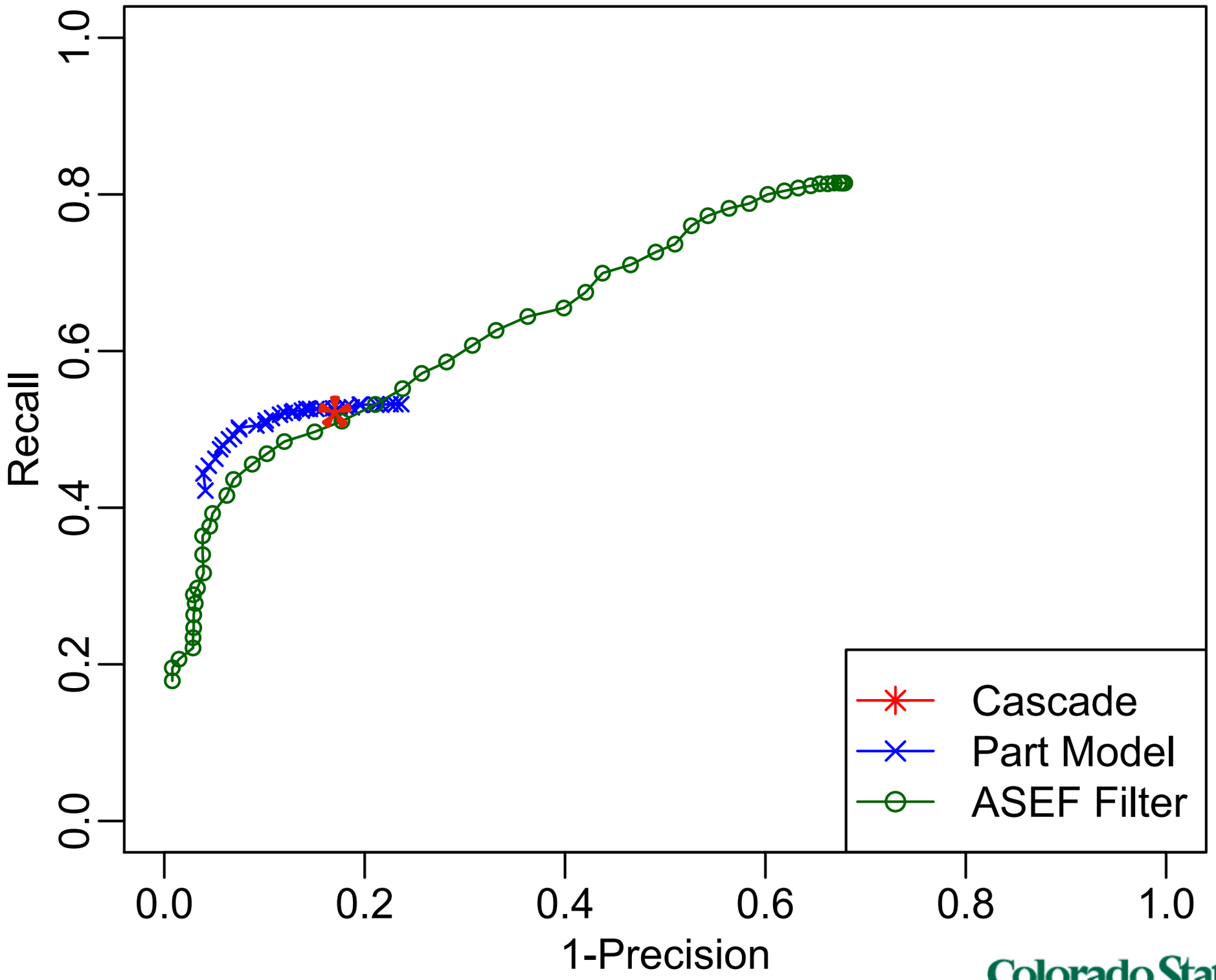
Sparse Sequence



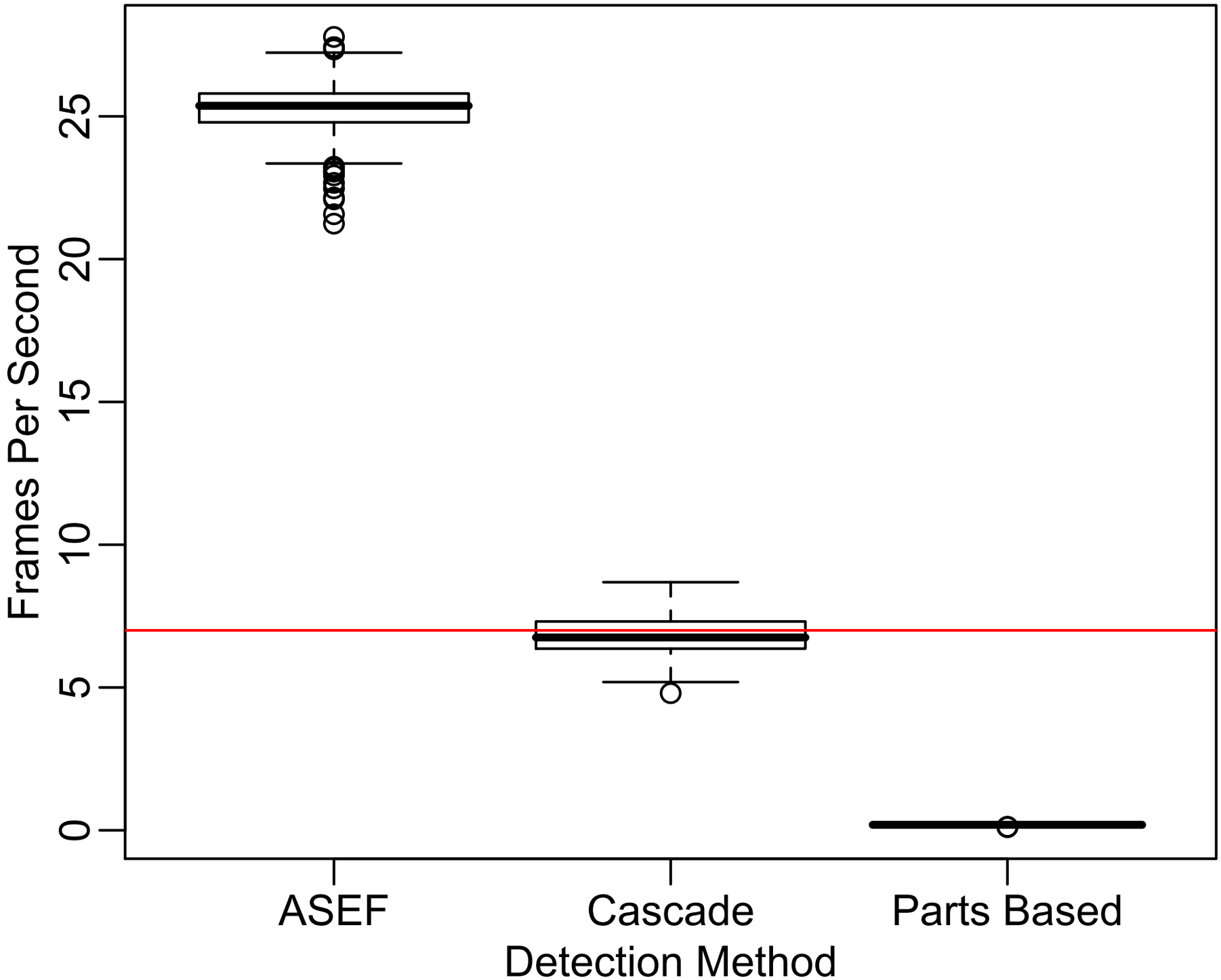
Crowded Sequence



Crowded Sequence



Detection Speed



Visual Tracking

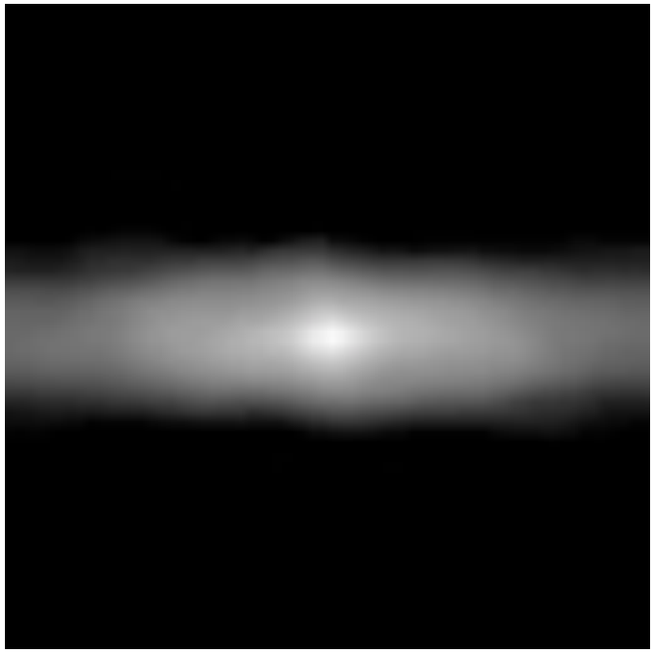
- MOSSE = Minimum Output Sum of Squared Error.
- Use ASEF or MOSSE to produce better peaks.
- Adapt to the changing appearance of the target.
- State of the Art tracking capability.
- Fast track updates: 669 fps

Simple Template

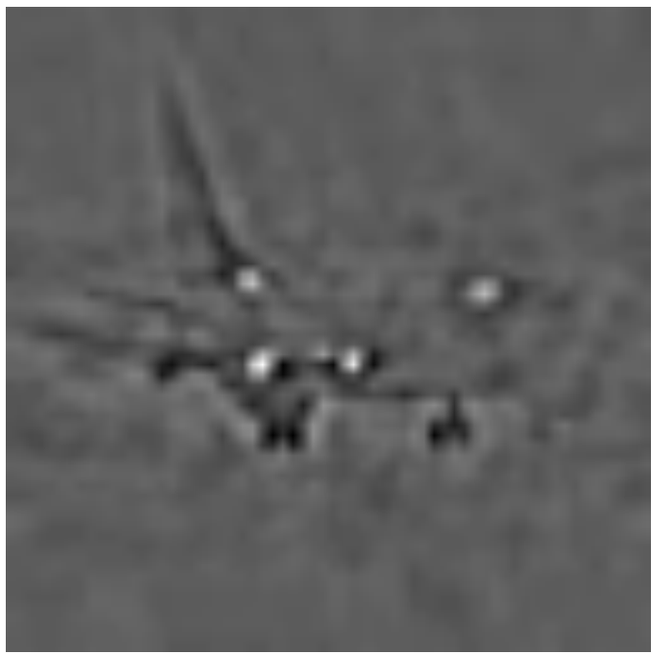
MOSSE Filter

Input, Filter, and Output

Mean Template



MOSSE



Failure Detection

- Peak-To-Sidelobe Ratio (PSR)

$$PSR = \frac{g_{\max} - \mu_{\mathtt{sl}}}{\sigma_{\mathtt{sl}}}$$

- Measures the strength of the output peak.
- For MOSSE or ASEF it can be used to determine occlusion or tracking failure.

MOSSE and ASEF

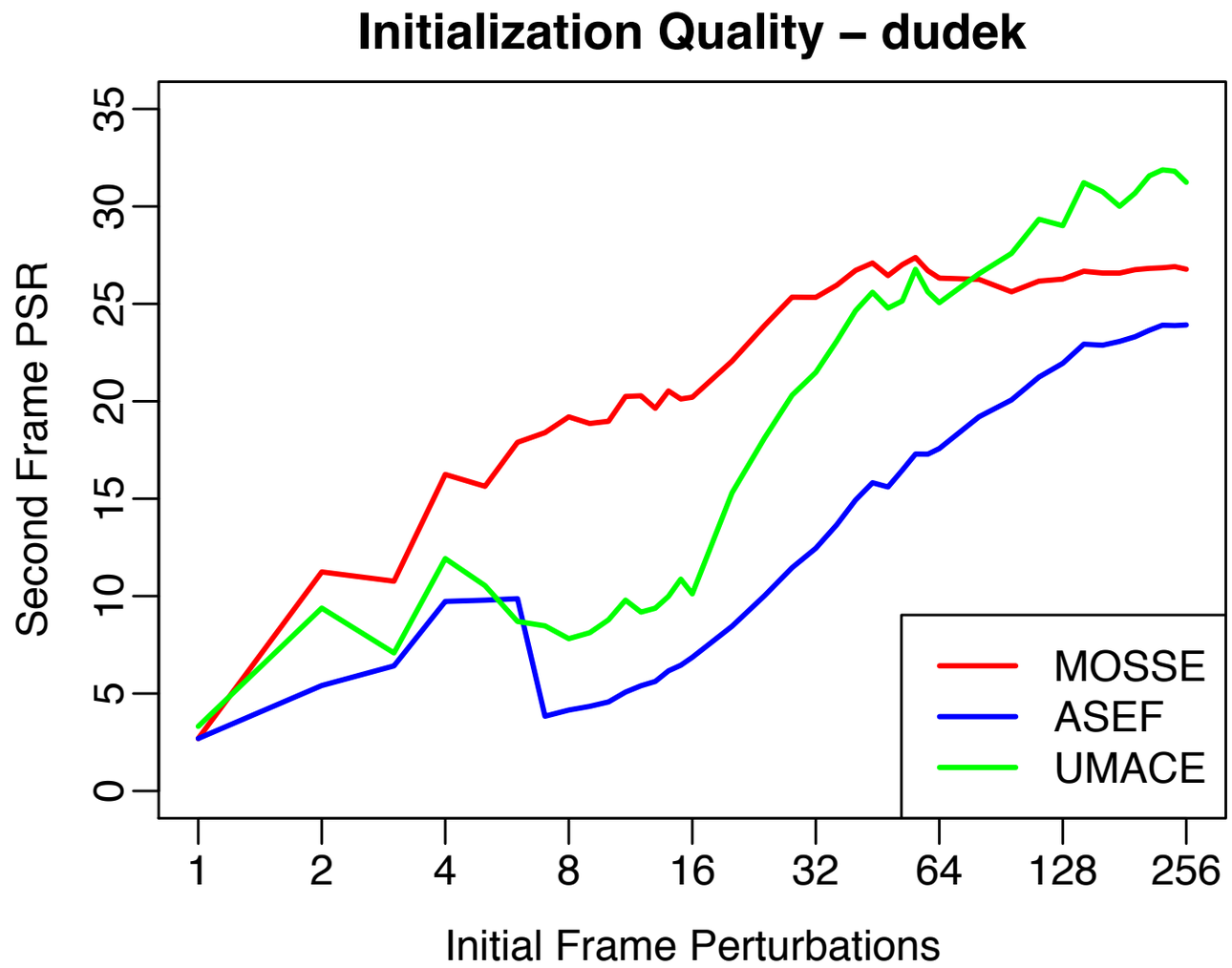
- MOSSE is in many ways similar to ASEF but is more stable when trained on just a few images.

ASEF

$$H^* = \frac{1}{N} \sum_i \frac{G_i \odot F_i^*}{F_i \odot F_i^*}$$

MOSSE

$$H^* = \frac{\sum_i G_i \odot F_i^*}{\sum_i F_i \odot F_i^*}$$

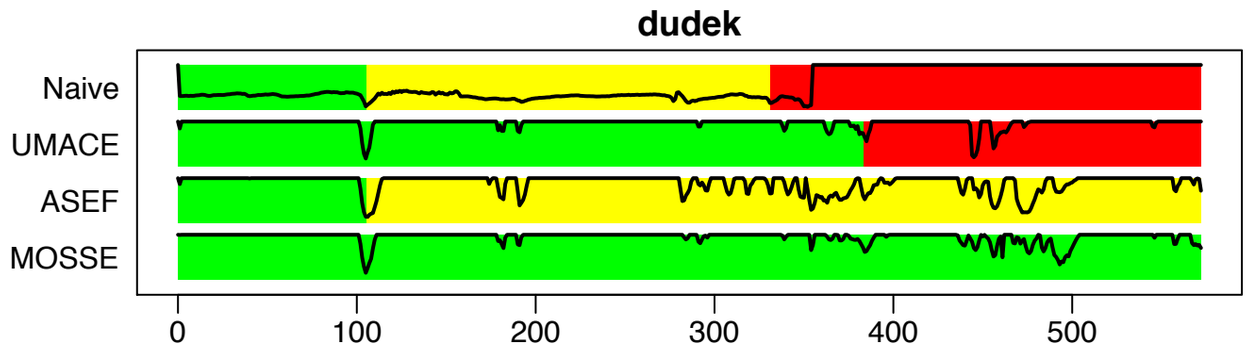


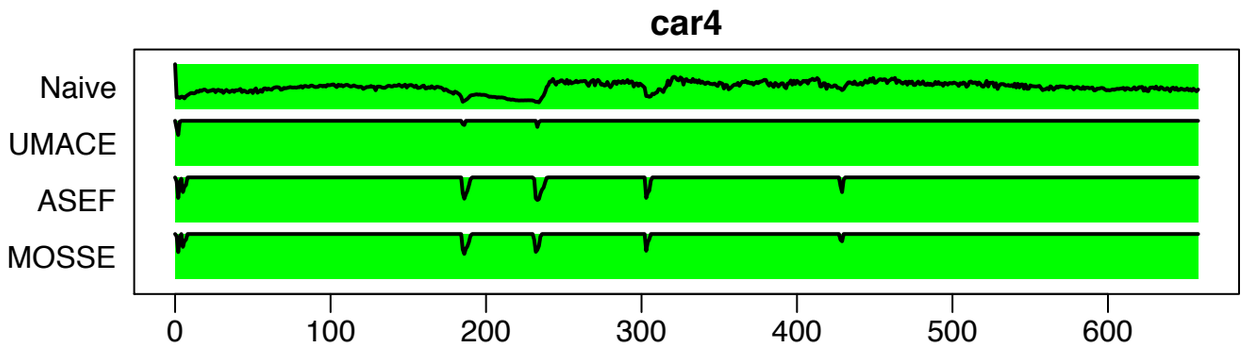
Tracking Algorithm

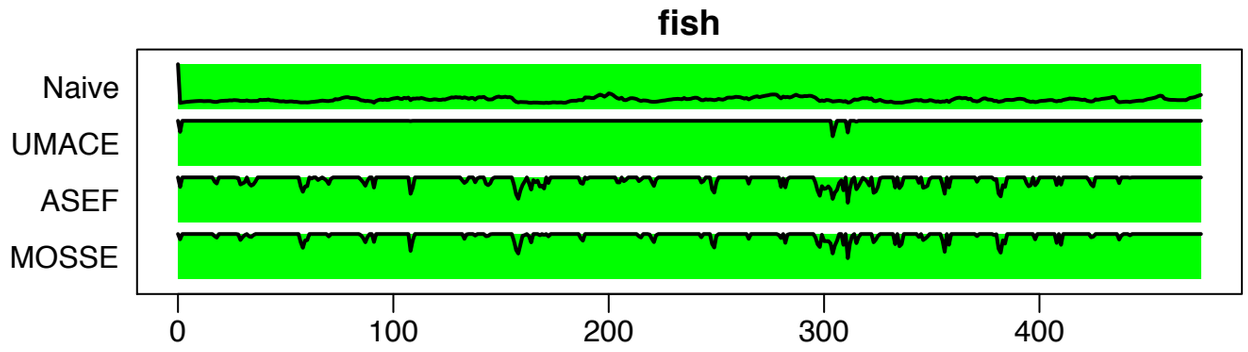
- Extract and rescale the tracking window.
- Correlate with the current filter to get the peak.
- Generate synthetic output with the peak at the new location.
- Update the filter using a running average.
- Update the track center to the new location.

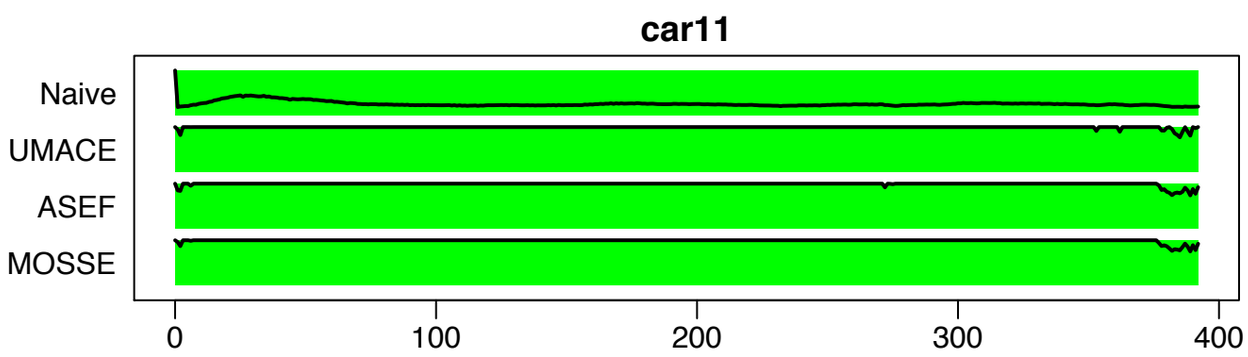
Tracking Ability

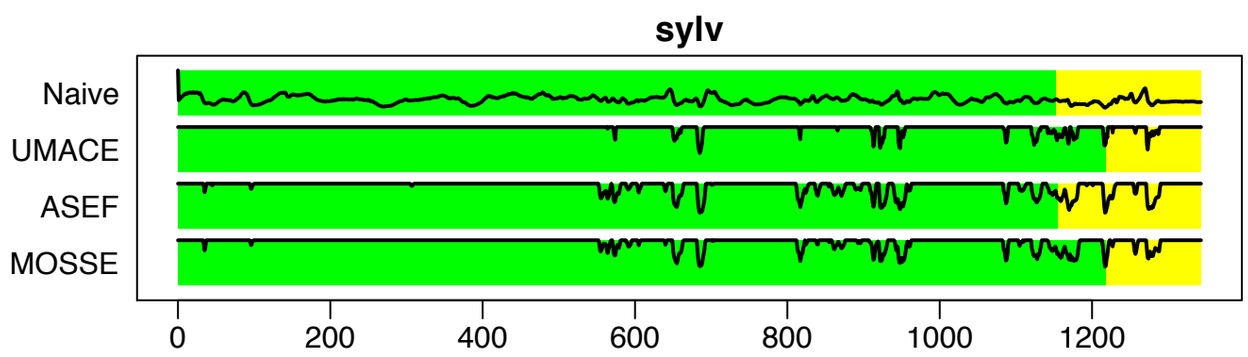
- Green - Good Track
- Yellow - Moderate Drift
- Red - Failed Track

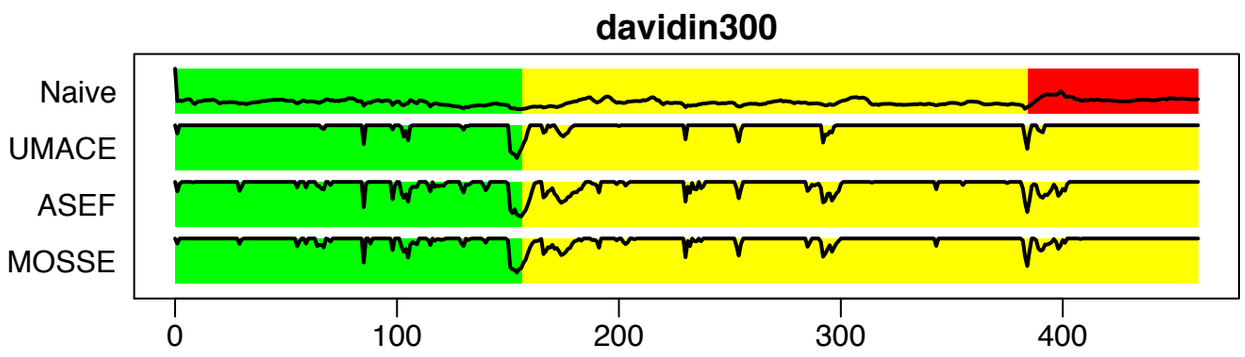


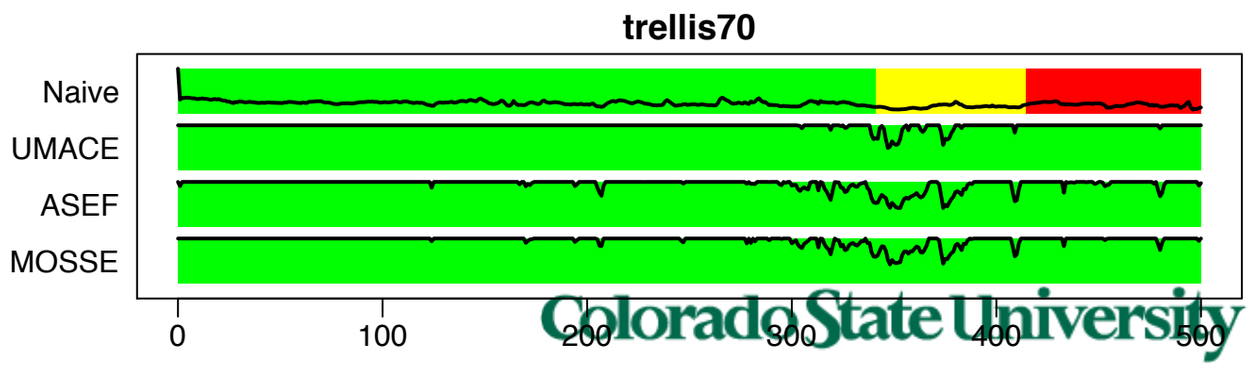




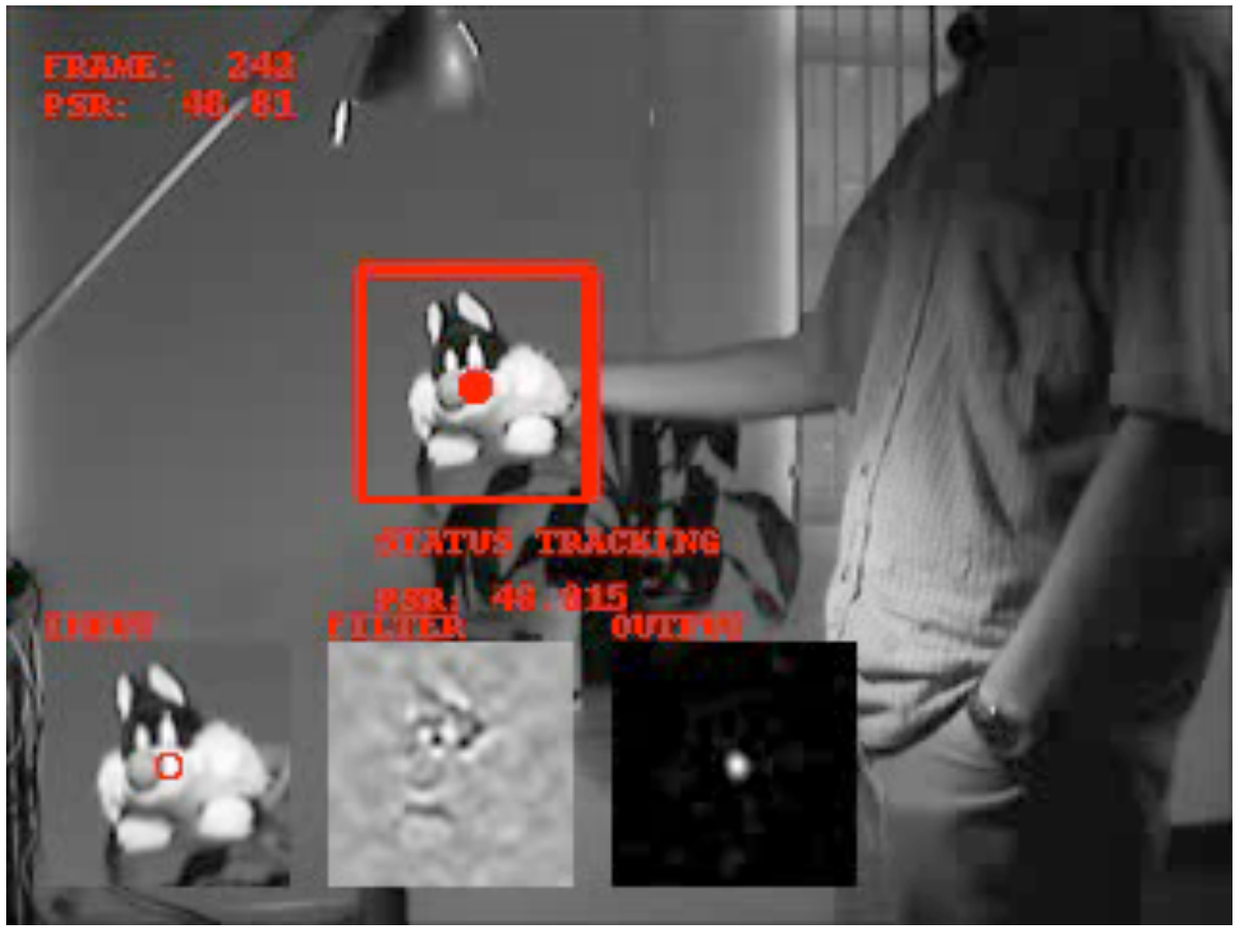






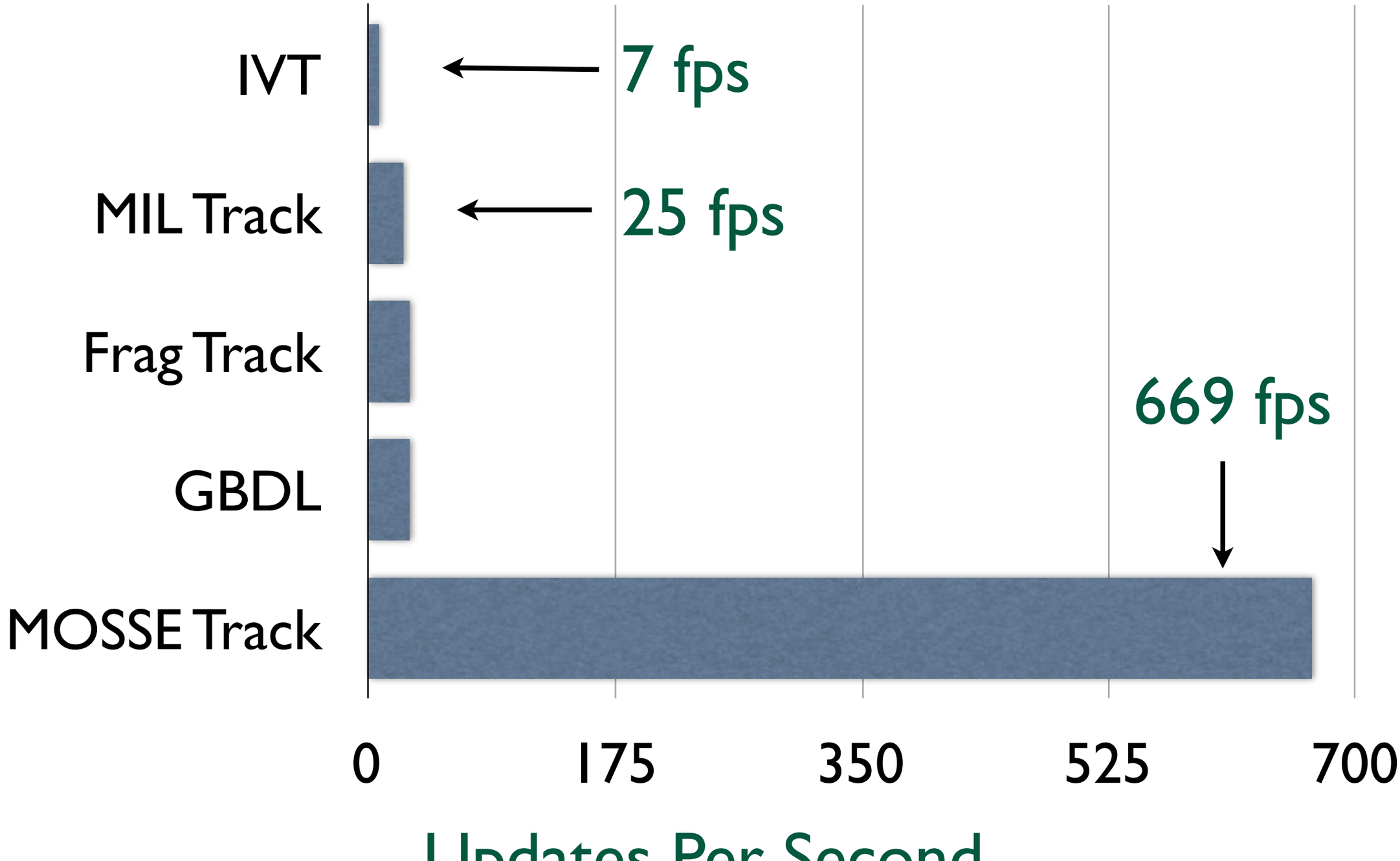


MOSSE vs. IVT


FRAME: 242
PSR: 46.81
STATUS TRACKING
PSR: 46.815
INPUT
FILTER
OUTPUT


243

Tracking Speed



Conclusions

- ASEF filters are easily trained for detection tasks.
- The concept is simple, map the training input to desired output.
- Better background suppression, stronger peaks.
- The recall and precision scores are better
- Much faster than the alternatives.

Tips for better tracking

- Only Convolve with a small portion of the input image.

  Focus on the Target. Boost Speed. Eliminate False Positives.

- Recenter Pixel Values: Zero Mean/One Standard Deviation.

  Average output of Zero. Reduces edge effects.

- Windowing: Cosine Window/Hamming Window

  Focus on the center of the image. Eliminate Edge Effect.

- Running Average: $\bar{H}_n = \lambda H_n + (1 - \lambda)\bar{H}_{n-1} \qquad \lambda = 0.05$

  Improves stability. Less susceptible to noise or occlusion.

- Use a technique such as MOSSE.