Tracking by learning

Arnold W.M. Smeulders

Tracking

Online tracking is to *determine the location of one target in video* starting from a bounding box in the first frame.

When conceived as an instant learning problem, the task is to discriminate object from background on the basis of N=1 sample (in the first frame) and N=k samples more (as long as the tracking is successful over k+1 frames).

So it is a hard and complex machine learning problem.

Tracking

Online tracking is to *determine the location of one target in video* starting from a bounding box in the first frame.

They consist at least of:

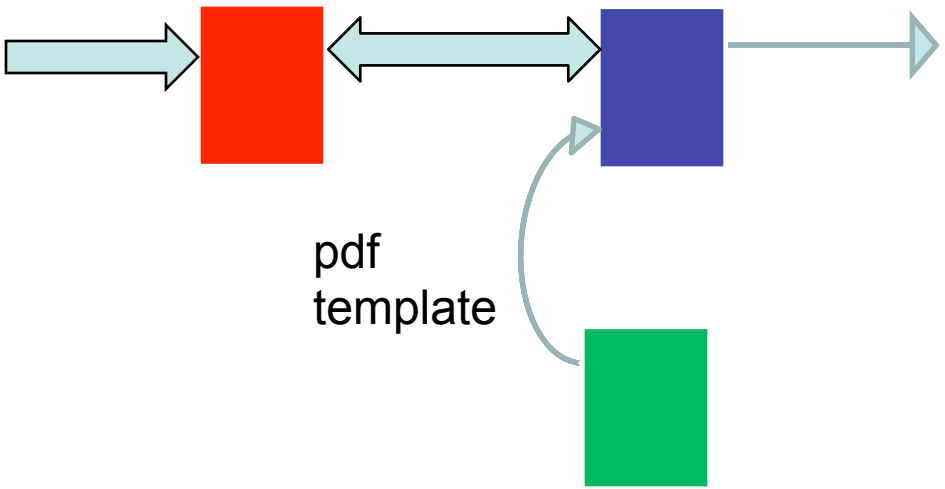
- a module observing the features of the image.
- a module selecting the actual motion.
- a module holding the internal representation of the object.
- a module updating the representation of the object.

Since ten years, trackers consist of learned observations.

Not a stupid tracker

The oldest, simplest and still good(!) non-discriminative tracker.

- Intensity values in the candidate box.
- Direct target matching by Normalized Cross-Correlation.
- Intensity values in the initial target box as template.
- No updating of the target.

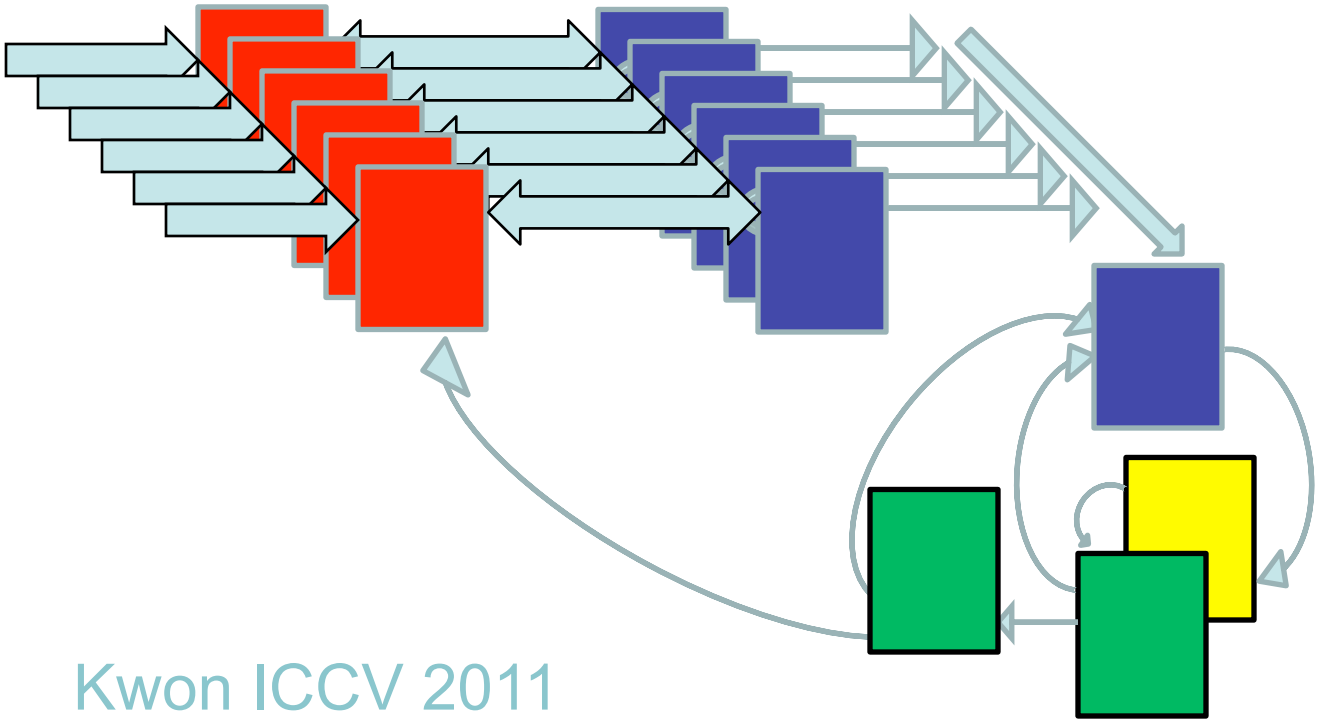


1970? Briechle SPIE 2001

TST The best non-discriminative

Tracking by Sampling Trackers is the best non-discriminative.

- HIS-color edges of many different trackers.
- Best match in image, followed by best state.
- Trackers store eigen images. State stores x, s, score.
- Sparse incremental PCA image representation with leaking.



Kwon ICCV 2011

Discriminative Trackers

In discriminative trackers, the emphasis on learning the current distinction between object and background.

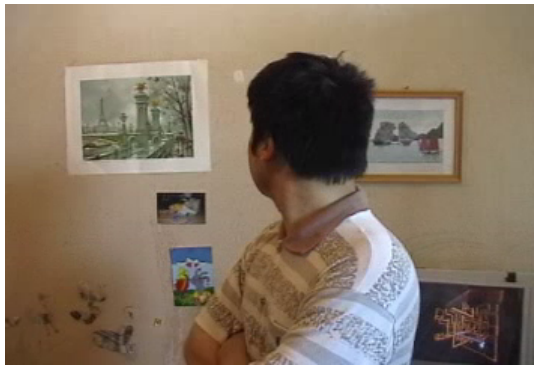
We discuss an old version: the Foreground – Background tracker.

Discriminative Trackers

Minor viewpoint change



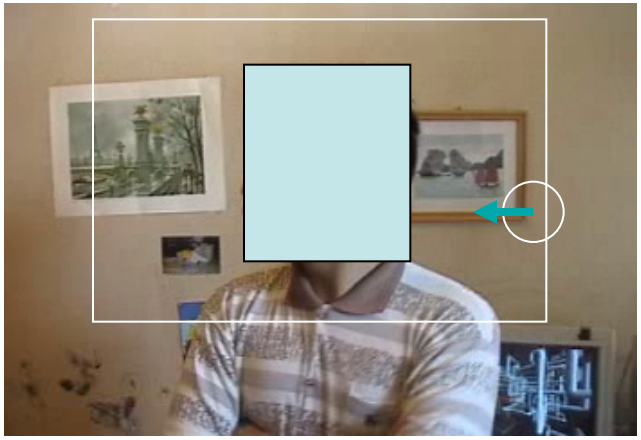
Severe viewpoint change


Nguyen IJCV 2006

Discriminative Trackers

The hole in the background leaves object entirely free:
The object may change abruptly in pose.

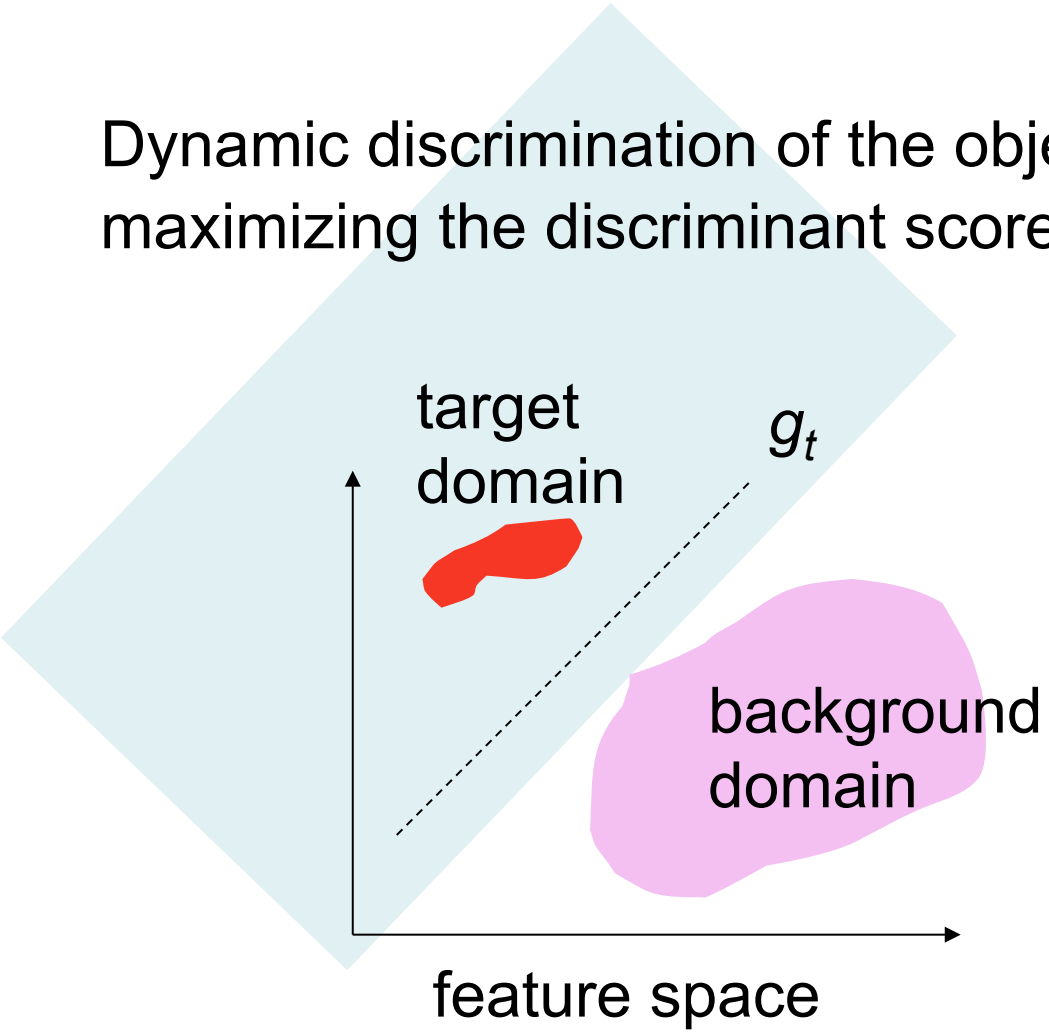


The background varies slower:
Background is better predictable.

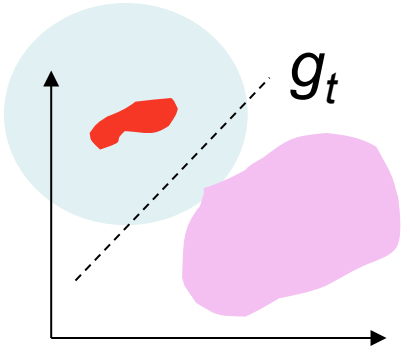
General scheme: Get foreground and background patches
+ Learn a classifier + Classify patches from new image.

Discriminative Trackers

Dynamic discrimination of the object from its background while maximizing the discriminant score of the target region.

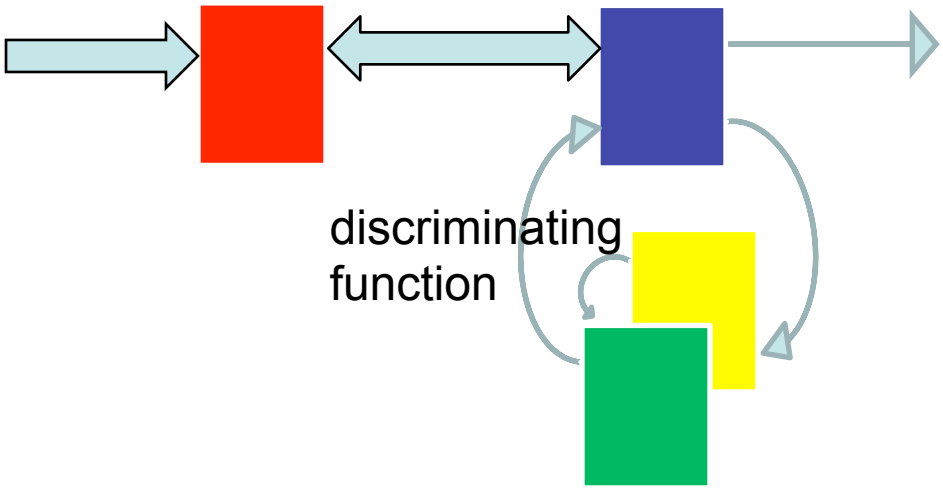


Much larger permitted deviation for target appearance than match



Foreground-Background Tracker

- SURF texture samples from target / background box.
- Trains a linear discriminant classifier.
- Classifier is foreground/background model (in feature space).
- Updated by a leaking memory on the training data.



Nguyen IJCV 2006, Chu 2012

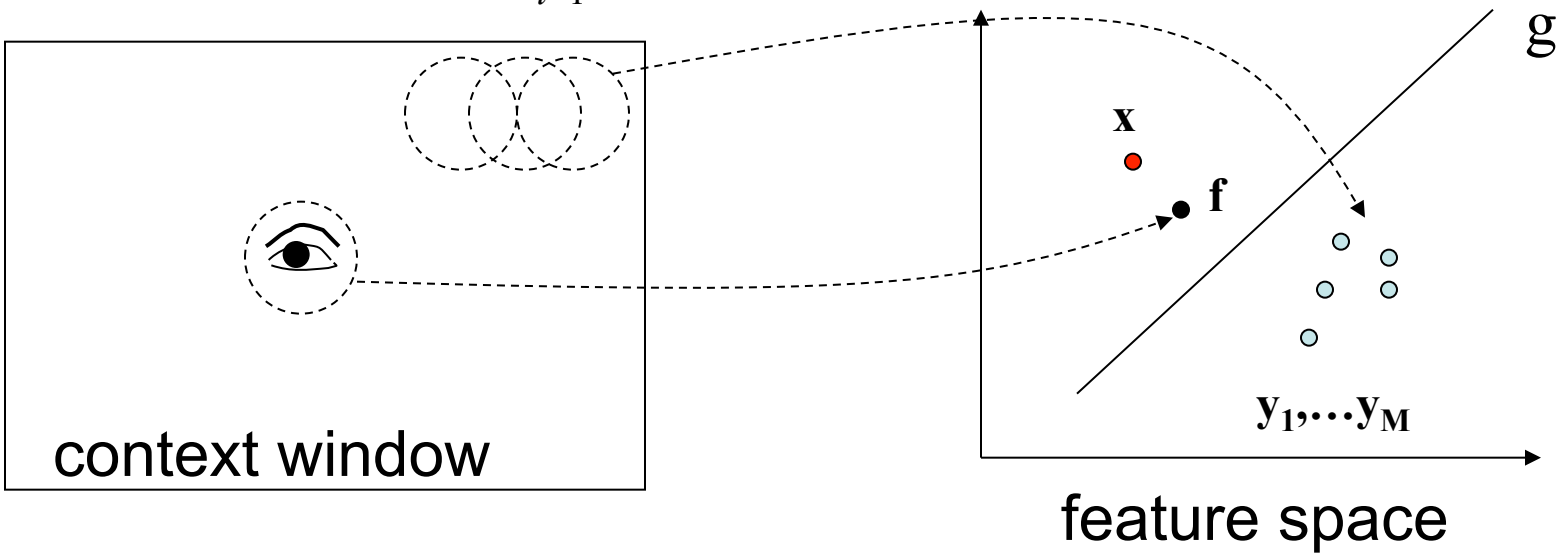
Foreground Background Classifier

Discriminant function

$$g(\mathbf{f}) = \mathbf{a}.\mathbf{f} + \mathrm{b} \rightarrow \max_{\text{target location}}$$

Train *g* by adopting linear discriminant analysis:

$$[g(\mathbf{x}) - 1]^2 + \sum_{i=1}^{M} \alpha_i [g(\mathbf{y}_i) + 1]^2 + \lambda \frac{|\mathbf{a}|^2}{2} \rightarrow \min_{\mathbf{a},b}$$



Foreground-Background Classifier

The solution is obtained in closed incremental form:

$$\mathbf{a} \propto [\lambda \mathbf{I} + \mathbf{B}]^{-1}[\mathbf{x} - \overline{\mathbf{y}}]$$

The weighted mean vector of background patterns:

$$\overline{\mathbf{y}} = \sum_{i=1}^{M} \alpha_i \mathbf{y}_i$$

The weighted covariance matrix:

$$\mathbf{B} = \sum_{i=1}^{M} \alpha_i [\mathbf{y}_i - \overline{\mathbf{y}}][\mathbf{y}_i - \overline{\mathbf{y}}]^T$$

Mean and covariance can be updated incrementally.

Foreground-Background Updating

The foreground template is updated in every frame:

$$\mathbf{x} = (1-\gamma)\mathbf{x}_{prev} + \gamma\mathbf{f}_{optimal}$$

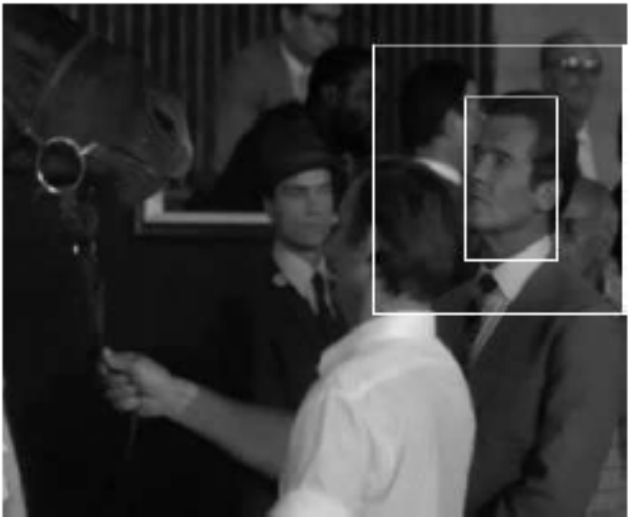
New patterns are added to the background patterns.

Background patterns are summed with leaking coefficients α_i.

New and old patterns predict mean $\mathbf{y}$ and cov **B** incrementally.

Foreground-Background Results

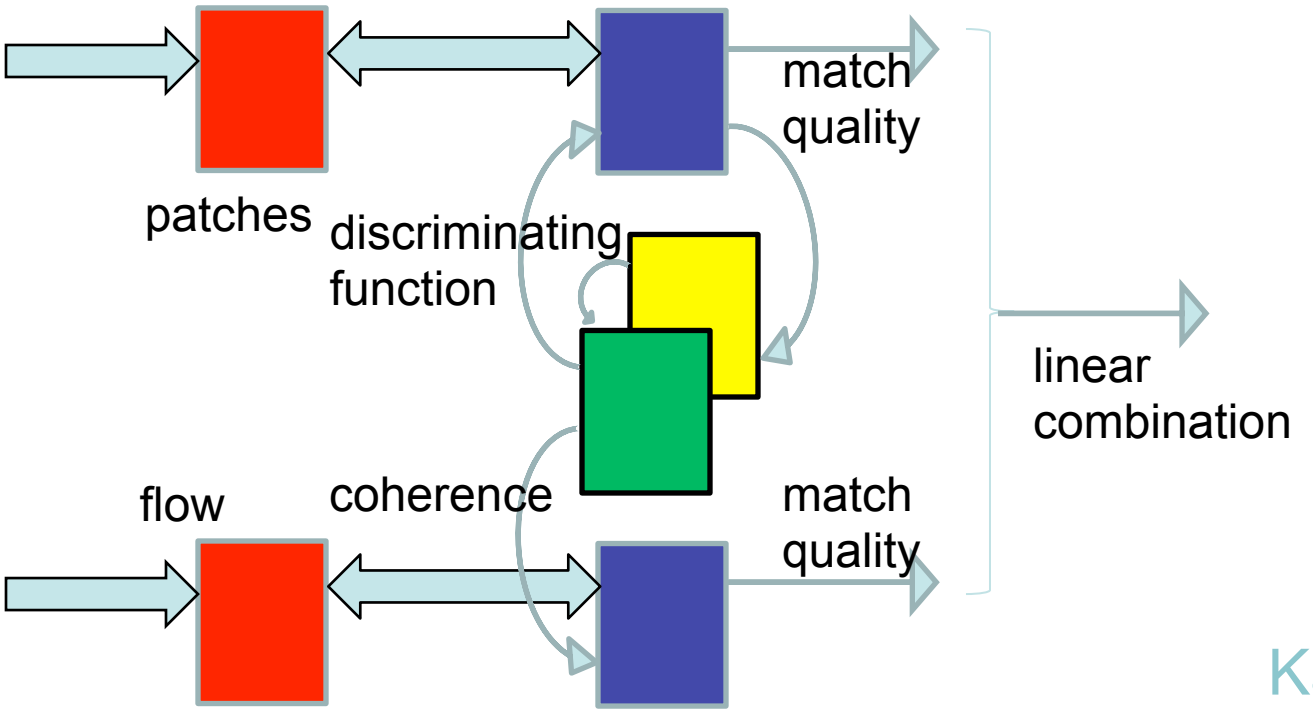


Tracking, Learning, Detecting

Tracking, Learning and Detecting

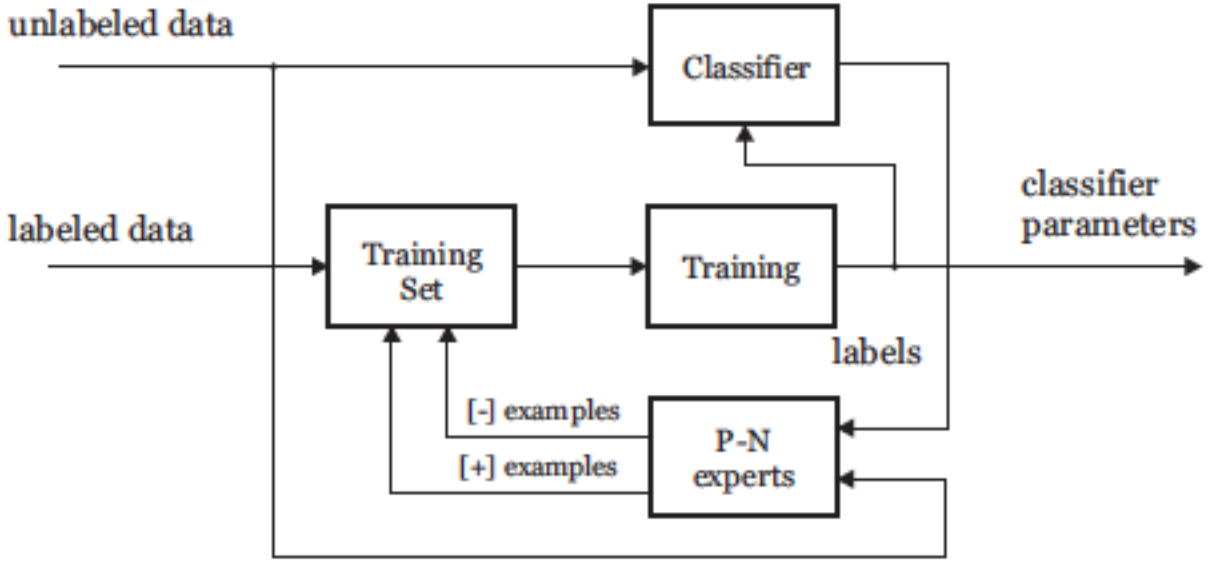
- Optic flow patches + Intensity patches.
- Discriminant on median flow + Normalized Cross Correlate.
- Weights of the classifier + Template of target.
- Experts label update + Recovery when lost.



Kalal CVPR 2010

Tracking, Learning and Detecting

At the core of TLD are the Positive – Negative experts.



The P-expert classifies negatives adding the false negatives, by using the reliable parts of the temporal position of the target by maintaining a core recent target model. Vice versa, the N-expert uses the spatial layout of the target.

Kalal CVPR 2010

Structured SVM Tracker

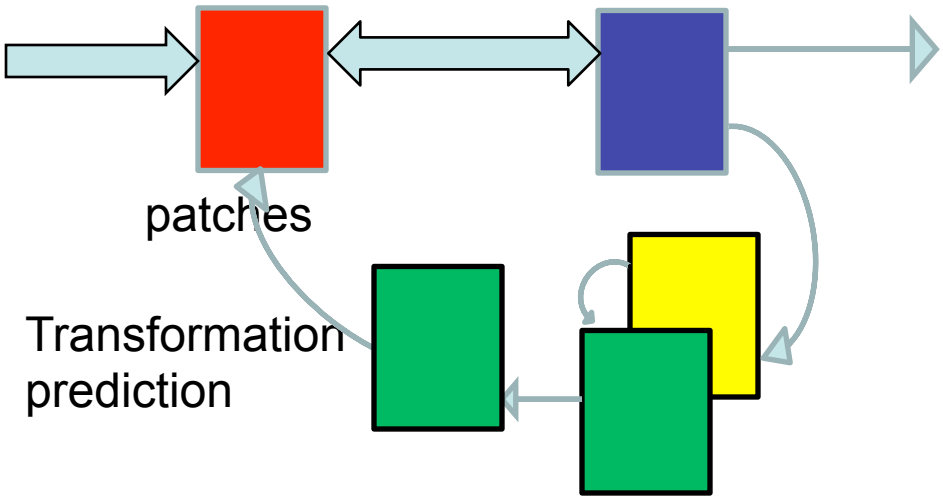
STRuctured output tracking

Windows by Haar features with 2 scales.

Structured SVM by {app, translation}, no labels.

Structured constraints + Transformation prediction.

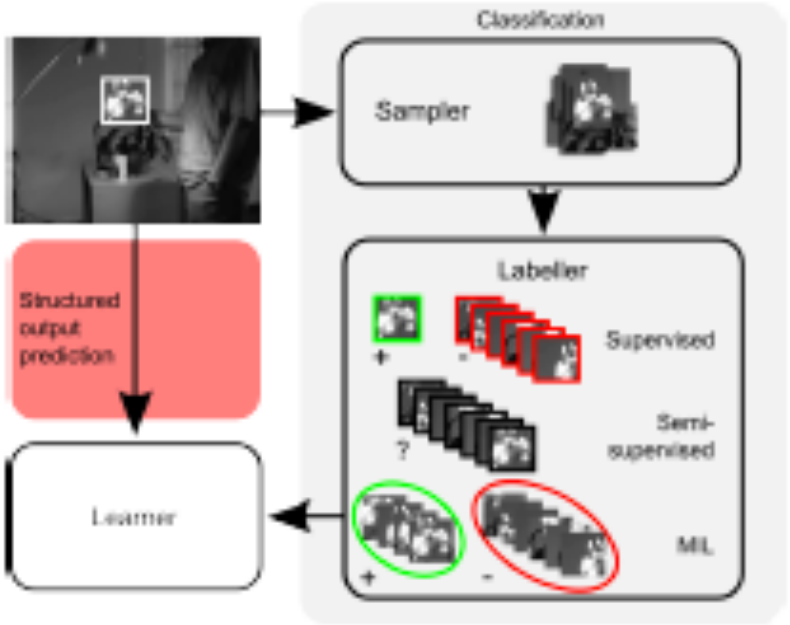
Update the constraints to stay at current **x**.



Hare ICCV 2011

STRuctured output tracking

The basic observation: When a tracker-classifier is used samples are first given a label and then used in learning.



This causes label noise. A better way is to directly output the displacement via structured SVM.

Hare ICCV 2011

STRuctured output tracking

In STR, a labeled example is (**x**,**y**) where **x** is the observed state and **y** is the desired transformation.
The objective function on joint kernel map $\Phi(\mathbf{x},\mathbf{y})$ is:

$$\begin{aligned}
\min_{\mathbf{w}} \quad & \frac{1}{2}\|\mathbf{w}\|^2 + C\sum_{i=1}^{n}\xi_i \\
\text{s.t.} \quad & \forall i:\ \xi_i \geq 0 \\
& \forall i, \forall \mathbf{y} \neq \mathbf{y}_i:\ \langle \mathbf{w}, \delta\Phi_i(\mathbf{y})\rangle \geq \Delta(\mathbf{y}_i,\mathbf{y}) - \xi_i
\end{aligned}$$

Can be rewritten into the online version:

$$\begin{aligned}
\max_{\boldsymbol{\beta}} \quad & -\sum_{i,\mathbf{y}} \Delta(\mathbf{y},\mathbf{y}_i)\beta_i^{\mathbf{y}} - \frac{1}{2}\sum_{i,\mathbf{y},j,\bar{\mathbf{y}}} \beta_i^{\mathbf{y}}\beta_j^{\bar{\mathbf{y}}}\langle \Phi(\mathbf{x}_i,\mathbf{y}), \Phi(\mathbf{x}_j,\bar{\mathbf{y}})\rangle \\
\text{s.t.} \quad & \forall i, \forall \mathbf{y}:\ \beta_i^{\mathbf{y}} \leq \delta(\mathbf{y},\mathbf{y}_i)C \\
& \forall i:\ \sum_{\mathbf{y}} \beta_i^{\mathbf{y}} = 0
\end{aligned}$$

Hare ICCV 2011

STRuctured output tracking

The kernel function measures the effort to crop a patch on the target:

$$k_{xy}(\mathbf{x}, \mathbf{y}, \bar{\mathbf{x}}, \bar{\mathbf{y}}) = k(\mathbf{x}^{\mathbf{p} \circ \mathbf{y}}, \bar{\mathbf{x}}^{\bar{\mathbf{p}} \circ \bar{\mathbf{y}}}).$$

By averaging several kernels with gradients, histograms, tracking becomes more robust:

$$k(\mathbf{x}, \bar{\mathbf{x}}) = \frac{1}{N_k} \sum_{i=1}^{N_k} k^{(i)}(\mathbf{x}^{(i)}, \bar{\mathbf{x}}^{(i)})$$

Hare ICCV 2011

STRuctured output tracking

The loss function is based on the overlap score:

$$\Delta(\mathbf{y}, \bar{\mathbf{y}}) = 1 - s^o_{\mathbf{p}_t}(\mathbf{y}, \bar{\mathbf{y}}),$$

Updating is by inserting the true displacement as a positive support vector and the hardest by the loss function as a negative.

Older support vectors are removed at random when they loss functions shows too big a deviation.

Existing support vectors are reprocessed to update their weights given the current state.

Hare ICCV 2011

Data set

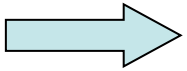
ALOV300++ dataset

Smeulders Dung et al PAMI 2014

13 Aspects & Hard Cases

Light	Disco light
Object surface cover	Person redressing
Object specularity	Mirror transport
Object transparency	Glass ball rolling
Object shape	Octopus swimming
Motion smoothness	Brownian motion
Motion coherence	Flock of birds
Scene clutter	Camouflage
Scene confusion	Herd of cows
Scene low contrast	White bear on snow
Scene occlusion	Object getting out of scene
Camera moving	Shaking camera
Camera zooming	Abrupt switch of lens
Length of sequence	Return of past appearance

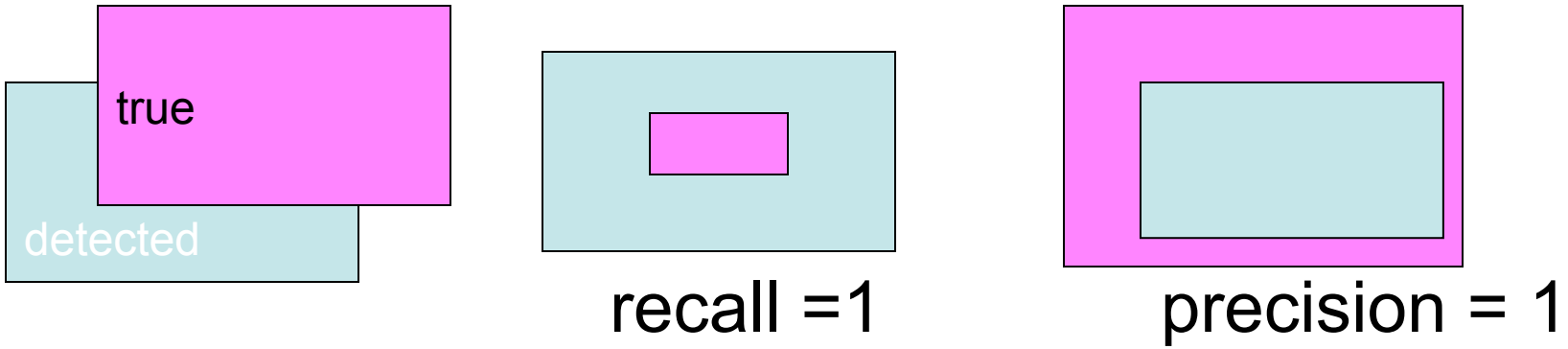
Hard Cases for Tracking



19 Assorted Trackers

1.	Normalised cross correlation	NCC	1970?
2.	Lucas Kanade tracker	LKT	1984
3.	Kalman appearance prediction tracker	KAT	2004
4.	Fragments-based tracker	FRT	2006
5.	Mean shift tracker	MST	2000
6.	Locally orderless tracker	LOT	2012
7.	Incremental visual tracker	IVT	2008
8.	Tracking on the affine group	TAG	2009
9.	Tracking by sampling trackers	TST	2011
10.	Tracking by Monte Carlo sampling	TMC	2009
11.	Adaptive Coupled-layer Tracking	ACT	2011
12.	L1-minimization Tracker	L1T	2009
13.	L1-minimization with occlusion	L1O	2011
14.	Foreground background tracker	FBT	2006
15.	Hough-based tracking	HBT	2011
16.	Super pixel tracking	SPT	2011
17.	Multiple instance learning tracking	MIT	2009
18.	Tracking, learning and detection	TLD	2010
19.	Structured output tracking	STR	2011

Success of tracking



f = detected .and. true / detected .or. true

Declared tracked when f > 0.5.

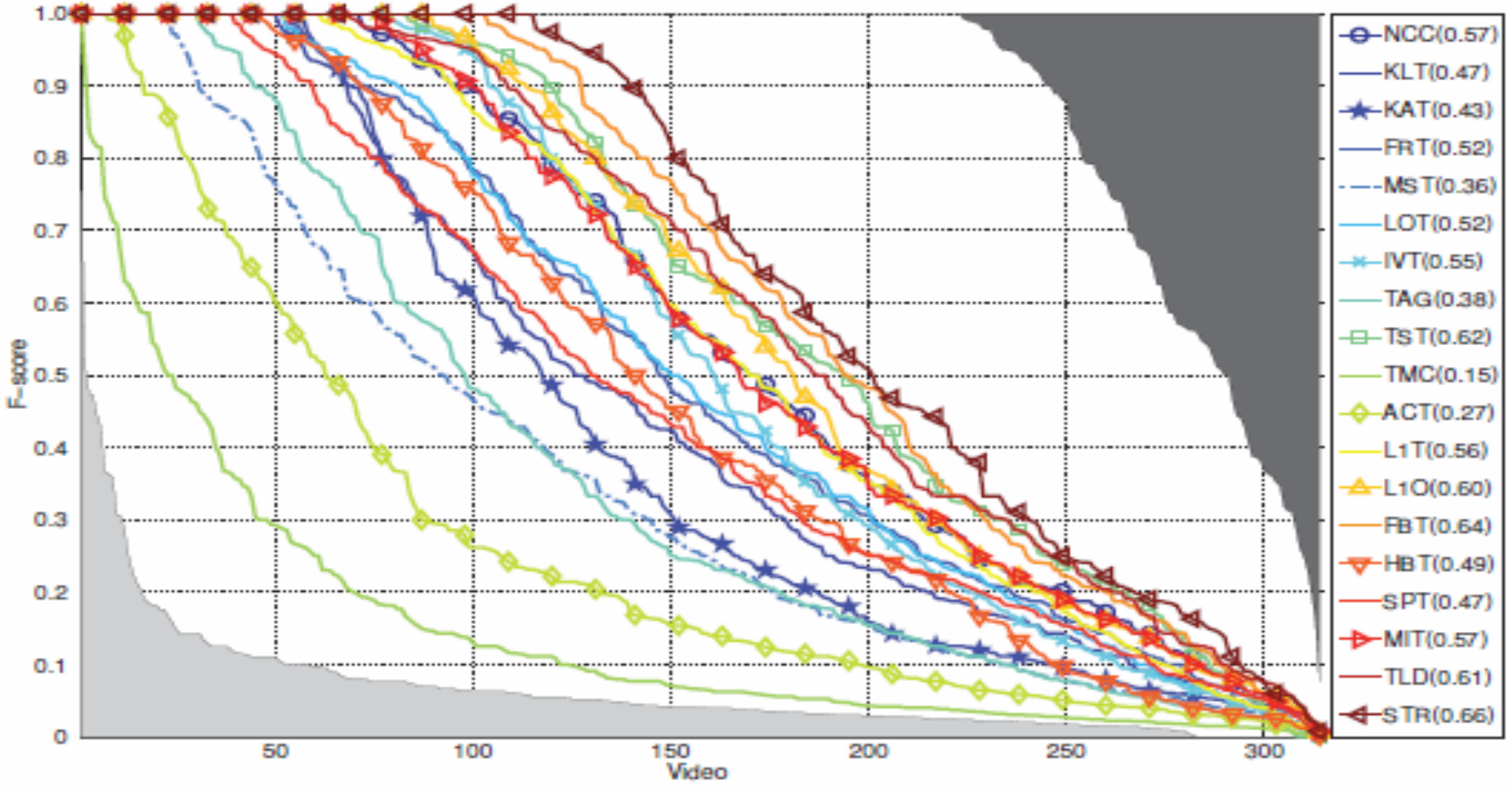
$F = \Sigma\, p\_i / 2N + \Sigma\, r\_i / 2N$

Kasturi PAMi 2009

Everingham IJCV 2010

Experimental results

Survival curves by Kaplan-Meijer



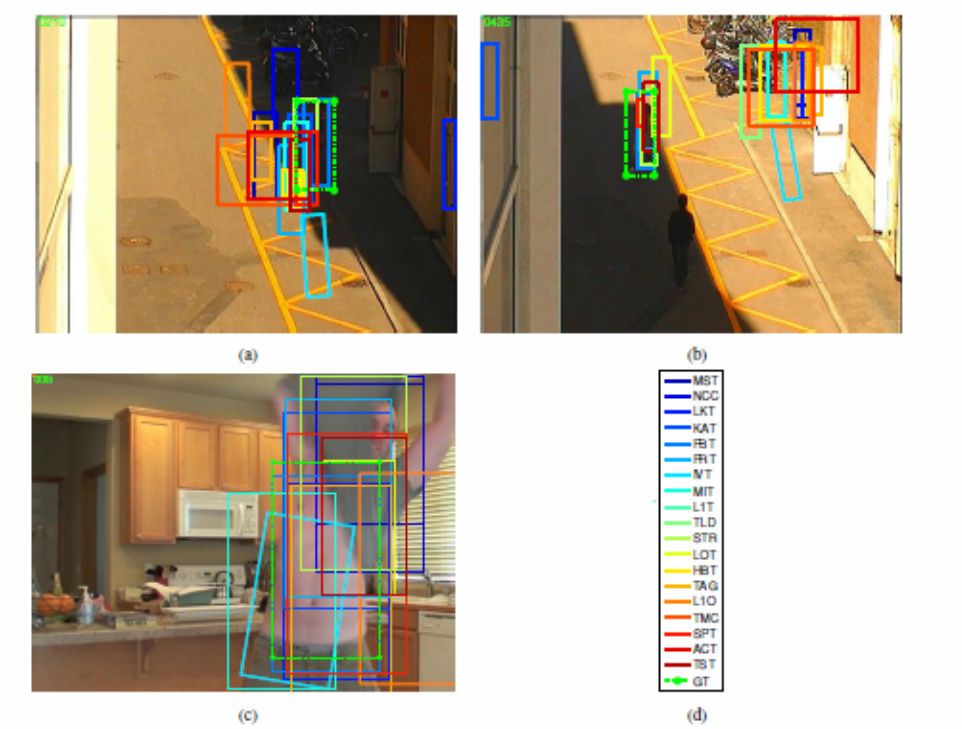
Conclusion: STR (.66) is best by small margin, followed by FBT (.64), TST (.62), TLD (.61), L1O (.60), all different types.

Very hard

On shadows

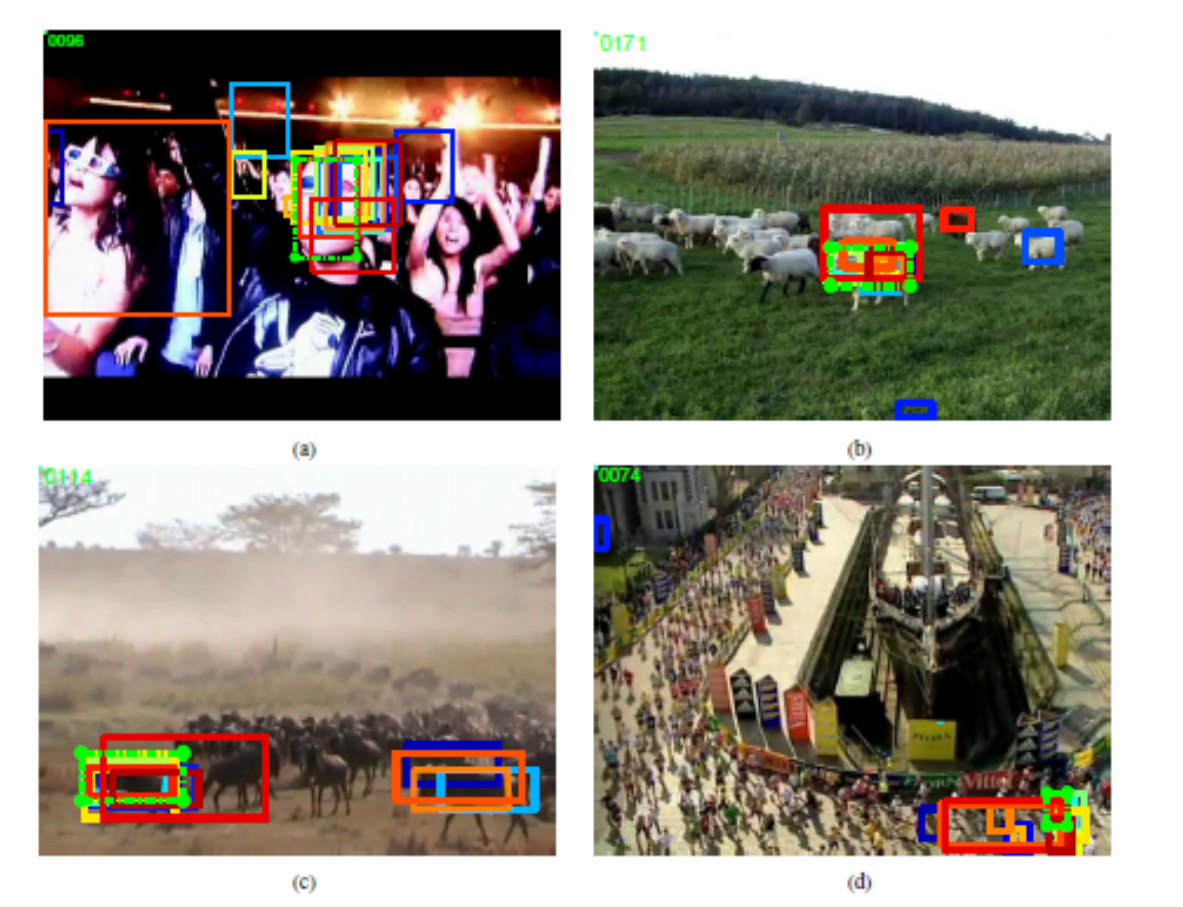
The effect of shadows.

Heavy shadow has an impact almost for all.



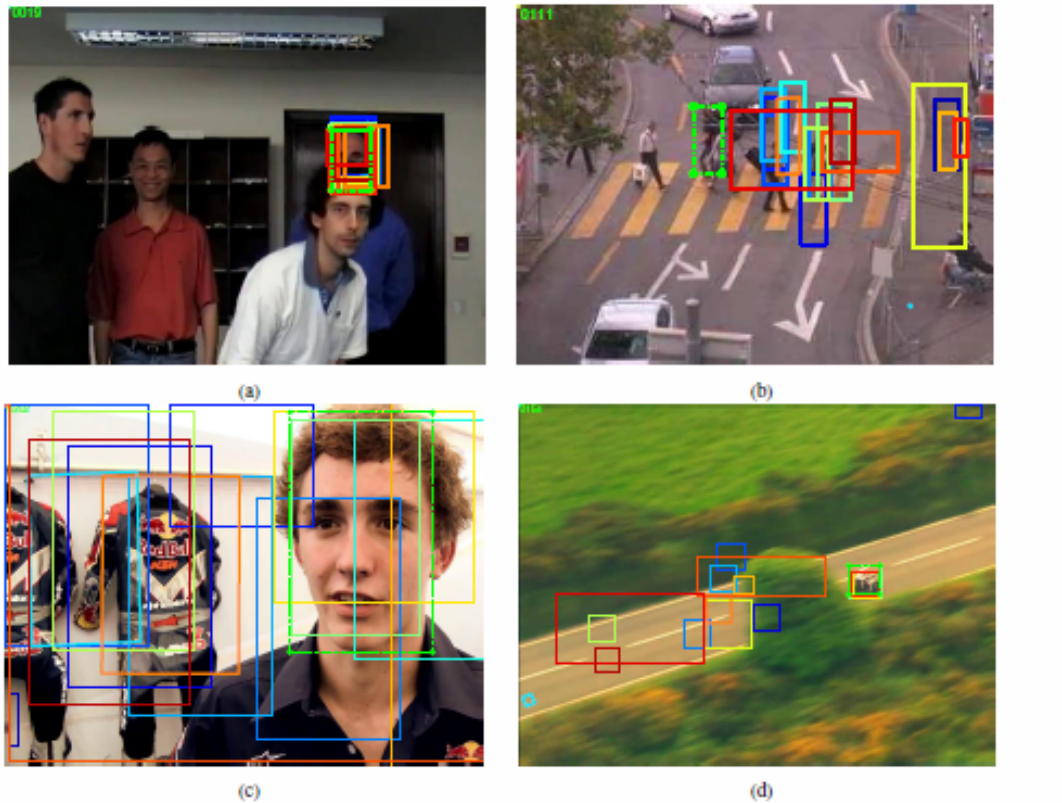
FBT (.73) performs best.

On clutter



Success is better than expected even if very hard.

On occlusion



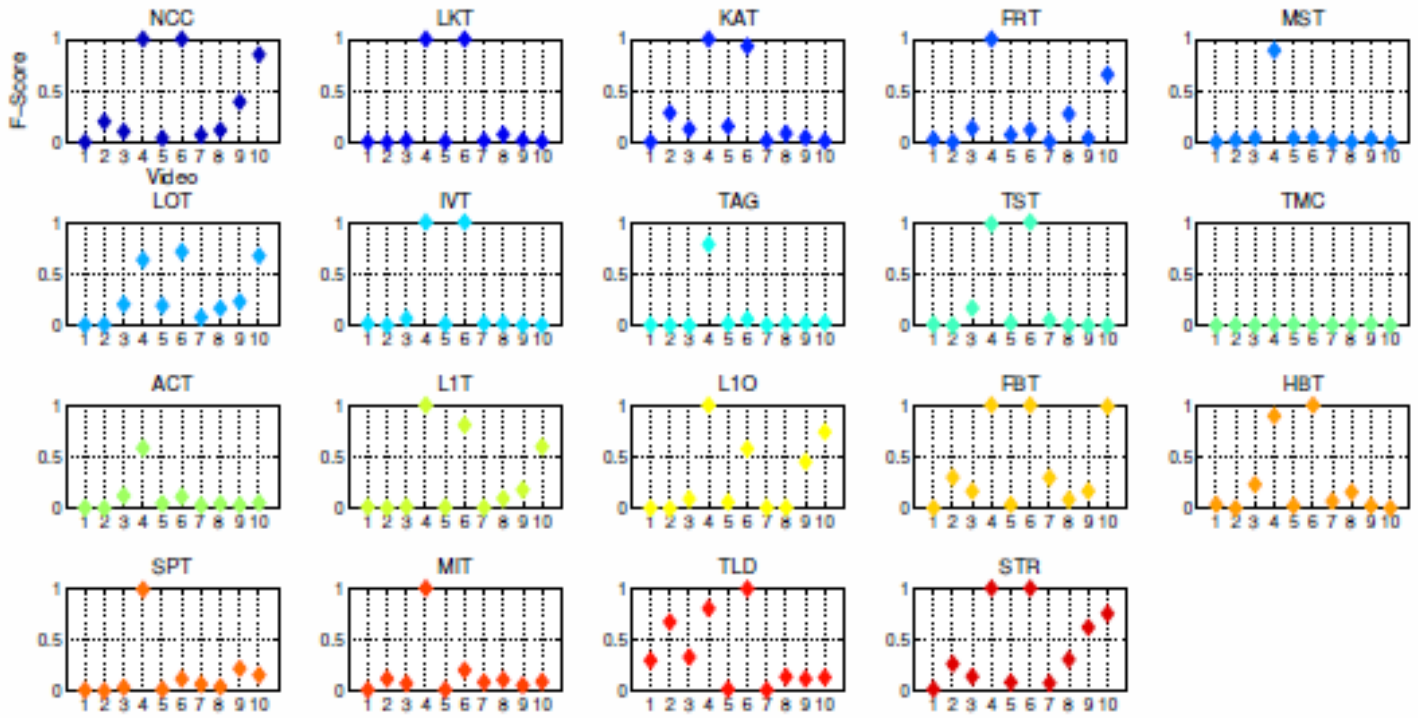
STR, FBT, TST, and TLD are best here (!).

Light occlusion is approximately solved.

Full occlusion is still hard for most.

On long videos

The F-score on ten 1 – 2 minute videos

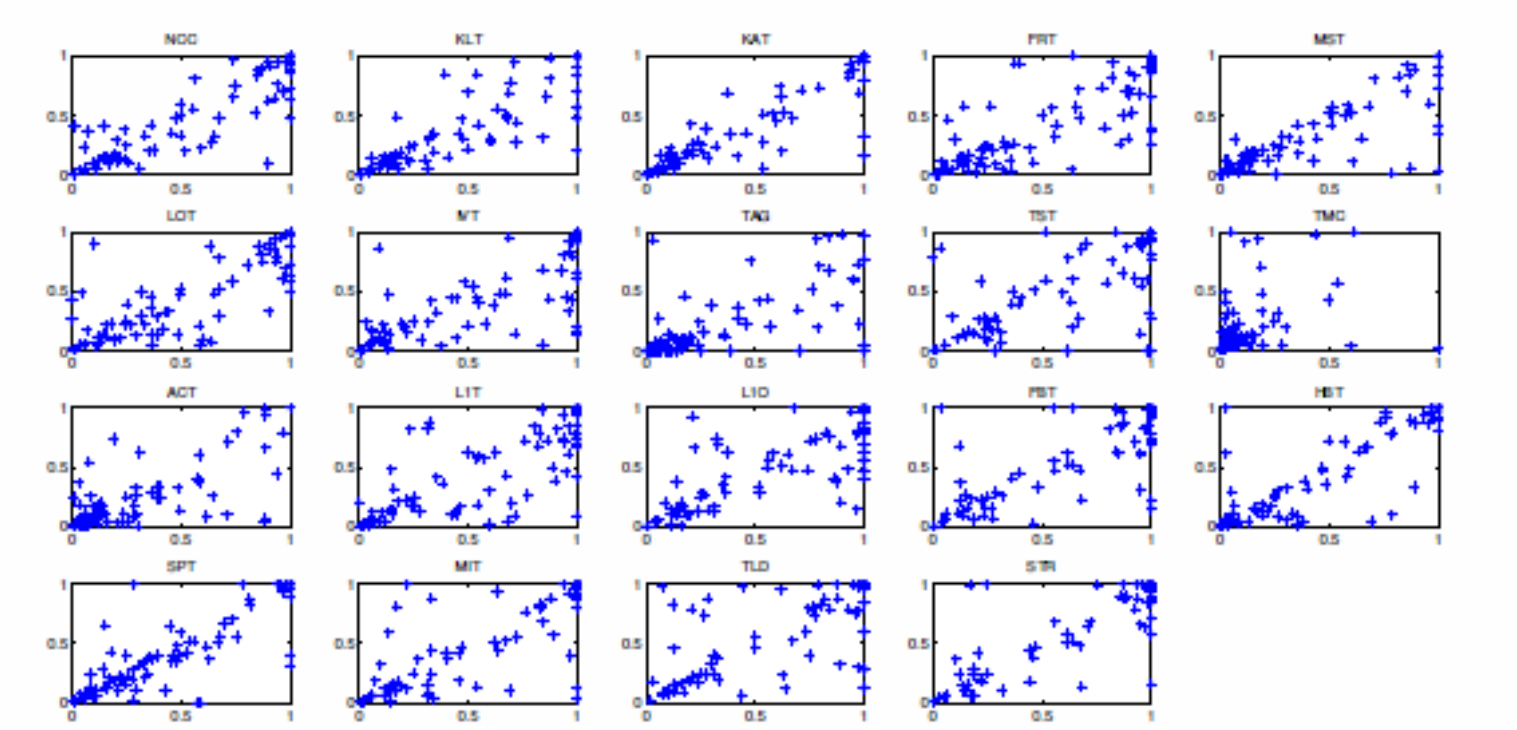


STR, FBT, NCC (no updating!), TLD perform well (!).

TLD excels in sequence 1 which is hard.

On stability of the initial box

F-scores of 20% right shift (y-axis) vs original (x-axis)



Overall loss of .05 %.

STR has a small loss.

Outstanding results by Grubs

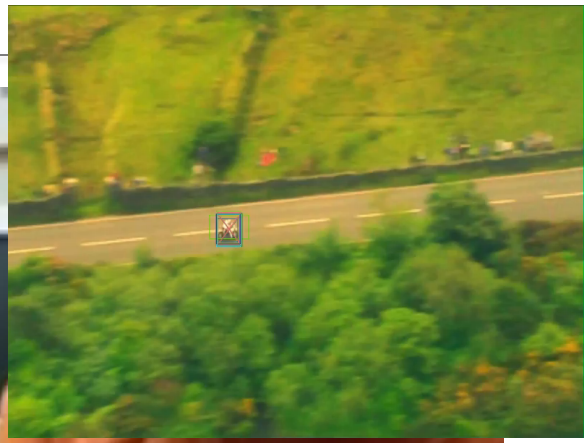
TABLE III: The list of outstanding cases resulted from the Grubbs' outlier test and with $F \geq 0.5$.

Sequence	Tracker	Sequence	Tracker	Sequence	Tracker	Sequence	Tracker
0112	TLD	0411	ACT	1102	TLD	1203	MIT
0115	STR	0510	L1T	1103	HBT	1206	STR
0116	KAT	0512	STR	1104	TLD	1210	TLD
0122	TLD	0601	STR	1107	HBT	1217	TLD
0203	FBT	0611	MST	1112	STR	1218	TLD
0301	L1T	0705	TLD	1116	TLD	1221	TLD
0305	L1T	0901	HBT	1119	TLD	1303	TLD
0312	L1T	0916	STR	1128	TLD	1402	TLD
0314	KAT	0925	STR	1129	FBT	1409	STR
0404	FBT	1020	FBT	1134	FRT		

Many excel in 1 video. (Favorable selection.)

TLD excels in camera motion, occlusion.

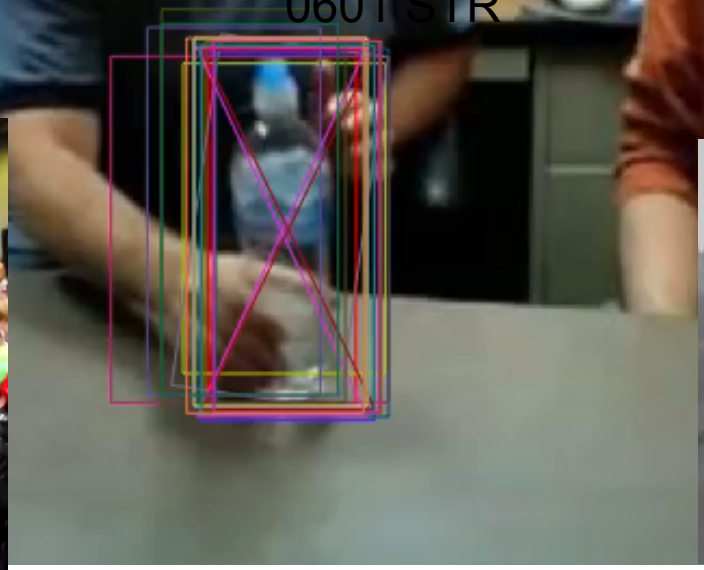
FBT in target appearance, light.



0916 STR

0601 STR

1107 SPT HBT



1129 FBT > FRT

0404 FBT

1402 TLD

The hardness of tracking

Tracking aims to learn a target from the first few pictures; the target and the background may be dynamic in appearance, with unpredicted motion, and in difficult scenes.

Trackers tend to be under-evaluated, they tend to specialize in certain types of conditions.

Most modern trackers have a hard time beating the oldies. We have found no dominant strategy yet, apart from *simplicity*.