P-N LEARNING

Bootstrapping Binary Classifiers by Structural Constraints

OBJECTIVES

Given a single patch and a video sequence, simultaneously learn an object classifier and label all patches in the video as "object" or "background".

CHALLENGES

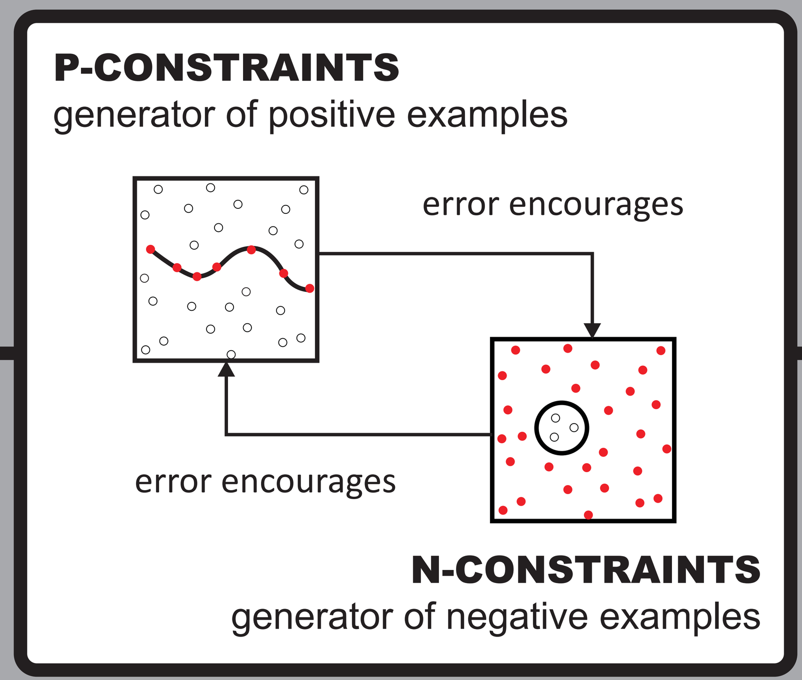
The video is unconstrained, the object might significantly change appearance, temporarily exit the scene. The learning should be **robust** and **real-time**.

MOTIVATION

Efficient learning of object classifiers is applicable in: long-term tracking, surveillance, video analysis, HCI, games, etc.

EXAMPLE

VIDEO



SEMI-SUPERVISED LEARNING

CLASSIFIER

CONTRIBUTION

Semi-supervised algorithm which learns a classifier from a single example and a video. Learning is guided by structural constraints for objects in videos. Application to long-term tracking problem. Simultaneous tracking, learning and detection. Real-time, state-of-the-art performance.

FUTURE WORK

Design of more sophisticated constraints. Generalization to training from multiple examples. Offline processing of sequence.

AVAILABLE ONLINE

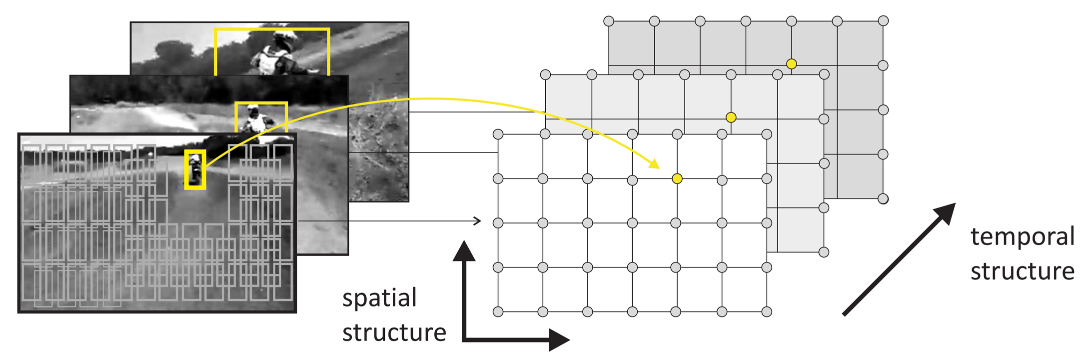
The demo application, sequences with ground truth.
http://info.ee.surrey.ac.uk/Personal/Z.Kalal/

Zdenek **Kalal**, Jiri **Matas**, Krystian **Mikolajczyk**

UNIVERSITY OF SURREY

STRUCTURAL CONSTRAINTS

Restrict labelling of the unlabeled data (patches).



P-CONSTRAINTS define patterns of **positive** examples in unlabeled data.
N-CONSTRAINTS define patterns of **negative** examples in unlabeled data.

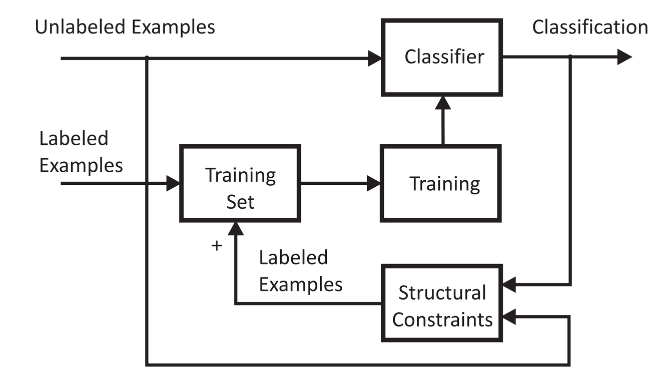
EXAMPLE

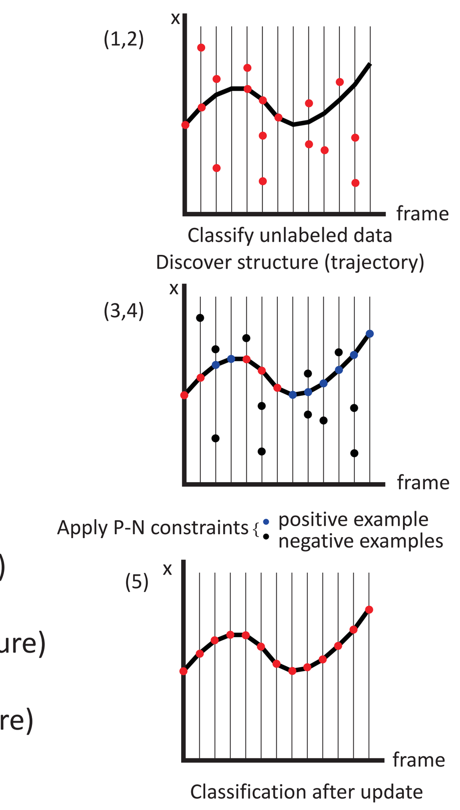
P-constraint (trajectory): object moves on a piece-wise continuous trajectory, patches close to the trajectory are positive.
N-constraint (non-maxima-suppresion): a unique object occupies a single location in a single frame, responses far from the maximally confident patch are negative.

Every constraint introduces errors (e.g. tracker drifts, the maximum response is a false positive). **Errors of P-constraints encourages application of N-constraints and vice versa (negative feedback).** The performance of the classifier grows until long-term stability is achieved.

P-N LEARNING

A semi-supervised algorithm which guides the learning by a pair of structural constraints, i.e. generators of positive and negative examples.





ALGORITHM

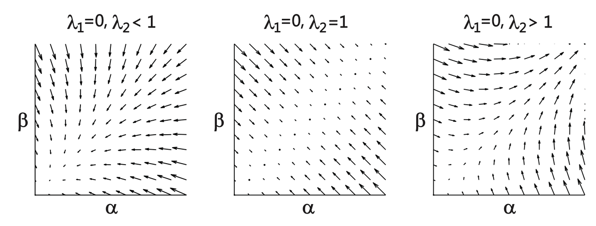
Train a classifier using all labeled data available.
Iterate {
(1) Classify unlabeled data
(2) Discover structure in the data (e.g. track the patch)
(3) Apply **P-constraints**
generates positive data (false negatives w.r.t. structure)
(4) Apply **N-constraints**
generates negative data(false positive w.r.t. structure)
(5) Update classifier
}

CONVERGENCE ANALYSIS

Model of constraints:
P-constraints: Precision (P+), Recall (R+) (ability to identify false negatives)
N-constraints: Precision (P-), Recall (R-) (ability to identify false positives)
Classifier performance: α(k) false negatives, β(k) false positives

$$\begin{bmatrix}\alpha(k+1)\\ \beta(k+1)\end{bmatrix}=\begin{bmatrix}1-R^- & \frac{1-P^+}{P^+}R^+\\ \frac{1-P^-}{P^-}R^- & 1-R^+\end{bmatrix}\times\begin{bmatrix}\alpha(k)\\ \beta(k)\end{bmatrix}$$

$$x(k+1)=\mathrm{M}\cdot x(k)$$

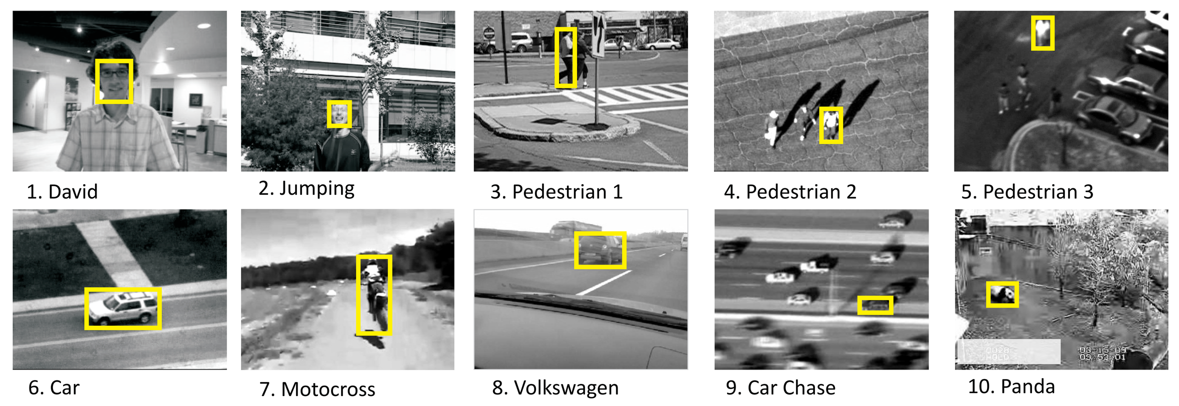


P-N learning improves the classifier, if eigenvalues of M are smaller than one.
Individual constraints can have arbitrarily precision/recall,
stability of the learning is achieved by mutual error compensation.

OBSERVATION

It is difficult to hand design an accurate classifier, but it is easy to design complementary constraints which mutually compensate their own errors.

PERFORMANCE

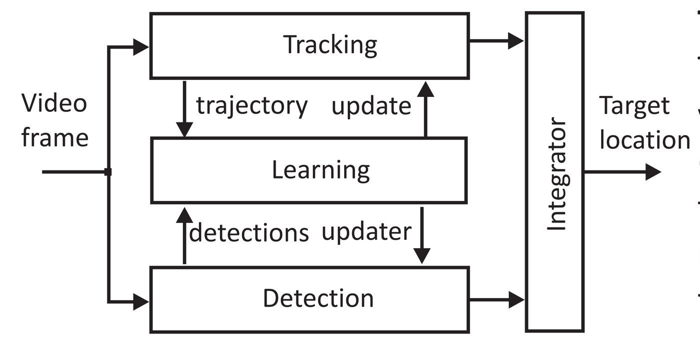


Sequence	Frames	Initial Detector Precision /Recall/F-measure	Final Detector Precision /Recall/F-measure	P-N Tracker Precision /Recall/F-measure	P-constraints P^+ ; R^+	N-constraints P^- ; R^-	Eigenvalues λ_1, λ_2
1. David	761	1.00 /0.01/**0.02**	1.00 /0.32/**0.49**	0.94 /0.94/0.94	1.00 /0.08	0.99 /0.17	0.92 /0.83
2. Jumping	313	1.00 /0.01/**0.02**	0.99 /0.88/**0.93**	0.86 /0.77/0.81	0.86 /0.24	0.98 /0.30	0.70 /0.77
3. Pedestrian1	140	1.00 /0.06/**0.12**	1.00 /0.12/**0.22**	0.22 /0.16/0.18	0.81 /0.04	1.00 /0.04	0.96 /0.96
4. Pedestrian2	338	1.00 /0.02/**0.03**	1.00 /0.34/**0.51**	1.00 /0.95/0.97	1.00 /0.25	1.00 /0.24	0.76 /0.75
5. Pedestrian3	184	1.00 /0.73/**0.84**	0.97 /0.93/**0.95**	1.00 /0.94/0.97	0.98 /0.78	0.98 /0.68	0.32 /0.22
6. Car	945	1.00 /0.04/**0.08**	0.99 /0.82/**0.90**	0.93 /0.83/0.88	1.00 /0.52	1.00 /0.46	0.48 /0.54
7. Motocross	2665	1.00 /0.00/**0.00**	0.92 /0.32/**0.47**	0.86 /0.50/0.63	0.96 /0.19	0.84 /0.08	0.92 /0.81
8. Volkswagen	8576	1.00 /0.00/**0.00**	0.92 /0.75/**0.83**	0.67 /0.79/0.72	0.70 /0.23	0.99 /0.09	0.91 /0.77
9. CarChase	9928	0.36 /0.00/**0.00**	0.90 /0.42/**0.57**	0.81 /0.43/0.56	0.64 /0.19	0.95 /0.22	0.76 /0.83
10. Panda	3000	0.79 /0.01/**0.01**	0.51 /0.16/**0.25**	0.25 /0.24/0.25	0.31 /0.02	0.96 /0.19	0.81 /0.99

SEE LIVE DEMO, DOWNLOAD IT ONLINE.

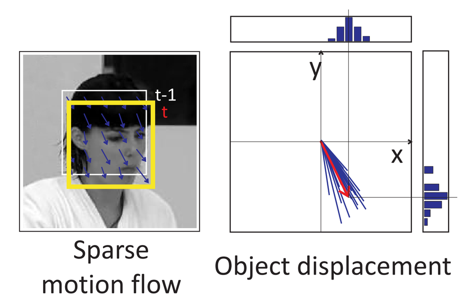
APPLICATION: LONG-TERM TRACKING

Track an object in unconstrained video
(appearance change, object exits and enters the scene, etc.)



Tracking-Learning-Detection
Track an object by a tracker, validate the tracker's trajectory (identification of constraints), train a detector online, re-initalize tracker after its failure.

TRACKER: Median Shift



DETECTOR: randomized forest, 2bitBP features