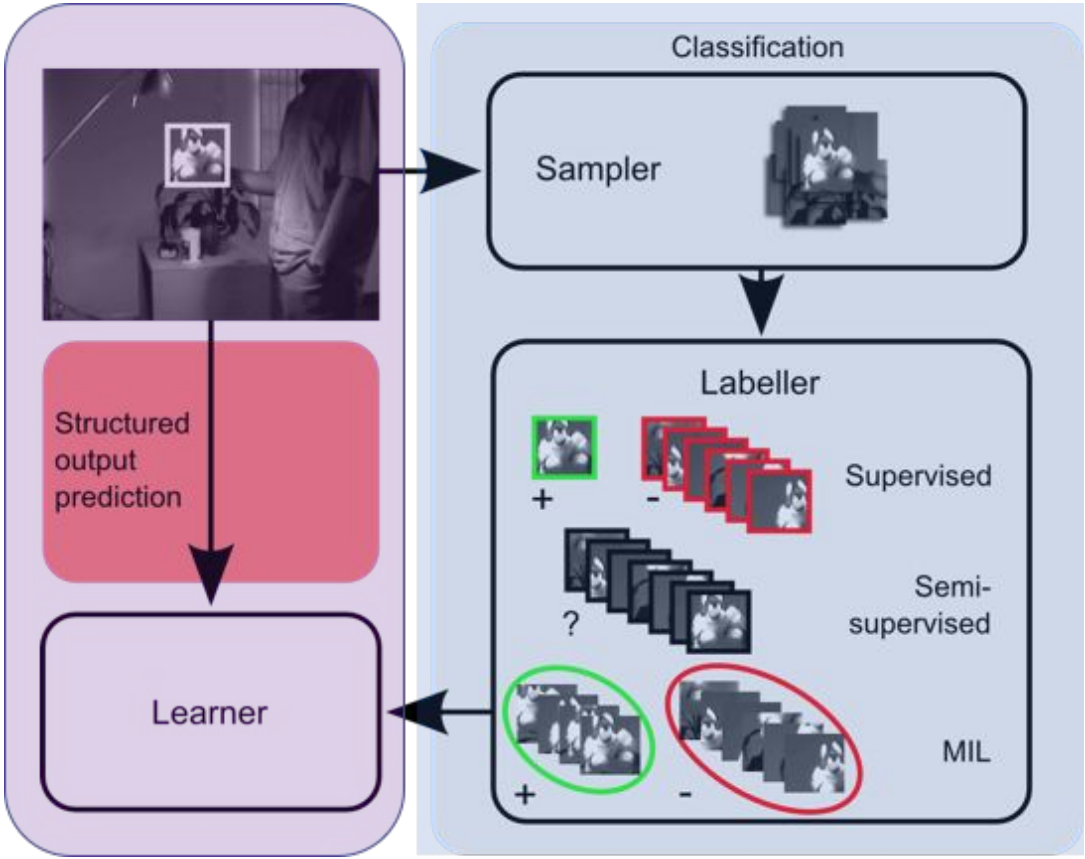
Reimplementation of Struck: Structured Output with Kernels

Mike Liu, Yuhang Ming, and Jing Wang
June 7, 2017

Quick Review of Struck

Proposed Approach



Traditional Approach

Quick Review of Struck

Learner

Algorithm 2 Struck: Structured Output Tracking

Require: $\mathbf{f}_t, \mathbf{p}_{t-1}, \mathcal{S}_{t-1}$

Estimate change in object location

$\mathbf{y}_t = \arg\max_{\mathbf{y} \in \mathcal{Y}} F(\mathbf{x}_t^{\mathbf{p}_{t-1}}, \mathbf{y})$

$\mathbf{p}_t = \mathbf{p}_{t-1} \circ \mathbf{y}_t$

Update discriminant function

$(i, \mathbf{y}_+, \mathbf{y}_-) \leftarrow \text{PROCESSNEW}(\mathbf{x}_t^{\mathbf{p}_t}, \mathbf{y}^0)$

$\text{SMOSTEP}(i, \mathbf{y}_+, \mathbf{y}_-)$

$\text{BUDGETMAINTENANCE}()$

for $j = 1$ to n_R **do**

  $(i, \mathbf{y}_+, \mathbf{y}_-) \leftarrow \text{PROCESSOLD}()$

  $\text{SMOSTEP}(i, \mathbf{y}_+, \mathbf{y}_-)$

  $\text{BUDGETMAINTENANCE}()$

  for $k = 1$ to n_O **do**

    $(i, \mathbf{y}_+, \mathbf{y}_-) \leftarrow \text{OPTIMIZE}()$

    $\text{SMOSTEP}(i, \mathbf{y}_+, \mathbf{y}_-)$

  end for

end for

return $\mathbf{p}_t, \mathcal{S}_t$

What We Did

- Reimplement the Struck model with python and python-opencv.
- Focus on performance evaluation with Haar feature & Gaussian Kernel
- Perform some experiments on the benchmark dataset.
- Analyze the strength and the limits of the Struck model.
- Possible improvements of the Struck model.

Dataset: http://cvlab.hanyang.ac.kr/tracker_benchmark/index.html

Result - Evaluation

- Evaluate the performance using Pascal VOC overlap criterion

$$a_o = \frac{area(B_p \cap B_{gt})}{area(B_p \cup B_{gt})}$$

B_p : *predicted bounding box*; B_{gt}: *ground truth bounding box*

Assume $area(B_p) = area(B_{gt})$, *then only when*:

$$area(B_p) \cap area(B_{gt}) = \frac{2}{3} area(B_{gt}),$$

$$Then\ area(B_p) \cup area(B_{gt}) = \frac{4}{3} area(B\_gt)$$

$$a_o = \frac{area(B_p \cap B_{gt})}{area(B_p \cup B_{gt})} = \frac{\frac{2}{3} area(B_{gt})}{\frac{4}{3} area(B_{gt})} = \frac{1}{2}$$

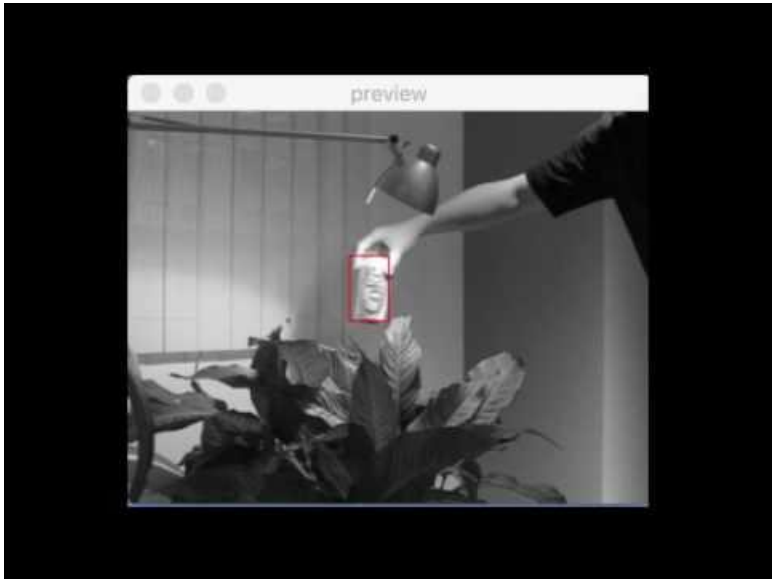
Result - Experiments on the dataset

- Comparison with the original work and the other traditional model

Sequence	PyStruck	Struck	MIForest	OMCLP	MIL
Coke	**0.56**	0.57	0.35	0.24	0.33
Girl	**0.72**	0.80	0.71	0.64	0.53
Face 1	0.78	0.86	0.77	**0.80**	0.60
Face 2	**0.79**	0.86	0.77	0.78	0.68
Sylvester	0.61	0.68	0.59	**0.67**	0.60

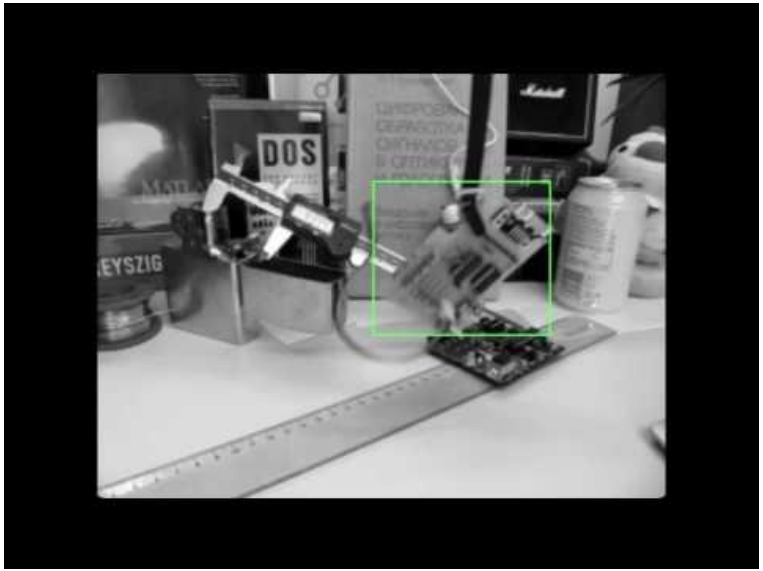
Result - Strength of the Struck Model

Face1 - score: 0.76



Coke - score: 0.56

Result - Limits of the Struck Model



Box - score: 0.61

Tiger2 - score: 0.329

Result - Speed

- Reimplement the Struck model with python and python-opencv:
  - Implementation in python is much slower than that in C++
    - C++: 13.2 FPS vs Python: 0.055 FPS (30 Search Radius).
    - Reason: Extracting Haar features for 2800 samples.
  - Possible improvements in speed:
    - Use external library to calculate the matrix.

Possible Improvements

- Accuracy:
  - Expand search space for scale and rotation variance.
  - Investigate in different kinds of feature combinations.
- Speed:
  - Change search area during tracking guided by optical flow.