Object Detection in Low Resolution Overhead Imagery

CASIS Presentation

May 13, 2015

Kofi Boakye* and Paul Kidwell

Lawrence Livermore National Laboratory

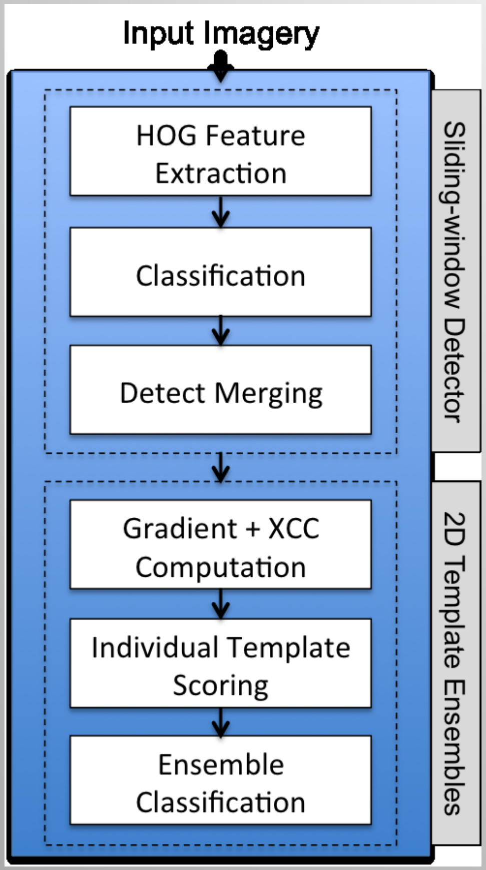
LLNL-PRES-670528

This work was performed under the auspices of the U.S. Department of Energy by Lawrence Livermore National Laboratory under Contract DE-AC52-07NA27344. Lawrence Livermore National Security, LLC

Motivation

- Objective
  - Reliable detection of rigid objects in relatively low resolution, e.g. ~1m GSD, overhead imagery
- Applications
  - Civil transportation, military reconnaissance, environmental monitoring
- Challenges
  - Potentially few training examples → Limits statistical learning
  - Lighting, view angle, environmental conditions → Possibly hard to reliably capture in 3D models
- Solution
  - Hybrid approach combining statistical and physical models

System Overview



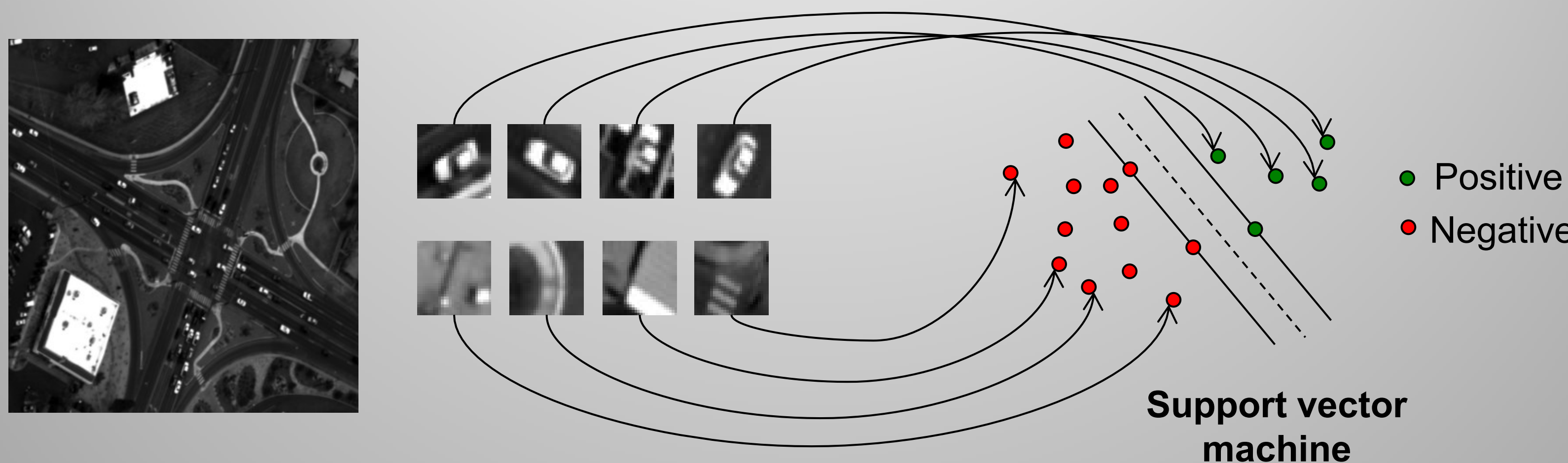
- Two-stage detection algorithm
  1. Fast / high recall: Sliding-window HOG/SVM detector
  2. High accuracy: Ensembles of 2D templates

- Desired properties
  - Requires few training data
  - Reasonable computational cost
  - Not too complicated
  - Empirical
  - Ability to differentiate similar objects (fine-grained)
  - Reliable under varied conditions

- Assumptions
  - Rigid object
  - Known and constant spatial scale (ortho-rectified)
  - NOT assumed: illumination and appearance/texture

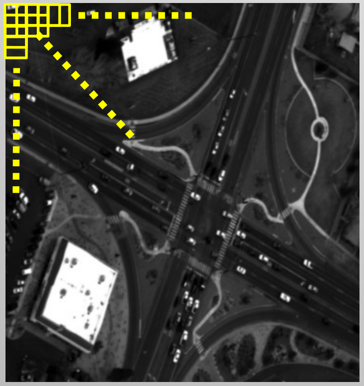
Sliding Window Detector: Training

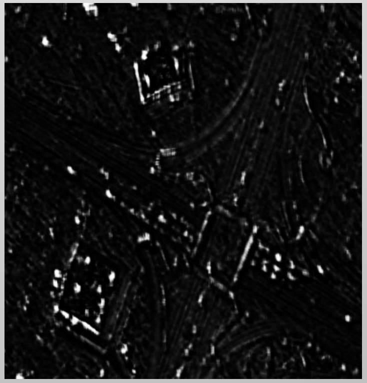
Input imagery → **Extract positive/negative examples** → **Encode examples into feature vectors** → **Learn binary classifier** →

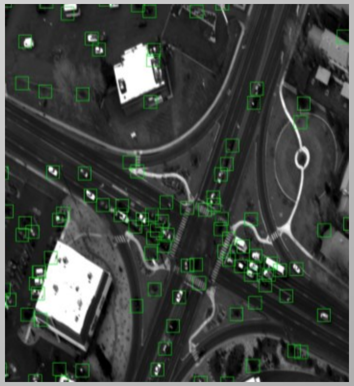


Sliding Window Detector: Detection

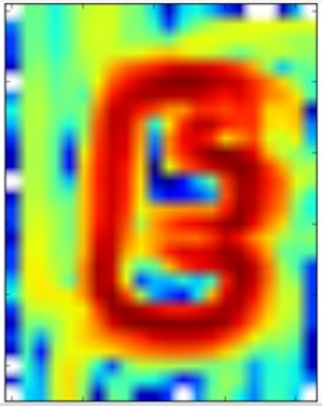
Input imagery → **Scan image at regularly spaced interval** → **Run classifier to obtain object/non-object decisions** → **Merge multiple detections** → Bounding box object detections



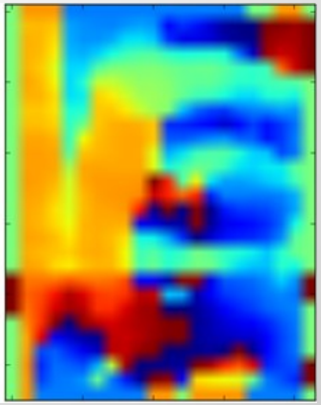




2D Template Ensemble (2TE): Image Preprocessing and Scores



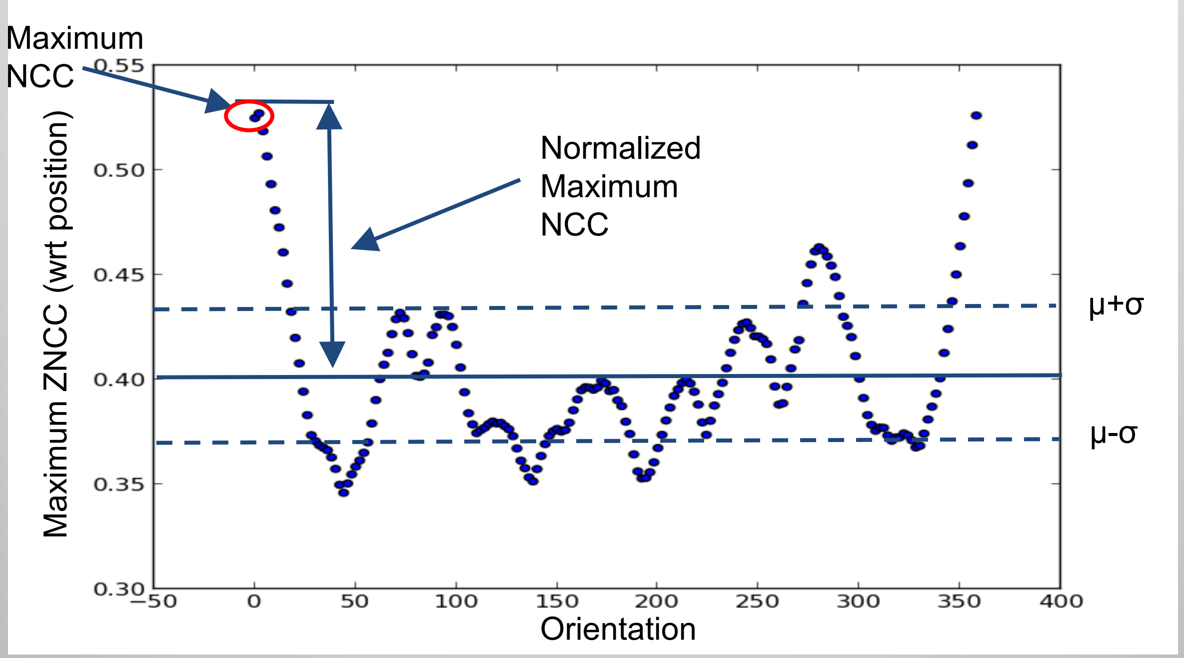
Magnitude



Orientation

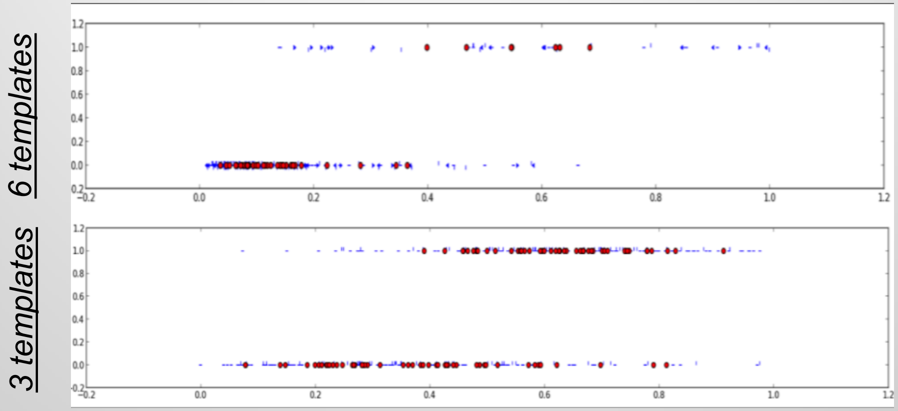
- "While a simple matched filter by itself lacks the necessary invariance properties for object detection in aerial imagery, with suitable transformations, matching criteria, and dimensionality reduction, invariance to rotations, translations, and more general variations may be achieved" - Brunelli 2009
- Computation of zero-mean normalized cross-correlation
  - With respect to gradient, Sobel filter
  - Independently for each dimension
  - Weighted combination based on magnitudes

2TE: Distributions of template matching scores



- In practice, for every template-image pair many values of ZNCC are computed
  - ZNCC is a function of position and orientation (3D distribution)
- Identify representative summary statistics!
  - Maximum NCC
  - Normalized (for a template-image pair) Maximum NCC

2TE: Template models and Ensemble object recognition



- Template ensemble classifier
  - Individual logistic regression by template

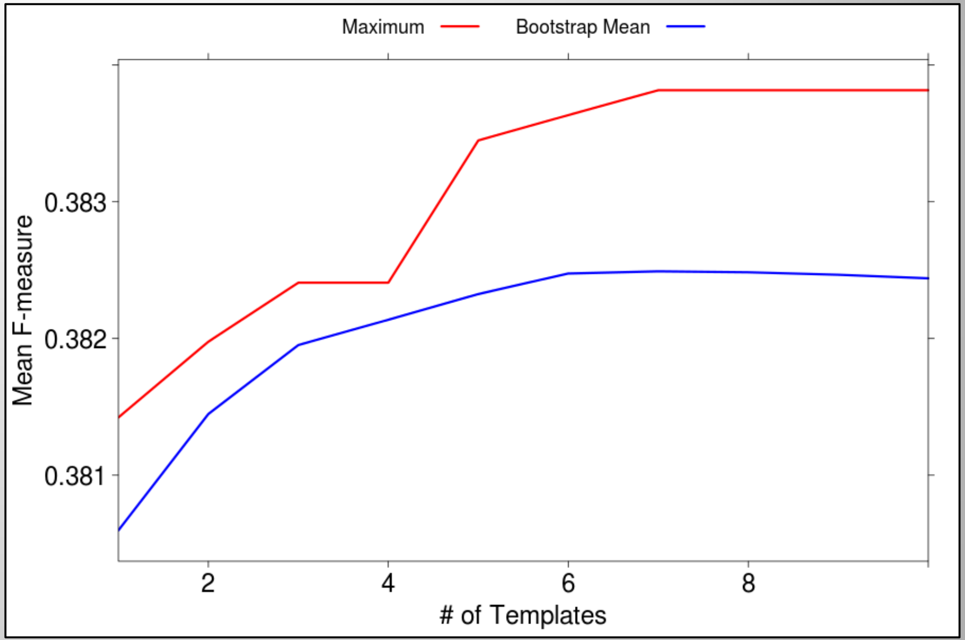
$$\log\left(\frac{p_{TI}}{1-p_{TI}}\right) = \beta_0 + \beta_1 s_m + \beta_2 s_n$$

    - Class balance is dependent upon stage 1
  - Ensemble rule

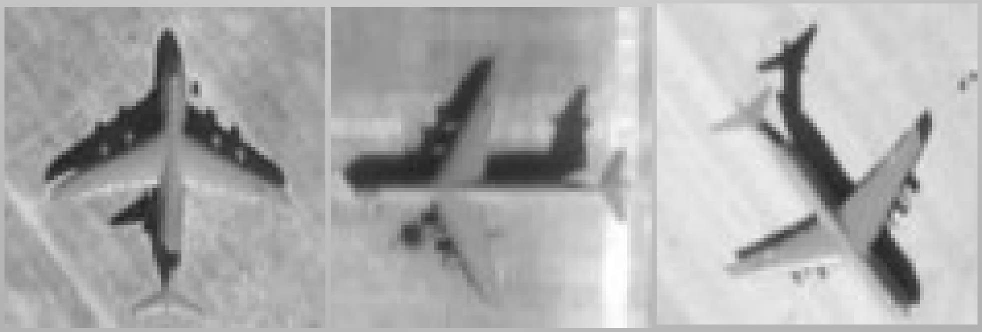
$$p_I = \max_T p_{TI}$$

2TE: Greedy Template Selection

Input: Candidate templates, C; stopping threshold, t;
Cross-validation data
Variables: Predictor templates, $P = \emptyset$;
mean f-measure, $\bar{f}_{old} = 0$
Algorithm:
Step 0: Compute the mean f-measure over all thresholds for each combination of P and $c \in C$, setting $\bar{f}_{new}$ to the maximum.
Step 1: If $\bar{f}_{new} - \bar{f}_{old} > t$ and $C \neq \emptyset$ then $P = P \cup c$, $C = C \setminus c$, $\bar{f}_{old} = \bar{f}_{new}$, and repeat Step 0.
Output: P



Example Templates

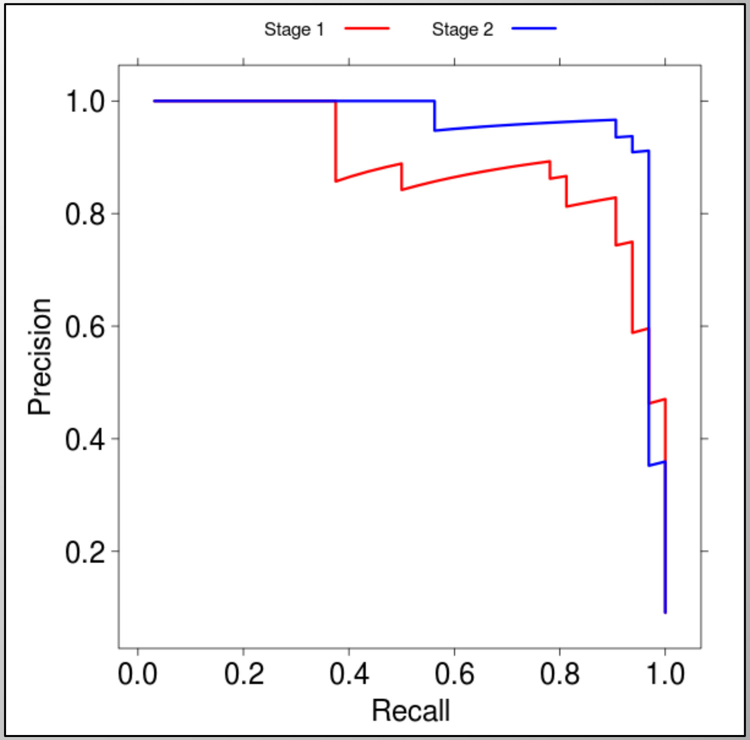


Evaluation Dataset

- Three select airfields with historical data from Google Earth
  - 25 Images spanning 1725m x 1120m (GSD=1.15 m/pix)
  - Randomly partitioned into training/validation/test (60/20/20 split)

Results: Component Performance Analysis

- Compare performance of finalized Stage 1 and Stage 2 components
  - Positive and negative examples extracted and scored independently
- Sliding window detector does well in high precision regime
- 2D template ensemble outperforms sliding window detector overall
  - Tradeoff: higher computational cost



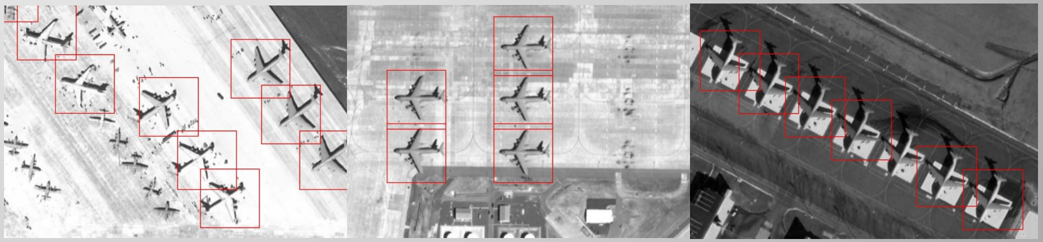
Stage 1 detector helps better balance tradeoff

Results: End-to-End Performance Analysis

- Two-stage approach performs effectively
  - Stage 1 achieves high recall, while stage 2 raises overall performance
- Greedy selection comparable to larger ensembles
  - ~80% fewer templates
- Related comparison: Gao et al. ’13
  - Comparable results with much higher resolution imagery (0.17 m/pix vs. 1.15 m/pix), although our test aircraft, C-5, were large
  - General purpose aircraft detector vs specific type of aircraft

Stage 1

Stage 1 + Stage 2



Algorithm	Precision	Recall	F-Measure
Stage 1	0.185	0.986	0.311
Stages 1+2 (greedy8)	0.809	0.809	0.809
Stages 1+2 (manual20)	0.979	0.701	0.817
Stages 1+2 (all50)	0.946	0.779	0.855

Results: End-to-End Performance Analysis

- Per-image performance results
  - Obtained using ‘greedy8’ system
- Detection performance largely consistent across images for common threshold
- Performance change small compared to oracle threshold (optimal threshold for each image)

Test Image	Oracle Threshold			Common Threshold		
	Precision	Recall	F-Measure	Precision	Recall	F-Measure
‘003’	0.762	0.941	0.842	0.762	0.941	0.842
‘004’	0.842	0.842	0.842	0.842	0.842	0.842
‘006’	0.857	0.667	0.750	0.857	0.667	0.750
‘008’	0.625	1.0	0.769	0.625	0.733	0.709
‘019’	1.0	1.0	1.0	1.0	0.778	0.875

Additional indicator of system robustness

Conclusion

- Presented: Automated approach to object detection in low-res imagery with limited training data
  - Two-stage approach can yield good vehicle detection performance with limited variability
  - Exploited distribution of correlation scores
  - Template selection procedure maintains good performance while reducing computational burden
- Next steps
  - Joint exploration of tuning parameter space
  - Improve sliding window detector (e.g., boosting, alternative SVM kernels, alternative image classification approaches)
  - Explore procedures for better handling occlusions
  - Extend analysis to varied datasets and different objects