Timely Object Recognition

Sergey Karayev[1], Tobias Baumgartner[2], Mario Fritz[3], Trevor Darrell[1]

[1]UC Berkeley, [2]RWTH Aachen, [3]MPI Informatics

1. Motivation

Problem

- Detect objects of many classes in an image.
- Trained detectors and classifiers already exist, but there is **not enough time to run all detectors**.

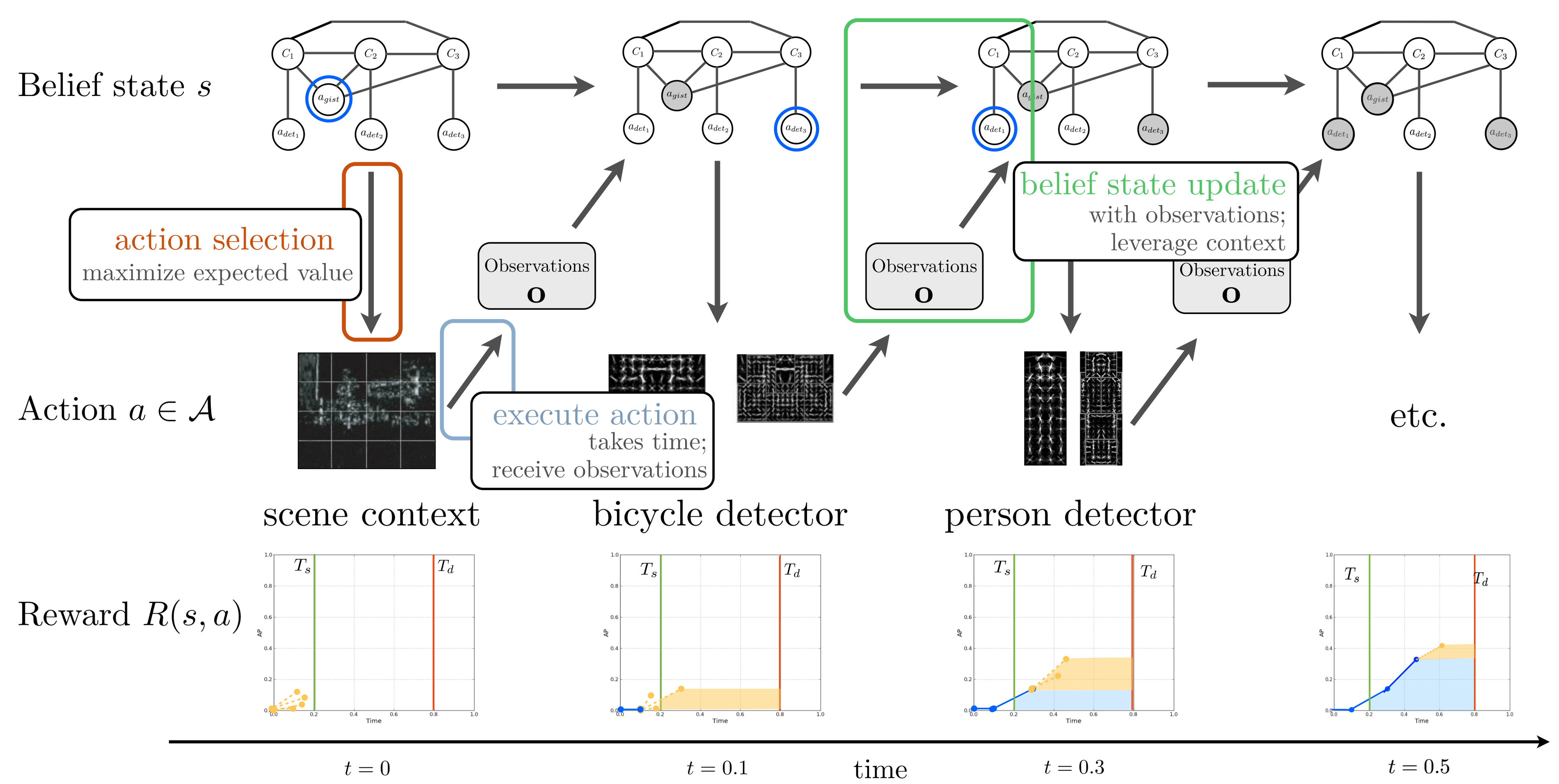
Approach

- Formulate **timeliness** evaluation of detection performance vs. time.
- Treat detectors and classifiers as "black boxes"; use reinforcement learning to find a **dynamic** policy for deploying them.

Results

- Learn to take actions that do not provide immediate reward.
- Wrapping per-class detectors in our system and setting a deadline **increases the multi-class Average Precision at deadline and before.**

2. Sequential Multi-class Detection



- Action $a \in \mathcal{A}$ can run detector of object class k on whole image, or classify the scene.
- The selected action returns observations $\mathbf{o}$: a list of detections, or an evaluated feature.
- Define the policy as taking the untaken action with maximum value, linearly approximated:

$$\pi(s) = \arg\max_{a \in \mathcal{A} \setminus \mathcal{O}} Q(s, a) = \theta_\pi^\top \phi(s, a)$$

- Learn an accurate approximation to the expected rewards to the end of the episode:

$$Q^\pi(s^j, a^j) = \mathbb{E}[R \mid s^j, a^j, \pi]$$

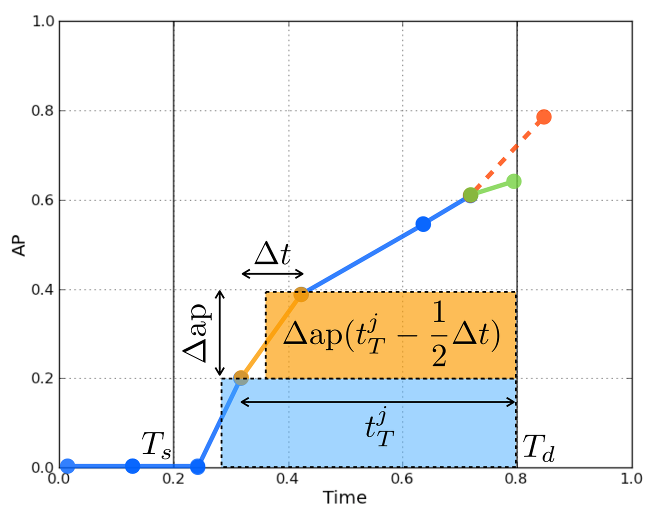
where $R = \sum_{i=j}^{J} \gamma^{i-j} R(s^i, a^i)$ and J is the index of the last action before deadline time T_d.

3. Implementation

AP vs Time and the Reward Function

- The final evaluation is the normalized area under the AP vs. Time curve between T_s and T_d.
- Because this is additive per action, we define

$$R(s^j, a) = \Delta\text{ap}(t_T^j - \frac{1}{2}\Delta t)$$



Reward Discount γ

- $\gamma = 0$ induces a completely greedy policy; at $\gamma = 1$ is full lookahead.
- $\gamma = 0.4$ performs best in cross-validation on the final evaluation.

Learn $\pi(s)$ with Monte Carlo policy iteration.

- Gather samples under current π by running detection episodes on several thousand images.
- Update policy-defining weights θ with L_1 regularization.
- Gradually decrease ϵ-greediness of policy.

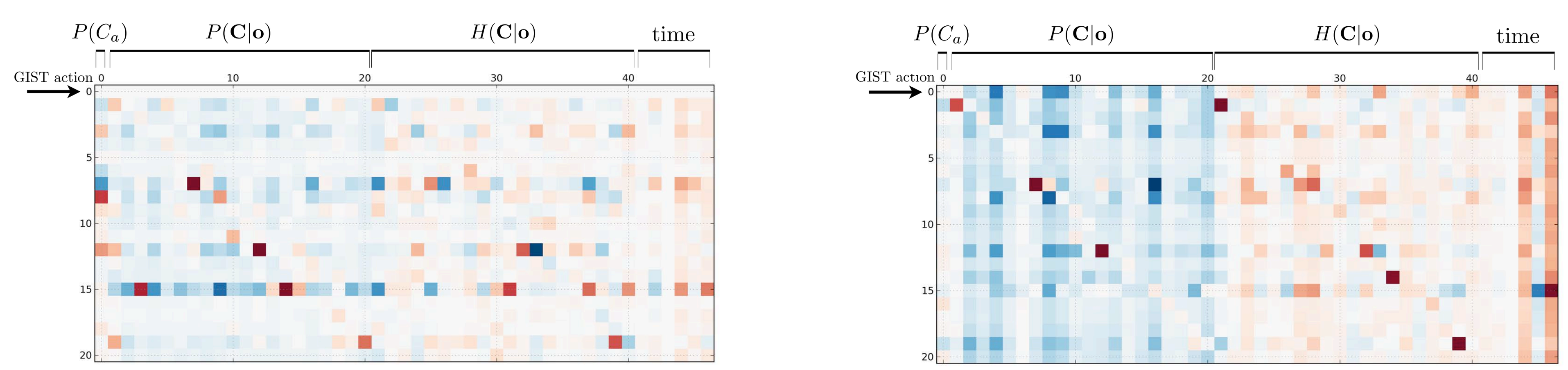
The belief state posterior over class presences $P(\mathbf{C}|\mathbf{o})$ is updated with observations $\mathbf{o}$.

- *Direct* method assumes independence between classes, and simply replaces the posterior of the class(es) corresponding to the action.
- *MRF* method sets evidence node and runs loopy BP in a fully-connected MRF model learned with L_1-regularization.

4. Feature Representation and Policy Weights

The featurization of the belief state and considered action $\phi(s, a)$ reflects

- current probabilities of presence for all classes, and associated entropies;
- current time to T_s and to T_d, and the expected time of the action.



Greedy **Reinforcement Learning**

Learning with a higher γ results in policies reliant on the global scene feature.

5. Evaluation

We evaluated on the PASCAL VOC 2007 detection task, using

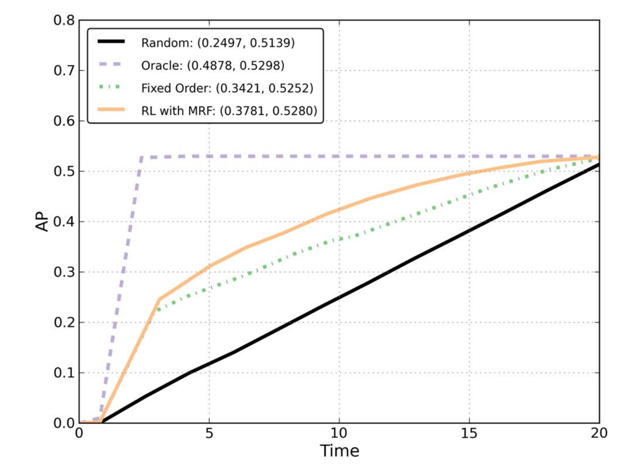
- 20 Deformable Part Model detectors (one per class)
- a scene context action based on the GIST feature.

Performance vs. Time

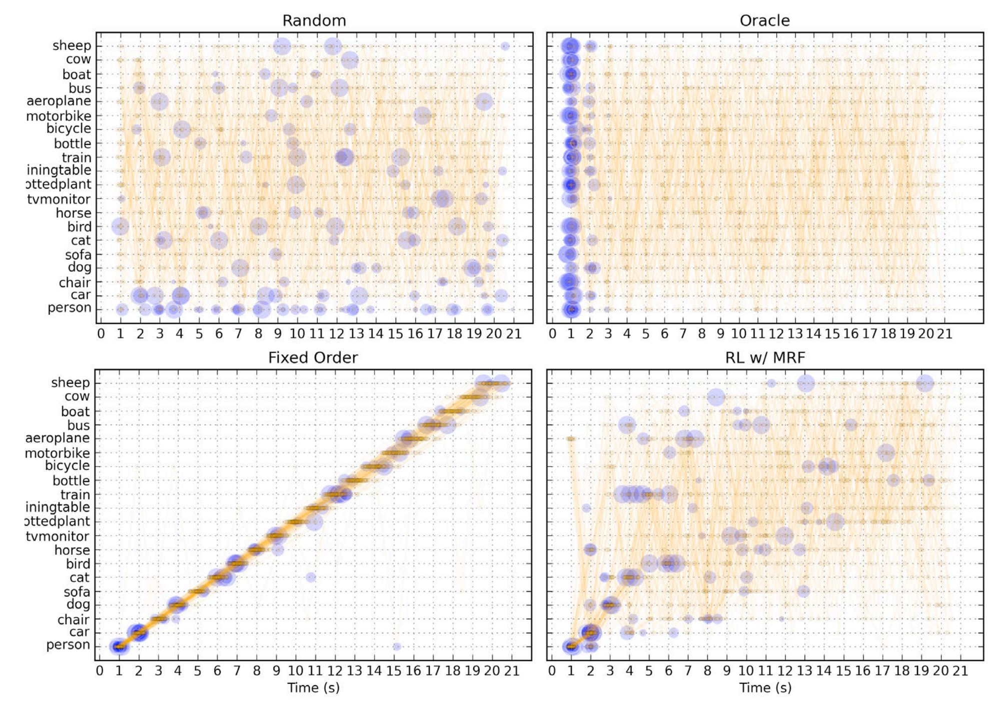
Our method is compared to a random, optimal static, and oracle orderings, evaluated at different settings of start and deadline times.

Bounds	Random	Fixed Order	RL	RL w/ GIST	Oracle
(0,20)	0.250	0.342	0.378	**0.382**	0.488
(0,10)	0.119	0.240	0.266	**0.267**	0.464
(5,15)	0.257	0.362	0.418	**0.420**	0.530

Areas under the AP vs. Time curve.



Policy Trajectories



Action selection traces are plotted over many episodes; the size of the circles correspond to the increase in AP obtained by the action. Our policy selects actions dynamically to maximize the rewards obtained early on.

6. Conclusion

If execution is stopped with only half of the detectors deployed, we obtain at least 66% better AP than a random ordering, and 14% better than an intelligent baseline. On the timeliness measure, we obtain at least 11% better performance.

Our method is easily extensible.
Code is available at `http://sergeykarayev.com/work/timely/`.

Next Steps

- Define actions on regions of the image.
- Account for feature computation cost with multiple detector types.