Anytime Recognition of Objects and Scenes

Sergey Karayev · Mario Fritz · Trevor Darrell
UC Berkeley MPI for Informatics UC Berkeley

http://sergeykarayev.com/recognition-on-a-budget/

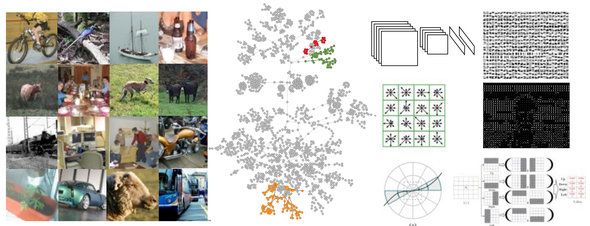
Motivation

- Human perception is Anytime and progressive.

"bright object" "possibly outdoors" "on the coast, at least three birds"

Time (ms) 40 60 500

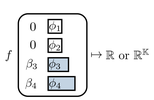
- We deal with varied images, classes, and recognition actions.



- Even when budget does not let us compute everything, want to provide best performance.

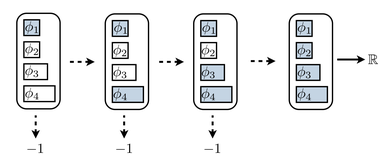
Related Work

Static feature selection



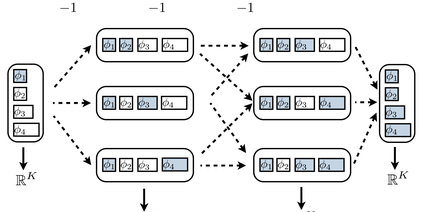
Cascade

Viola & Jones CVPR 2001
Chen et al. ICML 2012



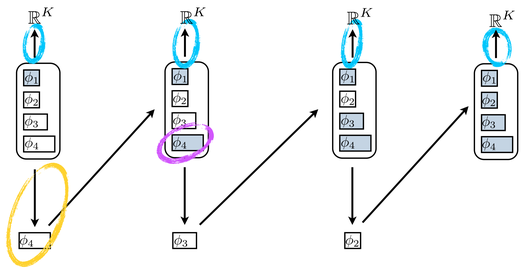
Dynamic feature selection

Gao & Koller NIPS 2011
Benbouzid et al. NIPS 2012
Karayev et al. NIPS 2012



Model

- We model the problem of state traversal as a Markov Decision Process.
  - We learn to **select action** of maximum expected value from every state.
  - State is **updated** with the result of the selected action.



- We **train classifier** on subsets of features, to give answer at any time.

Action selection

Policy:

$$\pi(s) = \arg\max_{a_i \in \mathcal{A} \setminus \mathcal{O}} Q(s, a_i)$$

Action-Value function:

$$Q^\pi(s, a_i) = \mathbb{E}_{s'}[R(s', a_i) + \gamma Q^\pi(s', \pi(s'))]$$

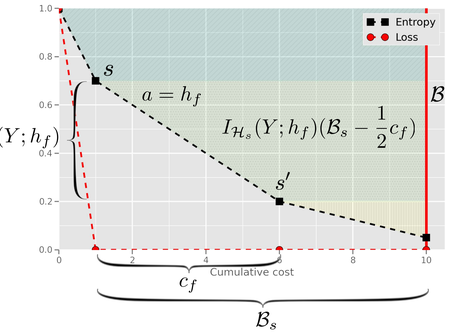
reward definition

Linear weights:

$$Q^\pi(s, a_i) = \theta_\pi^\top \phi(s, a_i)$$

learning the policy

Reward definition



Training algorithm

Input: $\mathcal{D} = \{x_n, y_n\}_{n=1}^N$; $\mathcal{L}_\mathcal{B}$
Result: Trained π, g

$\pi_0 \leftarrow$ random;
for $i \leftarrow 1$ **to** *max_iterations* **do**
  States, Actions, Costs, Labels $\leftarrow$ `GatherSamples` ($\mathcal{D}$, π_{i-1});
  $g_i \leftarrow$ `UpdateClassifier` (*States, Labels*);
  Rewards $\leftarrow$ `ComputeRewards` (*States, Costs, Labels,* $g_i, \mathcal{L}_\mathcal{B}, \gamma$);
  $\pi_i \leftarrow$ `UpdatePolicy` (*States, Actions, Rewards*);
end

Evaluation setup

We evaluate these baseline policies:

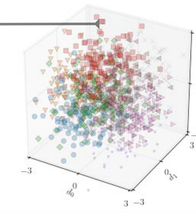
- **Static, greedy**: does not observe feature values, selects actions greedily.
- **Static, non-myopic**: does not observe feature values, but plans ahead (using MDP with $\gamma = 1$).
- **Dynamic, greedy**: observes feature values, but selects actions greedily.

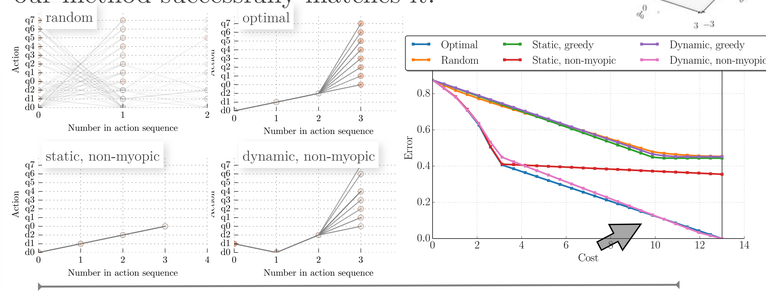
Our method is the **Dynamic, non-myopic** policy: observes feature values, and has plans ahead.

Evaluation

Synthetic Example

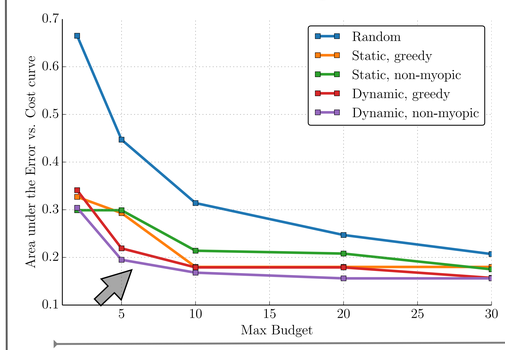
- On a task where the optimal policy is known to be dynamic and non-myopic, we show that our method successfully matches it.

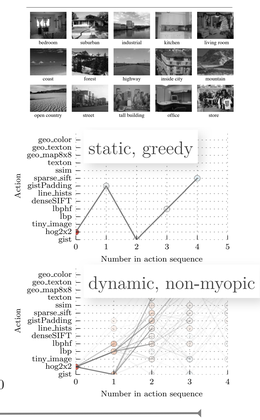




Scenes-15

- Actions are feature computations.





ImageNet

- Actions are class evaluations.
- Progressively enhance specificity with increasing budget.