Bootstrap Learning for Place Recognition

Benjamin Kuipers
Patrick Beeson

University of Texas at Austin

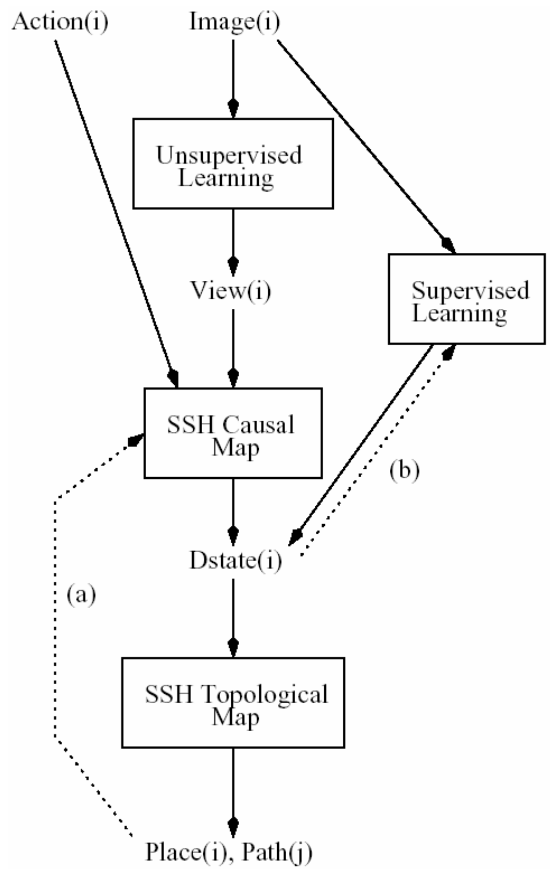
Place Recognition

- Identify current position and orientation
  - from sensory image
  - “global localization”
- Problem 1: ***Perceptual aliasing***.
  - Different places look the same.
- Problem 2: ***Image variability***.
  - The same place looks different.
- Rich sensors make variability more important.

Solution: Bootstrap Learning

- *Use an **unsupervised learning** method*
  - Cluster sensory images into views
- *to prepare for a **deductive** method*
  - Build a causal/topological map
- *that supports a **supervised learning** method*
  - Nearest neighbor
- *that achieves **high performance**.*
  - Two real-world robot experiments.

Bootstrap Learning Diagram



Only Learn *Distinctive* Places

- A *distinctive state* is the isolated fixed-point of a hill-climbing control law.
  - distinctive place and orientation.

- A causal link <*x*,*a*,*x'*> asserts:
  - *x* and *x'* are distinctive states (dstates),
  - Action *a* consists of trajectory-following then hill-climbing, leading *reliably* from *x* to *x'*.

- Part of the Spatial Semantic Hierarchy (SSH).

Contrast with Occupancy Grids

Occupancy grids

- Single global frame of reference
- Designed for range-sensors.
- Problematic to define $p(o|x,m)$ for image o.

Topological maps

- Multiple local frames of reference
- No assumption about sensors.
- Reasonable definition of $p(v|x,m)$, clustering images o to views v.

(1) Unsupervised Learning: Cluster Images to Views

- An *image* is a sensory snapshot.
  - A *view* is a cluster of similar images.
- Cluster images so aggressively that:
  - Image variability is eliminated, but
  - Perceptual aliasing is increased.
- SSH map-building requires:
  - a distinctive state has a unique view, but
  - multiple dstates can have the same view.

Markov Localization

- Within current map m
  - Update location belief distribution: $p(x \mid m) \rightarrow p(x' \mid a, o, m)$
  - After action a: $p(x' \mid x, a, m)$
  - After sensory image o: $p(o \mid x', m)$
  - Normalization constant: α

$$p(x' \mid a, o, m) = \alpha p(o \mid x', m) \int p(x' \mid x, a, m) p(x \mid m) dx$$

Markov Simplified

- Markov localization is useful for both occupancy grids and topological maps.
- Markov update is greatly simplified in the topological map.
  - Many fewer states,
  - Reliable actions,
  - Sensory images clustered to views.

Reliable Actions

- The causal link $\langle x, a, x' \rangle \Rightarrow p(x' \mid x, a, m) = 1$

  while $x'' \neq x' \Rightarrow p(x'' \mid x, a, m) = 0$

- Simplifies the Markov update equation:

  from:

  $$p(x' \mid a, o, m) = \alpha p(o \mid x', m) \int p(x' \mid x, a, m) p(x \mid m) dx$$

  to:

  $$p(x' \mid a, o, m) = \alpha p(o \mid x', m) \sum \{ p(x \mid m) : \langle x, a, x' \rangle \}$$

Cluster Images into Views

- $p(o|x,m)$ is too small to be meaningful.
  - A sensory image o is very high-dimensional.
  - Cluster into a small set of views v.
  - $p(v|x,m)$ is meaningful, and can be estimated.

- Since a dstate has one view

$$p(x' \mid a, v, m) = \alpha\, p(v \mid x', m) \sum \left\{ p(x \mid m) : \langle x, a, x' \rangle \right\}$$

becomes

$$p(x' \mid a, v, m) = \alpha \sum \left\{ p(x \mid m) : \langle x, a, x' \rangle \wedge view(x', v) \right\}$$

- Prior uncertainty is carried forward and pruned.

How Many Clusters?
How Much Perceptual Aliasing?

- Use *k*-means clustering. Search for *k*.

- Agent uses the ***decision metric*** M:
  - Rewards tight clusters, and clear separation.
  - Agent select *k* that gives largest value of M.

- Researchers use ***evaluation metric*** U:
  - Information dstate *x* provides about view *v*.

- Ideal result: largest *k* for which U=1.

(2) Explore the Environment: Build Causal/Topological Map

- Alternating sequence of images and actions.
  - Cluster images to views. Define dstates.

$$
\begin{array}{ccccccc}
o_0 & a_0 & o_1 & a_1 & \Lambda & a_{n-1} & o_n \\
v_0 & & v_1 & & \Lambda & & v_n \\
x_0 & & x_1 & & \Lambda & & x_n
\end{array}
$$

  - Minimize model: dstates, paths, places. [Remolina & Kuipers, IJCAI-2001]

- Exploration eliminates uncertainty, and labels each image with the correct dstate.

(3) Supervised Learning to Recognize Dstates from Images

- Subtle discriminating features are lost in the noise to an unsupervised learner.
- With a supervisory signal,
  - the noise washes out, and
  - the subtle but true feature is reinforced.
- We use ***nearest neighbor*** learning:
  - Accuracy rises rapidly to 100%
  - because the sensory signal is very rich.

Physical Robot Experiments

- Lassie
  - RWI Magellan Pro
  - Sonar ring to avoid obstacles.
  - Laser range-finder gives sensory images.

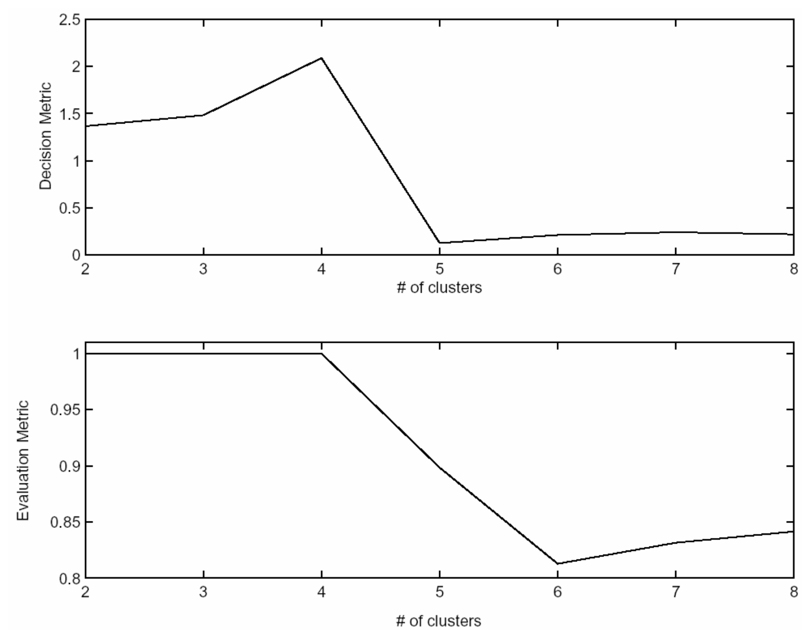
$$o_i \in \Re^{180}$$

Experiment 1:
A Super-Simple Environment

- The simplest environment with
  - perceptual aliasing and image variability,
  - masking a true discriminating feature.

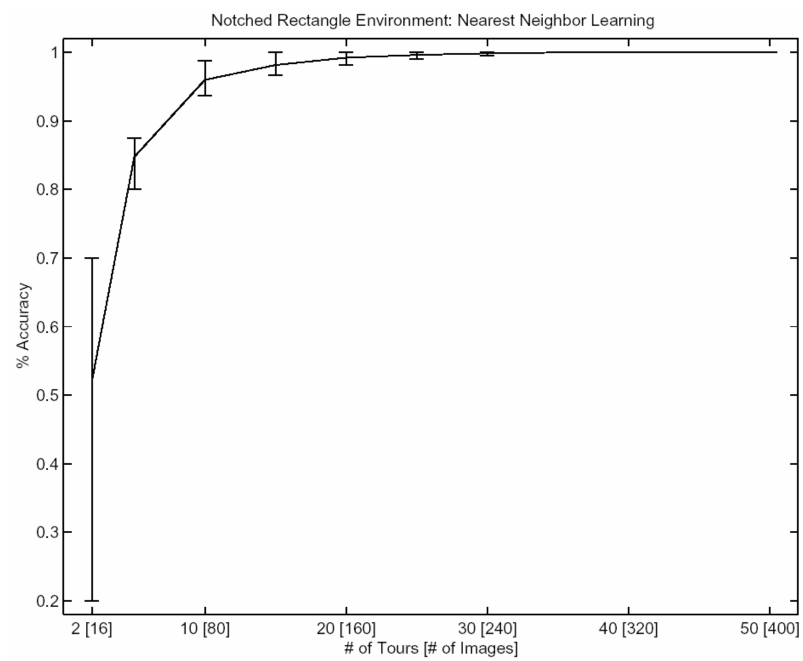
Experiment 1: Clustering and Mapping

- 50 clockwise cycles, 200 images.
- Decision metric picks k=4 clusters (views).
- Evaluation metric confirms optimality.
- Mapper identifies
  - 8 dstates,
  - 4 places, 4 paths.



Experiment 1: Place Recognition from Images

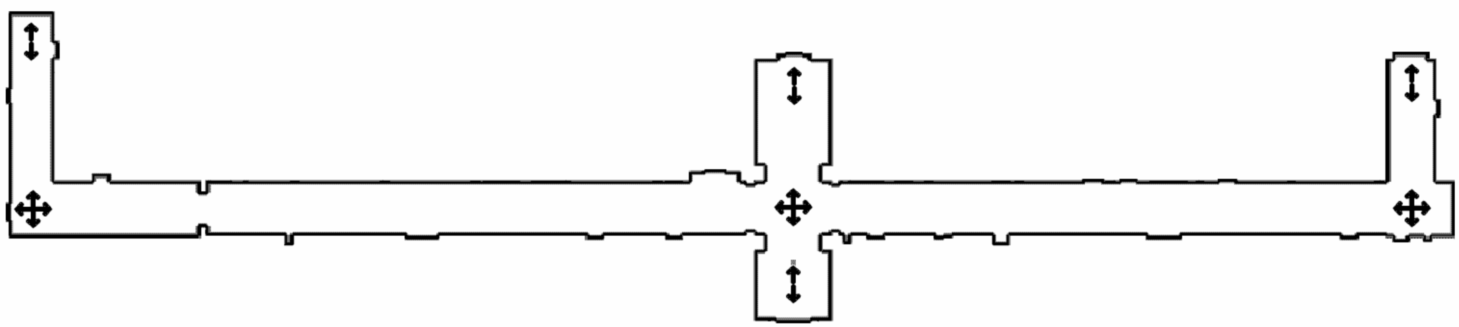
- 10-fold cross validation.
- Accuracy rises rapidly to 100%.



Experiment 2: Natural Office Environment

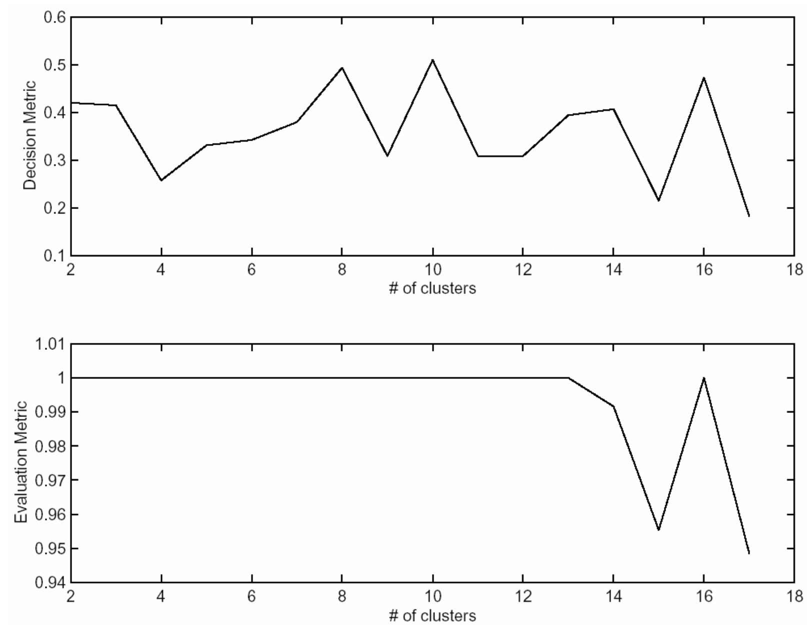
- Classroom building: 80 m long, cluttered.
  - Map has 20 dstates, 7 places, 4 paths.

- Image variability is the major problem.



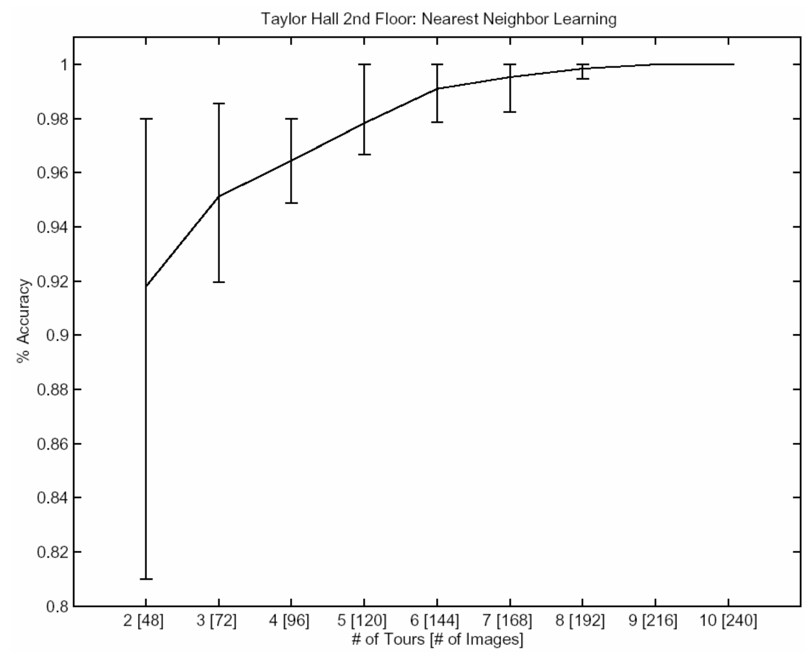
Experiment 2: Clustering and Mapping

- 10 circuits, 240 images
- Decision metric picks k=10 clusters (views)
- Evaluation metric says k=13 would work.
- Mapper identifies
  - 20 dstates
  - 7 places, 4 paths



Experiment 2:
Place Recognition from Images

- 10-fold cross validation
- Accuracy rises rapidly to 100%.
- Rich sensory images support better recognition



Future Work

- Extend to visual sensors.
  – Representation does not rely on range sensors.
  – cf. [Ulrich & Nourbakhsh, 2000]
- Eliminate need for physical hill-climbing.
  – Exploit strengths of *local* metrical maps.
- Error recovery when reliable actions fail.
  – Fall back to Markov localization, temporarily.

Conclusions

- Bootstrap learning works:
  - Unsupervised clustering abstracts the world.
  - Deductive inference builds a correct model.
  - Supervised learning with accurate labels gives high performance from real inputs.