In *The International Conference on Computer Vision Theory and Applications* (VISAPP 08),
Funchal, Portugal, January 2008.

Long-term vs. Greedy Action Planning for Color Learning on a Mobile Robot

Mohan Sridharan, Peter Stone
The University of Texas at Austin, USA
smohan@ece.utexas.edu, pstone@cs.utexas.edu

Keywords: Action Planning, Color Modeling, Real-time Vision, Robotics

Abstract: A major challenge to the deployment of mobile robots is the ability to function autonomously, learning appropriate models for environmental features and adapting those models in response to environmental changes. This autonomous operation in turn requires autonomous selection/planning of an action sequence that facilitates learning and adaptation. Here we focus on color modeling/learning and analyze two algorithms that enable a mobile robot to plan action sequences that facilitate color learning: a *long-term* action selection approach that maximizes color learning opportunities while minimizing localization errors over an entire action sequence, and a *greedy/heuristic* action selection approach that plans incrementally, one step at a time, to maximize the benefits based on the current state of the world. The long-term action selection results in a more principled solution that requires minimal human supervision, while better failure recovery is achieved by incorporating features of the greedy planning approach. All algorithms are fully implemented and tested on the Sony AIBO robots.

1 INTRODUCTION

Recent developments in sensor technology have enabled the use of mobile robots in several fields (et. al, 2003; Minten et al., 2001; Thrun, 2006). But these sensors require extensive manual calibration in response to environmental changes. Widespread use of mobile robots is feasible *iff* they can autonomously learn useful models of environmental features and adapt these models over time. But mobile robots need to operate in real-time under constrained resources, making learning and adaptation challenging.

We aim to achieve autonomous learning and adaptation in color segmentation – the mapping from pixels to color labels such as red, blue and orange. Significant amount of human effort is involved in creating the *color map*, and it is sensitive to environmental changes such as illumination. We enable the robot to exploit the *structure* of the environment – objects with known features, to autonomously plan an action sequence that facilitates color learning.

Planning approaches (Boutillier et al., 1999; Ghallab et al., 2004) typically require that all actions (and their effects) and contingencies be known in advance, along with extensive state knowledge. Mobile robots operate with noisy sensors and actuators, and possess incomplete knowledge of state and the results of their actions. Here the robot builds probabilistic models of the results of its actions. The models are used to plan action sequences that maximize color learning opportunities while minimizing localization errors over the action sequence. We show that this *long-term* action selection is more robust than a *greedy* approach that uses human-specified heuristics to plan actions incrementally (one step at a time).

2 RELATED WORK

Segmentation and color constancy are well-researched sub-fields in computer vision (Comaniciu and Meer, 2002; Shi and Malik, 2000; Maloney and Wandell, 1986; Rosenberg et al., 2001). But most approaches are computationally expensive to implement on mobile robots with constrained resources.

On mobile robots, the color map is typically created by hand-labeling image regions over a few hours (Cohen et al., 2004). (Cameron and Barnes, 2003) construct closed regions corresponding to known objects, the pixels within these regions being used to build classifiers. (Jungel, 2004) maintains layers of color maps with increasing precision levels, colors being represented as cuboids. (Schulz and Fox, 2004) estimate colors using a hierarchical Bayesian model with Gaussian priors and a joint posterior on robot position and illumination. (et. al, 2005) model colors using a mixture of Gaussians and compensate for minor illumination changes by modifying the parameters. (Thrun, 2006) distinguish between safe and unsafe road regions, modeling colors as a mixture of Gaussians whose parameters are updated using EM. Our prior work (Sridharan and Stone, 2007) presented a scheme to learn colors and detect large illumination changes, actions being planned one step at a time using human-specified heuristic functions. Instead, we propose an algorithm that enables the robot to learn the appropriate functions autonomously, so as to generate complete action sequences that maximize color learning opportunities while minimizing localization errors over the entire sequence.

3 EXPERIMENTAL PLATFORM AND COLOR MODEL

Figure 1: Aibo and field.

The experiments reported in this paper are run on the SONY *ERS-7* Aibo, a four-legged robot with a CMOS color camera with a limited field-of-view (56.9^o horz., 45.2^o vert.). The images are captured at 30Hz with a resolution of 208×160 pixels. The robot has three degrees-of-freedom in each leg, and three in its head. *All processing for vision, localization, motion and strategy is done on-board a* 576*MHz processor*. The Aibos are used in the RoboCup Legged League, a research initiative where teams of four robots play a game of soccer on an indoor field (Figure 1).

In order to operate in a color coded environment, the robot needs to recognize a discrete number of colors (N). A *color map* provides a color label for each point in the 3D color space (say RGB). Typically a human observer labels specific image regions over a period of an hour or more, and the color map is obtained by generalizing from these labeled samples. We compare two action-selection algorithms for autonomous color learning: (a) a novel approach that maximizes learning opportunities while minimizing localization errors over the entire sequence, and (b) an approach that plans actions incrementally, based on human-specified heuristics. Both planning schemes generate a sequence of poses (x, y, θ) that the robot moves through, learning one color at each pose. As described in (Sridharan and Stone, 2007), we assume that the robot can exploit the known environmental structure (position, shapes and color labels of objects) to extract suitable image regions at each pose, and model each color's distribution as either a 3D Gaussian or a 3D histogram. Assuming all colors are equally likely, i.e. $P(l) = 1/N$, $\forall l \in [0, N-1]$, each color's *a posteriori* pdf is proportional to the a priori pdfs. The color space is discretized and each cell in the color map is assigned the label of the *most likely* color pdf.

4 ALGORITHMS

In both action-selection algorithms for color learning, the robot starts out with no prior information on color distributions, and the illumination is assumed to be constant during learning. The robot knows the positions, size and color labels of objects in its environment, and its starting pose.

4.1 Long-term Planning

Algorithm 1 presents the long-term planning approach. The algorithm aims to maximize color learning opportunities while minimizing localization errors over the entire motion sequence. Three components are introduced: a motion error model (MEM), a statistical feasibility model (FM), and a search routine.

The MEM, represented as a back-propagation neural network (Bishop, 1995), predicts the error in the robot pose in response to a motion command, as a function of the colors used for localization (the locations of color-coded markers are known). For each robot pose, the FM provides the probability of learning each of the desired colors given that a certain set of colors have been learned previously. During training, the possible robot poses are discretized into cells, and the robot moves between randomly chosen poses running two localization routines, one with all colors known (to provide ground truth), and another with only a subset of colors known, collecting data for the MEM. At each pose, it also attempts to learn colors and stores a success count, which is normalized to provide the probability value in the FM.

During the testing phase, given a starting pose, the robot evaluates all possible paths through the discretized pose cells. The MEM provides the error expected if the robot travels from the starting pose to the first pose. The vector sum of the error and target pose provides the actual pose. If the desired color can

Algorithm 1 Long-term Action Selection.

Require: Ability to learn color models.
Require: Positions, shapes and color labels of the objects of interest in the robot's environment (*Regions*). Initial robot pose.
Require: Empty Color Map; List of colors to be learned - *Colors*.
Move between randomly selected target poses.
CollectMEMData() – collect data for motion error model.
CollectColLearnStats() – collect color learning statistics.
NNetTrain() – Train the Neural network for MEM.
UpdateFM() – Generate the statistical feasibility model.
GenCandidateSeq() – Generate candidate sequences.
EvalCandidateSeq() – Evaluate candidate sequences.
SelectMotionSeq() – Select final motion sequence.
Execute motion sequence and model colors (Sridharan and Stone, 2007).
Write out the color statistics and the Color Map.

be learned at this pose, the move to the next pose in the path is evaluated. Of the paths that provide a high probability of success, the one with the least pose error is executed by the robot to learn the parameters of the color models.

4.2 Greedy Action Planning

Algorithm 2 shows the greedy planning algorithm, where actions are planned one step at a time, maximizing benefits based on the current knowledge of the state of the world. The functions used for action selection, are manually tuned and heuristic, as with typical planning approaches (Ghallab et al., 2004).

The robot needs to decide the order in which the colors are to be learned, and the best candidate object for learning a color. The algorithm uses heuristic action models to plan one step at a time. Three functions are used to compute the *weights* for each color-object combination (l,i). Function 1 assigns a smaller weight to larger distances, modeling the fact that the robot should move minimally to learn the colors. Function 2 assigns larger weights to larger candidate objects, as larger objects provide more samples (pixels) to learn the color parameters. Function 3 assigns larger weights if the particular object (i) can be used to learn the color (l) without having to wait for any other color to be learned.

Algorithm 2 Greedy Action Selection.

Require: Ability to learn color models.
Require: Positions, shapes and color labels of the objects of interest in the robot's environment (*Regions*). Initial robot pose.
Require: Empty Color Map; List of colors to be learned - *Colors*.
1: $i = 0$, $N = MaxColors$
2: **while** $i < N$ **do**
3: $Color$ = BestColorToLearn(i);
4: $TargetPose$ = BestTargetPose($Color$);
5: $Motion$ = RequiredMotion($TargetPose$)
6: Perform $Motion$ {Monitored using visual input and localization}
7: Model the color (Sridharan and Stone, 2007) and update color map.
8: $i = i + 1$
9: **end while**
10: Write out the color statistics and the Color Map.

In each planning cycle, the robot uses the weights to dynamically determine the *value* of each color-object combination, and chooses the combination that provides the highest value. The robot then computes and moves to the target pose where it can learn from this target object, extracts suitable image pixels, and models the color's distribution (lines 6-7). The known colors are used to recognize objects, localize, and provide *feedback* for the motion, i.e. *the knowledge available at any given instant is exploited to plan and execute the subsequent tasks efficiently*.

5 EXPERIMENTAL SETUP AND RESULTS

We ran experiments to compare the performance of the two action-planning algorithms. The planning success (ability to learn all colors) averaged over different object configurations (six objects that can be placed anywhere along the outside of the field), each with 15 different robot starting poses, is shown in Table 1. We also had the robot move through a set of poses using the learned color map and measured localization errors – see Table 2.

Config	Plan (%)
Learn (global)	100
Learn (heuristic)	89.3 ± 6.7

Table 1: Planning Accuracies using the two planning schemes. Global planning is better.

With the global planning scheme, the robot is able to generate a valid plan over *all* the trials. The lo-

Config	Localization error		
	X (cm)	Y (cm)	θ (deg)
Learn (global)	7.6 ± 3.7	11.1 ± 4.8	9 ± 6.3
Learn (heuristic)	11.6 ± 5.1	15.1 ± 7.8	11 ± 9.7
Hand-labeled	6.9 ± 4.1	9.2 ± 5.3	7.1 ± 5.9

Table 2: Localization Accuracies using the two planning schemes. Global planning is better.

calization accuracies are comparable to that obtained from a hand-labeled color map, and better than the heuristic planning scheme. Modeling the motion errors and the feasibility of color learning enables the global planning scheme to generate robust plans, and the *replanning* feature of the heuristic approach can be used when the plan fails due to unforeseen reasons.

The online color learning process takes a similar amount of time with either planning scheme (≈ 6 minutes of robot effort) instead of more than two hours of human effort. The initial training of the models (in global planning) takes 1-2 hours, but it proceeds autonomously and needs to be done only once for each environment. The heuristic planning scheme, on the other hand, requires manual parameter tuning over a few days, which is sensitive to minor environmental changes.

6 CONCLUSIONS

The potential of mobile robots can be exploited in real-world applications only if they function autonomously. For mobile robots equipped with color cameras, two major challenges are the manual calibration and the sensitivity to illumination. Prior work has managed to learn a few distinct colors (Thrun, 2006), model known illuminations (Rosenberg et al., 2001), and use heuristic action sequences to facilitate learning (Sridharan and Stone, 2007).

We present an algorithm that enables a mobile robot to autonomously model its motion errors and the feasibility of learning different colors at different poses, thereby maximizing color learning opportunities while minimizing localization errors. The global action selection provides robust performance that is significantly better than that obtained with manually tuned heuristics.

Both planning schemes require the environmental structure as input, which is easier to provide than hand-labeling several images. One challenge is to combine this work with autonomous vision-based map building (SLAM) (Jensfelt et al., 2006). We also aim to extend our learning approach to smoothly detect and adapt to illumination changes, thereby making the robot operate with minimal human supervision under natural conditions.

REFERENCES

Bishop, C. M. (1995). *Neural Networks for Pattern Recognition*. Oxford University Press.

Boutillier, C., Dean, T. L., and Hanks, S. (1999). Decision Theoretic Planning: Structural Assumptions and Computational Leverage. *Journal of AI Research*, 11:1–94.

Cameron, D. and Barnes, N. (2003). Knowledge-Based Autonomous Dynamic Color Calibration. In *The International RoboCup Symposium*.

Cohen, D., Ooi, Y. H., Vernaza, P., and Lee, D. D. (2004). *UPenn TDP, RoboCup-2003: RoboCup Competitions and Conferences*.

Comaniciu, D. and Meer, P. (2002). Mean Shift: A Robust Approach Towards Feature Space Analysis. *PAMI*.

et. al, F. A. (2005). On-Line Color Calibration in Non-Stationary Environments. In *The International RoboCup Symposium*.

et. al, J. P. (2003). Towards Robotic Assistants in Nursing Homes: Challenges and Results. *RAS Special Issue on Socially Interactive Robots*.

Ghallab, M., Nau, D., and Traverso, P. (2004). *Automated Planning: Theory and Practice*. Morgan Kaufmann, San Francisco, CA 94111.

Jensfelt, P., Folkesson, J., Kragic, D., and Christensen, H. I. (2006). Exploiting Distinguishable Image Features in Robotic Mapping and Localization. In *The European Robotics Symposium (EUROS)*.

Jungel, M. (2004). Using Layered Color Precision for a Self-Calibrating Vision System. In *The International RoboCup Symposium*.

Maloney, L. T. and Wandell, B. A. (1986). Color Constancy: A Method for Recovering Surface Spectral Reflectance. *Opt. Soc. of Am. A*, 3(1):29–33.

Minten, B. W., Murphy, R. R., Hyams, J., and Micire, M. (2001). Low-Order-Complexity Vision-Based Docking. *IEEE Transactions on Robotics and Automation*, 17(6):922–930.

Rosenberg, C., Hebert, M., and Thrun, S. (2001). Color Constancy Using KL-Divergence. In *The IEEE International Conference on Computer Vision (ICCV)*.

Schulz, D. and Fox, D. (2004). Bayesian Color Estimation for Adaptive Vision-Based Robot Localization. In *IROS*.

Shi, J. and Malik, J. (2000). Normalized Cuts and Image Segmentation. *In IEEE Transactions on PAMI*.

Sridharan, M. and Stone, P. (2007). Color Learning on a Mobile Robot: Towards Full Autonomy under Changing Illumination. In *IJCAI*.

Thrun, S. (2006). Stanley: The Robot that Won the DARPA Grand Challenge. *Journal of Field Robotics*, 23(9):661–692.