Multiple-Object Detection in Natural Scenes with Multiple-View Expectation Maximization Clustering*

David R. Thompson and David Wettergreen
The Robotics Institute
Carnegie Mellon University
5000 Forbes Ave.
Pittsburgh, PA 15213, USA
{*drt, dsw*}*@ri.cmu.edu*

***Abstract*— Mobile robots and robot teams can leverage multiple views of a scene to improve the accuracy of their maps. However non-uniform noise persists even when each sensor's pose is known, and the uncertain correspondence between detections from different views complicates easy "multiple view object detection." We present an algorithm based on Expectation/Maximization (EM) Clustering that permits a principled fusion of the views without requiring an explicit correspondence search. We demonstrate the use of this algorithm to improve mapping performance of robots in simulation and in the field.**

***Index Terms*— Vision and Recognition, Mapping, Distributed Robots and Systems, Sensor Fusion**

I. Introduction

Fusing multiple views of a scene yields an analysis is generally more accurate than any one view alone. This is crucial in unstructured environments where occlusion, shadows, and aspect changes thwart attempts at universal recognition algorithms. In particular, multiple-view object detection offers an alternative that improves performance and increases a system's chance at finding the correct objects. This duplication could involve a mobile sensor in a static environment — by looking back at an area that it has already seen a robot gathers additional information that it can use to increase the accuracy of its maps — or a team of robots that view the same scene from several angles and ranges. Also several inexpensive, static detectors can combine data to approximate the precision of a single more expensive sensor.

Multiple-view object detection involves finding attributes of the real objects that maximize the likelihood of the observed detections. Researchers have already shown that a team of robots can track a single object accurately by employing Kalman filtering over multiple views [1], [2]. This paper addresses the more complicated problem of tracking multiple detections. Here techniques like Kalman filtering only suffice when the correspondence between objects from different views is known. Finding these associations is hard because the number of possible correspondences scales exponentially with the number of objects. Data association is now the most difficult part of the estimation.

Finding correspondences in multiple views from a mobile platform is the elegant dual of tracking moving objects from stationary sensors. It is not surprising then that the data association problem is a fundamental issue in the tracking and surveillance field. A complete survey of tracking techniques is outside our scope, but a brief overview gives insight into the point correspondence problem.

Fig. 1. The exploration rover used for the field experiment, shown here in the Atacama Desert of Chile. A pan-tilt camera mount permits improved object detection through "fly-by" imaging from multiple angles.

Searching through all possible correspondences between detections is intractable, so practical tracking methods apply various constraints to narrow the search. For a single camera, tracking multiple objects, one can limit possible correspondences to a small spatio-temporal window around each detection and use assumptions of constant motion or appearance to resolve ambiguities. Techniques like Multiple

*This research was supported by NASA under grants NNG0-4GB66G and NAG5-12890

Hypothesis Tracking [4] prune the correspondence search by considering only high-probability associations. Joint Probabilistic Data Association [5] accomplishes the same task with a probabilistic weighting of associations.

The multiple view case is more difficult because correspondences must also account for transformations between features as viewed from different angles. Many solutions register object positions using scene geometry [6]. A more general method treats correspondence as a "weighted graph matching" where points from neighboring views are associated in pairs that minimize some distance metric [7]. Finally, a host of heuristic methods use assumptions about proximity and motion to match points in neighboring images [3].

Unfortunately the particular challenges of robot mapping limit the use of these algorithms for multiple-view object detection. Tracking solutions assume that the views of a scene are ordered so that consecutive frames differ only by a small amount. To avoid combinatorial explosion they search for correspondences only within neighboring frames. However a mobile robot mapping a large area may obtain widely separated views. A multiple-view algorithm for robot mapping should be able to take account of all views simultaneously to find an optimum.

This paper discusses an algorithm based on a Gaussian mixture model [8] and Expectation/Maximization (EM) clustering [9] that finds a fast maximum-likelihood solution without resorting to an explicit correspondence search. It begins with a well-localized robot that observes a scene containing many objects, and builds an optimum object map by estimating correspondence probabilities between all observations and objects simultaneously. Computation scales according to $O(mnq)$, where m is the number of views, n is the number of detections in each view, and q is the number of objects in the model.

The following section presents the basic idea behind multiple-view EM. Then we suggest a practical algorithm that applies multiple-view EM to realistic cases with false negatives and non-overlapping data. Finally we demonstrate the algorithm's use in simulation and a simple field experiment (Figure 1).

II. Multiple-view Expectation / Maximization

Consider a set $V = \{v_1, v_2, ..., v_m\}$ of views of the scene. For each view v_i an object detector provides a list of n probable detections $X_i = \{x_{i1}, x_{i2}, ..., x_{in}\}$. Some are false positives while others correspond to real objects in the environment. Often there are duplicates — two detections from different views that actually refer to the same real object. Each detection x_{ij} is associated with a real-valued vector of p features — the object's position and size, for example — that are subject to sensor inaccuracy.

We represent real objects in the scene as multivariate fixed-width Gaussian probability densities in this feature space. The task is to find a scene model M consisting of the q Gaussian means $\{\mu_1, \mu_2, \ldots, \mu_q\}$ which has the highest probability of generating the observed detections. For simplicity of example we assume that sensor noise and detection likelihood is the same for each object and uniform over the entire feature space. The probability of an object at μ_k generating a detection x_{ij} is proportional to the object's probability density function,

$$P(x_{ij}|\mu_k) \propto G(x_{ij}, \mu_k) \tag{1}$$

where $G(x_{ij}, \mu_k)$ is the value at x_{ij} of a Gaussian distribution centered on μ_k. In a scene with many objects the probability of a Gaussian at μ_k having generated a detection depends on the locations of other objects (Fig. 2). We use the standard notation $P(\mu_k|x_{ij})$ to represent the posterior "membership probability" that the object at μ_k is responsible for having generated x_{ij}:

$$P(\mu_k|x_{ij}) = \frac{G(x_{ij}, \mu_k)}{\sum_{h=1}^{q} G(x_{ij}, \mu_h)} \tag{2}$$

A traditional maximum-likelihood approach for fitting μ_k would treat detections as independent samples and ignore important information about which view generated each detection (Fig. 3). If we assume that each object generates a single detection per view then the membership probability given other detections in the view, written $P(\mu_k|x_{ij}, X_i)$, should sum to unity for each cluster over the detections in X_i. While finding appropriate values for these probabilities is difficult, we can approach a solution by parameterizing $P(\mu_k|x_{ij}, X_i)$ in order to minimize the error:

$$\delta^2 = ([\sum_j P(\mu_k|x_{ij}, X_i)] - 1)^2 \tag{3}$$

Our parameterization simply weights the Gaussian component at μ_k with a different mixing factor β_{ik} for each view v_i.

$$P(\mu_k|x_{ij}, X_i) = \frac{\beta_{ik} G(x_{ij}, \mu_k)}{\sum_{h=1}^{q} \beta_{ih} G(x_{ij}, \mu_h)} \tag{4}$$

To force the sum of conditional probabilities toward 1 we define β_{ik} to increase as the difference δ is positive and decrease it if δ is negative. A logistic transform keeps the mixing coefficient bounded and positive.

$$\beta_{ik} = \frac{1}{1 + \exp(\sum_j P(\mu_k|x_{ij}, X_i) - 1)} \tag{5}$$

The result of the adaptive mixture weights is a "soft normalization," where membership probabilities shift to ensure that all objects contribute equally to the detections in every view.

If membership values were known with certainty it would be possible to calculate the most probable positions of the objects in feature space. Because the objects have fixed-width Gaussian distributions we could express their maximum-likelihood locations as weighted sums of the detections:

$$\mu_q = \frac{\sum_i \sum_j P(\mu_k|x_{ij}, X_i)x_{ij}}{\sum_i \sum_j P(\mu_k|x_{ij}, X_i)} \quad (6)$$

These membership probabilities are not known in general. However, we can estimate their values using expressions (4) - (6) that permit Expectation Maximization [2]. By estimating the membership probabilities $P(\mu_k|x_{ij}, X_i)$ and optimum means μ_k in turn one can reach a local maximum to the data likelihood.

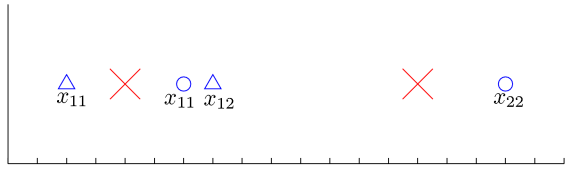

Fig. 2. A scene where two objects are imaged from two views which results in four different observations. The triangles represent the objects as observed in the first view; the circles represent objects detected in the second view.

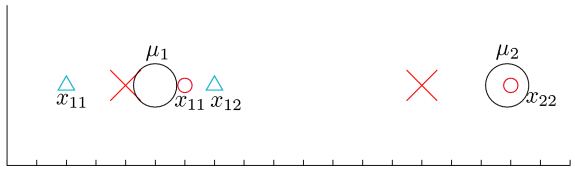

Fig. 3. A traditional mixture model treats detection memberships as independent. This results in non-intuitive estimates for object locations (large open circles at μ_1 and μ_2).

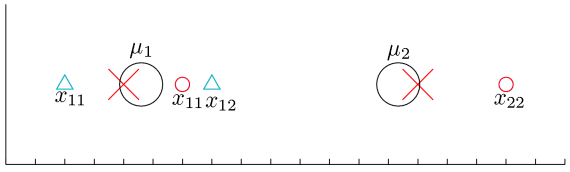

Fig. 4. With normalization all of the objects contribute equally to each view. This associates object μ_2 with observation x_{12} and yields a more reasonable position estimate.

In practical applications false positives and negatives might make it difficult to choose the appropriate number of clusters. One could test different numbers of clusters using cross-validation with additional views that have been reserved exclusively for that purpose. In the simulations of section III we employ a simpler strategy which places a cluster center on each detection of a single view. The complete algorithm is as follows:

initialize with a μ_k for each detection from view 1
do:
  for all clusters μ_k and detections x_{ij}:
    1) calculate every $P(x_{ij}|\mu_k)$ with (1)
    2) calculate every β_{ik} with (5)
    3) calculate every $P(\mu_k|x_{ij}, X_i)$ with (4)
  for all clusters μ_k:
    update cluster means according to (6)
while any μ_k has moved by an amount $> \epsilon$

III. Experiments

We provide two experiments to measure the improvements offered by multiple views. The first uses a simulated environment with idealized sensor noise. Next we provide more realistic results with field tests from a robotics expedition in the Atacama Desert of Chile.

A. Simulation Results

Consider a scene containing objects arranged at regular intervals on the unit circle (Fig. 5). These generate detections in separate views where each detection is perturbed by some amount of Gaussian noise. The experiments that follow test various single- and multiple-view detection techniques with this scene. In each case we calculate model error by matching each real object in turn with the closest remaining unmatched cluster. The total error score for the scene is the sum of squared distances between objects and their associated estimates.

Figure 6 shows performance with varying noise levels for three detection techniques. The test consisted of 50 trials with four views of the scene available for each trial. One detection method ignored the multiple views entirely, simply assigning a cluster to each detection in the original. Another utilized a traditional EM clustering without normalizing coefficients. Finally, the third method implemented the multiple-view EM algorithm described above. In each case the Gaussian widths were set according to the level of sensor noise. The experiment suggests that multiple views yield greater benefits for high noise levels, and either multiple-view method provides a better estimate of the objects' locations than a single view considered alone.

Curiously, the multiple-view constraint does little to increase accuracy over standard EM. The improvement is statistically significant; multiple-view EM achieved lower error in 42 out of 50 trials. This corresponds to a chi-square value of 11.56 and suggests with over 99.9% confidence that multiple-view EM outperforms regular EM for the task. Nevertheless, the magnitude of the improvement is small. This implies that scenarios like those detailed in Fig. 2 may be unusual occurrences.

Figure 7 shows a second simulation that held the noise level fixed while varying the number of views between one and five. Most of the the performance benefit comes from a single additional view; sum-squared error improves dramatically at first, and additional views after the second provide a smaller benefit. Nevertheless, there is a significant difference between the accuracy of models constructed with two views and five, suggesting that a large number of views may still be useful for applications requiring extremely high accuracy.

These simulations are unlike field conditions in several ways. In practice detector error is not perfectly Gaussian, nor is it independent from one view to the next; the difficult objects will generate noisy data for most or all of the views in which they appear. False positives are also likely. Nevertheless, it is encouraging that most of the performance benefits in this simple case are realized with the first additional view of the scene. It suggests that multiple-view detection need not demand an inappropriate amount of a field robot's time (or in the case of static sensor networks, an excessive number of nodes) to yield detection accuracy benefits.

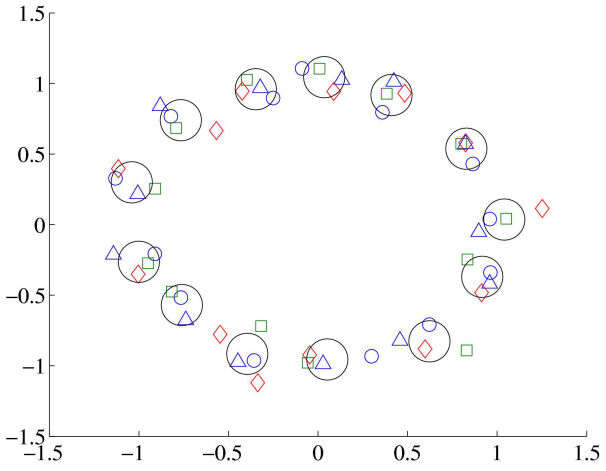

Fig. 5. A plot of detections using multiple views generated by perturbing objects on a unit circle by Gaussian noise. The small shapes correspond to the detections from different views while the large circles correspond to cluster centers found by EM.

B. Field Results

In order to test the system in field conditions we applied the multiple-view detection algorithm to the problem of mapping science targets during robotic planetary exploration. In future Mars missions, autonomous geology would permit rovers to make intelligent decisions about what experiments to perform and what data to return to Earth. In particular, accurate maps of the locations of rocks would be a valuable asset to an autonomous rover geologist. Several aspects of this task make it a good candidate for multiple view detection. First, finding geological targets autonomously is difficult — in the case of rocks, detectors often identify less than 80% of the real targets [10]. Second, the environment is static; we can assume it does not change between images from different locations.

The rover platform employed for the experiment was Zoë, an exploration robot constructed at Carnegie Mellon University (Fig. 1). Zoë is a wheeled platform featuring a suite of cameras that offer a 21-degree field of view with a resolution of 1280x960 pixels. They are mounted on a 2-meter mast with a pan-tilt unit that provides full 360-degree coverage of the environment. The cameras and pan-tilt unit are free to operate while the rover is in motion, permitting "fly-by" imaging of the terrain during an autonomous traverse.

The test was performed during navigation tests in the Atacama Desert of Chile. A data collection script caused Zoë's cameras to image a rocky patch of ground during an extended drive action. The sampling locations occurred several meters apart (Figs. 9 and 10). A science target detection algorithm [10] running off-board identified the rocks in each image. Finally, the detections from both views were registered to a world coordinate frame (Fig. 8). The feature space consisted

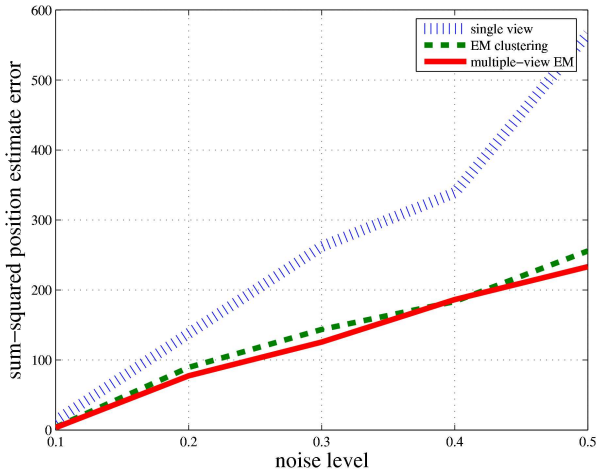

Fig. 6. Object detection error for four views of the synthetic scene with varying amounts of noise.

of the position, in latitude and longitude, of each rock.

After registration, multiple view EM was applied to the two views. Because the number of objects in the scene was unknown, the procedure began with a cluster on *every* detection. With each iteration of EM the number of clusters was reduced adaptively by merging any that had collapsed to the same point. Finally, the detected rocks were projected back in to the image to provide a visual representation of performance. Fig. 11 shows a traditional detection using only the rocks discovered from the first view. The gray area represents the ground visible from the second image. Each white circle within the region corresponds to a positive detection.

Fig. 12 illustrates detections from the two views synthesized using the multiple view EM algorithm. Within the common field of view the additional image improves accuracy; single view detection finds 31 rocks while the two fused views correctly detect 48 rocks.

IV. Conclusions

In this paper we explored techniques for boosting detector accuracy by fitting a world model to detections from multiple locations. The method uses a modified EM clustering algorithm to solve the data association problem for an arbitrary number of views. This technique is general enough that it could be extended to other more sophisticated mapping and detection scenarios. Most environments will exhibit non-uniform noise — the detection of distant objects is generally more uncertain than near objects. Variable-width Gaussian distribution functions might alleviate this problem.

Much object recognition work to date focuses on detecting objects from a sensor snapshot of the world at a single spatial and temporal location. This has the advantage of simplicity: the object recognition task can be insulated from other issues like robot localization and mapping. Despite this we believe that accuracy benefits can be gleaned by fusing data from multiple locations and times. Multiple view object detection is a general technique that holds the potential to improve accuracy without changing the basic detection method.

Acknowledgment

Many thanks to Scott Niekum, Trey Smith, Michael Wagner, Dominic Jonak, Joseph Flowers and Chris Williams.

References

[1] A. Stroupe, M.C. Martin, and T. Balch "Distributed Sensor Fusion for Object Position Estimation by Multi-Robot Systems", *IEEE International Conference on Robotics and Automation*, May, 2001.

[2] R. Hanek, T. Schmitt, M. Klupsch, and S. Buck. "From Multiple Images to a Consistent View", *RoboCup-2000: Robot Scoccer World Cup IV*, Springer-Verlag, 2001.

[3] C. J. Veenmman, M.J.T. Reinders, and E. Backer. "Resolving Motion Correspondence for Densely Moving Points." IEEE Transactions on Pattern Analysis and Machine Intelligence, 23:1, Jan 2001.

[4] D. B. Reid, "An Algorithm for Tracking Multiple Targets", *IEEE Transactions on Automatic Control* vol. 24:6, 1979, pp. 843–854.

[5] T.E. Fortmann, Y. Bar-Shalom, and M. Sheffe. "Sonar tracking of multiple targets using joint probabilistic data association", *IEEE Journal of Oceanic Engineering*, vol. 8:3, July 1983, pp 173–184.

[6] S. Khan and M. Shah. "Consistent Labelling of Tracked Objects in Multiple Cameras with Overlapping Fields of View", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25:10, October 2003, pp. 1355–1360.

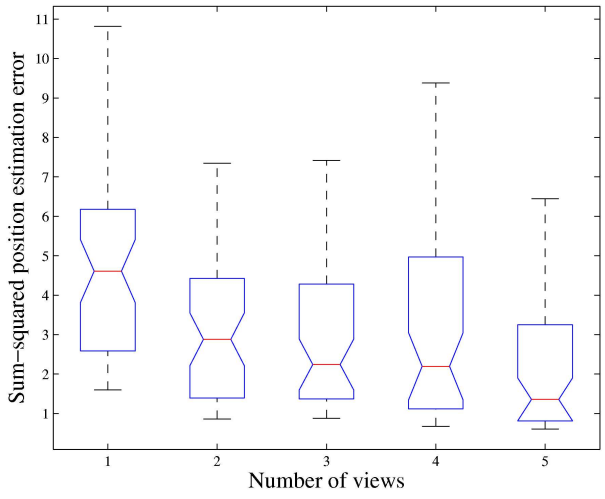

Fig. 7. Object detection error with varying numbers of views. Notches represent confidence intervals of 95% for the median, while boxes show the extent of upper and lower quartiles.

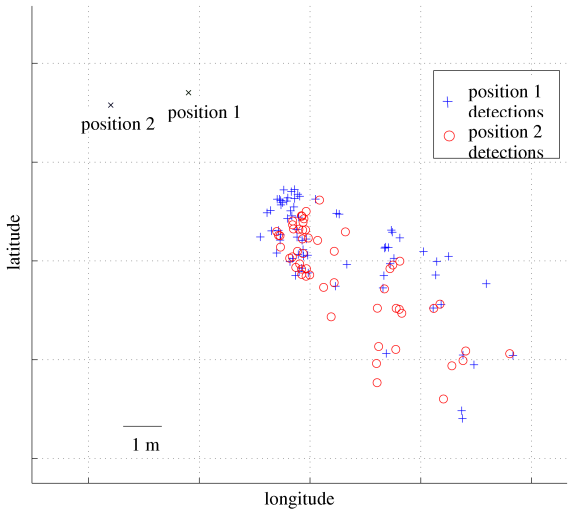

Fig. 8. A map of detections from the autonomous geology field experiment. A rover images rocky terrain from multiple locations and finds different rocks in each view.

[7] K. Shafique, M. Shah. “A Non-Iterative Greedy Algorithm for Multi-frame Point Correspondence”, *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 27:1, January 2005, pp. 51–65.
[8] Bishop, C. M. *Neural Networks for Pattern Recognition* Oxford U. P., Oxford, 1995.
[9] A. P. Dempster, N. M. Laird, and D. B. Rubin. “Maximum likelihood from incomplete data via the em algorithm”, *Journal of the Royal Statistical Society*, vol. 39, 1997, pp 1–38.
[10] D. Thompson, S. Niekum, T. Smith, and D. Wettergreen. “Automatic Detection and Classification of Geological Features of Interest”, *Proceedings of the IEEE Aerospace Conference*, Big Sky Montana, March 2005.

Fig. 9. The initial view from position 1 showing a rocky patch in the Atacama desert.

Fig. 10. An additional image captured from position 2. The rover traveled approximately 2 meters between the views.

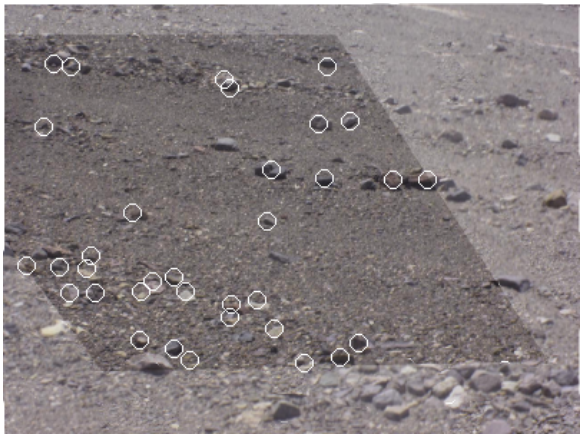

Fig. 11. Rock detection using only the first view. The gray area represents the terrain visible to the second view. Circles show the locations of detections after projection back into the image.

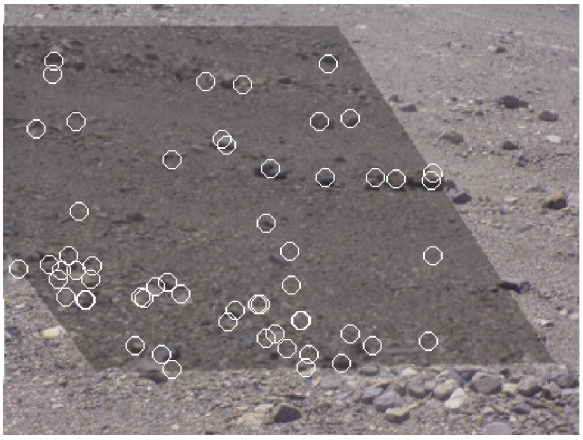

Fig. 12. Synthesis of detection results from both views. Detection locations from two images were combined using the EM Clustering method.