A Particle Method for Finding Distributed Objects

John H. Greenewald
Nonlinear Vision, LLC
P.O. Box 307
Xenia, OH 45385
USA
john.greenewald@wpafb.af.mil

Stanton H. Musick
Sensors, AF Research Laboratory
2241 Avionics Circle
Wright-Patterson AFB, OH 45433
USA
stanton.musick@wpafb.af.mil

Teri L. Piatt
Jacobs Sverdrup
4200 Colonel Glenn Hwy, Suite 500
Beavercreek, OH 45431
USA
teri.piatt@wpafb.af.mil

Abstract – *Detecting and identifying distributed objects in an image is a challenge with wide research application. This search problem is difficult, in part, because the individual, multi-pixel "spots" that make up a distributed object must be taken in the aggregate to have any relevance, and identification is possible only when the majority of the individual spots are detected and found to conform to an expected pattern. In this paper, particle filtering methods are extended in order to detect, localize, and identify a distributed object in a single cluttered image by maximizing the joint probability that a particular collection of spots is the object of interest. The method is illustrated using a "surrogate" estimation problem. Results demonstrate that the proposed method gives a high probability of correct detection and low object location error when the signal to clutter-plus-noise ratio is above 5 decibels.*

Keywords: distributed object, detection, nonlinear estimation, particle filter.

1 Introduction

Detecting and identifying distributed objects in an image is a recurrent problem, not just in Automatic Target Recognition (ATR) applications, but also in areas like astronomy, speech recognition, and biomedical imaging. It is a challenging problem since the individual, multi-pixel "spots" that make up a distributed object may hold little intrinsic identification information. Identification can only be accomplished when a collection of spots forms an object that conforms to an expected pattern. In this paper, knowledge of the object's geometric shape and configuration makes its detection possible, even amid heavy clutter.

Although distributed objects in clutter can sometimes be detected and identified by exhaustively searching the image, that approach would be computationally prohibitive here. For decades, researchers across many disciplines have developed peak picking algorithms to associate distributed objects using a variety of heuristic methods. A robust alternative to the heuristic methods is to use a particle filter method, which can greatly reduce the search time [1]. Once an object is detected, the algorithm will provide an estimate of the likelihood that the detection is the desired object, that is, the algorithm tells how well the detection conforms to the expected model. Information available to support this estimation includes a single gray-scale image, a rough estimate of the location of the camera, and an accurate location of the distributed object along with its unique spot distribution values.

Various well-known techniques, such as snakes and condensation, have successfully employed particle filtering methods with deformable templates and shape models to automatically detect and segment objects in the presence of clutter [2]. Deformable templates are used to model objects with B-splines to form continuous curves. The object templates are used to guide the sampling scheme to acquire sufficient edge data, which allows the object to be grossly segmented from its background. It is known that condensation works best when the tracked object is accurately modeled and has a precise location, either by manual determination or via training; however, the literature indicates that minor gaps in the object's edges can be tolerated, but significant gaps will impede segmentation leading to a failed solution [2]. Since the distributed objects described in this paper have gaps that occupy at least 80 percent of the object's span, these methods did not appear directly applicable. Additionally, even if condensation (or snakes) could be made to tolerate very large gaps in its deformable model, the distributed object of this problem is not precisely positioned. Thus, condensation (and related techniques) is still inappropriate.

Even though snakes and condensation may be ill-suited for the distributed objects of this problem, a clever implementation of their particle filtering foundation still holds promise for learning the location and structure of a distributed object. The general idea is that the particle filter could generate proposals for the object's location, size, orientation, and spot configuration, based on known constraints, and then reason recursively over the image to detect the true object of interest while rejecting clutter.

In the problem of this paper, it was necessary for the particle filter to detect, localize and identify a distributed object in a single cluttered image. The filter that emerged for doing this was tuned to the problem particulars, e.g. no time evolution, and an estimator for the unknown geometry variables of the state. In the end, the filter maximizes the joint probability that a particular collection of detected spots in the image is the object of interest. The proposed filter is evaluated using synthesized images with several levels of clutter/noise and various object configurations. The performance metric will be the receiver operating characteristic (ROC) curve for the detection process, where detection requires an object location error of under three pixels.

The setting for this work is a government program that has restricted the release of information about the actual problem. However, ideas regarding the detection of distributed objects have wide research application and can be released. In order to avoid excessive abstraction in the discussions that follow, Section 2 introduces a "surrogate" estimation problem that embodies all of the major challenges and constraints of the actual problem, while providing a reasonable platform for describing the solution methodology.

2 Problem Statement

For the "surrogate" problem, consider a UAV that must land at a particular friendly airfield at night. The onboard sensor is a camera that has, among its many responsibilities, the role of imaging the airfield so the precise location of the desired runway can be determined. The desired runway is "bar coded" with a series of lights (spots) along its flight line, as shown in Fig. 1. As the UAV nears the airfield, the sensor is tasked to make an image of the desired runway region. The automatic detection algorithm then reasons over the image, searching for the expected bar code pattern. When found (by matching estimated and expected patterns to within some minimum probability), the UAV steers its flight path to the desired runway and proceeds with the landing.

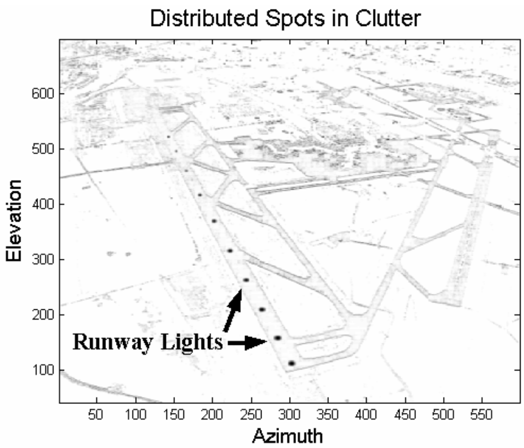

Fig. 1. Notional runway scene

Each distributed object is comprised of several multi-pixel spots, each spot caused by a single light in the runway bar code pattern. Typically, each discrete spot of the object spans many pixels in the image, although this attribute depends on both geometry and camera sensitivity and is therefore quite variable. Features such as pattern geometry, spot intensity, and spot count help identify the desired object in the grayscale image. Thus when data about the configuration of an object is properly accrued and interpreted, object recognition is possible. However, an automatic identification algorithm may struggle to overcome a combination of corrupting factors including inaccurate initial conditions, competing similar patterns, small spot size, large regions of clutter between spots, only one image, and a myriad of possible solutions. It should be noted that, as the runway landing problem is a surrogate, corruption levels are used that are probably far greater than would be experienced in practice.

Fig. 2 illustrates an image containing a single distributed object comprised of eight spots in clutter. The situation depicted in Fig. 2 is one of low clutter and noise. Note that the discrete spots of the desired object are clearly separated from their neighbors (which may be larger or smaller), are possibly obscured by clutter, may be commingled with the spots of other objects, and may be distorted or misplaced. In addition, both failed and false detections are possible, arising perhaps from smoke, water vapor, and other contaminants in the air.

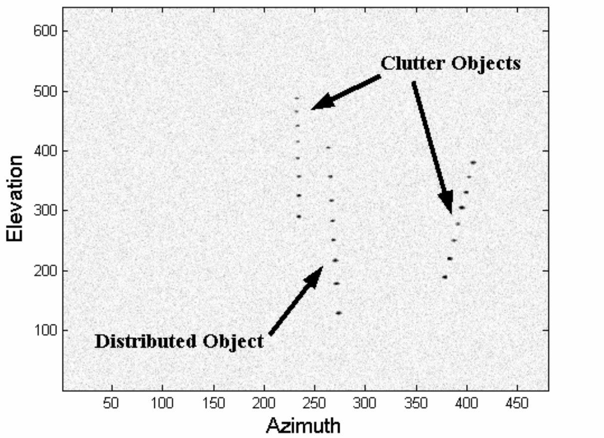

Fig. 2. Sample grayscale image of distibuted object in low clutter

The center points of the spots in the distributed object are assumed to be arranged along an arc that is compactly described by an analytic model of known form. The realized shape and size of the distributed object varies greatly because it depends on the range and line-of-sight of the object to the camera, somewhat unpredictable variables dictated by scenario conditions. In fact, the object shape may be fairly straight (presumably the usual case if the lights lie along a runway), slightly curved, or may have several inflection points. Furthermore, the object may appear anywhere in the image, may lie at any orientation within a known range of values, and has an unknown scale due to navigation errors that produce a poor estimate for the platform-to-runway range.

Many mature detection methods, such as change detection and constant false alarm rate, that work well in other situations fail in the single-image scenario of this problem. For instance, change detection approaches are precluded because there are no "revisit" images to compare and the object of interest is stationary. Approaches based on adjusting a threshold to achieve a constant false alarm rate (CFAR) require data of high SNR and processing that takes additional steps to associate spots to the object of interest. CFAR approaches are ill-suited because thresholding is not possible when spots exhibit low SCNR (signal to clutter-plus-noise ratio) and when throughput efficiency is low, due to the heavy clutter that results from the many confusers that must be considered.

Motivated by the success of condensation on estimation problems similar to the one considered here, this paper adapts the core methodology of particle filtering to detecting distributed objects that comply with the configuration of the lights on the desired runway. Although object identity information cannot be derived from individual object spots, it can be derived by reasoning about the locations and features of an entire set of spots constrained as a viable object configuration. A primary lever in this reasoning process is the analytic model that describes where spots may lie geometrically. If the geometry is favorable, the candidate spot probably belongs to an object; if not, it's probably clutter. A valid distributed object may then be distinguished from false contenders by the degree to which it conforms to the expected model. Model conformance is assessed probabilistically by sampling the image to accrue evidence against a proposed model that includes parameters of spot count, object shape, and relative spot intensity.

To test the robustness of the detection solution, realistic confusers of similar size, shape, and intensity were added to simulated sensor images in a manner that maximized confusion in the search process. The identification algorithm's performance was evaluated against a varying number of these clutter objects. For simplicity, confusers contained the same number of spots as the objects of interest.

3 Particle Filtering Essentials

The goal of this section is to provide the essentials of particle filtering (PF) as a foundation for understanding the proposed detection method. PF is an active field of research, and the reader is referred to [1-6] for a more complete background.

PF uses a simulation approach to approximately solve the prediction and update equations that form a state estimate for a stochastic system [4]. The conditional distribution of the state, given the observation(s), is approximated in PF using a finite sum of Dirac delta measures. PF uses Bayes rule, prior state information, expected models, and randomization to estimate probability densities without assuming their form. Samples are drawn from a known state proposal density without prior examination of the measurement. These proposals are then evaluated by sampling the measurement at the discrete particle points and weighting the result according to the proposal density. This step is called importance sampling. These resulting weights are used as the empirical sampling of the joint density of the state conditioned on the measurement. During the sampling step, particle filtering may generate many particles of low importance due to using randomization in the proposal process. A resampling step is used to replace low importance particles with duplicates of higher importance particles so the distribution of the number of particles matches their density. This helps particle witnesses to maintain their focus on the density of interest instead of on clutter.

Though robust to clutter, particle filtering introduces risk since even high SNR events may be undetected unless the true state density is sampled. Correctly sampling the state is assured only in the limit, as the number of particles increases without bound. Current research is focused on shaping proposals using overarching optimal filters [6]. However, these techniques do not apply to our single measurement case. Some method is required to reduce the dimensionality of the problem so accurate processing may be accomplished in near real time. Therefore, an alternative is to coarsely preview the measurement before sampling to guide the proposals to higher density regions instead of allowing "blind" sampling.

4 Detection Method

As stated previously, several obstacles impede successful detection in this problem. There are the geometric unknowns that arise from navigation errors. These unknowns are captured in the 3-element state of the distributed object, and include its scale, orientation, and the node number for the spot currently being considered (note that runway location can be directly inferred from these variables, once estimation is complete). Before UAV steering commands can be issued, the runway must be localized in the image by estimating this state. Clutter and noise are two additional obstacles that corrupt the detection process. Although all five of these obstacles are random, statistical data describing their distribution is available in the initial pdf which serves as the starting point in the detection process. In addition to this initial statistical data, deterministic data about the runway is also available, including its analytic model – represented here as a low-order polynomial – and the nodes (locations on that polynomial) where runway lights are present. Knowing the node locations equates to knowing spot spacing. The number of clutter objects within the image ranges from 10 for "clean" images to more than 70 for "moderately corrupted" images. The images are also corrupted with varying degrees of noise in order to simulate the effects of smoke, fog, mist, rain, dust, and other conditions that impede detection of the distributed object.

Turning now to solution processing, the first step convolves the image with a Gaussian point spread function in order to sharpen and regularize the image peaks and thereby increase the detection range. All subsequent reasoning is against this enhanced image, for which SCNR is evaluated as the ratio of the sum of the spot intensities of the distributed object to the sum of the intensities of nearby (within a validation gate) clutter and noise. The second step divides the full enhanced image into M rectangular sub-regions, each with equal area and the same aspect ratio (6::1 ~ elevation::azimuth). This research found that it was important to use a value for M that would ensure that at least one sub-region contained a "guide spot" from the true runway. Thus M was computed by analyzing the image, with more difficult images receiving increasingly higher values of M, i.e. M grows super-linearly as a function of the number of clutter objects in the full image, which is itself an approximation. The action of sub-dividing helps in cluttered images by providing a foundation to allow sub-regions of weak signature, that may happen to contain the runway, to compete with sub-regions that have strong signatures but no runway.

The third step in the solution procedure is to determine a rough measure G_m of the spot density in each sub-region, $m = 1, \cdots, M$. This measure is computed as the sum over the sub-region of the absolute values of the intensity gradients in the elevation direction. It is assumed that spot density, were it needed, could be computed as kG_m, where k is a constant scaling factor. In what follows, only the relative values of G_m are important.

The fourth step is to resample the spot density to spread the number of guide spots to be detected according to the observed spot density. The goal is to initialize the particle filter with at least one guide spot associated with the true distributed object. This approach seeks to minimize missed guide spot detection by concentrating resources on high importance regions.

The fifth step detects the guide spots for each sub-region. Each sub-region is examined to locate the number of spots assigned by step 4. Spots are repelled for regions with multiple guide spots to prevent multiple guide spots on a single object.

A sixth step samples the measurement at points radiating from the guide spots distributed at the expected orientation and spot spacing. Low likelihood guide spots based on orientation are removed from further consideration. This thinning step is a simple step that would be used in a final system but does not add much value here since clutter orientation is distributed as the runway in this simulation.

These six steps comprise a preprocessing phase using single-step particle filtering with resampling to provide M distributed guide spots with at least one on the true object of interest, i.e. one of the runway lights.

The final step in this solution procedure uses a particle filter to sequentially reason over the guide spot pixel locations, analytic models, expected densities, and the convolved image to determine which detected guide spot corresponds to the runway as well as geometric parameters that fully describe light locations.

The particle filter has states that represent orientation, scale, and a probability for each spot order number. The particle filter is initialized with the orientation determined from the noisy navigational aid, unity scale, and all eight spot numbers are equally probable.

The particle filter is challenged by the high dimensional requirements of the state and the intolerance of the measurement due to small spot sizes, wide spot separation, wide search regions due to navigation uncertainty, and clutter. Therefore, we describe an alternative sequential particle filter that employs fewer particles while reducing filter dimensionality. The algorithm seeks to jointly estimate some states while withholding direct sampling of other states and allowing them to be updated later. This is attempting to reduce the dimensionality of the optimization and the required number of particles. Therefore, a sequential optimization over the state space is proposed.

The particle filter methodology is used to sequentially propose particles to sample part of the state space according to a known expected range of state values. The importance sampling of these proposals is then used to update the estimates for all states. Resampling replaces low weight particles with duplicates of higher weight particles. This allows the search process to make proposals quickly and test them instead of employing an exhaustive search over a large domain.

The first step in this optimization will thin the particles and estimate the spot order. This is completed by proposing particles radiating from both sides of the guide spot according to a sampling of the orientation and spot separation densities and computing the importance weight. These weights are resampled expecting low weights are clutter and high weights are candidate spots within the true object. The distance between selected candidates from both sides and the guide spot is then determined. The pair of distances is then normalized to the detected arc length. These values are used to update the state representing spot order and orientation. The spot order is updated by comparing the ratios for each order and estimating their probability. For cases where the spacing is close to linear the likelihoods are nearly equal but for polynomial spacing the likelihood is more peaked at the correct order. Each particle is then importance sampled according to amplitude and likelihood observations. In the case where the guide spot is the first or last in order and all samples adjacent in one direction return with low weight then the order estimate is skewed to the appropriate order. The posterior weights are then resampled based on the importance sampling.

The resulting particles are compared in orientation to determine if a particular object has multiple detected spots. It is assumed that by this time, after several thinning and resampling iterations, that the true object will have more than one guide spot. If there are spots within an orientation validation gate, then the spot order states are examined to determine the spot order of the guide spot.

By this time the search space is reduced by the thinning aspect of the resampling step. However, the updated orientation for each guide spot is still a noisy estimate so the particle filter generates proposals distributed in angle and then generates spot center locations according to the updated models and geometry parameters of the state. The measurement is then sampled in the immediate neighborhood (2-3 pixels) of the proposal to center the spot. The orientation proposals are then importance sampled according to the spacing, amplitude, adherence to the expected curve, and proposal density. These importance weights provide the empirical sampling of the true state density. This concludes the state estimation recursion so the joint density samples are evaluated to determine the maximum likelihood that returns the estimated geometric parameters and likelihood for the true spot. The true spot location estimates are then determined from the models and geometric parameters.

The final declaration succeeds or fails depending on whether the true runway or a false candidate is found. With so many possible “spot” combinations, various model accuracy conditions, and low SCNR, a particle filtering approach of generating and testing proposals provides an efficient means of sampling the measurement to extract the truth.

5 Results

The detection algorithm described above was evaluated in a Monte Carlo simulation designed to emulate the lost runway detection problem. In each Monte Carlo run, an image was created and subjected to the detection algorithm in order to score the algorithm's success or failure. This section describes the simulation procedures, and then presents the Monte Carlo results.

The image formation stage begins by assembling a distributed object representing the runway. This object has eight spots, each shaped as a bivariate Gaussian function of 4-6 pixel extent. Both spot size and intensity vary inversely with range – corresponds to the elevation axis in all figures. The shape of the curve through the spot centers of the object is a randomly-generated low-order polynomial in range, while the spacing of the spots along that curve is a randomly-generated low-order polynomial in index number. The resulting object is seeded into the image frame at a random orientation (uniform between two limits) and a random location (also uniform across the dimensions of the image frame).

Next, a number of clutter objects were seeded into the image to create the desired SCNR level. Each clutter object was composed of eight spots of the same size as the true object, arranged on a curve sampled from the same density as the true object. The seeding operation corrupted the image using orientation values sampled from the same density as the true object and placed uniformly. Significantly, clutter objects employed a linear spacing model. Finally, the image was further corrupted with randomly generated Rayleigh noise. The sum of clutter and noise produced SCNR levels of 0, 12 and 20 dB.

Thus, the image synthesis stage produced images in which the fundamental difference between true and clutter objects was the analytic model for the spot spacing. These models do not help distinguish objects when only two adjacent spots are compared. Nearly all clutter objects satisfy validation gate conditions for the detection algorithm. Most clutter objects have adjacent spots that would also satisfy the true curve analytic models. The difference between truth and clutter is determined in the aggregate spot spacing. Therefore, a joint detection is required over multiple spots in order to make a correct identification. To make the detection more difficult, the clutter objects were allowed to interact and even obscure the true object spots.

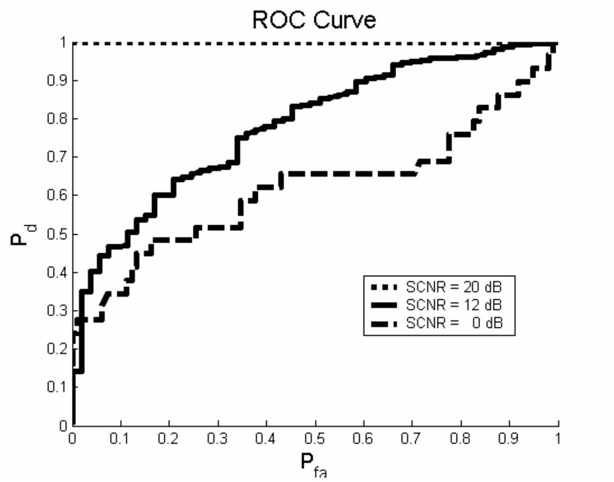

Fig. 3. ROC curves for three SCNR levels

The detection algorithm was evaluated using an ensemble of 400 Monte Carlo runs. As shown in the ROC curve of Fig. 3, detection and false alarm performance was very good from 20 dB down to 12dB SCNR. As expected, the algorithm performs well when only a few objects are imaged, as is the case at 20 dB where the algorithm was essentially perfect. As SCNR drops towards 0 dB, the densities for detection and false alarm begin to overlap, which leads to lowered declaration accuracy. This performance falloff may be due in part to inaccuracies in how the likelihood score (the score that allowed the experimental results to be ordered from best to worst) was constructed. Finally, note that computation efficiency was $O(M)$.

Low SCNR simulations lead to many challenging detection situations. Fig 4 shows an example image with 70 confuser objects and additive noise that together produced an SCNR of 0dB. The true distributed object is shown as superimposed circles and the estimated object as diamonds. The true object is in a high clutter region and the algorithm failed to reason correctly over the spots, thereby resulting in a false alarm.

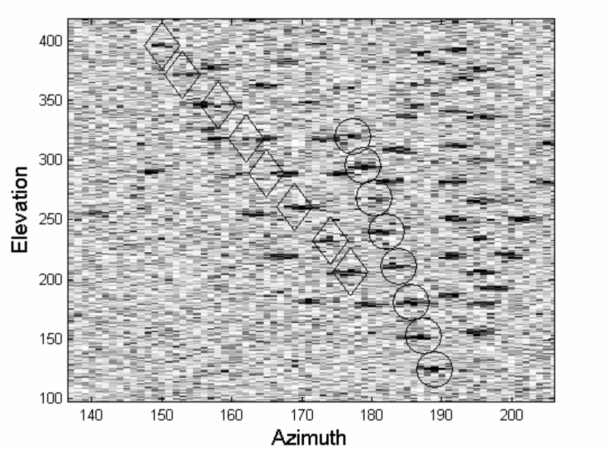

Fig. 4. Sample failed detection

Fig. 5 and Fig. 6 are close-up views of different regions within a single image. The clutter spots are distributed in a pattern that closely resembles the true object. In this case the detector selected the clutter, although both posterior likelihoods were high. Therefore, low SCNR detections may require more stringent requirements when the posterior likelihood is multi-modal.

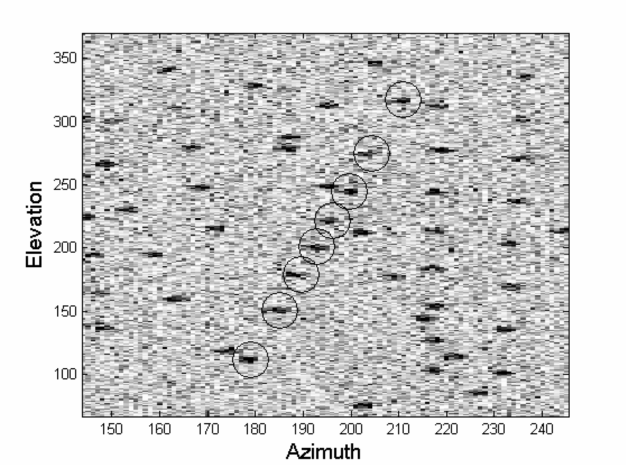

Fig. 5. Close-up of sample failed detection at 0dB. True object shown. See Fig. 6 for estimated.

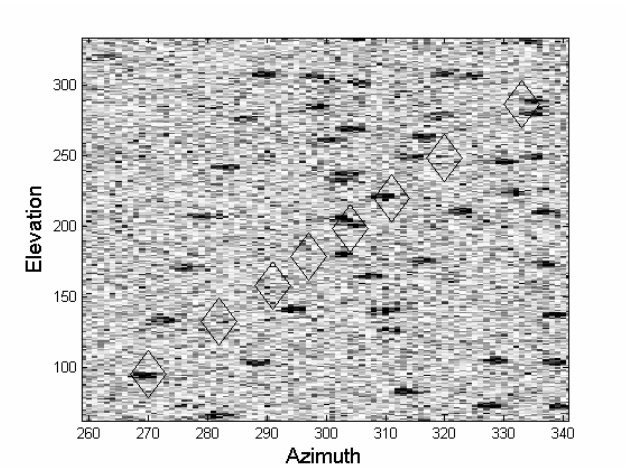

Fig. 6. Close up of sample failed detection. Diamonds mark the estimated object. See Fig 5 for true object.

Fig. 7 shows a correct detection at 0dB SCNR. In this case, spots at increased range (elevation) are dim and induced a small orientation error. The rms pixel error was small enough to declare this as a correct detection.

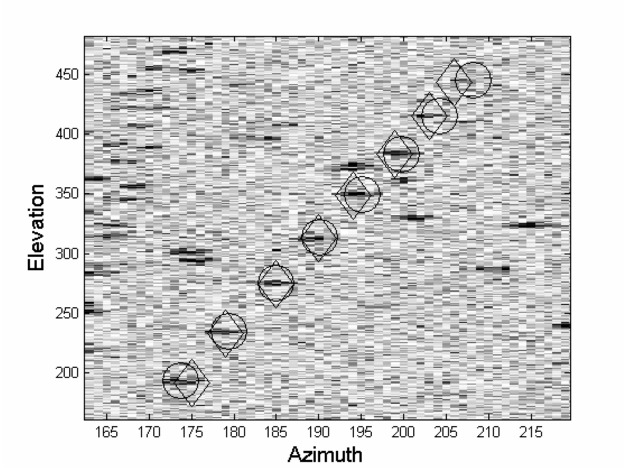

Fig. 7. Sample correct detection at 0dB

Fig. 8 shows a missed detection due to large orientation error. The estimation error was induced by using the low SNR guide spot and failing to sample at the correct orientation. This problem may be overcome by allowing more orientation particles.

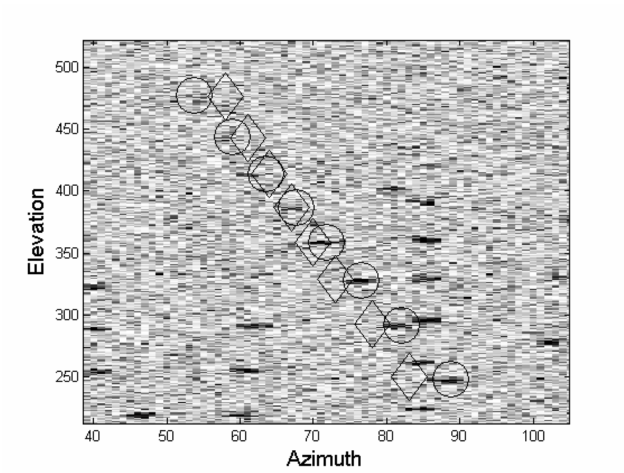

Fig. 8. Sample orientation error

6 Conclusions

This research developed a method to detect a distributed object in a single cluttered and noisy image. The method was applied to the problem of assisting an airborne platform to find a particular runway at night by detecting a display of lights placed along its length. The resources for this detection included a single, poorly-registered camera image of the runway region, and fore-knowledge of the geometric pattern of the light display. The proposed method, whose algorithm leans heavily on particle filtering technology, effectively detected the runway despite many confusers, significant noise, and severe image registration problems arising from platform navigation errors. The method is suitable for online implementation because it can be automated to execute quickly. Based on searches of the available literature and extensive reading, the method appears to be new. Other potential applications for recognizing distributed objects include reconstruction of signals and symbol detection in communications.

Acknowledgements

This work was funded by the U.S. Air Force Sensors Directorate under contracts F33615-03-C-1408 and F33615-99-D-1501, and by Grant 2304 ES01 from the Air Force Office of Scientific Research. The authors gratefully acknowledge the support and technical assistance of Dr. Jack Parker, Research Director in AFRL/SNJT, and Dr. Mark Delong, General Dynamics Advanced Information Systems, Dayton.

References

[1] Arnaud Doucet, Nando de Freitas, and Neil Gordon (eds). *Sequential Monte Carlo Methods in Practice*. Springer-Verlag, New York, NY, 2001.

[2] Andrew Blake and Michael Isard. *Active Contours*. Springer-Verlag, New York, NY, 2000.

[3] Stanton Musick, John Greenewald, Chris Kreucher, and Keith Kastella. Comparison of particle method and finite difference nonlinear filters for low SNR target tracking. In *Proc. of Fourth Int. Conf. Information Fusion,* Montreal, Canada, August 6-10, 2001.

[4] Sanjeev Arulampalam, Simon Maskell, Neil Gordon, and Tim Clapp. A tutorial on particle filters for online nonlinear/non-Gaussian Bayesian tracking. *IEEE Trans. on Signal Proc.*, Vol. 50, No. 2, February 2002.

[5] Augustine Ooi, Arnaud Doucet, Ba-Ngu Vo, and Branko Ristic. Particle filter for tracking linear Gaussian target with nonlinear observations. *SPIE AeroSense*, Orlando FL, April 2003.

[6] Shawn M. Herman. Joint passive radar tracking and target classification using radar cross section. In *SPIE Signal and Data Proc. of Small Targets*, pages 402-417, Vol. 5204, 2003.