

Real-Time Image-Based Topological Localization in Large Outdoor Environments

David M. Bradley, Rashmi Patel, Nicolas Vandapel and Scott M. Thayer

Robotics Institute

Carnegie Mellon University

Pittsburgh, Pennsylvania 15213

{dbradley,rpatel,vandapel,sthayer}@cs.cmu.edu

Abstract—This paper presents a real-time implementation of a topological localization method based on matching image features. This work is supported by a unique sensor pod design that provides stand-alone sensing and computing for localizing a vehicle on a previously traveled road. We report extensive field test results from outdoor environments, with the sensor pod mounted on both a small and a large all-terrain vehicle. Off-line analysis of the approach is also presented to evaluate the robustness of the various image features tested against different weather and lighting conditions.

I. INTRODUCTION

This paper addresses the problem of topological localization in large scale, partially known outdoor environments. The long term goal of this project is to develop integrated, stand alone capabilities for real-time localization in all weather conditions and over very long distances, on the order of several hundred kilometers. A typical task envisioned is to determine whether or not a vehicle is following a previously traversed path. Thus we are primarily interested in place recognition rather than metric localization in a map. The ideal approach will have minimal requirements in term of perception, instrumentation, computational power and memory storage.

Because of the susceptibility of GPS to jamming and satellite occlusions, we favor camera and laser based approaches. Terrain-based registration methods between data collected from the ground and the air have been used successfully for outdoor localization, but such approaches require ground features (i.e. rough terrain) that might not be present on long roads. In addition, these approaches are computationally expensive and require the storage of a large amount of 3-D data to handle long traverses [21]. Another approach is to match skyline features extracted from a camera against features from topographic maps. However, many areas either lack significant topographic features or have them obscured by vegetation, meaning another approach is needed for these situations [17]. Elevation profile matching has also been used successfully for aircraft navigation. In this approach, a ground profile extracted from range data or a history of robot poses is matched against a priori data. Such an approach requires a good local pose estimate which we do not assume here [6]. Landmark-based approaches can provide a viable solution, but the existence of densely located, unique,

easily detectable and stable scene elements is unlikely [23]. Therefore, this paper investigates image based topological localization methods that have been proven effective in indoor environments and also to some extent in urban outdoor environments [18].

This paper utilizes a unique stand-alone sensor pod with on-board computing to make two main contributions. First, a real-time implementation of image-based topological localization running on-board the sensor pod is demonstrated and tested on several vehicles in large outdoor urban and natural environments spanning several kilometers and thousands of images. Secondly, results are presented from extensive validation and testing of the effectiveness of weighted gradient orientation histograms as a global image feature for localization in these large outdoor environments.

The next section describes previous work relevant to this paper. Section III details the approach taken in this paper to image-based topological localization. The implementation of that approach in hardware and software is described in Section IV, and the results from extensive field testing are shown in Section V.

II. STATE OF THE ART

This section provides a brief review of the literature in topological localization along five dimensions:

- 1) Type of environment considered (indoor, outdoor urban, outdoor natural).
- 2) Perception system (sensing modality, features, data selection).
- 3) Localization model (brute force, probabilistic).
- 4) Integration level.
- 5) Field testing reported.

Numerous papers deal with indoor environments [20], [4] but only a very few deal with outdoor urban [18] or natural environments [5]. To our knowledge no work has been reported for the long range natural outdoor environments used in this paper.

Early work reported in the literature used monocular cameras, but to avoid field of view problems panoramic cameras became the sensor of choice [9]. The use of range data is far less common [2]. This paper employs monocular cameras, for ease of mounting on a wide variety of vehicles.

Our paper does not propose any new image-based features. From the many options available to us from the image

indexing and appearance-based object recognition literature, we selected weighted gradient orientation histograms of image sub-regions, for their efficiency and demonstrated performance [8], [7]. Other possible features include global features such as eigen-images [13] or color histograms [14], distributions of local attributes, or landmark detection and recognition in images [23].

Kuipers [10] proposes an approach to cluster images, using bootstrap learning, to reduce perceptual aliasing between scenes close in image space. Dima [3] uses Machine Learning techniques to select the optimal images to introduce into a database for learning autonomous driving. We use a complete database of all the images for each path, even non-informative or ambiguous images, since storing and searching our database of image feature vectors is relatively cheap.

Current work on place-based localization has focused on introducing probabilistic models to improve feature matching and the integration of sensor observations over time. For instance Gaussian mixture models have been used to model places, and hidden Markov models have shown to be useful for integrating sensor measurements over time [22], [19]. Transition detections are also used by some authors [15]. For the phase of our work reported in this paper we decided to implement a simple nearest-neighbor-based localization framework and evaluate its capabilities, reserving a more complex approach for the next phase.

Most reported work uses data collected from a mobile robot, but processed off-line. Torralba in [19] experiments with a fully integrated wearable set-up similar to [1]. In [22] Wolf presents results from on-board data processing using an indoor robot. To our knowledge this paper reports the first fully integrated system for place-recognition based localization in outdoor environments.

Finally, almost all of the work in the literature is based on very small data sets of a few dozen meters and several hundred images. To our knowledge this is the only reported work using paths containing thousands of images acquired over several kilometers.

III. APPROACH

A. Overview

The problem considered here is determining whether or not the vehicle is in an area that it has visited before, and if so where it is in that area. At each time step k a new sensor observation is available, which in this paper consists of a pair of color and near-infrared images (I_{nir}^k, I_{color}^k) . The sensor observations from previous traverses are stored in a database DB . Localization then consists of finding a function $F[(I_{nir}^k, I_{color}^k), DB] = M$ which returns the correct set of matches, if any exist in the database, for the current sensor observation.

As storing and comparing full images is expensive, this problem is made more tractable by defining a description function $D(I_{nir}^k, I_{color}^k)$ which creates a compact descriptor — a vector in feature space — for the image pair. Ideally this description function creates a bijection

between points in feature space and unique places in the environment. Solving the matching problem then involves defining a matching function for the descriptors, $G[D(I_{nir}^k, I_{color}^k), DB]$. A list of the notations used in this paper is given below.

I_{nir}^k ,

Near infrared image for sensor observation k .

I_{color}^k ,

Color image for sensor observation k .

$DB = \{DB^k\} = \{I_{nir}^k, I_{color}^k\}, k = 1 \dots m$:

Database of m sensor observations, each consisting of a pair of images.

$F[(I_{nir}^k, I_{color}^k), DB] = M$:

Function F that finds the set of observations M in the database which match observation k .

$D(I_{nir}^k, I_{color}^k) = X_k$:

Description function which produces a compact descriptor X_k for the image pair.

$G[D(I_{nir}^k, I_{color}^k), DB] = M$:

Function G that finds the set of observations M in the database which match descriptor k .

$d[X_k, X_j]$:

distance metric between two descriptors X_k and X_j in feature space.

S_t :

Location (state) of the robot at time t .

B. Image Features

The image features we selected were inspired by SIFT descriptors [11], and are similar to features used by Kosecka et al. in indoor environments [7]. In this paper we refer to the features as Weighted Gradient Orientation Histograms (WGOH). Each feature vector is created by dividing up the image into an m by n grid of sub-images (see Figure 1). Particularly when paired with a wide angle lens, dividing the image into subregions means that moderate violations of the static world assumption, such as pedestrians or cars passing through the camera's field of view, typically affect only a few subregions and leave most of the feature vector unchanged. Then for each region an 8-bin histogram is taken of the gradient orientations which is weighted by the magnitude of the gradient at each point, and the distance from the center of the region. Gradients near the edges of each region are weighted less than gradients near the centers, as those gradients are more likely to fall in a different region if the image is translated slightly. The sixteen, 8-bin histograms are then concatenated into a 128-element feature vector, which is normalized to unit length. We then follow the practice described in [12] of capping the size of any individual element of the feature vector to 0.2, and re-normalizing. This is done to reduce the dependence of the feature vector on any one particular region or strong gradient.

C. Database creation

Before the localization stage begins, the feature vectors are first computed from all of the images collected during a

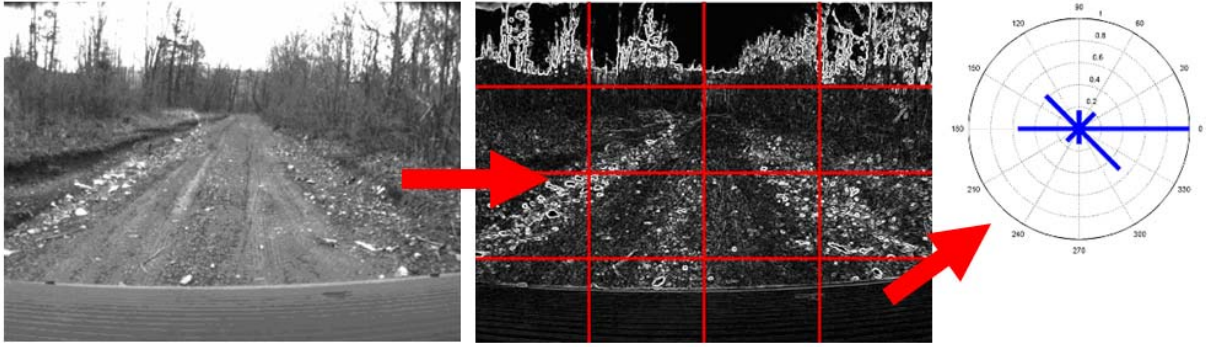


Fig. 1. Creation of the WGOH feature vectors: The image is first divided into 16 regions and an 8-bin histogram is taken of the gradient orientations in each region. The histograms are concatenated into a 128-element feature vector, and normalized to unit length. Any elements greater than 0.2 are then thresholded to 0.2, and the vector is re-normalized.

traverse through the environment, and placed in a database. This database can then be used as a nearest-neighbor classifier for localizing the robot in future runs through the same environment.

D. Localization

1) Finding the maximum likelihood location in the map:

The localization of the robot in the topological map is accomplished using a nearest-neighbor classifier trained on a database of descriptors collected in previous runs. Feature space distances are computed as 1 minus the dot product of the normalized feature vectors (Equation 1). This metric was chosen because of its superior performance to the cross-correlation method used in [8] on the test set described in Section V.

$$f(X, Y) = 1 - X^T Y \quad (1)$$

Our localization strategy assumes that the probability that two images come from the same location is a monotonically decreasing function of the feature-space distance between them. This assumption is well supported by the results from the field test described in section V. Figure 2 shows the probability that an image pair corresponds to the same location, versus the feature space distance between the feature vectors corresponding to each image. This probability was estimated from an exhaustive computation of all of the pairwise feature-space distances between the 37,481 sensor observations (pairs of NIR and color images) collected during the morning and afternoon of the second day of the field. This test set included over two million matching pairs and over one billion non-matching pairs. Matching pairs are defined as pairs of sensor observations taken within 3 m and 9° of orientation of each other.

2) *Detecting path deviations:* The minimum feature-space distance returned by the nearest neighbor classifier is then used to classify whether or not there are any valid matches for the current sensor observation in the database. Figure 2 shows the probability that a potential match is valid for a given feature space distance, as estimated from the data collected in section V. Determining whether the

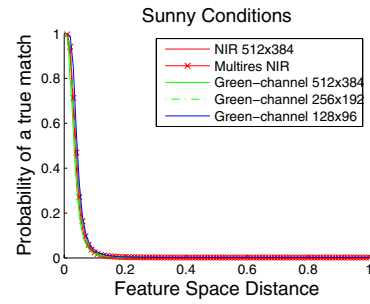


Fig. 2. Probability of correctness vs. feature-space distance for daytime runs collected during the field test described in section V. Shown are results for features computed on the green channel of images with different resolutions, and on the NIR channel at one resolution and at multiple resolutions simultaneously.

vehicle has left the path stored in the database is then decided by comparing two hypotheses:

- 1) The last k observations match locations in the map.
- 2) The last k observations do not match locations in the map.

A ratio of these probabilities is compared to a threshold α , which can be adjusted to encode information about the prior probability of being in the map (Equation 2).

$$\frac{\sum_{j=0}^k \log [\arg \max_{i \in M} P(S_{t-j} = i | D(x_{t-j}, DB^i))] }{\sum_{j=0}^k \log [1 - \arg \max_{i \in M} P(S_{t-j} = i | D(x_{t-j}, DB^i))] } > \alpha \quad (2)$$

For computational efficiency these probabilities are approximated using two assumptions: that the observations are independent, and that the probability that the closest match is correct is equal to the probability that the sensor observation was taken from a location in the map. For numerical stability a sum of logs is used instead of a product.

IV. SENSING AND COMPUTING UNIT

The SLAM In A Box (SIAB) unit was developed as a stand alone sensor and computing unit to provide the



Fig. 3. A SIAB unit mounted on a HMMWV during the field tests described in Section V. A special mounting plate was designed to hold the unit on the hood of the vehicle and provide shock absorption.

perception and data processing capabilities required for outdoor topological navigation as presented in Section III.

The sensing and computing components are encased in a sealed and shock isolated aluminum box. The unit occupies a volume of $20 \times 33 \times 31$ cm³, and draws approximately 70 W of power from a 24 V DC source.

A. Sensing

Figure 3 shows a SIAB unit mounted on the hood of a High-Mobility Multipurpose Wheeled Vehicle (HMMWV) during the extensive field tests described in Section V. Figure 4 presents a close up view of the unit. It includes an actuated scan line laser, two cameras and an inertial measurement unit (IMU). A Garmin 16A GPS unit is also present but only to collect ground truth data with 3 m accuracy. The scan line laser, a SICK LMS-291, provides 3-D scans of the environment as it is spun around the horizontal axis at 1 HZ. The scan line is sampled every $\frac{1}{2}$ degree for 180 degrees at 37.5 Hz per scan line. The field of view of the laser is thus the half hemisphere in front of the vehicle. The unit also includes a 1032×778 pixels, 90° field of view (FOV), Marlin color camera and a 1024×768 pixels, 90° FOV, Point Grey Flea camera¹. The Flea is used to sense Near-Infra-Red (NIR) by adding an 800 nm long pass filter, which combined with the response of the CCD gives a useful bandwidth of 800 to 1000 nm. Finally a 6-DOF IMU, a O-navi FalconGX, provides feedback on the SIAB's inertial motion at 75 Hz.

B. Computing

The SIAB unit houses a Mini-ITX motherboard with a 1.6 GHz Pentium M processor and a 40 GB hard drive for data logging and storage of the a priori data used for online localization.

The localization and data-logging software that runs on this computer is split into three processes. The first process controls the laser range finder, the IMU, and the GPS. The second process interfaces to both cameras and receives streamed images at 7.5 frames per second (fps). Both processes place the most recent sensor data in a shared memory buffer, and log the rest to a hard drive.

¹The cameras and the laser are not co-registered.

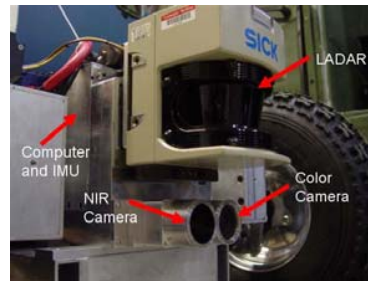


Fig. 4. A close-up view of the prototype SIAB unit. The laser and two cameras are visible. The computer is located in the housing behind the laser spinning mechanism.

Color images are stored as 110 KB JPEG compressed images and the NIR images as 768 KB uncompressed PGM images. The final process accesses data from the shared memory buffer and localizes the vehicle using the approach described in Section III.

C. Real-time localization

The latest camera images, stored in the shared memory buffer, are used for localization. Features vectors are extracted as explained in Section III and matched against the a priori data stored on the hard drive. To reduce the number of false alarms, the matching results are accumulated over time and a vote performed in order to either inform the user that the vehicle is on-course or to warn him that it might have veered off-course.

V. FIELD TEST RESULTS

A. Early validation

Our features and localization approach were validated by tests performed around the Carnegie Mellon campus and in a nearby park. The sensor pod was mounted on a roboticized John Deer e-Gator which provided high-quality wheel odometry and heading information. Five loops of approximately 1-mile each were recorded and used for off-line validation of the features described above as well as elimination of other possible features, such as color histograms, which proved early on to be much less effective.

B. Field test site

Extensive experiments were conducted in a large section of undeveloped land in central Pennsylvania containing a variety of natural terrains (open fields, wooded sections, ponds, meadows) traversed by a network of dirt and gravel roads. The sensor pod was mounted on the hood of a HMMWV, and driven around manually. Experiments consisted of driving 23 loops of approximately 3 km apiece both on-road and off-road. Figure 5 shows the paths of the different loops driven, and Table I contains information about the different runs by weather condition.

In total, 104,658 pairs of color and near-infrared images were captured in widely varying lighting and weather conditions over a two day period in December 2004. Day one was overcast with occasional snow flurries. Day two

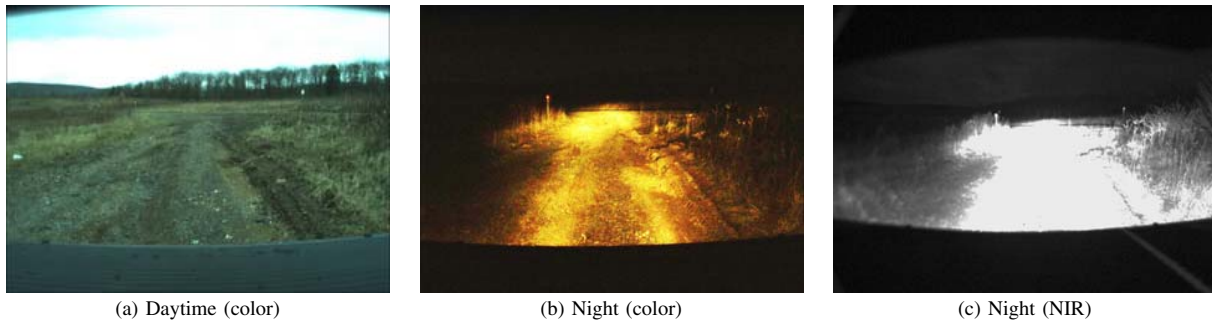


Fig. 6. Example images from the same scene for different sensors and illumination conditions. The images taken at night have been increased in brightness for visualization purposes.

TABLE I
STATISTICS FROM THE DIFFERENT RUNS PER WEATHER CONDITION

Weather	# of runs	Avg. # of images	Avg. path length
Sunny	10	4723	2.8 km
Overcast	8	4919	3.05 km
Dusk	2	4333	2.8 km
Night	3	4044	3.1 km

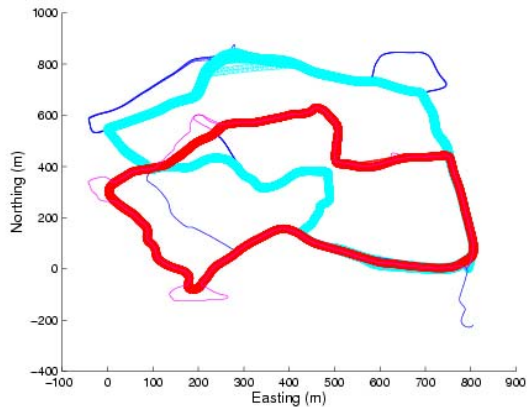


Fig. 5. Paths used for data collection and field testing in central Pennsylvania. Eight data-collection runs followed path #1 (light blue), six runs followed path #2 (dark red), and the other nine runs were either variations of path #1 (thin dark blue) or variations of path #2 (thin dark red).

was sunny and clear. On both days runs were collected at dusk and after nightfall to provide examples of extreme changes in lighting conditions. The only illumination used at night was the headlights of the HMMWV. For examples of night-time images see Figure 6. GPS readings were used as ground truth measurements.

C. Database creation

Each one of 23 loops represents approximately 5 GB of raw sensor data that is reduced to 3 MB of features using the method describe in Section III. For each loop, all of the images collected are placed in the database; no image selection is performed.

D. Off-line verification

1) *Performance Criteria:* For our testing we largely adopted the methodology presented in [16], which concen-

trates on the Receiver Operating Characteristics (ROC) of probability of target detection versus probability of false alarm generation. ROC curves are created by comparing test “query” descriptors against a library of “reference” descriptors computed from a separate set of sensor data. To perform this comparison we use the distance function $d(X_a, X_b)$ defined in Section III, which gives a scalar output for every pair of descriptors X_a and X_b . The query descriptor is said to match any reference descriptors in the library for which the feature space distance $d(X_q, X_r)$ is less than an arbitrary threshold δ (3). Thus, every query descriptor X_q returns a (possibly empty) set of reference descriptors M_F that are within δ of it in feature space, where δ is selected based on the relative costs of missed matches and false alarms for the current application. Testing all possible values of δ produces a ROC curve.

$$M_F = \{X_r | d(X_q, X_r) < \delta\} \quad (3)$$

We then compare the set of feature space matches against the set of reference descriptors that are actually within β meters and γ degrees in orientation of the query descriptor (as estimated from the GPS measurements for ground truth). The choice of β and γ is essentially arbitrary and should be motivated by the performance requirements of the intended task. For the results presented β was chosen to be 3 m, and γ was chosen to be 90° . The probability of detection is computed as the number of correct matches retrieved divided by the total number of correct matches (Equation 4). The false positive rate is the number of reference descriptors falsely classified as matches divided by the total number of reference descriptors examined (Equation 5). Since our intended application is topological localization in a large environment, a false match that is 10 m away from the true location is not as harmful as a false match that is 1 km away from the true location. To reflect this fact, matching reference descriptors that were between 3 m and 20 m away from the query descriptor were counted as neither matches nor false positives. The probability of detection, p_d , is defined as:

$$p_d = \frac{\# \text{ of correct matches retrieved}}{\# \text{ of matches}} \quad (4)$$

The probability of false alarms, p_{fa} is defined as:

$$p_{fa} = \frac{\text{\# of false matches retrieved}}{(\text{\# reference descriptors})(\text{\# queries})} \quad (5)$$

2) *Analysis:* The top row of Figure 7 shows the performance of the WGOH features described in Section III-B for sunny, overcast and nighttime conditions. The WGOH features show a significant tolerance to the resolution of the image, and the frequency of light used. Features computed from the green pixels on the color CCD's Bayer color filter (around 500 nm) show very similar performance to features computed from the NIR camera (800-1000 nm). Also, features computed from 512×384 pixel images show almost identical performance to features computed from 256×196 pixel images, although performance degradation does become significant for 128×98 pixel images.

The second row of Figure 7 displays the ROC curve computed for the WGOH features using only the closest match of each query. These graphs are indicative of the performance of the online system used in the tests described in Section V-E, which considered only the descriptor in the reference run that was closest to the query descriptor. This reference descriptor was then classified as a match, non-match, or uncertain based on its feature-space distance to the query. Under daytime conditions, two thirds of the possible matches are found with only a 5% false positive rate, and this false positive rate is further reduced in the online system by voting using a history of recent matches.

The third row of Figure 7 displays the probability that a pair of images which match in feature space were taken within X meters of each other, where X varies from one to 50. These figures were generated by selecting the feature distance threshold for each feature that detected at least 50% of the possible matching image pairs, and generating a cumulative histogram of the ground truth distance between the image pairs. It shows that the vast majority of localization errors observed in overcast conditions are not serious, and even the nighttime performance is significantly better than chance.

Large natural environments contain some regions which are visually very similar to others, resulting in false matches. For a typical pairs of images that are close in feature space, but far apart in the actual environment see Figure 8. Current work is underway to eliminate these false matches by using other sensor modalities, such as the onboard laser scanner and vehicle odometry.

The performance of the WGOH feature computed from the green channel is shown across a variety of weather condition in Figure 9. Unlike figure 7, which compared runs of the same weather condition, in this figure the query feature vectors are computed from a different weather condition as the reference feature vectors. Light snow presented very little difficulty for the WGOH features, as the snow was probably largely filtered out by the Gaussian smoothing applied to the image and regardless would probably not bias the gradients towards any particular orientation. Sunny vs cloudy runs provided a much greater challenge for the features as runs from the sunny day included the shadow of

the vehicle and occasional saturation of the cameras from staring into the sun. As expected, features created during the day did not compare accurately with features created at night, because of the extreme illumination changes.

E. On-line processing

The last day of testing also included two on-line trials to verify that the system can run in real-time. Current sensor observations were matched at 6.8 Hz against all 4897 feature vectors from a data-collection run along path #1, which had occurred on the afternoon of the previous day. Unfortunately, because of the high data rate of its sensors, the prototype sensor pod could not both log data and perform on-line localization, so evaluation of the system was limited to visual confirmation. During the two trials the system used a δ value of 0.06, and successfully detected four of four deviations from path #1, although it did report several transient false positives and false negatives, none of which lasted longer than a few seconds.

Most importantly, though, this test demonstrated that the approach used to generate off-line results could be implemented on current hardware and run — without significant optimization — at a rate sufficient to quickly (within three seconds for the system tested) warn a driver that the vehicle had deviated from the desired path. The most computationally intensive part of this approach, the creation of the image features, was accomplished in real-time on our Pentium-M based system with non-optimized code. The best all-around feature tested, the 16-region WGOH computed from the color camera, was computed at an average rate of 7.8 frame per second (fps).

The computational cost of exhaustively comparing each incoming feature vector with a set of feature vectors from almost 5000 images is insignificant next to the cost of computing the feature vector, since each image is reduced to only one feature vector and an efficient distance metric is used. For larger reference sets, however, several techniques are available to speed up feature vector comparison and improve its accuracy. Approximate nearest neighbor algorithms have shown great promise using similar feature vectors, and [12] reports a reduction in search time of approximately two orders of magnitude with a 5% loss in accuracy. The largest reductions, however, are likely to come from using the results of previous searches to limit the current search window to only images taken near the most probable location of the robot. Research is currently underway to use vehicle odometry measurements to both reduce the number of necessary comparisons and increase the reliability of the localization process by rejecting matches that would require unrealizable vehicle motions.

VI. CONCLUSIONS AND FUTURE WORK

This work presents results for real-time topological localization in large outdoor environments using weighted gradient orientation histogram (WGOH) features. These features prove to be discriminative enough to tell all but the most visually similar spots apart in a dataset covering over 100,000 images and 67 km of traverse through a

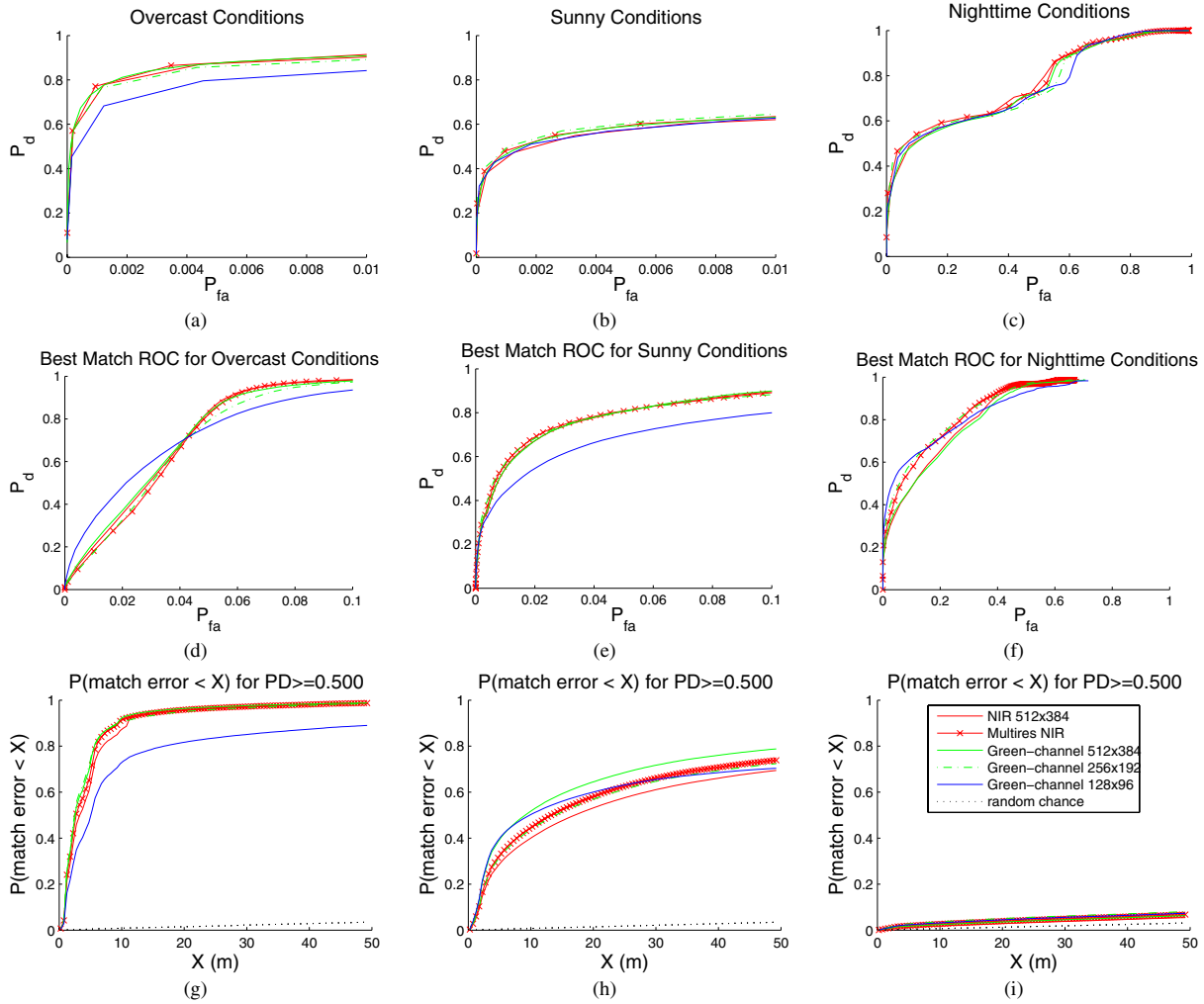


Fig. 7. Comparison of several different image features from runs with varying weather conditions. The leftmost column of plots gives results for overcast weather conditions, while the center and right columns give results for sunny and nighttime weather conditions respectively. The first row shows the ROC curves (as described in section V-D.1) as calculated over all possible pairs of images with the given weather conditions in the data set. Note the change in scale for figures c & f. The second row shows the ROC curves for the localization method used in the online-tests, which considered only the reference descriptor with the lowest distance from the query. The third row depicts the probability of a feature-space match being within X meters of the ground truth location, for a 0.5 probability of detection.



Fig. 8. A typical pair of images which are close in feature space, but far apart in the actual environment.

large outdoor environment. However, challenges remain in discriminating between visually similar locations and matching features across extreme illumination changes.

Our future work address these challenges by proceeding along three directions. First, we will increase performance

by adding a validation stage to check possible matches with a more computationally intensive, but hopefully more accurate, method such as SIFT feature matching. Second, work is underway to develop discriminative and computationally efficient features for the 3-D point data returned by the

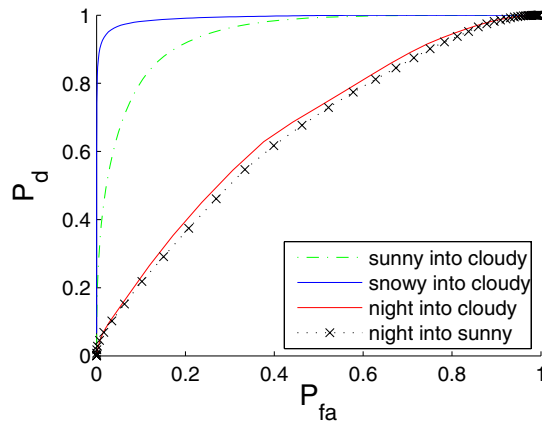


Fig. 9. ROC curves between different weather conditions

laser scanner, which should help nighttime performance significantly. Third, the temporal information in the history of best matches will be better exploited by building a topological map of the environment using odometry estimates, and localizing in the map with a Hidden Markov Model.

ACKNOWLEDGMENTS

The authors would like to thank General Dynamics Robotics Systems and the U.S. Army Tank-automotive and Armaments Command for their funding and operational support of this work. Christopher Baker, Zachary Omohundro and Christopher Atwood for the design and construction of the SIAB sensor pod. Anthony Stentz and Marc Zinck for their help in collecting data for preliminary testing and validation.

REFERENCES

- [1] H. Aoki, B. Schiele, and A. Pentland. Realtime personal positioning system for a wearable computer. In *International Symposium on Wearable Computers*, pages 37–41, 1999.
- [2] J.L. Crowley, F. Wallner, and B. Schiele. Position estimation using principal components of range data. *Robotics and Autonomous Systems*, 23(4):267–76, 1999.
- [3] Cristian Dima, Martial Hebert, and Anthony (Tony) Stentz. Enabling learning from large datasets: Applying active learning to mobile robotics. In *Proceedings of the IEEE International Conference on Robotics and Automation*, 2004.
- [4] G. Dudek and D. Jugessur. Robust place recognition using local appearance based methods. In *Proceedings of the IEEE International Conference on Robotics and Automation*, 2000.
- [5] J-J. Gonzalez-Barbosa and S. Lacroix. Rover localization in natural environments by indexing panoramic images. In *Proceedings of the IEEE International Conference on Robotics and Automation*, 2002.
- [6] F. Gustafsson and al. Particle filters for positioning, navigation, and tracking. *IEEE Transactions on Signal Processing*, 50(2):425–37, 2002.
- [7] J. Kosecka and F. Li. Vision based topological markov localization. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 1481– 1486, 2004.
- [8] J. Kosecka, L. Zhou, P. Barber, and Z. Duric. Qualitative image based localization in indoors environments. In *Proceedings of the IEEE International Conference on Pattern Recognition*, 2003.
- [9] B. Krose, R. Bunschoten, N. Vlassis, and Y. Motomura. Appearance based robot localization. In *Proceedings of the International Joint Conferences on Artificial Intelligence*, 1999.
- [10] B. Kuipers and P. Beeson. Bootstrap learning for place recognition. In *Proceedings of the National Conference on Artificial Intelligence*, pages 174–180, 2002.
- [11] David G. Lowe. Object recognition from local scale-invariant features. In *Proceedings of the International Conference on Computer Vision*, pages 1150–1157, 1999.
- [12] David G. Lowe. Distinctive image features from scale-invariant keypoints. *International Journal of Computer Vision*, 60(2):91–110, 2004.
- [13] L. Paletta, S. Frintrop, and J. Hertzberg. Robust localization using context in omnidirectional imaging. In *Proceedings of the IEEE International Conference on Robotics and Automation*, 2001.
- [14] G. Pass and R. Zabih. Comparing images using joint histograms. *ACM Journal of Multimedia Systems*, 7(3):234–40, 1999.
- [15] D. Radhakrishnan and I. Nourbakhsh. Topological robot localization by training a vision-based transition detector. In *Proceedings of the IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 468–73, 1999.
- [16] C. Schmid and K. Mikolajczyk. A performance evaluation of local descriptors. In *Proceedings of the International Conference on Pattern Recognition*, pages 257–263, 2003.
- [17] F. Stein and G. Medioni. Map-based localization using the panoramic horizon. *IEEE Transactions on Robotics and Automation*, 11(6), 1995.
- [18] Yataka Takeuchi and Martial Hebert. Finding images of landmarks in video sequences. In *Proceedings of the IEEE International Conference on Computer Vision and Pattern Recognition*, 1998.
- [19] A. Torralba, K. P. Murphy, W. T. Freeman, and M. A. Rubin. Context-based vision system for place and object recognition. In *Proceedings of the IEEE International Conference on Computer Vision*, 2003.
- [20] I. Ulrich and I. Nourbakhsh. Appearance-based place recognition for topological localization. In *Proceedings of the IEEE International Conference on Robotics and Automation*, pages 24–8, 2000.
- [21] N. Vandapel, R.R. Donamukkala, and M. Hebert. Experimental results in using aerial lidar data for mobile robot navigation. In *Proceedings of the International Conference on Field and Service Robotics*, 2003.
- [22] J. Wolf, W. Burgard, and H. Burkhardt. Robust vision-based localization for mobile robots using an image retrieval system based on invariant features. In *Proceedings of the IEEE International Conference on Robotics and Automation*, 2002.
- [23] Jiang Y. Zheng and Saburo Tsuji. Panoramic Representation for Route Recognition by a Mobile Robot. *International Journal of Computer Vision*, 9(1), 1992.