

# Monte Carlo Localization and Registration to Prior Data for Outdoor Navigation

David Silver and Anthony Stentz  
Carnegie Mellon University

**Abstract**—GPS has become the de facto standard for obtaining a global position estimate during outdoor autonomous navigation. However, GPS can become degraded due to occlusion or interference, to the detriment of autonomous performance. In addition, GPS positions must be aligned with prior data, a tedious and continual process. This work presents a solution to these two problems based on learning generic observation models in the presence of GPS to use in its absence. The models are non-parametric and compared to traditional approaches require few assumptions about either the prior data available or a robot's onboard sensors. Along with allowing for localization to prior data under GPS-denied conditions, this learning approach can be coupled with an EM procedure to automatically register GPS and prior data positions. Experimental results are presented based on data from more than 15 km of autonomous navigation through challenging outdoor terrain.

## I. INTRODUCTION

Autonomous navigation by a mobile robot generally consists of traversing to a defined goal. This naturally requires that the robot have a method of determining its position relative to said goal. In the realm of navigation through indoor or structured environments, this has encouraged a vast body of work in localization and SLAM. However, for outdoor navigation over long distances the determination of a robot's position in a globally consistent frame is generally left to the Global Positioning System (GPS).

While in general reliable GPS is not infallible: it requires reasonable line of sight to multiple locations in the sky, and can suffer significantly degraded performance around tall buildings or tree canopy. GPS is also susceptible to jamming (whether intentional or accidental). Finally, while low quality GPS systems have become sufficiently low cost to become almost standard in consumer vehicles, high quality systems remain expensive. In addition to providing the robot with the relative position of its goal, GPS is also used to index into any prior environmental data available to a robot during operation. Particularly for off-road navigation, prior data such as satellite imagery or other overhead sources have proven to be quite effective at increasing the safety and efficiency of autonomous navigation [1].

Achieving these gains requires that the robot be reasonably well registered to its prior data; that is, that the offset between a GPS position and the corresponding location in the prior data be known. While this offset is usually small (if the prior data was previously georeferenced) errors on the order of 5 to 10 meters are still common. Such errors are too large to allow for the most efficient use of prior data. Additionally, navigation goals are often chosen or specified with respect to the prior data, increasing the importance of

proper registration. Complicating matters further is the fact that the offset between prior data and GPS is not constant due to GPS drift, which can necessitate re-registering to prior data every few days. Finally, orthorectification errors can require different alignments to different regions of prior data.

Thus, outdoor autonomous navigation is faced with two core problems related to the use of GPS: continuing operation by bounding global position error in the face of degraded or absent GPS, and aligning GPS to prior data in the face of suboptimal georeferencing and GPS drift. This paper addresses these two problems through the use of Monte Carlo Localization (MCL) [2]. The partial availability of GPS data is exploited to allow construction of non-parametric observation models by learning to correlate whatever channels of prior and onboard sensor data are available. In the absence of an accurate initial registration, an Expectation Maximization (EM) based approach allows for the estimation of the offset between the GPS and prior coordinate frames and its refinement during autonomous operation. Experimental results are presented based on recorded data from the Crusher autonomous vehicle [3], [4] (Figure 1) over 15 km of autonomous traverse through challenging off-road terrain.

## II. RELATED WORK

Localization for mobile robots has historically received more attention with respect to indoor or well structured environments. Especially when operating indoors with small robots, short range sensors, and less expensive hardware it can not be assumed that a robot will be provided with a clear measure of its position in a globally consistent frame; instead such robots are generally tasked with bounding position by aligning themselves to a prior map of their environment, or creating one on the fly (SLAM).

In this context, Monte Carlo Localization (MCL) [2] has become quite popular for determining a robot's position based on a prior map of the environment. Application of MCL in such environments is typically based on range sensors such as LiDAR or Sonar, as well as a building map or schematic of the 2D structure of the environment. MCL works well in such circumstances partially because the behavior of range sensors under structured conditions is well studied and understood, and sufficient observation models can be constructed with relative ease. The endpoint model [5] (Figure 2) is one popular example, which approximates the probability of various range readings given the ground truth range (derived from a prior map) and a set of noise parameters.

The challenge of constructing an accurate observation model is one of the primary barriers to the application of MCL in more unstructured environments. Indoor sensors and prior maps are well approximated in 2D, allowing for the easy prediction of what range measurements a sensor should report under ideal conditions. In contrast, in outdoor environments it is more difficult to model and predict what readings an ideal sensor should produce conditioned on a specific pose and prior map. This problem is exacerbated when the prior data was not produced by the same class of sensors as the robot's onboard perceptual system (e.g. onboard range sensors versus prior satellite imagery). For very large outdoor environments, this is almost always the case. In addition, the size of such environments precludes human engineering to add a sufficient number of easily distinguishable landmarks. Therefore, in general localization in unstructured outdoor environments requires an observation model capable of relating heterogeneous data.

Recent work has approached this problem by constructing observation models under specific assumptions about the environment and a robot's onboard sensors. For example, [6] fits edges and curves to range data to construct landmarks under the assumption of the presence of man-made structures. Similarly, [7] detects roads onboard the robot and correlates them to a prior road network definition. There has also been recent work to localize directly to satellite data. In [8] overhead imagery is segmented into categories corresponding to freespace or a tall obstacle, and then uses a range based observation model similar to those used for indoor localization. A similar approach is applied in [9] but with contours in the imagery as a proxy for obstacle boundaries. In [10], the conversion of both prior and perceptual data into navigation costs is used as the basis of the observation model, while in [11] prior elevation models are compared to those generated online. A common property of these approaches are the specific assumptions about either the type of prior data available, the robot's sensors, or the type of environment; these assumptions preclude easy generalization across different robotic platforms or operating domains.

One way to improve the generalization of such approaches is to learn an observation model for the specific environment. [12], [13] seek to learn an observation model over an entire range scan (as opposed to an individual range reading) and exploit the correlations within such a percept. In contrast, [14] seeks to learn rather than tune the parameters of the standard endpoint LiDAR model. However, these approaches have so far remained limited to working with 2D range sensors for operating in indoor or well structured environments.

In contrast, this work seeks a general approach that can learn an observation model between generic prior data and a generic perception system. In addition, the approach should work regardless of the type of outdoor environment (e.g. urban, rural, off-road) as long as there is some relationship between the two data sources (i.e. they are based on observations of the mostly the same structures). Fortunately, there are some aspects of this specific problem that make it more tractable. First, it does not require map building or SLAM; the existence of a prior, sufficiently consistent map

is assumed. Instead, the task is only to bound the error with respect to global position, and register an external estimate of global position to the prior data. Second, such a solution need only augment rather than fully replace GPS for it to be immediately effective and useful; therefore it can take advantage of GPS information when available. The next two sections describe the proposed solution to this challenge.

### III. LOCALIZATION TO PRIOR DATA

This section first provides a brief overview of the MCL algorithm. It is by no means intended to be complete; for a thorough and detailed description see [2], [5].

Define the pose of a mobile robot at time  $t$  as  $X_t$ . As this work is interested in 2D localization and registration, we consider  $X_t = (x_t, y_t, \theta_t)$ .  $u_t$  is defined as a control input (the robot's commanded motion), and  $Z_t$  as a set of sensor observations.  $M$  represents a (static) prior map, with which sensor observations can potentially be correlated.

MCL estimates a probability distribution over  $X_t$  through the use of a particle filter: a set of samples  $X_t^i$  from  $X_t$  and associated weights  $w^i$ . At each timestep, the particles are advanced forward in time (the prediction step) by sampling from the motion model, defined as

$$P(X_t|X_{t-1}, u_t) \quad (1)$$

Next, the particles are reweighted (the correction step) based on the observation model, defined as

$$P(Z_t|X_t, M) \quad (2)$$

The observation model defines the probability of producing a specific sensor reading, conditioned on the robot's position and the prior map. Through the use of Bayes rule, the likelihood of each particle's pose hypothesis conditioned on the current sensor observation can be computed as

$$P(\bar{X}_t|Z_t, M) \sim P(Z_t|X_t, M)P(X_t) \quad (3)$$

This leads to a simple update rule for each particle's weight

$$\bar{w}^i = w^i * P(Z_t|X_t^i, M) \quad (4)$$

Occasionally, the particles are resampled (with replacement) with probability proportional to their weights, and the weights reset. Through this importance sampling procedure, the particle filter is able to maintain an estimate of  $P(X_t)$  in a non-parametric manner. Figure 4 shows an example of MCL in action.

With this basic algorithm in place, the core challenge in applying MCL is the construction of accurate motion and observation models for a specific domain. The construction of motion models for mobile robots is a challenging and well studied problem unto itself. Deriving such models for outdoor vehicles is well outside the scope of this work; however as it is a requirement for accurate localization it deserves a brief mention.

The study of motion models for indoor mobile robots has focused mostly on various models of odometry. It is well understood how such models should evolve in the absence of measurement noise. The key therefore is proper modeling of such noise and other errors, in order to understand how the



Fig. 1. **Left:** The Crusher autonomous navigation platform. **Right:** a 15km autonomous traverse through off-road terrain by Crusher, overlaid on the satellite imagery used as prior data in these experiments. Green represents the GPS recorded path. Blue and red represent a simulated odometry (low accuracy) and MCL corrected path, respectively, from a single experiment (see Section V). The brightness of the red path represents the particle filter’s uncertainty as to its own position estimate; the uncertainty estimates correlate well to the actual offset from the GPS path. The simulated odometry path finishes over 250m away from the final GPS position, while the MCL path finishes only 3m away. Quickbird imagery courtesy of Digital Globe, inc.

model’s uncertainty should evolve over time. There is far too much work in this area to properly mention here; however the approaches of [14] and [15] deserve mention through their use of an EM approach to learning noise parameters of odometry models (similar to the approach described in Section IV).

Motion models for outdoor robotic vehicles are generally more complex, as such vehicles often operate at speeds where dynamics can no longer be ignored. However, as a consequence such vehicles are generally outfitted with more proprioceptive sensing to aid in estimating local motion: along with odometry inertial units and gyros are quite common. This additional sensing generally allows outdoor robots to travel distances of tens or hundreds of meters while still maintaining a position estimate that is reasonable for local navigation. In addition, this higher level of local accuracy, especially when fused with GPS information, provides an accurate and bounded position estimate that is quite useful for learning a motion model through observation of vehicle’s response to various control inputs. However, this also serves as a reminder of just how much outdoor robots can come to depend on GPS or some other source of bounded global position information.

The remainder of this paper will not consider motion models in depth, but instead will assume that a reasonable model can be constructed for the robot in question. For the Crusher vehicle used in these experiments, see [16] for an approach to learning the forward motion model.

The remaining challenge to applying MCL to outdoor localization is the construction of the observation model. Rather than construct a specific model based on assumptions about the nature of  $Z_t$  and  $M$ , this work considers generic prior data and sensor updates. The only requirement for the prior data is that for any location  $(x, y)$  in the environment, the prior map can provide some description of the environment; that is  $M^{x,y} = f_p$  for some  $f_p \in \mathcal{F}_p = \mathbb{R}^n$ . Equivalently, each sensor observation  $Z_t$  is considered as a set of individual observation  $z_t$  at locations  $(x_r, y_r)$  relative to the current vehicle position; that is  $z_t^{x_r, y_r} = f_s$  for some  $f_s \in \mathcal{F}_s = \mathbb{R}^m$ . This requirement is sufficiently general to

allow use of many different sensor modalities (e.g. LiDAR, RADAR, stereo, etc.) Environment dynamics, when properly modeled and understood, can also be represented in this fashion (or simply ignored). The only assumption about the data itself is that when properly aligned there is some relationship between a corresponding  $f_p$  and  $f_s$ . However, there are no additional constraints (e.g. linearity) on the actual form of this relationship.

Initially, consider the case where  $n = m = 1$ ; that is the prior and sensor updates each consist of a single feature at each location in their respective frames. The observation model can now be defined as

$$\begin{aligned} P(z_t^{x_r, y_r} = f_s | M^{x_w, y_w} = f_p, X_t) \\ x_w = x_t + x_r \cos \theta_t - y_r \sin \theta_t \\ y_w = y_t + x_r \sin \theta_t + y_r \cos \theta_t \end{aligned} \quad (5)$$

Simply stated, the observation model expresses the probability of a local sensor observation  $f_s$ , given that under the current pose hypothesis  $X_t$  the local observation would correspond to  $f_p$  in the prior map. Factoring out  $X_t$  separates uncertainty related to the robots position and uncertainty related to the individual measurements.

$$P(z_t^{x_w, y_w} = f_s | M^{x_w, y_w} = f_p) P(X_t) \quad (6)$$

This equation clearly states what is required to construct a generic observation model: it must be capable of predicting with what probability the robot will observe  $f_s$  when the corresponding prior observation is  $f_p$ .

This leads to the core of the proposed approach: under the assumption that GPS position, when available, represents near ground truth, the relationship between  $f_s$  and  $f_p$  can be directly observed and used to build a model of  $P(z_t | M)$  through experience. That is, every sensor and prior observation pair  $(f_s, f_p)$  can be recorded, and a model relating these two quantities constructed. Then, in the future absence of GPS, this model can be used for localization purposes. While there may not always be a direct relationship between how terrain appears to a ground sensor and in a prior map, recent work [17] has shown that in practice these observations will

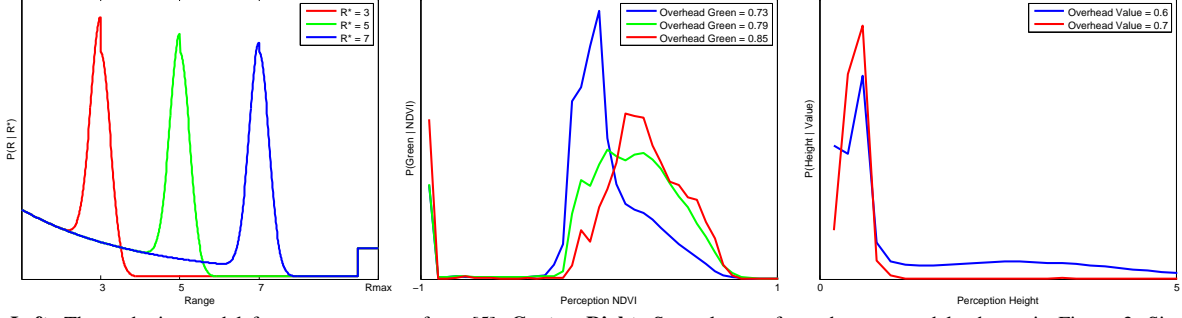


Fig. 2. **Left:** The endpoint model for a range sensor from [5]. **Center, Right:** Several rows from the two models shown in Figure 3. Similar to the endpoint model, the learned models show a clear region of maximum likelihood, along with varying noise characteristics observed empirically.

be sufficiently coupled to learn an effective model relating them. Therefore, consider a simple algorithm for training an observation model in the presence of GPS

- As the robot traverses, align local sensor readings  $f_s$  with prior measurements  $f_p$  according to GPS.
- Discretize  $f_s$  and  $f_p$  at some resolution over the feature spaces  $\mathcal{F}_s$  and  $\mathcal{F}_p$  into  $\bar{f}_s$  and  $\bar{f}_p$ . For every possible  $\bar{f}_s, \bar{f}_p$  pair, maintain a count  $C_{\bar{s}, \bar{p}}$  of how many times that pair is observed
- When needed, compute  $P(z_t|M)$  as

$$P(z_t^{x_r, y_r} = f_s | M^{x_w, y_w} = f_p) = \frac{C_{\bar{s}, \bar{p}}}{\sum_{s', \bar{p}} C_{s', \bar{p}}} \quad (7)$$

that is, normalize across all observed instances of  $\bar{f}_p$ .

When a robot is receiving high quality GPS information, it can continue to record observation pairs and construct a model for future use. When GPS drops out or becomes degraded, MCL can begin from the last known GPS position, using the learned observation model to correct the motion model. Examples of learned observation models are shown in Figures 2 and 3.

One final caveat is the issue of regularizing the observation model. It is a well known problem [13] that if an observation model is too peaked over certain sensor readings, MCL can perform poorly and lose track of the correct position. A common approach is to smooth or regularize the model by mixing it with a uniform probability over all possible observations. In this specific case, it is explicitly necessary to account for the possibility of observation pairs that were not seen during training.

The algorithm as described assumes both sensor and prior observations consist of a single real number. However, this algorithm can easily be extended to multidimensional readings in two ways. The first approach would be to try and model a full joint distribution over all dimensions; instead of maintaining frequency counts over 2 dimensional bins,  $m+n$  dimensional bins would be necessary. Such an approach would very quickly fail to generalize as the dimensionality increases. Therefore, a more robust approach involves a naive Bayes assumption about the independence of the various dimensions of the feature spaces

$$P((f_{s_1}, \dots, f_{s_m}) | (f_{p_1}, \dots, f_{p_n})) \sim$$

$$\begin{aligned} & P(f_{s_1} | f_{p_1}) * \dots * P(f_{s_m} | f_{p_1}) * \\ & \vdots \quad \ddots \quad \vdots \\ & * P(f_{s_1} | f_{p_n}) * \dots * P(f_{s_m} | f_{p_n}) \end{aligned} \quad (8)$$

This approach would require  $m \times n$  separate models, although each model would be easier to train. In practice, it is expected that expert intuition and experience, as well as empirical evaluation, could identify the most useful subset of correspondences. In our own experiments, we have found that even a single dimension of observations is sufficient to perform accurate localization (see Section V).

#### IV. REGISTRATION TO PRIOR DATA

The algorithm for learning an observation model as just described made two fundamental assumptions about GPS positions provided during training: that there was no error, and that the GPS was already properly registered to the prior information. This section relaxes both of these assumptions and provides an algorithm that can register GPS to prior data, even in the absence of any existing observation model.

GPS, like any other positioning system, provides only an uncertain estimate. With perhaps the exception of systems using differential corrections (which requires additional infrastructure) this uncertainty will generally be on the order of several meters. The use of augmentation systems such as WAAS and the filtering of GPS and inertial sensors can reduce this uncertainty, but in general it will still be greater than the resolution of local sensor information. Therefore, it must be accounted for when learning an observation model.

Fortunately, this uncertainty can be easily incorporated. If at time  $t$ , the GPS position estimate is a distribution  $P(X_t|GPS)$ , (6) becomes

$$P(z_t^{x_w, y_w} = f_s | M^{x_w, y_w} = f_p) P(X_t|GPS) \quad (9)$$

Each potential  $X_t$  implies a different pairing of  $f_s$  and  $f_p$ , and a different  $C_{\bar{s}, \bar{p}}$  whose count should be incremented. Rather than simply increment  $C_{\bar{s}, \bar{p}}$  for the  $f_s, f_p$  implied by the maximum likelihood estimate of  $X_t$ , for each possible  $X_t$ , the corresponding  $C_{\bar{s}, \bar{p}}$  can be incremented by  $P(X_t|GPS)$  to reflect that there is uncertainty in the correlation of  $f_s$  and  $f_p$ . While  $P(X_t|GPS)$  may be nonzero (or rather greater than some small  $\epsilon$ ) for an infinite number of estimates of  $X_t$ , in practice there are only a fixed number of possible alignments, due to discretization of local sensor and prior measurements.

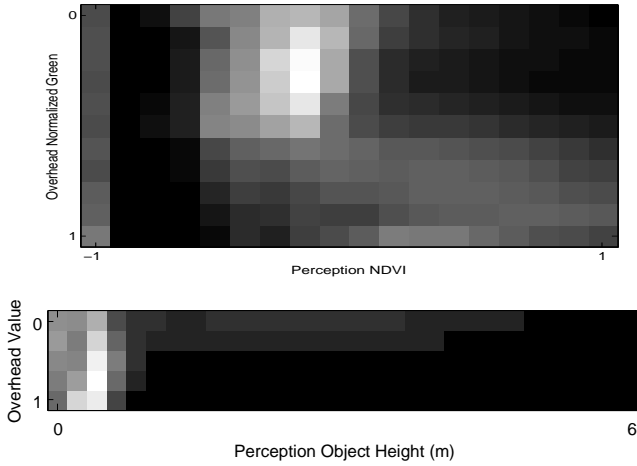


Fig. 3. Two of the 6 Learned Observation Models. Brighter pixels indicate higher probability of an observation. In both examples, the model captures the non-linear correlations between the prior and onboard observations.

This same basic approach can also be used to train a model in the presence of registration errors. Registration errors between GPS and prior positions are inevitable due to a number of factors, such as errors in georeferencing or orthorectification and the use of different projections. In addition, due to GPS drift over time this error is not constant, and even if corrected once can grow slowly over several days.

Assume that the registration error is unknown but upper bounded by some  $R_{max}$ . This is a reasonable assumption, as even a single landmark correspondence or tie point can limit error to within one or two dozen meters. Without additional information, and given a maximum likelihood GPS position, a new distribution  $P(X_t|GPS, R_{max})$  can be constructed, that combines not only the GPS uncertainty, but a uniform distribution over the registration error between 0 and  $R_{max}$ .

$$P(z_t^{xw, yw} = f_s | M^{xw, yw} = f_p) P(X_t | GPS, R_{max}) \quad (10)$$

This new distribution can then be used when incrementing the various  $C_{s, \bar{p}}$  during training of an observation model, to reflect both sources of uncertainty in the sensor to prior correspondences.

The observation model learned in such a manner will have been trained both on correct and incorrect sensor to prior correspondences. However over time the incorrect correspondences are likely to occur with essentially a uniform frequency (due to the uniform distribution over registration errors). In contrast, assuming any spatial dependence whatsoever amongst both sensor and prior features, correct (or near correct) correspondences should occur with slightly higher frequency. The result is that the resulting model will still have signal, if highly regularized.

This model can then be used to register to prior data through the use of MCL.  $P(X_t)$  is initialized to  $P(X_t|GPS, R_{max})$ . Instead of sampling from the normal motion model, relative updates in GPS position are used to project the filter forward in time (with just enough noise to account for sampling error). The correction step is then applied as before, using the highly regularized observation model. Since the prediction step will be adding little to no additional noise, over time the filter will converge. The dif-

ference between the filter and GPS global position estimates represents a hypothesis as to the current registration error.

If  $R_{max}$  is small (on the order of a few meters), then the observation model learned in this manner is likely sufficient to also localize in the presence of noise in the forward model (due to none or poor GPS information). However, if  $R_{max}$  is quite large, then such a highly regularized model will likely be unable to overcome such uncertainty. Furthermore, it probably is not capable of producing a sufficiently certain registration estimate to assume that all registration error has been accounted for. However, even in such cases, the registration estimate will be more accurate than  $R_{max}$ , and the actual uncertainty will be reflected in the final distribution over  $P(X_t)$ . Thus, the same process could be repeated, but starting with a smaller value for  $R_{max}$ , and lead to another improvement in registration.

This leads to the final algorithm for registering GPS information to the prior data. The iterative approach is based on the well known Expectation Maximization(EM) [18] algorithm, and is quite similar to the approaches of [15], [14] for learning parameters of motion and observation models. There are also similarities with mutual information based approaches used in medical image registration [19]. A key difference is that the proposed registration approach explicitly learns a non-parametric observation model, and thus is both more generic and not as sensitive to initial parameter values. In fact, the only parameter it requires initially is  $R_{max}$ , which can be a very loose upper bound.

The full registration algorithm proceeds as follows.

- **E-Step:** Starting with an initial  $R_{max}$ , learn an observation model according to (10) and (7)
- **M-Step:** Based on the learned observation model, perform MCL for some period of time using relative GPS updates in place of the forward model. The difference between the maximum likelihood positions of  $P(X_t|Z_0, \dots, Z_t)$  and  $P(X_t|GPS)$  represents the current registration offset, and the uncertainty of  $P(X_t|Z_0, \dots, Z_t)$  can be used to produce a new upper bound  $R'_{max}$ .
- **Iterate:** Replace  $R_{max}$  with  $R'_{max}$ , update GPS positions based on the estimated registration offset, and then repeat the above process until convergence.

As long as the current observation model is able to account for the small amount of noise added from the GPS based forward model,  $R'_{max}$  will be less than  $R_{max}$  and this process will converge to a final registration estimate. Additionally, the above iteration can be performed as the robot continues to drive, or repeatedly over the same sequence of sensor data. With the latter approach, a sequence of only a few minutes can be sufficient to register the robot.

Rather than use the distribution over possible registrations after an M-step at the next E-step, the proposed algorithm condenses this information into a single registration hypothesis and a corresponding  $R'_{max}$ . This choice of a hard EM approach over a more traditional (soft) approach was made for two reasons. The first is computational, as it dramatically speeds up the computation of the observation model during the E-Step. The second is practical, as empirically the

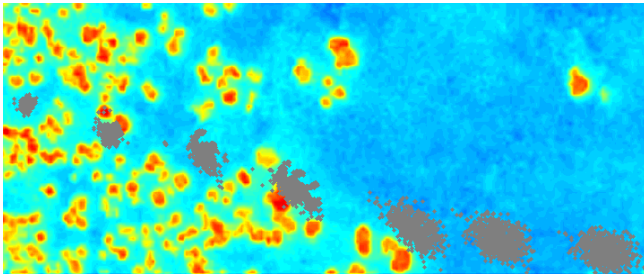


Fig. 4. Timelapse of MCL in action, shown against prior NDVI. After leaving a large open area with few features to localize from, the filter has an error of 11.5m from ground truth, and a  $\sigma$  of 10m. 2 minutes later, after traversing into a small forest the filter has reduced its error to 3m and  $\sigma$  to 2.75m. This sequence corresponds to km 2.9 to 3.4 in Figure 5.

distribution after each M-Step over registrations was found to be well approximated by a uniform distribution.

## V. EXPERIMENTAL RESULTS

Experiments were performed in simulation based on data logs from the Crusher autonomous vehicle [3], [4] (Figure 1). Crusher’s onboard sensing consists 3D LiDAR scanners and multispectral cameras, to provide both geometric and appearance data. For these experiments, two specific processed sensor features were used: The height of obstacles from the ground plane, and the Normalized Difference Vegetation Index (NDVI) of each location. NDVI values are related to the amount of chlorophyll present in a structure, and thus are useful for the detection of vegetation in many fields, including robotic perception [20]. Features were produced at 20cm resolution out to a range of 20m.

Datalogs were provided for a 15km autonomous run through difficult unstructured terrain, shown in Figure 1. Prior data for the environment consisted of 2.4m resolution multispectral satellite imagery (R,G,B,NIR) which was up-sampled and smoothed to 60 cm. The terrain features visible in the prior data consist of hills, trails, open ground, tall grass, bushes, ditches, and tree canopy. From the upsampled data, 3 prior features were computed: NDVI, normalized green, and the image intensity (i.e. the value component of HSV). Figure 6 shows an example of the 3 prior features and 2 sensor features from the same terrain location.

Experiments in using learned observation models for GPS-denied localization made use of the entire 15 km run. The first 15 minutes of the run (approximately 3 km) were used to train an observation model. Then, GPS dropout was simulated, and MCL was used to maintain Crusher’s position for the remaining 60 minutes and 12 km of the run (The starting location is evident in Figure 1). Instead of using Crusher’s raw local position information as the forward model input, a simulated local pose system was used. This was done to allow comparison of the models under different levels of local pose accuracy. Two simulated local pose systems were used: a high accuracy system with approximately  $1 \frac{\circ}{hr}$  heading drift and low slip, and a low accuracy system with  $5 \frac{\circ}{hr}$  drift and high slip. For each system, an observation model was trained for each sensor to prior feature combination described above. Figures 2 and 3 show examples from 2 of these models. In total, this resulted in 12 sets of experiments, each run 5 times (the same

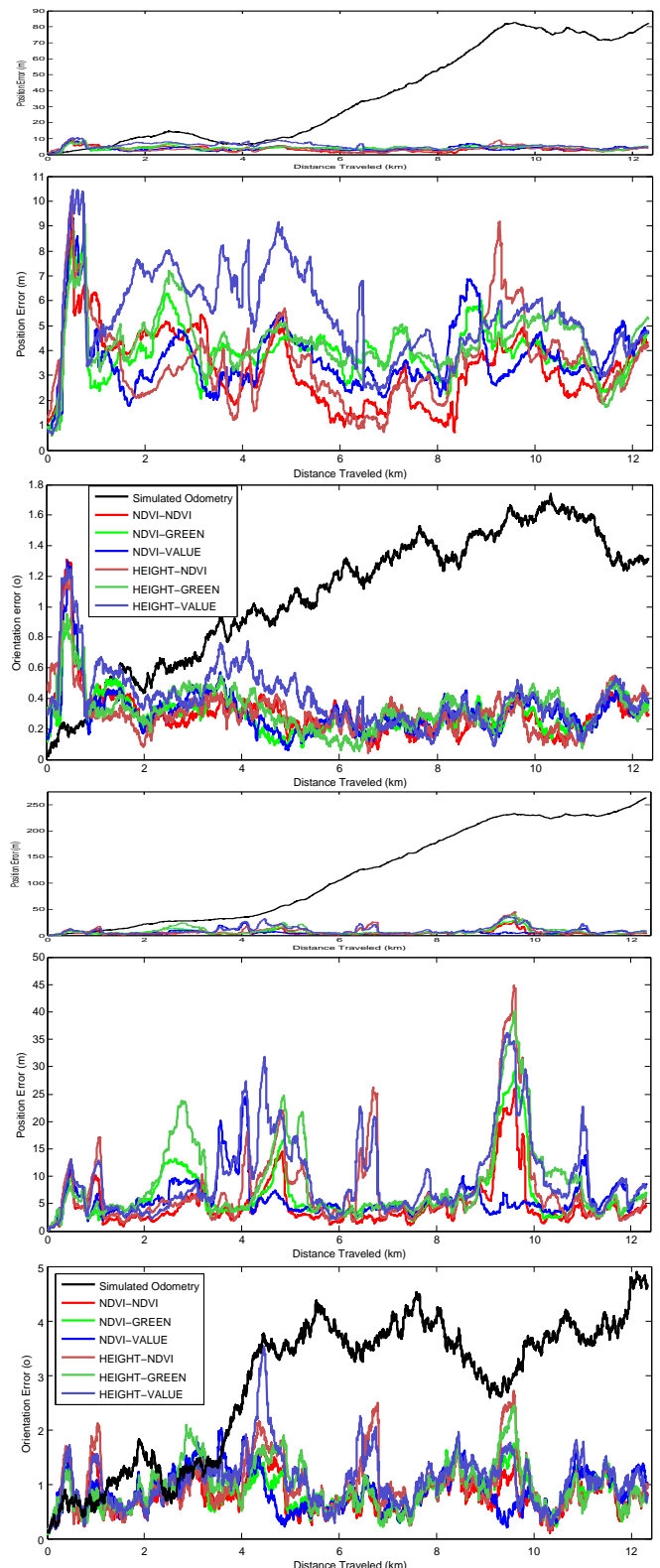


Fig. 5. Results for the GPS-denied localization experiment. **Top:** simulating a high end odometry system and using 25 particles to localize. **Bottom:** simulating a low end odometry system and using 1000 particles to localize. Position error is shown with and without odometry to provide scale. Odometry position and orientation error grows unbounded over time. In contrast, the particle filter using each of 6 different learned observation models is able to bound its error. Additionally, when temporarily unable to converge due to a feature sparse region of the environment, the filter maintains an accurate measure of its uncertainty, and recovers in the future.

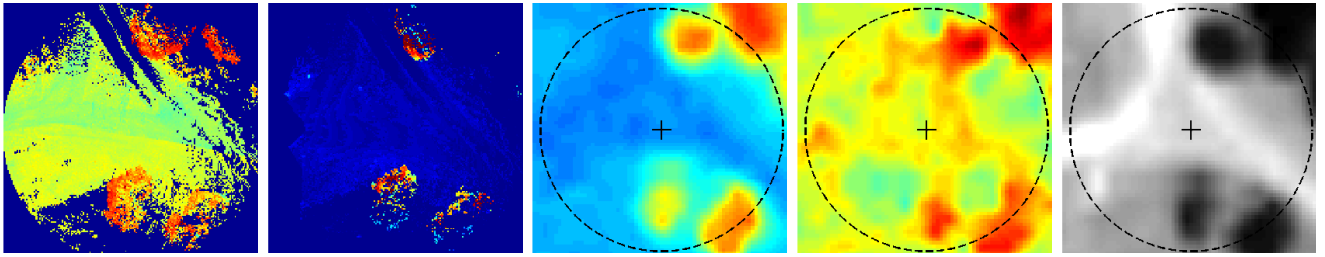


Fig. 6. The 2 Sensor and 3 Prior Features used in Crusher experiments. From left to right: object height, NDVI (onboard), NDVI (prior), Green, Value. All features are properly registered. The dashed circle in the prior images represents the robot's 20m perception range. Terrain features present include tree canopy, tall grass, and dirt trails.

simulated odometry was used in each group of experiments over different feature pairs).

Figure 5 shows the position and heading error of each system with respect to the ground truth GPS position (data that was not available to the filter during simulation). In the case of the high accuracy local pose solution, the odometry slowly drifts over time. By the end of the 12 km of GPS-denied travel, the odometry is off by nearly 100m. In contrast, all 6 of the different feature pairs were sufficient for robust localization: despite some early struggles, even the worst system stays within 10m of the correct position 99% of the time. This is more impressive considering that only 25 particles were used in these experiments (to demonstrate the robustness of the system).

In the case of the low accuracy pose solution, the odometry drifts much faster, and is off by nearly 300m after 12km. Despite this increased error, all of the various feature pairs and their corresponding learned observation models are sufficient to keep the vehicle localized (see Table I for summary statistics). Although at times errors can accrue of tens of meters, the filters always maintain a proper model of their own uncertainty, and therefore are able to robustly recover once presented with sufficient information to localize. Maintaining such an uncertain distribution does require more particles (1000 was used for this set of experiments).

Figure 4 provides an example scenario of these experiments. Crusher has just traveled through a 5 minute stretch of completely open terrain, with almost no discernible features in the prior data (the bottom right section of the path in Figure 1). As a result, there is high uncertainty regarding the correct position. However, once distinguishing features are encountered, MCL is able to use the learned model to reduce both the filter's error and uncertainty. A similar situation occurs starting at km 9, where the environment is temporarily confusing and the filter actually starts modeling a bimodal distribution (the bottom left section of Figure 1); over time one of the hypothesis is contradicted by local sensor data and the filter recovers. Important in both instances is the filter's own measure of its uncertainty (the brightness of the red path in Figure 1), which grows in confusing or feature-poor areas and then shrinks in feature-rich areas. This recovery from large errors also shows that this approach would be suitable for solving more global localization problems (although such problems are not commonly addressed in outdoor robotics).

It should be noted that although these experiments were offline, they were performed on real data with numerous practical issues. For example, the prior data was 4 years old

System	Avg Error(m,°)	% < 5,10,20 m	% < 1, 2 °
<b>Odometry</b>	<b>119.0, 2.86</b>	<b>0.05, 0.09, 0.15</b>	<b>0.13, 0.29</b>
NDVI - NDVI	4.47, 0.78	0.76, 0.93, 0.98	0.77, 0.99
NDVI - GRN	6.30, 0.83	0.59, 0.87, 0.96	0.71, 0.99
NDVI - VAL	6.01, 0.89	0.52, 0.91, 0.98	0.63, 0.99
HGHT - NDVI	7.47, 0.98	0.49, 0.79, 0.93	0.62, 0.95
HGHT - GRN	8.46, 0.94	0.46, 0.74, 0.91	0.61, 0.99
HGHT - VAL	9.72, 1.09	0.26, 0.65, 0.89	0.43, 0.97

TABLE I

SUMMARY STATISTICS FOR THE LOW ACCURACY ODOMETRY SYSTEMS IN FIGURE 5. ALONG WITH MEAN ERROR, THE PERCENTAGE OF TIME BELOW FIXED THRESHOLDS IS PROVIDED.

when the run was performed and sensor data recorded, and there were numerous inconsistencies where the environment had changed. There is significant noise in the features, especially in the onboard NDVI when Crusher drives directly at the setting sun for several minutes. Perhaps most importantly, Crusher was navigating using the prior data as in [1], [4], and therefore was actively avoiding the complex areas that are useful for localization in favor of open areas that are most difficult. Additionally, 2.4m satellite data is relatively low resolution, as 40cm data is now commercial available and aerial resolutions are even higher.

The above experiments assumed that the GPS position had already been properly registered. This was accomplished through the use of nearly 300 hand chosen tie points between the prior and sensor data. In actual operation of autonomous vehicles, such a tedious process would only provide an initial registration; over time additional work would be necessary to account for GPS drift. Therefore, automatic registration would be of enormous operational value.

The registration algorithm from Section IV was tested by repeatedly registering in the presence of an artificially induced bias. A 5 minute section of sensor data was used; each EM iteration played back the same data. GPS positions were used directly in place of the simulated local pose system of the previous experiments; however there was still WAAS error of approximately 1m to contend with. Four sets of experiments were performed. The best and worst feature pairs from the localization experiments were used (NDVI-NDVI and Height-Value), with registration errors between 5-10m ( $R_{max} = 10$ ), and between 10-20m ( $R_{max} = 20$ ).

Figure 7 shows a single example of the registration process: beginning with a significant amount of initial error, the registration is quickly corrected; as a result the vehicle's path properly aligns with the prior data. The results for all experiments are shown in Figure 8. For all experiments, the registration offset quickly converges to the correct position.

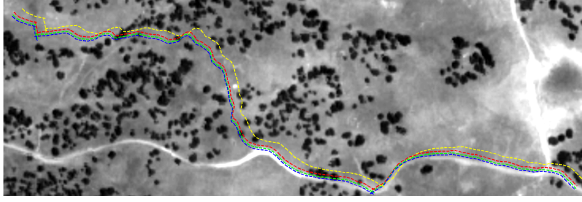


Fig. 7. The path of the robot in a single registration experiment, after 0,1,2 and 5 EM iterations (yellow, red, green, and blue respectively). More than 18 m of initial error is eventually reduced to less than 1 m by the registration algorithm.

Regardless of the initial error, the final registration averages 40-50cm of error, which is within the underlying prior data resolution. The final distributions all reject the null hypothesis of  $\mu \geq 1m$  for  $p = 0.05$ . Across different feature pairs, there is no statistically significant difference in the mean registration error, although there is in the variance for  $p = 0.1$ . This indicates that even the worst feature pair has high performance on average, although it may be slightly more susceptible to local minima.

## VI. CONCLUSION

This work presents a solution to the dual problems of localizing and registering to generic prior data for an outdoor mobile robot. A procedure is presented to learn an observation model relating heterogeneous sets of prior and onboard sensor information that can be used in standard localization algorithms such as MCL. An EM-based algorithm is also presented that can automatically detect and correct for large errors in the registration of GPS and prior locations, as long as the error can be initially bounded. Combined, these two approaches can improve the robustness and efficiency of autonomous navigation in outdoor environments. Future work will focus on the online and real time application of these approaches to automatic registration and GPS denied localization for autonomous navigation. In addition, the suitability of these techniques will be further investigated for use with low cost and low accuracy GPS systems. Finally, the possibility of training such approaches in the complete absence of any GPS information will be explored.

This work was conducted through collaborative participation in the Robotics Consortium sponsored by the U.S. Army Research Laboratory under the Collaborative Technology Alliance Program, Cooperative Agreement W911NF-10-2-0016, extending preliminary work sponsored by Cooperative Agreement DAAD19-01-2-0012. The data used in this work was provided from the DARPA sponsored Unmanned Ground Combat Vehicle - PerceptOR Integration project (contract MDA972-01-9-0005). The views and conclusions contained in this document are those of the authors and should not be interpreted as representing the official policies, either expressed or implied, of the U.S. Government.

## REFERENCES

- [1] D. Silver, B. Sofman, N. Vandapel, J. A. Bagnell, and A. Stentz, "Experimental analysis of overhead data processing to support long range navigation," in *International Conference on Intelligent Robots and Systems*, October 2006.
- [2] D. Fox, W. Burgard, F. Dellaert, and S. Thrun, "Monte carlo localization: Efficient position estimation for mobile robots," in *National Conference on Artificial Intelligence*, 1999.

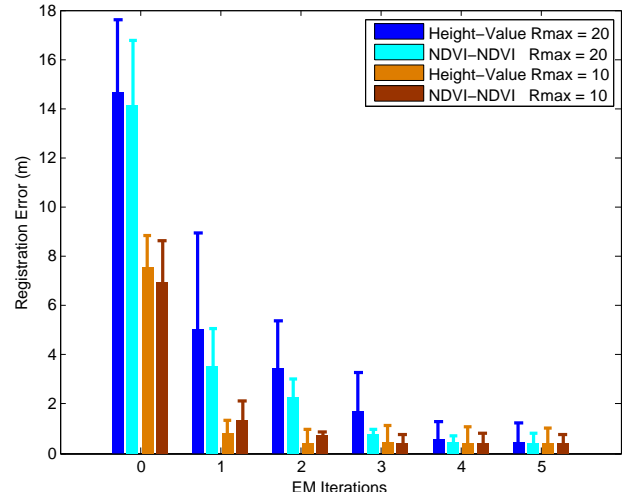


Fig. 8. Experiments in correcting for registration error. 10 trials were performed for each set of experiments; error bars represent the standard deviation. Over time, the registration error is rapidly corrected, resulting in final error of only 40cm on average.

- [3] A. Stentz, J. Bares, T. Pilarski, and D. Stager, "The crusher system for autonomous navigation," in *AUVSs Unmanned Systems*, 2007.
- [4] J. A. Bagnell, D. Bradley, D. Silver, B. Sofman, and A. Stentz, "Learning for autonomous navigation: Advances in machine learning for rough terrain mobility," *IEEE Robotics & Automation Magazine*, vol. 17, no. 2, pp. 74–84, June 2010.
- [5] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics*, 2005.
- [6] M. Adams, S. Zhang, and L. Xie, "Particle filter based outdoor robot localization using natural features extracted from laser scanners," in *Int. Conf. on Robotics and Automation*, 2004, pp. 1493 – 1498.
- [7] J. Guivant and R. Katz, "Global urban localization based on road maps," in *International Conference on Intelligent Robots and Systems*, 2007, pp. 1079–1084.
- [8] C. Dogruer, B. Koku, and M. Dolen, "Global urban localization of outdoor mobile robots using satellite images," in *International Conference on Intelligent Robots and Systems*, 2008, pp. 3927–3932.
- [9] R. Kummerle, B. Steder, C. Dornhege, A. Kleiner, G. Grisetti, and W. Burgard, "Large scale graph-based SLAM using aerial images as prior information," in *Robotics: Science and Systems*, June 2009.
- [10] L. Xu and A. Stentz, "Cost-based registration using a priori data for mobile robot localization," Carnegie Mellon University, Pittsburgh, PA, Tech. Rep. CMU-RI-TR-08-05, 2008.
- [11] P. J. Carle, P. T. Furgale, and T. D. Barfoot, "Long-range rover localization by matching lidar scans to orbital elevation maps," *Journal of Field Robotics*, vol. 27, no. 3, pp. 344–370, 2010.
- [12] C. Plagemann, K. Kersting, P. Pfaff, and W. Burgard, "Gaussian beam processes: A nonparametric bayesian measurement model for range finders," in *Robotics: Science and Systems*, June 2007.
- [13] P. Pfaff, C. Plagemann, and W. Burgard, "Improved likelihood models for probabilistic localization based on range scans," in *International Conference on Intelligent Robots and Systems*, 2007, pp. 2192–2197.
- [14] T. Yap and C. Shelton, "Simultaneous learning of motion and sensor model parameters for mobile robots," in *IEEE Conference on Robotics and Automation*, 2008.
- [15] A. I. Eliazar and R. Parr, "Learning probabilistic motion models for mobile robots," in *ICML*, 2004.
- [16] M. W. Bode, "Learning the forward predictive model for an off-road skid-steer vehicle," Carnegie Mellon University, Pittsburgh, PA, Tech. Rep. CMU-RI-TR-07-32, September 2007.
- [17] B. Sofman, E. L. Ratliff, J. A. Bagnell, J. Cole, N. Vandapel, and A. Stentz, "Improving robot navigation through self-supervised online learning," *Journal of Field Robotics*, vol. 23, no. 12, December 2006.
- [18] A. Dempster, N. Laird, and D. Rubin, "Maximum likelihood from incomplete data via the em algorithm," *Journal of the Royal Statistical Society*, vol. 39, no. 1, pp. 1–38, 1977.
- [19] P. Viola and W. M. Wells, "Alignment by maximization of mutual information," *International Journal of Computer Vision*, vol. 24, no. 2, pp. 137–154, 1997.
- [20] D. Bradley, R. Unnikrishnan, and J. A. Bagnell, "Vegetation detection for driving in complex environments," in *International Conference on Robotics and Automation*, April 2007.