Biologically-Inspired Robotics Vision Monte-Carlo Localization in the Outdoor Environment

Christian Siagian and Laurent Itti

***Abstract*— We present a robot localization system using biologically-inspired vision. Our system models two extensively studied human visual capabilities: (1) extracting the "gist" of a scene to produce a coarse localization hypothesis, and (2) refining it by locating salient landmark regions in the scene. Gist is computed here as a holistic statistical signature of the image, yielding abstract scene classification and layout. Saliency is computed as a measure of interest at every image location, efficiently directing the time-consuming landmark identification process towards the most likely candidate locations in the image. The gist and salient landmark features are then further processed using a Monte-Carlo localization algorithm to allow the robot to generate its position. We test the system in three different outdoor environments - building complex (126x180ft. area, 3794 testing images), vegetation-filled park (270x360ft. area, 7196 testing images), and open-field park (450x585ft. area, 8287 testing images) - each with its own challenges. The system is able to localize, on average, within 6.0, 10.73, and 32.24 ft., respectively, even with multiple kidnapped-robot instances.**

I. INTRODUCTION

The problem of localization is central to endowing mobile machines with intelligence. Range sensor such as sonar and ladar [1], [2], [3] are particularly effective indoors due to many structural regularities such as flat walls and narrow corridors. In the outdoors, however, these sensors become less robust given all the protrusions and surface irregularities [4]. For example, a slight change in pose can result in large jumps in range reading because of tree trunks, moving branches, and leaves. GPS, coupled with other sensors or by itself [5], has also been extensively used. However, GPS may not be applicable in environments where there is no satellite visibility, such as underwater, in caves, indoors, or on Mars. In those places vision, human's main perceptual system for localization, should be a viable alernative.

Existing vision-based localization systems can be categorized based on several groupings. The first one is according to image-view types, where some systems use ground-view images [6], [7] and others use omni-directional images [8], [9], [10]. Another categorization is according to localization goal, such as actual metric location [11] or a coarser place or room number [8]. Yet another grouping is according to whether or not the system is provided with a map. Presently, simultaneous localization and mapping (SLAM) [12], [13], [14] is an active branch of robotics research.

One additional categorization to consider here comes from the vision perspective, which classifies systems according to the type of visual features used: local features and global features. Local features are computed over a limited area of the image, as opposed to global features which may pool information over the entire image into, e.g., histograms. Before analyzing the variety of approaches, which by no means is exhaustive, it should be pointed out that, like other vision problems, any localization and landmark recognition system faces the general issues of occlusion, dynamic background, lighting, and viewpoint changes.

A popular starting point for local features are SIFT keypoints [15]. There has been a number of systems that utilize SIFT features [6], [16], [17] in recent years for object recognition because they can work in the presence of occlusion and some viewpoint changes. Other examples of local features are Kernel PCA features [18] and Harris corners [19]. Some systems [20], [21] extend their scope of locality by matching image regions to recognize a location. At this level of representation, the major hurdle lies in achieving reliable segmentation and in robustly characterizing individual regions. This is especially difficult with unconstrained environments such as a park where vegetation dominates (figure 4).

Global feature methods usually utilize color [8], [9], textures [7], or a combination of both [22], [23]. Holistic approaches, which do not have a segmentation stage, may sacrifice spatial information (the location of the features). Yet, some systems [7], [22] try to recover crude spatial information by using a predefined grid and computing global statistics within each grid tile. These methods are limited, for the most part, to recognizing places (as opposed to exact geographical locations) because with global features, it is harder to deduce a change in position even when the robot moves considerably.

Today, with many available studies in human vision, there is a unique opportunity to develop systems that take inspiration from neuroscience and bring a new perspective in solving vision-based robot localization. For example, even in the initial viewing of a scene, the human visual processing system already guides its attention to visually interesting regions within the field of view. This extensively studied early course of analysis [24], [25], [26], [27] is commonly regarded as being guided by perceptual saliency. Saliency-based or "bottom-up" guidance of attention highlights a limited number of possible points of interest in an image,

This work was supported by ONR, NSF, NGA, DARPA and HFSP.

C. Siagian is with Department of Computer Science, University of Southern California, Hedco Neuroscience Building - Room 10, 3641 Watt Way, Los Angeles, California, 90089-2520, USA. siagian@usc.edu

L. Itti is with the Faculty of Computer Science, Psychology, and Neuroscience, Univesity of Southern California, Hedco Neuroscience Building - Room 30A, 3641 Watt Way, Los Angeles, California, 90089-2520, USA. itti@pollux.usc.edu

which would be useful [28] in selecting landmarks that are the most reliable in a particular environment (a challenging problem in itself). Moreover, by focusing on specific sub-regions and not the whole image, the matching process becomes more flexible and less computationally expensive.

Recent discoveries in human vision show that humans are able to recognize scenes at multiple levels. Concurrent with the mechanisms of saliency, humans also exhibit the ability to rapidly summarize the “gist” of a scene [29], [30], [31], [32] in less than 100ms. Human subjects are able to consistently answer detailed inquiries such as the presence of an animal in a scene [33], [34], general semantic classification (indoors vs. outdoors, room types: kitchen, office, etc.) and rough visual feature distributions such as colorful vs. grayscale images or several large masses vs. many small objects in a scene [35], [36] It is reported that gist computations may occur in brain regions which also respond to “places”, that is, it prefers scenes that are notable by their spatial layout [37] as opposed to objects or faces. In addition, gist perception is affected by spectral contents and color diagnosticity [32], [38], which leads to the implementation of models such as [39], [40].

In spite of how contrasting saliency and gist are, both of these modules rely upon raw features that come from the same area, the early visual cortex. Furthermore, the idea that gist and saliency are computed in parallel is demonstrated in a study in which human subjects are able to simultaneously discriminate rapidly presented natural scenes in the peripheral view while being involved in a visual discrimination task in the foveal view [41]. From an engineering perspective it is an effective strategy to analyze a scene from opposite resolution levels, a high-level, image-global layout (corresponding to gist) and detailed pixel-wise analysis (saliency). It is also important to note that while saliency models primarily utilize local features [27], gist features are almost exclusively global or holistic [38], [7], [22]. Our model presented below seeks to employ these two complementary concepts of biological vision, implemented faithfully and efficiently, to produce a critical capability such as localization. Figure 1 shows a diagram of the full system with each sub-system projected onto its respective anatomical location.

After early preprocessing that takes place at both the retina and LGN (following figure 1), the visual stimuli arrive at Visual Cortex (cortical visual areas V1, V2, V4, and MT) for low-level feature extractions which are then channeled to the saliency and gist module. Along the Dorsal Pathway or “where” visual processing stream [42] (posterior parietal cortex), the saliency module builds a saliency map through the use of spatial competition of low-level feature responses throughout the visual field. This competition silences locations which, at first, may produce strong local feature responses but resemble their neighboring locations. Conversely, the competition strengthens points which are distinct from their surroundings. On the contrary, in the Ventral Pathway or the “what” visual processing stream (inferior temporal cortex), the low-level feature-detector responses are combined to yield a gist vector as a concise global synopsis of the scene as a whole. Both pathways end up at the pre-frontal cortex where concious decisions and motor commands are formed.

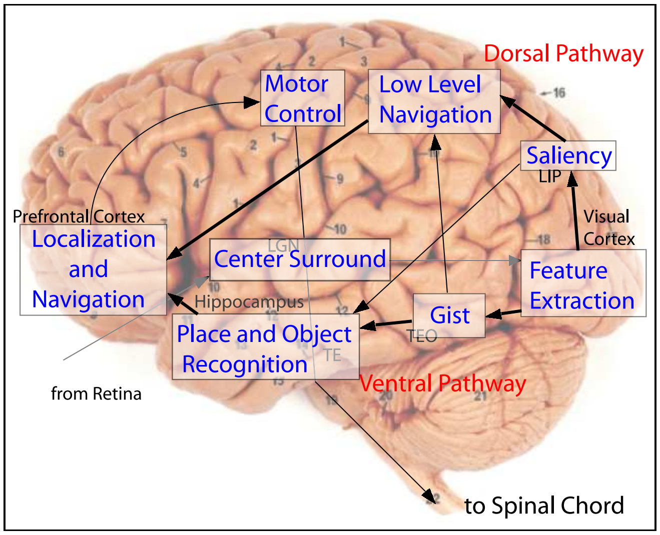

Fig. 1. Model of Human Vision with Gist and Saliency.

There is an additional consequence of the Dorsal and Ventral pathway division that is applicable in vision robotics. The Ventral pathway, which includes areas such as the hippocampus and para-hippocampus (known to be involved in recognition and spatial memory recollection [43]), performs at a slower speed than the Dorsal pathway, which is real time. The dorsal pathway module performs navigational tasks such as obstacle avoidance, which require fast reaction but not recognition. The dorsal module makes use of the salient features for tracking objects, motion cues for lane following. It may also need stereo vision to perform obstacle avoidance. In effect, the brain can be viewed as a behavior-based architecture [44]. In this paper, we concentrate mostly on the ventral pathway, which is responsible for localization.

II. DESIGN AND IMPLEMENTATION

Figure 2 displays the overall flow of the localization system which can be divided to three stages: feature extraction, object and place recognition, and localization. The feature extraction stage takes an image from a camera (or retina in figure 1) and outputs the gist [22] and salient feature computations [26], [45], which are already implemented previously. Our main contribution is utilizing both of the sub-systems concurrently in the two subsequent stages. The place and object recognition stage then tries to match these features with memorized information about the environment. These matches are then used as an input to the localization stage to make a decision of where the robot might be.

As part of the object and place recognition stage, a map of the environment is associated with the visual information. The map, which is currently provided to the system, is an augmented-topological map. It is a graph-based map with each node having a cartesian coordinate and each edge having its cost set to the distance between the edge’s

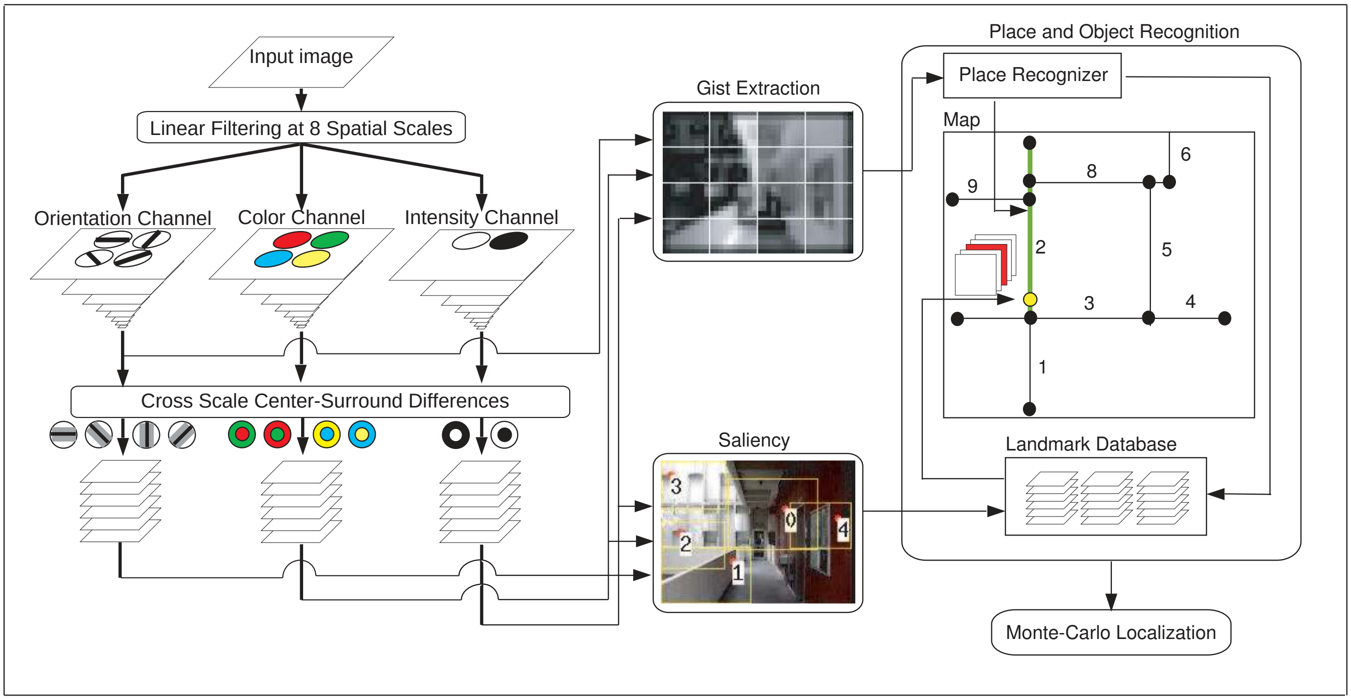

Fig. 2. Diagram for the Vision Localization System

corresponding end-nodes. This way the system benefits from the compact representation of a graph while preserving the important metric information of the environment. In the map, a robot state (position and viewing direction) is represented by a point which can lie on a node or an edge.

For further analysis, we introduce the concept of a segment. A segment is an ordered list of edges with each edge connected to the next one in the list to form a continuous path. This grouping is motivated by the fact that views/layout in one path-segment are coarsely similar. The selected three-edge segment (highlighted in green) in the map of figure 2 is an example. From this point forward, "place" (as in figure 1) and "segment" will be used interchangably to refer to the same notion of a region in a map. Because the map includes a rectangular boundary and an origin, a location can be noted as both cartesian coordinates (x, y) or a pair of segment number and a fraction of length traveled (between 0.0 to 1.0) along the segment $(snum, ltrav)$.

A. Feature extraction, Saliency, and Gist

As mentioned above, the low-level visual-feature extraction [26], [45], the saliency computation stage, [26], [45] and the gist model [22] have previously been reported. In summary, the low-level features consist of center-surround color, intensity, and orientation that are computed in separate channels. Each of these channels are run in parallel, which in our robot (including the gist and saliency extraction) runs at 80ms/frame (12.5 fps). At this point the gist features are ready to use for recognition while we need to further process the output of the saliency model. We select up to five of the most salient points per frame and use a shape estimator algorithm [46] to segment out the respective salient regions for each point. We put a limit in the number of salient regions per frame because, from experiments, the subsequent regions have a much lower likelihood of being repeatable in other training or testing runs.

B. Salient Region/Object and Segment/Place Recognition

There are two separate training steps for the system. The first one is to build a visual landmark database, the second is to train a segment classifier using gist. These two sets of information are then connected through the environment map with the landmarks compartmentalized by their respective segment of origin. Later on, this will enable the system to prioritize the on-line landmark search process using gist. The training procedure involves a guided traversal of the robot through all the paths in the map. This should be performed several times to have ample lighting coverage as well as to allow identification of landmarks that are consistent over a number of runs. In what follows, the term landmark refers to an actual point of interest in the environment, not image of an object. An image is an evidence of a landmark and a landmark is memorized as a set of images which provide different views as the robot passes through it in the environment. And thus, the phrase "matching a salient region to a landmark" means to match a region image with one of the memorized images on the list of a landmark.

1) Segment/Place Recognition: The segment estimator is implemented using a neural network classifier, trained on gist features using the back-propagation algorithm. One of the main reasons why the classifier succeeds is because we decided to group edges to segments as it would have been difficult to train an edge-classifier using coarse features like gist. Each segment in the environment has an associated output node and the output potential is the likelihood that the scene belongs to that segment, stored in a vector $z_t^{'}$ to be used by the localization algorithm as an observation where

$$z_t^{'} = \{ \; sval_{t,j} \; \}, \; j = 1 \; ... \; N_{segment} \tag{1}$$

with $sval_{t,j}$ being the segment likelihood for time t and segment j which is one of the $N_{segment}$ segments in the environment.

2) Salient Region/Object Recognition: The object recognition module generates SIFT features [15] for each salient region and uses them along with their corresponding salient feature vectors for all matching processes (training and testing). Salient feature vectors are sets of feature-map values (from each low-level channel) taken at the salient points. In the landmark database building phase, the first incoming salient regions are used to create initial landmarks. When the next video frame arrives, the system tries to match the new salient regions to the existing landmarks. The ones that are matched can then be added to the corresponding landmarks while the remaining salient regions are used to create new landmarks. Once all the frames are processed, the landmarks are pruned by setting minimum thresholds such as number of images and range of video frame numbers, both of which indicate how persistent the landmark is in the environment. In addition, the landmarks can also be pruned across traversals by only considering the ones that occur in more than one runs.

As alluded before, the landmarks are arranged by the segments of origin which are used to prioritize search order (in test runs) using gist. The salient feature vector can also be used to prime the order based on Euclidian distance. In real time systems such as robots, it is a given that the database search ends after the first object found, as the system does not require the best match. So, it is desirable to have a high matching threshold, one that tend to give some false negatives but almost no false positives.

Once the incoming salient regions are compared with the landmark database, we obtain a number of successful matches which are denoted for observation as $z_t^{''}$, where

$$z_t^{''} = \{ \; omatch_{t,k} \; \}, \; k = 1 \; ... \; M_t \tag{2}$$

with $omatch_{t,k}$ being the found object/salient region match k (one of M_t matches) at time t. Note that the object recognition module may not produce an observation for every time t ($M_t = 0$) as it may either find no matches or still be currently processing.

C. Monte-Carlo Localization

We estimate the robot's position by implementing Monte-Carlo Localization (MCL) [1], [11], [13]. It formulates the location belief as a set of weighted particles: $S_t = \{ \; x_{t,i}, \; w_{t,i} \} \; i = 1 \; ... \; N$ at time t. Each particle $x_{t,i}$ is composed of a segment number and percentage of length traveled along the segment edges $x_{t,i} = \{snum_{t,i}, \; ltrav_{t,i}\}$ and has a weight of $w_{t,i}$ which is proportional to the likelihood of observing incoming data modeled by the segment and salient region observation model (sections II-C.2 and II-C.3). From experiments, $N = 100$ seemed to suffice for the simplified localization domain where a hallway is represented by an edge and not a two dimensional space. However, it should be pointed out that in this system, the dorsal pathway will be the one responsible to keep the robot in the middle of the path, avoiding a need to localize laterally.

We estimate the location belief $Bel(S_t)$ by evolving posterior $p(S_t|z^t, u^t)$ - z_t being an evidence and u_t the motion measurement - by recursively updating $Bel(S_t)$ [47]:

$$\begin{aligned} Bel(S_t) &= p(S_t|z^t, u^t) \\ &= \alpha p(z_t|S_t) \int_{S_{t-1}} p(S_t|S_{t-1}, u_t) Bel(S_{t-1}) \, dS_{t-1} \end{aligned} \tag{3}$$

We first compute $p(S_t|S_{t-1}, u_t)$ (called the prediction/proposal phase) to take motion into consideration by applying the motion model to the particles. Afterwards, $p(z_t|S_t)$ is computed in the update phase to incorporate the visual information by applying the observation model to the weight of each particle for a weighted resampling step.

As explained above (sections II-B.1 and II-B.2), the system observes two types of evidence: $z_t^{'}$ and $z_t^{''}$ which are segment estimation and object recognition, respectively. Segment estimation is available at each time step while object recognition is not always available as it might not be ready or returns no match. In the procedure below, the object recognition observation is treated as evidence that arrives at the following time step after a zero motion. Consequently, this condenses the procedure to having compound observations because the prediction phase is effectively non-existent. And thus, at each time step t the system computes belief estimation $Bel(S_t)$ in the following order:

1) apply motion model to S_{t-1} to create $S_t^{'}$
2) apply segment observation model to $S_t^{'}$ to create $S_t^{''}$
3) if ($M_t > 0$)
    a) apply object observation model to $S_t^{''}$ to yield S_t
    b) else $S_t = S_t^{''}$

We specify two intermediate states $S_t^{'}$ and $S_t^{''}$, the former being the belief after the motion model is applied to the particles, moving it by the measured movement, while the latter is the state after the segment observation is then subsequently applied. Lastly the object observation model (if there are found objects at time t) is applied to $S_t^{''}$ to produce S_t.

1) Motion Model: The system employs a straightforward motion model for an odometry reading u_t. We apply the motion to each particle from the set S_{t-1} by sampling a random particle $x_{t,i}^{'}$ from the density $p(x_{t,i}^{'}|u_t, x_{t-1,i})$. Included in the probability density is noise drawn from a gaussian distribution to account for wheel slippage with a standard deviation of .1ft which is proportional to 1/6th of a typical single step. The standard deviation controls the level of uncertainty in the robot movement measurement, the bigger the number, the greater the level of noise added. From our experiment, we find that this number does not affect the end result as much because the number of particles around the vicinity of a converged location is large enough that motion error in any direction is well covered by the neighborhood of particles.

2) Segment-Estimation Observation Model: We weigh each particle $x^{'}_{t,i}$ with $w^{'}_{t,i} = p(z^{'}_t|x^{'}_{t,i})$ for resampling (with added 10 percent random particles to avoid the well known population degeneration problem in Monte Carlo methods) to create belief $S^{''}_t$ by taking into account the segment-estimation vector $z^{'}_t$ (equation 1).

$$p(z^{'}_t|x^{'}_{t,i}) = \frac{sval_{t,snum^{'}_{t,i}}}{\sum_{j=1}^{N_{segment}} sval_{t,j}} * sval_{t,snum^{'}_{t,i}} \qquad (4)$$

Here, the likelihood that a particle $x^{'}_{t,i}$ observes $z^{'}_t$ is proportional to the percentage of its segment value $sval_{t,snum^{'}_{t,i}}$ (measures its dominance with respect to other entries) times its absolute value (to preserve its ratio with respect to maximum possible value of 1.0). In the implementation, the denominator is taken out because it is equal for all particles.

3) Salient-Region-Recognition Observation Model: We weigh each particle $x^{''}_{t,i}$ with $w^{''}_{t,i} = p(z^{''}_t|x^{''}_{t,i})$ for resampling (with added 20 percent random noise, also to combat the population degeneracy problem in Monte Carlo methods) to create belief S_{t+1} by taking into account the object matches $z^{''}_t$ (equation 2).

$$p(z^{''}_t|x^{''}_{t,i}) = \prod_{k=1}^{M_t} p(omatch_{t,k}|x^{''}_{t,i}) \qquad (5)$$

Here, each object-match observation is independent and thus is processed individually. The probability $p(omatch_{t,k}|x^{''}_{t,i})$ is modeled by a gaussian with σ set to 5% of the environment map diagonal. The likelihood value is the probability of drawing a length longer than the distance between the particle and the location where the database object matched is acquired. σ is set proportional to the map diagonal to reflect how the larger the environment means the higher the level of observation uncertainty. The added noise is twice that of segment observation because the object observation probability density is much narrower than the previous one and we find that 20% keeps the particle population diverse enough so that in a kidnapped robot event, the particles are able to disperse and reconverge to the new location. Also, although the SIFT matching scores are available for weights, we decided to assume all object match accuracies are equal.

III. TESTINGS AND RESULTS

The localization system is tested at three outdoor sites: ACB, AnF, and FDF, each composed of 9 segments (labeled in figure 3, 4, and 5, respectively), using videos provided by [48] with the maps illustrated in [22]. The ACB scenes are filmed throughout the narrow corridors of a 126x180ft. building complex. Most of the surroundings are flat walls with little texture and solid lines that delineate the walls and different parts of the buildings. The scenes of the 270x360ft.

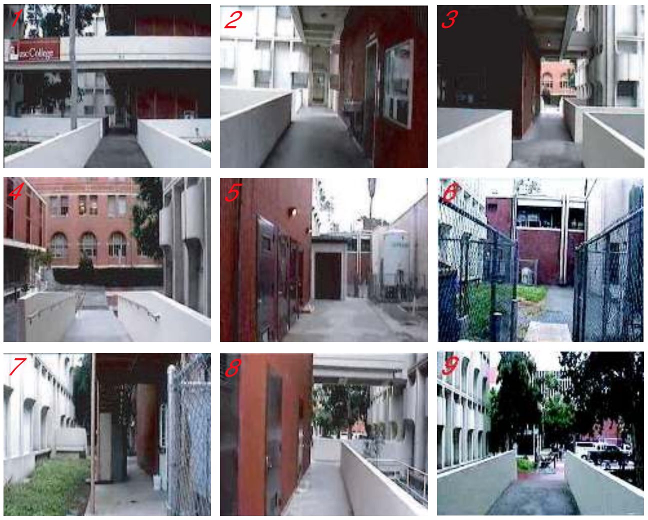

Fig. 3. Examples of images in each of the nine segments (with corresponding label) of ACB.

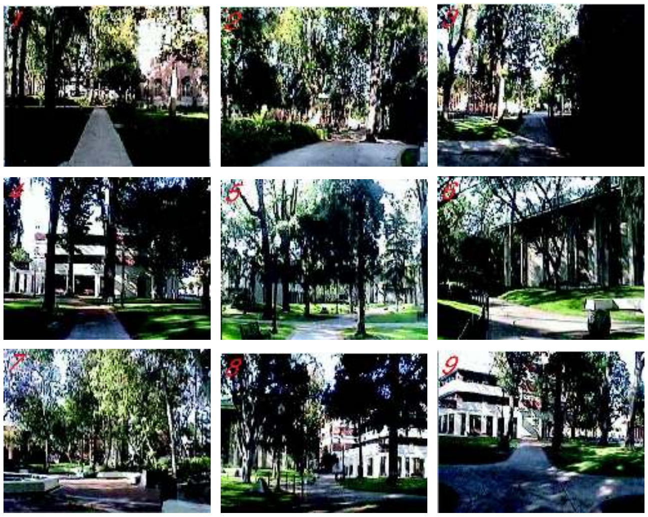

Fig. 4. Examples of images in each of the nine segments (with corresponding label) of AnF.

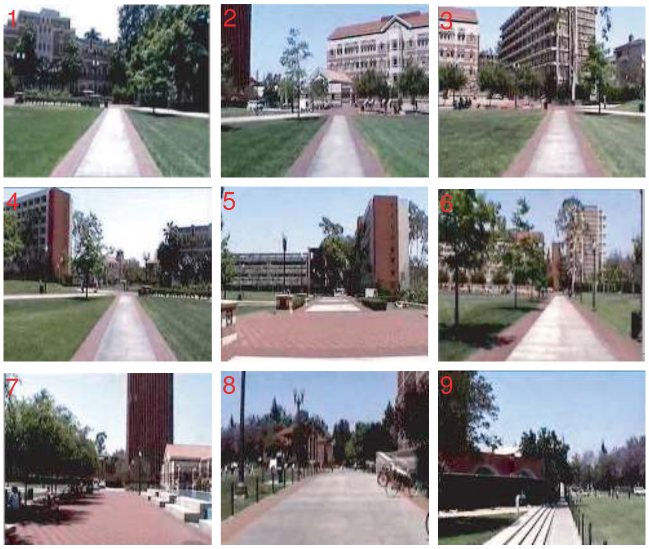

Fig. 5. Examples of images in each of the nine segments (with corresponding label) of FDF.

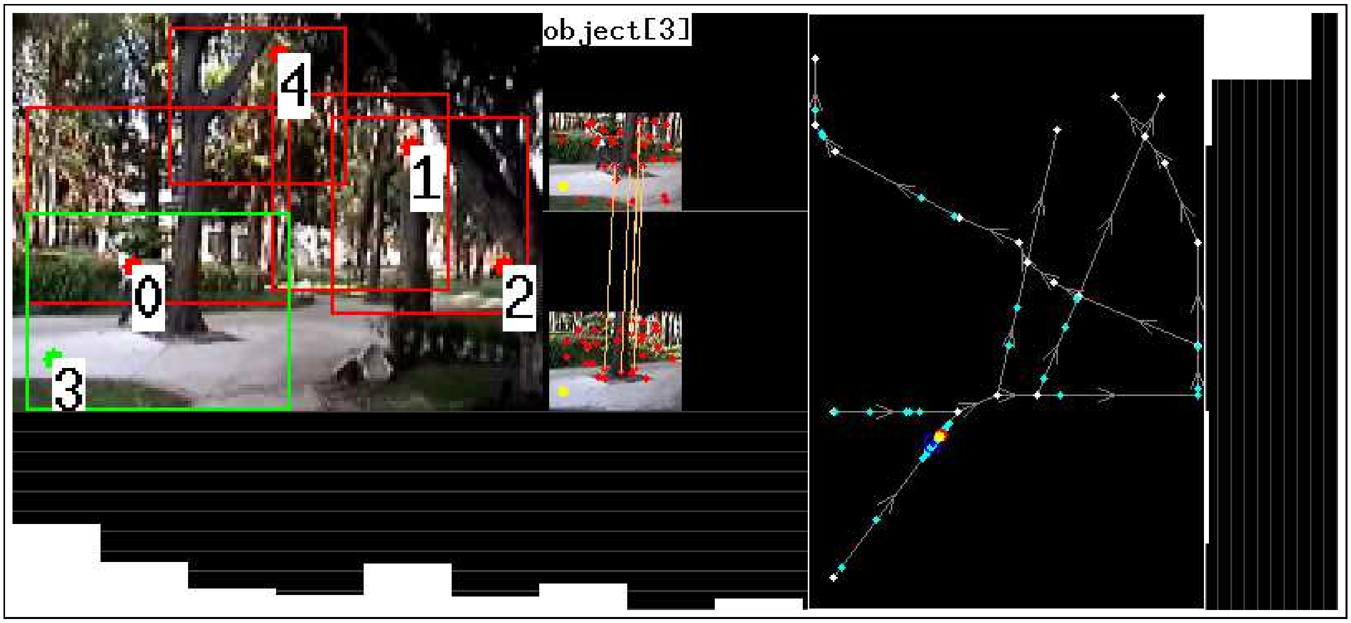

Fig. 6. A snapshot of the system test-run. Top-left (main) image contains the salient region windows. Green window means a database match, while red is not found. An object match is displayed next to the main image. Below the main image is the segment estimation vector derived from gist (there are 9 possible segments in the environment). The middle image projects the robot state onto the map: cyan disks are the particles, the yellow disks are the location of the matched database objects, the blue disk (the center of the blue circle, here partially covered by a yellow disk) is the most likely location. The radius of the blue circle is equivalent to five feet. The right-most histogram is the number of particles at each of the 9 possible segments. The robot believes that it is towards the end of the first segment, which is correct within a few feet.

AnF site are dominated by vegetation. The length of the segments at this site are about twice the length of the segments in ACB. A large portion of the scenes of the 450x585ft. FDF site is sky, which is mostly textureless space with scattered light clouds. The lengths of the FDF segments are about 50% longer than the ones in AnF and three times that of ACB. Because the data is taken by a person carrying a handheld camera walking at approximately constant speed, we use interpolation to come up with the ground-truth location of the person for both training and testing data.

The data for each site consists of 12 to 15 runs, spanning various lighting conditions. The gist model has been shown to work in these lighting conditions [22]. However, in the current testing setup, we will take two runs for each site that are of comparable lighting conditions, training the system on the first one, and testing it on the second one.

Figure 6 displays results at one time step.

Table I reports the results for the three different environments. For ACB and AnF, the error is quite uniform throughout the segments except for two spikes in segment 8 for both sites (11.22 and 22.99ft, respectively). The main problem encountered here, as well as everywhere else to a certain degree, is that of scale. The SIFT object recognition module is able to perform scale-invariant matching (with the scale ratio included as part of the result). However, this presents a problem as there is not enough information to deduce a location where the matching would be close to 1-to-1. Because of the nature of the localization problem, where landmarks in the outdoor environment are not easily measured, systems usually are not able to obtain actual sizes of objects unless they are pre-specified. In segment 8 of AnF, we have matches with the side of a building that looks almost identical for a long stretch of the path leading to it. A better way to solve this problem would be to track the landmark and use the change in scale from a measured movement of the robot to obtain the landmark size. Currently the localization system has not yet incorporated the dorsal tracking module to perform this method. Instead, the system limits the matching-scale threshold to between 2/3 and 3/2. This is not entirely effective, however, as a scale ratio of 0.8 (the object found is smaller than the matched database object) can translate to a geographical difference of 15 feet.

The results for the FDF site, however, are not quite as good. There are two reasons for this: scale (as with the other sites) and object recognition failure because of lighting conditions. Segment 6 severely exhibits this problem because its path is straight, leading to a large building (figure 5). The large error in segment 7, on the other hand, is caused by the inability of the SIFT module to recognize any object in sight for long stretches of time. We mentioned earlier that the training and testing pairs are selected for lighting. It seems to be the case that, for this segment, lighting was quite different between the two instances as the training data is much brighter than the testing data. It should be noted, however, that in other sites, the object recognition module performed well in the presence of occlusion, viewpoint changes, and some lighting changes. The authors would suggest that a way to alleviate this problem is to simply train the system with more samples from other lighting conditions.

TABLE I

EXPERIMENTAL RESULTS

Segment	ACB		AnF		FDF	
	frame num.	error (ft)	frame num.	error (ft)	frame num.	error (ft)
1	377	6.77	866	7.34	782	12.38
2	496	7.65	569	4.87	813	12.97
3	541	5.24	896	8.31	869	9.37
4	401	4.92	496	7.66	795	30.36
5	304	7.21	769	8.87	857	18.24
6	555	4.94	1250	13.51	1434	45.20
7	486	2.70	588	5.84	839	101.62
8	327	11.22	844	22.99	1149	27.00
9	307	5.50	918	11.22	749	23.97
Total	3794	6.00	7196	10.73	8287	32.24

IV. DISCUSSION AND CONCLUSIONS

We introduced several new ideas in robotics vision localization which have been proven in our testing to be quite beneficial. The first one is the use of complementary features (gist and saliency) and how they could possibly interact. The vision model implements both in parallel, especially in the computation-heavy feature extraction phase, as the study of the human visual cortex would suggest. Through the saliency model, the system automatically selects salient objects so that it does not have to perform whole-scene view matching. In addition, the gist features which approximate layout come with saliency at almost no computation time. In essence what we have is the use of multiple experts, implemented in an efficient manner through sharing of some of the computations.

The system also performs both hierarchical recognition and multi-level localization. Hierachical recognition, which has been shown [49], [50] to speed up the process, is done by prioritizing the landmark database search through segment estimation, salient feature matching, and the current state (e.g. matching landmarks that are in the vicinity of the most belief location) before performing object recognition. Multi-level localization, on the other hand, is done by using both segment estimation and object recognition as observations in the Monte-Carlo localization. Many scene-based methods [8], [9], [7], [22], [23] that are limited to recognizing places (as opposed to geographical points) indicate that their results can be used as a filter for a more accurate attempt at localization with the use of finer yet more volatile local features. Our system is the implementation of such extension.

As for performance benchmark, to the best of our knowledge, we have not seen other systems that are tested in multiple outdoor environments (building complex, vegetation dominated park, and open field) and are localizing to the coordinate level (not a place). At the 2005 ICCV Computer Vision contest[51], where the goal was to localize from a database of street-level photographs tagged with GPS coordinates of a stretch (about 1 city block) of urban street, the winning team [52] returns 9/22 answers within 4 meters of the actual location. Our system has a 6ft. (1.8m) error in a 126x180ft. building complex, although our system can store as many pictures as it wants. Most other purely vision-based systems are tested indoors and a majority of them reports just the recognition rate, that is, if the current view is correctly matched with stored images, not the location. However, if we compare just the system segment prediction with other scene based methods (which are place recognizers and usually report results in the mid to upper ninetieth percentile), it fares quite well. The times where our system is lost are when the segment and salient region modules are both confused, which usually occurs in extremely different lighting condition than the one used in training.

The system now needs to resolve the issue of integrating the localization module (in the ventral pathway, figure 1) with the rest of the architecture, namely the autonomous navigation or dorsal tracking module. The cooperation of both pathways occurs in the recognition process, where the ventral module relies on the dorsal module to track a salient region as it is being matched to the landmark database.

A problem related to localization is the goal-seeking task. Here the robot also needs to follow a path to the goal location. One way to do this is through landmark hopping. After the robot is able to localize, it sets the path to the goal and creates the corresponding list of landmarks to look for. It would then go to the first landmark in the path and, while going to that direction, the system tries to attend and recognize the subsequent landmarks. When the next landmark that will advance the robot's even further is recognized, it switches to that one. This is done until the goal location is found. Biasing the saliency module to look specifically into the vicinity of the next object in the path is possible because the database stores all object image-coordinate locations from the training process.

V. ACKNOWLEDGMENTS

The authors gratefully acknowledge the contribution of Eric Choi, Rob Peters, and Lior Elazary.

REFERENCES

[1] D. Fox, W. Burgard, F. Dellaert, and S. Thrun, "Monte carlo localization: Efficient position estimation for mobile robots," in *Proceed-*

ings of the Sixteenth National Conference on Artificial Intelligence (AAAI'99)., July 1999.

[2] S. Thrun, D. Fox, and W. Burgard, "A probabilistic approach to concurrent mapping and localization for mobile robots," *Machine Learning*, vol. 31, pp. 29–53, 1998.

[3] F. Lu and E. Milios, "Robot pose estimation in unknown environments by matching 2d range scans," *Journal of Intelligent and Robotic Systems*, vol. 18, pp. 249 – 275, 1997.

[4] K. Lingemann, H. Surmann, A. Nuchter, and J. Hertzberg, "Indoor and outdoor localization for fast mobile robots," in *IROS*, 2004.

[5] M. Agrawal and K. Konolige, "Real-time localization in outdoor environments using stereo vision and inexpensive gps," in *ICPR06*, 2006, pp. III: 1063–1068.

[6] S. Se, D. G. Lowe, and J. J. Little, "Vision-based global localization and mapping for mobile robots," *IEEE Transactions on Robotics*, vol. 21, no. 3, pp. 364–375, 2005.

[7] A. Torralba, K. P. Murphy, W. T. Freeman, and M. A. Rubin, "Context-based vision system for place and object recognition," in *IEEE Intl. Conference on Computer Vision (ICCV)*, Nice, France, October 2003, pp. 1023 – 1029.

[8] I. Ulrich and I. Nourbakhsh, "Appearance-based place rcognition for topological localization," in *IEEE-ICRA*, April 2000, pp. 1023 – 1029.

[9] P. Blaer and P. Allen, "Topological mobile robot localization using fast vision techniques," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 2002.

[10] N. Winters, J. Gaspar, G. Lacey, and J. Santos-Victor, "Omni-directional vision for robot navigation," in *In IEEE Workshop on Omnidirectional Vision*, June 2000, pp. 21 – 28.

[11] S. Thrun, M. Bennewitz, W. Burgard, A. Cremers, F. Dellaert, D. Fox, D. Hähnel, C. Rosenberg, N. Roy, J. Schulte, and D. Schulz, "MINERVA: A second generation mobile tour-guide robot," in *Proceedings of the IEEE International Conference on Robotics and Automation (ICRA)*, 1999.

[12] D. Haehnel, W. Burgard, D. Fox, and S. Thrun, "An efficient fastslam algorithm for generating maps of large-scale cyclic environments from raw laser range measurements," in *IROS*, 2003.

[13] M. Montemerlo, S. Thrun, D. Koller, and B. Wegbreit, "Fastslam: A factored solution to the simultaneous localization and mapping problem," in *AAAI*, 2002.

[14] A. Ranganathan and F. Dellaert, "A rao-blackwellized particle filter for topological mapping," in *ICRA*, 2006, pp. 810– 817.

[15] D. G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, no. 2, pp. 91–110, 2004.

[16] L. Goncalves, E. D. Bernardo, D. Benson, M. Svedman, J. Ostrowski, N. Karlssona, and P. Pirjanian, "A visual front-end for simultaneous localization and mapping," in *ICRA*, April 18 - 22 2005, pp. 44–49.

[17] P. Elinas and J. J. Little, "σMCL: Monte-Carlo localization for mobile robots with stereo vision," in *Proceedings of Robotics: Science and Systems*, Cambridge, USA, June 2005.

[18] H. Tamimi and A. Zell, "Vision based localization of mobile robots using kernel approaches," in *IROS*, 2004.

[19] S. Frintrop, P. Jensfelt, and H. Christensen, "Pay attention when selecting features," in *Int l Conf. on Pattern Recognition*, Hong Kong, August 2006.

[20] H. Katsura, J. Miura, M. Hild, and Y. Shirai, "A view-based outdoor navigation using object recognition robust to changes of weather and seasons," in *Proceedings of the 2003 IEEE/RSJ International Conference on Intelligent Robot and Systems (IROS 2003)*, Las Vegas, Nevada, US, October 27 - 31 2003, pp. 2974–2979.

[21] R. Murrieta-Cid, C. Parra, and M. Devy, "Visual navigation in natural environments: From range and color data to a landmark-based model," *Autonomous Robots*, vol. 13, no. 2, pp. 143–168, 2002.

[22] C. Siagian and L. Itti, "Rapid biologically-inspired scene classification using features shared with visual attention," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 29, no. 2, pp. 300–312, Feb 2007.

[23] A. Pronobis, B. Caputo, P. Jensfelt, and H. Christensen, "A discriminative approach to robust visual place recognition," in *IROS*, 2006.

[24] A. M. Treisman and G. Gelade, "A feature-integration theory of attention," *Cognit Psychol*, vol. 12, no. 1, pp. 97–136, 1980.

[25] J. Wolfe, "Guided search 2.0: A revised model of visual search," *Psychonomic Bulletin and Review*, vol. 1, no. 2, pp. 202 – 238, 1994.

[26] L. Itti, C. Koch, and E. Niebur, "A model of saliency-based visual attention for rapid scene analysis," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 20, no. 11, pp. 1254–1259, Nov 1998.

[27] L. Itti and C. Koch, "Computational modeling of visual attention," *Nature Reviews Neuroscience*, vol. 2, no. 3, pp. 194–203, Mar 2001.

[28] S. Frintrop, P. Jensfelt, and H. Christensen, "Attention landmark selection for visual slam," in *IEEE/RSJ International Conf. on Intelligent Robots and Systems (IROS)*, Beijing, October 2006.

[29] M. C. Potter, "Meaning in visual search," *Science*, vol. 187, no. 4180, pp. 965–966, 1975.

[30] I. Biederman, "Do background depth gradients facilitate object identification?" *Perception*, vol. 10, pp. 573 – 578, 1982.

[31] B. Tversky and K. Hemenway, "Categories of the environmental scenes," *Cognitive Psychology*, vol. 15, pp. 121 – 149, 1983.

[32] A. Oliva and P. Schyns, "Coarse blobs or fine edges? evidence that information diagnosticity changes the perception of complex visual stimuli," *Cognitive Psychology*, vol. 34, pp. 72 – 107, 1997.

[33] S. Thorpe, D. Fize, and C. Marlot, "Speed of processing in the human visual system," *Nature*, vol. 381, pp. 520 – 522, 1995.

[34] M. M. MJ, S.J.Thorpe, and M. Fabre-Thorpe, "Rapid categorization of achromatic natural scenes: how robust at very low contrasts?" *Eur J Neurosci.*, vol. 21, no. 7, pp. 2007 – 2018, April 2005.

[35] T. Sanocki and W. Epstein, "Priming spatial layout of scenes," *Psychol. Sci.*, vol. 8, pp. 374 – 378, 1997.

[36] R. A. Rensink, "The dynamic representation of scenes," *Visual Cognition*, vol. 7, pp. 17 – 42, 2000.

[37] R. Epstein, D. Stanley, A. Harris, and N. Kanwisher, "The parahippocampal place area: Perception, encoding, or memory retrieval?" *Neuron*, vol. 23, pp. 115 – 125, 2000.

[38] A. Oliva and P. Schyns, "Colored diagnostic blobs mediate scene recognition," *Cognitive Psychology*, vol. 41, pp. 176 – 210, 2000.

[39] A. Torralba, "Modeling global scene factors in attention," *Journal of Optical Society of America*, vol. 20, no. 7, pp. 1407 – 1418, 2003.

[40] C. Ackerman and L. Itti, "Robot steering with spectral image information," *IEEE Transactions on Robotics*, vol. 21, no. 2, pp. 247–251, Apr 2005.

[41] F. Li, R. VanRullen, C. Koch, and P. Perona, "Rapid natural scene categorization in the near absence of attention," in *Proc. Natl. Acad. Sci.*, 2002, pp. 8378 – 8383.

[42] L. G. Ungerleider and M. Mishkin, "Two cortical visual systems," in *Analysis of visual behavior*, D. G. Ingle, M. A. A. Goodale, and R. J. W. Mansfield, Eds. Cambridge, MA: MIT Press, 1982, pp. 549–586.

[43] N. Broadbent, L. Squire, and R. Clark, "Spatial memory, recognition memory, and the hippocampus," in *Proc National Academy of Science USA*, vol. 11, 2004, pp. 14 515 – 14 520.

[44] R. Arkin, *Behavior-Based Robotics*. Cambridge, MA: MIT Press, 1998.

[45] L. Itti, "Models of bottom-up and top-down visual attention," Ph.D. dissertation, California Institute of Technology, Pasadena, California, Jan 2000.

[46] U. Rutishauser, D. Walther, C. Koch, and P. Perona, "Is bottom-up attention useful for object recognition?" in *CVPR (2)*, 2004, pp. 37–44.

[47] S. Thrun, D. Fox, W. Burgard, and F. Dellaert, "Robust monte-carlo localization for mobile robots," *Artificial Intelligence*, vol. 128, no. 1-2, pp. 99–141, 2000.

[48] C. Siagian. (2007) Gist/context of a scene. [Online]. Available: http://ilab.usc.edu/ siagian/Research/Gist/Gist.html

[49] W. Zhang and J. Kosecka, "Localization based on building recognition," in *IEEE Workshop on Applications for Visually Impaired*, June 2005, pp. 21 – 28.

[50] J. Wang, H. Zha, and R. Cipolla, "Coarse-to-fine vision-based localization by indexing scale-invariant features," *IEEE Trans. Systems, Man and Cybernetics*, vol. 36, no. 2, pp. 413–422, April 2006.

[51] R. Szeliski, "Iccv2005 computer vision contest where am i?" http://research.microsoft.com/iccv2005/Contest/, Nov. 2005.

[52] W. Zhang and J. Kosecka, "Image based localization in urban environments," in *International Symposium on 3D Data Processing, Visualization and Transmission, 3DPVT*, Chapel Hill, North Carolina, 2006.