Biologically Inspired Mobile Robot Vision Localization

Christian Siagian, *Member, IEEE* Laurent Itti, *Member, IEEE,*

***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 points 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 features and salient regions 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 (38.4x54.86m area, 13966 testing images), vegetation-filled park (82.3x109.73m area, 26397 testing images), and open-field park (137.16x178.31m area, 34711 testing images) — each with its own challenges. The system is able to localize, on average, within 0.98, 2.63, and 3.46m, respectively, even with multiple kidnapped-robot instances.**

***Index Terms*—Gist of a scene, saliency, scene recognition, computational neuroscience, image classification, image statistics, landmark recognition, robot vision, robot localization.**

I. Introduction

THE problem of localization is central to endowing mobile machines with intelligence. Range sensors such as sonar and ladar [1], [2] are particularly effective indoors due to many structural regularities such as flat walls and narrow corridors. In the outdoors, these sensors become less robust given all the protrusions and surface irregularities [3]. 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 [4], 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, our main perceptual system for localization, should be a viable alternative.

We first describe traditional vision localization techniques as background information to better demonstrate the advantages of using biological approaches. In section I-B, we then introduce a robust biologically plausible vision system that concurrently observes a scene from two contrasting perspectives: its rough overall layout (using gist) and detailed recognition only on select globally conspicuous locations (using saliency). In addition, section I-C describes how using topological maps, which is analogous to how humans deal with spatial information, allows for a compact and accurate representation.

A. Traditional Vision-Based Localization

Existing vision-based localization systems can be categorized along several lines. The first one is according to image-view types, where some systems use ground-view images [5], [6] and others use omni-directional images [7], [8]. Another categorization is according to localization goal, such as actual metric location [9] or a coarser place or room number [7]. Yet another grouping is according to whether or not the system is provided with a map, or must build one as it locates itself (SLAM) [10], [11].

One additional categorization to consider comes from the vision perspective, which classifies systems according to visual feature type: local and global features. Local features are computed over a limited area of the image, whereas global features pool information over the entire image, e.g., into histograms. Before analyzing various 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 [12]. There have been a number of systems that utilize SIFT features [5], [13] 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 SURF [14] and GLOH [15]. Some systems [16], [17] 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 full of trees.

Global feature methods usually rely on comparing image statistics for color [7], [8], textures [6], or a combination of both [18], [19]. Holistic approaches, which do not have a segmentation stage, may sacrifice spatial information (feature location). Yet, some systems [6], [18] 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 (e.g. rooms in a building, as opposed to exact metric geographical locations) because with global features, it is harder to deduce a change in position even when the robot moves considerably.

C. Siagian and L. Itti are with the University of Southern California, Departments of Computer Science, Psychology, and Neuroscience Program, Hedco Neuroscience Building - Room 30A, 3641 Watt Way, Los Angeles, California, 90089-2520. Correspondence should be addressed to siagian@usc.edu.

Manuscript received XXX XX, 200X; revised XXX XX, 200X.

B. Biologically Plausible Scene Recognition

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 [20]–[23] is commonly regarded as perceptual saliency. Saliency-based or "bottom-up" guidance of attention highlights a limited number of possible points of interest in an image, which would be useful [24] in selecting landmarks that are 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.

Concurrent with the mechanisms of saliency, humans also exhibit the ability to rapidly summarize the "gist" of a scene [25]–[27] in less than 100ms. Human subjects are able to consistently answer detailed inquiries such as the presence of an animal in a scene [28], [29], 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 [30], [31]. It is reported that gist computations may occur in brain regions which respond to "places", that is, prefer scenes that are notable by their spatial layout [32] as opposed to objects or faces. In addition, gist perception is affected by spectral contents and color diagnosticity [33], which leads to the implementation of models such as [34], [35].

In spite of how contrasting saliency and gist are, both modules rely on 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 [36]. From an engineering perspective it is an effective strategy to analyze a scene from opposite coarseness levels, a high-level, image-global layout (corresponding to gist) and detailed pixel-wise analysis (saliency). Also, note that, while saliency models primarily utilize local features [23], gist features are almost exclusively holistic [6], [18], [33]. Our presented model (figure 1) seeks to employ the two complementary concepts of biological vision, implemented faithfully and efficiently, to produce a critical capability such as localization.

After early preprocessing at both retina and LGN (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 fed to saliency and gist modules. Along the Dorsal Pathway or "where" visual processing stream [37] (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 conscious decisions and motor commands are formed. In this paper, we concentrate mostly on the biologically-inspired localization computations of the ventral pathway.

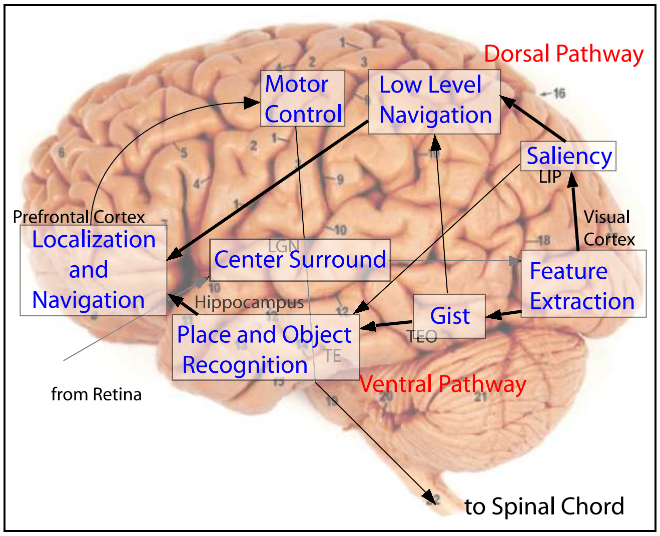

Fig. 1. A sketch of the full system with each sub-system projected onto anatomical locations that may putatively play similar roles in human vision.

C. Topological Maps

In addition to biological vision, our utilization of topological maps also draws from various human experiments. A topological map [38], [39], which refers to a graph annotation of an environment, assigns nodes to particular places and edges as paths if direct passage between pairs of places (end-nodes) exist. One of the distinct ways humans manage spatial knowledge is by relying more on topological information than metric. That is, although humans cannot estimate precise distances or directions [40], they can draw a detailed and hierarchical topological (or cognitive) map to describe their environments [41]. Nevertheless, approximate metric information is still deducible and is quite useful. In addition, the amount of added information is not a heavy burden (in terms of updating and querying) for the system, because of the concise nature of a basic graph organization. This is in sharp contrast to a more traditional metric grid map in robotics localization literature [1], [9], where every area in the map is specified for occupancy, as opposed to being assumed untraversable if not specified as places or paths.

In our system, as well as a number of others [38], [42], we use an augmented topological map with directed edges. The map has an origin and a rectangular boundary, and each node has a Cartesian coordinate. In addition, each edge has a cost, which is set to the distance between the corresponding end-nodes. This way the system benefits from the compact representation of a graph while preserving the important metric information of the environment. The robot state (position and

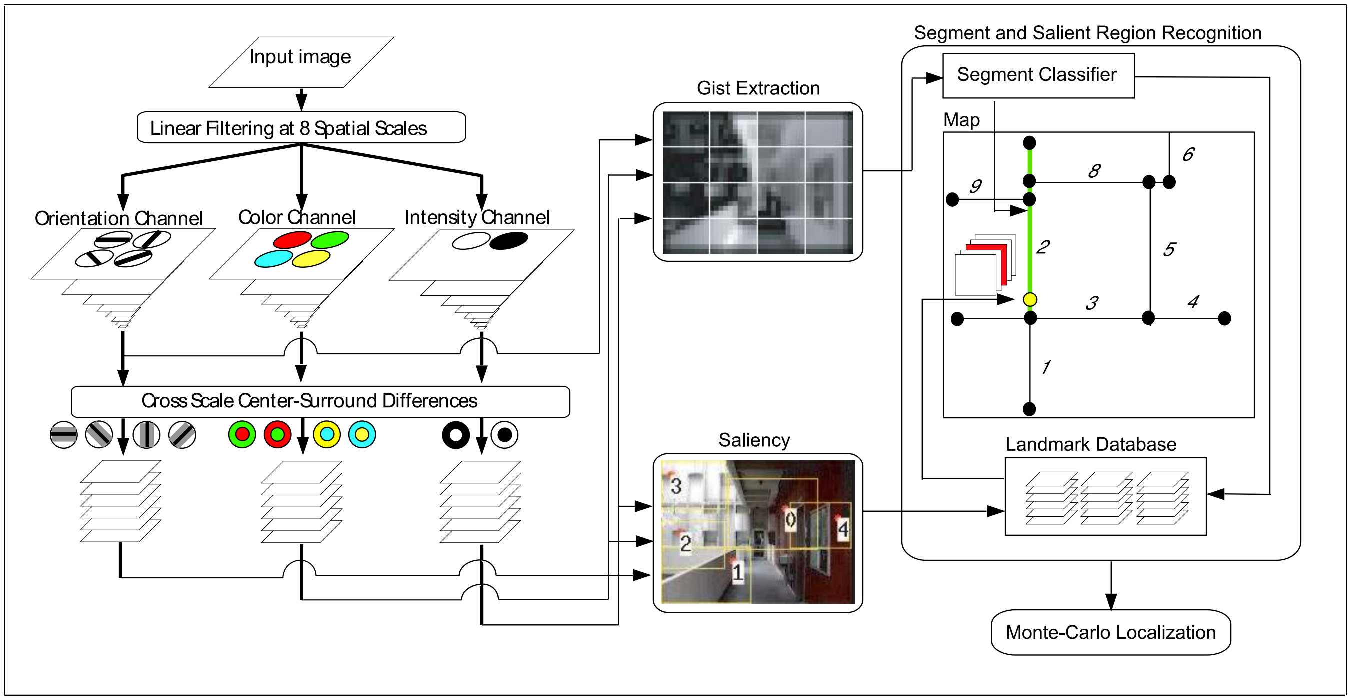

Fig. 2. Diagram for the Vision Localization System. From an input image the system extracts low-level features consisting of center-surround color, intensity, and orientation that are computed in separate channels. They are then further processed to produce gist features and salient regions. We then compare them with previously obtained environment visual information. The results are used to estimate the robot's location.

viewing direction) is represented by a point which can lie on a node or an edge.

It should be noted that various parts of our system, such as the localization module (we use a standard probabilistic approach [1], [9], [10]) may not be biologically plausible. This is why we simply claim that the system is biologically inspired. Our philosophy is that although we are committed to studying biological systems (human vision in particular), we also would like to build systems that are useful in the real world now. We see this dual intention as a two-way street, where engineering ideas can help bring inspiration to explain scientific phenomena, not just the other way around in building neuromorphic robots.

II. Design And Implementation

In this paper we describe our biologically inspired vision localization system. We have reported in [18] our gist-based place recognition system, which is only a part of the presented system. We define the gist features as a low-dimensional vector (compared to raw image pixel array) that represents a scene and can be acquired over very short time frames. Place classification based on gist then becomes possible if and when the vector can be reliably classified as belonging to a given place. In the presented system, we also utilized salient landmarks obtained from the attention system to refine the place estimation to a more accurate metric localization. Previously [43], we reported a preliminary result. Here, the original contribution is explaining the system in more detail (especially the salient landmark acquisition and recognition) and, more importantly, rigorously testing it in multiple challenging outdoor environments at various times of the day to demonstrate its lighting invariance. In addition, we also test the individual modules within the system — salient region recognition (a local-feature system) and gist-based localization — to gauge their contributions to the end result.

The localization system (illustrated in figure 2) is divided into 3 stages: feature extraction, recognition, and localization. The first takes a camera image and outputs gist features and salient regions. In the next stage, we compare them with memorized environment visual information. These matches are input to the localization stage to decide where the robot is.

The term salient region refers to a conspicuous area in an input image depicting an easily detected part of the environment. An ideal salient region is one that is persistently observed from different points of view and at different times of the day. A salient region does not have to isolate a single object (often times it is part of an object or a jumbled set of objects), it just has to be a consistent point of interest in the real world. To this end, the set of salient regions that portray the same point of interest are grouped together and the set is called a landmark. Thus, a salient region can be considered as an evidence of a landmark and "to match a salient region with a landmark," means to match a region with the landmark's saved regions. It is also important to note that the process of discovering salient regions is done using biological computations, but the process of region matching is not. We use SIFT keypoints [12] because they are the current gold standard for pattern recognition.

Within the augmented topological map we group an area in the environment as a segment. A segment is an ordered list of edges with one edge connected to the next to form a continuous path. This grouping is motivated by the fact that views/layout in one path-segment are coarsely similar.

An example is the selected three-edge segment (highlighted in green) in the map in figure 2. Geographically speaking, a segment is usually a portion of a hallway, path, or road interrupted by a crossing or a physical barrier at both ends for a natural delineation. The term segment is roughly equivalent to the generic term “place” for place recognition systems (mentioned in section I-A), which refer to a general vicinity of an environment. With this, the robot location can be noted as both Cartesian coordinate (x, y) or a pair of segment number $snum$ and the fraction of length traveled (between 0.0 to 1.0) along the path $ltrav$.

In the following sub-sections we will describe the details of each of the three stages in its order of operation.

A. Feature extraction: Gist and Salient Regions

The shared raw low-level features (which emulate the ones found in the visual cortex) for gist [18] and saliency [22], [44] models are filter outputs computed in color, intensity, and orientation channels. Within them, there are sub-channels to account for sub-categories: color opponencies (in color channel), degree orientation (orientation channel), intensity opponency (intensity channel). Each sub-channel has a nine-scale pyramidal representation of filter outputs. Within each sub-channel, the model performs center-surround operations (commonly found in biological-vision which compares image values in center-location to their neighboring surround-locations) between filter output maps at different scales in the pyramid. These center-surround maps (also called feature maps) are then fed into both gist and saliency modules.

1) Gist Feature Extraction: The gist model [18] computes average values (biologically plausible accumulation operations) from 4-by-4 grid sub-regions of the feature maps. Figure 2 illustrates gist extraction on an intensity feature map. By doing so, we encode information from various visual domains with a small number of values, while still taking into account coarse spatial information. The raw gist feature dimension is 544: 34 feature maps (from all sub-channel center-surround combinations) times 16 regions per map.

2) Salient Region Selection and Segmentation: The saliency model [22], on the other hand, uses the feature maps to detect conspicuity regions in each channel. It first performs a linear combination (simple unweighted pixel-wise addition) between feature maps within each channel to produce conspicuity maps (one per channel). The model then combines the maps through winner-take-all mechanisms, which emphasize locations that substantially differ from their neighbors, to yield a saliency map. We then further process the saliency map to produce a set of salient regions (figure 3).

The system starts at the pixel location of the saliency map’s highest value. To extract a region that includes the point, we use a shape estimator algorithm [45] (region growing with adaptive thresholding) to segment the feature map that gives rise to it. To find the appropriate feature map, we compare the values of the conspicuity maps at the salient location and select the channel with the highest value (this is the winning channel). Within the winning channel, we compare values at the same location for all the feature maps. The one with the highest value is the winning center-surround map.

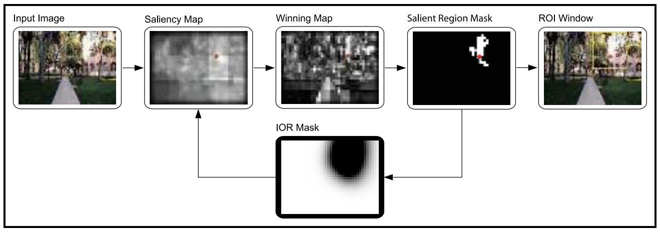

Fig. 3. A salient region is extracted from the center-surround map that gives rise to it. We use a shape estimator algorithm to create a region-of-interest (ROI) window and use inhibition-of-return (IOR) in the saliency map to find other regions.

The system then creates a bounding box around the segmented region. Initially, we fit a box in a straight-forward manner: find smallest-sized rectangle that fits all connected pixels. The system then adjusts the size to between 35% and 50% in both the image width and height, if it is not yet within the range. This is because small regions are hard to recognize and overly large ones take too long to match. In addition, the system also creates an inhibition-of-return (IOR) mask to suppress that part of the saliency map to move to subsequent regions. This is done by blurring the region with a Gaussian filter to produce a tapering effect at the mask’s border. Also, if a new region overlaps any previous regions by more than 66%, it is discarded but is still suppressed.

We continue until 1 of 3 exit conditions occur: unsegmented image area is below 50%, number of regions processed is 5, and the saliency map value of the next point is lower than 5% of the first (most salient). We limit the regions to 5 because, from experiments, subsequent regions have a much lower likelihood of being repeatable in testing. Figure 4 shows extraction of 5 regions. There are reasons why multiple regions per image is better. First, additional perception (there are many salient entities within the field of view) contributes to a more accurate localization, given the possibility of occlusion in an image. Second, the first region may be coincidental or a distraction. In figure 4, the first one returned is a ray of sunshine hitting a building. Although from the saliency perspective, it is correct, it is not a good location cue. The second region is better because it depicts details of a building.

B. Segment and Salient Region Recognition

This stage attempts to match the visual stimuli (salient regions and gist features) with stored environment information. The results are then used to localize at the next stage. The system acquires the information through two training steps: building a landmark database and training a segment classifier using gist features. The procedure involves a guided traversal of the robot through all the paths in the map. As the robot moves about the environment, we store the salient regions found along with the corresponding robot locations when they are discovered. We perform the traversal several times for ample lighting coverage. At the same time, we also store the gist features from each input frame for segment classification training. To determine how many segments to classify, we group the edges according to view similarity by a human operator estimation. The operator uses a simple heuristic: start

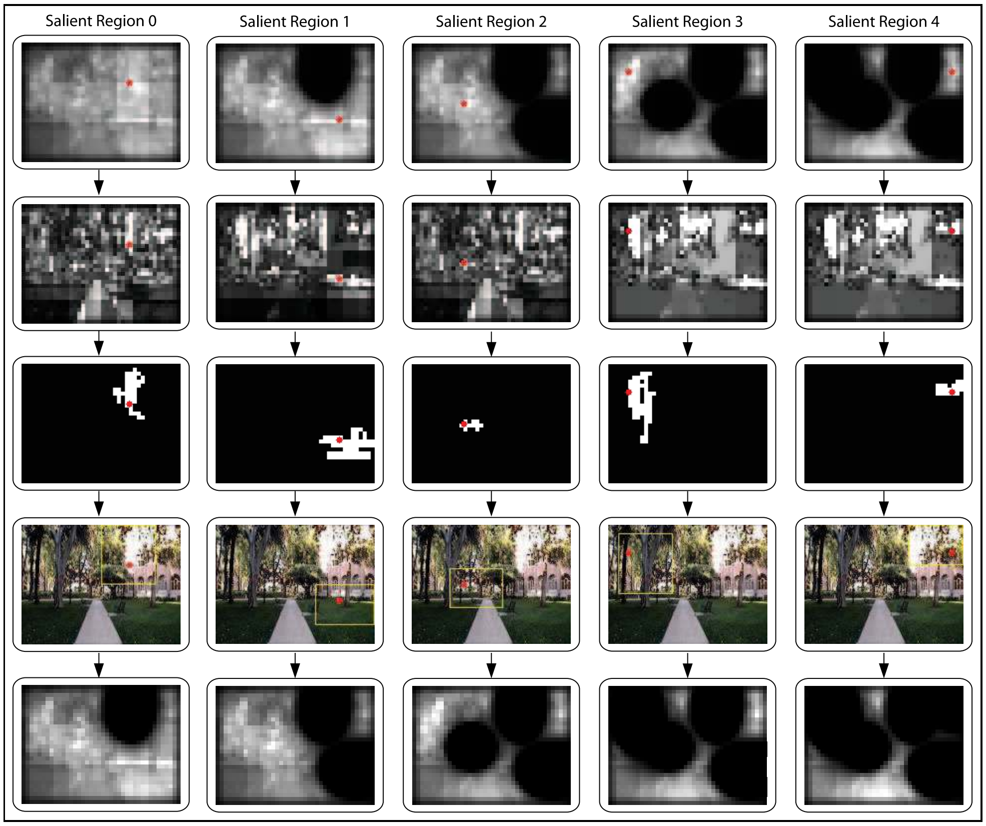

Fig. 4. Process of obtaining multiple salient regions from a frame, where the IOR mask (last row) dictates the shift in attention of the system to different parts of the image.

a new segment and stop the previous one when an abrupt visual change occurs. This is usually because an intersection is reached and the robot is turning in place to another direction.

The following sub-sections describe the run-time matching process, and formulate the output for our back-end Monte Carlo localization (MCL) module [1], [9], [10]. Within the MCL framework, we need to provide observation models to weight the likelihood of a particular observation to occur in a given state. The system observes two types of evidence: segment classification and matched salient regions.

1) Segment Classification: The segment estimator is implemented as a 3-layer neural network classifier trained using the back-propagation algorithm on gist features that have already undergone PCA/ICA dimension reduction [18]. One of the main reasons why the classifier succeeds is because of the decision to group edges into segments. It would have been difficult to train an edge-classifier using coarse features like gist as adjacent edges that are part of the same segment usually are moving toward the same general direction and thus tend to share a lot of the background scene. Each segment in the environment has an associated classifier 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 as an observation where

$$z_t^{'} = \{ \ sval_{t,j} \ \} \ j = 1 \ ... \ N_{segment} \quad (1)$$

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

2) Salient Region Recognition: In order to recall the stored salient regions we have to find a robust way to recognize them. We use two sets of signatures: SIFT keypoints [12] and salient feature vector. We employ a straight-forward SIFT recognition system [12] (using all the suggested parameters and thresholds) but consider only regions that have more than 5 keypoints to ensure that the match is not a coincidence.

A salient feature vector [43] is a set of values taken from a 5-by-5 window centered at the salient point location (yellow disk in figure 5) of a region $sreg$. These normalized values (between 0.0 to 1.0) come from the sub-channels' feature maps [22], [44] for all channels (color, intensity, and orientation). In total, there are 1050 features (7 sub-channels times 6 feature maps times 5x5 locations). Because the feature maps are produced in the previous feature extraction step (section II-A), even though they are computed over the entire image for each visual domain, from the salient feature vector perspective, they come at almost no computational cost.

To compare salient feature vectors from two salient regions $sreg_1$ and $sreg_2$, we factor in both feature similarity $sfSim$ (equation 2) and salient point location proximity $sfProx$ (equation 3). The former is based on the Euclidian-distance in feature space:

$$sfSim(sreg_1, sreg_2) \ = \ 1 - \frac{\sqrt{\sum_{i=1}^{N_{sf}} \ (sreg_{1,i} - sreg_{2,i})^2}}{N_{sf}} \quad (2)$$

N_{sf} is the total number of salient features. For a match to be confirmed, the feature similarity has to be above .75 out of the maximal 1.0. The location proximity $sfProx$, on the other hand, is the Euclidian distance in pixel units (denoted by the function $dist$), normalized by the image diagonal length:

$$sfProx(sreg_1, sreg_2) \; = \; 1 - \frac{dist(sreg_1, sreg_2)}{l_{Diagonal}} \qquad (3)$$

The positive match score threshold for the distance is 95% (within 5% of input image diagonal). Note that the proximity distance is measured after aligning $sreg_1$ and $sreg_2$ together, which is after a positive SIFT match is ascertained (observe the fused image in figure 5). The SIFT recognition module estimates a planar (translational and rotational) transformation matrix [12] that characterizes the alignment. In short, individual reference-test keypoint pairs are first compared based on the descriptor's similarity. Each matched pair then "votes" for possible 2D affine transforms (there is no explicit notion of an object location in 3D space) that relate the two images. An outlier elimination is performed using the most likely transform given all matches. Using the remaining pairs, we compute a final affine transform. With this matrix, the system can check the alignment disparity between the two regions' salient point location.

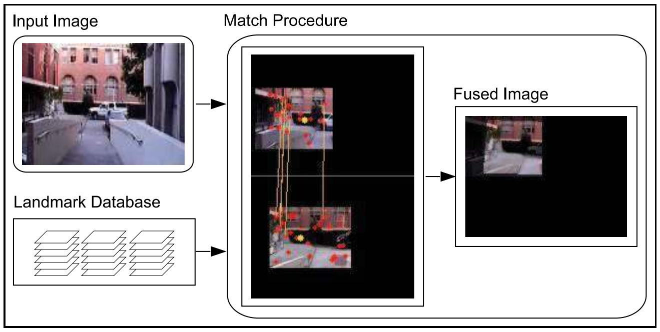

Fig. 5. Matching process of two salient regions using the SIFT keypoints (drawn as red disks) and salient feature vector, which is feature map values taken at the salient point (drawn as the yellow disk). The lines indicate the correspondences that are found. The fused image is added to show that we also estimate the pose change between the pair.

Once the incoming salient regions are compared with the landmark database, the successful matches (ones which pass both salient feature vector and SIFT match thresholds described above) are denoted as observation $z_t^{''}$, where

$$z_t^{''} = \{ \; omatch_{t,k} \; \}, k = 1 \; ... \; M_t \qquad (4)$$

with $omatch_{t,k}$ being the k-th matched database salient region at time t. M_t denotes the total number of positive matches at time t. Note that the recognition module may not produce an observation for every time t, it is possible that it finds no matches, $M_t = 0$.

C. Monte-Carlo Localization

We estimate robot position by implementing Monte-Carlo Localization (MCL) which utilizes Sampling Importance Resampling (SIR) [1], [9], [10]. We formulate the location belief state S_t as a set of weighted particles: $S_t = \{x_{t,i}, \; w_{t,i}\}$ $i = 1 \; ... \; N$ at time t and N being the number of particles. Each particle (possible robot location) $x_{t,i}$ is composed of a segment number $snum$ and percentage of length traveled $ltrav$ along the segment edges, $x_{t,i} = \{snum_{t,i}, \; ltrav_{t,i}\}$. Each particle has a weight $w_{t,i}$, which is proportional to the likelihood of observing incoming data modeled by the segment and salient region observation model (explained in sections II-C2 and II-C3 below, respectively). Note that the segment observation is applied before salient region observation because segment estimation can be calculated almost instantaneously while the salient region matching is much slower. We have not tried it, but, if the order of application is reversed, we believe that the results would be similar given that the observations are integrated over time. From experiments, $N = 100$ suffices for the simplified localization domain where a hallway is represented by an edge and not a two dimensional space. We tried N as high as 1000 with unnoticeable performance or computation speed change. With $N = 50$ the performance starts to degrade, namely in kidnapped robot instances. We estimate the location belief $Bel(S_t)$ by recursively updating posterior $p(S_t|z^t, u^t)$ — z_t being an evidence and u_t the motion measurement using [46]:

$$\begin{aligned} Bel(S_t) &= p(S_t|z^t, u^t) && (5) \\ &= \alpha p(z_t|S_t) \int_{S_{t-1}} p(S_t|S_{t-1}, u_t) Bel(S_{t-1}) \, dS_{t-1} \end{aligned}$$

We first compute $p(S_t|S_{t-1}, u_t)$ (called the prediction/proposal phase) to take robot movement into account 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 models — segment estimation $z_t^{'}$ (eqn. 1) and matched salient regions $z_t^{''}$ (eqn. 4) — to each particle for weighted resampling steps. The following algorithm shows the order in which the system computes belief estimation $Bel(S_t)$ at each time step t:

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 salient region observation model to $S_t^{''}$ to yield S_t
    b) else $S_t = S_t^{''}$

Here, we specify two intermediate states: $S_t^{'}$ and $S_t^{''}$. $S_t^{'}$ is the belief state after the motion model is applied to the particles. $S_t^{''}$ is the state after the segment observation (first step of update phase $p(z_t|S_t)$) is subsequently applied to $S_t^{'}$. Segment observation application is done by weighted resampling using likelihood function $p(z_t^{'}|x_{t,i}^{'})$ (equation 6 below) as weights. This function denotes the likelihood that a segment estimation $z_t^{'}$ is observed at location $x_{t,i}^{'}$. Afterwards, the salient region observation model (second step of update phase $p(z_t|S_t)$) is applied to the belief state $S_t^{''}$ to produce S_t. This is done with weighted resampling using the likelihood function $p(z_t^{''}|x_{t,i}^{''})$ (equation 7 below) as weights, representing the likelihood that salient region match $z_t^{''}$ is found at $x_{t,i}^{''}$.

1) Motion Model: The system employs a straightforward motion model to each particle $x_{t-1,i}^{'}$ in S_{t-1} by moving it

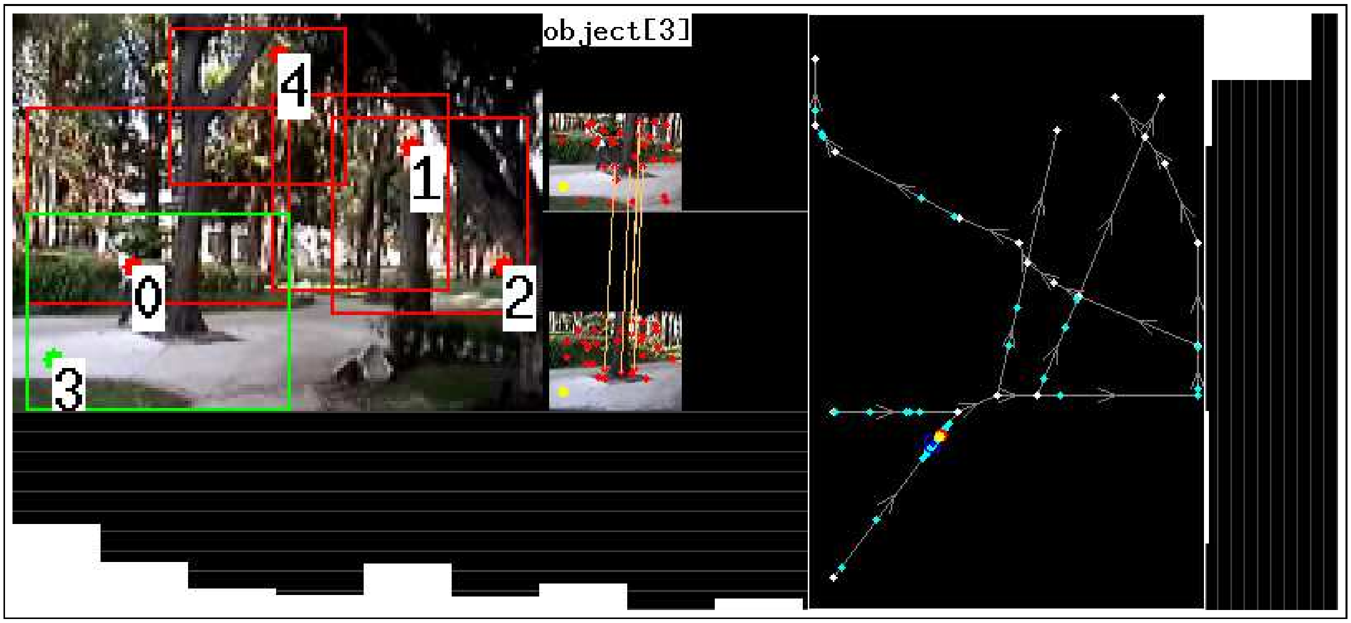

Fig. 6. A snapshot of the system test-run. Top-left (main) image contains the salient region windows. Green window means a database match, while red is not found. A salient region match is displayed next to the main image. Below the main image is the segment estimation vector derived from gist (there 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 salient region, 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.

with the distance traveled (odometry reading u_t) plus noise to account for uncertainties such as wheel slippage. We model this by drawing a particle $x'_{t,i}$ from a Gaussian probability density $p(x'_{t,i}|u_t, x_{t-1,i})$, where the mean is the robot location in the absence of noise and standard deviation of .1ft (about 1/6th of a typical single step). The latter controls the level of noise in the robot movement measurement. From our experiments, we find that this number does not affect the end result as much because the neighborhood of particles around a converged location (observe the belief map in figure 6) is large enough that motion error in any direction is well covered.

In the procedure, the distribution spawns a new location by only changing the length traveled $ltrav$ portion of a particle $x'_{t,i}$. It is then checked for validity with respect to the map as $ltrav$ has a range of 0.0 to 1.0. If the value is below 0.0, then the robot has moved back to a previous segment in the path, while if it is above 1.0, the robot has moved to a subsequent segment. We take care of these situations by changing the segment $snum$ and normalizing the excess distance (from the end of original segment) to produce a corresponding $ltrav$. If the original segment ends in an intersection with multiple continuing segments, we simply select one randomly. If no other segment extends the path, we just resample.

2) Segment-Estimation Observation Model: This model estimates the likelihood that the gist feature-based segment estimation correctly predicts the assumed robot location. So, we weigh each location particle $x'_{t,i}$ in S'_t 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. We take into account the segment-estimation vector z'_t by using:

$$p(z'_t|x'_{t,i}) = \frac{sval_{t,snum'_{t,i}}}{\sum_{j=1}^{N_{segment}} sval_{t,j}} * sval_{t,snum'_{t,i}} \qquad (6)$$

Here, the likelihood that a particle $x'_{t,i}$ observes z'_t is proportional to the percentage of estimation value of the robot's segment location $sval_{t,snum'_{t,i}}$ over the total estimation value (first term) times the robot segment location value (second term). The rationale for the first term is to measure the segment's dominance with respect to all values in the vector; the more dominant the more sure we are that the segment estimation is correctly predicting the particle's segment location. The second term preserves the ratio of the robot segment location value with respect to maximum value of 1.0 so that we can make a distinction of confidence level of the segment estimation prediction. Note that the likelihood function only makes use of the segment $snum'_{t,i}$ information from particle $x'_{t,i}$, while $ltrav'_{t,i}$ is left unused as the precise location of the robot within the segment does not have any effect on segment estimation.

3) Salient-Region-Recognition Observation Model: In this model we want to measure the likelihood of simultaneously observing the matched salient regions given that the robot is at a given location. We weigh each particle $x''_{t,i}$ in S''_t with $w''_{t,i} = p(z''_t|x''_{t,i})$ for resampling (with added 20% random noise, also to combat population degeneracy) to create belief S_{t+1} by taking into account the salient region matches z''_t using:

$$p(z''_t|x''_{t,i}) = \prod_{k=1}^{M_t} p(omatch_{t,k}|x''_{t,i}) \qquad (7)$$

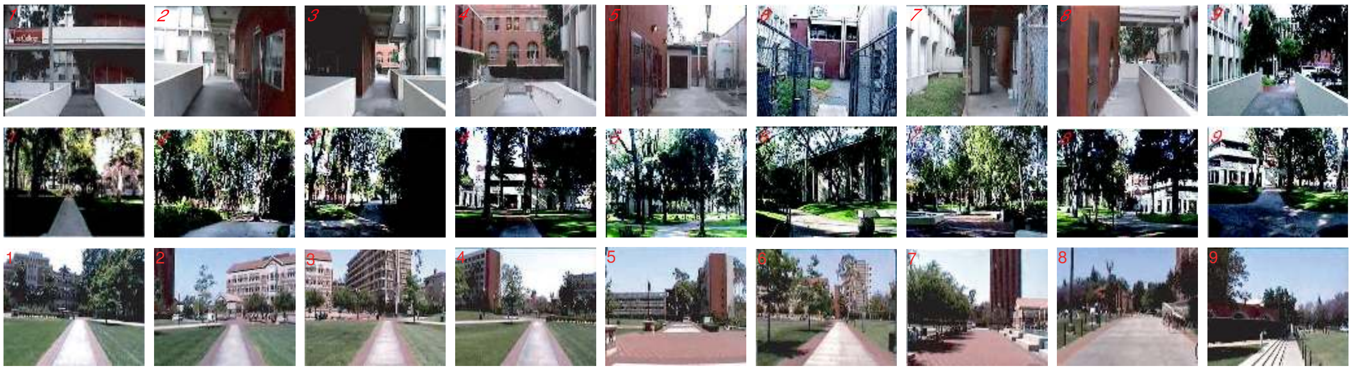

Fig. 7. Examples of images in each of the nine segments (with corresponding label) of ACB (first row), AnFpark (second row), and FDFpark (third row)

Given that each salient-region match observation is independent, we simply multiply each of them to calculate the total likelihood. The probability of an individual match $p(omatch_{t,k}|x''_{t,i})$ is modeled by a Gaussian with the standard deviation σ set to 5% of the environment map's diagonal. The likelihood value is the probability of drawing a length longer than the distance between the particle and the location where the matched database salient region is acquired. σ is set proportional to the map diagonal to reflect how the larger the environment, the higher the level of uncertainty. The added noise is twice that of segment observation because the salient region observation probability density is much narrower and we find that 20% keeps the particle population diverse enough to allow for dispersion and correct re-convergence in a kidnapped robot event. Also, although the SIFT and salient feature vector matching scores (explained in section II-B2 above) are available for weights, we do not use them in the likelihood function directly. These matching scores were thresholded to come up with the positive salient region matches we are now considering in this section. We do not reason with match quality because the thresholds alone eliminate most false positives.

Figure 6 illustrates how the system works together.

III. Testing and Results

We test the system at 3 sites (each has 9 segments) on campus with example scenes of each occupying a row of figure 7. The same data is used to test the gist model [18] in segment classification. In this work we localize to a coordinate location within the map. The first site is the 38.4x54.86m Ahmanson Center for Biological Research (ACB) building complex (first row of figure 7). Most of the surroundings are flat walls with little texture. The second site (second row) is a 82.3x109.73m area comprising two adjoining parks full of trees: Associate and Founders park (AnF). The third testing (third row) site is the Frederick. D. Fagg park (FDF), a 137.16x178.31m open area where large portions of the scenes are the sky.

We also compare our system, which employs both local features (SIFT keypoints within salient regions and salient feature vector at the salient point) as well as global (gist) features with two systems that use only salient regions or only gist features. The back-end Monte-Carlo localization modules in all three instances are identical. For the SIFT-only system, we take out the salient feature vector from the region signature to end up with only SIFT features. Also, in [47] we have compared our gist system with other place recognition systems and found that the results are comparable. Thus, the gist-only localization comparison may also be indicative of what place recognition systems can do in a metric localization task.

The visual data is gathered using an 8mm handheld camcorder carried by a person. There is no camera calibration or lens distortion correction which may help in salient region matching. Because the data is recorded at approximately constant speed and we record clips for individual segments separately, we use interpolation to come up with the ground-truth location. Also, the map (edge lengths and node locations) is currently constructed manually. With this, we calculate the walking velocity using the distance of a particular path divided by the amount of time it took for the person to traverse it (identical to the clip duration). We can place the location of the start and end of the clip because they are prespecified. For the frame locations in between, we assume a uniform capture interval to advance the person's location properly. In all experiments, a denoted error signifies a measured difference (in feet) between the robot belief and this generated ground truth location. To roughly mimic odometry noise such as slippage, we add zero-mean Gaussian noise with a standard deviation 1/6 the average walking speed for each site.

The main issue in collecting training samples is filming time selection that includes all lighting conditions. Because lighting space is hard to gauge, we perform trial-and-error to come up with the times of day (up to 6 per day): from the brightest (noon time) to the darkest (early evening). Note that 10 of 12 of the testing clips are taken at a different date than the training clips. As for the two other clips, the testing data was recorded in the early evening (dark lighting) while training data was taken near noon (bright lighting). In all, there are 26,368 training and 13,966 testing frames for the ACB cite, 66,291 training and 26,387 testing frames for the AnF site, and 82,747 training and 34,711 testing frames for the FDF site.

Currently, we test the system offline on a 16-core 2.6GHz machine, operating on 160x120 images. We time individual sub-modules and find that the slowest part by far is the salient region recognition process (3 seconds on average). This is in spite of a parallel search implementation using 16 dispatched threads that compare input regions with different parts of the landmark database. The gist and saliency computation time (also implemented in parallel where each sub-channel has its own thread) is about 20ms. In addition, the salient region acquisition (windowing) takes 10ms, while the segment estimation takes less than 1ms. The back end localization itself takes less than 1ms as it only uses 100 particles.

A. Experiment 1: Ahmanson Center for Biological Research (ACB)

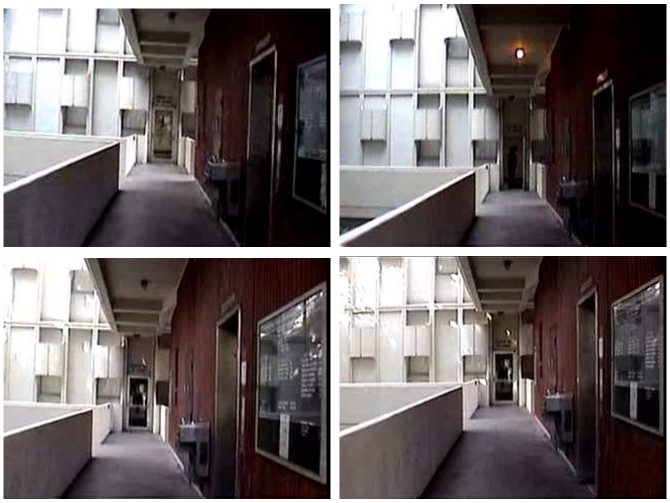

Fig. 8. Lighting conditions used for testing at Ahmanson Center for Biology (ACB). Clockwise from top left: late afternoon (trial 1), early evening (trial 2), noon (trial 4) , and mid-afternoon (trial 3)

This experiment site is chosen to investigate what the system can achieve in a rigid and less spacious man-made environment. Each segment (scenes displayed in first row of figure 7) is a straight line and part of a hallway. Figure 8 depicts different lighting conditions that are tested: late afternoon (trial 1), early evening with the lights already turned on (2), mid-afternoon (3), and noon (4).

Table I shows the result with an overall error of 0.98m In general, the error is uniformly distributed across segments, although spikes in segments 2 and 5 are clearly visible. The error rate for segment 2, which comes from trials 1, 2, and 4, occurred because the identified salient regions (mainly the textured white building and its entrance door in figure 8) are at the end of the hallway and they do not change sizes as much even after a 3m robot displacement. It is also the case for the error spike in segment 5 for trial 4, as the system latches to a water tower (fifth image of the first row of figure 7).

The errors in segment 5 from trials 3 and 4 (bright lighting) partially originate from the camera's exposure control that tries to properly normalize the range of frames with wide intensity contrast (the scenes are comprised of very bright sky and dark buildings) and it ends up darkening the building for a few seconds — something to consider when selecting a camera to film outdoor scenes. During this time, the segment estimator produces incorrect values and the SIFT module is unable to recognize any regions in the image, which throws off the robot belief completely. It seems that for the system to fail, all parts (saliency, SIFT, and gist matching) have to fail.

B. Experiment 2: Associates and Founders Park (AnF)

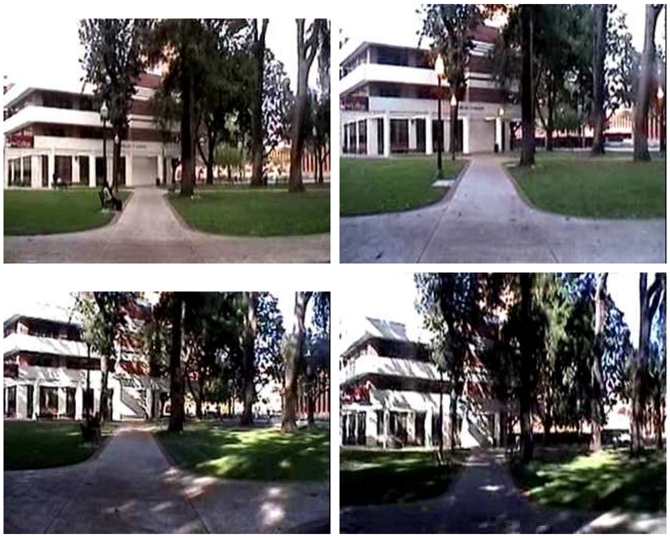

Fig. 9. Lighting conditions used for testing at Associate and Founders park (AnF). Clockwise from top left: overcast (trial 1), early evening (trial 2), noon (trial 4), and mid-afternoon (trial 3)

We compare experiment 1 results with, conceivably, a more difficult vegetation-dominated site (scenes shown in the second row of figure 7) that also has longer paths (about twice the lengths of ACB segments). Figure 9 shows four lighting conditions tested: overcast (trial 1), early evening with lights already turned on (2), mid-afternoon (3), and noon (4). As we can see in the images, there are fewer rigid structures and the few object that exist in the environment (lamp posts and benches) tend to look small with respect to the image size. Also, objects can either be taken away (e.g. the bench in the top right image in figure 9) or added such as service vehicles parked or a large storage box placed in the park for a day. In addition, whole scene matching using local features would be hard because the tree leaves produce high numbers of random texture-like patterns that significantly contaminate the process.

The results (table II) reveal an overall error of 2.63m but with noticeably higher performance disparity between segments. The errors are also different across trials for which segment produces high displacements. On average (last column of the table) though, all segments have roughly equal errors. Between trials, the error difference between the two dim lighting trials (3 and 4) and the bright lighting trials (1 and 2) is significant. It seems that low lighting, or more importantly the lack of unpredictable and ephemeral sunlight (observe the grass in the bottom two images of figure 9), allows for uniform lighting and better correlation between training and testing runs. In the end, although the results are worse than experiment 1, it is quite an accomplishment given the challenges presented by the scenes and no by-hand calibration is done in moving from the first environment to the second.

C. Experiment 3: Frederick D. Fagg park (FDF)

The third site is the Frederick D. Fagg park, an open area used to assess the system's response on sparser scenes (third row of figure 7) and in an even larger environment (the

TABLE I
AHMANSON CENTER FOR BIOLOGY EXPERIMENTAL RESULTS

	Trial 1		Trial 2		Trial 3		Trial 4		Total	
Segment	number frames	error (m)	number frames	error (m)	number frames	error (m)	number frames	error (m)	number frames	error (m)
1	387	0.96	410	1.02	388	0.73	411	0.75	1596	0.87
2	440	1.87	436	2.87	461	0.70	438	1.66	1775	1.76
3	465	1.06	485	0.69	463	0.89	474	1.35	1887	1.00
4	359	0.99	321	0.96	305	1.00	249	0.98	1234	0.98
5	307	1.17	337	0.62	321	1.77	319	1.96	1284	1.37
6	556	0.60	495	1.15	534	0.75	502	0.56	2087	0.76
7	438	0.48	445	0.60	398	0.85	400	0.82	1681	0.68
8	290	0.59	247	1.14	274	0.77	288	0.88	1099	0.83
9	341	0.66	373	0.50	313	0.60	296	0.59	1323	0.59
Total	3583	0.93	3549	1.08	3457	0.87	3377	1.06	13966	0.98

TABLE II
ASSOCIATE AND FOUNDERS PARK EXPERIMENTAL RESULTS

	Trial 1		Trial 2		Trial 3		Trial 4		Total	
Segment	number frames	error (m)	number frames	error (m)	number frames	error (m)	number frames	error (m)	number frames	error (m)
1	698	1.21	802	1.88	891	4.19	746	1.76	3137	2.36
2	570	2.40	328	1.90	474	5.76	474	1.90	1846	3.05
3	865	1.61	977	3.32	968	2.01	963	4.65	3773	2.93
4	488	3.20	597	1.73	688	1.57	632	2.85	2405	2.28
5	617	3.34	770	1.33	774	1.70	777	3.36	2938	2.39
6	1001	1.55	1122	1.80	1003	3.28	1098	3.38	4224	2.50
7	422	1.09	570	4.01	561	2.45	399	2.80	1952	2.68
8	598	2.52	692	3.11	797	2.21	768	1.68	2855	2.35
9	747	2.14	809	1.66	862	3.54	849	5.04	3267	3.14
Total	6006	2.06	6667	2.29	7018	2.89	6706	3.21	26397	2.63

segments are about 50% longer than the ones in the AnF experiment, three times that of ACB). Figure 10 represents the 4 lighting conditions tested: late afternoon (trial 1), evening (2), noon (3), and mid-afternoon (4).

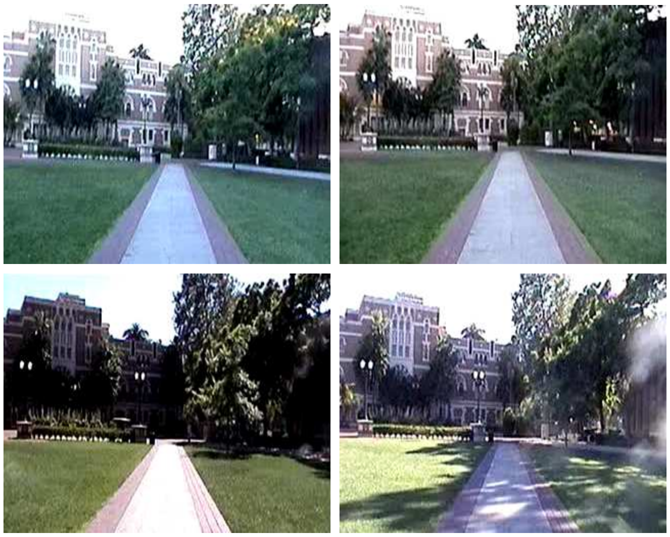

Fig. 10. Lighting Conditions use for Testing at Frederick D. Fagg park (FDF). Clockwise from top left: late afternoon (trial 1), evening (trial 2), noon (trial 4), and middle of afternoon (trial 3).

Table III shows the results, listing an overall error of 3.46m, worse than the other two sites. It seems that an increase in environment size affects the results. However, the more direct cause is scale. Currently, the system uses the location of where the matched database salient region is found as a hypothesis of where the robot currently is. Because the SIFT module can perform scale-invariant matching (with the scale ratio included as part of the result), the system limits the matching-scale threshold to between 2/3 and 3/2. This is not entirely effective as a scale ratio of 0.8 (the region found is smaller than the one matched in the database) can translate to a geographical difference of 5m. This is because, in this environment, far away buildings are salient and, as the robot moves toward them, their appearance hardly changes. Thus, although these are stable localization cues, they are not good for fine-grained location pin-pointing. We would need closer ($< 3m$ away) regions.

One encouraging point is that the system seems to be able to cope with a variety of lighting conditions. The results are better than the preliminary results [43] because of better lighting coverage in training despite the fact that training and testing are done on separate days. In this site, for example, we have dark (trial 1 and 2) and bright (trials 3 and 4) conditions, even with long shadows cast on the field (trial 4 scene in figure 10).

TABLE III
FREDERICK D. FAGG PARK EXPERIMENTAL RESULTS

Segment	Trial 1		Trial 2		Trial 3		Trial 4		Total	
	number frames	error (m)	number frames	error (m)	number frames	error (m)	number frames	error (m)	number frames	error (m)
1	881	1.44	670	1.98	847	2.88	953	1.41	3351	1.90
2	788	6.57	740	4.92	797	2.30	878	3.99	3203	4.42
3	858	3.45	696	4.12	922	1.49	870	2.14	3346	2.71
4	837	4.54	740	4.28	837	1.97	821	4.59	3235	3.83
5	831	3.42	748	3.78	694	4.69	854	3.03	3127	3.68
6	1680	5.52	1565	3.84	1712	3.24	1672	3.79	6629	4.10
7	1037	3.44	923	2.97	857	3.34	894	3.35	3711	3.28
8	1172	4.94	1211	3.22	1355	2.19	1270	3.36	5008	3.38
9	739	3.03	825	2.73	794	3.67	743	3.75	3101	3.29
Total	8823	4.18	8118	3.54	8815	2.82	8955	3.29	34711	3.46

TABLE IV
MODEL COMPARISON EXPERIMENTAL RESULTS

System	ACB				AnF				FDF			
	Trial 1	Trial 2	Trial 3	Trial 4	Trial 1	Trial 2	Trial 3	Trial 4	Trial 1	Trial 2	Trial 3	Trial 4
	err. (m)	err. (m)	err. (m)	err. (m)	err. (m)	err. (m)	err. (m)	err. (m)	err. (m)	err. (m)	err. (m)	err. (m)
gist	7.81	7.37	9.12	6.09	13.09	18.24	20.12	14.28	23.96	26.12	24.46	27.25
SIFT	1.60	1.69	1.92	1.67	2.70	2.99	3.46	3.70	4.58	4.96	3.89	4.73
bio-system	0.93	1.08	0.87	1.06	2.06	2.29	2.89	3.21	4.18	3.54	2.82	3.29

D. Experiment 4: Sub-module Analysis

Table IV shows a comparison of systems that use only local features (SIFT), only global features (gist features), and the presented bio-system, which uses both global and local features. The gist-only system cannot localize to the metric level because it can only pin-point location to the segment level and some segments have lengths that are more than 100 feet. The SIFT-only system, on the other hand, is close to the presented system. However, there is a clear improvement between the two. In the ACB site, the improvement is 42.53%, from 1.72m in SIFT-only to 0.98m in our system, (one-sided t-test $t(27930) = -27.3134$, $p < 0.01$), while the AnF site is 18.65%, from 3.23m to 2.63m (one-sided t-test $t(52792) = -15.5403$, $p < 0.01$), and the FDF site is 23.74% from 4.53m to 3.46m (one-sided t-test $t(69420) = -32.3395$, $p < 0.01$). On several occasions, the SIFT-only system completely misplaced the robot. In our system, whenever the salient region (SIFT and salient feature vector) matching is incorrect, the gist observation model is available to correct mistakes. In contrast, the SIFT-only system can only make a decision from one recognition module. Additionally, in kidnapped robot situations (we inserted 4 instances per run for ACB and AnF, and 5 for FDF, about once every several thousand frames), the presented system is faster to correctly relocalize because it receives twice the amount of observations (both global and local) as the SIFT only system.

The search time for the SIFT-only model is also much longer than our system. In our system, we use the gist features (segment estimation) not only as an observation model, but also as a context information for order of comparison between input and stored salient regions. That is, we compare the database salient regions from the most likely segment first. By the same token, we also use the salient feature vector as an initial comparison (if the salient feature vector between reference and test region differs significantly, there is no need for SIFT matching). In [48] we showed that the technique cuts down search time by at least 87%, a speed up of 8.

IV. DISCUSSIONS AND CONCLUSION

We introduced new ideas in vision localization which have proven to be beneficial in our testing. The first is the use of complementary gist and saliency features, implemented in parallel using shared raw feature channels (color, intensity, orientation), as study of human visual cortex suggests. Through the saliency model, the system automatically selects persistently salient regions as localization cues. Because the system does not perform whole-scene matching (only regions), the process is more efficient in the number of SIFT keypoints compared. Also, the gist features, which come with saliency at almost no computation cost, approximate the image layout and provide segment estimation. The system then performs multi-level localization by using both as MCL observations. Many scene-based methods [6]–[8] that are limited to recognizing places indicate that their results can be used as a filter for more accurate metric localization using finer yet more volatile local features. Our system is the implementation of such an extension.

Currently, segment estimation is used for both localization and match ordering; we compare input regions with database landmarks from the most likely segments first. Because robots are real-time systems, it is a given that the database search ends after the first match is found; the system does not have time to consider all positive matches to find the best. Therefore, the ordering indirectly influences the salient region recognition step. This method of utilization of multiple experts, which is in the spirit of hierarchical recognition, has been shown [49], [50] to speed up the database search process.

As for performance benchmark, to the best of our knowledge, we have not seen other systems tested in multiple outdoor environments localizing to coordinate level. At 2005 ICCV Vision contest [51], teams have to localize from a database of GPS-coordinates-tagged street-level photographs of a stretch (1 city block) of urban street. The winner [52] returns 9/22 answers within 4 meters of the actual location. Most purely vision-based systems are tested indoors and report just the recognition rate (whether the current view is correctly matched with stored images), not the location.

One issue to discuss is the system's readiness for autonomous localization and navigation. With the current setup, testing is done uni-directionally: all images are taken from the same perspective, the middle of the road. In autonomous control using lane following, a bit of swerving may occur. We may need to consider training the system on a multidirectional data set. However, recording from every perspective in the environment may put the recognition systems, both segment classification and salient region recognition, past their limits. A workable compromise would be to have the camera pan left to right (up to 45°) while the robot is on the road. We can also add, in each of the stored salient regions, where the road should be with respect to it, to aid road recognition.

Acknowledgment

The authors would like to thank Rob Peters and Lior Elazary and gratefully acknowledge NSF, HFSP, NGA, and DARPA.

References

[1] D. Fox, W. Burgard, F. Dellaert, and S. Thrun, "Monte carlo localization: Efficient position estimation for mobile robots," in *Proc. of 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] K. Lingemann, H. Surmann, A. Nuchter, and J. Hertzberg, "Indoor and outdoor localization for fast mobile robots," in *IROS*, 2004.

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

[5] 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.

[6] A. Torralba, K. P. Murphy, W. T. Freeman, and M. A. Rubin, "Context-based vision system for place and object recognition," in *ICCV*, Nice, France, October 2003, pp. 1023 – 1029.

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

[8] P. Blaer and P. Allen, "Topological mobile robot localization using fast vision techniques," in *Proc. IEEE ICRA*, 2002.

[9] 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 *Proc. of the IEEE ICRA*, 1999.

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

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

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

[13] 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.

[14] C. Valgren and A. J. Lilienthal, "Incremental spectral clustering and seasons: Appearance-based localization in outdoor environments," in *ICRA*, Pasadena, CA, 2008.

[15] A. Ramisa, A. Tapus, R. L. de Mantaras, and R. Toledo, "Mobile robot localization using panoramic vision and combination of local feature region detectors," in *ICRA*, Pasadena, CA, May 2008, pp. 538–543.

[16] 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 *IROS*, Las Vegas, NV, Oct 27 - 31 2003, pp. 2974–2979.

[17] 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.

[18] 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.

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

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

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

[22] 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.

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

[24] S. Frintrop, P. Jensfelt, and H. Christensen, "Attention landmark selection for visual slam," in *IROS*, Beijing, October 2006.

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

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

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

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

[29] 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.

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

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

[32] 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.

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

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

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

[36] 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.

[37] 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.

[38] S. Thrun, "Learning metric-topological maps for indoor mobile robot navigation," *Artificial Intelligence*, vol. 99, no. 1, pp. 21–71, 1998.

[39] B. Kuipers, "An intellectual history of the spatial semantic hierarchy," in *Robot and Cognitive Approaches to Spatial Mapping*, M. Jefferies and A. W.-K. Yeap, Eds., vol. 99, no. 1. Springer Verlag, 2008, pp. 21–71.

[40] B. Tversky, "Navigating by mind and by body," in *Spatial Cognition*, 2003, pp. 1–10.

[41] T. P. McNamara, "Memory's view of space," in *The psychology of learning and motivation: Advances in research and theory*, G. H. Bower, Ed., vol. 27, no. 1. Academic Press, 1991, pp. 147–186.

[42] J. L. Blanco, J. Gonzalez, and J. A. Fernndez-Madrigal, "Consistent observation grouping for generating metric- topological maps that improves robot localization*," in *ICRA*, Barcelona, Spain, 2006.

[43] C. Siagian and L. Itti, “Biologically-inspired robotics vision monte-carlo localization in the outdoor environment,” in *Proc. IEEE International Conference on Intelligent Robots and Systems (IROS)*, Oct 2007.

[44] L. Itti, “Models of bottom-up and top-down visual attention,” Ph.D. dissertation, Pasadena, California, Jan 2000.

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

[46] 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.

[47] C. Siagian and L. Itti, “Comparison of gist models in rapid scene categorization tasks,” in *Proc. Vision Science Society Annual Meeting (VSS08)*, May 2008.

[48] ——, “Storing and recalling information for vision localization,” in *IEEE International Conference on Robotics and Automation (ICRA), Pasadena, California*, May 2008.

[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.

Christian Siagian is currently working towards a Ph.D. degree in the field of Computer Science. His research interests include robotics and computer vision, such as vision-based mobile robot localization and scene classification, particularly the ones that are biologically-inspired.

Laurent Itti received his M.S. degree in Image Processing from the Ecole Nationale Supérieure des Télécommunications in Paris in 1994, and his Ph.D. in Computation and Neural Systems from Caltech in 2000. He is now an associate professor of Computer Science, Psychology, and Neuroscience at the University of Southern California. Dr. Itti’s research interests are in biologically-inspired computational vision, in particular in the domains of visual attention, gist, saliency, and surprise, with applications to video compression, target detection, and robotics.