Robot Localization Using Soft Object Detection

Roy Anati, Davide Scaramuzza, Konstantinos G. Derpanis, Kostas Daniilidis

***Abstract*— In this paper, we give a new double twist to the robot localization problem. We solve the problem for the case of prior maps which are semantically annotated perhaps even sketched by hand. Data association is achieved not through the detection of visual features but the detection of object classes used in the annotation of the prior maps. To avoid the caveats of general object recognition, we propose a new representation of the query images that consists of a vector of the detection scores for each object class. Given such soft object detections we are able to create hypotheses about pose and to refine them through particle filtering. As opposed to small confined office and kitchen spaces, our experiment takes place in a large open urban rail station with multiple semantically ambiguous places. The success of our approach shows that our new representation is a robust way to exploit the plethora of existing prior maps for GPS-denied environments avoiding the data association problems when matching point clouds or visual features.**

I. INTRODUCTION

In this paper, we are dealing with the problem of localization based on known maps. Research on SLAM—metric or topological—has flourished in the past decade producing maps in terms of point clouds, occupancy grids, or graphs of poses and landmarks. Such maps are used by robots to localize themselves and retraverse the mapped space. While we use the same probabilistic inference as in the classic probabilistic map-based localization [1], our representation is completely different for both the prior map and the sensorial data.

There is an abundance of prior maps for GPS-denied environments, which are semantically annotated and have been produced even with a rough sketch. Semantic annotations represent positions on the map of objects of known classes. Even if more sensor data is available about a map (like visual features) we believe that matching a query image to any place in the database is more robust to changes in viewpoint and illumination variations when we match object classes rather than visual features like SIFT and SURF or range features like corners and walls. Obviously, these advantages are offset by the increased complexity required to learn and detect objects. Furthermore, traditional object detectors are heavily biased towards images in which the object is centered and well focused.

Although these assumptions are safe in a traditional object detection task, they are not true in our localization setting. Not only do images from a moving camera suffer from bad focus and motion blur, but the desired objects are generally found off-center, away from the camera's motion path (Fig. 1).

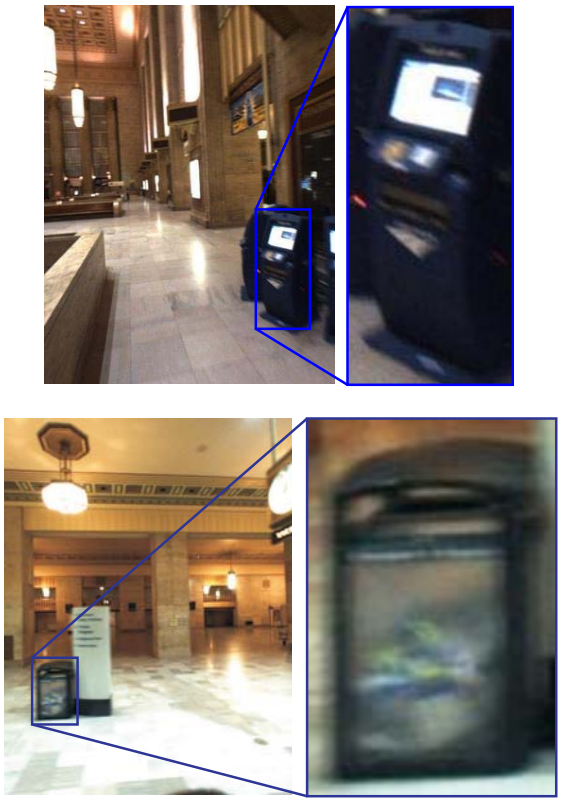


Fig. 1: Typical images from a moving platform. (top) Frontal view, object is off-center with severe perspective distortion. (bottom) Lateral view, significant motion blur.

We depart from the traditional object recognition and localization paradigm by avoiding hard detection of objects. Instead, we produce for every object a *heatmap* on the image representing the probability that the object of a particular scale appears at a particular position. Such heatmaps might be multi-modal, indicating the existence of multiple objects or the *hallucination* of multiple objects (e.g., false positives). Given several object classes, the image is then represented with a vector at each pixel containing all detection scores for that pixel. This is equivalent to a projection of the image on a basis consisting of templates representing the object classes. We have the freedom to learn these templates or even use single instances of the objects. We can use gradient and/or color representations for the appearance.

The soft detection signature (or heatmap) of a query image can be filtered using context for the expectation of objects

The authors are with the GRASP Laboratory, Department of Computer and Information Science, University of Pennsylvania, Philadelphia, PA, USA. {royanati,derpanis,kostas}@cis.upenn.edu, davide.scaramuzza@ieee.org

and can be marginalized in scale and in the vertical direction yielding a score vector for every bearing angle. Using this score vector over bearings and prior locations of objects we try to estimate 2D position and orientation using particle-based localization. We perform extended experiments in a huge indoor space (urban train station) with a hand annotated prior map. We show that short sequences of panoramic images are sufficient for localization using only few object categories.

To summarize, this paper advances the state of the art with the following contributions:

- It solves the localization problem using prior maps containing objects instead of point clouds or visual features.
- We present a new image representation that instead of containing the transition of signal to symbol[1] contains at every pixel a signature of all object-detection scores avoiding a hard-detection commitment. Precision-recall curves are not necessary on our approach because we do not apply any threshold.
- Object detection scores can be established with very simple representations, such as color or gradient distributions. We localize the robot with particle filter and show experimentally that it converges to one or more locations based on the intrinsic ambiguity of the environment.

The structure of the paper is the following. Section II reviews the related work. Section III presents our soft object-detection strategy. Section IV describes the particle filter localization based on heatmaps. Finally, sections V and VI present the experimental results and draw the conclusions, respectively.

II. RELATED WORK

Our approach on using semantic information for localization has been motivated by Kuipers' spatial semantic hierarchy paradigm [2]. The most related work to ours is Rangananathan and Dellaert's [3], who use a generative model for a place which has the form of a 3D constellation with object attributes of shape and appearance. While the model is probabilistic, the object detection produces a "hard" unimodal distribution as opposed to our "soft" detection modeling the probability of having an object at each bearing. Soft detection has been applied in [4]—called object bank response map—using a large number of pre-trained generic object detectors with the goal of scene classification. We have also been inspired by the concept of *classemes* introduced in [5] for novel category discovery while we use it for location modeling.

Most of the research in object-based localization is tailored for small indoor environments like offices with simple topology and few object categories. Many approaches detect doors or gateways ([6], [7], [8]) for place recognition as well as for detection of passages in a topological sense. Espinace et al. [9] detect the semantics of spaces (kitchens, etc.) and the objects therein starting from metric and topological maps in an indoor environment. They use both appearance features as well as 3D geometry to detect seven object and four scene categories. Galindo et al. [10] apply a conceptual hierarchy of things, objects, and rooms to label existing maps. Vasudevan et al. [11] detect objects as well as passages in order to categorize and recognize places. Similar to our notion of location is the notion of clusters in the work of Posner et al. [12] although it uses directly low level features and not objects.

Several approaches can be used to produce automatically the prior semantic maps we use in this paper. Civera et al. [13] apply appearance and geometry based recognition to annotate feature maps established with monocular SLAM. Similar annotation of features or regions on top of SLAM are undertaken in [14], [15]. Wolf and Sukhatme [16] label 3D maps based on traversability of terrain using hidden-Markov-model and support-vector-machine techniques. A different approach producing maps consisting only of semantic entities (relational object maps) has been introduced in [17] by modeling spaces with *relational Markov networks*.

III. OBJECT DETECTION

Traditional object detectors yield one of the following representations when processing a query image: boolean flag [18], bounding boxes [19], and full object (or clutter) segmentation [20]. These often include a score (or probability) measure of the detection as a whole, but always culminate in a hard decision (i.e., indicated by a bounding box) as to the presence of an object.

Instead of relying on bounding boxes or image segmentation, our system operates on object heatmaps. These are maps of the likelihood of the object being present. They provide a value for each pixel (or block of pixels) and give a dense-detection response for a given query image, as opposed to a sparse-detection response. This likelihood is normalized in a global fashion to ensure that results for different templates and different features are comparable.

We compute heatmaps for two types of local image features. These are histograms of *normalized* gradient energies (HOGE) computed over blocks of pixels (Sec. III-A) and histograms of quantized colors (HQC, Sec. III-B). Matching is performed separately with each feature. Then, the resulting heatmaps are multiplied together to yield the final object detection heatmap (Sec. III-C).

Any number of additional image features can be combined with this approach. We only require two properties from an image feature: first, that it is computed densely over the image, yielding a detection value for every pixel (or block of pixels); second, to successfully combine the output of an additional feature, it must be possible to normalize its detection results independently of the template size and the support-region size. These requirements are very flexible and, therefore, our system can incorporate a variety of additional image features to increase discriminability and improve results.

[1]By this, we mean that in standard object detection the image (i.e. the signal) is replaced by a list of labeled bounding boxes (i.e. symbols).

A. Histogram of Gradient Energies (HOGE)

The desired spatial orientation measurements are realized via filtering using a set of Gaussian derivative filters, point-wise squaring and summation over a given spatial region,

$$E_{\hat{\theta}}(u,v) = \sum_u \sum_v \Omega(u,v)[G_{N_{\hat{\theta}}}(u,v) * I(u,v)]^2, \qquad (1)$$

where $I(u,v)$ denotes the input image, $*$ convolution, $\Omega(u,v)$ a mask defining the integration region, and $G_{N_{\hat{\theta}}}(u,v)$ the Nth derivative of the Gaussian with $\hat{\theta}$ the filter's orientation.

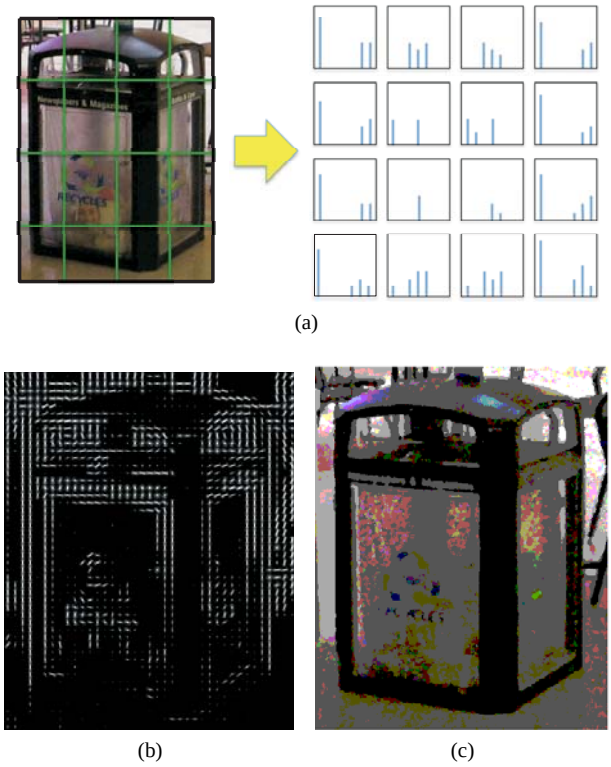
The initial definition of local energy measurements, (1), is confounded by local image contrast. This makes it indeterminate whether a high response in the filtered imagery, (1), indicates the presence of the particular spatial orientation or instead is a low match but yields a high response due to strong image contrast. To remove contrast-related information, the energy measures, (1), are normalized locally by the ensemble of oriented responses at each point,

$$\hat{E}_{\hat{\theta}_i} = \frac{E_{\hat{\theta}_i}}{\varepsilon + \sum_{\hat{\theta} \in \mathbb{S}} E_{\hat{\theta}}}, \qquad (2)$$

where $\mathbb{S}$ denotes the set of considered oriented energies, (1), and ε is a constant that serves as a noise floor (set to 1% of the expected maximum filter response). In addition, a normalized ε is computed, as in (2), to explicitly capture lack of structure within the region delineated by $\Omega(u,v)$.[2] The result is a distribution (i.e. histogram) within a given region of support, $\Omega(u,v)$, indicating the relative presence of a particular set of spatial orientations within neighborhoods of the input imagery. Finally, to define the template representation, the image is divided into non-overlapping regions, $\Omega_i(u,v)$, and a normalized energy histogram is computed for each region (see Fig. 2b).

In summary, (1)-(2) culminate in a distribution (histogram) indicating the relative presence of a particular set of spatial orientations within neighborhoods of the input imagery. Significantly, the derived measurements are invariant to additive and multiplicative bias in the image signal, due to the band-pass nature of (1) and the normalization, (2), respectively. Invariance to such biases provides a degree of robustness to various potentially distracting photometric effects (e.g., overall scene illumination, sensor sensitivity). Owing to the oriented energies being defined over a spatial support region, (1), the representation can deal with input data that are not exactly spatially aligned. Owing to the distributed nature of the representation, clutter can be accommodated: Both the desirable pattern structure and the undesirable clutter-related structure can be captured jointly so that the desirable components remain available for matching. Finally, the representation is efficiently realized via linear (separable convolution, point-wise addition) and point-wise non-linear (squaring, division) operations; thus, efficient computations are realized [21].

[2]Note that regions where structure is less apparent, e.g., region of textureless wall, the summation in the denominator approaches zero; hence, the normalized ε approaches one and thereby indicates lack of structure.



(a) (b) (c)

Fig. 2: Object Feature Computation. (a) Feature histograms over uniform pixel blocks (b) Histogram of gradient energies with 8 orientations (c) Quantized to 64 color image.

B. Histograms of Quantized Colors (HQC)

To incorporate color information, color histograms of images are computed. In this work, we use the RGB model but different ones can also be considered. An image is first quantized from RGB into an indexed color space. The target color map is created by uniformly sampling the RGB cube in all three channels. Quantizing each channel into $k << 256$ bins generates a colormap with k^3 distinct RGB values (in our case $k = 4$). Each pixel in the image is then mapped to the closest value in the target colormap. Once quantized, a histogram of the color indices is computed for each block of $n \times n$ pixels. Finally each histogram is normalized to unit energy (sum of values equal 1) by dividing each histogram by the number of pixels per block.

Although this representation is not invariant to large changes in illumination, drastically reducing the size of the color space—in this case to k^3—eliminates small changes in illumination since they are lost in the quantization (Fig. 2c).

C. Matching

The output of the matching step of our detector is not a list of bounding boxes, but rather a two-dimensional heatmap. The heatmap for each object category is computed via the Bhattacharyya similarity measure [22] of the object template features and query features. The maximum responses over all scales is selected and results from different features are multiplied together to yield the final heatmap.

Formally, for a $m \times n$ image I, the heatmap for a specific object template T maps every pixel coordinate (u,v) to a value in $[0...1]$:

$$H_T^F(I) = \max_{Scales} \frac{Corr\left(\sqrt{F(I^s)}, \sqrt{F(T)}\right)}{b_T}, \quad (3)$$

where F is the image feature function (either HOGE, or HQC), and I^s represents the image at scale s. *Corr* is the standard correlation function and b_T is the number of blocks in the template. Dividing the result by the template size scales the values to the range $[0...1]$. We then combine the heatmaps for each type of image feature using point-wise product:

$$H_T = H_T^{HOGE}(I) \cdot H_T^{HQC}(I). \quad (4)$$

Example heatmaps for HOGE, HQC, and their product are shown in Fig. 3, b-d respectively.

IV. OBJECT-BASED LOCALIZATION

The goal of localization is to retrieve the robot absolute position in the environment using all the available information from its on-board sensors, such as wheel encoders and cameras. From computer vision, it is known that the absolute position of a single calibrated camera can be inferred from a minimum of three 3D-2D correspondences, that is, the 3D absolute positions of three scene points and the 2D coordinates of their projections in the camera image [23], [24]. This method is known as the "perspective three-point algorithm" (P3P). There are three drawbacks in using P3P for object-based localization. First, we are not using points but objects, whose position in the image is not as well localized as with points. The second one is that we are using a soft detector and therefore we only get a "likelihood" (i.e. heatmap) that the object is at a given image coordinate. Third, P3P requires that three objects be viewed by the robot simultaneously, a situation that is unlikely to happen in real environments. In our challenging dataset, no more than two objects are visible at the same, with most of the images containing a single object. For these reasons, we opted for the *particle filter localization* strategy [1], [25].

In probabilistic map-based localization, we want to estimate the state of the robot at the current time step t, given the knowledge about its initial state and all the measurements Z^t up to the current state. In our setup, the robot moves in a planar environment, therefore, our state vector is $\mathbf{x} = [x,y,\theta]^T$, with (x,y) denoting the robot position and θ its orientation.

The particle filter represents the probability distribution $p(\mathbf{x}|Z^t)$ of the robot pose by a set of N particles $S_t = \{\mathbf{s}_t^i, i = 1..N\}$ drawn from it. This is done in two phases.

1) Prediction Update: In this phase, we compute the set of particles S_t from the previous set S_{t-1} by sampling from the motion model. We use the motion model of a differential-drive robot:

$$\begin{cases} x_t^i &= x_{t-1}^i + (v_t + \Delta V)\cos\left(\theta_{t-1}^i + \Delta\theta/2\right) \\ y_t^i &= y_{t-1}^i + (v_t + \Delta V)\sin\left(\theta_{t-1}^i + \Delta\theta/2\right) \\ \theta_t^i &= \theta_{t-1}^i + \Delta\theta \end{cases} \quad (5)$$

where $\Delta\theta = (\omega_{t-1}^i + \Delta\Omega)\Delta t$, v_t and ω_t being the translational and angular control speeds, and ΔV and $\Delta\Omega$ normally-distributed random variables that account for the noise in the motion.

2) Perception Update: In this phase, we incorporate the information $\mathbf{z}_t = \{H_{T_1}(I_t), \ldots, H_{T_n}(I_t)\}$ (the collection of heatmaps for all objects $\{T_1, \ldots, T_n\}$ for the current image I_t) from the camera and weight each sample in S_t by the weight $w_t^i = p(\mathbf{s}_t^i|\mathbf{z}_t)$, that represents the likelihood of $\mathbf{s}_t^i$ given $\mathbf{z}_t$. Finally, we compute the new set S_t' from the weighted set using *importance resampling* [25].

The delicate part is the computation of the weight w_t^i. This is a function of the observed heatmap $\mathbf{z}_t^i$, the object map $M = \{\mathbf{m}_j, j = 1..n\}$, and the particle pose $\mathbf{s}_t^i$. In particular, we want w_t^i to tell us how well the *observed* heatmap $\mathbf{z}_t^i$ matches the *expected* heatmap $\hat{\mathbf{z}}_t^i$, that is, the heatmap that the particle $\mathbf{s}_t^i$ is expected to observe from its position. A good measure of the similarity of two functions is their inner product; therefore, we use the following expression for w_t^i (from now on, we omit the subscript t to simplify the notation):

$$w^i = \sum_{j=1}^{n} < \mathbf{z}_j^i, \hat{\mathbf{z}}_j^i > \quad (6)$$

where the subscript j denotes a specific object category. As observed, the weight of each particle is a sum of the inner products between the observed and the expected heatmap of each object.

Remember that the heatmap of an object is a two dimensional function of the image coordinates. However, since we consider planar motion it is reasonable to convert the heatmap into a one-dimensional signal that depends only on the azimuthal angle δ. Since our camera is calibrated and approximately perpendicular to the soil, we do this 2D-to-1D conversion by simply taking the maximum along each column of the heatmap image. Like the 2D heatmap, also the 1D heatmap assumes values in the range $[0...1]$. An example 1D heatmap is shown in Fig. 3e for the case of the clock template in Fig. 4.

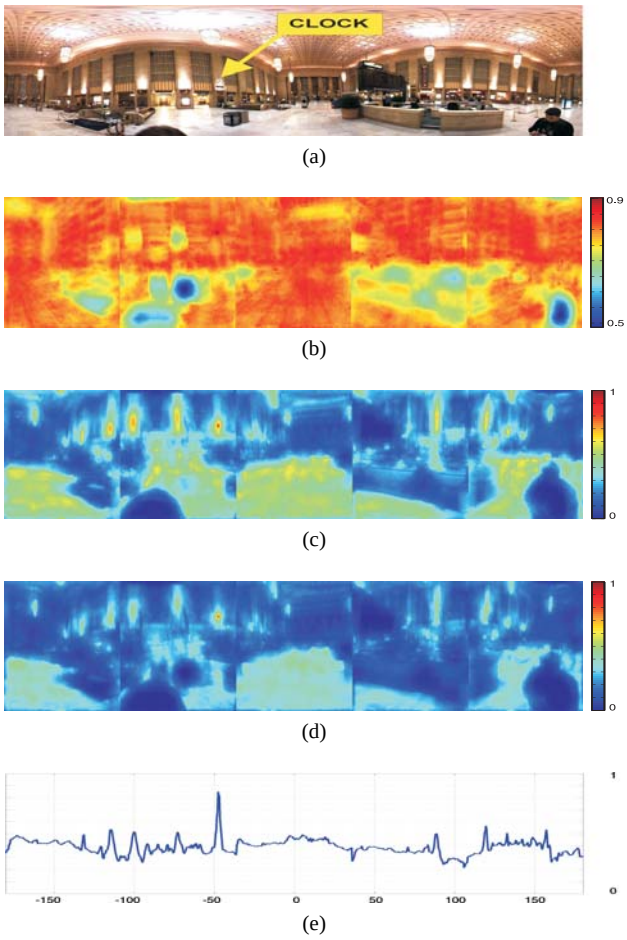
The expected heatmap $\hat{\mathbf{z}}_j^i$ for a given particle $\mathbf{s}^i$ and for a specific object category j is computed as

$$\hat{\mathbf{z}}_j^i(\delta) = \max_{\alpha} \Phi(\alpha) \exp\left(-\frac{\left(\delta - \hat{\delta}_{j_\alpha}^i\right)^2}{2\sigma^2}\right) \quad (7)$$

where α denotes a particular instance of the same object category j and $\hat{\delta}_{j_\alpha}^i$ is the view angle of the object in the particle's reference frame. $\Phi(\alpha)$ is 1 if the instance is within the visibility range (20m) and 0 otherwise. We used the same σ for every object, regardless of its distance to the particle. Also, we use the $\max(\cdot)$ instead of the $\sum(\cdot)$ operator so that if two instances are occluding each other their heatmaps do not sum up.

V. EXPERIMENTS

We captured the dataset for our experiments in the major urban train station of Philadelphia. The dataset consists of 20



(a) (b) (c) (d) (e)

Fig. 3: (a) original 360° panoramic image. (b) HOGE heatmap. (c) HQC heatmap. (d) The final heatmap is computed as point-wise product between the HOGE and the HQC heatmaps. (e) One-dimensional heatmap. These heatmaps were computed for the clock. Notice the well distinguishable peaks in the heatmaps in correspondence of the clock.

thousand omnidirectional images (360-degree field of view) captured using a PointGrey Ladybug 3 camera. The unit consists of six cameras mounted in a hemi-sphere with five cameras in a circle and the sixth camera pointing upwards. In this work we consider only images from the five sideways cameras. Therefore, our total image set consists of 100 thousand images. The camera was mounted on a differential-drive robot and driven around the environment.

The train station is a large indoor environment containing both large open spaces and small spaces, such as hallways, shops, restaurants, and booths. Being primarily a pedestrian environment, our motion was unconstrained. We were able to traverse portions of the station multiple times, approaching previously visited locations from different and opposite directions. This is in contrast to data captured with an outdoor vehicular setup which is often restricted to retracing its path, visiting previous locations with identical trajectories.

Since our focus is localization in large open indoor spaces, we restricted ourselves to the station's main hall. This room is 88 by 41 meters and 29m high. About 40 percent of the image data, around 8,000 views, were captured within this area. We sample these at a rate of 1-in-10 resulting in a final video sequence of 791 views (resulting in an image approximately every 2m).

A. Object Detection

For the purposes of localization, the map of the environment needs to be populated with the locations of the objects. As such, we selected objects present in the environment that are either permanent fixtures (clocks, payphones) or that are unlikely to move significant distances (trashcans). The final set of objects we employed are: trashcan, clock, payphone, ticket machine.

Each object template is constructed by a single object exemplar. We compute both histograms of oriented energies, and histograms of quantized colors and use the resulting product to perform detection.

Object detection in this setting is especially challenging due to the method the images were captured. The fact that the camera was moving precluded high-quality capture of images from the lateral cameras due to motion blur (Fig. 1). Additionally, this motion also prevented objects from being clearly centered in the camera images (Fig. 1), a condition that is commonly assumed in traditional object recognition to facilitate detection.

We manually selected a *single example* for each object category. Employing a single positive example eliminates the need for extensive labeling and training that is common with most approaches [26], [19]. This results in a simpler detector with almost no offline pre-preprocessing. The primary disadvantage is complete lack of generalization, or ability to handle intra-class variation.[3] Although the ability of an object detector to handle intra-class variation is critical for the general task of object detection, it is not a necessity for the localization problem.

Of the categories chosen, the clock has the most peaky heatmaps. It's distinctive color, white, and very clear boundary combine to yield isolated hot spots (Fig. 4 top row). The ticket machines (Fig. 4 second row) are generally installed in groups; this confused the HQC feature creating "warmer" regions around the objects. Note however that clearly distinct red "hot" spots appear centered on the machines. The detections for the trashcan are somewhat weaker (Fig. 4 bottom row) with lukewarm peaks in the heatmaps. These are caused by occlusions, transparencies, and perspective. In some cases, the trashcan is partially obscured, lowering the match scores. Furthermore, we do not take into account the transparency of the plastic bag that covers most of the trashcan template. As such, both objects behind the trashcan (Fig. 2c) and its content adversely affect the matching, especially with respect to color histograms. Furthermore, a distinct disadvantage of our single-exemplar object model is its inability to recognize the trashcan when viewed from a

[3]Intra-class variation is the variation in appearance between two object instances belonging to the same object category, e.g. two ATMs of different banks.

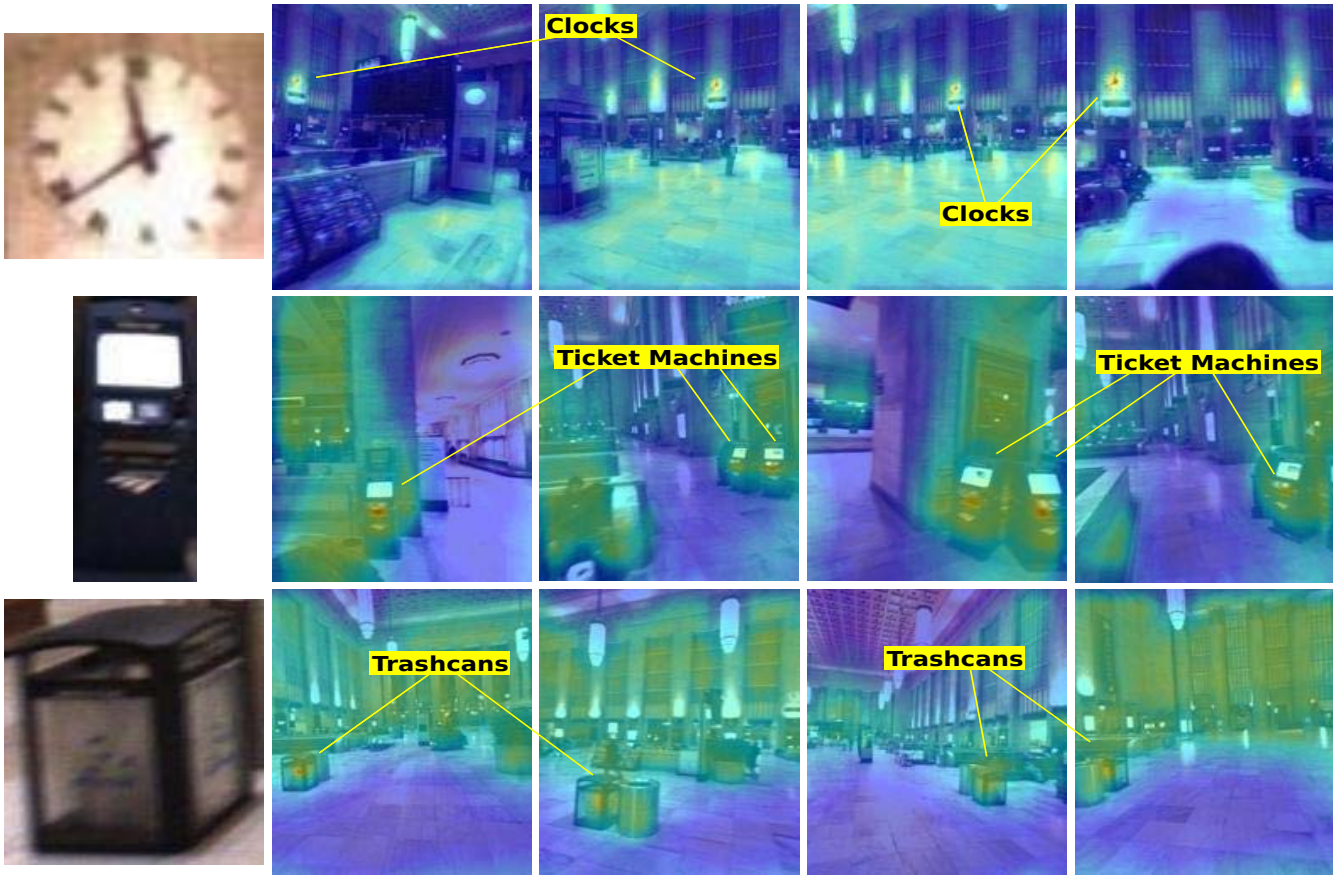


Fig. 4: Detection results, with the object template in the first column and resulting heatmaps in the remaining columns. Hot spots, orange to red, indicate increased object presence (Best viewed in color).

different perspective. This shortcoming could be addressed by adding multiple templates for multiple views of each object. This would indeed increase performance but also the time required to detect objects, and require finding and extracting multiple exemplars for each object to cover multitude angles. However, a full coverage of each object is not necessary. Our histogram based approach is still able to detect the presence of the trashcan even when flipped with a single exemplar. By taking advantage of partial symmetry in the objects, we can compute heatmaps using flipped templates and select the maximal response from the original and mirrored templates.

Other considerations that lead to reduced detection performance include extreme lighting, low resolution imagery, and scene clutter. Although the desired object is clearly visible (Fig. 5a), a combination of low resolution and lighting change yield a low match score resulting in a cold spot. On the other hand, the alignment of a person with dark clothing with a white advertisement creates the hallucination of a ticket machine (Fig. 5b).

(a) (b)

Fig. 5: Some detection failures: (a) Missed object, resulting in a cold spot. (b) Hot area resulting from object hallucinations.

B. Localization

Getting accurate ground-truth data in an indoor setting is a challenging problem in itself. Key frames in the video sequence were annotated with their ground truth position information during collection on a simplified building schematic. These were then used to interpolate position information for all remaining images.

In order to use objects for localization, we manually annotated a map of the train station with the locations of the objects. Objects were treated as points, covering no area.

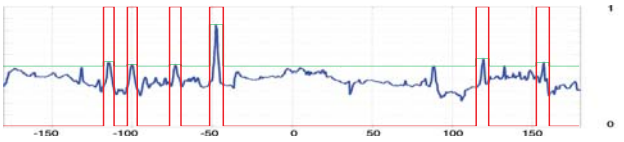


Fig. 8: Creating a Hard Detector: Original heatmap (blue), with fixed thresholding and non-maxima supression (green) resulting in a binary signal (red).

This did not greatly affect localization results, as the objects in question all have small footprints.

The map of the station with the object is showcased in Fig. 6a. The red line indicates the hand-labeled ground-truth camera trajectory, with the green star denoting the current ground-truth position. We use 10,000 particles, denoted with red dots. Note how the uniform particle cloud coalesces into clusters (Fig. 6b). Multiple instances of each object category creates ambiguity generating multiple location hypotheses (Fig. 6c). Eventually, enough objects are observed that the system focuses around the true location (Fig. 6d).

To demonstrate the advantages of soft object detectors we also perform localization using a "hardened" version of our detector. In order to generate traditional bounding boxes from our soft detector, we threshold the projected heatmaps (Fig. 8 blue) with a fixed threshold (chosen appropriately for each object category). Then non-maxima supression is applied to find local maxima in the response map; these determine the positive detections (Fig. 8 green). Each local maxima represents a positively detected bounding box, with a fixed width. The final signal fed into the localizer is a binary one-dimensional heatmap with value 1 in the direction where there exists a positive detection, and 0 elsewhere (Fig. 8 red).

The use of a more traditional hard detector has a strong clustering effect on the particles. Starting even at one iteration (Fig. 7a) the particles are noticeably less spread out. Although this provides increased confidence in the computed position with smaller, tighter clusters, it is more susceptible to incorrect detections, in the end failing to correctly localize the camera (Fig. 7d).

The ability of soft detections to include weak signals prevents from over-committing to an incorrect localization. This comes at the cost of larger uncertainty in the positioning leading to larger particle clusters. One could argue that a similar effect could be achieved by inserting random particles at each iteration of the localization process, adding random noise. Our system takes a more systematic approach, incorporating the uncertainty at the source of the detection (where it originates) rather that in the processing.

VI. CONCLUSIONS AND FUTURE WORK

We showed it is possible to localize in a challenging indoor environment using only objects from images. Our approach is composed of a simple object detector and a particle filter approach to localization. Instead of trying to tackle the object detection problem as a separate component, we instead tailored our approach to the task of localization. The resulting object detection scheme relies on soft detections using object response maps, or heatmaps. By employing a simple soft detection we were able to perform accurate localization without the need for learning or training.

Currently, the feature parameters—support region size (for both HOGE and HQC), the ε (for HOGE), and the color quantization bins k (for HQC)—are manually determined. A more careful study of these would both increase the quality of our soft detections—thus, improving the accuracy—and rate of convergence during localization.

In this work, we focused on two properties of images to generate heatmaps: gradient energies and colors. Although a richer library of image features would increase the detector's discriminability, it would increase the computational burden. Additionally, the temporal aspect of sequential input images is not considered. Devising a method to "track" object heatmaps through time (as opposed to tracking objects) would also improve heatmap quality.

The need for a pre-processed map already containing localized objects is a significant burden on users of this system. Incorporating a traditional mapping solution is insufficient as it lacks automatic localization of semantically meaningful objects. We propose that an enhanced mapping solution can be developed by incorporating soft detections. Soft detection, unlike traditional object detection frameworks, is built specifically to tackle the challenges of localization and associated imagery.

VII. ACKNOWLEDGMENTS

The authors would like to thank Andrea Censi for his fruitful comments and suggestions. Support through the grants NSF-IIP-0742304, NSF-IIP- 0835714, NSF-OIA-1028009, ARL MAST-CTA W911NF-08-2-0004, and ARL RCTA W911NF-10-2-0016, NSF-DGE-0966142 is gratefully appreciated.

REFERENCES

[1] F. Dellaert, D. Fox, W. Burgard, and S. Thrun, "Monte carlo localization for mobile robots," in *IEEE International Conference on Robotics and Automation*, 1999.

[2] B. J. Kuipers, "The spatial semantic hierarchy," *Artificial Intelligence*, 2000.

[3] A. Ranganathan and F. Dellaert, "Semantic modeling of places using objects," in *RSS*. Citeseer, 2007.

[4] L. Li, H. Su, E. Xing, and L. Fei-Fei, "Object bank: A high-level image representation for scene classification & semantic feature sparsification," in *Neural Information Processing Systems (NIPS)*, 2010.

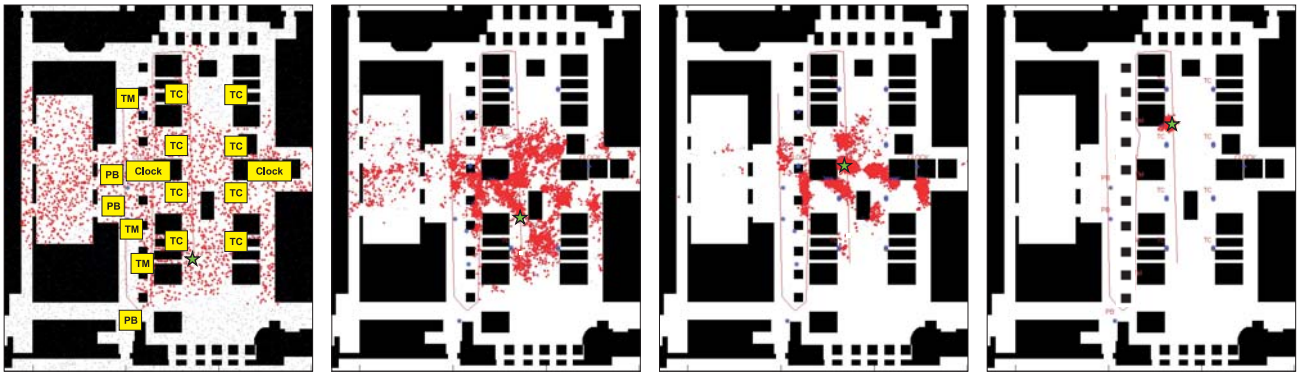
[5] L. Torresani, M. Szummer, and A. Fitzgibbon, "Efficient object category recognition using classemes," *Computer Vision–ECCV 2010*, pp. 776–789, 2010.

[6] A. C. Murillo, J. Kosecka, J. J. Guerrero, and C. Sagues, "Visual door detection integrating appearance and shape cues," *Robotics and Autonomous Systems*, vol. 56, no. 6, pp. 512–521, 2008.

[7] D. Schroter, M. Beetz, and J. Gutmann, "Rg mapping: Learning compact and structured 2d line maps of indoor environments," in *Robot and Human Interactive Communication*. IEEE, 2002, pp. 282–287.

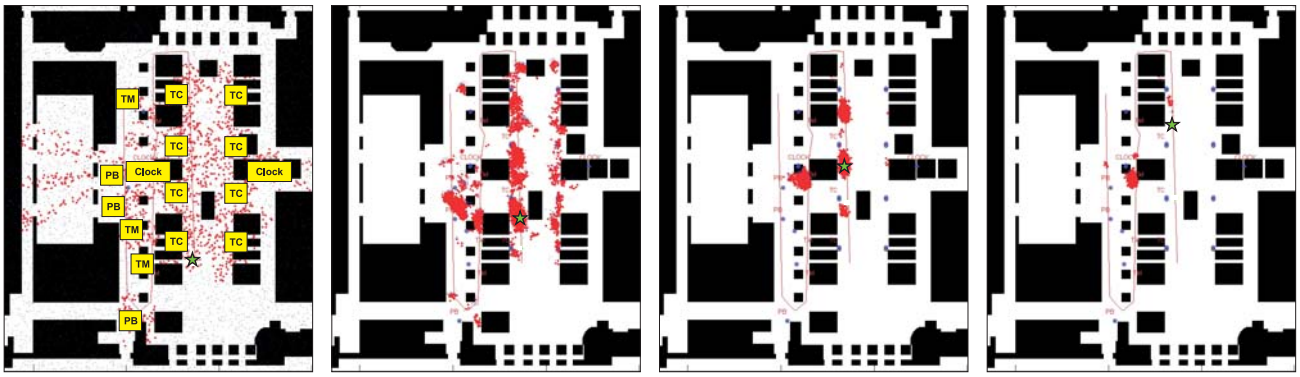
[8] S. Stoeter, F. Le Mauff, and N. Papanikolopoulos, "Real-time door detection in cluttered environments," in *Intelligent Control, 2000. Proceedings of the 2000 IEEE International Symposium on*. IEEE, 2000, pp. 187–192.

[9] P. Espinace, T. Kollar, A. Soto, and N. Roy, "Indoor scene recognition through object detection," in *ICRA*. IEEE, 2010, pp. 1406–1413.



(a) (b) (c) (d)

Fig. 6: Global localization results using the proposed soft object detector. Particle locations at (a) one, (b) ten, (c) twenty, and (d) forty iterations. (a) The objects used for localization are labeled in the positions where they appear in the map: clock, trashcan (TC), phone booth (PB), ticket machine (TM).



(a) (b) (c) (d)

Fig. 7: Global localization results using a hard object detector. Particle locations at (a) one, (b) ten, (c) twenty, and (d) forty iterations.

[10] C. Galindo, A. Saffiotti, S. Coradeschi, P. Buschka, J. A. Fernández-Madrigal, and J. González, "Multi-hierarchical semantic maps for mobile robotics," in *IROS*. IEEE, 2005, pp. 2278–2283.

[11] S. Vasudevan, S. Gachter, V. Nguyen, and R. Siegwart, "Cognitive maps for mobile robots–an object based approach," *Robotics and Autonomous Systems*, vol. 55, no. 5, pp. 359–371, 2007.

[12] I. Posner, D. Schroeter, and P. M. Newman, *Using Scene Similarity for Place Labelling*. Berlin / Heidelberg: Springer, 2008, vol. 39, pp. 85–98.

[13] J. Civera, D. Glvez-Lpez, L. Riazuelo, J. Tards, and J. M. M. Montiel, "Towards semantic slam using a monocular camera," in *IROS*, 2011.

[14] J. Oberlander, K. Uhl, J. Zollner, and R. Dillmann, "A region-based slam algorithm capturing metric, topological, and semantic properties," in *Robotics and Automation, 2008. ICRA 2008. IEEE International Conference on*. IEEE, 2008, pp. 1886–1891.

[15] A. Trevor, C. Nieto-Granda, J. Rogers, and H. Christensen, "Feature-based mapping with grounded landmark and place labels," in *Intelligent Robots and Systems (IROS), 2010 IEEE/RSJ International Conference on*. IEEE, 2010, pp. 1451–1456.

[16] D. Wolf and G. Sukhatme, "Semantic mapping using mobile robots," *Robotics, IEEE Transactions on*, vol. 24, no. 2, pp. 245–258, 2008.

[17] B. Limketkai, L. Liao, and D. Fox, "Relational object maps for mobile robots," in *International Joint Conference on Artificial Intelligence*, vol. 19. Citeseer, 2005, p. 1471.

[18] S. Lazebnik, C. Schmid, and J. Ponce., "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *CVPR*, vol. 2, 2006, pp. 2169–2178.

[19] P. F. Felzenszwalb, R. B. Girshick, D. McAllester, and D. Ramanan, "Object detection with discriminatively trained part-based models," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 32, no. 9, pp. 1627–1645, 2010.

[20] A. Toshev, B. Taskar, and K. Daniilidis, "Object detection via boundary structure segmentation," in *CVPR*. IEEE Computer Society, 2010, pp. 950–957.

[21] W. Freeman and E. Adelson, "The design and use of steerable filters," *PAMI*, vol. 13, no. 9, pp. 891–906, September 1991.

[22] A. Bhattacharyya, "On a measure of divergence between two statistical populations defined by their probability distribution," *Bull. Cal. Math. Soc.*, vol. 35, pp. 99–110, 1943.

[23] X. Gao, X. Hou, J. Tang, and H. Cheng, "Complete solution classification for the perspective-three-point problem," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 25, no. 8, pp. 930–943, 2003.

[24] L. Kneip, D. Scaramuzza, and R. Siegwart, "A novel parameterization of the perspective-three-point problem for a direct computation of absolute camera position and orientation," in *IEEE Conference on Computer Vision and Pattern Recognition*, 2011.

[25] S. Thrun, W. Burgard, and D. Fox, *Probabilistic Robotics (Intelligent Robotics and Autonomous Agents)*. MIT Press, 2001.

[26] C. M. Bishop, *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer, 2006.