Path Localization Using Gabor-Gist

Michael Mills$^1$ and Hong Zhang$^2$

Abstract—Learning and then recognizing a path is a challenging task for state of the art algorithms in computer vision and robotics. In this paper, we present a new approach to visual localization along a path. Classically visual paths have been described using keyframes, single images taken at specific locations. Our method uses all the images of a path segment, Gabor-Gist and, principal component analysis to represent a segment as segment specific principal components. Localization is achieved by comparing a query image descriptor to the segment’s principal components using a new reconstruction similarity measure, choosing the path segment which best reconstructs the original query descriptor. Using two datasets of indoor and outdoor environments we compare our method to the same path represented using keyframes. While the feature-based keyframes perform poorly, the new method is able to correctly localize the robot 93% of the time.

I. INTRODUCTION

Recently robot localization and navigation have made considerable progress. Navigation can be seen as the ability for a robot to move autonomously from an initial position to a desired one. Thanks to sensor based navigation, autonomous robots have appeared in various challenging areas such as highway, deserts and even on Mars. In this paper, we propose a new method to localize a robot along a path using principal components and a new similarity measure, without tracking, matching nor processing of keypoint features.

Many approaches consider a (partial) 3D reconstruction of the environment, leading to SLAM-like techniques, thus the navigation task is achieved using a classical pose-based control of the robot in the metric space. Despite the complexity of the underlying problem, SLAM has proved to be a viable solution to create accurate maps of the environment [3], [13]. In this context, the problems are 1) the complexity of the 3D reconstruction step, 2) matching of visual features observed during learning to current observations.

Appearance-based approaches [7], [2], [11], [1], [4] seek to define and then measure in a qualitative sense the pathness of an image. This leads to the definition of a visual path no longer described in metric space but, as a set of reference images or keyframes. Localization in these terms is about deciding which principal components using a new reconstruction similarity measure, choosing the path segment which best reconstructs the original query descriptor. Using two datasets of indoor and outdoor environments we compare our method to the same path represented using keyframes. While the feature-based keyframes perform poorly, the new method is able to correctly localize the robot 93% of the time.

A. Gabor-Gist

Gist refers to the meaningful information that an observer can identify from just a glimpse of a scene [12]. The idea of Gist is to define what a “scene” is, as opposed to an “object” or “texture” within a scene. When viewing a scene for a short time, humans extract enough visual information to accurately recognize its categorical properties (e.g., trees on a mountain side). Most of the information concerning individual objects and their locations is overlooked, rather viewing a scene as having its own shape, carrying its identity [12]. Categories like cars or animals, look alike because they have the same “function”. Oliva [12] showed that scenes belonging to the same category share a similar and stable spatial structure (shape) that can be extracted and used to classify a scene into categories. They showed that perceptual properties exist that can be uncovered using simple computations, and that these properties can be translated into a meaningful description of the scene shape.

The Gist description of a scene is useful beyond scene classification. Torralba and Oliva expand their methodology to the estimation of depth from image structure. They demonstrate that, by recognizing the properties of the structures present in the image, they can infer the scale of the scene and, therefore, its absolute mean depth [16], expanding further to place and object recognition [17]. In both [16], [17] they use a wavelet image decomposition, where each image location is represented by the output of filters tuned to different orientations and scales.

The representation of an image is given by a collection of micro-feature statistics. This collection of micro-features is the responses to a set of Gabor filters $h_k(x)$ convolved with an image $I^1$.

$h_k(x)$

1Michael Mills is with the Department of Computing Science at the University of Alberta, Canada mmills1 at ualberta.ca

2Hong Zhang is with the Department of Computing Science at the University of Alberta, Canada hzh@ualberta.ca

1Here we only consider gray scale images but the same technique can be applied to RGB images.
A vector image is extracted from the cells of a grid placed over each response.

V. II. BUILDING THE VISUAL PATH

In training the robot’s current view \( v_1^G \) is compared with the keyframe \( v_1^G \) of the lastest created segment. When the similarity between the two descriptors \( (v_1^G \) and \( v_1^G \)) falls below an empirically determined threshold, a new segment is created and the process repeats. The result is a sequence of non-overlapping segments.

Segment images are similar but vary in unique ways as compared to other segments. These variations can be used to recognize a segment. One approach to capturing this variation in the images is to determine principal components of segment’s image descriptor distribution. These eigenvectors can be thought of as features which together characterize a segment we have dubbed our representation an eigensegment.

To create this representation image descriptors between keyframes \( v_1^G , ..., v_{M-1}^G \) are subjected to principal component analysis (PCA). PCA seeks a set of \( N \) orthogonal vectors and eigenvalues which best describe the distribution of descriptors, where \( N \) is the dimension of the descriptors. The eigenvectors are then sorted by their eigenvalues, the top \( k \) of which form our matrix \( U_i \) representing the \( i \)th eigensegment.

III. LOCALIZATION USING EIGENSEGMENTS

PCA has traditionally been used to reduce the number of dimensions allowing for a faster comparison between descriptors. Our method uses PCA to measure an eigensegment’s ability to reproduce or reconstruct the descriptor of the current
Fig. 3: Selecting the next segment is done by projecting the current view into a range of eigensegments and then reprojected back. Which ever eigensegment does the accurately reconstructs the descriptor is selected as the current segment.

view. The current view descriptor is projected into a lower dimensional space and then reprojected back to the original space. This reprojection or reconstruction is then compared with the original descriptor, the eigensegment which best reconstructs the original descriptor is selected as the robots current position (see Figure 3).

The following steps summarize the localization process:

1) Project the current view’s descriptor $v_t^G$ into an eigensegments lower dimensional spaces $v_t^U$.
2) Reproject the descriptor $v_t^U$ back to the high dimensional space $v_t^{G'}$.
3) Compare the original and reconstructed vectors using cosine distance.
4) Set the location to the eigensegment which best reconstructs the current view’s descriptor.

Eigensegments are stored in a sequence matrices $\{U_0, ..., U_{M-1}\}$. At each time step the current view of the robot $v_t^G$ is compared with the sequence of eigensegments. The first step in this comparison is to project $v_t^G$ current view’s descriptor into the subspace of the eigensegments.

$$v_t^U = U_i(v_t^G - \mu_i)$$

where $\mu_i$ the average Gist descriptor of the segment $i$. The result of this is a reduced Gist vector $v_t^{G'}$ of dimension $k$, where $k \leq N$ and $N$ is the Gist descriptor’s original dimension.

We then reconstruct the original descriptor $v_t^{G'}$ by:

$$v_t^{G'} = (U^T \Phi_k) + \mu_i$$

The resulting vector $v_t^{G'}$ is of the same dimension as the original i.e. $N$, but is not an exact reproduction of the original
Fig. 4: Sample images from outdoor’s (top) and indoor’s (bottom) environments.

descriptor because less than $N$ eigenvectors are used. The reconstruction error $e$ between $v_{Gt}^G$ and $v_{Gt}^G$ is determined using the cosine distance $e = v_{Gt}^G \cdot v_{Gt}^G$. The eigensegment which minimizes the reconstruction error $e$ is then selected as the current segment, and thus localizes the robot along the path. Figure 3 shows the entire process of selecting the next segment.

IV. EXPERIMENTS

Two datasets were obtained, each containing two loops along the same path. The first dataset is an indoor environment, containing many repeating patterns and two regions in particular exhibiting high visual aliasing. The second an outdoor environment, containing both urban and natural visual elements such as building and trees. The outdoor environment also exhibits lighting variations due to changing cloud cover and time of day. Figure 4 shows some example images.

Initially the eigensegment method was verified in a simple and intuitive way. The robot’s belief about where it is along a path can be visualized as a 2D graph with position on the x-axis and likelihood on the y. The experiment shown in Figure 5 illustrates the belief the robot has about where it is along a path given an image. The eigensegments are trained using the first loop of the datasets. From the second loop an image near the middle of the loop are selected and compared with all the eigensegments. In PCA the number of principal components determines how much of the original descriptor’s information is maintained. As such the eigensegments are trained with 2, 10, 50, and 200 principal components. Figure 5 shows the spike in similarity near the middle segments and how the number of principal components affects the result $^2$. Interestingly the number of principal components does not appear to significantly affect the method, two PCs performing similar to 50 PCs. The results show that variations which best identify a segment are encoded in the first principal components.

To test the performance of localization 1000 random test images from the second loop of the datasets are selected and compared with eigensegments. Figure 6 shows the results as compared with the previous methods of comparing keyframes using Gabor-Gist and SURF features. The paths where trained so as to have roughly the same number of segments for each method. Gabor-Gist keyframes are compared in one of three ways (1) reprojection the proposed method, (2) segment PCA where the descriptors are projected and compared in local subspaces specific to a segment and, (3) global PCA where descriptors are projected into a subspace defined by PCs extracted using all images of a path.

(a) Indoors path consisting of 32 segments. The correct segment is 14.

(b) Outdoors path consisting of 49 segments. The correct segment is 33.

These results in Figure 6 show improvement over the method of keyframe comparisons. In the outdoor environment the proposed method even outperformed SURF features at this task. The performance of SURF is similar to the results reported in [18]. The results shown for SURF use a matching method known as 1 to 1, two features are matched only if they are each others closest match in cosine distance. The image with the most matching pairs to the current view is selected as the location. Indoors the results are certainly an improvement over direct comparisons of descriptors in lower dimensions but do not outperform SURF. This improvement is due in part to using all the images from a segment to build its representation rather than a single frame. It is also
due to the proposed similarity measure of comparing the reconstruction of a descriptor rather than a direct comparison of two descriptors.

The experiments show SURF features perform poorly in the outdoor sequence which is a counter intuitive result. Generally SURF features are considered one of the gold standard methods for image matching. The poor performance here is due to the effects of visual aliasing, specifically the sidewalk, both visually similar throughout the dataset and occupying a large proportion of the image. SURF features are easily mismatched when dealing with visually similar elements such as sidewalks as can be seen in Figure 7.

The differences in performance measured between indoors and outdoors are believed to be due in part to factors such as texture characteristics, visual aliasing and, the scale of the environment. Outdoor environments typically contain a wider more diverse set of textures in unique spatial configurations. This naturally leads to a more distinct look for a scene when viewed in the context of a whole image descriptor. This is in opposition to indoor environments where objects are often repeated and commonly configured in similar ways at many locations. For example hallways often have a similar look independent of where you are in a building, leading to visual aliasing. Lastly the scale of the environment may play a role. Indoors the walls are typically close and even a relatively small movement can produce a large visual shift due to parallax. This large visual change maybe producing a variation in the data that affects the PCs, making it harder to localize a robot indoors using our method, compared to local features which may not be affected by this because they consider only a single image.

The method was then applied to a real-time robot performing visual homing in both indoor and outdoor environment. The system used the parameters in Table I for both environments.
In indoor environments (see Figure 8) the robot correctly localized itself 80% in 5 trials. In outdoor (see Figure 9) environments the success rate increased to 100%.

V. CONCLUSIONS

In this paper we presented a new way to characterize and compare segments of a visual path. This new approach uses data classically discarded when using keyframes. Our approach uses the images between keyframes and PCA to build a compact representation of a segment we call an eigensegment. We further introduced a new method of comparison, one which uses a reconstruction of a descriptor rather than a direct comparison. These contributions tested using both simulations and real world trials, achieving success rates as high as 100%.

ACKNOWLEDGMENT

This work was supported in part by the Natural Sciences and Engineering Research Council (NSERC) through the NSERC Canadian Field Robotics Network (NCFRN).

REFERENCES


TABLE I: The parameters of the system. Image size refers to the size of image used to create the Gabor-gist descriptor. Gabor-gist grid refers to the spatial resolution the descriptor maintains. Eigensegment PCs control how much information is retained to characterize a path segment. Localization window controls which segments around the current segment $i$ are checked during segment selection.

<table>
<thead>
<tr>
<th>Parameters</th>
<th>Value</th>
</tr>
</thead>
<tbody>
<tr>
<td>Image Size</td>
<td>600x200 pixels</td>
</tr>
<tr>
<td>Gabor-gist grid</td>
<td>6x4 cells</td>
</tr>
<tr>
<td>Eigensegment PCs</td>
<td>2 PCs</td>
</tr>
<tr>
<td>Segment threshold</td>
<td>0.1 rad</td>
</tr>
<tr>
<td>Localization window</td>
<td>$[i - 1, i + 4]$</td>
</tr>
</tbody>
</table>

3Indoors test video http://youtu.be/eKvNEXcVzI
4Outdoors test video http://youtu.be/kof6o6_AmV8