Filtering 3D Keypoints Using GIST For Accurate Image-Based Localization

Charbel Azzi[1]
cazzi@uwaterloo.ca
Daniel Asmar[2]
da20@aub.edu.lb
Adel Fakih[1]
afakih@uwaterloo.ca
John Zelek[1]
jzelek@uwaterloo

[1] Systems Design Engineering
University of Waterloo
Waterloo, Canada

[2] Vision and Robotics Lab
Mechanical Engineering Department
American University of Beirut
Beirut, Lebanon

Abstract

Image-Based Localization (IBL) is the problem of estimating the 3D pose of a camera with respect to a 3D representation of the scene. IBL is quite challenging in large-scale environments spanning a wide variety of viewpoints, illumination, and areas where matching a query image against hundreds of thousands of 3D points becomes prone to a large number of outliers and ambiguous situations. The current state of the art IBL solutions attempted to address the problem using paradigms such as bag-of-words, features co-occurrence, deep learning and others, with varying degrees of success. This paper presents GIST-based Search Space Reduction (GSSR) for indoor and large scale Image-Based Localization applications such as relocalization, loop closure and location recognition. GSSR explores the use of global descriptors, in particular GIST, to introduce a new similarity measure for keyframes that combines the GIST descriptor scores of all neighboring frames to qualify a limited number of 3D points for the matching process, hence reducing the problem to its small size counterpart. Our results on standard datasets show that our system can achieve better localization accuracy and speed than the main state of the art. It obtains approximately 0.24m and 0.3° in less than 0.1 seconds.

1 Introduction

Image-Based localization (IBL) addresses the problem of estimating the 6 DoF camera pose in an environment, given a query image and a representation of the scene. It is an important step in many applications such as relocalization and loop closure [3, 6, 17], place recognition [23, 24] and Augmented Reality(AR) [1, 2].

There are two standard ways to solve the IBL problem: (1) 2D-2D image retrieval where a set of 2D features from the query image is matched against the 2D features from the database image, which essentially defines the image pose. (2) 2D-3D matching which consists of matching 2D features from a query image to a 3D point cloud map of the environment. The 3D map is usually extracted by using Structure from Motion (SfM) techniques [23, 24, 27, 28].

Pages 127.1-127.12
DOI: https://dx.doi.org/10.5244/C.30.127

IBL applications face several challenges that can affect their robustness, accuracy, and speed. First, feature matching is dependent on feature viewpoint invariance. Second, when dealing with large-scale scenes, where there are hundreds of thousands of 3D points and their associated descriptors, searching the space for the exact correspondences becomes challenging. The tree-based approach [14] is the standard solution for the search space reduction in IBL. This approach aims to speed up the search in 2D-3D by finding the approximate nearest neighbors with respect to all the 3D points in the map. When dealing with large-scale environments, matching against all those points becomes computationally expensive. Thus, the need to reduce the search space of the tree-based approach becomes the main focus. One of the notable solutions for reducing the search space is the use of Bag-of-Words (BoW) [3]. This approach is well known to trade accuracy for speed due to the quantization effect.

Our main contribution in this paper is to reduce the search space in IBL by using context and combining global and local descriptors using a SfM map. We propose the Gist-based Search Space Reduction (GSSR) system to reduce the search space by using global descriptors to find candidates keyframes in the database, then match against the 3D points that are only seen from these candidates. Eventhough GIST descriptors [26] have been used for many applications, especially for topological localization [15, 21, 22] and web image search[5], we appear to be at the forefront to present a keyframe approach with global descriptors in IBL.

Our approach is validated on a challenging indoor dataset and on two larges-scale datasets. Our system shows consistency by achieving better accuracy than the main state-of-the-art approaches, while maintaining considerable speedups. To our knowledge, we appear to be the first work to attain a better accuracy than the standard tree-based in IBL.

2 Related Work

There are different IBL approaches in the literature. Given a SfM point cloud, Irschara et al. [9] propose an image retrieval system for large scale scenes by creating synthetic views for the database images to cover the point cloud map, then use a vocabulary tree to retrieve similar database images for the query one. Li et al. [12] addressed the problem by introducing their co-occurrence RANSAC, consisting of a probabilistic model that relies on a visibility graph [11]. Then, they used 3D-2D to guarantee that a sufficient number of inliers is found. The disadvantage of the technique is that the co-occurrence check does not have any significant advantage when all features are from the same environment. Similarly, Sattler et al. [19] improved the IBL accuracy and speed from [11] by clustering the 3D points into bag-of-words. Their main idea is to actively search the surrounding of a 2D-3D match in order to identify its nearest neighbors, using vocabulary trees, and then conduct a 3D-2D matching step to recover the matches using their Vocabulary-based Prioritized System (VPS) [18]. The major shortcoming of this system is that is predisposed to errors, which are prevalent in the clustering stage.

Furthermore, Donoser [4] solved the image registration problem using classification. They performed 2D-3D matching similar to [19] but replaced the Approximate Nearest neighbor (ANN) by a discriminative classification step relying on random ferns. Their goal was to improve the feature matching by considering previous sightings of a specific feature as a class rather them reducing the search space. Their major shortcoming is the weakness of the classifier in global matching and its reliance on GPS tags to partition the search space into smaller regions. Heisterklaus [8] used another classification method to provide a better

localization accuracy. He started by classifying database images into multiple views using MPEG compact global descriptors. Then, he generated synthetic camera poses to cover all the remaining spaces in the environment in hopes of getting more robust correspondences in shorter time. The problem with this approach is its sensitivity to illumination changes, which is not surprising as pixel intensity values are used directly. Also, an inaccurate estimation of focal length and skew during calibration proved to be detrimental to the success of the system.

On the other hand, Shotton et al. [20] was the first to use deep learning to address the IBL problem. They used regression forests to train a featureless 3D map reconstructed from an RGB-D sensor in small indoor scenes. They performed 2D-3D matching and then relied on a modified preemptive RANSAC to account for the inliers matches. The major shortcoming in this method is the fact that it is designed for depth maps and not tailored for appearance-based solutions. In another work, Glocker et al. [7] used image retrieval techniques to match query images to keyframes in the database. Their method relied on random ferns for matching and performed 2D-2D matching. The last two approaches presented promising results for camera relocalization in IBL. The disadvantage of this method is its sensitivity to the distance between the keyframes and query since it aims to match images that are almost identical rather than images of the same scene taken from different viewpoints. Recently, Kendall et.al [10] used deep convolutional neural networks to solve the problem. Their idea was similar to the regression forest in [20]. They used the 3D poses from a SFM technique [28] as labels to train a pose regressor for image recognition. They used GoogleNet [25] classifier to minimize an objective loss function to regress the camera pose. Their results scored significantly high speed with good accuracy for location recognition applications where the accuracy is traded for faster speed. PoseNet is very sensitive to the scaling factor used in the objective function. It requires significant trial and error for each scene to find good scaling factor and weights. Also, the accuracy of this approach is enough for location recognition applications, but is not enough to compete with the accuracy of the main IBL systems.

All of the techniques show shortcomings in their results, particularly in search space reduction, feature matching, clustering and sensitivity to where the query image is taken. Finding a method that simplifies the search space problem and is less prone to false keypoints matches in an image is desirable to speed up to the localization while achieving better accuracy than the state-of-the-art.

3 GIST-based Search Space Reduction (GSSR)

We propose the Gist-Based Search Space Reduction (GSSR) IBL system to overcome scaling issues as well as other traits such as illumination variance. Figure 1 shows an overview of our system. GSSR relies on the GIST global scene descriptor [26], which is not dependent on illumination changes. While most search space reduction methods rely on bag of words to do so, GSSR relies on GIST descriptors to establish context for both the saved images in the database as well as for any query image.

The GIST descriptor proposed in [26] aims to develop a low-dimensional representation for each image. The proposed descriptor represents the dominant spatial structure of a scene. Thus, GIST summarizes the gradient information (scale and orientation) for different parts of an image. It starts by convolving the image with 32 Gabor filters at 4 scales, 8 orientations, producing 32 feature maps of the same size of the input image. Then, it divides each feature map into 16 regions of 4x4 grids. Finally, a 512 dimension GIST descriptor is computed by

concatenating the 16 averaged values of all 32 feature maps (16x32 = 512).

GSSR is presented in Algorithm 1. In a pre-processing offline stage, a 3D map of the target environment is built using SfM. The resulting map is parsed and contains the following: (1) 3D points + their SIFT [13] descriptors + keyframes in which each 3D point was observed. (2) Keyframes + a single GIST for each keyframe. Note that each 3D point will be represented by the mean value of all of its SIFT descriptors. Once GSSR is initiated, each query is processed to produce its own GIST descriptor as well as its SIFT descriptors. The GIST distance between the query and all the keyframes is computed (L2 norm of the GIST feature). If the distance is below a certain threshold than the keyframe is considered a candidate match, otherwise it is discarded since it does not belong to the same view of the query image. The threshold chosen here is determined empirically and can lead to unsuitable keyframes that do not share a large enough number of 3D points with the query image. In order to remove these outlier keyframes, a simple consensus test is done as a refinement step. Each candidate keyframe is checked with all the other candidates according to Eq. 1:

$$F_k = \frac{\sum_{i=1}^{N} P_i(KF_i, KF_k)}{N}, \tag{1}$$

where N is the total number of candidate keyframes and P_i is the number of 3D points in common between the tested candidate KF_k and the keyframe KF_i at i. P_i is computed in an offline stage after the map is reconstructed and parsed. If the ratio F_k is high enough the candidate keyframe qualifies for localization, otherwise it is discarded. The net result at this point is a constellation of keyframes that qualify for localization.

In the next stage, the search space is reduced by matching only to 3D points seen in the qualified keyframes. This consists of only considering 3D points that are viewed by the constellation of keyframes. The search for those 3D points is performed efficiently. First, the candidate keyframes are sorted based on the highest P_i scores, secondly we start gathering all the 3D map points in the ordered candidate keyframes. Then, 2D-3D matching between SIFT descriptors and the retained 3D points is done using Approximate Nearest Neighbor [14]. If more than 12 inliers are found, the image is registered and its pose is estimated. Otherwise, we increase the GIST distance threshold and repeat until 12 inliers are found or until 0.5 second has elapsed in the online stage. Note that GSSR is designed to always start with an

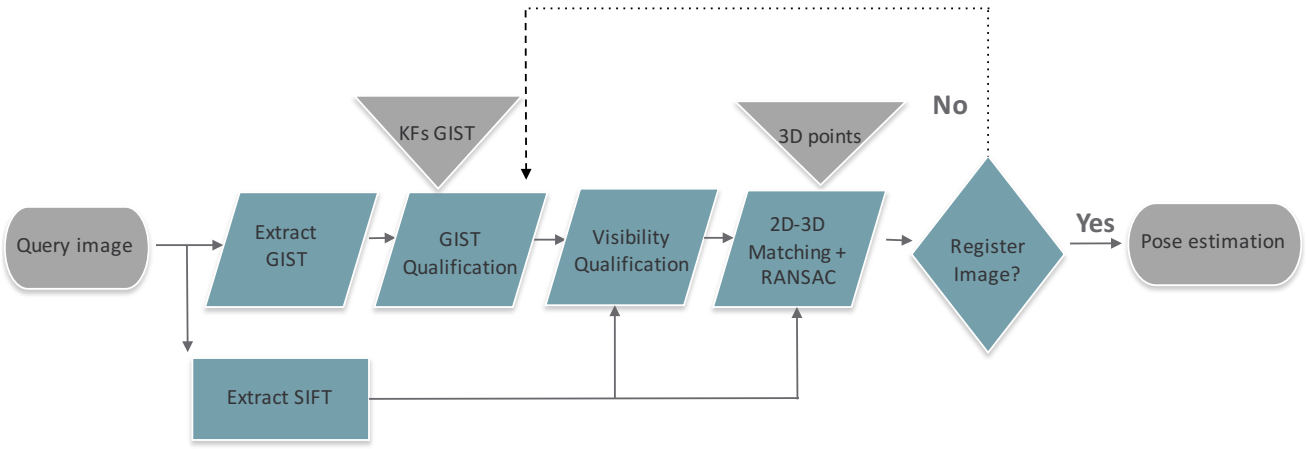


Figure 1: GSSR system overview, where the GIST of the query image is matched against all the Keyframes (KFs) GIST descriptors to qualify candidate KFs. The 3D points of those candidates will be matched to the SIFT features of the image before removing the outliers via RANSAC. Only images with enough inliers will qualify to the pose estimation step.

Algorithm 1: GSSR Algorithm

Get the GIST for each KF(Keyframe) + the 3D pts and all the Kf's each pt is visible in from VSFM map + the camera transformation estimates from VSFM
for *all database KFs* **do**
    $F_k = \frac{\sum_{i=1}^{N} P_i(KF_i, KF_k)}{N}$
Take a query image Q and extract its GIST
for *all database KFs* **do**
    Compute the cost $C(Q, KF_i)$ = GIST distance between Q and KF_i
    Sort $C(Q, KF_i)$ by ascending order
while # $Inliers_{RANSAC} < 12$ & $elapsedtime < 1s$ **do**
    Qualify KFs that have $C(Q, KF_{1:all}) < N_{(min)threshold}$
    for *All qualified KFs* **do**
        if KF_i *has enough number of visible 3D pts between the other KFs* **then**
            Qualify KF_i to localization step
        else
            Discard KF_i
    Take the 3D pts viewed only in the qualified KFs
    Perform a 2D-3D match between the query and those 3D pts
    Image Registration: Reject outliers via RANSAC and get the total # $Inliers_{RANSAC}$
    if # $Inliers_{RANSAC} < 12$ **then**
        Increase $N_{(min)threshold}$
Pose Estimation

initial GIST distance threshold of 0.15. The empirical value of 12 inliers is taken from the relevant body of literature [11].

4 Datasets

For the evaluation of GSSR, two large-scales datasets and one indoor dataset were used. The Cambridge Landmarks is a sequential dataset of 5 larges-scale urban scenes. The scenes consists in total of around 1.9M 3D points and 4081 query images originally reconstructed using VSFM [28]. This dataset is challenging as its images include dynamic changes such as pedestrians and vehicles, and the data was collected at different time changes representing different lighting and weather conditions. The Cambridge Dataset provided the testing models and training models separately including the 3D maps, SIFT descriptors and ground truth.

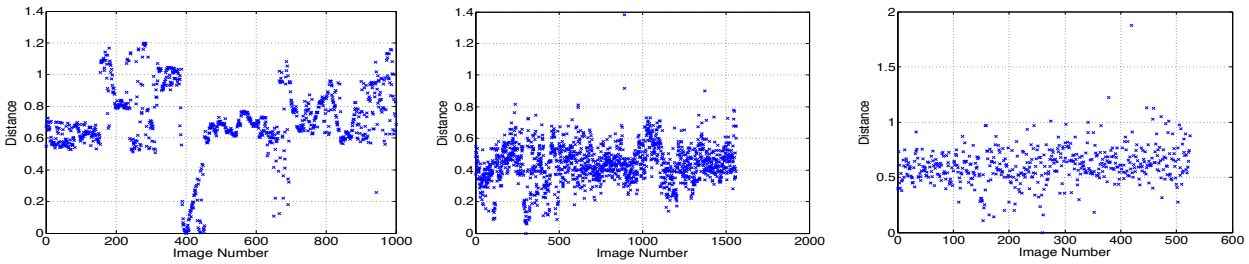
To demonstrate the robustness of our system for location recognition, we chose The Notre Dame dataset. It is a large scale set that consists of a large number of unordered images taken by Flickr users using different cameras and during different times. This set was originally reconstructed using Bundler [23, 24] but no training or test images nor the SIFT descriptors corresponding to the reconstruction were available. Thus we use all the images to obtain a 3D reconstruction via VSFM. The reconstructed version consists of 250k 3D points and 715 query images. To test on indoor sets specifically for relocalization applications, the Microsoft 7 scenes dataset [7, 20] is used. The 7 scenes dataset consists of seven different indoor locations; each originally mapped using an RGB-D Kinect camera, resulting in a

3D metric ground truth map for each scene. The choice of each scene represents different challenges, namely (1) motion blur and illumination changes, (2) flat surface and repetitive structures, and (3) reflectivity. The dataset offers training and testing images for each set, which were all used for reconstructing the scenes using VSFM [27, 28]. The reconstructed version consists in total of 600k 3D points and 17000 query images. For the test images in the 7 scenes and Notre Dame datasets, we remove their corresponding 3D points from the original reconstructed model, along with their corresponding SIFT descriptors and poses. Finally, we align the provided ground truth with our VSFM ground truth for those two datasets to obtain a metric scale.

5 Results

5.1 GIST Matching

Figure 2 shows the distance between the GIST descriptor of a query image taken at random and keyframes in the database for one randomly chosen scene of each of the three datasets. For example in 2(a) the query image is number 400 on the x axis and the clusters of points(KFs) around it are the ones who have the smallest distance. Note the close matching of the GIST descriptors of the keyframes located in the vicinity of Frame 400. Another cluster is noted around images number 600-700 where the user revisits part of the scene corresponding to Image 400. Results are similar on the other 6 scenes and on the Cambridge dataset 2(b). Note the challenging aspect of the Notre Dame dataset, where the database consists of unordered images. The cluster formed in Figure 2 shows the consistency of GIST in finding the similar database keyframes to the query image even in an unordered database. Figure 3 shows some of the retrieved images around the query image 300: we can see where two close groups of clusters and the corresponding candidate KFs retrieved for each group.



(a) Chess at query=400 (b) Kings College at query=300 (c) Notre Dame at query=260

Figure 2: This graph shows the GIST distance between a sample query image randomly taken from one scene for each of the three datasets to Keyframes in the corresponding datasets.

5.2 Performance of GSSR

The Cambridge dataset was used to benchmark GSSR against the main state-of-the-art approaches. For the evaluation against PoseNet [10], we used the provided training labels to train and test their system. The results came very close to the reported ones in [10]. As for ACG Localizer, we clustered the 3D map points into 25k clusters of BOW using [16] (except for the Shop Facade scene where 10k clusters were used). Table 1 shows that GSSR scores

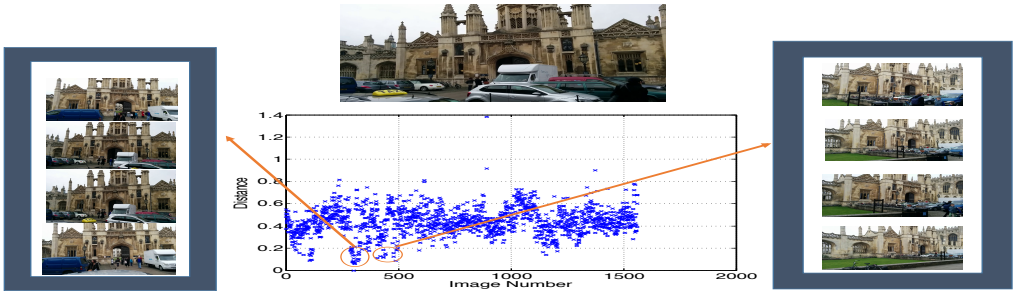


Figure 3: The candidate KFs (inside the left and right boxes) for the formed clusters for the Kings College scene from Figure 2(b) around the query image 300 (top image).

the best localization accuracy among all the approaches. GSSR, tree-based and ACG Localizer registered almost all the query images which is not shown in this table. Note that GSSR has 10 times better accuracy than PoseNet and significantly better than ACG Localizer which is one of the best large-scale IBL systems.

Table 2 presents the results of the GSSR approach benchmarked against the tree-based approach on both datasets. On all 7 scenes, GSSR produces superior localization accuracy. The differences are small due the small scale of the dataset but the improvement is always consistently in favor of GSSR. Also, on the Notre Dame dataset GSSR was better than tree-based (For Notre Dame GSSR starts with an initial distance threshold of 0.25 instead of 0.15). GSSR shows around 5% better localization accuracy than tree-based where the differences are considerable.

We further compare GSSR to the published results of Shotton et al. [7, 20] on the 7 scenes dataset using the decision forest [20] and the Keyframe approach [7] for relocalization application. The aim of this comparison is not to compare featureless methods to ours, rather to show the powerful relocalization aspect of GSSR. The results are shown in Table 3. The percentage accuracy reported in the table is calculated as the number of query images featuring a translational error less than 2 cm and at the same time a rotational error less than 2 degrees. Note the considerable improvement in GSSR over the other techniques, even for difficult scenes like Pumpkin and RedKitchen, featuring ambiguous scenes.

Table 1: GSSR performance benchmarked against tree-based [14], PoseNet [10] and ACG Localizer [19] on the Cambridge Dataset. Note that GSSR presents the best localization accuracy in less than 0.1 seconds

Dataset	# Train Images	# Query Images	Median Error				Average Time (s)			
			Tree-Based	PoseNet	ACG Localizer	GSSR	Tree-Based	PoseNet	ACG Localizer	GSSR
Kings College	1220	343	0.229m, 0.194deg	1.992m, 3.261deg	0.910m, 0.761deg	0.213m, 0.188deg	1.8	0.01	0.74	0.102
StMarys Church	1487	530	0.180m, 0.424deg	2.645m, 5.102deg	0.662m, 0.634deg	0.175m, 0.309deg	4.852	0.01	0.51	0.105
Old Hospital	895	182	0.341m, 0.272deg	2.441m, 2.923deg	1.044m, 0.846deg	0.299m, 0.228deg	1.179	0.01	0.33	0.065
Shop Façade	231	103	0.138m,0.220deg	1.490m, 4.299deg	0.548m, 0.575deg	0.140m, 0.217deg	0.302	0.01	0.39	0.044
Street	3015	2923	0.410m, 0.672deg	3.910m, 3.75deg	0.972m, 1.004deg	0.364m,0.539deg	11.43	0.01	0.77	0.109
Average			***0.260m, 0.358deg***	***2.496m, 3.867deg***	***0.872m, 0.772deg***	***0.238m, 0.296deg***	***3.91***	***0.01***	***0.548***	***0.085***

Regarding the efficiency, GSSR was able to accurately localize a query image in less than 0.1 sec which makes it the fastest feature-based IBL system for large-scale scenes. Although, PoseNet scores the best computational time it is worth to note that PoseNet is a CNN system that trade accuracy for speed for fast location recognition applications, whereas GSSR does not do any trade-offs. Note that similar to most IBL work [7, 12, 19], the reported times exclude the time needed for SIFT and GIST feature extraction. The reported times include the time for GIST matching for selecting the candidate keyframes, searching for the 3D

Table 2: GSSR performance benchmarked against the tree-based approach on the Microsoft 7 scenes and Notre Dame datasets. Note that GSSR presents better accuracy in less than 20 ms on the 7 scenes set and less than 0.1 seconds on the Notre Dame set

Dataset		#Train Images	# Query Images	Mean Errors		Average Time(s)	
				GSSR	Tree-Based	GSSR	Tree-Based
Microsoft 7 Scenes	Chess	4000	2000	0.315cm, 0.2914deg	0.3235cm, 0.3139deg	0.014	0.107
	Fire	2000	2000	0.5412cm, 0.148deg	0.5557cm, 0.149deg	0.016	0.201
	Heads	1000	1000	0.7275cm, 0.226deg	0.7379cm, 0.232deg	0.008	0.093
	Office	6000	4000	0.2814cm, 0.234deg	0.2959cm, 0.238deg	0.018	0.098
	Pumpkin	4000	2000	0.2839cm, 0.291deg	0.2902cm, 0.311deg	0.012	0.054
	RedKitchen	7000	5000	0.4842cm, 0.151deg	0.4922cm, 0.156deg	0.019	0.108
	Stairs	2000	1000	2.751cm, 1.647deg	2.8517cm, 1.652deg	0.012	0.031
	Average			***0.784cm, 0.426deg***	***0.793cm, 0.436deg***	***0.014***	***0.099***
Notre Dame		**715**	**715**	2.229m, 1.814deg	2.331m, 1.871deg	0.095	1.551

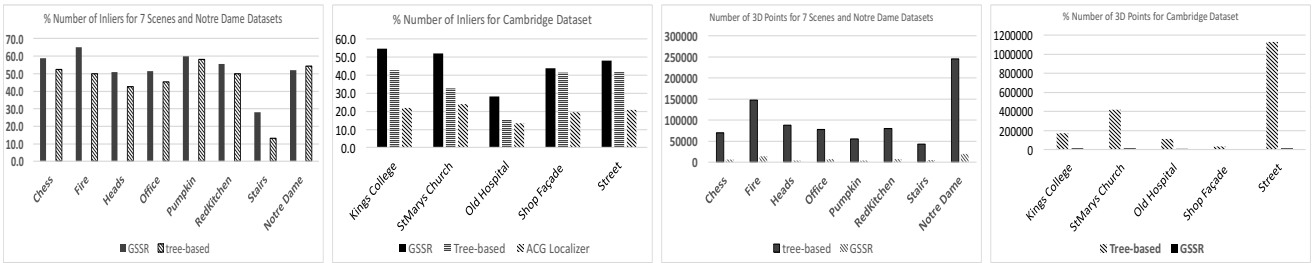
Table 3: GSSR performance in terms of accuracy benchmarked against the tree-based [14], Decision Forest [20], and Keyframe approach [7] on the 7 scenes dataset

Dataset	#Query Images	% Accurate Images			
		GIST Approach	Tree-Based	Decision Forest	Keyframe Approach
Chess	2000	97.4	97	92.6	85.3
Fire	2000	98	96.5	82.9	72
Heads	1000	93.2	93	49.4	79.8
Office	4000	99.2	98.8	79.1	74.7
Pumpkin	2000	99.3	98.1	73.7	62.8
RedKitchen	5000	97.5	96	72.9	54.1
Stairs	1000	40.3	36	27.8	34.1
Average		***89.3***	***87.9***	***68.3***	***66.1***

visible points, 2D-3D matching and, the RANSAC registration. The average searching time for the visible 3D points, which is not reported in the table, consists of 7% of the total computational time.

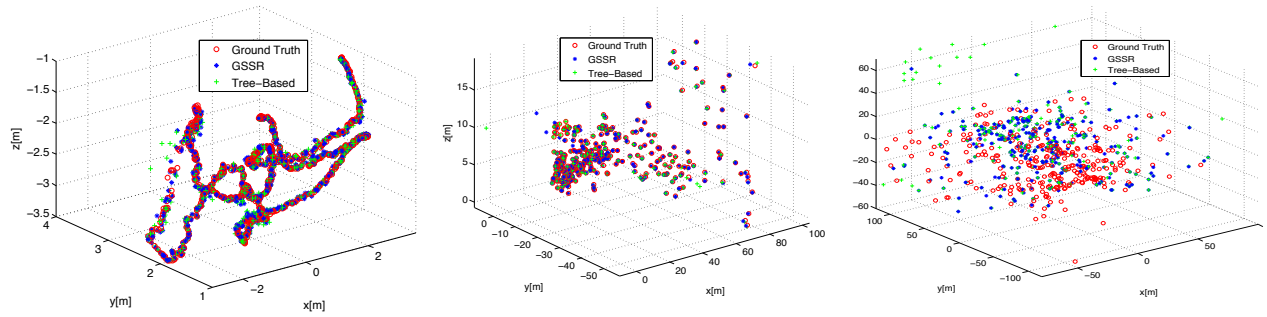
Furthermore, Figures 4(a) 4(b) show that using GSSR there is a notable increase in the percentage of inliers on the 7 scenes and Cambridge datasets (ACG Localizer is included) whereas it slightly decreased on the Notre Dame dataset. Note that PoseNet is not a feature-based method so there is no outlier rejection method. This illustrates the robustness of GSSR in outputting better quality inliers. In terms of the 3D points used for each scene, note in Figures 4(c) 4(d) the efficiency of GSSR at reducing the search space, reaching an average of 91% while maintaining a better accuracy than tree-based.

Table 4 shows the effect of the GIST distance threshold on the accuracy and time which are better reflected on the Notre Dame dataset. The results on 100 test images show that the accuracy was better for a threshold of 0.3 and 0.35 but at slower computational times and lower search space reduction percentage. Whereas at thresholds of 0.4 the accuracy suddenly



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

Figure 4: 4(a) 4(b) shows the average numbers of inliers for each scene. 4(c) 4(d) shows the number of 3D points initially in the map and after the GSSR for each scene



(a) (b) (c)

Figure 5: The 3D camera poses using GSSR (*) and tree-based IBL (+) for RedKitchen scene, Kings College scene, and Notre Dame dataset. Ground truth is shown as (o). Note the accuracy and consistency of GSSR in following the ground truth path of the camera.

decreases to slightly match the tree-based one and remain slightly constant at 0.45. Results are very similar on the other datasets. These results justifies the initial threshold we use as it consists of the best trade off between accuracy and speed.

Table 4: GIST distance threshold effect for 100 test images on Notre Dame dataset

Approach		R Error (Deg) (Mean,SD)	T Error (m) (Mean,SD)	Average Time(s)	% Inliers	% Search Space Reduction
GSSR	**d = 0.25**	1.377	1.944	0.493	61.9	90.9
GSSR	**d = 0.3**	1.275	1.879	0.567	63.6	86
GSSR	**d = 0.35**	1.307	1.823	0.651	64.7	81.4
GSSR	**d = 0.4**	1.345	2.039	0.764	76.2	75.5
GSSR	**d = 0.45**	1.481	2.041	0.894	76.4	69.4
Tree-based		1.444	2.027	1.57	65.2	-

6 Discussion

The results demonstrate the accuracy and robustness of GSSR. Indeed, the fact that GSSR outputs better quality of matches than tree-based justifies the results of higher accuracies. Since the 2D features of a query image is matched to 3D points belonging exclusively to the constellation of keyframes, a better number of correct matches are more likely to be found. This is reflected on all the scenes of Cambridge and the 7 scenes datasets, since they are composed of ordered set of images. Exceptionally, Notre Dame dataset scored a higher inliers ratio than GSSR, as shown in table 4. This is due to the fact that Notre Dame is a dataset of unordered images which shows that correct matches guarantee a better accuracy, not a higher inlier ratio for this type of datasets.

Figure 5 shows the camera 3D trajectories on the RedKitchen and Kings College scenes, and on the Notre Dame dataset. Note the accuracy and consistency of GSSR in following the ground truth path of the camera. The graph also shows a smoother path than the tree-based approach, especially in regions where tree-based drifts away from the ground truth.

GSSR showed the best localization accuracy and was the fastest feature-based IBL system. PoseNet was 10 times faster than GSSR but 10 times less accurate, where one has to note that PoseNet is designed for fast image recognition. PoseNet aims to do a fast location recognition with minimal accepted accuracy. This comparison shows that if PoseNet desires

to improve its accuracy, it will have to do something similar to all the feature-based IBL approaches, which will make it much slower. PoseNet is very sensitive to the initial weights given and to the scaling factor used in the objective function. It requires significant trial and error for each scene to find good scaling factor and weights. As for ACG localizer, multiple kmeans clustering trials were made and the results varied a lot with each trial. The system is very sensitive to clustering where it struggled to register a lot of images from the first time, the fact that explains its high computational time on some scenes. In short, ACG Localizer suffered from the quantization effect where it missed the best correspondences because the matched descriptor was not assigned to the correct visual word. It is also notable that there was significant drop in accuracy for both GSSR and tree-based on the Stairs scene which is probably due to its repetitive structure.

GSSR showed notable improvement over the Decision Forest and Keyframe systems, which performed decently on the Pumpkin and RedKitchen datasets. This is probably due to the reflectivity nature of those scenes, where the ground, the cupboards and the kitchen structure cause lots of reflections in the images, which may result in false positive matches. Nevertheless, GSSR performs generally poorly with structures such as the Stairs dataset but better than state-of-the art relocalization approaches, which register very few images. This is due to the repetitive nature of the structure of the scene, which causes false positive matches.

Finally, GSSR showed robustness on the challenging Notre Dame and Cambridge datasets with all the illumination changes, reflections and environment changes (environment changes, different cameras used, etc). Nevertheless, few query images had a minimum threshold distance to its closest keyframe that was higher than the one adopted for testing, thus GSSR rejected them totally whereas tree-based registered these images while reporting a high localization error(>70 m). This tradeoff between accuracy and speed will be considered for future work.

7 Conclusion

This paper presented a novel approach, namely the GIST-based Search Space Reduction, for reducing the search space in Image-Based Localization. GSSR was benchmarked against the Cambridge 5 scenes dataset, the 7 scenes indoor dataset of Microsoft and the Notre Dame dataset. Results show better localization accuracy than the main state-of-the art approaches on all datasets due to the powerful ability of GSSR to find better quality inliers. At the same time, GSSR showed considerable speed-ups in computational times, where a query image will be localized in less than 0.1 seconds, which makes it the fastest feature-based IBL system. The speed-up is primarily due to the ability of GSSR to considerably reduce the search space and yet produce superior accuracy than other state of the art techniques. Also, the results also show that a higher inlier ratio does not necessarily guarantee a better localization accuracy.

As future work, we plan to investigate an alternative for the GIST distance threshold in order to better address the large scale problems in terms of both accuracy and efficiency.

Acknowledgments

This work was supported by the Natural Sciences and Engineering Research Council of Canada (NSERC) and the Ontario Centers of Excellence (OCE).

References

[1] Clemens Arth, Daniel Wagner, Manfred Klopschitz, Arnold Irschara, and Dieter Schmalstieg. Wide area localization on mobile phones. In *Mixed and Augmented Reality, 2009. ISMAR 2009. 8th IEEE International Symposium on*, pages 73–82. IEEE, 2009.

[2] Robert Castle, Georg Klein, and David W Murray. Video-rate localization in multiple maps for wearable augmented reality. In *Wearable Computers, 2008. ISWC 2008. 12th IEEE International Symposium on*, pages 15–22. IEEE, 2008.

[3] Mark Cummins and Paul Newman. Fab-map: Probabilistic localization and mapping in the space of appearance. *The International Journal of Robotics Research*, 27(6): 647–665, 2008.

[4] Michael Donoser and Dieter Schmalstieg. Discriminative feature-to-point matching in image-based localization. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 516–523, 2014.

[5] Matthijs Douze, Hervé Jégou, Harsimrat Sandhawalia, Laurent Amsaleg, and Cordelia Schmid. Evaluation of gist descriptors for web-scale image search. In *Proceedings of the ACM International Conference on Image and Video Retrieval*, page 19. ACM, 2009.

[6] Ethan Eade and Tom Drummond. Scalable monocular slam. In *Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on*, volume 1, pages 469–476. IEEE, 2006.

[7] Ben Glocker, Jamie Shotton, Antonio Criminisi, and Shahram Izadi. Real-time rgb-d camera relocalization via randomized ferns for keyframe encoding.

[8] Iris Heisterklaus, Ningqing Qian, and Artur Miller. Image-based pose estimation using a compact 3d model. In *Consumer Electronics Berlin (ICCE-Berlin), 2014 IEEE Fourth International Conference on*, pages 327–330. IEEE, 2014.

[9] Arnold Irschara, Christopher Zach, J-M Frahm, and Horst Bischof. From structure-from-motion point clouds to fast location recognition. In *Computer Vision and Pattern Recognition, 2009. CVPR 2009. IEEE Conference on*, pages 2599–2606. IEEE, 2009.

[10] Alex Kendall, Matthew Grimes, and Roberto Cipolla. Posenet: A convolutional network for real-time 6-dof camera relocalization. In *Proceedings of the IEEE International Conference on Computer Vision*, pages 2938–2946, 2015.

[11] Yunpeng Li, Noah Snavely, and Daniel P Huttenlocher. Location recognition using prioritized feature matching. In *Computer Vision–ECCV 2010*, pages 791–804. Springer, 2010.

[12] Yunpeng Li, Noah Snavely, Dan Huttenlocher, and Pascal Fua. Worldwide pose estimation using 3d point clouds. In *Computer Vision–ECCV 2012*, pages 15–29. Springer, 2012.

[13] David G Lowe. Distinctive image features from scale-invariant keypoints. *International journal of computer vision*, 60(2):91–110, 2004.

[14] Marius Muja and David G Lowe. Fast approximate nearest neighbors with automatic algorithm configuration. *VISAPP (1)*, 2:331–340, 2009.

[15] Ana C Murillo, Gagan Singh, Jana Kosecka, and José Jesús Guerrero. Localization in urban environments using a panoramic gist descriptor. *Robotics, IEEE Transactions on*, 29(1):146–160, 2013.

[16] James Philbin, Ondřej Chum, Michael Isard, Josef Sivic, and Andrew Zisserman. Object retrieval with large vocabularies and fast spatial matching. In *Computer Vision and Pattern Recognition, 2007. CVPR'07. IEEE Conference on*, pages 1–8. IEEE, 2007.

[17] Duncan P Robertson and Roberto Cipolla. An image-based system for urban navigation. In *BMVC*, pages 1–10. Citeseer, 2004.

[18] Torsten Sattler, Bastian Leibe, and Leif Kobbelt. Fast image-based localization using direct 2d-to-3d matching. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 667–674. IEEE, 2011.

[19] Torsten Sattler, Bastian Leibe, and Leif Kobbelt. Improving image-based localization by active correspondence search. In *Computer Vision–ECCV 2012*, pages 752–765. Springer, 2012.

[20] Jamie Shotton, Ben Glocker, Christopher Zach, Shahram Izadi, Antonio Criminisi, and Andrew Fitzgibbon. Scene coordinate regression forests for camera relocalization in rgb-d images. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 2930–2937. IEEE, 2013.

[21] Christian Siagian and Laurent Itti. Biologically inspired mobile robot vision localization. *Robotics, IEEE Transactions on*, 25(4):861–873, 2009.

[22] Gautam Singh and J Kosecka. Visual loop closing using gist descriptors in manhattan world. In *ICRA Omnidirectional Vision Workshop*. Citeseer, 2010.

[23] Noah Snavely, Steven M Seitz, and Richard Szeliski. Photo tourism: exploring photo collections in 3d. *ACM transactions on graphics (TOG)*, 25(3):835–846, 2006.

[24] Noah Snavely, Steven M Seitz, and Richard Szeliski. Modeling the world from internet photo collections. *International Journal of Computer Vision*, 80(2):189–210, 2008.

[25] Christian Szegedy, Wei Liu, Yangqing Jia, Pierre Sermanet, Scott Reed, Dragomir Anguelov, Dumitru Erhan, Vincent Vanhoucke, and Andrew Rabinovich. Going deeper with convolutions. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition*, pages 1–9, 2015.

[26] Antonio Torralba, Aude Oliva, Monica S Castelhano, and John M Henderson. Contextual guidance of eye movements and attention in real-world scenes: the role of global features in object search. *Psychological review*, 113(4):766, 2006.

[27] Changchang Wu. Towards linear-time incremental structure from motion. In *3DTV-Conference, 2013 International Conference on*, pages 127–134. IEEE, 2013.

[28] Changchang Wu, Sameer Agarwal, Brian Curless, and Steven M Seitz. Multicore bundle adjustment. In *Computer Vision and Pattern Recognition (CVPR), 2011 IEEE Conference on*, pages 3057–3064. IEEE, 2011.