What Should I Landmark? Entropy of Normals in Depth Juts for Place Recognition in Changing Environments Using RGB-D Data

Daniel Gutiérrez-Gómez[1], Walterio Mayol-Cuevas[2] and J.J. Guerrero[1]

***Abstract*— One open problem in the fields of place recognition and mapping is to be able to recognise a revisited place when its appearance and layout have changed between visits. In this paper, we investigate this problem in the context of RGB-D mapping in indoor environments. We propose to segment the scene in *juts* (neighbourhood of 3D points with normals that stick out from the surroundings) and look at low-level features, like textureness or entropy of the normals. These could differentiate those zones of the scene that change or move along time from those that are likely to remain static. We also present a method which improves the matching between images of the same place taken at different times by pruning details basing on these features. We evaluate on a number of communal areas and also on some scenes captured 6 months apart. Experiments with our approach, show an increase up to 70% in inlier matching ratio at the cost of pruning only less than 20% of correct matches, without the need of performing geometric verification.**

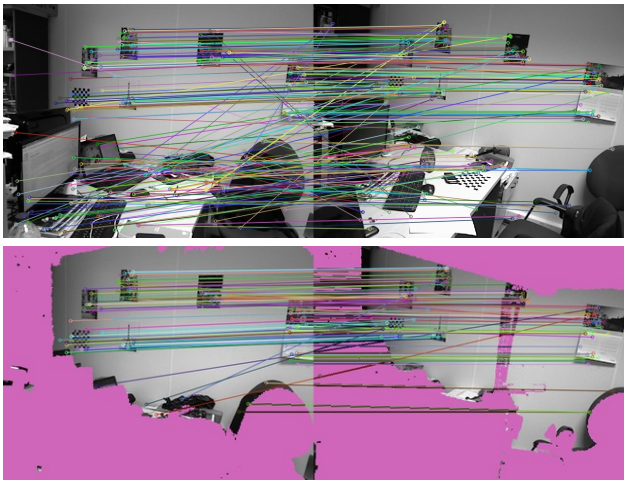


Fig. 1. Depth-based detection of zones which moved between mappings. A simple masking based on the entropy of different zones of the scene improves the inliers ratio of matched visual features.

I. Introduction

In computer vision, the problem of place recognition consists in being able to tell if two given images correspond to the same scenario or not. Place recognition is the key element to perform topological mapping [20], but also it is important in the context of geometric localisation and mapping to relocalise when the system is lost or to be able to close loops when revisiting previously mapped areas [22].

Robust place recognition is a fundamental step to perform life long mapping. In [12], Konolige and Bowman give a concise definition of what a lifelong map implies. A lifelong map system must carry out an incremental mapping, be able to operate in dynamic environments and to relocalise and close loops by recognising revisited places when given the chance. Focusing on the dynamic environment problem, two different issues are pointed: ephemeral objects, which move at the same time a zone is being mapped, and long-term changes, which involve changes in the scene which take place between different tracks of the scene. Ephemeral objects are usually ignored during mapping when using the traditional approaches such as visual odometry, and this makes such maps brittle beyond a few moments after being captured. In some cases, optimisation techniques based on RANSAC [3] or robust M-estimators [11] could be used to diminish the effect of ephemeral objects on the localisation and mapping, but this can only be considered an indirect approach to address the fundamental nature of scene change.

Naturally occurring long-term scene changes, such as objects on a common room or bedroom appearing, disappearing or changing places can affect loop closure detection and this thus must be dealt with at the place recognition stage. Standard approaches for place recognition though perhaps featuring some robustness to scene changes, do not handle these alterations actively nor identify where potential future changes may occur.

In this paper we address the issue of place recognition under long-term changes. Concretely we focus on robust place recognition with RGB-D cameras in indoor scenes. From the point cloud and the RGB images we can compute low-level features of parts of the scene like textureness, planarity or entropy of normals. This information can be very valuable to both decide where not to place landmarks but also to identify moving scene objects.

To define and evaluate our proposed method we perform two types of experiments. The first experiment is performed with images taken at fixed poses of different real scenes over which we have no control of changes and over different days, and where Ground Truth of static and changing zones is available. This experiments aims to inform us on which features are better to discriminate static from changing scene parts.

In the second experiment, we compute 3D maps of some of the scenes on different days with KinectFusion method

*This work was supported by spanish project DPI2012-31781, FEDER and FPU scholarship AP-2012-5507.

[1]Daniel Gutiérrez-Gómez and Josechu Guerrero are with the Departamento de Informática e Ingeniería de Sistemas (DIIS) y el Instituto de Investigación en Ingeniería de Aragón (I3A), Universidad de Zaragoza, Spain. {danielgg,josechu.guerrero}@unizar.es

[2]Walterio Mayol-Cuevas is with the Department of Computer Science, University of Bristol, UK. wmayol@cs.bris.ac.uk

[13], which is more akin to the case of a system of lifelong mapping for a personal or robotic device. Here we evaluate the matching between snapshots from reconstructions on different days by masking out zones below a given score, based on the previously studied features. We show how this masking improves the inlier ratio (Fig. 1).

II. Related work

The work of Konolige and Bowman [12] is focused towards efficient storage, keeping a map whose size scales with the explored area instead of with time. Keyframes are clustered by appearance similarity, trying to keep the maximum number of clusters, while fixing the maximum number of total stored keyframes. However this work does not address the problem of maintaining matches over long term changes, but propose the idea of picking up stable features as a path for improvement of long-term life-long mapping.

In the context of place recognition with RGB-D cameras Gee and Mayol-Cuevas [8] pointed out the robustness to map changes as one important property for place recognition. They propose a regression of small synthetic views of 80x60, or even 20x15 pixels, which offers a degree of tolerance to changes in order to recognise revisited places. Whelan et al. [21] focus on using optimisation to produce a consistent map deformation when closing the loop and address the problem of place recognition by using a visual based bag of words scheme with SURF descriptors.

Also, when considering RGB-D sensors, many authors have focused in discovering moving objects in a scene. This problem is related to lifelong mapping in the sense that moving objects are precisely what we want to discard when performing place recognition. Herbst et al. [9] discover objects in the scene after aligning two 3D maps of the same scene with different objects. For the alignment they assume that the moving objects occupy a small fraction of the map. Finman et al. [7] detect objects from changes in maps and then train a segmentation method to segment the discovered objects in future runs.

Karpathy et al. [10] propose a method to discover objects, which in contrast does not rely in an object displacement to detect them. Instead they perform a non-semantic segmentation [6] of the scene based on the map of normals, and then each segment is ranked with different objectness features. To avoid ambiguities, in this work we name the 3D segments as superjuts as the result of extending the concept of superpixel to 3D points with normals.

However, when dealing with lifelong mapping with RGB-D, though recognising specific objects can be useful, it would be desirable to discard anything which can move, no matter which object it is or if it has been seen or not before. To do this, we can take advantage of some results and proposals from works in object discovery, like for example a non-semantic superjut segmentation. But instead of detecting objects explicitly, we propose the use of statistics and entropy of low-level features to identify areas that are good and those that are likely to be unstable landmarks. This approach removes the elusive definition of object.

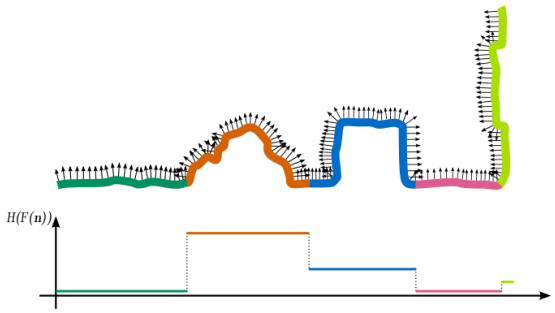


Fig. 2. 2D simplified scheme of the segmentation in superjuts and the entropy of the normals $H(F(\mathbf{n}))$ of each superjut.

When looking for features which characterise movable objects, some ideas can also be drawn from supervised segmentation methods for RGB-D scenes [2],[14],[16],[4]. These methods usually classify the scene into coarse classes (*e.g.*walls, floor, furniture, and props). The "props" class usually makes reference to objects which can be easily carried, which corresponds precisely to the kind of objects we want to remove from place recognition routines. Thus, features used in these methods as inputs for categorisation can provide some clues for the task of discarding movable objects.

III. Proposed method

Our proposal for robust place recognition tries to detect zones which can move no matter which objects are in them, using the information obtained from both 3D point clouds and RGB images. Fig. 2 shows an intuition of the proposed method. With a point cloud segmentation algorithm based on the similarity of normals between adjacent points [10], a point cloud is segmented in superjuts, represented with different colors. Then at each superjut we extract low level features which allow to discriminate static from moving parts of the scene. Looking at everyday indoor environments one can probably note that zones of the environment which are likely to change often show a high degree of derangement, which yields a more cluttered map of normals and more textured areas in the image. In the figure for example, the entropy $H(F(\mathbf{n}))$ of the histogram of normals is represented, noting that for superjuts with irregular shape is greater than in superjuts composed by planar surfaces.

A. Point cloud segmentation

First we use the method proposed in [19] and extended in [10] to segment a dense 3D point cloud in objects or small groups of objects. It is based on the segmentation algorithm by Felzenszwalb and Huttenlocher [6], but applied on a map of normals instead of a RGB image. A graph $G = (V, E)$ is constructed, where each vertex in V represents a 3D point $\mathbf{X}_i$ and the edges E represent the neighbouring relations between points. Having the normals at every point, a weight w_{ij} for the edge joining vertices i and j is computed:

$$w_{ij} = \begin{cases} (1 - \mathbf{n}_i^T \mathbf{n}_j)^2 & \text{if } \mathbf{n}_j^T(\mathbf{X}_j - \mathbf{X}_i) > 0 \\ 1 - \mathbf{n}_i^T \mathbf{n}_j & \text{if } \mathbf{n}_j^T(\mathbf{X}_j - \mathbf{X}_i) \leq 0 \end{cases} \quad (1)$$

where the squared weight is applied to convex edges, reflecting the fact that convex regions usually contain points belonging to the same object and concave regions are likely to arise in frontiers between objects. After computing the weights, the segmentation algorithm is run and essentially groups points sharing edges with low weights in the same segment. A parameter k must be tuned such that the higher k, the larger segments will be obtained. In this work we always use $k = 0.6$ for all the experiments.

Depending on the characteristics of the point cloud we compute the normals and edges required for segmentation in two different ways.

Triangular mesh: The first option is to build a triangular mesh from a given point cloud. This is specially preferable if we have dense 3D map resulting from mapping as the camera moves, e.g. with KinectFusion [13] approach. This way, an edge between two points $\mathbf{X}_i$ and $\mathbf{X}_j$ is created if they have any triangle, among the ones they belong to, in common. The normal for a point is computed as the average of the normals of the triangles it belongs to. Using this approach the segmentation algorithm is like the one proposed in [10], resulting in a set of superjuts.

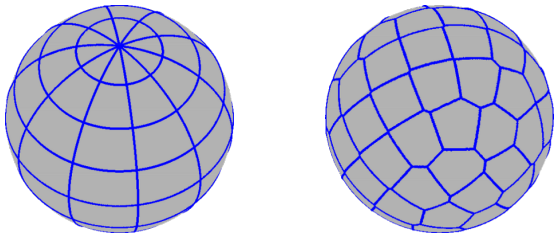
Image domain: Alternatively the point cloud can be projected onto an image domain, for example if the point cloud has been cast from a single depth image and the 3D map might be so noisy to build a triangular mesh. In this case the segmentation would be done directly in image projected superjuts, with the edges being created between adjacent pixels horizontally, vertically and diagonally. The normal for a point a 3D point projected in pixel (k, l) is computed as:

$$\mathbf{n}_{k,l} = \frac{(\mathbf{X}_{k,l+1} - \mathbf{X}_{k,l}) \times (\mathbf{X}_{k+1,l} - \mathbf{X}_{k,l})}{||(\mathbf{X}_{k,l+1} - \mathbf{X}_{k,l}) \times (\mathbf{X}_{k+1,l} - \mathbf{X}_{k,l})||} \quad (2)$$

B. Computation of superjut low-level features

Once the segmentation is done, we extract the low-level features to discriminate static from moving parts of the scene. We take the hypothesis that parts which are likely to move can be discriminated by their 3D structure and their textureness.

1) Structure by histogram of normals: To gather information about the structure of each superjut we compute its histogram of normals $F(\mathbf{n})$. Since normals are 3-dimensional unit vectors, they can be represented as points over the surface of a unit sphere. Thus, a first attempt would be to compute a 2-dimensional histogram by binning the angles of azimuth ϕ and elevation θ as proposed in [17]. However we noted that this binning might not be adequate if we want to obtain a good distribution of the bins on the sphere. This is graphically shown in Fig. 3a, where it can be noted that the area covered by each bin greatly changes with the elevation angle. Also it must be noted that points situated near the poles would be spread among all the confluent bins. These two facts can severally affect the quality of the histogram.



(a) (b)

Fig. 3. Two possible binnings of the 3D sphere to compute the histogram of normals: (a) by discretising azimuth and elevation angles and (b) by approximate uniform distribution of points in the sphere. The number of bins is set to $M = 80$ in both cases.

To overcome these problems we opt instead for the computation of a 1-D histogram with M bins. Each bin is a Voronoi cell corresponding to one point out of M points uniformly distributed over the sphere. Since there is no analytical solution to the problem of evenly distributing M points on the sphere for any M, we use a simple solution based on the golden section spiral [1], which results in a discretisation with bins covering areas of similar size and avoids the confluence of more than 4 bins in a single corner (Fig. 3b).

Given the histogram of normals, we extract the following properties:

- Entropy of the histogram $ent(F) = \sum_{i=1}^{M} F(\mathbf{n}_i) \log(F(\mathbf{n}_i))$.
- Planarity, measured as the relative frequency of the dominant normal $plan(F) = \max F(\mathbf{n})$.
- Horizontality, given by $hor(F) = \mathbf{e}_\mathbf{y}^T \mathbf{n}_{mode}$, with $\mathbf{n}_{mode} = \arg\max_\mathbf{n} F(\mathbf{n})$, where $\mathbf{e}_\mathbf{y}^T = (0\ 1\ 0)$.

2) Textureness from the eigen-transform: To obtain texture information we apply the eigen Transform proposed by Targui et al. [18] over the grey scale images. This approach produces a greyscale map, where more textured areas of the image yield a higher response than less textured ones. Then for each superjut projected in a superpixel over the intensity image, we compute the mean eigentransform value over the region it covers.

IV. Experiments

We have performed two experiments. In the first experiment we recorded different scenarios from fixed poses in order to capture the changes which took place during time and facilitate a Ground Truth separation of moved and static zones, on which discriminative power of structural and texture characteristics can be evaluated.

In the second experiment, we recorded sequences of 3 scenarios on different days with a moving camera and reconstructed a 3D volume to evaluate the use of structural information with more accurate depth maps in the context of place recognition. To demonstrate the performance even with very long term changes one set of sequences was acquired more than 6 months after the first one.

Fig. 4. Images taken on 9 different scenes at different locations (2 labs and 1 beedroom).

A. Evaluating features for detection of moved areas

RGB and depth images have been taken on 9 scenes at different locations: two laboratories at different universities and one bedroom (Fig. 4). For each scene, images have been acquired on different days to capture the changes in their layout. Each day, the camera is carefully positioned at each of the selected poses so that the same scene is captured as in the acquisitions of previous days. To diminish the effect of illumination noise in intensity images, and decrease the holes in the depth images, we captured frames during some seconds and computed the average of all. In the averaging of the depth image, hole pixels are zero-weighted.

The first step is building a Ground Truth to separate static and moved zones of the different scenes. To do so, first, images of to the same scene on different days are robustly aligned to reduce the disparity due to non exact camera positioning with respect to previous days. To do so we use a similar scheme as for computing the RGB-D dense visual odometry [8].

Once we have the aligned images and depth maps, we have to extract the parts of the scenes which may correspond to moved objects. This is done by computing the disparity in the depth maps. Note, however, that the existence of a disparity does not imply a moved object since it might correspond to a static area in the background revealed by an object which was occluding it. For a depth map D_i of a scene M, the parts which have changed are those which show a negative disparity in any difference with every other of the N_M depth maps of that scene. Matemathically we can express this by:

$$maskMoved_i = \left(\sum_{j=1}^{N_M} \mathbf{1}_{x>th} \left(\text{medFilt}(D_j - D_i) \right) \right) > 0 \quad (3)$$

where $\mathbf{1}_{x>th}$ is an indicator function. To eliminate disparity caused by a non-perfect alignment between frames, all the disparity maps are median filtered with a window size of 10x10, and the threshold for the indicator function is set to $50mm$. After this step we have N_M masks per scene extracting the changing zones.

Once we have aligned RGB and depth images as well as computed the masks, we follow the pipeline shown in Fig. 5 to obtain the map of normals, the Eigen-transform and the segmentation, divided in static and moved superjuts. This data is used to compute both the structural and textureness features for the superjuts as described in section III-B.

The final results shown in Fig. 6 support the initial intuition of moving areas being more likely to present more anarchic variations on the normals as they tend to concentrate in zones of higher entropy and lower planarity. It is shown also a correlation between entropy and planarity, which is not a surprise since the existence of a dominant bin in a histogram reduces its entropy. Texture information from the Eigen transform and horizontality seems not to be very useful to discriminate static from moved areas.

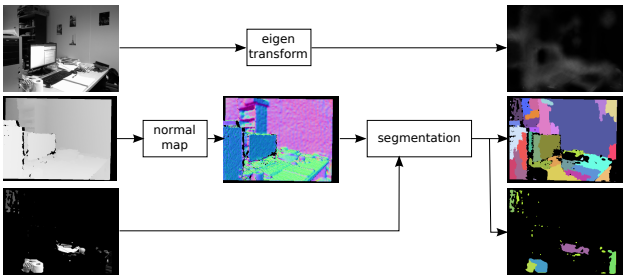


Fig. 5. Starting with the aligned RGB and depth images and the masks for moved objects, we compute the eigen transform as well as a segmentation separated in static and moved superjuts. This data is used to compute superjut features for the first experiment.

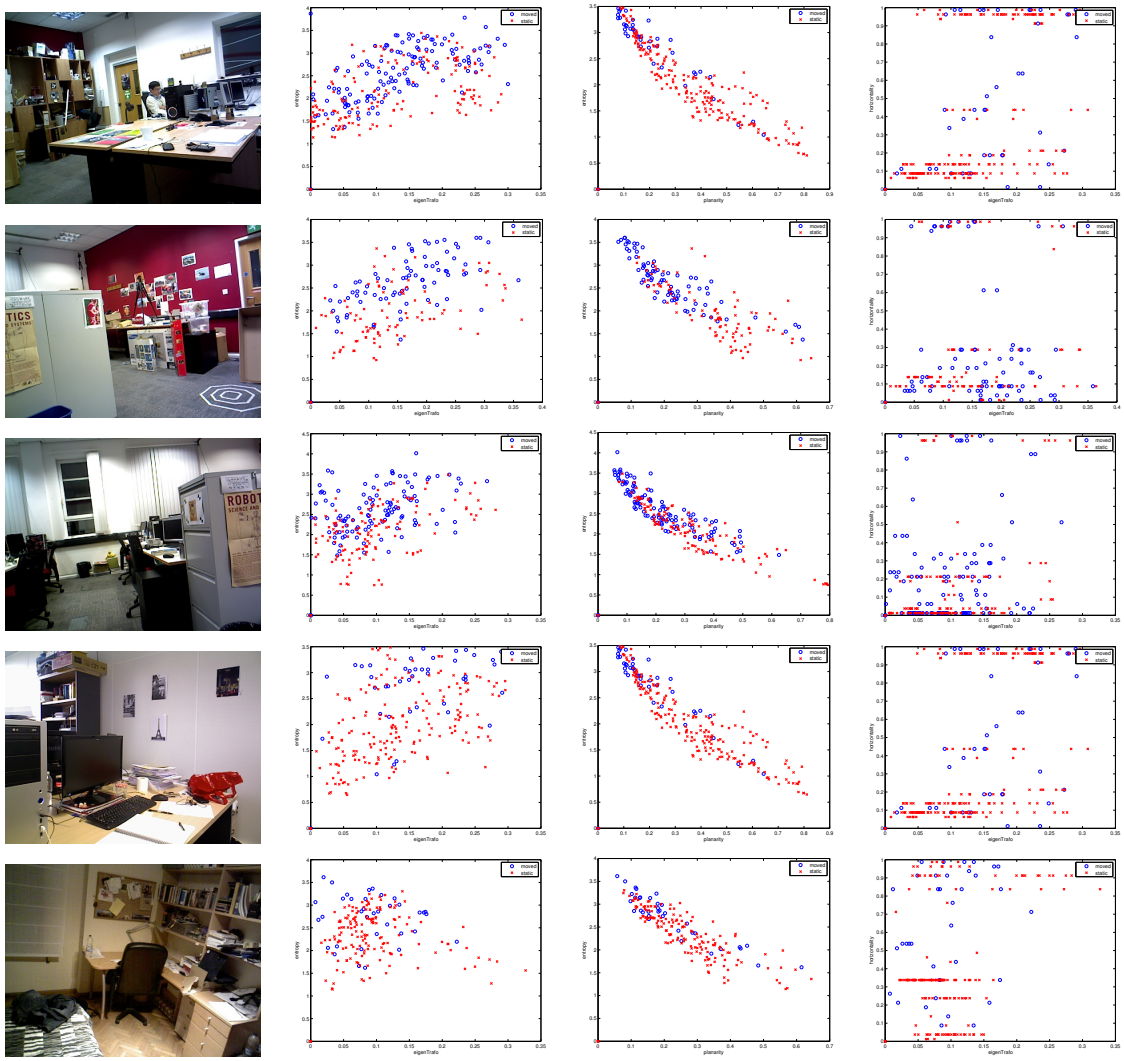
In some scenes however, distinction of moved from static areas based on the used features is less clear. It must be noted that in this experiment we lack from some accuracy due to the restriction of recording scenes from fixed poses. On the one hand the Ground Truth we take for moved and static elements is not perfect, due to residual misalignment of the images corresponding to the same poses and noisy depth measurements of far away areas of the scene. On the other hand, the segmentation on the image domain (Fig. 5) is not as good as computed on smooth 3D meshes as it is done in the next experiment (see $3th$ row of Fig. 7).

B. Improving place recognition on 3D meshes

For this experiment, we have selected 3 of the scenes captured in the previous experiment. These scenes have been selected due to their availability to record sequences on them. The idea is to prove that using information about 3D structure, based on the features proposed and analysed in previous experiment, to discard zones of the scene can benefit place recognition algorithms.

First, a point cloud and its corresponding triangular mesh are constructed for every scene for different days using the large scale KinectFusion implementation from the PCL library [15]. Then meshes are segmented using the algorithm of [10] and for every superjut, a histogram of normals is computed as explained in section III. Since entropy was shown to be quite discriminative in previous experiment, we use it as the only feature to prune moved parts of the scene. Each segment is scored then by computing its normalised neg-entropy:

$$negent(F) = 1 - \frac{ent(F)}{ent_{max}(F)} = 1 - \frac{ent(F)}{\log(M)} \quad (4)$$



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

Fig. 6. Distribution of static and moved superjuts in (a) some of the tested scenes, for (b) entropy-eigen transform, (c) entropy-planarity, (d) horizontality-eigen transform scores. Superjuts corresponding to moved/static areas are shown in blue/red.

To evaluate our approach we match keypoints between snapshots selected from the image sequences of the same scene at different days, using SURF descriptors. During mapping, the poses of the camera are saved so that the 3D point cloud can be projected and then obtain the scores and the segmentation in the corresponding 2D image domain (Fig. 7). The matching pipeline was set to take keypoints with a Hessian score greater than 200 and discard matches with a ratio greater than 0.85 for the distance between the two best matching hypotheses.

For a given image pair, a Ground Truth set of inliers I_{GT} is obtained by computing a robust fundamental matrix to select the geometrically consistent matches. On the other hand, the set of successful matches S_{th} for a given threshold th for the normalised neg-entropy score are those whose keypoints do not belong to any superjut with a score below the threshold. Given these definitions we can compute the precision and the recall for a given threshold

$$precision(th) = \frac{\text{size}(I_{GT} \cap S_{th})}{\text{size}(S_{th})}, \tag{5}$$

$$recall(th) = \frac{\text{size}(I_{GT} \cap S_{th})}{\text{size}(I_{GT})} \tag{6}$$

For every scene, we have computed the precision-recall curves on each of the possible combination of image pairs, by shifting the neg-entropy threshold th to prune likely-to-move areas. Fig. 8 show how the masking varies for different values of th. In Fig. 9 it is shown how the matching is affected only between image pairs corresponding to pairs of sequences acquired with 6 months of difference, for each of the 3 considered scenes. Note that for pairs of images where the zones of high entropy have been masked we obtain a higher ratio of inliers than matching the raw images. The neg-entropy threshold established for the masking is in each image pair, the one which produces the highest peak

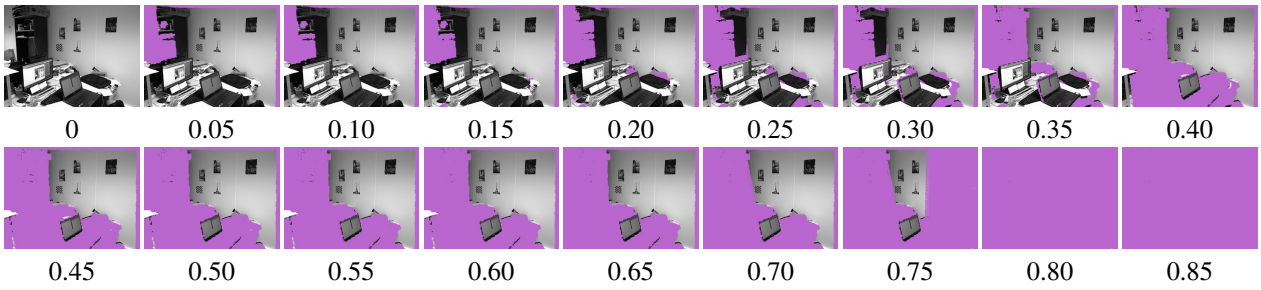


Fig. 8. Masks obtained in one frame of the $labDesk$ scene for different values of the score threshold th of the superjut's entropy of normals.

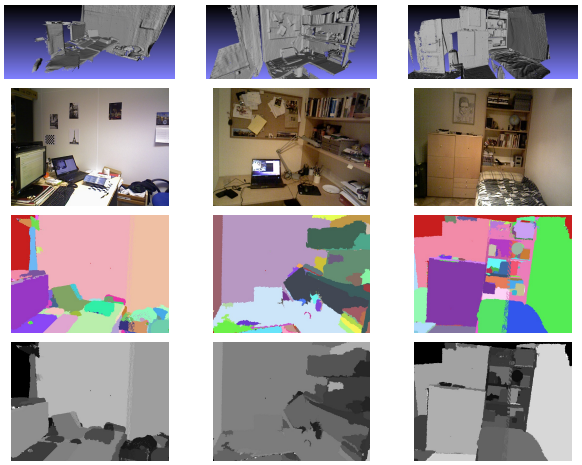


Fig. 7. Samples of data used in the experiments with depth map. $1st$ row shows the obtained 3D maps, $2nd$ row represents snapshots selected for matching, 3rd row is the segmentation of the map of normals computed on the 3D mesh and 4th row shows the normalised neg-entropy for every superjut projected on the image.

in the precision-recall curve. It can be observed that the zones that remain unmasked correspond in their majority to planar zones like walls, which contain very stable features. On the other hand masking can also eliminate zones which remain static between acqusitions, such as the selves in the $roomBed$ scenario. Also planar objects which are likely to change their position, such as laptop screens can remain unmasked. Howerver, in spite of discarding some stable zones and accepting unstable ones for matching, the precission and recall curves indicate that the effect of that masking is beneficial for some neg-entropy threshold.

In Fig. 10, we show the average precision and recall curves for each scene, including all the possible combinations of image pairs from all the sequences acquired for the 3 scenarios. Note that for the first scene ($labDesk$) we obtain an increase in precision of more than 75% with a decrease in recall of only 20% with respect not using any masking. This is caused by the presence of textured elements on walls which leads to a great number of matches in a low entropy zones. However, in the third scene ($roomBed$), the lack of highly textured low-entropy areas makes the beneficial effect less noticeable.

Note that though being evaluated in an image matching context, our approach computes the masks prior to the matching process. Considering this fact and also that a greater inliers rate between two matched images is caused by a greater appearance similarity between both images, our approach is likely to be applied within the framework of efficient image search algorithms [5] increasing their performance. Though the computational cost of performing the mesh segmentation and superjut projection and scoring is relatively high, around the order of 5 seconds, it must be noted that every time a new keyframe is tested for place recognition the cost of our approach is constant and does not grow linearly with the number of frames as it surely would a computationally costly exhaustive matching and geometric verification scheme over all the keyframe search space.

V. Conclusions

This work is concerned with the problem of long-term indoor mapping with RGB-D cameras. Concretely we focused on finding and evaluating properties of the elements of the scene which can allow to discard parts of the scene which easily change with time and could affect severally to place recognition algorithms for example when relocalising or detecting loop closures. In the experiments we have shown that parts of the scene which move along time tend to present a more chaotic structure, which is reflected in a high entropy of the histogram of normals. We have validated this observation, by matching images of the 3D map built on different days, with a maximum difference between acquisitions of 6 months apart, for 3 selected scenes. Though the validation was performed in a standard matching scheme, the results are promising and can be extended to place recognition approaches, were similarity between images to make a quick decision has to be measured in first place without considering a matching and geometric verification scheme with high computational complexity with grow linear with the number of keyframes in the search database.

References

[1] P. Boucher. Points on a sphere. http://www.softimageblog.com/archives/115, 2006.
[2] C. Cadena and J. Kosecka. Semantic parsing for priming object detection in rgb-d scenes. In *Semantic Perception Mapping and Exploration (SPME) Workshop, held with ICRA*, 2013.
[3] J. Civera, O. G. Grasa, A. J. Davison, and J. M. M. Montiel. 1-Point RANSAC for EKF Filtering: application to real-time structure from motion and visual odometry. *J. of Field Robotics*, 27(5):609–631, 2010.
[4] C. Couprie, C. Farabet, L. Najman, and Y. LeCun. Indoor semantic segmentation using depth information. *CoRR*, abs/1301.3572, 2013.

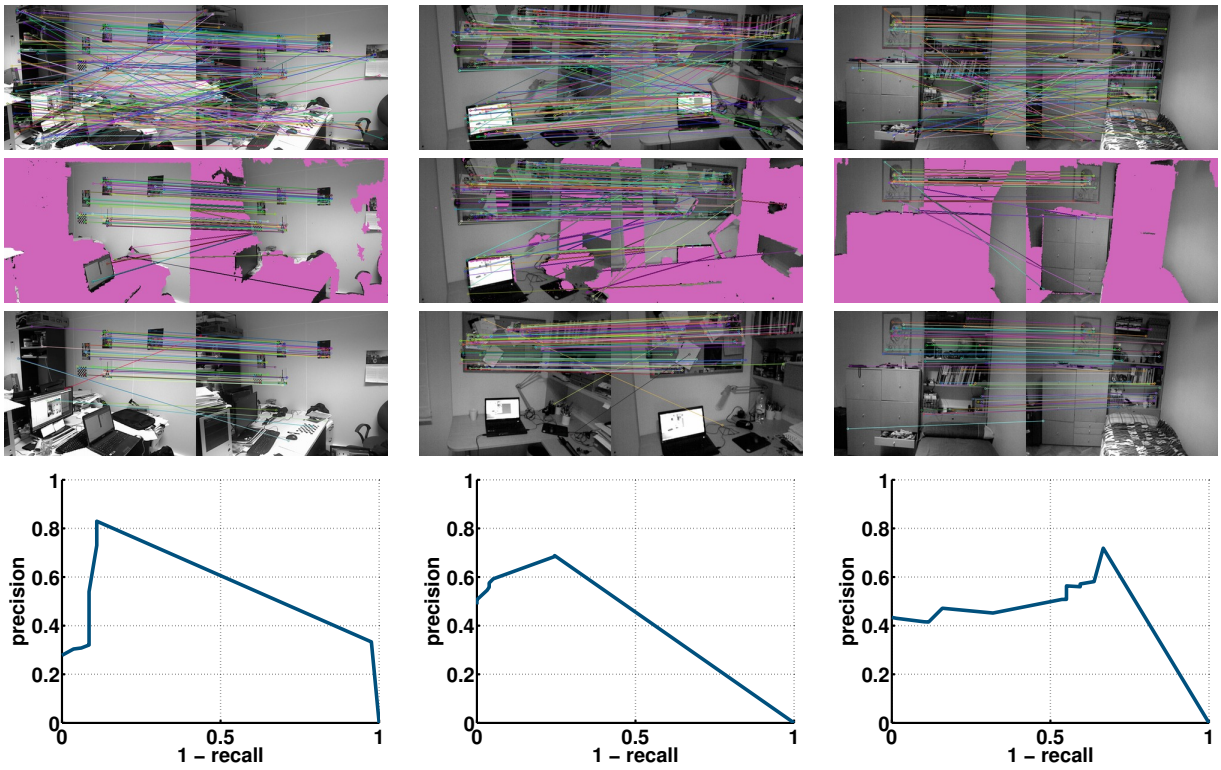


Fig. 9. Evaluation of our method in 3 different scenes taken with 6 months of difference in an uncontrolled area (from left to right: $labDesk$, $roomDesk$ and $roomBed$). (1st row) Unmasked raw matches, (2nd row) matches after masking for the score corresponding to the highest peak in the precision-recall curves in 4th row (masked areas are in magenta), (3rd row) Ground Truth matches by geometric consistency.

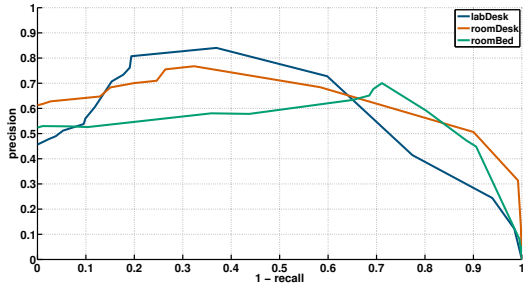


Fig. 10. Average precision-recall curves for the 3 considered scenes

[5] M. Cummins and P. Newman. FAB-MAP: Probabilistic Localization and Mapping in the Space of Appearance. *The International Journal of Robotics Research*, 27(6):647–665, 2008.

[6] P. F. Felzenszwalb and D. P. Huttenlocher. Efficient graph-based image segmentation. *Int. J. of Computer Vision (IJCV)*, 59(2):167–181, 2004.

[7] R. Finman, T. Whelan, M. Kaess, and J. J. Leonard. Toward lifelong object segmentation from change detection in dense rgb-d maps. In *European Conf. on Mobile Robotics (ECMR)*, pages 178–185, 2013.

[8] A. P. Gee and W. W. Mayol-Cuevas. 6d relocalisation for rgbd cameras using synthetic view regression. In *British Machine Vision Conf. (BMVC)*, pages 1–11, 2012.

[9] E. Herbst, P. Henry, X. Ren, and D. Fox. Toward object discovery and modeling via 3-d scene comparison. In *Int. Conf. on Robotics and Automation (ICRA)*, pages 2623–2629, 2011.

[10] A. Karpathy, S. Miller, and L. Fei-Fei. Object discovery in 3d scenes via shape analysis. In *Int. Conf. on Robotics and Automation (ICRA)*, pages 2088–2095, 2013.

[11] C. Kerl, J. Sturm, and D. Cremers. Robust odometry estimation for rgb-d cameras. In *Int. Conf. on Robotics and Automation (ICRA)*, pages 3748–3754, 2013.

[12] K. Konolige and J. Bowman. Towards lifelong visual maps. In *Proc. IEEE/RSJ Int. Conf. on Intelligent Robots and Systems (IROS)*, pages 1156–1163, 2009.

[13] R. A. Newcombe, S. Izadi, O. Hilliges, D. Molyneaux, D. Kim, A. J. Davison, P. Kohli, J. Shotton, S. Hodges, and A. W. Fitzgibbon. Kinectfusion: Real-time dense surface mapping and tracking. In *Int. Symp. on Mixed and Augmented Reality (ISMAR)*, pages 127–136, 2011.

[14] X. Ren, L. Bo, and D. Fox. Rgb-(d) scene labeling: Features and algorithms. In *IEEE Conf. on Computer Vision and Pattern Recognition (CVPR)*, pages 2759–2766, 2012.

[15] R. B. Rusu and S. Cousins. 3d is here: Point cloud library (pcl). In *IEEE Int. Conf. on Robotics and Automation (ICRA)*, pages 1–4, 2011.

[16] N. Silberman, D. Hoiem, P. Kohli, and R. Fergus. Indoor segmentation and support inference from rgbd images. In *European Conf. on Computer Vision (ECCV)*, pages 746–760, 2012.

[17] S. Tang, X. Wang, X. Lv, T. X. Han, J. M. Keller, Z. He, M. Skubic, and S. Lao. Histogram of oriented normal vectors for object recognition with a depth sensor. In *Assian Conf. on Computer Vision (ACCV)*, pages 525–538, 2012.

[18] A. T. Targhi, E. Hayman, J.-O. Eklundh, and M. Shahshahani. The eigen-transform and applications. In *Assian Conf. on Computer Vision (ACCV)*, pages 70–79, 2006.

[19] R. Triebel, J. Shin, and R. Siegwart. Segmentation and unsupervised part-based discovery of repetitive objects. In *Robotics: Science and Systems (RSS)*, 2010.

[20] I. Ulrich and I. R. Nourbakhsh. Appearance-based place recognition for topological localization. In *Int. Conf. on Robotics and Automation (ICRA)*, pages 1023–1029, 2000.

[21] T. Whelan, M. Kaess, J. J. Leonard, and J. B. McDonald. Deformation-based loop closure for large scale dense RGB-D SLAM. In *IEEE/RSJ Intl. Conf. on Intelligent Robots and Systems (IROS)*, 2013.

[22] B. P. Williams, G. Klein, and I. Reid. Automatic relocalization and loop closing for real-time monocular slam. *IEEE Trans. Pattern Analysis and Machine Intelligence (PAMI)*, 33(9):1699–1712, 2011.