Autonomous Robots (2021) 45:407–420
https://doi.org/10.1007/s10514-021-09973-w

Scale-invariant localization using quasi-semantic object landmarks

Andrew Holliday[1] · Gregory Dudek[2]

Received: 21 January 2020 / Accepted: 23 January 2021 / Published online: 25 February 2021

Abstract

This work presents **Object Landmarks**, a new type of visual feature designed for visual localization over major changes in distance and scale. An **Object Landmark** consists of a bounding box **b** defining an object, a descriptor **q** of that object produced by a Convolutional Neural Network, and a set of classical point features within **b**. We evaluate **Object Landmarks** on visual odometry and place-recognition tasks, and compare them against several modern approaches. We find that **Object Landmarks** enable superior localization over major scale changes, reducing error by as much as 18% and increasing robustness to failure by as much as 80% versus the state-of-the-art. They allow localization under scale change factors up to 6, where state-of-the-art approaches break down at factors of 3 or more.

Keywords Visual features · Visual odometry · Place recognition · Robotic localization

1 Introduction

Visual localization is an important capability in mobile robotics. In order for a robot to operate "in the wild" in unstructured environments, it has to be able to perceive and understand its surroundings. It must recognize previously-visited locations under new conditions and perspectives, and estimate its own position in the world. Vision sensing is a modality well-suited to this task due to its low power requirements and richness of information. Much research (Brown and Lowe 2002; Bay et al. 2008; Simo-Serra et al. 2015; Yi et al. 2016 among others) has looked at how to make visual localization and place recognition more robust to changes in viewing angle and appearance, such as under variations in lighting or weather. But comparably little attention has been given to the problem of localizing under large differences in scale, when a scene is viewed from two (or more) very different distances.

This problem can arise in numerous cases. One is that of repeated missions carried out by aquatic robots over a coral reef. A high-altitude robot might build a map of a reef, which is then used by a low-altitude robot on a subsequent mission to navigate in the reef. Another case is indoor navigation, in which a robot may first see a location from far away, and then later visit that location, viewing it much more closely, without having moved directly between those two views. The robot must recognize that these two disparate viewpoints contain the same scene, and determine the implied spatial relationship between the two views.

As we will show in Sect. 4, state-of-the-art techniques such as the Scale-Invariant Feature Transform (SIFT) have poor robustness to scale changes greater than about 3×. By contrast, humans can recognize known landmarks and accurately estimate their own positions over a very wide range of visual scales, and may possess this ability from as early as three years of age (Spencer and Darvizeh 1981). This work builds on the hypothesis that a key to human navigation over large scale changes is the use of semantically-meaningful objects as landmarks (Fig. 1).

In our prior work (Holliday and Dudek 2018), we proposed an image feature we refer to as an **Object Landmark**. **Object Landmarks** combine learned object features like those used in Sünderhauf et al. (2015) with more traditional point features. They are composed of "off-the-shelf" compo-

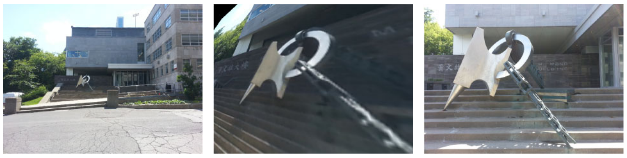
Supplementary Information The online version contains supplementary material available at https://doi.org/10.1007/s10514-021-09973-w.

✉ Andrew Holliday
ahollid@cim.mcgill.ca

Gregory Dudek
greg.dudek@samsung.com

[1] McGill University Center for Intelligent Machines, 3480 University Street, McConnell Engineering Building, Room 410, Montréal, QC H3A 0E9, Canada

[2] Samsung AI Center, 1250 René-Lévesque, 37th floor, Montréal, Canada



Far Image

Far mapped to near

Near Image

Fig. 1 A homographic mapping between two images computed using **Object Landmarks**, as described in Sect. 3.2. Despite a 6× difference in visual scale, the highly-distinctive foreground object allows our system to determine an accurate homography between the images. See also Fig. 8

nents and require no environment-specific pre-training. We demonstrated that by using matches between object features to guide the search for point feature matches, accurate metric pose estimation could be achieved on image pairs exhibiting major scale changes.

The present work builds on Holliday and Dudek (2018) in the following ways:

- We refine **Object Landmarks** by substituting new object-proposal and object-description components that improve speed and accuracy, and present new results using these enhancements.
- We evaluate two recent learned feature point extractors from the literature, Learned Invariant Feature Transform (LIFT) and D2Net, and compare their performance with **Object Landmarks**.
- We show that **Object Landmarks** can be used for long-range place recognition, and report new results showing that they improve on the state-of-the-art.

2 Background and related work

2.1 Classical visual features

Most classical methods for producing descriptors of image content are based on image gradients and responses to engineered filter functions. Most can be categorized as either point-feature methods, where a set of keypoints in an image are detected with associated descriptor vectors, or whole-image methods, where a single vector is computed to describe the entire image.

One notable point-feature method is SIFT, originally proposed by Lowe (1999). As the name implies, SIFT features are robust to some visual scale change. Their extraction is based on the principles of scale-space theory (Lindeberg 1994). Keypoints are taken as the extrema of difference-of-Gaussian functions applied at various scales to the image, and descriptors are computed from image patches rotated and scaled accordingly. Other widely-used point feature types include Speeded-up Robust Features (SURF) (Bay et al. 2008) and Oriented FAST and rotated BRIEF (ORB) (Rublee et al. 2011). Point features have the advantage of retaining explicit geometric information about an image.

Whole-image methods include GIST (Oliva and Torralba 2001) and bag-of-words approaches, among others. GIST describes an image based on its responses to a variety of Gabor-wavelet filters. In bag-of-words, the vector space of the descriptors of some point feature type, such as SIFT, is discretized to form a dictionary of “visual words”. One then extracts point features from an image, finds the nearest word in the dictionary to each feature, and computes a histogram of word frequencies to serve as a whole-image descriptor.

Classical visual feature methods like these tend to break down under large changes in perspective and appearance, since different perspectives on a scene can produce very different patterns of gradients in the image. **Object Landmarks** address this weakness by basing descriptors of image components on learned high-level abstractions.

A third category of classical methods partition an image into components that can be interpreted as objects. Uijlings et al. (2013) propose Selective Search, which hierarchically groups pixels based on their low-level similarities. Zitnick and Dollar (2014) propose Edge Boxes, which detects edges in an image and outputs boxes that tightly enclose many edges. Both approaches propose image regions to treat as objects, but do not provide descriptors for those regions. **Object Landmarks** make use of Edge Box object proposals as the first step in the process, and build representations of these proposed objects that can be used for visual localization.

2.2 Visual localization

As described in Dudek and Jenkin (2010), robotic localization can broadly be divided into two problems. In “local localization”, some prior on the robot’s pose estimate is given, while in global localization, the robot’s position must be estimated without any prior. Early work on these problems was carried out by Leonard and Durrant-Whyte (1991), MacKenzie and Dudek (1994), and Fox et al. (1999), among others. Most contemporary visual localization approaches are based on classical low-level methods of image description, and they suffer from the deficiencies of these methods described in Sect. 2.1.

Visual place recognition is a case of global localization. In this problem, the environment is modeled as a set of previously-observed scenes. New observations are determined either to match some previous scene, or to show a new scene. Fast Appearance-Based Mapping, or FAB-MAP, is a classic method proposed by Cummins and Newman (2010). It uses a bag-of-words scene representation, building a Bayesian model of visual-word occurrence in scenes. It performs well only under very small changes in perspective between the query and its nearest match. More recent systems, such as the Convolutional Autoencoder for Loop Closure (CALC) of Merrill and Huang (2018) and Region-VLAD of Khaliq et al. (2019), as well as the work of Chen et al. (2018) and Garg et al. (2018), have used Convolutional Neural Networks (CNNs) to improve on FAB-MAP's performance under appearance and perpsective change.

Visual odometry is a case of "local localization" in which a robot attempts to estimate its trajectory from two or more images captured by its camera(s). Given a pair of images and certain other data, points or regions in one image can be matched to the other, and the matches can be used to triangulate the relative camera poses. Most approaches to date have established matches either using point-feature matching, as in PTAM by Klein and Murray (2007) and ORB-SLAM by Mur-Artal et al. (2015), or direct image alignment of image intensities, as in LSD-SLAM by Engel et al. (2014). Both such approaches are quite limited in their robustness to changes in scale and perspective. Our proposed **Object Landmark** feature is designed to provide much greater robustness to scale change.

2.3 Learning visual features and localization

Much work has explored ways of learning to extract useful localization features from images. Linegar et al. (2016) and Li et al. (2015) propose similar approaches: both identify distinctive image patches as "landmarks" over a traversal of an environment, and train Support Vector Machines (SVMs) to recognize each patch.

The notion of using semantically-based observables for mapping or navigation can be traced to Kriegman et al. (Dec 1989), and there are many examples of its use in robotics, such as the Simultaneous Localization And Mapping (SLAM) system SLAM++ of Salas-Moreno et al. (2013), and the work of Galindo et al. (2005). Bowman et al. (2017) train Deformable Parts Models to detect several known object types, such as doorways and office chairs, and use these to propose semantic landmarks for use in SLAM, demonstrating state-of-the-art performance on small office environments. More recently, Li et al. (2019) use semantic landmarks to localize over image pairs with very wide baselines, focusing on changes in perspective (front vs. side or rear views) rather than changes in scale. All of these systems rely on some form of pre-training, either on the environment in question or on a known class of objects.

Kaeli et al. (2014) use the distinctiveness of low-level feature histograms to cluster images from a robotic traversal of an environment, and their system requires no pre-training. However, they explicitly limit observed scale changes to 2.67×, ignoring images outside that range.

Other approaches make use of the semantic capabilities of Deep Neural Networks (DNNs) to provide visual descriptors. Sünderhauf et al. (2015) use the intermediate activations of an ImageNet-trained CNN (Russakovsky et al. 2015) as whole-image descriptors to perform place recognition. In their follow-up work (Sünderhauf et al. 2015), they use the same CNN feature extractor on image patches proposed by an object detector. They define a similarity score based on matching objects between images. Using this for place recognition, they report better precision and recall than FAB-MAP.

Simo-Serra et al. (2015) train a CNN to produce local descriptors for keypoints from 64 × 64-pixel patches around those keypoints. By detecting SIFT keypoints and using their CNN to generate the descriptors, they show improved point-matching accuracy vs. plain SIFT under large rotations, lateral translations, and appearance changes. The LIFT of Yi et al. (2016), SuperPoint of DeTone et al. (2018), and D2Net of Dusmanu et al. (2019) all train CNNs that both detect keypoints and compute descriptors, and demonstrate further improvements over SIFT and Simo-Serra et al. (2015)'s method. None of this work reports experiments involving large scale changes, however.

Learning systems have also been trained to localize directly from images. One of the earliest efforts was that of Dudek and Zhang (1995, 1996) wherein a neural network was used to map edge statistics directly to 3D pose in a small indoor environment. Kendall and Cipolla (2016) train a CNN to predict camera pose from an image of an outdoor environment. Mirowski et al. (2018) train a neural-network system to navigate through the graph of a city extracted from Google Street View. Because this system operates only on the limited set of viewpoints in the Street View graph, it is not suitable for real-world deployment. Both this system and PoseNet require a neural network to be trained from scratch for each new environment in which they operate, making deployment costly.

In all of this work, the proposed systems are either unsuited by design to localizing across large scale changes, or their evaluation does not include large scale changes, making their performance in such cases unknown.

3 Proposed system

The core idea of this work is that objects can be used as landmarks for coarse localization under major scale changes, because their semantics are robust to such changes. But more precise localization between viewpoints requires consideration not just of the object, but of its parts. If enough points on an object's surface in one view can be matched to points in the other, that one object can be enough to precisely localize a robot from a new viewpoint, even over significant scale change (Fig. 2).

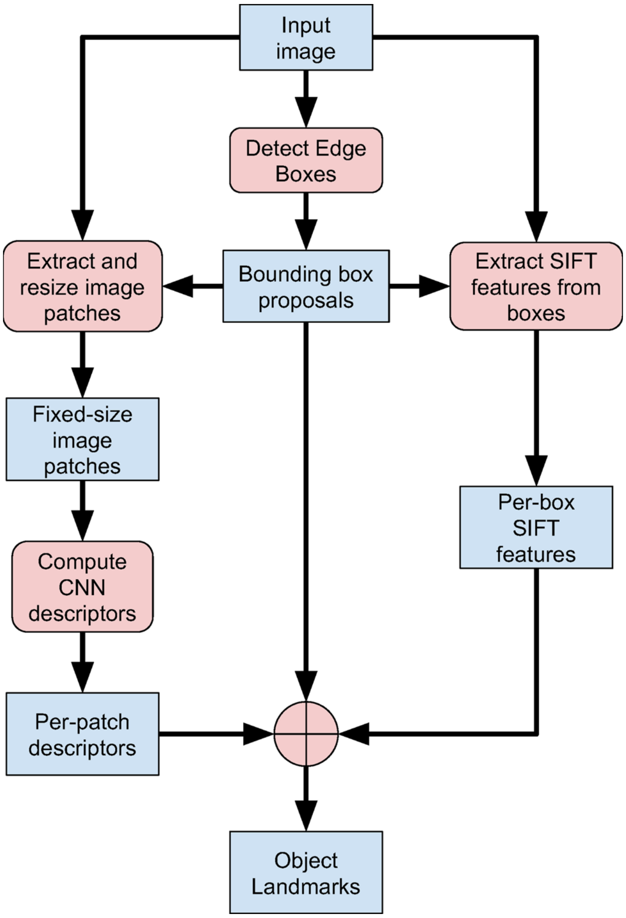


Fig. 2 A schematic of the **Object Landmarks** extraction process. Blue boxes represent data, red boxes are operations. Bounding boxes **b** for objects are detected. The image patches enclosed by these boxes are passed through a CNN to get a descriptor **q** for each object, and a set P of SIFT features are also extracted from each patch. The **Object Landmark** feature consists of **b**, **q**, and P taken together

3.1 Object landmarks

We define an **Object Landmark** as a triplet, $\mathbf{o} = (\mathbf{b}, \mathbf{q}, P)$ where:

- $\mathbf{b} = \langle x_{\text{left}}, y_{\text{top}}, w, h\rangle$ is a 2D bounding box defining the object's location in the image,
- $\mathbf{q}$ is a quasi-semantic descriptor vector of dimension k_q,
- P is a set of point features inside **b**. Each point feature $p = (l, \mathbf{v})$ consists of a location $\mathbf{l} = \langle x, y\rangle$ and a descriptor vector **v** of dimension k_v.

The **Object Landmark** extraction pipeline is as follows: first, object-proposal bounding boxes are computed from an image. The rectangular image patch corresponding to each box **b** is extracted and resized to a fixed-size square. This distorts the contents, but helps normalize for perspective change. Each resized patch is passed through a CNN trained for image classification on ImageNet, and the activation of an intermediate layer (flattened to a vector) is taken as the descriptor **q** of the object. We refer to **q** as a "quasi-semantic" descriptor, because it is a highly abstract representation of the image used to construct the CNN's semantic output, but lacks its own semantic interpretation. We use ImageNet-trained CNN features because they have been demonstrated to be useful for localization tasks in works such as Sünderhauf et al. (2015, ?).

Point features are also extracted from the original image. The set of point features inside an **Object Landmark**'s **b** are designated the sub-features of the landmark, P.

3.1.1 Object proposals

In general, the more unique an object is, the better it will serve as a localization landmark - but the less likely it is to belong to any object class known in advance. For this reason, we prefer an object-detection mechanism that is class-agnostic and based on low-level image features, rather than a DNN-based detection method that may be biased by the contents of its training data. Selective Search (Uijlings et al. 2013) was shown to be effective for robust scale-invariant localization in Holliday and Dudek (2018), but in this work, we use Edge Boxes (Zitnick and Dollar 2014), which we found to be faster and more accurate.

3.1.2 Object descriptors

Our prior work (Holliday and Dudek 2018) employed a ResNet CNN (He et al. 2015) to extract **q**. We reported results for a range of input sizes and network layers on a held-out dataset. The best-performing descriptors ranged from $k_q \approx$ 32k to $k_q \approx$ 100k. For the large-scale place-recognition experiments in this work, we sought to reduce k_q, both to speed up the computation of distances between different **q**s and to limit our memory footprint. Our final choice and rationale are presented in Sect. 4.1.2.

3.1.3 Sub-features

We use SIFT for the landmark sub-features in our experiments, because of their well-demonstrated robustness to scale changes. We use the recommended SIFT configuration proposed in Lowe (1999): 3 octave layers, $\sigma = 1.6$, contrast threshold 0.04, and edge threshold 10. As the bounding boxes of object proposals can overlap one another, we allow one point feature to be associated with multiple **Object Landmarks**.

3.2 Transform estimation

Transform estimation is a limiting case of visual odometry with just two frames. Given a pair of images I_1 and I_2 of a scene, we use **Object Landmarks** to estimate the transform between the camera poses as follows. After extracting landmarks from both images, for every landmark $i \in I_1$ and every landmark $j \in I_2$, we compute the cosine distance between their $\mathbf{q}$s:

$$d_{ij} = \text{dcos}(\mathbf{q}_i, \mathbf{q}_j) = 1 - \frac{\mathbf{q}_i \cdot \mathbf{q}_j}{\|\mathbf{q}_i\| \, \|\mathbf{q}_j\|} \tag{1}$$

The matches are all pairs of landmarks (i, j) for which:

$$j = \arg\min_{j'} d_{ij'} \tag{2}$$

$$i = \arg\min_{i'} d_{i'j} \tag{3}$$

Once these matches are found, we match the sub-features P_i, P_j of each match (i, j) in the same way, but using Euclidean distance between SIFT $\mathbf{v}$s instead of cosine distance. This produces a set of point matches of the form $(\mathbf{l}^1_{i,m}, \mathbf{l}^2_{j,n})$ between sub-feature $m \in P_i$ and $n \in P_j$. If no sub-feature matches can be found for (i, j), the pair produces a single point match between the centroids of $\mathbf{b}_i$ and $\mathbf{b}_j$. The point matches are then used to estimate either an essential matrix E or a homography matrix H that relates the two images. A homography H projects points from one 2D space to another, and is used when we know that the scene being viewed is roughly planar. Given four point matches, H is computed via least-squares so as to minimize their reprojection error. If the scene is not expected to be planar, we instead compute E via the five-point algorithm of Nistér (2004). Faugeras et al. (2001) describes the meaning of homographies and essential matrices.

In either case, a Random Sample Consensus (RANSAC) process is used that estimates matrices from many different subsets of the point matches, and returns the matrix that is consistent with the largest number of them.

Once H or E has been calculated, a set of possible transforms consistent with the matrix can be derived. Cheirality checking (Hartley 1993) is performed on each transform to eliminate those that would place any matched points behind either camera, leaving a single transform.

3.3 Place recognition

To perform place recognition using **Object Landmarks**, we propose a modification of the technique of Sünderhauf et al. (2015). This technique consists of matching the **Object Landmarks** in a query image against those of each candidate in the map set, and computing a similarity score for each candidate:

$$S_{q,c} = \frac{1}{\sqrt{n_q n_c}} \sum_{(i,j)} 1 - (d_{ij} s_{ij}) \tag{4}$$

where q is the query, c is a candidate, n_q and n_c are the number of **Object Landmarks** detected in q and c, and s_{ij} is a shape similarity score for $\mathbf{b}_i$ and $\mathbf{b}_j$. The match to the query is $\arg\max_c S_{q,c}$. In Sünderhauf et al. (2015), s_{ij} was defined to reflect the difference in the size of the two boxes. Under large scale changes, landmarks corresponding to the same world object will have different sizes, so we instead base s_{ij} on the difference in aspect ratios:

$$s_{ij} = \exp \frac{\left| \frac{w_i}{h_i} - \frac{w_j}{h_j} \right|}{\max(\frac{w_i}{h_i}, \frac{w_j}{h_j})} \tag{5}$$

The other significant modification we make is the addition of a geometric-consistency check. This is done by attempting transform estimation between q and c as described in Sect. 3.2: if this fails to estimate a valid transform, the candidate is rejected, and the candidate c' with the next-highest $S_{q,c'}$ is considered, and so on until a consistent match is found or all possibilities are exhausted. The system reports the matched candidate if one was found, as well as the estimated transform.

4 Transform estimation experiments

In this section, we evaluate our transform estimation approach against two datasets: the KITTI urban-driving dataset, and the Montreal scale-change dataset published in Holliday and Dudek (2018). The KITTI dataset consists of stereo image pairs with precise ground truth poses, allowing a strict metric evaluation of accuracy, but also contains considerable variation in viewing angle. The Montreal scale-change dataset varies much less in viewing angle, allowing an analysis more closely focused on scale, and its scene contents are very different from KITTI's, captured by a pedestrian in a dense urban environment.

4.1 KITTI odometry

The KITTI Odometry dataset (Geiger et al. 2012) consists of data gathered during 22 traversals of urban and suburban environments in the city of Karlsruhe, Germany, by a car equipped with a sensor package. The data includes stereo colour image pairs captured at a rapid rate, and the first eleven traversals have ground-truth poses for every stereo pair. We sample every fifth frame from the first eleven traversals to reduce the scope of our experiments. For each frame in a subsampled traversal, we estimate the transforms between that frame and the next ten frames using the images from the left-hand colour camera. Frame pairs in which the true angle between gaze directions is greater than the camera's FOV are discarded, as such pairs rarely share any content.

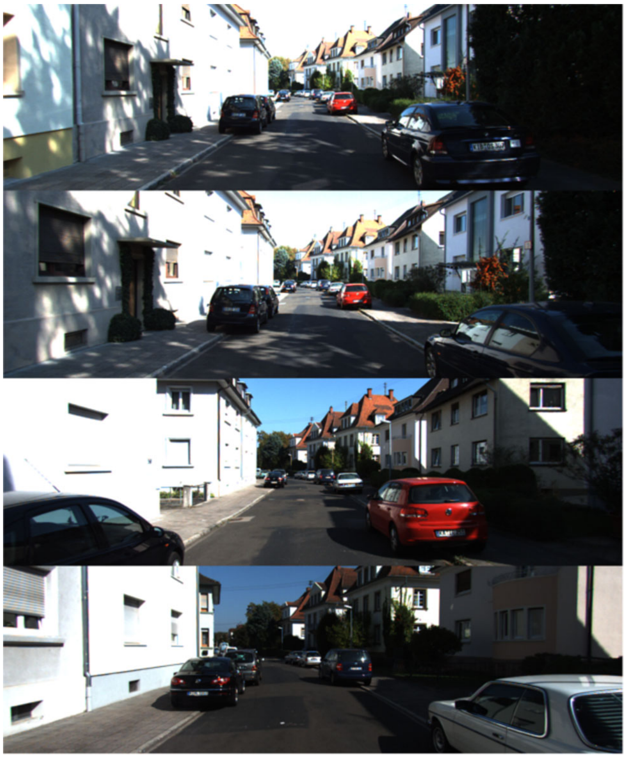


Fig. 3 A set of images from one of our the subsampled KITTI odometry traversals. The second, third, and last images are from one, five, and ten frames after the first. This gives a sense of the range of visual scale changes that are present in our KITTI evaluation

The resulting frame pairs cover a wide range of scale changes and scene types (Fig. 3).

When estimating transforms from monocular image pairs, the translation's magnitude cannot be determined. But the stereo image pairs of each KITTI frame allow us to estimate the magnitude. A disparity map D is computed for images I_{left} and I_{right} via the block-matching algorithm of Konolige (1998), such that $D_{x,y}$ is the distance (in pixel space) between the pixel at location $\langle x, y \rangle$ in I_{left} and its match in I_{right}. An unscaled 3D world point $\mathbf{L}' = \langle x', y', z' \rangle$ returned by our transform-estimation algorithm is related to the true world point $\mathbf{L}$ by a scale factor: $\mathbf{L} = \mathbf{L}'s$. Given the corresponding pixel location $\mathbf{l}_{\text{left}} = \langle x, y \rangle$ of $\mathbf{L}'$ in I_{left}, the baseline b between the left and right cameras in meters, and the focal length f of the left camera in pixels, we can compute s:

$$s = \frac{bf}{z' D_{x,y}} \tag{6}$$

In principle, s is the same for any $\mathbf{L}'$ and $\mathbf{L}$, and relates the estimated unitless translation $\mathbf{t}'_{\text{est}}$ to the scaled translation $\mathbf{t}_{\text{est}}$: $\mathbf{t}_{\text{est}} = \mathbf{t}'_{\text{est}}s$. We estimate s separately using $\mathbf{l}^{1,\text{left}}_{i,m}$ for each point match $(\mathbf{l}^{1,\text{left}}_{i,m}, \mathbf{l}^{2,\text{left}}_{j,n})$ produced by the transform-estimation technique in question, and we compute $\mathbf{t}_{\text{est}}$ as:

$$\mathbf{t}_{\text{est}} = \mathbf{t}'_{\text{est}} \times \text{median}_k(s_k) \tag{7}$$

4.1.1 Error metrics

We examine two metrics of error in these experiments. The first is the error in the estimated pose between the two cameras 1 and 2, which we call the pose error:

$$t_{\text{err}} = ||{}^2_1\mathbf{t}_{\text{est}} - {}^2_1\mathbf{t}_{\text{gt}}||_2 \tag{8}$$

$$r_{\text{err}} = \text{dcos}({}^1_2\mathbf{r}_{\text{est}}, {}^1_2\mathbf{r}_{\text{gt}}) \tag{9}$$

$$\text{pose error} = t_{\text{err}} + r_{\text{err}} * ||{}^2_1\mathbf{t}_{\text{gt}}||_2 \tag{10}$$

where ${}^i_j\mathbf{t}_{\text{est}}$, ${}^i_j\mathbf{t}_{\text{gt}}$, and ${}^i_j\mathbf{r}_{\text{est}}$, ${}^i_j\mathbf{r}_{\text{gt}}$ are the estimated and true positions and orientation quaternions of camera j in camera i's frame of reference. r_{err} is unitless and ranges from 0 to 2, while t_{err} has units of meters. To compose a single error metric, we scale r_{err} by the length of the ground-truth translation. Our rationale is that an error in estimated orientation at the beginning of a motion will contribute to an error in estimated translation at the end of the motion proportional to the real distance travelled.

The second error metric is the localization failure rate. Localization failure means that no transform could be estimated for an image pair. This can occur when too few point matches are discovered to estimate E or H, or when none of the candidate transforms pass the cheirality check. Such failure can be catastrophic in applications such as SLAM. Depending on the robot's sensors, no recovery may be possible, so failure rate is a very important metric.

4.1.2 Parameter search

Before evaluating our method, we performed a parameter-tuning phase on a "tuning set" made from traversals 01, 06, and 07. This contained 4,055 frame pairs, about 11% of the whole pose-annotated KITTI odometry dataset. The first part was a search over network architectures and layers for a suitable extractor for $\mathbf{q}$. We considered four families of networks: AlexNet (Krizhevsky et al. 2012), VGGNet (Simonyan and Zisserman 2014), ResNet (He et al. 2015), and DenseNet (Huang et al. 2017), testing several variants of each. To constrain k_q and allow fast $\mathbf{q}$ matching, inputs to the networks were resized to 64×64 pixels, and we did not consider any network layers with $k_q > 2^{16}$. For all experiments, we used the pre-trained ImageNet weights available through the PyTorch deep learning framework (Paszke et al. 2017). The full details of this architecture search and analysis of its results were presented in Holliday and Dudek (2020), but are omitted here for brevity.

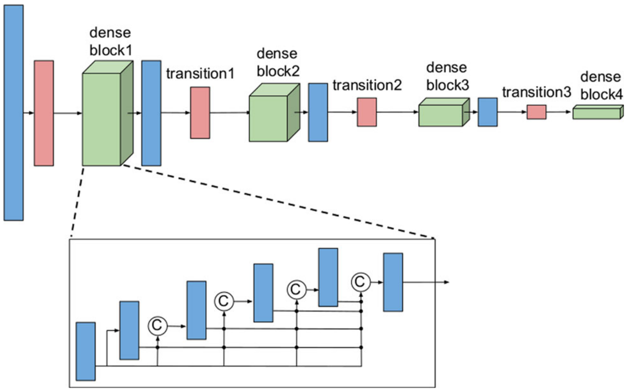


Fig. 4 A schematic illustration of a DenseNet. The green block represents a composite dense-block. The contents of the first denseblock are displayed in the insert; the circled 'C' means concatenation of all inputs along the channel dimension. This basic structure, in which each layer's input is the concatenated outputs of every previous layer, is common to all denseblocks, but they differ in their number of layers. Only convolutional (blue) and pooling (red) layers are shown

We found that most layers provided comparable accuracy, but varied widely in k_q. DenseNets produced much smaller k_q than other networks, while being among the most accurate. We settled on an intermediate layer of a DenseNet with 169 total layers, specifically the pooling layer labelled "transition3" in Fig. 4, as the **q** extractor for all subsequent experiments with **Object Landmarks** and object features. It has $k_q = 2560$. Its pose error on the tuning set was only 5% higher than the lowest-error layer, and its k_q was 40% smaller than the smallest layer with lower error.

We then ran a grid search over the parameters α and β of Edge Boxes (Zitnick and Dollar 2014), as well as the number of object proposals to use, n_p, and an upper limit on aspect ratio $r_{\max}$, where the aspect ratio is defined as $r = \max(\frac{h}{w}, \frac{w}{h})$. Our best results were obtained with $\alpha = 0.55$, $\beta = 0.55$, and $r_{\max} = 6$. Accuracy increased monotonically with n_p, but so did running time (since more boxes needed to be matched for each image pair), and the gains in accuracy diminished rapidly after $n_p = 500$, so 500 was settled on.

Each sub-feature p may be associated with multiple landmarks, so the final set of point matches used to estimate H or E may include matches of one point in I_1 to multiple points in I_2, or may include multiple instances of the same point match. We tested three approaches to handling this on our tuning set:

1. Score match (p_1, p_2) as $1/d(\mathbf{v}_1, \mathbf{v}_2)$, and keep only the highest-score match for each p_1,
2. Score match (p_1, p_2) as $n/d(\mathbf{v}_1, \mathbf{v}_2)$, where n is the number of times that match occurs, and keep only the highest-score match for each p,
3. Simply keep all matches to each point.

Table 1 This table summarizes the performance of the five methods being evaluated on the KITTI odometry set

Feature Type	Mean pose error (m)	# failures
Object Landmarks	40.402	**1**
Plain object features	67.918	94
SIFT	127.062	56
LIFT	104.279	105
D2Net	**16.578**	5

Object Landmarks are by far the best. They outperform the next-best method by 68% on pose error, and result in only one localization failure, while the next-best method results in 56

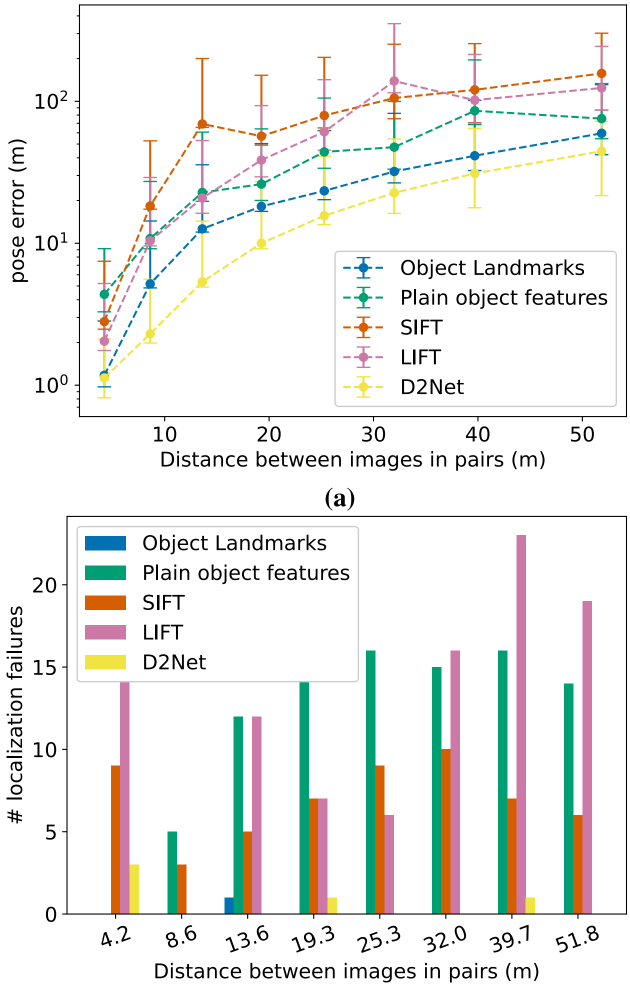
Approach 3 outperformed 1 and 2 by 18% and 9% on pose error, respectively, and had no failures. We believe this is because it preserves the most information from the matching process. RANSAC handles this naturally: if some $p^1 \in I^1$ has multiple matches in I^2, RANSAC finds which match is most consistent with other matches. If (p^1, p^2) occurs multiple times, this suggests the system has more confidence in the match. It will be counted multiply by RANSAC, which has the effect of "up-weighting" the pair in proportion to the system's confidence.

In all of our KITTI experiments, a RANSAC threshold of 6 was used to estimate the essential matrix E.

4.1.3 Final evaluation

The final evaluation set consists of 35, 744 frame pairs taken from traversals 00, 02 to 05, and 08 to 10. We evaluate the following approaches: SIFT feature matching alone; **Object Landmark** matching as described in Sect. 4; and plain object feature matching, where we use the same set of **b**s and **q**s as **Object Landmark** matching, but ignore P, using the centroids of matched objects as point matches. As the object-feature and SIFT approaches are components of **Object Landmarks**, comparing them to **Object Landmarks** serves as an ablation analysis.

To provide a comparison with contemporary learned image features, we also evaluate LIFT (Yi et al. 2016) and D2Net (Dusmanu et al. 2019) feature matching, the latter of which was published as this work was in the final stages of preparation. LIFT features were extracted with the network models trained with rotation augmentation. The D2Net weights used were those trained only on D2Net's authors' "MegaDepth" dataset, and D2Net was run in pyramidal mode to extract features at multiple scales. Only the 3500 highest-score D2Net features on an image were used, since this was roughly the average number of SIFT features detected in each KITTI image. Table 1 summarizes the performance of each approach.



(a)

(b)

Fig. 5 Comparison of pose error (**a**) and failure rates (**b**) on image pairs, clustered by the true distance between the images. Upper and lower error bars indicate the mean of only the errors above and below the whole cluster's mean error, respectively. Image pairs for which failure occurred under some method are not included when computing the mean pose error for that method on that cluster

As Fig. 5a shows, for frame separation $d_{\text{sep}} > 20$ m, transforms estimated from **Object Landmarks** are 50–70% more accurate than SIFT features alone. Plain object features give lower errors than SIFT at $d_{\text{sep}} \geq 30$ m, but their error is consistently greater than complete **Object Landmarks** when $d_{\text{sep}} \geq 20$m, and the difference grows with d_{sep}. This shows that the quasi-semantic descriptors provided by the CNN are much more robust to changes in scale than are SIFT descriptors. Figure 5b reveals a greater disparity: the failure rate of plain object features is about double that of SIFT features at separations greater than 10 m, while over the 35, 744 frame pairs, just one failure occurs with **Object Landmarks**. This shows that the more numerous and precise matches provided by **Object Landmarks**' sub-features are key to reliable

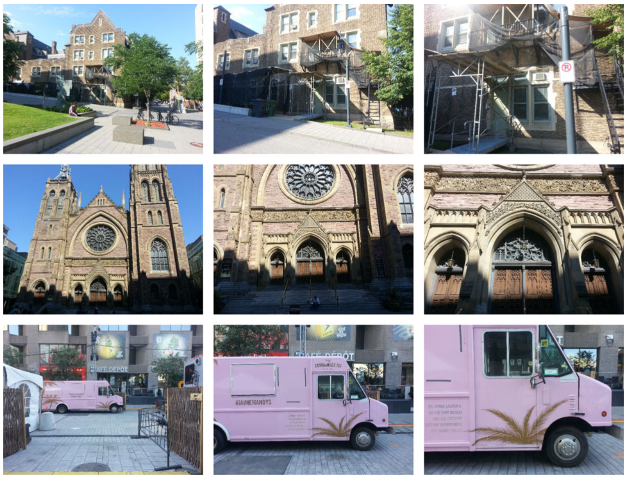


Fig. 6 Examples from the Montreal scale-change dataset. Each row depicts all images of one scene, ordered from far to near

transform estimation, and the use of **q** descriptors to guide SIFT matching greatly improves accuracy.

At all d_{sep}, LIFT feature matching is notably less accurate than **Object Landmarks**, and has the highest failure rate of any evaluated method. D2Net features show 59% better accuracy than **Object Landmarks** on this evaluation, concentrated at small values of d_{sep}. But they fail 5 times as often as **Object Landmarks**, a crucial fact for some applications. In light of the results in Sect. 4.2, it appears D2Net's reduced error is mostly due to better robustness to viewing angle.

4.2 Montreal scale-change dataset

Collected from around McGill University's campus in downtown Montreal, this dataset is designed to capture large, nearly uniform changes in visual scale in real-world environments. The dataset consists of 31 image pairs formed from two to three images of 11 diverse scenes. Each scene contains a central subject chosen to be both approximately planar and visually rich. Each pair of images has ground truth in the form of ten manually-annotated point matches ($\mathbf{g}^{\text{near}}, \mathbf{g}^{\text{far}}$), where $\mathbf{g} = \langle x_{\text{gt}}, y_{\text{gt}} \rangle$. We have made the dataset available online[1]. Figure 6 displays some examples.

We evaluate the same five feature types as on KITTI: SIFT, plain object features, **Object Landmarks**, LIFT, and D2Net. Since these scenes are roughly planar, we use the point matches from each method to compute a homography H, which projects points from one image into the space of the other. For each image pair, we project each $\mathbf{g}^{\text{near}}$ to the far image, and vice-versa, and compute the sum of the distance of each projected point to its true match. This is the

[1] http://www.cim.mcgill.ca/~mrl/montreal_scale_pairs/.

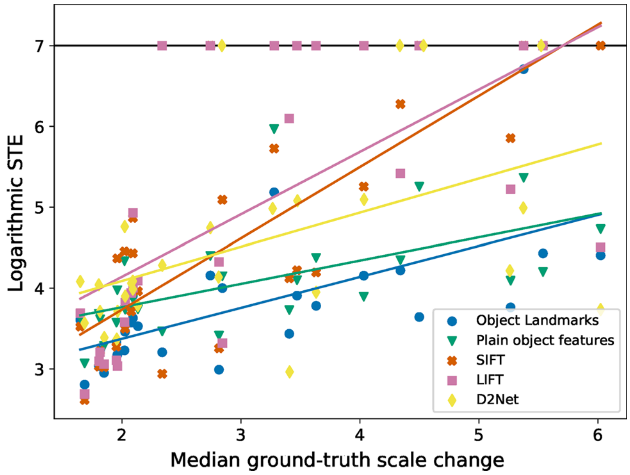


Fig. 7 Logarithmic STE of the homographies computed using each method on each image pair over the Montreal dataset. Best-fit lines to the results are plotted for each feature type. The black horizontal line indicates 10^7, the STE value that was substituted when no transform could be estimated. Points lying on this line indicate that the method in question failed to estimate a transform for this image pair

symmetric transfer error:

$$\text{STE} = \sum_{i}^{N} ||\mathbf{g}_i^{\text{far}} - H\mathbf{g}_i^{\text{near}}||_2 + ||\mathbf{g}_i^{\text{near}} - H^{-1}\mathbf{g}_i^{\text{far}}||_2 \quad (11)$$

If no valid H was found (localization failure), we used a fixed error value of 10^7, which was approximately double the maximum error we observed on any image pair for any method giving a valid H. We define the scale change between a pair of ground-truth point matches i, j as:

$$\text{scale change}_{i,j} = \frac{||\mathbf{g}_i^{\text{near}} - \mathbf{g}_j^{\text{near}}||_2}{||\mathbf{g}_i^{\text{far}} - \mathbf{g}_j^{\text{far}}||_2} \quad (12)$$

For each image pair, we plot the median scale change over all pairs of ground-truth point matches versus the logarithmic STE in Fig. 7. Because Edge Boxes is a stochastic algorithm, and because we are here dealing with a very small dataset, we ran both object-based methods ten times on the dataset, and plotted the mean of the ten trials.

All input images were resized to 900 × 1200 pixels before being processed. The parameters for SIFT, Edge Boxes, LIFT, and D2Net were those described in Sect. 4.1. A RANSAC threshold of 75 was used for plain object features, as it was found to give better performance, while a threshold of 6 was used for all other methods. Table 2 summarizes the performance of each method.

Figure 7 shows that **Object Landmarks** consistently outperform SIFT, LIFT, and D2Net features at scale changes greater than 2.5×. Both SIFT and LIFT provide wildly inaccurate estimates, or fail outright, on all but one image pair with scale change above 4×. D2Net fares better but, surprisingly in light of the results in Sect. 4.1.3, still has 17% greater logarithmic STE than **Object Landmarks**, and fails on four instances. **Object Landmarks** also outperform plain object features. The difference is greatest at small scale changes, where SIFT on its own performs well. This is expected, as plain object features lack spatial precision, and their more scale-robust descriptors make little difference to matching accuracy under small scale changes.

Table 2 This table summarizes the performance of the five methods on the Montreal dataset

Feature Type	Mean log STE	# failures
Object Landmarks	**3.767**	**0**
Plain object features	4.061	0
SIFT	4.641	8
LIFT	4.934	10
D2Net	4.523	4

Object Landmarks here have the lowest errors, and again show far fewer total failures than any other method

A homography H can also be used to map a whole image into the space of another. When this is done with the H matrices computed for these image pairs, they ought to transform the farther image of the pair into a “zoomed-in” image that resembles the nearer image. This allows a qualitative assessment of the results of these experiments. Some example homographies for a large-scale-change pair are displayed in Fig. 8. The improvement of **Object Landmarks** over other methods is especially striking in this assessment. Due to space constraints, we provide the whole set as supplementary material.

5 Place recognition experiments

For each place-recognition experiment, we use images with ground truth camera poses captured over a trajectory by a sensor-equipped vehicle. We sample the images from the vehicle’s trajectory such that every consecutive pair of frames are separated by either some minimum distance, d_{min}, or a minimum angle between gaze directions, θ_{min}. These form the sampled set T. We then treat each image in T as a query q against a “map” set $M = T \setminus \{q\}$. The method under evaluation is used to find a match $m \in M$.

We repeat this process over a range of different values of d_{min}; larger values of d_{min} require a place-recognition method to recognize scenes across greater scale changes. For each value of d_{min}, we perform multiple subsamplings so that similar numbers of queries are performed for each value of d_{min}. θ_{min} is set to the horizontal field-of-view (FOV) of the dataset’s images.

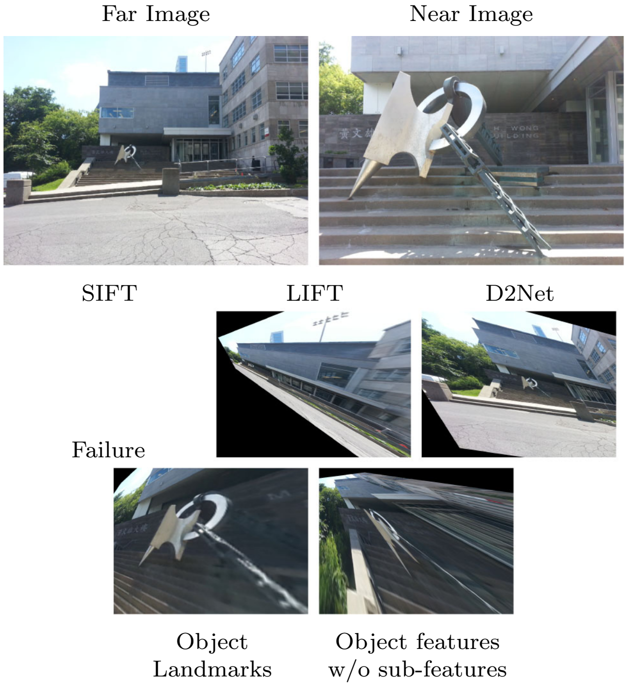


Fig. 8 Homographies computed by each of the methods for an image pair with 6× scale change. SIFT failed to produce any valid homography, so is not displayed here

In a place-recognition context, localization across large changes in scale can mean that multiple map images are "correct" matches to a query, if they view parts of the same scene. Evaluating place recognition by counting correct and incorrect matches, or computing precision and recall, is therefore inappropriate. Instead, where possible, we perform transform estimation between q and m as described in Sect. 3.2, and compute the error in the resulting pose estimate of q. This is more relevant to real robotic localization than a simple correct-incorrect tally: what matters is that the robot have an accurate estimate of its pose, not which specific image from its database it used to estimate that pose.

In these experiments, we evaluate the place recognition method described in Sect. 3.3. We use the same **Object Landmark** configuration as in the transform estimation experiments, except that we set $n_p = 250$ for Edge Boxes, to reduce the computational burden of matching queries over a dataset.

We compare this method against FAB-MAP 2.0 (Cummins and Newman 2011), using the open-source implementation openFABMAP (Glover et al. 2011) with SURF features, as they were used in the original work on FAB-MAP 2.0. We tested using FAB-MAP 2.0 with SIFT features, but found this had reduced accuracy vs. SURF. We also evaluate two recent CNN-based place recognition schemes: CALC (Merrill and Huang 2018), and Region-VLAD (Khaliq et al. 2019). In fairness, we modify the CALC by concatenating the descriptors produced for each colour channel instead of using only the descriptor of a grayscale image. This improves its performance slightly. For all methods, the transform estimation step is performed using **Object Landmarks**, so that differences in accuracy will be due only to the place recognition method. We perform these experiments on the KITTI odometry benchmark, as well as on a subset of the COLD Saarbrucken dataset (Pronobis and Caputo 2009).

5.1 Outdoor: KITTI odometry

Our primary place-recognition experiment was performed on traversal 02 from the KITTI odometry benchmark (Geiger et al. 2012). It is one of the longest traversals and has very little self-overlap. FAB-MAP 2.0 was pre-trained on images sampled with a minimum separation of 30 meters from KITTI odometry traversals 01, 04, 05, 06, 08, 09, and 10 (none of which overlap with 02). The other methods required no pre-training.

We conducted experiments with $d_{\min}$ values of 2, 5, 10, 20, 40, and 80 meters. To more evenly compare FAB-MAP 2.0 with the other methods, which cannot declare the query to be a new location, the most probable match reported by FAB-MAP 2.0 was used, regardless of whether it indicated the query was more likely a new location. The metric of main interest is the pose error defined in Eq. 8. The results are summarized in the "KITTI" section of Table 3, showing that **Object Landmarks** outperform other methods significantly. t_{err} and r_{err} are also presented to show how much each contributes to the pose error: for all methods, about $74-79\%$ of error is translational, and both components show the same relationship between the methods as the mean pose error.

As shown in Fig. 9, our method substantially outperforms FAB-MAP and CALC from $d_{\min} = 5$ to 40m. Region-VLAD is more competitive with our method, but is still substantially outperformed from $d_{\min} = 10$ to 40m: at $d_{\min} = 40$, our method's first error quartile is the lower than Region-VLAD's by a factor of 2. The difference is most pronounced at $d_{\min} =$ 20m, while by $d_{\min} =$ 80m all methods perform very poorly; mean errors are not much better than random matches for any method, suggesting that all methods give mostly incorrect matches.

This experiment assumes that all queries have a correct match in the map. But a robot may also need to consider whether it is visiting an entirely new location. FAB-MAP 2.0 outputs a probability that q represents a new location along with its match probability for each $c \in M$. Our place-recognition scheme can accomplish the same by establishing a threshold on the similarity score.

Table 4 and Fig. 10 illustrate the trade-offs of different settings of the similarity threshold, and the distances at which this place-recognition method performs well. A threshold of 0.1, which would make the "new" rate about equal to that of

Table 3 This table summarizes the results from the place-recognition experiments

	KITTI				COLD	
Method	Mean pose error (m)	Mean t_{err}(m)	Mean r_{err}	# failures	Mean dist. to match (m)	Failure rate (%)
OL	**128.9**	**101.8**	**0.077**	**0**	**5.796**	2.43
FAB-MAP 2.0	279.2	208.2	0.169	0	6.565	**1.42**
CALC	238.8	189.7	0.132	2	7.168	18.33
Region-VLAD	156.2	123.8	0.097	0	6.392	1.55

Failures are queries where no spatially-consistent match was found. **Object Landmarks** have the best accuracy on both the KITTI and COLD experiments, although on COLD, both FAB-MAP 2.0 and Region-VLAD have a lower failure rate. This is because the COLD trajectory that we use has some segments where the robot sees only untextured wall, a case to which the **Object Landmark** method is brittle, since it cannot detect any objects

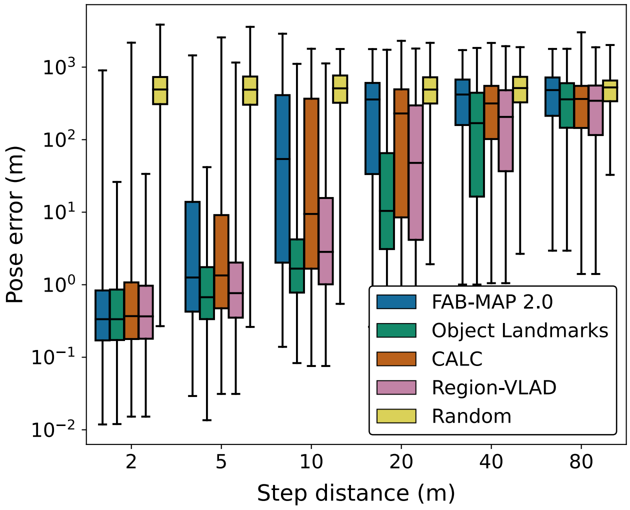


Fig. 9 Pose errors on the KITTI Odometry place-recognition experiments, separated by step distance $d_{\min}$. The y-axis is log-scaled. Each box shows the quartiles of errors for one method at one $d_{\min}$ value. The whiskers indicate the minimum and maximum errors observed. Results from matching queries randomly are included as a baseline. **Object Landmarks** achieves much lower median error than both FAB-MAP and CALC at $d_{\min}$ values from 5 to 40m, and notably outperforms Region-VLAD at $d_{\min} = 10$ and 20m. No method performs much better than chance at $d_{\min} = 80$

Table 4 Statistics of $S_{q,m}$ scores for each $d_{\min}$ in the KITTI experiments

$d_{\min}$	Mean $S_{q,m}$	SD
2	0.34	0.044
5	0.25	0.052
10	0.19	0.068
20	0.14	0.068
40	0.13	0.061
80	0.11	0.025

The scores decline with $d_{\min}$

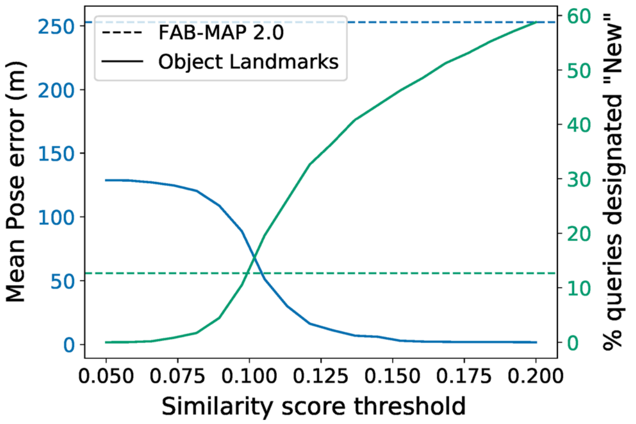


Fig. 10 Impact of a range of $S_{q,m}$ thresholds. At each threshold, queries with no match above the threshold are considered new locations. FAB-MAP 2.0 does not depend on a threshold, so its values are constant. FAB-MAP's mean error is much higher than our system's at any threshold. Even at a threshold that produces the same number of "new" declarations from both systems, FAB-MAP's mean pose error is 223% greater than that of our system

FAB-MAP 2.0, would cut off about half of the $d_{\min} = 80$m and 40m queries. A threshold of 0.16 gives a mean error very close to 0m, but a "new" rate of about 50% - at this threshold, most of the queries in the $d_{\min} = 20$, 40, and 80m experiments are cut off, as are approximately half of the queries in the $d_{\min} = 10$m experiments.

It should be noted that in these experiments, we do not exploit any priors on robotic motion. Such priors could greatly constrain the search space for matches and thus increase the robustness of localizing over large distances, allowing higher thresholds to be used while maintaining a desired accuracy. Ultimately, the choice of threshold must depend on the application, balancing tolerable error against feasible loop-closure distance and map size.

5.2 Indoor: COLD Saarbrucken

To evaluate the performance of our method on an indoor environment, we conducted place-recognition experiments on the COsy Localization Database (COLD) produced by Pronobis and Caputo (2009). COLD is an indoor navigation dataset consisting of sensor data collected over traversals of various university buildings. The data was gathered by a robot

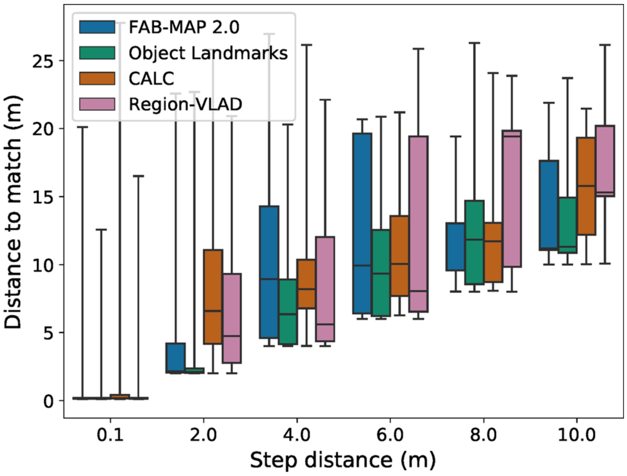


Fig. 11 Box plots of ground-truth distances between queries and matches proposed by each method, for each value of $d_{\min}$. At $d_{\min} = 8$ m, FAB-MAP 2.0's second quartile is almost equal to the third quartile, and at $d_{\min} = 8$, the second is almost equal to the first, making it hard to discern in each case

equipped with a forward-facing monocular camera, and has precise ground truth poses associated with each image, as well as annotations indicating which room in the building the robot is in at each frame.

For COLD, we ran experiments with $d_{\min}$ values of 0.1, 2, 4, 6, 8, and 10 meters. These values were chosen because of the small size of these traversals, and because changes in visual scale observed under a given motion in the z direction will tend to be larger in an enclosed indoor space than outdoors.

The monocular images in COLD are unrectified, and the intrinsic parameters of the cameras used are not available, so we cannot accurately estimate camera poses, though geometric consistency checking via the cheirality check can still be performed by assuming arbitrary camera parameters. For this reason, in our experiments on COLD we consider the ground truth distance of q from proposed match m as our metric of interest. By construction, this cannot be less than $d_{\min}$ for a given experiment, but it is a useful proxy for the "correctness" of the pose estimate, as smaller baselines between image pairs make transform estimation more accurate, as shown in our other experiments. The results are summarized in the "COLD" section of Table 3.

Figure 11 shows that **Object Landmarks** usually outperforms the other methods at $d_{\min} \geq 2$ m. At $d_{\min} = 4$ and 6 m, Region-Vlad has a slightly lower second quartile, but a much higher third quartile, than **Object Landmarks**, and FAB-MAP 2.0 and CALC are strictly dominated. Region-VLAD's performance degrades at $d_{\min} = 8$ m, while FAB-MAP 2.0 and CALC anomalously perform as well or better than **Object Landmarks**. At $d_{\min} \geq 10$ m, where the most extreme scale changes are experienced, **Object Landmarks** dominate all other methods. It is notable that of the methods other than **Object Landmarks**, the only one that operates by detecting and matching object-like regions between images is Region-VLAD, which has the best performance of the three.

6 Conclusions

We have proposed **Object Landmarks**, a new feature type for visual localization. An **Object Landmark** consists of an object detection with a quasi-semantic descriptor from intermediate activations of a CNN, as well as a set of point features such as SIFT on the object. This design is motivated by a holistic philosophy of image processing, which considers that information at both high and low levels of abstraction from the raw image data are important to the task. We have demonstrated that **Object Landmarks** offer major improvements over the state of the art when used to localize over changes in visual scale greater than 3×. These results are supportive of our holistic approach, and we believe that future work on visual localization, and on image processing more broadly, may benefit from taking such an approach.

Our experimental evaluation did not investigate the robustness of **Object Landmarks** to large changes in viewing angle and scene appearance, both of which are of considerable importance in visual localization. In future work, we would like to investigate these aspects of the feature's performance in more detail.

The off-the-shelf construction of **Object Landmarks** has the advantage that they do not require any re-training to apply them to new environments. Nonetheless, one direction for future research would be to attempt to train a neural network to replace some or all of the stages of this pipeline. For such a system to perform reliably over large scale changes would require training data exhibiting such changes over a wide variety of types of environment, which may be difficult to obtain. Nonetheless, this could provide gains in accuracy and speed, and would remove the need for some of the parameter-tuning associated with components like Edge Boxes.

Acknowledgements This work was conducted at the Samsung AI Center Montreal.

References

Bay, H., Ess, A., Tuytelaars, T., & Van Gool, L. (2008). Speeded-up robust features (surf). *Computer Vision and Image Understanding*, *110*(3), 346–359. https://doi.org/10.1016/j.cviu.2007.09.014.

Bowman, S. L., Atanasov, N., Daniilidis, K., & Pappas, G. J. (2017). Probabilistic data association for semantic slam. In: *2017 IEEE international conference on robotics and automation (ICRA)*, pp. 1722–1729.

Brown, M., & Lowe, D. G. (2002). Invariant features from interest point groups. In: *BMVC*, Vol. 4.

Chen, Z., Liu, L., Sa, I., Ge, Z., & Chli, M. (2018). Learning context flexible attention model for long-term visual place recognition. *IEEE Robotics and Automation Letters*, *3*(4), 4015–4022.

Cummins, M., & Newman, P. (2010) Invited Applications Paper FAB-MAP: Appearance-based place recognition and mapping using a learned visual vocabulary model. In: *27th International conference on machine learning (ICML2010)*.

Cummins, M., & Newman, P. (2011). Appearance-only slam at large scale with fab-map 20. *The International Journal of Robotics Research*, *30*(9), 1100–1123. https://doi.org/10.1177/0278364910385483.

DeTone, D., Malisiewicz, T., & Rabinovich, A. (2018). Superpoint: Self-supervised interest point detection and description. In: *Proceedings of the IEEE conference on computer vision and pattern recognition workshops*, pp. 224–236.

Dudek, G., & Jenkin, M. (2010). *Computational principles of mobile robotics*. Cambridge: Cambridge University Press.

Dudek, G., & Zhang, C. (1995). Pose estimation from image data without explicit object models. In: *Research in computer and robot vision*. World Scientific, pp. 19–35.

Dudek, G., & Zhang, C. (1996). Vision-based robot localization without explicit object models. In: *IEEE International conference on robotics and automation*, Vol. 1. IEEE, pp. 76–82.

Dusmanu, M., Rocco, I., Pajdla, T., Pollefeys, M., Sivic, J., Torii, A., & Sattler, T. (2019). D2-Net: A Trainable CNN for joint detection and description of local features. In: *Proceedings of the 2019 IEEE/CVF conference on computer vision and pattern recognition*

Engel, J., Schöps, T., & Cremers, D. (2014). Lsd-slam: Large-scale direct monocular slam. In: *European conference on computer vision*. Springer, pp. 834–849.

Faugeras, O., Luong, Q.-T., & Papadopoulo, T. (2001). The geometry of multiple images: The laws that govern the formation of multiple images of a scene and some of their applications. MIT press

Fox, D., Burgard, W., & Thrun, S. (1999). Markov localization for mobile robots in dynamic environments. *Journal of Artificial Intelligence Research*, *11*, 391–427.

Galindo, C., Saffiotti, A., Coradeschi, S., Buschka, P., Fernandez-Madrigal, J. A., & Gonzalez, J. (2005). Multi-hierarchical semantic maps for mobile robotics. In: *2005 IEEE/RSJ international conference on intelligent robots and systems*, pp. 2278–2283.

Garg, S., Suenderhauf, N., & Milford, M. (2018). Lost? appearance-invariant place recognition for opposite viewpoints using visual semantics. *arXiv preprint* arXiv:1804.05526.

Geiger, A., Lenz, P., & Urtasun, R. (2012). Are we ready for autonomous driving? the kitti vision benchmark suite. In: *Conference on computer vision and pattern recognition (CVPR)*

Glover, A., Maddern, W., Warren, M., Reid, S., Milford, M., & Wyeth, G. (2011). Openfabmap: An open source toolbox for appearance-based loop closure detection. In: *The international conference on robotics and automation*. St Paul, Minnesota: IEEE.

Hartley, R. I. (1993). Cheirality invariants. In: *Proceedings of DARPA image understanding workshop*, pp. 745–753.

He, K., Zhang, X., Ren, S., & Sun, J. (2015) Deep residual learning for image recognition. arXiv preprint arXiv:1512.03385

Holliday, A., & Dudek, G. (2018). Scale-robust localization using general object landmarks. In: *2018 IEEE/RSJ international conference on intelligent robots and systems (IROS)*, pp. 1688–1694

Holliday, A., & Dudek, G. (2020). Pre-trained cnns as visual feature extractors: A broad evaluation. In: *2020 17th conference on computer and robot vision (CRV)*. IEEE, pp. 78–84.

Huang, G., Liu, Z., Maaten, L. v. d., & Weinberger, K. Q. (2017). Densely connected convolutional networks. In: *2017 IEEE conference on computer vision and pattern recognition (CVPR)*, pp. 2261–2269.

Kaeli, J. W., Leonard, J. J., & Singh, H. (2014). Visual summaries for low-bandwidth semantic mapping with autonomous underwater vehicles. In: *2014 IEEE/OES Autonomous underwater vehicles (AUV)*, pp. 1–7.

Kendall, A., & Cipolla, R. (2016). Modelling uncertainty in deep learning for camera relocalization. In: *Proceedings of the international conference on robotics and automation (ICRA)*

Khaliq, A., Ehsan, S., Chen, Z., Milford, M., & McDonald-Maier, K. (2019). A holistic visual place recognition approach using lightweight cnns for significant viewpoint and appearance changes. *IEEE Transactions on Robotics*, pp. 1–9.

Klein, G., & Murray, D. (2007). Parallel tracking and mapping for small ar workspaces. In: *Proceedings of the 2007 6th IEEE and ACM international symposium on mixed and augmented reality, ser. ISMAR '07*. Washington, DC: IEEE Computer Society, pp. 1–10. https://doi.org/10.1109/ISMAR.2007.4538852

Konolige, K. (1998). Small vision systems: Hardware and implementation. *Robotics Research*. Springer, pp. 203–212.

Kriegman, D. J., Triendl, E., & Binford, T. O. (1989). Stereo vision and navigation in buildings for mobile robots. *IEEE Transactions on Robotics and Automation*, *5*(6), 792–803.

Krizhevsky, A., Sutskever, I., & Hinton, G. E. (2012). Imagenet classification with deep convolutional neural networks. In F. Pereira, C. J. C. Burges, L. Bottou, & K. Q. Weinberger (Eds.), *Advances in neural information processing systems* (Vol. 25, pp. 1097–1105). Curran Associates Inc.

Leonard, J. J., & Durrant-Whyte, H. F. (1991). Mobile robot localization by tracking geometric beacons. *IEEE Transactions on Robotics and Automation*, *7*(3), 376–382.

Li, J., Eustice, R. M., & Johnson-Roberson, M. (2015). High-level visual features for underwater place recognition.

Li, J., Meger, D., Dudek, G. (2019). Semantic mapping for view-invariant relocalization. In: Proceedings of the. (2019). *IEEE international conference on robotics and automation (ICRA 19), Montreal, Canada*

Lindeberg, T. (1994). Scale-space theory: a basic tool for analyzing structures at different scales. *Journal of Applied Statistics, 21*(1–2), 225–270. https://doi.org/10.1080/757582976.

Linegar, C., Churchill, W., Newman, P. (2016). Made to measure: Bespoke landmarks for 24-hour, all-weather localisation with a camera. In: *2016 IEEE International Conference on Robotics and Automation (ICRA)*, pp. 787–794.

Lowe, D. G. (1999). Object recognition from local scale-invariant features. In: *Proceedings of the international conference on computer vision-volume 2*, ser. ICCV '99. IEEE Computer Society, Washington, pp. 1150, http://dl.acm.org/citation.cfm?id=850924.851523

MacKenzie, P., & Dudek, G. (1994). Precise positioning using model-based maps. In: *1994 IEEE international conference on robotics and automation*. IEEE, pp. 1615–1621.

Merrill, N., & Huang, G. (2018). Lightweight unsupervised deep loop closure. In: *Proceedings of robotics: science and systems (RSS)*, Pittsburgh

Mirowski, P., Grimes, M. Koichi, Malinowski, M., Moritz Hermann, K., Anderson, K., Teplyashin, D., Simonyan, K., Kavukcuoglu, K. Zisserman, A., & Hadsell, R. (2018). Learning to navigate in cities without a map, 03.

Mur-Artal, R., Montiel, J. M. M., & Tardós, J. D. (2015). Orb-slam: A versatile and accurate monocular slam system. *CoRR*

Nistér, D. (2004). An efficient solution to the five-point relative pose problem. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, *26*(6), 756–777. https://doi.org/10.1109/TPAMI.2004.17.

Oliva, A., & Torralba, A. (2001). Modeling the shape of the scene: A holistic representation of the spatial envelope. *International Journal of Computer Vision*, *42*(3), 145–175. https://doi.org/10.1023/A:1011139631724.

Paszke, A., Gross, S., Chintala, S., Chanan, G., Yang, E., DeVito, Z., Lin, Z., Desmaison, A., Antiga, L., & Lerer, A. (2017). Automatic differentiation in PyTorch. In: *NIPS Autodiff workshop*

Pronobis, A., & Caputo, B. (2009). COLD: COsy localization database. *The International Journal of Robotics Research (IJRR), 28*(5):588–594 http://www.pronobis.pro/publications/pronobis2009ijrr

Rublee, E., Rabaud, V., Konolige, K., & Bradski, G. (2011). Orb: An efficient alternative to sift or surf. In: *proceedings of the 2011 international conference on computer vision*, ser. ICCV '11. Washington, DC: IEEE Computer Society, pp. 2564–2571. https://doi.org/10.1109/ICCV.2011.6126544

Russakovsky, O., Deng, J., Su, H., Krause, J., Satheesh, S., Ma, S., et al. (2015). ImageNet large scale visual recognition challenge. *International Journal of Computer Vision (IJCV)*, *115*(3), 211–252.

Salas-Moreno, R. F., Newcombe, R. A., Strasdat, H., Kelly, P. H., & Davison, A. J. (2013). Slam++: Simultaneous localisation and mapping at the level of objects. In: *Proceedings of the IEEE conference on computer vision and pattern recognition*, pp. 1352–1359.

Simo-Serra, E., Trulls, E., Ferraz, L., Kokkinos, I., Fua, P., & Moreno-Noguer, F. (2015). Discriminative learning of deep convolutional feature point descriptors. In: *2015 IEEE international conference on computer vision (ICCV)*, pp. 118–126.

Simonyan, K., & Zisserman, A. (2014). Very deep convolutional networks for large-scale image recognition. *CoRR*, abs/1409.1556, arXiv:1409.1556

Spencer, C., & Darvizeh, Z. (1981). The case for developing a cognitive environmental psychology that does not underestimate the abilities of young children. *Journal of Environmental Psychology*, *1*(1), 21–31.

Sünderhauf, N., Dayoub, F., Shirazi, S., Upcroft, B., & Milford, M. (2015). On the performance of convnet features for place recognition. *CoRR*, arXiv:1501.04158

Sünderhauf, N., Shirazi, S., Jacobson, A., Dayoub, F., Pepperell, E., Upcroft, B., & Milford, M. (2015). Place recognition with convnet landmarks: Viewpoint-robust, condition-robust, training-free. In: *Proceedings of robotics: science and systems (RSS)*

Uijlings, J. R. R., van de Sande, K. E. A., Gevers, T., & Smeulders, A. W. M. (2013) Selective search for object recognition. *International Journal of Computer Vision, 104*(2):154–171, https://ivi.fnwi.uva.nl/isis/publications/2013/UijlingsIJCV2013

Yi, K. M., Trulls, E., Lepetit, V., & Fua, P. (2016). Lift: Learned invariant feature transform. In: *European conference on computer vision (ECCV)*, pp. 467–483.

Zitnick, L., & Dollar, P. (2014) Edge boxes: Locating object proposals from edges. In: *ECCV: European conference on computer vision*, https://www.microsoft.com/en-us/research/publication/edge-boxes-locating-object-proposals-from-edges/

Publisher's Note Springer Nature remains neutral with regard to jurisdictional claims in published maps and institutional affiliations.

Andrew Holliday is a PhD candidate at the Mobile Robotics Laboratory of McGill University. He is interested in visual navigation, neural representations for computer vision, and applying machine learning to combinatorial optimization problems.

Gregory Dudek does research on sensing for mobile robotics including vision, robot pose estimation (position estimation), recognition, path planning. He is also looking into a few non-robotic problems such as those related to recommender systems.