Honda Research Institute EU
Honda Research Institute Europe GmbH
https://www.honda-ri.de/

Unsupervised Fast Visual Localization and Mapping with Slow Features

Muhammad Haris, Mathias Franzius, Ute Bauer-Wersing

2021

Preprint:

This is an accepted article published in IEEE International Conference on Image Processing (ICIP). The final authenticated version is available online at: https://doi.org/[DOI not available]

UNSUPERVISED FAST VISUAL LOCALIZATION AND MAPPING WITH SLOW FEATURES

Muhammad Haris[1], *Mathias Franzius*[2], *Ute Bauer-Wersing*[1]

[1]Frankfurt University of Applied Sciences, Germany
[2]Honda Research Institute Europe GmbH, Germany

ABSTRACT

Visual localization is the task of accurately estimating the camera's position in a known environment. State-of-the-art methods use the 3D structure of a scene for precise visual localization. However, 3D scene reconstruction is resource-intensive in terms of hardware requirements and computation time, making it infeasible to run on low-cost embedded hardware. Unsupervised spatial representation learning with Slow Feature Analysis (SFA) enables computationally inexpensive localization and mapping. This paper analyzes SFA-based and the well-known structure-based localization, i.e., Active Search, in two distinct settings: short-term temporal and extreme spatial generalization. We present the experimental results from an outdoor environment and compare both methods w.r.t localization accuracy and computation time. Results show that the SFA-based approach is 886x faster in mapping time and 34x faster in localization than Active Search while achieving comparable localization accuracy in our test scenario.

Index Terms— Localization, Mapping, Unsupervised learning, Computer vision, Service robots

1. INTRODUCTION

Visual localization estimates the pose of a camera relative to a known scene representation. It has become an attractive alternative to non-visual localization techniques due to the low cost, less weight, and high availability of cameras. Localization, in general, is a prerequisite for many robotic applications, autonomous cars, and AR/VR systems.

Visual localization is among the well-researched areas; hence, various methods address this challenging task. State-of-the-art *structure-based* methods [1, 2] use a pre-built 3D map of a scene and estimate a test image's pose relative to it. The localization procedure consists of extracting and matching 2D features of a test image to a 3D point cloud reconstruction computed from a structure-from-motion (SfM) step [3]. Afterwards, a subsequent step computes a test image's pose by solving a Perspective-n-Point (PnP) problem. *Image retrieval* [4] matches the 2D features of a test image to the 2D features from an image database and approximates its pose by retrieving the most similar database image. Visual localization based on *convolution neural networks (CNNs)* has gained significant interest recently. Some researchers have replaced the localization pipeline's parts with CNN-based learning [5, 6]. In [7], the authors have trained a neural network in an end-to-end way to regress pose from single images directly. Despite impressive results, CNN-based localization does not compete with the structure-based methods w.r.t pose accuracy and fails to generalize beyond the training data [8].

The finding of Place- and Head-Direction Cells to be involved in rodent localization ability [9, 10] has led to different *bio-inspired models* for localization [11, 12, 13]. Earlier learning approaches [12] reproduce these cells' firing characteristics using a hierarchical model. The model uses the concept of slow features analysis (SFA) [14], and the intuition behind it is that behaviorally meaningful information (e.g., position or orientation of an animal in space) changes on a slower timescale compared to the primary sensory input (e.g., pixel values in a video). Previous work [15] implements SFA-based localization on a robot in an outdoor environment and achieves similar performance compared to state-of-the-art visual SLAM methods, i.e., ORB and LSD-SLAM [16, 17] in different test scenarios.

In this work, we compare SFA-based localization with one of the fastest and most widely used *structure-based* localization method, i.e., Active Search [1], in real-world test scenarios. It achieves similar or higher registration performance than tree-based search and faster run-times [2] while producing precise localization results independent of the environment scale [2, 8]. In contrast to traditional SLAM systems, SFA-based and Active Search methods have separate mapping and localization phases that allows a straightforward comparison. We test both methods on subsequent robot recordings along a similar path (i.e., temporal generalization) and for the case when the train and test recordings consist of significantly different robot paths (i.e., spatial generalization). We measure localization accuracy and computation time as performance metrics. In summary, this work's motivation is two-fold: First, to show through real-world experiments if an *unsupervised learning* approach can compete with the existing state-of-the-art *structure-based* approach w.r.t localization accuracy. Second, to compare both methods, w.r.t run-time, and hardware-requirements in the mapping and localization phase, respectively.

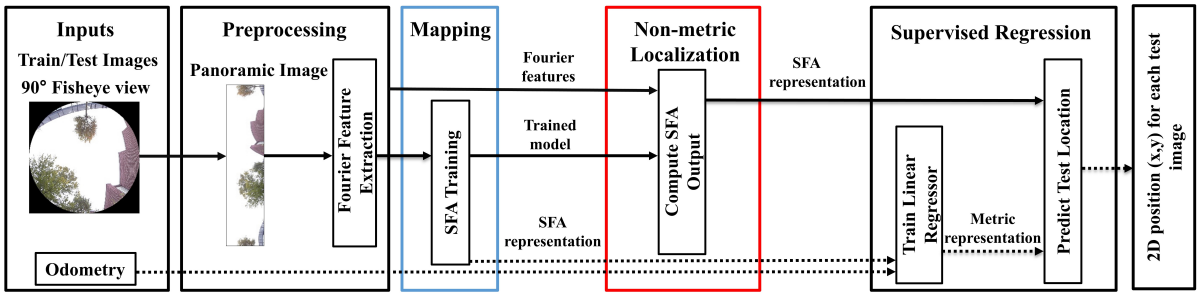


Fig. 1. **SFA Localization and Mapping Pipeline:** The inputs to the pipeline are fisheye views and odometry data. The preprocessing module projects input images to panoramic views and extract Fourier features from them. The mapping module uses the extracted features of training set images to learn SFA representations. The localization module uses the Fourier features of the test images and the trained SFA model to obtain the SFA output. The learned SFA representations, together with the odometry data, are used to compute a regression function, which maps SFA space to metric space. Finally, it applies the learned regression function on the SFA output to predict a 2D position (x, y) for each test image.

2. METHODS

2.1. Learning Spatial Representations with SFA

Slow feature analysis (SFA) transforms a multidimensional time series $\mathbf{x}(t)$ to slowly varying output signals [14]. The objective is to find instantaneous scalar input-output functions $g_j(\mathbf{x})$ such that the output signals $s_j(t) := g_j(\mathbf{x}(t))$ minimize $\Delta(s_j) := \langle \dot{s}_j^2 \rangle_t$ with $\langle \cdot \rangle_t$ and $\dot{s}$ indicating temporal averaging and the derivative of s, respectively. The Δ-value measures the slowness of an output signal, and its minimization is the optimization objective.

The goal of learning spatial representations with SFA is to find functions representing the robot's position on the x- and y-axis as slowly varying features. SFA-based learning depends on temporal statistics of the input data; therefore, if the relevant information (here: robot position) during training changes on a slower timescale compared to other variables, it will be encoded as the slowest features in the learned representations. The work in [12] provides a detailed analysis of SFA for localization based on the inputs' temporal statistics.

2.2. SFA-based Localization and Mapping Pipeline

Fig. 1 shows the proposed pipeline. In our setup, for each recording, an autonomous robot traverses the environment and captures $180°$ fisheye views with the lens pointing upwards while storing the associated odometry data. Extracting Fourier magnitudes from upright images is a simple and effective preprocessing for localization, since all views under yaw-rotation from the same position yield the same feature vector. In this work, we reduce the field of view (FoV) of input images to $90°$ because there are technical limitations (c.f. 2.3) while running Active Search on $180°$ fisheye views. A fair comparison of both localization approaches requires identical inputs. Therefore, we use the images with the reduced FoV for both localization methods. Please note SFA representation learning with the entire FoV of input images is possible [15], and it also improves localization accuracy.

The ***preprocessing module*** unwraps the input images to panoramic views and extract Fourier features from them. The feature extraction step performs row-wise Fourier expansion and retains the magnitude part corresponding to the first 15 Fourier coefficients. While SFA localization on pixel images is even higher in accuracy [18], the use of Fourier signatures retains sufficient sensory representations and is much faster. The preprocessing steps are the same for both training and test set images. The ***mapping module*** uses the extracted Fourier features of the training images to learn an SFA representation. The learning phase has two steps, i.e., linear SFA followed by quadratic SFA. Linear SFA can find better solutions with a nonlinear expansion of its inputs, similar to nonlinearities in neural networks. However, this expansion increases feature dimensionality. Therefore, the first step performs dimensionality reduction with linear SFA for computational efficiency. Finally, the second step projects the features into the space of quadratic monomials and then applies SFA to extract slow features. The input and output dimensionality of the first step is 450 and 20, respectively. The second step outputs an eight-dimensional vector $s_{1...8}$ for each image, where each unit represents the slow features learned by SFA. The ***localization module*** uses the extracted Fourier features of the test images and the trained model to compute the slow feature output. The computation step is instantaneous, i.e., only require a single image to compute the model output. Since SFA is unsupervised, it does not require metric ground truth labels for training, and previous experiments [19, 20] show that it is possible to perform robot navigation directly in the slow feature space. The ***regression module*** projects the learned SFA representation to the metric space by using the odometry data. The last step in the regression module uses the learned metric representation and the SFA output to predict 2D-position (x, y) for each test image. Thus, this

projection mainly serves the purpose of metric evaluation of the SFA-based localization method.

2.3. Structure-based Localization and Mapping

We use Active Search [1] to obtain baseline results. It determines a test image's pose relative to an offline built 3D scene reconstruction. Its available implementation[1] expects the input (3D reconstruction) in the Bundler[2] file format. However, Bundler uses a pinhole camera model and cannot handle more sophisticated models, e.g., fisheye lenses. Due to this limitation, we reduced the FoV of input images to 90° and computed the 3D scene reconstruction with the COLMAP's [3] pinhole camera model. Despite the reduced FoV, the remaining image content was sufficient to generate a 3D point cloud (mapping phase, fig. 2). Please note that we did not use test images in the mapping phase to prevent their influence on the final model. The localization phase uses the 3D model and the Active Search pipeline [1] to estimate a test image's pose.

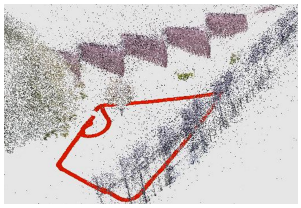


Fig. 2. **3D scene reconstruction:** A robot traverses the area's border (red) to collect an image dataset. We computed the 3D scene reconstruction with COLMAP's pipeline [3].

3. EXPERIMENTS

We perform experiments in an outdoor environment with an area of 15×9 meters. We use an autonomous robot equipped with a 180° fisheye camera for the recordings. The camera captures images of size 2880×2880 pixels. Each recording phase has two working periods; in the first one, the robot traverses the border of the area (fig. 2) using a standard lawn mower wire guidance technology. For this working period, it is possible to obtain highly precise metric coordinates (x, y) with odometry using loop closure and subsequent error correction [21]. In the second working period, the robot traverses freely within the premises of the area enclosed by border wire. During recordings, it saves images and odometry data. A preprocessing step reduces the FoV of all the collected views to 90°. To obtain the SFA results, we use the same procedure for learning spatial representations (mapping) and estimating the 2D position (x, y) of a test image (localization), as described in section 2.2.

[1]https://github.com/hanjianwei/ACG-Localizer
[2]http://www.cs.cornell.edu/ snavely/bundler/

As Active Search needs a 3D scene reconstruction for localization, we first use COLMAP [3] to create it, followed by recovering the model's scale using odometry and exporting it in the Bundler file format. We also use COLMAP to extract SIFT features [22] of the test set images. The localization phase uses the Active Search pipeline for pose estimation of test images. We report median and mean Euclidean distance between the predicted (x', y') and actual test locations (x, y) for both methods.

3.1. Short-term Temporal Generalization

This experiment aims to test the re-localization ability of both methods over a short time. We recorded the training and test set images on a similar trajectory but over a time difference of 30 minutes, causing a slight variation in lighting conditions. The training and test sets consist of 1083 and 284 images, respectively. Table 1 reports both median and mean localization performance. In this experiment, Active Search performed better than SFA-based localization, both in terms of median and mean error.

3.2. Spatial Generalization

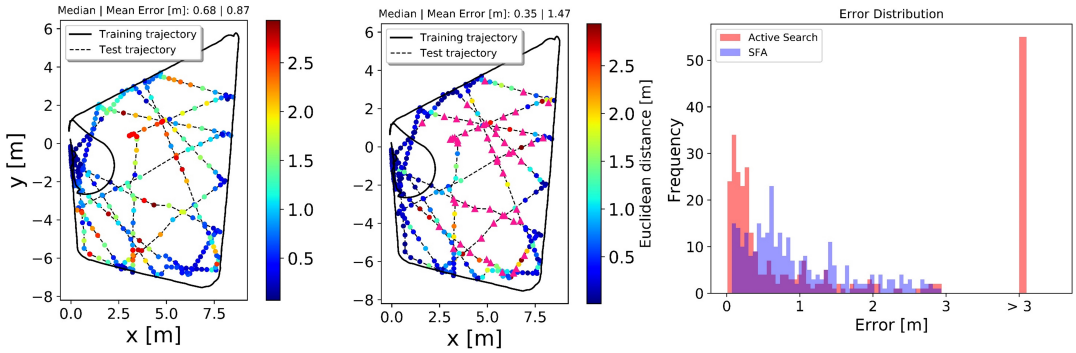
This experiment aims to test the re-localization ability of both methods when the train and test sets contain images from different trajectories but in similar conditions. We use images captured by the robot from the border and inner field positions as training and test data. The training and test sets consist of 1141 and 298 images, respectively. Table 1 reports both median and mean localization performance, and Fig. 3 visualizes the localization error. Active Search has again achieved a better median localization accuracy. However, when comparing the mean accuracy, SFA has shown better results. This is due to the 55 outliers (out of n=298) with localization error greater than three meters produced by Active Search, especially in the center of the traversed area c.f. Fig. 3b. For detailed error statistics refer to Table 1 and Fig. 3c.

3.3. Time Evaluation

We perform time evaluation for mapping and localization phases using one of the datasets containing 1083 training and 284 test images, but it also extrapolates to the second dataset. Both phases of the SFA-based method were computed on a standard CPU (Intel i5 $-$ 6500 with 3.20Ghz). In contrast, for Active Search, we use a GPU (GeForce RTX 2070) for 3D scene reconstruction (mapping) and a standard CPU for localization. The mapping phase in Table 2 reports the preprocessing and mapping times for learning SFA representations and creating the 3D scene reconstruction. The localization phase in Table 2 presents the time it took to localize 284 test images. The preprocessing time in the localization phase for Active Search includes extracting SIFT features from test images and relevant information from the

Table 1. Median and Mean errors of SFA-based and Active Search localization for generalization over time and space.

Scenario	Method	Median [m]	Mean [m]	Quartiles [m]	
				3rd	4th
Temporal Generalization	SFA	0.46	0.52	0.69	1.54
	Active Search	**0.28**	**0.40**	**0.44**	**1.09**
Spatial Generalization	SFA	0.68	**0.87**	**1.24**	**2.55**
	Active Search	**0.35**	1.47	1.5	15.32



(a) SFA (b) Active Search (c) Error Distribution

Fig. 3. **Spatial Generalization:** The robot follows the border wire (solid line) to collect the training images and freely traverses (dashed line) to collect test images. (a) and (b) visualizes the localization error indicated by the color codes for both methods. SFA performs well in spatial generalization and moderately degrades for test images towards the field's center. Active Search has quite good estimations near the border wire, but its performance degrades for the field's center's test images. Pink triangles indicate the positions where the estimated error is higher than three meters. (c) shows error distribution $(n = 298)$ for both methods. Almost 20% of the test locations predicted by Active Search have errors of more than three meters.

Table 2. The computation times in mapping indicate learning of SFA representations and creating the 3D scene reconstruction. The localization phase shows the time to localize 284 test set images. The SFA-based method is **886x** faster in mapping and **34x** faster in localization time than Active Search for this dataset.

Phase	Method	Preprocessing time [s]	Mapping time [s]	Localization time [s]	Ratio Computation time
Mapping	SFA [CPU]	**44**	**5**	-	43424 / **49** = 886
	3D Reconstruction [GPU]	90	43334	-	
Localization	SFA [CPU]	**12**	-	**2**	479 / **14** = 34
	Active Search [CPU]	185	-	294	

created 3D model. Based on the results, our method for this dataset is 886x faster in mapping than 3D reconstruction and 34x faster in localization than Active Search. We can further increase this factor by optimizing the preprocessing for SFA, which is currently implemented in python and requires more than 85% of the overall localization time.

4. CONCLUSION

This work aimed to compare unsupervised SFA-based and structure-based Active Search methods for visual localization in a real-world setting. Although Active Search has achieved better median localization accuracy, some highly inaccurate results (see Fig. 3b) might not befit a real-world robot operation. SFA-localization here provides a robot's 2D position (x, y) instead of a 6D pose when using Active Search. Despite that, many service robots, e.g., lawnmowers, do not require a 6D pose for localization. Our approach offers extremely fast run-times for mapping and localization phases, with a slight degradation of accuracy than Active Search. Moreover, the experiments show SFA localization's ability to generalize better to previously unseen trajectories than Active Search. The drastically faster run-times make the SFA-based localization an attractive solution to run on low-cost embedded hardware without a GPU.

5. REFERENCES

[1] Torsten Sattler, Bastian Leibe, and Leif Kobbelt, "Improving image-based localization by active correspondence search," in *Computer Vision – ECCV 2012*. 2012, pp. 752–765, Springer Berlin Heidelberg.

[2] T. Sattler, B. Leibe, and L. Kobbelt, "Efficient effective prioritized matching for large-scale image-based localization," *IEEE Transactions on Pattern Analysis and Machine Intelligence*, vol. 39, no. 9, pp. 1744–1756, Sep. 2017.

[3] Johannes Lutz Schönberger and Jan-Michael Frahm, "Structure-from-Motion Revisited," in *Conference on Computer Vision and Pattern Recognition (CVPR)*, 2016.

[4] Vassileios Balntas, Shuda Li, and Victor Prisacariu, "Relocnet: Continuous metric learning relocalisation using neural nets," in *The European Conference on Computer Vision (ECCV)*, September 2018.

[5] L. Meng, J. Chen, F. Tung, J. J. Little, J. Valentin, and C. W. de Silva, "Backtracking regression forests for accurate camera relocalization," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, Sep. 2017, pp. 6886–6893.

[6] Johannes L. Schönberger, Marc Pollefeys, Andreas Geiger, and Torsten Sattler, "Semantic visual localization," in *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, June 2018.

[7] Alex Kendall and Roberto Cipolla, "Geometric loss functions for camera pose regression with deep learning," *CoRR*, 2017.

[8] Torsten Sattler, Qunjie Zhou, Marc Pollefeys, and Laura Leal-Taixé, "Understanding the limitations of cnn-based absolute camera pose regression," *CoRR*, 2019.

[9] J. O'Keefe and J. Dostrovsky, "The hippocampus as a spatial map. preliminary evidence from unit activity in the freely-moving rat," *Brain Research*, vol. 34, no. 1, pp. 171 – 175, 1971.

[10] Jeffrey Taube, Robert Muller, and James Ranck, Jr, "Head-direction cells recorded from the postsubiculum in freely moving rats. i. description and quantitative analysis," *The Journal of neuroscience*, vol. 10, pp. 420–35, 03 1990.

[11] M. J. Milford, G. F. Wyeth, and D. Prasser, "Ratslam: a hippocampal model for simultaneous localization and mapping," in *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004*, April 2004, vol. 1, pp. 403–408 Vol.1.

[12] Mathias Franzius, Henning Sprekeler, and Laurenz Wiskott, "Slowness and Sparseness Lead to Place, Head-Direction, and Spatial-View Cells," *PLoS Computational Biology*, vol. 3, no. 8, pp. 1–18, 2007.

[13] Benjamin Metka, Mathias Franzius, and Ute Bauer-Wersing, "Outdoor self-localization of a mobile robot using slow feature analysis," in *Neural Information Processing - 20th International Conference, ICONIP 2013. Proceedings, Part I*. 2013, vol. 8226, pp. 249–256, Springer.

[14] Laurenz Wiskott and Terrence Sejnowski, "Slow Feature Analysis: Unsupervised Learning of Invariances," *Neural Computation*, vol. 14, no. 4, pp. 715–770, 2002.

[15] Benjamin Metka, Mathias Franzius, and Ute Bauer-Wersing, "Bio-inspired visual self-localization in real world scenarios using slow feature analysis," *PLOS ONE*, vol. 13, no. 9, pp. 1–18, 09 2018.

[16] Jakob Engel, Thomas Schöps, and Daniel Cremers, "LSD-SLAM: Large-Scale Direct Monocular SLAM," in *Computer Vision - ECCV 2014, Proceedings, Part II*, 2014, pp. 834–849.

[17] R Mur-Artal, J M M Montiel, and J D Tardós, "ORB-SLAM: A Versatile and Accurate Monocular SLAM System," *IEEE Transactions on Robotics*, vol. 31, no. 5, pp. 1147–1163, oct 2015.

[18] Muhammad Haris, Mathias Franzius, and Ute Bauer-Wersing, "Robust outdoor self-localization in changing environments," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2019)*. 2019, IEEE.

[19] Benjamin Metka, Mathias Franzius, and Ute Bauer-Wersing, "Efficient navigation using slow feature gradients," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems, IROS 2017*, 2017, pp. 1311–1316.

[20] Muhammad Haris, Mathias Franzius, and Ute Bauer-Wersing, "Robot navigation on slow feature gradients," in *Neural Information Processing*. 2018, pp. 143–154, Springer International Publishing.

[21] Nils Einecke, Keiji Muro, Jörg Deigmöller, and Mathias Franzius, "Working area mapping with an autonomous lawn mower," in *Conference on Field and Service Robotics*. September 2017, Springer.

[22] David G. Lowe, "Distinctive image features from scale-invariant keypoints," *International Journal of Computer Vision*, vol. 60, pp. 91–110, 2004.