SELF-LOCALIZATION ON TEXTURE STATISTICS

Sven Eberhardt　　*Christoph Zetzsche*

Cognitive Neuroinformatics
University of Bremen
Enrique-Schmidt-Strasse 5, 28359 Bremen, Germany

ABSTRACT

The ability to localize ourselves in the outdoor world based on visual input even in absence of prior positional information is an important skill of our daily lives that comes naturally to us. However, the underlying mechanisms of this ability are poorly understood. Here, we show how simple texture statistics can be sufficient to provide a strong prior for the self-localization tasks. We find that statistics of common outdoor features such as tree density, foliage type or road structure provide a stronger cue for self-localization than the matching and recognition of less common landmarks such as lamp posts. We encourage the use of such common feature vectors as priors for self-localization systems and hypothesize that humans may use similar priors to assess the location from an unknown image.

Index Terms— localization, image features, vision, classification

1. INTRODUCTION

When we wander around the world as part of our daily lives, we almost always know with high reliability where we are located. But the naturalness of this accomplishment conceals its complexity. To achieve this feat, humans integrate information from various sources, e.g., from the vestibular system to perform dead-reckoning, but also from vision, audition and proprioception.

Vision is of particular importance for localization, as becomes evident in cases where no reliable prior information about past location is available, e.g. if we need to find our way home after getting lost. A remarkable test on this is the popular game GeoGuessr[1], in which participants are placed in a random location in Google Street View[2] and need to localize themselves on a world map to score points.

How can humans solve this task? The default answer would be that they achieve this by the same visual system so successfully used for other vision tasks like, for example, object recognition. However, we have evidence that the optimal features for self-localization differ from those suited for other tasks such as object recognition due to different invariance requirements. For example, features used in object recognition need to be invariant to scale and perspective projections, while for self-localization these variations provide valuable information [1]. These investigations gave us first hints on the type of visual features suited for localization but it still far from clear which features can provide the optimal invariance properties for different views from the same localization while still being discriminative to different locations.

Good candidate features for localization may be landmark-based, i.e. match to sparsely distributed, highly discriminative objects such as the Eiffel tower to determine if you are in Paris. Another option would be to use more generic, statistics-based features such as the density of trees, foliage, types of house façades and similar attributes and hope that any combination of such statistics is sufficient to classify where an image was taken.

This research paper aims at answering which features are best suited for the self-localization task and how they are distributed.

2. RELATED WORK

In computer vision for mobile robots, localization is most prominently implemented in simultaneous localization and mapping (SLAM) [2] contexts, where salient features are matched between successive camera frames to calculate geometric relations between the robot movement and its surrounding environment. Due to the small scale of movement in which these robots operate, they typically work well by matching very simple image features such as SIFT [3] descriptors or normalized image patches [4]. However, the context of this paper is closer to the ‘lost robot problem’ [5], where a mobile agent is deprived of any prior knowledge of its whereabouts and needs to determine its position within some previously recorded terrain.

The problem of localization from only one image has been investigated for the special case of urban locations e.g. by [6]. Later approaches by [7], [8] and [9] extended the approach to work on a large number of locations. All these approaches typically operate by recognition of house façades.

[1] http://geoguessr.com/

[2] https://google.com/streetview

978-1-4799-5751-4/14/$31.00 ©2014 IEEE

Doearsch et al. looked at the more general problem of discriminating between two cities [10]. They determined suitable image features by comparing a large set of Google Street View images and found that small features of façades such as door or window decorations are most discriminate to determine the city.

Despite relatively poor image quality with distorted edges and Google watermarks, the sheer mass of available data has made Google Street View imagery an increasingly popular source for image datasets and has been used to test algorithms for self-localization [8, 11, 12, 13, 14, 15], 3D map reconstruction [16, 17, 18], text recognition [19, 20, 21] and image segmentation [22, 23, 24]. Typically, the datasets derived from Google Street View show a strong bias towards inner city locations, either because researchers explicitly select such city locations or due to the denser sampling of Google Street View within cities.

We are interested in general features for localization which are not specific to façades, doors, or house silhouettes. We have established an appropriate localization dataset based on Google Street View imagery [25]. With this set we tested a variety of typical features used for object detection and found, somewhat surprisingly, that they are all outperformed by simple global texture statistics called *Textons* by Malik et al.[26] However, from our investigations it became not quite clear what these texture statistics exactly encode with respect to the localization problem. Here we approach this question by evaluating which part of the Texton descriptor contributes to localization performance and to which image features they correspond.

3. METHODS

3.1. Localization dataset

The choice of datasets has a strong impact on model performance results. Critical studies have shown that many datasets sampled from image databases from the internet often suffer from biases introduced by photographer's preference to center objects of interest or prefer certain picture layouts [27]. We therefore decided to sample images from the public API of Google Street View for your study. Unlike other datasets like e.g. geotagged flickr images, Google Street View imagery provides a huge amount of input, which does not have any bias towards frequently pictured points of intereset. Furthermore, it has been recorded in a standardized manner from a large number of countries worldwide.

Fig. 1. Sample images from the Google Street View localization dataset.

Three localization datasets for localization tasks on different scales have been sampled from Google Street View similar to the method used in [25]. Three sampling regions have been defined to create datasets for different scales. The smallest scale encompasses the city center of Berlin for the *city* dataset, a medium scale includes mainland France for the *country* dataset and the largest scale contains imagery from the whole world (where Street View is available) for the *world* dataset. Each dataset consists of 204 random locations with available Street View data. At each location, 36 grayscale pictures are sampled at a 512x512 pixel resolution by varying the yaw rotation in steps of 10°. The recorded field of view is 90° both horizontally and vertically. Sample images are shown in figure 1.

3.2. Models

For every input image, a Texton output vector is computed using the method described by [26]. Texton features are defined by first convolving an input image with number of linear edge detection filters in several scales and orientations. For each pixel in the resulting image, the responses of all filters are clustered using k-means clustering. The resulting cluster assignments are summed up over the whole image, producing a Texton histogram, which can then be fed into a classifier.

We use a freely available Texton implementation for MATLAB [28][3].

Several control models normally used in an object recognition context are run on the datasets to compare performance. The models used are *GIST* by Oliva et al.[29], *Spatial Pyramids* by Lazebnik et al.[30] and *HMax* by [31]. The implementation of the control models is the same as described in [25].

3.3. Classification

To assess feature quality for the localization task, we evaluate performance as percent correct labels in one-versus-all classification, where all images taken from one location (or point of view) are samples of one class. Ten training samples are provided for each location and the rest is used for testing. We do linear regression with leave-one-out cross-validation. Resulting performances and standard deviations are measured by repeating the test 15 times with different random splits of test and training sets.

All classifications are performed using the GURL classification package for MATLAB [32][4].

[3] http://www.eecs.berkeley.edu/Research/Projects/CS/vision/bsds/code/Textons/

[4] https://github.com/LCSL/GURLS/wiki

	city	country	world
gist	40.8 ± 0.5	41.3 ± 0.5	45.9 ± 0.5
hmax	44.4 ± 0.7	53.4 ± 0.3	64.4 ± 0.6
s.pyramid	61.7 ± 0.8	68.0 ± 1.3	75.6 ± 0.9
texton	69.7 ± 0.5	76.2 ± 0.8	76.6 ± 0.5
pixel	0.8 ± 0.2	0.7 ± 0.1	3.0 ± 1.4
lumhist	0.9 ± 0.0	0.9 ± 0.0	0.9 ± 0.1

Table 1. Classifier performances in percent correct for Textons and control models on all three Street View datasets. Data adapted from [25].

4. RESULTS

Classification performances on Texton features as well as comparison to the control models, a raw pixel descriptor and a simple luminance histogram is found in table 1. Performance on Textons is higher than on any other of the tested feature descriptors. We also show that the result cannot be solved trivially on raw pixels, as the performances both on a raw pixel descriptor as well as on luminance histogram is near chance level.

We refine this result by searching for optimal parameter choices to the feature set generation. Figure 2A shows the effect of cluster count on classification performance. Saturation is reached between 600 and 800 features for all datasets, which is a much larger cluster count than that used originally by [26].

Figure 2B shows the effect of controlling filter size by changing the spatial width σ. Performance is slightly affected and favors smaller filter sizes than those proposed by [26] for two of the datasets.

The graphs in figure 3 illustrate how a variation of yaw angle of a view influences the representation in feature space. The Texton representation shows little within-class variation caused by the rotation (variance within red/green lines) compared to between-class variance (variance between red and green line). In the HMax control model, discriminability between the two sample classes is still given, but the classes are much closer and samples within a class span a much larger range within the feature space, which means that a classifier will have a harder time discriminating between a large number of classes and yield a higher error rate. In a reference run on raw pixels, classification performance breaks down as inter-class representations between the two locations overlap in raw pixel feature space.

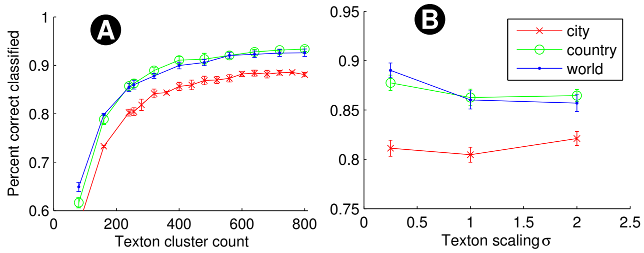

Fig. 2. Effect of Texton cluster count (A) and filter size (B) on classification performance in percent correct.

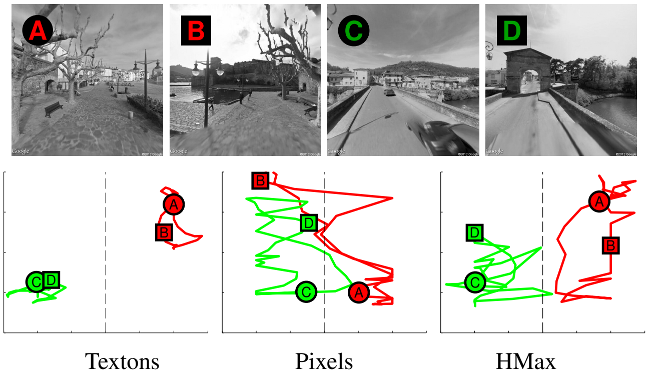

Fig. 3. Visualization of variance in several feature spaces between and within classes. The red and green line show data points from two locations. The horizontal axis shows values assigned by the linear classifier and the dashed line marks the separation plane between these two classes. The vertical axis shows feature values reduced to a single dimension by multidimensional scaling (MDS).

Next we look at the image features picked up by the Textons that contribute to the strong performance discriminating between locations. First we assign a score to each element of the Texton vector (i.e.: Each texture cluster) by summing the absolute value of the weight given to this entry by the classifier over all tested locations. The pictures in figure 4 show locations where the score of the assigned Textons are in the top .97 quantile of all Texton scores. Textons do not pick on particular landmarks such as the lamp post in picture 4A, but instead match to the surrounding grass, water and foliage areas (see figure 4, second row). Despite not matching to salient features, there is a noticeable difference in density of the matched areas between the two location while it is somewhat similar between the two images taken from the same location.

In contrast, the HMax output feature does recognize the lamp post (figure 4A, third row). However, the post is not discriminative for the location as it is absent when looking in a different direction (figure 4B) and there are lamp posts in other locations (figure 4C), which may be a source of missclassification on HMax-features.

5. DISCUSSION

We have investigated which visual features are suited for one-shot image based localization across a wide variety of environments. Our results show that a set of very fine-grained,

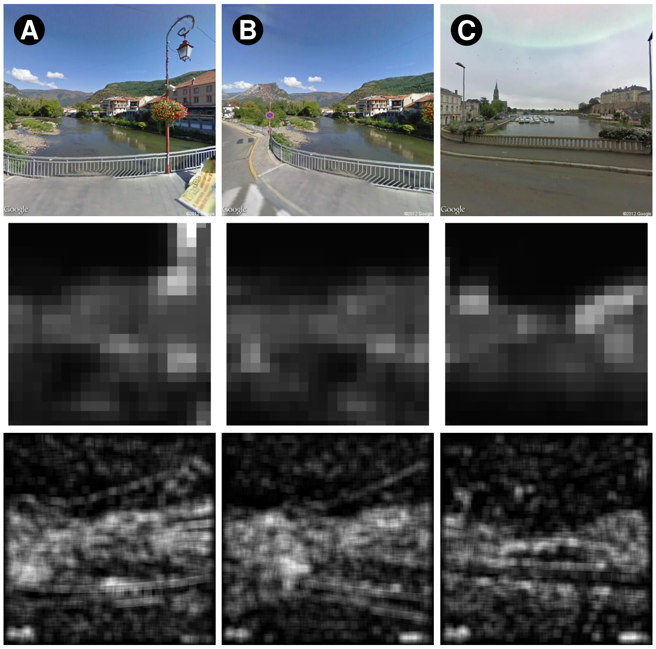

Fig. 4. Pictures A and B are taken from the same location at different angles (same class), while picture C is taken from a different location but shows a contentually comparable scenery. Second row shows corresponding visualizations of top-scoring Textons by classifier weights. Third row show the matching of a sample HMax feature for comparison.

local texture statistics can be representative of a certain location and this effect is stable across a large range of scales ranging from localization within a city to classification of locations around the planet.

How can such low-level features be so stable for a location? One would assume that salient landmarks such as unique buildings provide the best cue for localization. Such landmarks would be encoded well using high-level features of an object detector. However, for any given location around such a landmark, the object would only appear at maximum in half of the views. Therefore, a possible explanation for the good performance of texture statistics is that they pick up on characteristic of common objects like certain trees, foliage, ground structure, etc. which typically appear multiple times in every view of a location, so changes due to individual objects appearing and disappearing from the view are averaged out.

Despite strong classification performance, it would be presumptuous to claim that simple statistics on textures alone should be the sole basis for self-localization. However, in combination with other information they can provide a quite powerful contribution. For example, they could be used in conjunction with additional data coming e.g. from high-level knowledge related objects detected in the scene. And they can certainly work as a prior to improve reliability of other algorithms, for instance they might guide context in SLAM applications. On a more general level, they can be seen as a special form of ‘untangling’ effect, in which locations as categories are untangled from the influence of irrelevant variations. The strong separation seen in figure 3A hints at such an effect. It has been hypothesized by DiCarlo et al.[33] that the human brain, as part of learning to use its visual system, aims to build task-specific untangled features spaces. Textons may thus represent suitable features for a location-specific feature space.

Acknowledgments. This work was supported by DFG, SFB/TR8 Spatial Cognition, project A5-[ActionSpace].

6. REFERENCES

[1] Sven Eberhardt, Tobias Kluth, Christoph Zetzsche, and Kerstin Schill, “From pattern recognition to place identification,” in *Spatial cognition, international workshop on place-related knowledge acquisition research*, 2012, pp. 39–44.

[2] John J Leonard and Hugh F Durrant-Whyte, “Simultaneous map building and localization for an autonomous mobile robot,” in *Intelligent Robots and Systems. 1991 IEEE/RSJ International Workshop on*, 1991, pp. 1442–1447.

[3] DG Lowe, “Object recognition from local scale-invariant features,” in *Computer vision. 1999 IEEE seventh international conference on*, 1999, vol. 2, pp. 1150–1157.

[4] Javier Civera, Oscar G Grasa, Andrew J Davison, and JMM Montiel, “1-Point RANSAC for extended Kalman filtering: Application to real-time structure from motion and visual odometry,” *Journal of Field Robotics*, vol. 27, no. 5, pp. 609–631, 2010.

[5] Uwe Gerecke and Noel Sharkey, “Quick and dirty localization for a lost robot,” in *Computational Intelligence in Robotics and Automation. 1999 IEEE International Symposium on*, 1999, pp. 262–267.

[6] DP Robertson and R Cipolla, “An Image-Based System for Urban Navigation.,” *BMVC*, 2004.

[7] G Schindler, M Brown, and R Szeliski, “City-scale location recognition,” *Computer Vision and Pattern Recognition. 2007 IEEE Conference on*, 2007.

[8] Jan Knopp, Josef Sivic, and Tomas Pajdla, “Avoiding confusing features in place recognition,” in *Computer Vision - ECCV*, 2010, pp. 748–761.

[9] DM Chen, G Baatz, and K Koser, “City-scale landmark identification on mobile devices,” *Computer Vision and Pattern Recognition. IEEE Conference on*, 2011.

[10] C Doersch, S Singh, and A Gupta, "What makes Paris look like Paris?," *ACM Transactions on Graphics*, 2012.

[11] Minwoo Park, Robert T Collins, and Yanxi Liu, "Beyond GPS : Determining the Camera Viewing Direction of A Geotagged Image," in *Proceedings of the international conference on Multimedia*, 2010, pp. 631–634.

[12] Georg Schroth, Robert Huitl, and David Chen, "Mobile Visual Location Recognition," *Signal Processing Magazine, IEEE*, vol. 28, no. 4, pp. 77–89, 2011.

[13] Georg Schroth, A. Al-Nuaimi, R. Huitl, F. Schweiger, and E. Steinbach, "Rapid image retrieval for mobile location recognition.," in *Acoustics, Speech and Signal Processing. 2011 IEEE International Conference on*, 2011, pp. 2320–2323.

[14] Felix X Yu, Rongrong Ji, and Shih-Fu Chang, "Active Query Sensing for Mobile Location Search," in *Proceedings of the 19th ACM international conference on Multimedia*, 2011, pp. 3–12.

[15] Amir Roshan Zamir and Mubarak Shah, "Accurate Image Localization Based on Google Maps Street View," in *Computer Vision, ECCV*, 2010, pp. 255–268.

[16] Michal Jancosek, Alexander Shekhovtsov, and Tomas Pajdla, "Scalable multi-view stereo," in *Computer Vision Workshops. 2009 IEEE 12th International Conference on*. Sept. 2009, pp. 1526–1533, Ieee.

[17] Taehee Lee, "Robust 3D street-view reconstruction using sky motion estimation," *Computer Vision Workshops, ICCV, 2009 IEEE 12th International Conference on*, pp. 1840–1847, Sept. 2009.

[18] Branislav Micusik and Jana Kosecka, "Piecewise Planar City 3D Modeling from Street View Panoramic Sequences," in *Computer Vision and Pattern Recognition. 2009 IEEE Conference on*, 2009, pp. 2906–2912.

[19] James Lintern, "Recognizing Text in Google Street View Images," *Statistics*, vol. 6, 2008.

[20] Yuval Netzer, Tao Wang, Adam Coates, Alessandro Bissacco, Bo Wu, and Andrew Y Ng, "Reading Digits in Natural Images with Unsupervised Feature Learning," in *NIPS Workshop on Deep Learning and Unsupervised Feature Learning*, 2011.

[21] Kai Wang and Serge Belongie, "Word Spotting in the Wild," in *Computer Vision, ECCV*, 2010, pp. 591–604.

[22] Jianxiong Xiao, Clear Water Bay, and Hong Kong, "Multiple View Semantic Segmentation for Street View Images," in *Computer Vision, 2009 IEEE 12th International Conference on*, 2009, number Iccv, pp. 686–693.

[23] Chenxi Zhang, Liang Wang, and Ruigang Yang, "Semantic segmentation of urban scenes using dense depth maps," in *Computer Vision–ECCV 2010*, 2010, pp. 708–721.

[24] Honghui Zhang, Jianxiong Xiao, and Long Quan, "Supervised Label Transfer for Semantic Segmentation of Street Scenes," in *Computer Vision-ECCV 2010*, 2010, pp. 561–574.

[25] Sven Eberhardt and Christoph Zetzsche, "Low-level global features for vision-based localization.," in *Proceedings of the KI 2013 Workshop on Visual and Spatial Cognition*, 2013, pp. 5–13.

[26] Jitendra Malik, Serge Belongie, Thomas Leung, and Jianbo Shi, "Contour and texture analysis for image segmentation," *International journal of computer vision*, vol. 43, no. 1, pp. 7–27, 2001.

[27] J Ponce, T L Berg, M Everingham, D A Forsyth, M Hebert, S Lazebnik, M Marszalek, C Schmid, B C Russell, A Torralba, C K I Williams, J Zhang, and A Zisserman, *Dataset issues in object recognition*, Springer Berlin Heidelberg, 2006.

[28] T Leung and J Malik, "Representing and recognizing the visual appearance of materials using three-dimensional textons," *International Journal of Computer Vision*, 2001.

[29] Aude Oliva, Women Hospital, and Longwood Ave, "Modeling the shape of the scene: A holistic representation of the spatial envelope," *International journal of computer vision*, vol. 42, no. 3, pp. 145–175, 2001.

[30] S. Lazebnik, C. Schmid, and J. Ponce, "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories," in *Computer vision and pattern recognition, 2006 IEEE computer society conference on*, 2006, vol. 2, pp. 2169–2178.

[31] Thomas Serre, Aude Oliva, and Tomaso Poggio, "A feedforward architecture accounts for rapid categorization.," *Proceedings of the national academy of sciences*, vol. 104, no. 15, pp. 6424–6429, 2007.

[32] Andrea Tacchetti, Pavan K Mallapragada, Matteo Santoro, and Lorenzo Rosasco, "GURLS: a toolbox for large scale multiclass learning.," in *Big learning workshop at NIPS*, 2011.

[33] James J DiCarlo and David D Cox, "Untangling invariant object recognition.," *Trends in cognitive sciences*, vol. 11, no. 8, pp. 333–41, Aug. 2007.