# Indoor Localisation with Regression Networks and Place Cell Models

Jose Rivera-Rubio[1]
http://www.bicv.org
Ioannis Alexiou[2]

Anil A. Bharath[1]

[1] Biologically Inspired Computer Vision Group
Imperial College London
London, UK

[2] Computer Vision Research Group
Queen Mary University of London
London, UK

One of the key behaviours found in biological place cells (BPCs) is a rate-coding effect: a neuron's rate of firing decreases with distance from some landmark location. We used visual information from wearable and hand-held cameras in order to reproduce this rate-coding effect in artificial place cells (APCs). The accuracy of localisation using these APCs was evaluated using different visual descriptors and different place cell widths. Simple localisation using APCs was feasible by noting which APC yielded the maximum response. We also propose using joint position coding using a number of automatically defined APCs. Using both these approaches, we were able to demonstrate good self-localisation from very small images taken in indoor settings. Average localisation performance is favourable when compared with ground-truth and LSD-SLAM; even without the use of a motion model, errors using a single device were as low as 2 m for some journeys and corridors.

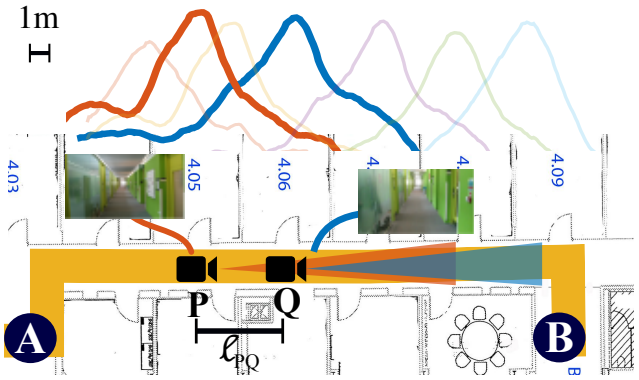


Figure 1: The concept of artificial place cells (APCs) for navigation. APC behaviour is produced by comparing FEVs from a series of frames to those at several landmark locations.

Given a series of video frames extracted from footage recorded during indoor navigation, we made use of the visual path concept [2, 3], to perform matching between locations of a physical environment being traversed and a database of views captured with two different devices, and at different moments in time. The similarity scores obtained from appearance-based comparison methods, applied between sequences of frames of a journey and these virtual landmarks, exhibit a behaviour (Fig. 1(a)) that is similar to those recorded in mammalian place cells.

We compared the accuracy of several patch-level descriptors for this purpose [3]. Patch descriptors were $L_2$ normalized, vector quantised and then frames were encoded into a 400-element Frame-Encoding Vector (FEV). For the comparison with a state-of-the-art SLAM method, we chose Engel's LSD-SLAM [1].

In order to model place cell behaviour, we need to map pairs of frames onto a scalar value that is analogous to a firing rate. One way to mimic the behaviour of BPCs within APCs is to introduce a *kernel* function that maps a pair of FEVs onto a positive scalar value. The following mapping between two vectors $\mathbf{v}_a$ and $\mathbf{v}_b$ maps the two FEVs onto a scalar value that takes a maximum when the two vectors are identical:

$$\kappa_{\chi^2}(\mathbf{v}_a, \mathbf{v}_b) = \sum_{j=1}^{400} \frac{v_a(j) \cdot v_b(j)}{v_a(j) + v_b(j)} \tag{1}$$

To model the response of a place cell to an image stimulus with respect to some reference location, $\ell_i$, the result of the $\kappa_{\chi^2}$ function is thresholded. The FEV, $\mathbf{v}_{r_i}$, is first constructed when the camera is at position $\ell_i$ from one or more reference journeys and calculating a supra-threshold response to some frame $\mathbf{v}_\ell$ acquired at location $\ell$. As $\ell$ is varied,

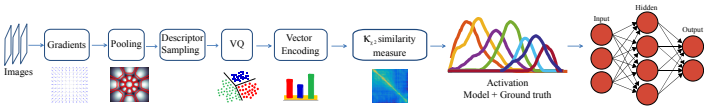


Figure 2: Overview of the training pipeline. The diagram of the neural network is merely illustrative, and does not represent the real GRNN architecture used.

$\kappa_{\chi^2}(\mathbf{v}_\ell, \mathbf{v}_{r_i})$ changes accordingly. Thresholds, $T_i$, relative to each response curve, are used to create a set of supra-thresholded response curves, $r_i(\ell)$:

$$r_i(\ell) = U(\kappa_{\chi^2}(\mathbf{v}_\ell, \mathbf{v}_{r_i}) - T_i) \cdot \kappa_{\chi^2}(\mathbf{v}_\ell, \mathbf{v}_{r_i}) \tag{2}$$

where $U(\cdot)$ represents the unit step function. Curves acquired by averaging responses from several journeys with respect to the same APC location may be referred to as an APC *tuning curve*.

Given a series of APC responses to visual cues of a person's location along some journey, there are two obvious ways of estimating location, $\ell$. The first is simply to use the APC which displays maximum activation (firing rate) as a rough indicator of where the person is. The second technique achieves more accurate localisation of a camera from its captured visual data by using the joint distribution $p(\mathbf{r}|\ell)$, of APC responses, $\mathbf{r}$ to infer location $\ell$ relative to some designated ground truth. We use a single index, $i$, to refer to the response, $r_i$, of a unique place cell. Given several active cells that are a subset of all place cells in a location, sub-APC localisation is possible using APC responses from previous journeys using empirical Bayes' techniques. For example, if three cells are active, the chain rule can be used to obtain successively refined estimates of $\ell$:

$$\begin{aligned} p(\ell|\mathbf{r}) &\propto p(r_3, r_4, r_5|\ell)p(\ell) \\ &\propto p(r_3|r_4, r_5, \ell) \times p(r_4|r_5, \ell) \times p(r_5|\ell) \times p(\ell) \end{aligned} \tag{3}$$

so that the responses of spatially close APCs can be used to infer sub-APC position. If the width of an APC is set to around 2 m, localisation of the order of tens of centimetres is plausible.

A Generalized Regression Neural Network (GRNN) was used to provide sub-APC position estimates, obviating the need to construct *ad-hoc* empirical estimators. This regression network consists of two-layers, and uses radial-basis functions. The responses from 16 place cells were input to the network, and ground truth of location within a section of corridor – up to 4 m long – used to train it as a regressor. In all experiments, dictionary generation was performed independently of the APC responses used in training the regression network.

In conclusion, this work demonstrates that computational models of place cells can provide effective estimates of camera location without relying on tracking or geometric models of the local environment. Such techniques, although simple, have achieved errors that range from tens of cm to few metres, matching and certainly complementing the more sophisticated position inference approaches used in computer vision.

[1] Jakob Engel, Thomas Schöps, and Daniel Cremers. LSD-SLAM: Large-scale direct monocular SLAM. In *Computer Vision–ECCV 2014*, pages 834–849. Springer, 2014.

[2] Yoshio Matsumoto, Masayuki Inaba, and Hirochika Inoue. Visual navigation using view-sequenced route representation. In *Robotics and Automation, 1996. Proceedings., 1996 IEEE International Conference on*, volume 1, pages 83–88. IEEE, 1996.

[3] Jose Rivera-Rubio, Ioannis Alexiou, Luke Dickens, Riccardo Secoli, Emil Lupu, and Anil A Bharath. Associating locations from wearable cameras. In *Proceedings of the British Machine Vision Conference*, 2014.