A Biologically Inspired Architecture for Visual Self-Location

Helio Perroni Filho* and Akihisa Ohya

Intelligent Robot Laboratory, University of Tsukuba,
1-1-1 Tennodai, Tsukuba, Japan
helio@roboken.iit.tsukuba.ac.jp, ohya@cs.tsukuba.ac.jp
http://www.roboken.iit.tsukuba.ac.jp/en/

Abstract. Self-location – recognizing one's surroundings and reliably keeping track of current position relative to a known environment – is a fundamental cognitive skill for entities biological and artificial alike. At a minimum, it requires the ability to match current sensory (mainly visual) inputs to memories of previously visited places, and to correlate perceptual changes to physical movement. Both tasks are complicated by variations such as light source changes and the presence of moving obstacles. This article presents the Difference Image Correspondence Hierarchy (DICH), a biologically inspired architecture for enabling self-location in mobile robots. Experiments demonstrate DICH works effectively despite varying environment conditions.

1 Introduction

Self-location (the ability to orient oneself relative to a known environment) is a fundamental cognitive skill [6]. It is also a requirement in mobile robotics applications such as *teach-replay navigation*, where a robot is first led through a route by a guide (the teach step), and must later autonomously retrace the original path (the replay step) [1]. Visual teach-replay is subject to largely the same challenges faced by living organisms, i.e., the need to account for variations in visual stimuli within a given environment. Therefore, a case can be made for self-location Biologically Inspired Cognitive Architectures (BICA's).

The Difference Image Correspondence Hierarchy (DICH) is a cognitive architecture designed to enable self-location in mobile robots equipped with a single monocular camera. Other BICA's capable of self-location have been developed, but often in the context of simulated environments that are very simplified [3], or allow state data to be acquired directly instead of inferred from sensor inputs [4]; lack of clear ways to specify goals (e.g., setting destinations) is also a common limitation [3, 4]. In contrast, DICH has been developed from the beginning to operate in real-world robots and environments, relies exclusively on visual data, and implements a learning model accommodating of goal-directed training and operation.

* This research work was supported by Conselho Nacional de Desenvolvimento Científico e Tecnológico (CNPq) (grant 201799/2012-0).

DICH is a continuation of previous work on visual search [7] and mobile robot navigation [8]. The next section details the architecture's design and fundamentals, followed by experiments that demonstrate its effectiveness. The article closes with a discussion on the results achievement and directions for further research.

2 Architecture

The DICH architecture was designed to work with a mobile robot equipped with a single front-mounted camera, under the teach-replay navigation scenario. During the teach step, the robot collects a video record of the trip, which is stored to long-term memory; during the replay step, camera inputs are used along with teach step data to update two working memory modules (the "similarity map" and the "shift map"), from which estimates of the robot's location along the route, and possible drift from it, can be computed. The next subsections describe each of the architecture's components in detail.

2.1 Difference Images

Mammal vision requires continuous stimulus change to function properly: even when the exterior world is static, fixational eye movements ensure the images falling on our retinas never remain the same for long [5]. DICH seeks to capture this principle through the use of *difference images* as its basic percept. Let I_t be the instantaneous visual input at time t for a mobile robot as it advances towards a landscape. Assuming reasonable conditions (i.e. a minimally stable environment, smooth movement, etc.) I_t will vary gradually as the robot advances. Such changes can be quantified by computing the difference image J_t such that:

$$\mathrm{J}_t = |\mathrm{I}_t - \mathrm{I}_{t-\delta}| \ , \ \delta = \arg\min_{\delta>0} \langle |\mathrm{I}_t - \mathrm{I}_{t-\delta}| \rangle \geq \tau \tag{1}$$

Where the subtraction operator is defined for images as pixel-wise euclidean distance (so e.g. subtracting RGB pixels will result in a single scalar value), $\langle \cdot \rangle$ is the average operator and τ a system parameter.

Given an appropriate gap $\delta > 0$ and absent of saturation artifacts, difference images are largely invariant to ambient brightness, providing a degree of normalization across illumination conditions; moreover, the amount of change will vary depending on whether the robot is moving, which can be used as a self-motion cue. Finding that appropriate gap is not trivial, though: too short a gap will result in a mostly empty image, too large and it may become impossible to relate differences to actual scene elements. However, by constraining δ so that every J_t will average a difference at least τ, a degree of consistency across difference images can be achieved, even in the face of changes to physical parameters such as the robot's speed and direction of movement.

2.2 Difference Image Pairing

A difference image implicitly denotes a location along a route as the pair of viewpoints from which the raw images used to compute it were taken. Therefore, DICH uses difference image matching to perform self-localization over the length of a route – if current visual input matches a stored snapshot, the robot should be at roughly the same place.

When animals look at their surroundings, their eyes dart among a small set of *salient points* (e.g., corners or edges), which seem to provide enough information for effective visual recognition. This behavior is modeled by Image Processing Algorithms (IPA's) that select small patches called algorithmic Regions-Of-Interest (aROI's) from input images [9]. Image matching can be performed for aROI's in place of whole images.

DICH uses aROI's for difference image matching as follows. Let J_i be the i^{th} teach difference image and J'_j the j^{th} replay difference image. A list of salient point image coordinates $p_{j,k} = (u, v)$ are selected from J'_j as points of maximum difference within square patches of side $2\alpha + 1$. The patches themselves are extracted as corresponding aROI's $\rho_{j,k}$. Now if J_i and J'_j are spatially related, then for each $\rho_{j,k}$, there must be a region of J_i not far from coordinates $p_{j,k}$ that is similar to it. Therefore, for each salient point $p_{j,k}$, a *neighborhood* $\phi_{i,j,k}$ of side $2\beta + 1$ is extracted from J_i, and the similarity between J_i and J'_j is defined as the sum of similarities between each $(\rho_{j,k}, \phi_{i,j,k})$ pair:

$$\kappa_s(\mathrm{J}_i, \mathrm{J}'_j) = \sum_k \max_{x,y} cos(\rho_{j,k}, \phi_{i,j,k}) \tag{2}$$

Where:

$$cos(\mathrm{A}, \mathrm{B}) = (\mathrm{A} \star \mathrm{B}) \circ (\mathrm{A}^{\circ 2} \star \mathbf{1}^{m_B \times n_B})^{\circ -\frac{1}{2}} \circ (\mathbf{1}^{m_A \times n_A} \star \mathrm{B}^{\circ 2})^{\circ -\frac{1}{2}} \tag{3}$$

Is the *sliding cosine similarity* between $\mathrm{A}^{m_A \times n_A}$ and $\mathrm{B}^{m_B \times n_B}$, for $\star$ the cross-correlation operator, $\circ$ and $^\circ$ respectively the Hadamard (i.e. element-wise) product and power operations [10], and $\mathbf{1}^{m \times n}$ a $m \times n$ matrix with all elements equal to 1. This is essentially a sliding version of the *cosine similarity* metric [2], where the cross-correlation between A and B computes the dot product for every translation of A over B, and the normalization factors are given by the other two formula terms. Figure 1a illustrates the process.

In animal self-localization, visual cues are combined with a sense of self-motion to produce reliable position estimates [6]. This is modeled in DICH by plotting all values of $\kappa_s(\mathrm{J}_i, \mathrm{J}'_j)$ over a range $(i_0, j_0) \leq (i, j) < (i_0 + h, j_0 + w)$ as a *similarity matrix*, and using linear regression to find a *trend* of high similarity values over it. This trend is represented by a line $l_j = (m_j, b_j)$.

Difference image pairing can then be defined as:

$$g(\mathrm{J}'_j) = \mathrm{J}_i \mid i = m_j j + b_j \tag{4}$$

Figure 1b illustrates pairing function estimation from a similarity map.

2.3 Shift Estimation

Difference image pairing estimates how far the robot advanced along the teach route. The deviation from the original route can be inferred by computing the *shift* between teach and replay images – a length of horizontal sliding of one image over the other, such that features of both are "matched" as well as possible. Because scenes may change between teach and replay trips (due e.g. to the presence of moving elements), it's not effective to compare images wholesale. Instead, given a matched pair ($\mathrm{J}_i = g(\mathrm{J}'_j), \mathrm{J}'_j$), *columns* of width γ are selected from teach image J_i one at a time for comparison to J'_j, and the resulting vectors summed into *shift vectors*:

$$\kappa_h(\mathrm{J}_i, \mathrm{J}'_j) = \sum_k (\mathbf{0}^n \parallel cos(\mathrm{J}_i[:, \gamma k : \gamma(k+1)], \mathrm{J}'_j)) \ll k \tag{5}$$

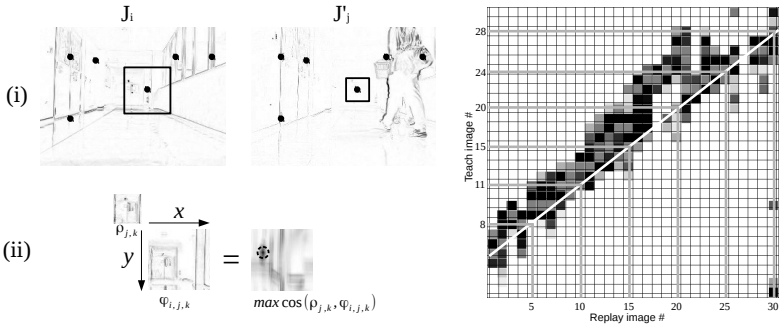
Where $\parallel$ and $\ll$ are the vector concatenation and left shift operators, and $\mathbf{0}^n$ the zero vector of dimension n (for n equal to the width of J'_j). Padding and shifting of individual column similarity vectors is necessary to properly align results (e.g., $cos(\mathrm{J}_i[:, 0 : \gamma], \mathrm{J}'_j)$) can only detect right shifts).

A shift vector describes shift likelihoods: the central value indicates the likelihood that no shift has taken place, while values prior to it represent the likelihood of a shift to the right, and values following, of a shift to the left. Concatenating successive shift vectors column-wise produces a *shift map*; see Figure 2b for an example. A hill climbing algorithm can then be used to find a route of high likelihood across it, which will indicate whether, and to which side, the robot is drifting from the original route.

3 Experiments

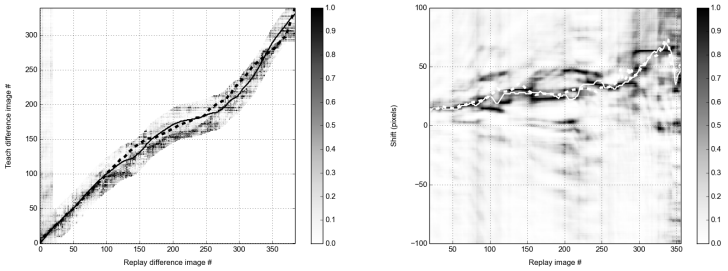
A mobile robot equipped with a top-mounted camera was used to record a test session composed of two trips – a reference or *teach step* trip, and a comparison or *replay step* trip – in a corridor of about $20m$ length. Teach and replay steps started from the same position and advanced in the same direction, but in the teach step, after starting close to the left wall, the robot slowly drifted right until stopping close to the right wall; whereas in the replay step the robot remained close to the left wall for the duration of the trip. The corridor was empty during the teach step, but $30s$ into the replay step three people come from behind the robot, staying on the right side of the field of view until walking away $20s$ later.

A batch implementation of DICH was developed and applied to the video records offline. System parameters were set to ($\tau = 15, \alpha = 25, \beta = 49, \gamma = 16, w = 20, h = 50$). In order to establish an initial location estimate over the whole route, the first similarity map is computed over the first w replay difference images and all teach difference images; after that, the estimate is iteratively updated by recomputing the similarity map for the h teach inputs closest to the latest pairing estimate and most recent w replay inputs. Ground truth data was computed manually by comparing the frames of teach and replay step video recordings.



(a) Difference image similarity (b) Similarity map

Fig. 1: Similarity map and trend computation. (a) Algorithmic Regions-of-Interest and neighborhoods are extracted at given salient points (i); sliding cosine similarity is computed for each (aROI, neighborhood) pair, and the maximum value (indicated here by the dashed circle) is taken as the similarity for that pair (ii). In this particular case, some salient points will produce weak responses, since they fell on an element not present in the teach difference image (a pedestrian who crossed in front of the robot during the teach step). However, this may still be compensated by the responses of the other points. (b) A similarity map represents at each cell the similarity between replay (horizontal axis) and teach (vertical axis) difference images. Darker shades of gray indicate higher similarity. The white line indicates the identified matching trend across the map. Gray lines indicate the estimated correspondence between replay and teach images.



(a) Similarity map (composite) (b) Shift map

Fig. 2: Test session results, displayed as (composite) similarity (a) and shift (b) maps. Darker shades of gray indicate higher correspondence / shift likelihood. Full lines over the maps represent estimated image correspondences / shifts, and dashed lines, ground truth data. Horizontal axis is replay image index; vertical axis is teach image index for the similarity map, and shift in pixels for the shift map (with positive values indicating left shift, and negative values, right). The displayed similarity map is actually a composite of several computations between h teach and w replay difference images, combined in a manner consistent with image indexes.

As shown in Figure 2, estimates and ground truth agree well in both similarity and shift maps; divergences arise occasionally, but always reverse later on. Both image pairing and shift estimation successfully coped with the variation in difference image change ratio, caused by the appearance of pedestrians in the replay step (this is the source of the slope change in the estimate curve in the middle of the similarity map). Results therefore indicate DICH was able to successfully estimate the robot's position along the route, as well as to identify sideways drift, in the image domain.

4 Conclusions

This article presented the Difference Image Correspondence Hierarchy (DICH), a biologically inspired cognitive architecture for enabling self-location in mobile robots. Experimental results show adequate performance under a range of input variations, suggesting there is merit in its premises. Currently the method's main weakness is the need to set parameters manually for optimal performance. However, research in the psychophysiology of vision may help determine reasonable defaults. The method's application in a robot navigation system (i.e. that would allow a mobile robot to drive itself) is also meant as a topic for future work.

References

1. Burschka, D., Hager, G.: Vision-based control of mobile robots. In: Robotics and Automation, 2001. Proceedings 2001 ICRA. IEEE International Conference on. vol. 2, pp. 1707–1713 (2001)
2. Dumais, S.T.: Latent semantic analysis. Annual review of information science and technology 38(1), 188–230 (2004)
3. Georgeon, O.L., Marshall, J.B., Manzotti, R.: Eca: An enactivist cognitive architecture based on sensorimotor modeling. Biologically Inspired Cognitive Architectures 6, 46–57 (2013)
4. Madl, T., Franklin, S., Chen, K., Trappl, R.: Spatial working memory in the lida cognitive architecture. In: Proceedings of the 12th international conference on cognitive modelling. pp. 384–390 (2013)
5. Martinez-Conde, S., Macknik, S.L.: Fixational eye movements across vertebrates: comparative dynamics, physiology, and perception. Journal of Vision 8(14), 28–28 (2008)
6. Moser, E.I., Kropff, E., Moser, M.B.: Place cells, grid cells, and the brain's spatial representation system. Annu. Rev. Neurosci. 31, 69–89 (2008)
7. Perroni Filho, H., De Souza, A.: On multichannel neurons, with an application to template search. Journal of Network and Innovative Computing 2(1), 10–21 (2014)
8. Perroni Filho, H., Ohya, A.: Mobile robot path drift estimation using visual streams. In: System Integration (SII), 2014 IEEE/SICE International Symposium on. pp. 192–197. IEEE (2014)
9. Privitera, C.M., Stark, L.W.: Algorithms for defining visual regions-of-interest: Comparison with eye fixations. Pattern Analysis and Machine Intelligence, IEEE Transactions on 22(9), 970–982 (2000)
10. Roger, H., Charles, R.J.: Topics in matrix analysis. Cambridge University Press (1994)