Instance-based Object Recognition with Simultaneous Pose Estimation Using Keypoint Maps and Neural Dynamics

Oliver Lomp[1], Kasim Terzić[2], Christian Faubel[1], J.M.H. du Buf[2] and Gregor Schöner[1]

[1] Institut für Neuroinformatik, Ruhr-Universität, Bochum, Germany
{oliver.lomp,christian.faubel,gregor.schoener}@ini.ruhr-uni-bochum.de
[2] Vision Laboratory (LARSyS), University of the Algarve, Faro, Portugal
{kterzic,dubuf}@ualg.pt

Abstract. We present a method for biologically-inspired object recognition with one-shot learning of object appearance. We use a computationally efficient model of V1 keypoints to select object parts with the highest information content and model their surroundings using simple colour features. This map-like representation is fed into a dynamical neural network which performs pose, scale and translation estimation of the object given a set of previously observed object views. We demonstrate the feasibility of our algorithm for cognitive robotic scenarios and evaluate classification performance on a dataset of household items.

Key words: Neural Dynamics, Biologically Inspired Keypoints, Vision, Object Recognition, Pose Estimation

1 Introduction

Object recognition is one of the central problems in vision and remains the focus of much research in Computer Vision, Artificial Intelligence and Computational Neuroscience. Vision for cognitive robotics poses additional constraints: (1) real-time performance is desired in order to support timely interaction with the environment; (2) processing power is often limited, requiring careful data selection and extensive optimisation; (3) fast and online learning in interactive scenarios is preferable to long training times.

In contrast to many computational architectures for cognitive robotics, we propose a neural architecture which uses dynamic neural fields for simultaneous object recognition and scale, translation and rotation estimation. It is capable of one-shot learning of object appearance based on single images and it can learn new objects online. We do not explore the real-time feasibility in the present work. However, the neural principles used throughout the paper lend themselves to parallel and thus fast implementation.

1.1 Related Work

Computationally efficient methods have been developed for a number of vision tasks in robotics, including pose-invariant object detection. Typically, invariant features are detected [1–4] and these are matched to stored templates using approximate methods such as RANSAC.

A number of biologically inspired methods have been proposed for object detection and recognition. Some of them expand the Neocognitron architecture [5], originally developed for character recognition, to recognise more general objects such as faces [6]. The HMAX model and its derivatives [7] are based on a simplified alternation of layers of simple and complex cells in order to extract features of increasing complexity, but they require an external classifier (usually an SVM) for final classification. Recently, deep convolutional networks have demonstrated excellent performance on a number of classification tasks, but at a considerable cost in terms of complexity and learning time [8]. All of these state-of-the-art approaches can deal with translation and scale change, but they do not address variation in pose, such as rotation and perspective effects. There is ample evidence that biological vision systems can recognise objects with unfamiliar poses, but that this process is considerably slower, hinting at a separate recurrent neural process [9]. We present a neural framework which addresses this problem.

Our work is based on map-seeking circuits of Arathorn [10], which were applied to object detection by Faubel and Schöner [11]. Instead of using global histograms as in [11], we apply biological keypoints for data selection and extract a set of localised colour features at keypoint locations. Keypoints play an important role in early attention. They indicate areas with large local complexity and exhibit excellent repeatability [4], which makes them useful for pose estimation. We reformulate the original algorithm from [11] so that it uses a consistent neural dynamic approach throughout. This reformulation makes it suitable for localised features, thus removing some ad-hoc parts of the original, global algorithm.

2 Method

2.1 Localised Colour Features Based on V1 Keypoints

We start by extracting multi-scale keypoints using the BIMP algorithm [4]. The image is first processed by a bank of complex Gabor filters representing cortical simple cells. The moduli of simple cell responses are used to model responses of complex cells. Another layer of cells computes spatial derivatives of the complex cell responses, which are combined with two inhibition schemes to obtain responses of end-stopped cells for detecting keypoints. The algorithm is applied at multiple scales by varying the wavelength of the Gabor filters which model simple cells. For a more detailed description of the algorithm we refer to [4].

At each keypoint location, we extract a colour histogram which represents a local neighbourhood with size proportional to filter scale. Each pixel in the neighbourhood is assigned the most similar of ten basic colours in the Lab colour

space. Then a Gaussian-weighted sum for each basic colour is computed over the local neighbourhood for each keypoint. The sum for each colour is stored in a 2D map at the keypoint location and then spatially smoothed with an isotropic Gaussian kernel. With ten basic colours, this gives a stack of ten 2D feature maps $I(x, y, c) \mapsto \mathbb{R}$, where x and y are subsampled image coordinates and c represents one of the basic colours. This process is applied at each keypoint scale s, resulting in a 4-dimensional feature vector $K(x, y, c, s) \mapsto \mathbb{R}$.

Localised colour histograms are fast and rotation-invariant, but relatively weak features. We use them here to highlight the importance of spatial configuration and our pose estimation algorithm. We plan to replace them by more powerful features, such as responses of HMAX-based C-cells [7].

2.2 Pose Estimation and Object Recognition

The main idea of map-seeking circuits [10] is that the pose and identity of an object in an input image can be estimated simultaneously by using a recurrent process. This process starts by assuming that all poses and identities are equally likely (although in principle, it is possible to bias certain classes based on domain priors or scene context). In each iteration, estimates are updated by a competitive process which adapts the relative weights of poses by estimating how well they match the current input given the estimates before the update. Over time, the weights converge to a state where only the correct pose and identity have non-zero weights. A version of this approach using dynamic neural fields provides good control over the convergence and enables coupling to online visual input [11].

We expand this earlier work in an architecture, shown in Fig. 1a, that consists of a pose estimation module and an object recognition module. Pose estimation entails a cascade of two-dimensional translation (shift), rotation, and scale that is processed concurrently. Pose is represented in dynamic neural fields (DNFs) and label information in a discrete variant, dynamic neural nodes. DNFs are patterns of activation, $u(\mathbf{x}, t)$, defined over a pose parameter or feature dimension, $\mathbf{x}$, that evolve as a dynamical system according to [12]

$$\tau \dot{u}(\mathbf{x}, t) = -u(\mathbf{x}, t) + h + s(\mathbf{x}, t) + \int (w(\mathbf{x} - \mathbf{x}') - \gamma) f(u(\mathbf{x}', t)) d\mathbf{x}' + \eta. \quad (1)$$

Here, τ is the time scale of the dynamics, $h < 0$ the resting level, $s(\mathbf{x}, t)$ external input; $w(\mathbf{x} - \mathbf{x}')$ the kernel of local excitatory interaction within the DNF, while $\gamma \geq 0$ represents the strength of global inhibitory interaction. The function $f(\cdot)$ is a sigmoid. Noise, η is added because the dynamics goes through instabilities and must escape reliably from unstable solutions. At appropriate values of γ the dynamics is selective, allowing only a single connected and bounded region of the DNF to become active at any given time.

Pose estimation happens in two cascaded DNFs. The first layer, $u_1(\mathbf{x}, t)$, forms an initial hypothesis, driven by the current pose estimate as input with weak global inhibitory interaction. The second layer evolves more slowly with

strong inhibitory interaction, and makes the final decision on the pose estimate. It receives input from the first layer, $s_2(\mathbf{x}, t) = \theta(g(u_1(\mathbf{x}, t)))$, where $\theta(u) = u$ for $u > 0$ and zero elsewhere. g is a spatial Gaussian filter. The current pose estimate $p(\mathbf{x}, t)$ is

$$p(\mathbf{x}, t) = c_{\text{mix}}(t) \cdot \theta(u_1(\mathbf{x}, t)) + (1 - c_{\text{mix}}(t)) \cdot f(u_2(\mathbf{x}, t)) \ , \tag{2}$$

where $c_{\text{mix}}(t) \in [0, 1]$ is the output of an additional, dimensionless DNF which is activated when $\int f(u_2(\mathbf{x}, t))d\mathbf{x}$ exceeds a threshold.

Object Identity is represented by associating labels with each object. In analogy with the pose representation, there are two layers of dynamic neural nodes, $u_{l,i}(t)$ ($i \in \{1, 2\}$) for each label l, governed by dynamics analogous to Eqn 1. Similar to self-excitation and global inhibition in the DNF, each node excites itself and inhibits all others. The current label estimate, $w_l(t)$, is calculated analogously to Eqn 2.

Matching Pose and Identity for Shifts We first describe the method for a single scale and for the shift estimate only. Inputs are three-dimensional functions $I(x, y, c) \mapsto \mathbb{R}$, where x, y are image coordinates and c is an additional feature dimension such as colour. $I(x, y, c)$ simply specifies how much of a colour c is perceived at a given location. For matching inputs to the stored views, we first apply the *inverse* of the current pose estimate $p(\delta_x, \delta_y, t)$, where $(\delta_x, \delta_y) = \mathbf{x}$. This is calculated as

$$I'(x, y, c) = \iint p(\delta_x, \delta_y, c) \cdot I(x + \delta_x, y + \delta_y, c)d\delta_x d\delta_y \ , \tag{3}$$

i.e., the cross-correlation of the current input and the current pose estimate. The matching value m_l with each memorised pattern $W_l(x, y, c)$ is also calculated using correlation, now taking all colours c into account:

$$m_l(t) = \iiint \hat{I}'(x, y, c) \cdot \hat{W}_l(x, y, c)dxdydc \ , \tag{4}$$

where $\hat{I}'$ and $\hat{W}_l$ are zero-mean and normalised versions of I' and W_l. For determining the current shift estimate, we calculate the superposition of the memorised patterns

$$W'(x, y, c) = \sum_l w_l(t) \cdot W_l(x, y, c) \tag{5}$$

given the current label estimates $w_l(t)$. Analogously to Eqn 4, we can then compute

$$s_1(\delta_x, \delta_y) = \iiint \hat{W}'(x, y, c) \cdot \hat{I}(x + \delta_x, y + \delta_y, c)dxdydc \ . \tag{6}$$

This is fed into Eqn 1 as input, forming a closed recurrent loop.

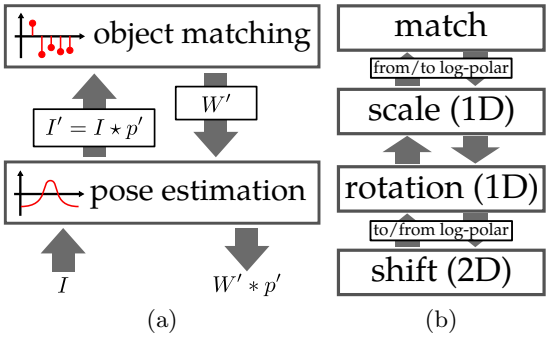

Fig. 1. (a) Interfaces between the *pose estimation* and *object matching* modules. Please refer to the text for an explanation of the symbols. (b) Log-polar transformations between pose estimation modules allow us to estimate scale and rotation.

Scale and Rotation can be estimated by cascading several transformation modules (see Fig. 1). To estimate rotation and scaling, the output of the shift estimation module is transformed to log-polar space, so that rotation and scaling are again shift operations (Fig. 1b).

Pattern Learning is supervised. During learning, Gaussian-shaped inputs are fed into the pose-DNFs. These inputs are centred around the correct pose parameters. The correct label is activated by a strong, localised input to the corresponding label node.

The weights of the pattern memory (W_l) are adapted by a linear dynamical system. This system is chosen so that at the fixed point, $W_l(x, y, c) = T(x, y, c)$, where T is the image used for training, but inversely transformed by the pose parameters in the DNFs.

2.3 Keypoint Scales

The feed-forward stream of the architecture contains keypoints on different scales. So far, we considered input from a single scale of the keypoint-stream. We can also control the scale of keypoints through a matching approach similar to the one presented above. Let $K(x, y, c, s) \mapsto \mathbb{R}$ be the maps of localised histograms for colour c at keypoint scales s.

We need to obtain a measure of the match for each scale. This can be determined analogously to how memory patterns are matched to the currently processed input pattern. Let $W'(x, y, c)$ be the superposition of memories, transformed according to the current pose estimates. Then the match value for scale s can be calculated by

$$m(s) = \iiint \hat{K}(x, y, c, s) \cdot \hat{W}'(x, y, c) dx dy dc \,, \tag{7}$$

Fig. 2. The four different poses used for our experiment, demonstrated for *Cookies*. The left-most image shows the object in the standard pose used for training. The remaining three views show the object in different orientations and positions.

where $\hat{K}$ and $\hat{W}'$ are again zero-mean and normalised versions of K and W'. Note that this matching value is a second input to the scale estimation field rather than having its own estimation process. In order to get the input pattern I for the recognition system, we again form a superposition of the different scales, weighted by the current scale estimate. The system then naturally converges to the keypoint scale that is most appropriate for classifying the given object.

3 Evaluation

We implemented the architecture using *cedar*, a software framework for neural dynamics [13]. For the evaluation, we use a subset (only the 120 images recorded in the center region out of the total of 300) of the images that were used to evaluate its predecessor [11, 14]. These images of 30 everyday objects have been recorded with a robotic scenario in mind. The camera looks at somewhat distant objects on a white tabletop. Of the nine different poses in the previous dataset, we chose the four located in the centre region of the image (see Fig. 2). As in the previous experiments, we cut out a subregion (a rectangle of 360×360 pixels in the centre of the image) of the original images (640×480 RGB).

3.1 Performance Evaluation

During the training phase, weights are learned by presenting each training image together with the pose information for a fixed duration. Images are presented only once. In the testing phase, the test images are presented to the system for recognition. The neural dynamics we use are stochastic. This may randomly lead to different outcomes on separate trials, thus we repeat the testing phase three times.

In both phases, the system is reset by lowering the resting level of the DNFs between the presentation of two images. This reset phase is considered complete when the difference between the minimal and maximal activation of the nodes in the second label layer falls below a fixed threshold. This removes residual information from previous trials without restarting the process.

The recognition is considered completed when the $f(u_{l,2})$ exceeds a certain threshold during the recognition phase and the change in $u_{l,1}$ falls below a threshold. A short grace period is given to account for possible changes in the decision.

System	Correct (%)
Proposed	68.0
LNBNN + SIFT	42.2
LNBNN + CH27	10.0
LNBNN + SIFTP10	40.0

(a) Comparison of recognition performance.

Pose Parameter	Average Error
Position (px)	23.8
Rotation (°)	69.3
Scale* (factor)	0.22

* Only evaluated on scaled images.

(b) Evaluation of pose estimation.

Table 1. Performance measurements. Refer to the text for an explanation.

After this period, the recognised label and pose are given by the location of the maximum activation in the layer two activation values. The recognised pose is read out in the same way. A reset is triggered and, once it is completed, a new recognition trial begins.

For comparison, we classified the original dataset using the state of the art LNBNN classifier [15] with three different kinds of features. The first approach uses a full range of SIFT features. The second approach (CH27) uses the same localised colour histograms we use for our approach, but with 27 colour bins. The third approach (SIFTP10) uses a randomly-selected set of ten SIFT prototypes and uses the Euclidean distance to each of them as a 10-dimensional feature vector.

3.2 Results

Results are shown in Table 1. Pose errors are reported as Euclidean distances between the recognised and the annotated pose.

As Table 1a shows, even with relatively weak features, we already obtain good recognition rates. Failures occur for objects with similar feature values. For example, *glue*, an object dominated by red and blue colours, was often confused with *blue_boxcutter*, *blue_tape* and *red_screwdriver*. Objects with low local complexity, such as *yellow_stapler* presented the biggest difficulty for our system.

4 Conclusions

We have presented a neurally-inspired model for object recognition with simultaneous pose estimation based on single views. We expanded and revised previous work that employed global features and non-neural processing steps such as histogram rotations. Our approach is entirely based on neural concepts, from neurally plausible local features to the neural-field-based shift, scale, and rotation estimation. Tests on 30 household items demonstrate the ability of our method to reliably recognise objects learned from single views.

Our implementation does not currently achieve speeds required for real-time processing. The current bottleneck is the calculation of the four-dimensional feature maps.

While our approach outperformed a state-of-the-art approach on our dataset, we only consider this a preliminary result. In future work, we aim to test our approach with stronger features on an established database.

We are currently working on stronger invariant features and integration with a scene representation system. We expect that stronger top-down guidance, which can be easily added to our framework, can significantly improve detection speed and results.

Acknowledgements This work was supported by the EU under the grant ICT-2009.2.1-270247 *NeuralDynamics* and the Portuguese FCT under the grant PEst-OE/EEI/LA0009/2011.

References

1. Lowe, D.G.: Distinctive image features from scale-invariant keypoints. IJCV **60** (2004) 91–110
2. Bay, H., Ess, A., Tuytelaars, T., Van Gool, L.: Speeded-up robust features (SURF). CVIU **110** (2008) 346–359
3. Rublee, E., Rabaud, V., Konolige, K., Bradski, G.: ORB: An efficient alternative to SIFT or SURF. In: ICCV, Barcelona (2011) 2564–2571
4. Terzić, K., Rodrigues, J., du Buf, J.: Fast cortical keypoints for real-time object recognition. In: ICIP, Melbourne (2013) 3372–3376
5. Fukushima, K.: Neocognitron for handwritten digit recognition. Neurocomputing **51** (2003) 161–180
6. Do Huu, N., Paquier, W., Chatila, R.: Combining structural descriptions and image-based representations for image, object, and scene recognition. In: IJCAI. (2005) 1452–1457
7. Serre, T., Wolf, L., Bileschi, S., Riesenhuber, M., Poggio, T.: Object recognition with cortex-like mechanisms. IEEE T-PAMI **29** (2007) 411–426
8. Schmidhuber, J.: Multi-column deep neural networks for image classification. In: Proceedings of the 2012 IEEE Conference on Computer Vision and Pattern Recognition (CVPR). (2012) 3642–3649
9. Ullman, S.: High-Level Vision: Object Recognition and Visual Cognition. The MIT Press (1996)
10. Arathorn, D.: Computation in the higher visual cortices: Map-seeking circuit theory and application to machine vision. In: AIPR. (2004) 73–78
11. Faubel, C., Schöner, G.: A neuro-dynamic architecture for one shot learning of objects that uses both bottom-up recognition and top-down prediction. In: IROS, IEEE Press (2009)
12. Amari, S.: Dynamics of pattern formation in lateral-inhibition type neural fields. Biological Cybernetics **27** (1977) 77–87
13. Lomp, O., Zibner, S.K.U., Richter, M., Rañó, I.n., Schöner, G.: A Software Framework for Cognition, Embodiment, Dynamics, and Autonomy in Robotics: Cedar. In: ICANN. (2013) 475–482
14. Faubel, C., Schöner, G.: Learning to recognize objects on the fly: a neurally based dynamic field approach. Neural networks **21** (2008) 562–576
15. McCann, S., Lowe, D.: Local naive bayes nearest neighbor for image classification. In: CVPR, Providence (2012) 3650–3656