Visual Objects Representation by Features Network

TITO SILVA, AGOSTINHO ROSA
Laseeb - Systems e Robotics Institute – IST
Technical University of Lisbon
Av. Rovisco Pais 1, Torre Norte 6.21, 1049-001 Lisboa
Portugal

Abstract: This paper presents a new paradigm on 2-D visual structures representation. It consists on a model, which has the property of storing object internal relations independently to their location or orientation, as well as storing content-addressable information about visual images under observation. The model is implemented as a Neural Network, which learns both its units' number and its units' weights in an unsupervised manner. Application example is provided for illustration of the concept.

Keywords: Image Processing, Unsupervised Learning, Visual Cortex, Visual Concepts Representation, Neural Networks, Object Recognition

1. Introduction

Related work

Several authors have lately used cortical models to achieve visual recognition. Rajesh Rao and Dana Ballard use a recognition model that uses the typical retroaction properties of Neuronal Networks [9]. That model uses an extend Kalman Filter [1,10] based on the Minimum Description Length [11], to obtain a hierarchical network model in order to achieve visual recognition.

This model is based on Visual Cortex neurophysiological data. Image under observation is divided into partially overlapping Receptive Fields. Information is processed in several layers. Several modules compose each layer. Sets of modules are combined and processed together in some module of the next layer. Superior layers supply predictions to the lower ones, and the prediction error enables learning.

Although this model is able to store structural relations and even to reconstruct partially occluded objects, it is not invariant to objects location, orientation or scale; the objects must be placed in exactly the same position and orientation as they were learned, in order to attempt recognition.

Furthermore, this network associates the image under observation to a pre-memorised object, not showing ability to learn structural rules that could enable predictions even on objects never seen before.

Biological inspiration of this work

Some cells in Infero-Temporal area have been found to detect particular shapes, colours or textures. These features are usually common to several different objects, thus constituting a building block of these objects [12,2]. An object will be recognised when a particular combination of those building blocks is detected. Other studies suggest that lateral connections among Visual Cortex cells will generate functionally coherent cell combinations [7].

There is also evidence that Infero-Temporal Cortex cells that have their Receptive Fields capturing parts of the same object will synchronise.

Features Network

In our approach we suggest this synchronisation ability may be due to some internal mechanism of structural coherence detection.

We worked with sequences of real images, with no previous learning. The basic structure of our learning model is a Neural Network, with the objectives of firstly to learn and recognise features of the observing data (Interpretation Layer), and secondly to learn and recognise structural coherences among those features (Coherence Layer). The neurones required for both learning

processes are created, as they are needed. The number of neurons belonging to the Network emerges from the unsupervised learning mechanism.

2. The Representation Model

The Representation model was organised in a Neural Network with two layers.

The Interpretation Layer

The first layer learns typical visual features of the world, and is designated by *Interpretation Layer*. It starts with no knowledge, containing no neuron. Its objective is to make feature extraction of the observing data, as well as capturing the orientation of extracted features. This layer evolves both in the number of neurons it contains (extractable features) and in the weights associated to each neuron, gaining the ability to recognise the relevant features that may exist within any real image. The weights associated to each neuron describe the visual pattern that the neuron is able to recognise.

The input of the Interpretation Layer is the image observed, which may for example be a frame from a video sequence. That image is captured in small circular overlapping units, the Receptive Fields. When observing the image, each Receptive Field will thus contain a small part of the observed image, which will be called in the next text as *Receptive Field Observation* or just *Observation* for short.

We define an *Interpretation* as a pair (*Index, Orientation*). *Index* represents the index of the Interpretation Layer neuron that best matches the Observation, regardless of the orientation of the latter. *Orientation* represents the orientation of the Observation towards the orientation of the neuron inner representation of the feature, which is considered the origin (0º) for determining the Observation orientation.

To achieve both unsupervised neurons formation and unsupervised weights learning, the Interpretation Layer uses a simple method described below:

Define W as the *Feature Base*. W is a matrix composed by *c* vectors, such that *c* is the number of neurons in the Interpretation Layer. Each of those vectors represents a particular feature, and is equivalent to the respective neuron weights. We start with c=0.

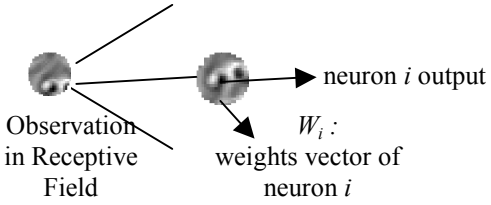

Figure 1. Feature extracting process of the Interpretation Layer. Interpretation Layer tries to recognise the input (which is the image falling inside a Receptive Field) with some of the features stored in its Feature Base. Each neuron has a weights vector W_i that describes the particular feature it is specialised in; the neuron output (disregarding Orientation detection) is the inner product between that vector and the input.

Interpretation Layer recognises features independently to their orientation. For that purpose, it considers 36 possible orientations, each at 10º distance from the nearest one[1]. W is expanded to a three-dimensional grid that includes any possible orientation of any stored feature: designate this expansion by W^{36}. This grid is thus able to recognise a feature even if presented with an orientation never observed before.

The maximum of the inner products between any possible orientation of a stored feature and the Observation is the output of the corresponding neuron.

The Observation is recognised as the feature corresponding to the neuron with greatest output. The Observation Orientation is obtained by recalling the Orientation that provided the recognising neuron output.

Interpretation Layer iterates this process of Feature and Orientation recognition of the observations inside each Receptive Field, through all Receptive Fields of an image under study.

The Interpretation Space ζ is the set of all possible interpretations recognisable by the Interpretation Layer. Every Receptive Field will be recognised in the Interpretation Layer as an element of ζ (the feature description, rotated by some orientation angle, that best represents the observation). Therefore, $\#\zeta = c \times 36$.

The output of the Interpretation Layer is a translation of the image to a compacted format (a set of elements of ζ, as well as their respective locations).

[1] Each hypercolumn found in Primates Primary Visual Cortex performs orientation recognition with 10º distance to the nearest ones [3, 5, 6]; this biological finding supports the chosen step.

Note that this space ζ is not constant as the Interpretation Layer evolves, by observing new images. In fact this space starts empty, because the initial number of Interpretation Layer neurons (designated before as *c*) is zero.

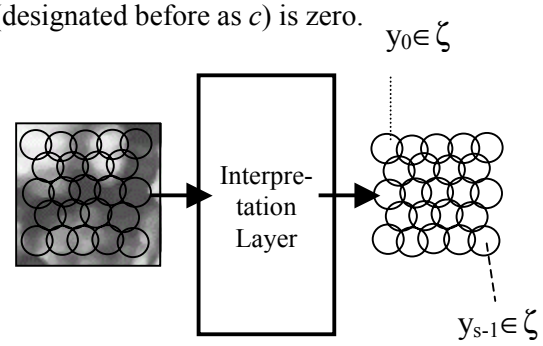

Figure 2. Interpretation Layer transforms an image to a set of *Interpretations*. For that purpose, the image is divided to a set of *s* (constant) Receptive Fields (the circular overlapping units shown in the figure). Each *Interpretation* is a pair (Index, Orientation. "Index" is the index of the neuron that recognised the observation inside the respective Receptive Field. "Orientation" is the index of the rotation that provided the neuron. ζ is the space of all possible *Interpretations*.

The unsupervised creation of the Interpretation Layer neurons is based on the analysis of how well the Layer is able to recognise the observed data. If the neuron that best recognises a feature does not have high correlation with it, then the Interpretation Layer creates a new neuron, specialised on that feature. The Neuron Output measures correlation, which is the greatest inner product between a feature (from its W^{36} expansion instances) and the Observation. The threshold for creation of a new neuron has been set to $\frac{\sqrt{2}}{2}$. By increasing this threshold, we loose generalisation power, and increase sharply the number of overall neurons. We obtained this result by simulating the Network generation on a high number of Receptive Field presentations (10000) and then measuring the number of neurons generated. We made this simulation for several threshold values. Results shown that threshold $\frac{\sqrt{2}}{2}$ is the critical value in the relation *neurons number created* versus *threshold* (the criteria was greatest second derivate).

If the correlation is greater than the threshold, then we have successful extraction of some feature, and we update the weights of the extracting neuron through a process similar to k-means clustering algorithm [9]:

$$W_i(t+1) = \frac{W_i(t) + \eta I}{\|W_i(t) + \eta I\|} = W_i(t) \to I(t), \qquad (1)$$

Where *I* is the Observation, *i* is the index of the extracting neuron, and η is some learning.

The Coherence Layer

The Coherence Layer learns and estimates structural relations within objects, using the Features generated in the Interpretation Layer.

After transforming the observed image in a set of Interpretations, the model starts analysing every image's Receptive Field, taking each one as Reference one at a time.

The model then processes one Reference by gathering information about the *v* nearest Receptive Field Interpretations around it (its neighbourhood).

This neighbourhood is used to set up input vector $\boldsymbol{z}$ that feeds Coherence Layer neurons. This input vector constitutes a description of Reference neighbourhood, and has $\boldsymbol{m} = v \times \#\zeta$ elements. We may see it as a flattening of a matrix with *v* rows and $\#\zeta$ columns. Each possible relative position of one neighbour, towards the Reference, indexes one row of $\boldsymbol{z}$. Each possible interpretation corresponds to a column. At each row, that represents a neighbour position, one and only one element have value $\alpha \neq 0$, corresponding to the Interpretation recognised in that relative position.

$\boldsymbol{z}$ is therefore a binary sparse vector, with values either $\boldsymbol{0}$ or $\boldsymbol{\alpha}$. Each element is associated with a particular relative position $\boldsymbol{p}$ (in polar coordinates, with coordinates origin in the Reference Receptive Field) and at a particular interpretation $\boldsymbol{y} \in \zeta$. This element has value α - we say it **is active** - if *y* is observed on relative position *p*; otherwise it has value 0.

We choose α such that *z* is normalised; as only *v* (number of chosen neighbours) elements of *z* are active, then we choose $\alpha = \frac{1}{\sqrt{v}}$.

Learning Mechanism

Coherence Layer uses Reference Interpretation and the neighbour Interpretations (the latter given by vector $\boldsymbol{z}$) to learn structural inter-positioning relations among features – Figure 3.

Recall that an Interpretation is pair (*Feature, Orientation*). Designate Reference Orientation by k'. Neighbourhood is rotated such that Reference becomes angular origin. Name this transformed neighbourhood vector z'.

Define ψ the Coherence Layer weights matrix. Notice that, by rotating neighbourhood $\mathbf{z}$ to $\mathbf{z}'$, Coherence Layer becomes orientation invariant. Also notice that weights ψ in the Coherence Layer do not need to contain data relative to the Reference Orientation. Therefore, those weights just refer to the Reference Feature (c rows), and to each element of the input (m columns). ψ is updated using Hebb rule [3].

$$\Delta\psi_{wi} = \eta \cdot z'_i \cdot d_{wk'}, \tag{2}$$

where η is the learning rate, and d (known desired vector) represents the output of the Interpretation Layer when the Reference Receptive Field actuates it. The elements of this vector represent all possible interpretations. It is a vector with all zeros, except in the desired interpretation $d_{w'k'}$, in which it has value one (w' designate the desired Reference Feature).

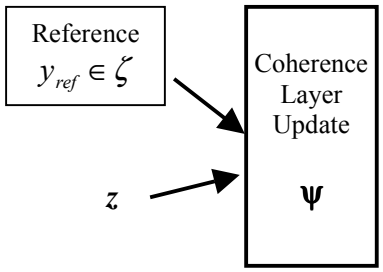

Figure 3. Receiving description vector z and the desired Interpretation for the Reference (by capturing its output from the Interpretation Layer), Coherence Layer reinforces the inter-positioning relations between the Reference and its neighbours (those relations are described in the Coherence Layer weights).

Estimation

The objective of the Coherence Layer is to identify areas of an image that are Coherent with each other, and thus may belong to the same object. We identify this coherency, when some Reference Neighbourhood allows Coherence Layer to estimate Reference interpretation correctly.

If Coherent Layer is not able to estimate correctly the Reference from the neighbourhood, then either the neighbourhood falls in a boundary among several objects, or it falls in one object that has not been previously observed by the model.

On the other hand, if the neighbourhood has been observed before in a coherent manner (that is, conserving structural inter-positioning relations), then Coherent Layer may successfully estimate the Reference Receptive Field.

We designate **Coherence Layer Estimator** a processing unit that uses Coherence Layer in order to make estimation.

Its input is Reference Neighbourhood, given by vector z – Figure 4.

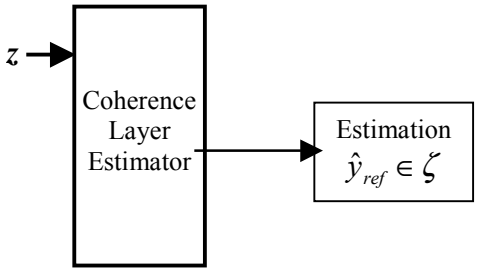

Figure 4. Receiving description vector z, Coherence Layer Estimator estimates the Reference interpretation.

Coherence Layer Estimator is an assemblage of a **Pre-Processing Unit** and 36 identical **Estimator Modules**, which are in fact Coherence Layer copies.

The Reference estimation procedure starts by postulating that the Reference has some particular Orientation k. The neighbourhood description vector z is then transformed to $z^{(k)}$, which is the result of a rotation in the neighbourhood such that the Reference becomes angular coordinates origin. This process is repeated to all possible Reference Orientations, so we obtain 36 $z^{(k)}$ vectors, with $k \in \{0,..,35\}$. Each of those vectors feeds one Estimator Module. Estimator Modules output constitute vector q. Although it is a vector, it may be addressed by 2 indexes, indicating the feature and orientation each element refers to. Each Estimator Module output will contribute to some elements of q (the ones assigned to the respective orientation). The greatest value in q will indicate not only the Reference Feature estimation but also the Reference Orientation.

Therefore, the output of the Coherence Layer Estimator is given by

$$q_{w,k} = \sum_i \psi_{wi} \cdot z^{(k)}{}_i \,. \tag{3}$$

Any presentation of an object, no matter how oriented or located, maintains its internal position and orientation relations. This representation model learns and recognises structural coherences

in these relations, indifferently to the absolute locations or absolute orientations where they were observed.

3. Features Network Properties

The model described is a representation structure that stores information both about existing features (Interpretation Layer) and about typical position relations among them (Coherence Layer).

We claim that the Coherence Layer, assembled as described in previous chapter, has the property of representing its information in a content-addressable way (the proof for this claim will be provided in further publications). That property, in conjunction with the Orientation Invariance referred in *Interpretation Layer* section, allows Features Network to have the following overall properties:

- Information stored is location and orientation independent;
- Topological information is content-addressable: given a piece of information about some object details, the network will raise expectations for the contiguous object details;
- Object recognition is performable: it is based on measuring the coherence between observation and expectation in particular image areas;
- Information about object boundaries is retrievable: when expectation level of uncertainty is higher than some threshold, the Network may expect a boundary.

These properties enable the establishment of criteria in order to decide which features from the Interpretation Layer are more relevant; for example, we may associate feature relevance to statistical variance of the weights in the Coherence Layer (the greater the variance, the greater the relevance of the feature to the estimation process).

4. Results

We have applied Features Network to a higher-level analysis system, capable of assembling areas where Features Network's expectation corresponded to observation, as well as determining areas where Features Network raised high levels of uncertainty (typically boundaries). To analyse the behaviour of the overall system, we presented a sequence of only two images. Both images contain the same real objects, but they are presented in different locations. The images are 128x128 pixels, and the Receptive Fields have 20 pixels diameter. We chosen two medially featured objects, an ox skull and a shell. The ox skull was purposely modified, such that it became fragmentary. The system had no previous information about how many objects were within the images, and also had no other knowledge about the kind of visual information it was supposed to receive, or about what correlation it was expected to find between the two images.

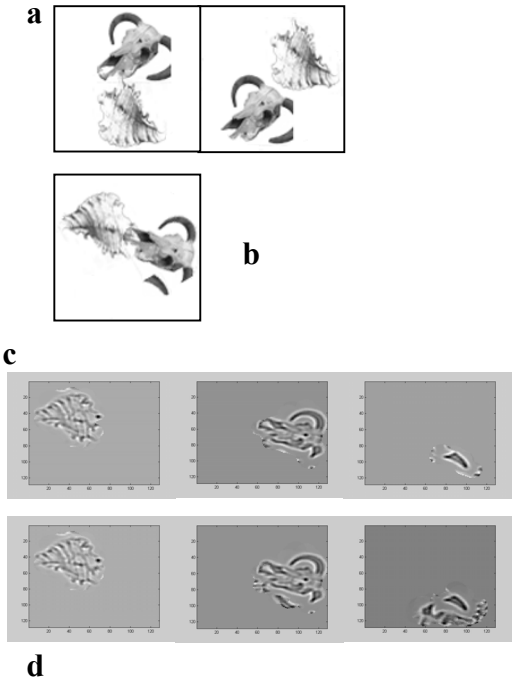

Figure 5. a. Learned images, in sequence. **b.** Test image. **c.** Objects segmentation in the test image, due to levels of uncertainty detection. **d.** Extrapolation of objects based on Features Network information.

Results expected

Given those images, the Features Network was expected to learn the relevant Features (in its Interpretation Layer) as well as to use those features to raise some learning about topological coherences it could find in the Coherence Layer.

As the objects were presented in different locations, we expected that topological coherences found between both objects in the first picture would disappear or be substantially diminished in the second picture, as they would be found to be inconsistent with that image. We also expected that the Features Network would represent the fragmentary object as a whole, as its inner coherences were not broken. Besides, all topological coherence recognition should be robust to objects rotation (as well as location).

Testing method

In order to test whether the Representation structure constructed in the Coherence Layer is location and orientation independent or not, i.e. to test if it would succeed to recognise both objects, we presented a third image, where the same two objects were rotated and located independently (Figure 5.b). To further test the representation consistency, we inconsistently rotated one part (the horn) of the fragmentary object.

In Figure 5.c we show maps of image topological coherences found in Figure 5.b, given the knowledge learned in figure 5.a. To build those maps, we gathered regions where Receptive Fields were coherent with their neighbourhoods. We also found Receptive Fields with high probability to correspond to boundaries (as commented in the Coherence Layer Estimation Section, those Receptive Fields should not be coherent with their neighbourhood). From the coherency regions and boundaries, we generated a set of "islands", with high probability of corresponding to independent objects. Each map corresponds showing the Observations found at every Receptive Field of a particular "island".

The maps tell us that the analysis system found three regions of high topological coherence among their inner features. Those results would be obviously expected due to the inconsistency deliberately introduced in the fragmentary object in Figure 5.b.

To show that the Features Network is indeed a Representation structure, we used the knowledge of the Coherence Layer to expand the maps in Figure 5.c. The result in Figure 5.d, shows clearly that the Features Network is a Representation Structure independent of location or orientation, and also that the fragmentary object was represented as a whole.

5. Discussion

The results we obtained, show that this model achieves the purposes it was designed for, namely the properties referenced in Section 3. As discussed in the last paragraph of that Section, these properties open doors to a self-controlling mechanism that prevents the indefinite grow of the number of neurones in the Interpretation Layer.

In further experiments, we found that the model also shows tolerance to scale (until about 30% for amplifications and 20% for reductions) and also to some degree of geometrical distortions. Using this scale tolerance, we may achieve further scale robustness simply by considering different scales. Images within Receptive Fields may be compared to features at all possible scales, and not only we detect the feature but also in which scale it is being presented.

Reference

[1] Kalman, R.E., A new approach to linear filtering and prediction theory, Trans. ASME J. Basic Eng. 82, pp.35-45 (1960)

[2] Gross, C.G., Rocha-Miranda, C.E., Bender, D.B., *Visual properties of neurons in inferotemporal cortex of the macaque, Journal of Neurophysiology*, 35, pp. 96-111 (1972)

[3] Hebb, D.O., *The Organization of Behavior*, New York: Wiley (1949)

[4] Hubel, D.H., Wiesel, T.N., *Functional Architecture of Macaque Monkey Visual Cortex, Proceedings of Royal Society London*, B198, pp.1-59 (1977)

[5] Hubel, D.H., Wiesel, T.N., *Brain Mechanisms of Vision*, Scientific American, 241, pp.150-162 (1979)

[6] Hubel, D.H., *Eye, Brain and Vision*, New York: Scientifican American Library (1988)

[7] Krishnaiah, P.R., Kanal, L.N., *Classification, Pattern Recognition, and Reduction of Dimensionality*, Handbook of Statistics, vol.2. Amsterdam: North Holland (1982)

[8] Löwel, S., Singer, W., *Selection of Intrinsic Horizontal Connections in the Visual Cortex by Correlated Neuronal Activity*, Science 255, pp.209-212 (1992)

[9] Maybeck, P.S., *Stochastic Models, Estimation, and Control (Volume I e II)*, New York: Academic Press (1979)

[10] Rao, R., Ballard, D., *Dynamic Model of Visual Recognition Predicts Neural Response Properties in the Visual Cortex*, Neural Computation, 9, nº4, pp.721-763 (1997)

[11] Rissanen, J., *Stochastic Complexity in Statistical Inquiry*. Singapura: World Scientific (1989)

[12] Tanaka, K., Saito, H., Fukuda, Y., Morika, M., *Coding Visual Images of Objects in the Inferotemporal Cortex of the Macaque Monkey*, Journal of Neurophysiology 66, nº 1, pp.170-189 (July 1991)