Journal of Physiology - Paris 107 (2013) 369–398

Contents lists available at SciVerse ScienceDirect

Journal of Physiology - Paris

journal homepage: www.elsevier.com/locate/jphysparis

A three-layer model of natural image statistics

Michael U. Gutmann *, Aapo Hyvärinen

Dept. of Mathematics and Statistics, P.O. Box 68, FIN-00014 University of Helsinki, Finland
Dept. of Computer Science and HIIT, P.O. Box 68, FIN-00014 University of Helsinki, Finland

ARTICLE INFO

Article history:
Received 4 September 2012
Received in revised form 22 December 2012
Accepted 11 January 2013
Available online 29 January 2013

Keywords:
Natural images
Probabilistic modeling
Visual processing
Selectivity
Invariance
Sparse coding
Deep learning

ABSTRACT

An important property of visual systems is to be simultaneously both selective to specific patterns found in the sensory input and invariant to possible variations. Selectivity and invariance (tolerance) are opposing requirements. It has been suggested that they could be joined by iterating a sequence of elementary selectivity and tolerance computations. It is, however, unknown what should be selected or tolerated at each level of the hierarchy. We approach this issue by learning the computations from natural images. We propose and estimate a probabilistic model of natural images that consists of three processing layers. Two natural image data sets are considered: image patches, and complete visual scenes downsampled to the size of small patches. For both data sets, we find that in the first two layers, simple and complex cell-like computations are performed. In the third layer, we mainly find selectivity to longer contours; for patch data, we further find some selectivity to texture, while for the downsampled complete scenes, some selectivity to curvature is observed.

1. Introduction

Our paper belongs to the larger body of work on Bayesian perception. This theory of vision entails that the visual system is adapted to the properties of the world which it senses. In other words, it "knows" about the regularities within the visual stimuli (see, for example, Barlow, 2001; Simoncelli and Olshausen, 2001; Hyvärinen et al., 2009; Freeman and Simoncelli, 2011). Knowledge about the regularities can be mathematically expressed as knowledge about the probability distribution of the visual stimuli. Our goal here is to model this distribution and relate it to visual processing.

One powerful class of models specifies the distribution in a top-down manner in terms of latent variables which explain the structure in the visual stimuli (Olshausen and Field, 1996; Hyvärinen et al., 2009; Karklin and Lewicki, 2009; Zoran and Weiss, 2009; Ranzato and Hinton, 2010; Cadieu and Olshausen, 2012). Another class of models corresponds to a bottom-up approach where the visual stimuli are processed in multiple layers of computation (Osindero et al., 2006; Köster and Hyvärinen, 2010; Gutmann and Hyvärinen, 2012b). The model in this paper belongs to this latter class.

It has been proposed that the layers should alternate between elementary selectivity and invariance, or tolerance computations (Fukushima, 1980; Riesenhuber and Poggio, 1999). In line with simple models for experimental data (see, for example, Hubel, 1995), the first layer should be selective to localized, oriented bandpass structure, and the second layer should be tolerant to variations in the localization of that structure. The idea is that after several layers of computations, high selectivity to specific structure could be combined with moderate tolerance to its possible variations. The combination of the opposing poles of selectivity and invariance is thought to be essential for reliable object recognition, or for biological and artificial visual processing in general (DiCarlo and Cox, 2007; Serre et al., 2007; Jarrett et al., 2009; Rust and Stocker, 2010).

A fundamental question that arises with the bottom-up approach is to know what should be selected or tolerated at each layer. We approach this issue by learning the selectivity and tolerance computations from natural images. This approach has previously accounted for the computations on the first two layers (Osindero et al., 2006; Köster and Hyvärinen, 2010; Gutmann and Hyvärinen, 2012b). Here, we learn all layers in a three-layer model, and pay particular attention to the computations which emerge in the third layer.[1]

* Corresponding author at: Dept. of Mathematics and Statistics, P.O. Box 68, FIN-00014 University of Helsinki, Finland. Tel.: +358 9 191 51496.
E-mail addresses: michael.gutmann@helsinki.fi (M.U. Gutmann), aapo.hyvarinen@helsinki.fi (A. Hyvärinen).

[1] Preliminary results were reported at the International Conference on Pattern Recognition 2012 (Gutmann and Hyvärinen, 2012a).

0928-4257/$ - see front matter © 2013 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.jphysparis.2013.01.001

2. Material and methods

In Section 2.1, we present the natural image data used. In Section 2.2, we introduce and explain the parametric model of the processing in the three layers. Section 2.3 shows how to learn the parameters by fitting a probability density function to the natural image data.

2.1. Data and preprocessing

We use two types of natural image data. The first data set consists of image patches that we have extracted from thirteen larger gray-scale images which have been used before to study properties of natural images (Hyvärinen et al., 2009). The patches are of size 32×32 pixels. The second data set is the tiny images data set by Torralba et al. (2008), converted to gray scale. That data set consists of about eighty million images which show complete visual scenes downsampled to 32×32 pixels. Examples from the two data sets are shown in Figs. 1a and 1b. When referring to both data sets at the same time, we will call them "natural images".

As preprocessing, we removed the DC component (average value of each tiny image, or image patch) and normalized the norm of the resulting image. The norm used here was computed in the whitened space. Unlike the ordinary norm without whitening, this norm is not dominated by the low-frequency content of an image (Hyvärinen et al., 2009, Chapter 5). This preprocessing is a form of luminance and contrast gain control. Further, the preprocessing makes it easier to model the statistical dependencies between the pixels by normalizing their marginal distributions to some extent. This preprocessing thus is motivated by both neuroscience and data-modeling considerations. After normalization, we reduced the dimensionality by PCA from 1024 to 600, which corresponds to low-pass filtering of the images. After dimension reduction, the images are elements inside a 600 dimensional sphere. The retained dimensions account for a bit more than 98% and 99% of the variance of the image patches and the tiny images, respectively. We denote the resulting, preprocessed images by $\mathbf{x}$.

Fig. 1c and d show the effect of the preprocessing for the natural image examples in Fig. 1a and b, respectively. For visualization, we scaled each preprocessed natural image such that the full colormap is used. The examples visualize the luminance and contrast gain control, and they show further that our dimension reduction does not cause a perceptible blurring.

2.2. Parametric model for the three layers of computation

After the initial preprocessing (gain control), an input image is processed in three layers of computation. The outputs of each layer form statistics which we use in Section 2.3 to define the value that a probability density function $p_{\mathbf{x}}$ takes at $\mathbf{x}$, that is, at a given image after gain control. The three layers are defined as follows.

2.2.1. First layer

The gain-controlled image $\mathbf{x}$ is projected onto features $\mathbf{w}_i^{(1)}$, followed by half-wave rectification. This gives the outputs $y_i^{(1)}$ of the first-layer units,

$$y_i^{(1)} = \max\left(\mathbf{w}_i^{(1)} \cdot \mathbf{x}, 0\right), \quad i = 1 \dots n^{(1)}. \tag{1}$$

Here, $\mathbf{w}_i^{(1)} \cdot \mathbf{x}$ denotes the dot-product between the vectors $\mathbf{w}_i^{(1)}$ and $\mathbf{x}$. The features $\mathbf{w}_i^{(1)}$ are parameters of the model which will be learned from the data using the procedure outlined in Section 2.3 below. The number of first-layer units was fixed to $n^{(1)} = 600$. A linear stage followed by rectification is a simple model for the steady-state firing rate of neurons (Dayan and Abbott, 2001, Chapter 7.2). In this model, the features $\mathbf{w}_i^{(1)}$ correspond to the receptive fields of the neurons.

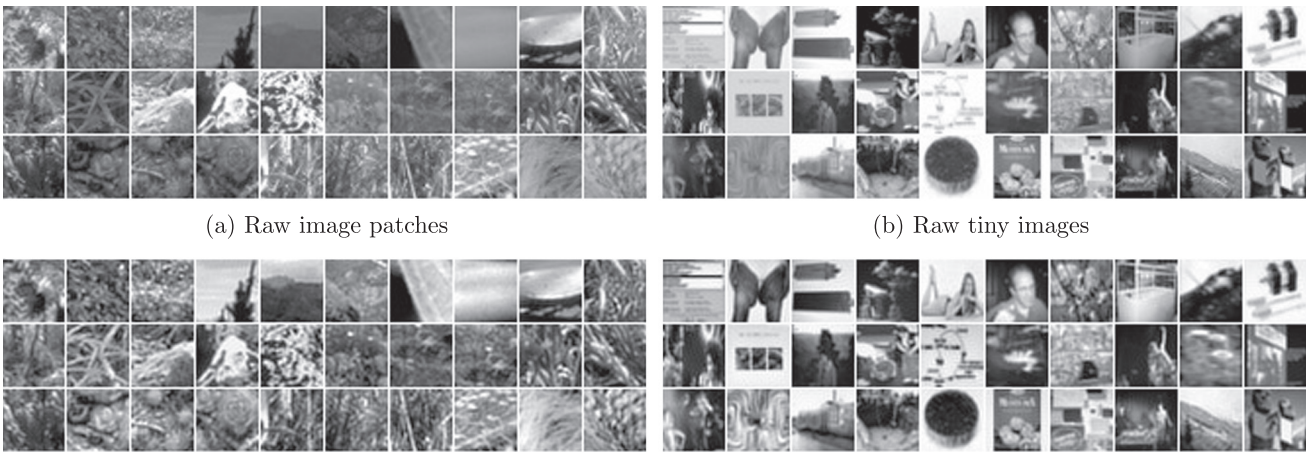
Based on the symmetry of natural images (see, for example, Gutmann and Hyvärinen, 2012b, Section 5.4), we make the simplifying assumption that for each receptive field $\mathbf{w}_i^{(1)}$, there exists a receptive field $\mathbf{w}_{i'}^{(1)}$ with a sign-inverted spatial pattern, that is $\mathbf{w}_{i'}^{(1)} = -\mathbf{w}_i^{(1)}$. We also assume that the weights in the second layer (see below) are the same for $y_i^{(1)}$ and $y_{i'}^{(1)}$. This assumption reduces the number of free parameters, and we can compute the first layer outputs as $y_i^{(1)} = \mathbf{w}_i^{(1)} \cdot \mathbf{x}$, for $i = 1 \cdots n^{(1)}/2 = 300$.

2.2.2. Second layer

After elementwise squaring, the outputs $\mathbf{y}^{(1)} = (y_1^{(1)}, \dots, y_{n^{(1)}}^{(1)})$ from the first layer are projected onto second-layer features $\mathbf{w}_i^{(2)}$. The outputs $y_i^{(2)}$ of the second-layer units are obtained as

$$y_i^{(2)} = \ln\left(\mathbf{w}_i^{(2)} \cdot (\mathbf{y}^{(1)})^2 + 1\right), \quad i = 1 \cdots n^{(2)}. \tag{2}$$

The number of second-layer units was fixed to $n^{(2)} = 100$. The weight vectors $\mathbf{w}_i^{(2)}$ are, again, parameters that we learn from the data. Each element $w_{ki}^{(2)}$ of a vector $\mathbf{w}_i^{(2)}$ is constrained to be nonnegative. The functional form of (2) corresponds to the energy model for complex cells (Adelson and Bergen, 1985), albeit with receptive



(a) Raw image patches (b) Raw tiny images

(c) Preprocessed image patches (d) Preprocessed tiny images

Fig. 1. Natural images before and after preprocessing. (a and b) Examples of extracted patches from larger images and examples from the tiny images data set. Pixel values of zero are shown in black, and white corresponds to pixel values of 255. (c and d) The same images after preprocessing. Preprocessing consists of removing the average value from each image, standardizing its norm, and PCA-based dimension reduction. Each preprocessed image was re-scaled to use the full color map.