A model for learning topographically organized parts-based representations of objects in visual cortex: topographic non-negative matrix factorization

Kenji Hosada, Masataka Watanabe, Heiko Wersing, Edgar Körner, Hiroshi Tsujino, Hiroshi Tamura, Ichiro Fujita

2009

Preprint:

Neural Computation (2009). In press.

A model for learning topographically organized parts-based representations of objects in visual cortex: topographic non-negative matrix factorization

Kenji Hosoda
hosoda@bs.t.u-tokyo.ac.jp
Masataka Watanabe
watanabe@bs.t.u-tokyo.ac.jp
Department of Quantum Engineering and Systems Science, University of Tokyo, Tokyo, Japan

Heiko Wersing
heiko.wersing@honda-ri.de
Edgar Körner
Edgar.Koerner@honda-ri.de
HONDA Research Institute Europe GmbH, Offenbach, Germany

Hiroshi Tsujino
tsujino@jp.honda-ri.com
HONDA Research Institute Japan Co., Ltd, Saitama, Japan

Hiroshi Tamura
tamura@bpe.es.osaka-u.ac.jp
Ichiro Fujita
fujita@fbs.osaka-u.ac.jp
Graduate School of Frontier Biosciences, Osaka University, Osaka, Japan

Object representation in the inferior temporal cortex (IT), an area of visual cortex critical for object recognition in the primate, exhibits two prominent properties; (1) objects are represented by the combined activity of columnar clusters of neurons, each cluster represents component features or parts of objects, and (2) closely related features are continuously represented along the tangential direction of individual columnar clusters. Here we propose a learning model that reflects these properties of parts-based representation and topographic organization in a unified framework. This model is based on a non-negative matrix factorization (NMF) basis-decomposition method. NMF alone provides a parts-based representation where non-negative inputs are approximated by additive combinations of non-negative basis functions. Our proposed model of topographic NMF (TNMF) incorporates neighborhood connections between NMF basis functions arranged on a topographic map and attains the topographic property without losing the parts-based property of the NMF. The TNMF represents an input by multiple activity peaks to describe diverse information whereas conventional topographic models, such as self-organizing map (SOM), represent an input by a single activity peak in a topographic map. We demonstrate the parts-based and topographic properties of the TNMF by constructing a hierarchical model for object recognition where the TNMF is at the top tier for learning high-level object features. The TNMF showed better generalization performance over NMF for a data set of continuous view change of an image, with more robustly preserving the continuity of the view change in its object representation. Comparison of the outputs of our model with actual neural responses recorded in the IT indicates that the TNMF reconstructs the neuronal responses better than the SOM, giving plausibility to the parts-based learning of the model.

1 Introduction

In the ventral pathway of the primate visual cortex, object features are gradually extracted with increasing specificity and invariance by a network of cortical areas. The inferior temporal cortex (IT) subserves a critical component of visual processing of objects at later stages of this pathway (Mishkin, Ungerleider, & Macko, 1983; Gross, 1994). How object images are represented in the IT has been investigated intensively with both theoretical and experimental approaches. Object representation in the IT has

been proposed to include two distinct properties. One is the parts-based representation in which objects are represented by combinations of redundant component features rather than by objects themselves or by fully distributed components like Fourier descriptors (Tanaka, Saito, Fukada, & Moriya, 1991; Fujita, Tanaka, Ito, & Cheng, 1992). Optical imaging of neural population activity in the IT demonstrates that a single object elicits multiple patches of activity across cortex; these appear to represent component features of the object (Tsunoda, Yamane, Nishizaki, & Tanifuji, 2001). Another property is that parametrically related object features are orderly represented along the cortical surface within each cortical patch (Tanaka, 2003). When the image of an object, such as face, is systematically transformed (e.g., successively rotated in depth), activity spots gradually shift their positions on IT cortex (Wang, Tanifuji, & Tanaka, 1998). Although these two representation properties have important functional implications (Fujita, 2002; Tanaka, 2003), how they are established has not been determined.

The self-organizing map (SOM) has been used widely to model the topographic organization of cortex (Kohonen, 1988; Durbin & Mitchison, 1990; Obermayer, Ritter, & Schulten, 1990; Swindale, 1991; Yu, Farley, Jin, & Sur, 2005). A SOM represents a high-dimensional input with a single point in a low-dimensional network that maximizes proximity relations between different inputs. If an input is parametrically changed, then the activity peak shifts in position. A SOM does not provide the property of parts-based representation because each of the mapped points reflects a holistic aspect of the input.

In this study, we propose a learning model that explains both of the parts-based and topographic properties of IT. This model is based on a non-negative matrix factorization (NMF) basis-decomposition method (Lee & Seung, 1999). In general, basis decomposition methods derive basis functions and coefficients from input data, and reconstruct the inputs by weighted combinations of these basis functions. Principal component analysis (PCA), a standard basis decomposition method, maximizes this reconstruction and derives fully distributed representations. Vector quantization (VQ), another method for basis decomposition, imposes the constraint that each input is represented by only one basis function. While all basis functions and coefficients in PCA and all basis functions in VQ may take both positive and negative values, the NMF requires all entries to be non-negative. The NMF has been demonstrated to yield intuitive parts-based representations for non-negative data (Lee & Seung, 1999; Buchsbaum & Bloch, 2002; Xu, Liu, & Gong, 2003; Cho & Choi, 2005).

Our model of topographic NMF (TNMF) extends the original NMF by introducing

neighborhood functions between NMF basis functions placed evenly in a low-dimensional space. With this extension, the non-negative constraint inherited from the NMF leads to an overlapping of basis functions along neighboring structures. Having been trained (i.e., having searched for the best set of basis functions and coefficients), a topographic organization or a map is formed of parts-based representation. Just as the SOM is a topographic extension of VQ, the TNMF is a topographic extension of the NMF.

As a statistical model, this topographic extension involves the *a priori* assumption that latent features embedded in the inputs are smoothly distributed. Given this assumption, the TNMF can learn latent features that are not apparent in training inputs, by interpolation through topographic neighborhood cooperation. Especially, the TNMF can smoothly embed and interpolate values of a variable encoded in a group of neurons by a population coding (Pouget, Zemel, & Dayan, 2000), in which neural responses are correlated with a Gaussian function representing a value. We first demonstrate this generalizing capability of the TNMF with artificial data which consisted of neural responses under a population coding.

To show the TNMF properties as a cortical learning model, we then applied the TNMF to a hierarchical model of neural computation by the ventral pathway (Wersing & Körner, 2003). The hierarchical model extracts visual features with increasing specificity and invariance based on alternating feature detection and integration processes, as in the neocognitron (Fukushima, 1980) and the HMAX model (Riesenhuber & Poggio, 1999). Progressive processing is biologically plausible and can explain some aspects of neural responses in the ventral pathway (Serre, Kouh, Cadieu, Knoblich, Kreiman, & Poggio, 2005; Serre, 2006; Zoccolan, Kouh, Poggio, & DiCarlo, 2007). In our model, the TNMF was used for training the highest layer of the hierarchy. The lower layers of the model are highly competitive with other current recognition algorithms (Wersing & Körner, 2003). We show that the hierarchical model with the TNMF captures the parts-based and topographic properties of object representation in IT. We first examine the topographic-induced generalizing capability of the TNMF for model responses to rotating views of an object. We then assessed the biological plausibility of this model at the single neuron level by comparing the model outputs with the responses of the monkey IT neurons we reported previously (Tamura, Kaneko, Kawasaki, & Fujita, 2004; Tamura, Kaneko, & Fujita, 2005). We also evaluated performance of models generated with SOM and the original NMF.

2 Methods

2.1 Proposed Model. The original NMF approximates a non-negative input vector **v** by an additive combination of r non-negative basis vectors $\mathbf{w}_a$ (a = 1, ...,r):

$$\mathbf{v} \approx \sum_{a=1}^{r} h_a \mathbf{w}_a = \mathbf{Wh},$$

where $\mathbf{h} = (h_1,h_2,...,h_r)^{\mathrm{T}}$ is a non-negative coefficient vector. The input vector **v** is represented by the coefficient vector **h** via the basis matrix $\mathbf{W} = (\mathbf{w}_1,\mathbf{w}_2, ...,\mathbf{w}_r)$. During training, basis functions and coefficients are iteratively updated to minimize errors between inputs and approximations with non-negative constraints on all entries.

The TNMF incorporates neighborhood connections between NMF basis functions arranged on a topographic map (see Figure 1):

$$\mathbf{v} \approx \sum_{b=1}^{r} h_b \sum_{a=1}^{r} M_{ab} \mathbf{w}_a = \mathbf{WMh},$$

where **v**, **W**, **h** are a non-negative input vector, a non-negative basis matrix, and a coefficient vector, respectively, as in the original NMF. The new term $\mathbf{M} = (M_{ab})$ is a non-negative $r \times r$ dimensional matrix that defines neighborhood connections between r basis functions. Choosing **M** as the identity matrix reduces the TNMF to the NMF. We arranged basis functions on a two-dimensional square-lattice topographic map, and set neighborhood connection weights to be normal distribution (Gaussian) functions on the map:

$$M_{ab} = \exp(-\|\mathbf{p}_a - \mathbf{p}_b\|^2 / 2\sigma^2),$$

where $\mathbf{p}_a$ and $\mathbf{p}_b$ are positions of basis functions 'a' and 'b' on the map, respectively. The Gaussian radius σ is a user-defined variable. In training, the neighborhood function **M** is fixed, while the basis **W** and coefficient **h** are updated to reconstruct the input **v** optimally under the non-negative constraint.

The data approximation is achieved by maximizing the following objective function:

$$F = \sum_{i=1}^{n}\sum_{j=1}^{m}\left[V_{ij}\log(WMH)_{ij} - (WMH)_{ij}\right],$$

where $\mathbf{V} = (\mathbf{v}_1,\mathbf{v}_2, ...,\mathbf{v}_m) = (V_{ij})$ indicates m input vectors and $\mathbf{H} = (\mathbf{h}_1,\mathbf{h}_2, ...,\mathbf{h}_m) = (H_{bj})$ indicates corresponding m coefficient vectors. The function F can be interpreted as a log likelihood in a model in which each input entry V_{ij} is generated by adding Poisson noise to the approximation $(WMH)_{ij}$ (Lee & Seung, 1999). The maximization of F is achieved by the following multiplicative updates:

$$W_{ia} = W_{ia}\frac{\sum_j (MH)_{aj}V_{ij}/(WMH)_{ij}}{\sum_j (MH)_{aj}}$$

$$H_{bj} = H_{bj}\frac{\sum_i (WM)_{ib}V_{ij}/(WMH)_{ij}}{\sum_i (WM)_{ib}}.$$

These updates monotonically maximize the function F along with satisfying the non-negative constraint (Lee & Seung, 2001). We also normalized the coefficient matrix **H** together with these updates:

$$H_{bj} = H_{bj}/\sum_j H_{bj}.$$

This normalization eliminates the indefiniteness of **WMH** under the transformation $\mathbf{W} \rightarrow \alpha^*\mathbf{W}$ and $\mathbf{H} \rightarrow \mathbf{H}/\alpha$, where α is scalar. The algorithm reaches a local maximum in the objective function by repeating these updates, and it does not always attain the global maximum. Therefore, we employed an exhaustive search in which the most optimal solution is selected from sufficient multiple solutions starting from random initial conditions, and an annealing method in which the neighborhood radius σ is narrowed with the number of iteration as in SOM algorithms (Kohonen, 1988).

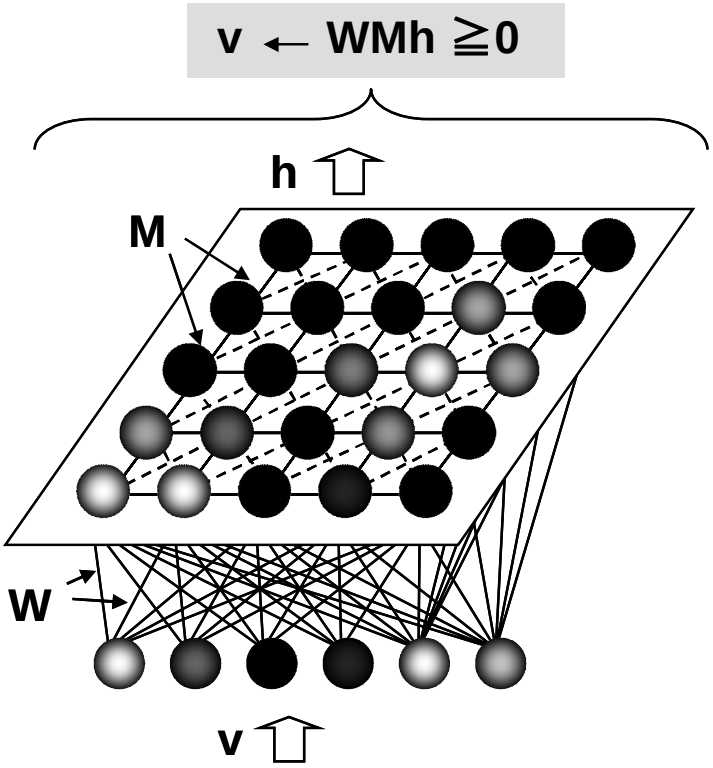

Figure 1: *A diagram of the proposed TNMF model where* ***v*** *indicates an input vector,* **W** *indicates basis functions,* **M** *indicates neighborhood functions, and* **h** *indicates a coefficient vector. Shading of nodes represents magnitudes of input and coefficient entries. All the entries are restricted to be non-negative. If* **M** *is the identity matrix, this model reduces to the original NMF. TNMF approximates* **v** *by* **WMH** *updating* **W** *and* **h**.

2.2 Hierarchical Visual Model. The hierarchical model of the ventral pathway comprises multiple layers. The proposed TNMF algorithm is used for training the highest layer (Figure 2). The lower layers（S1, C1, S2, S3）perform processing of visual form starting from edge detection, which has been described in detail in Wersing and Körner (2003). In this model, the S and C layers are alternately structured (Fukushima, 1980). The S layers increase specificity and the C layers increase invariance of input representations.

First the image vector **v** is processed in the S1 layer that extracts edge components through Gabor filters of 4 orientations in the retinotopic coordinate:

$$q_l^{(1)}(x, y) = \left| \mathbf{w}_l^{(1)}(x, y) \cdot \mathbf{v} \right|,$$

where $q_l^{(1)}(x, y)$ is a neural response in the S1 layer, and $\mathbf{w}_l^{(1)}(x, y)$ is a Gabor filter of the orientation l at the retinotopic position (x, y) vectorized in the same way as $\mathbf{v}$. The $q_l^{(1)}(x, y)$ is then modified by winner-take-most competition, suppressing suboptimal responses, over all orientation preferences at the position (x, y):

$$r_l^{(1)}(x, y) = \begin{cases} 0 & \text{if } \dfrac{q_l^{(1)}(x, y)}{M} < \gamma \\ \dfrac{q_l^{(1)}(x, y) - M\gamma}{1 - \gamma} & \text{else} \end{cases},$$

where $M = \max_k (q_k^{(1)}(x, y) + \varepsilon_{(<<1)})$. The response is rectified by a threshold function with a threshold θ which is common over the S1 layer:

$$s_l^{(1)}(x, y) = H(r_l^{(1)}(x, y) - \theta),$$

where $H(x) = 1$ if $x > 0$ and $H(x) = 0$ else and $s_l^{(1)}(x, y)$ is the final S1 response.

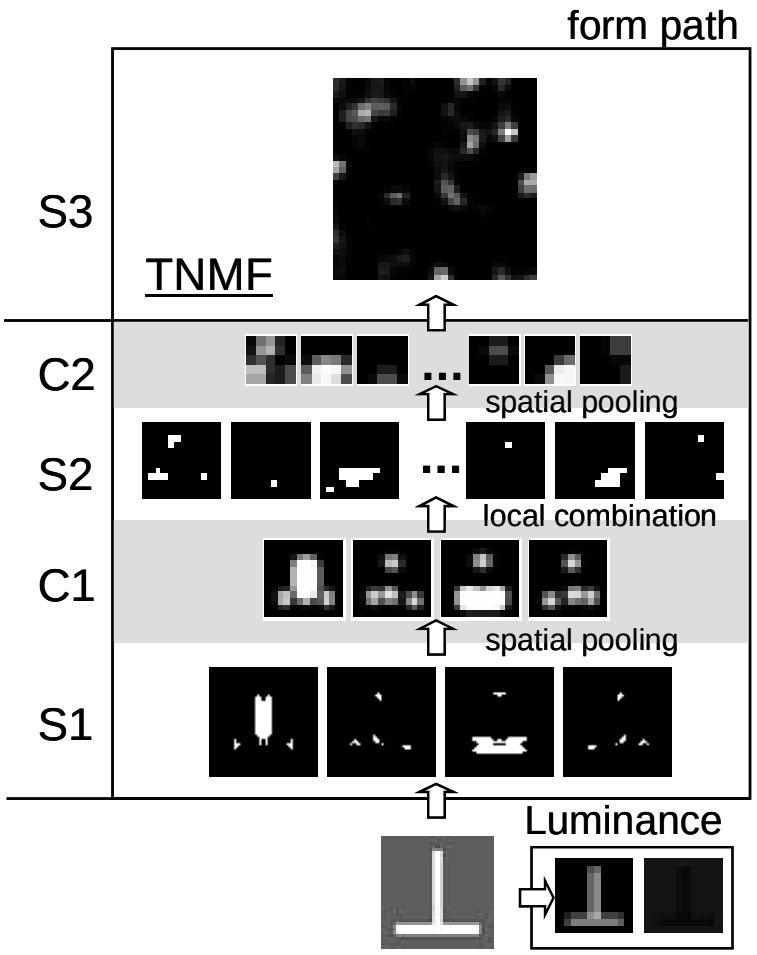

Figure 2: *Architecture of the hierarchical model. In the form path, S and C layers are alternately structured. This feature increases specificity and invariance of input representations, respectively. The proposed TNMF was used for training the S3 layer. An additional luminance layer processes white and black images of lower-resolution.*

The S1 outputs are then pooled over a range of the retinotopic coordinate with an OR-operation in the C1 layer:

$$c_l^{(1)}(x, y) = \tanh(g(x, y).s_l^{(1)}),$$

where $g(x, y)$ is a normalized Gaussian kernel with radius σ that integrates a local array

of S1 outputs with the same orientation preference. The function tanh is the hyperbolic tangent sigmoid transfer function and implements a smooth spatial OR-operation by the saturating nonlinearity. S1 neurons respond selectively to edges at specific orientation and location, and C1 neurons respond selectively to edges with a specific orientation across a larger area of the visual field, as observed for single and complex cells in the primary visual cortex (V1), respectively (Hubel & Wiesel, 1962).

Next the image signals are passed through the S2 and C2 layers. The S2 layer codes 50 types of local image features $\mathbf{w}_{lk}^{(2)}(x, y)$ (l = 1, .., 50; k = 1, …, 4) for each retinotopic position (x, y), such as corners and elongated edges. The 50 types of features are a product of sparse invariant basis decomposition (Wersing & Körner, 2003). Briefly, the "learning" by this layer is based on sets of C1 outputs within the 4×4 retinotopic patch across all four orientation preferences. The C1 data are approximated by weight-sharing basis functions representing the 50 types of features, allowing spatial shifts. The approximation is performed under a sparse and non-negative constraint that yields parts-based representations. The obtained basis functions represented more complex features than simple oriented edges.

The S2 layer then combines C1 outputs within local retinotopic patches across all orientation preferences:

$$q_l^{(2)}(x, y) = \sum_k \mathbf{w}_{lk}^{(2)}(x, y).\mathbf{c}_k^{(1)}.$$

The S2 response $q_l^{(2)}(x, y)$ is again modified by the winner-take-most competition as in the S1 layer. The S2 outputs are then pooled over a range of the retinotopic coordinate with an OR-operation in the C2 layer in the same way as in the C1 layer.

Model parameters such as the competition selectivity γ, threshold θ, and Gaussian radius σ are optimized for an object recognition task (Wersing & Körner, 2003): γ = 0.7, θ = 0.3, σ = 4.0 for the level 1 hierarchy (i.e., S1 and C1); γ = 0.9, θ = 1.0, σ = 2.0 for the level 2 hierarchy. The retinotopic resolution is set to be 14×14 for the C1 layer and 5×5 for the C2 layer, starting from 64×64 image pixels.

Finally, the C2 outputs are fed into the S3 layer under the TNMF model. The outputs of the S3 layer are basis coefficients **H**. In the experiment in Section 3.4, S3 neurons were trained with an image set consisting of five views (0°, 15°, 30°, 45°, 60°) of 50 objects (Figure 3). While the C2 layer and below have topographic maps of *a priori* retinotopic coordinates (Figure 2), the S3 layer acquires a topographic map explicitly representing latent object transformation parameters in the image set. The number of

basis functions was 32×32, and the Gaussian radius σ in the neighborhood function **M** was 1 (1/32 of map size).

For comparison between the model output and biological data, we also incorporated an additional luminance-processing layer to the hierarchical model. Some IT neurons are less selective for shape but respond to many stimuli either darker or brighter than background (Ito, Fujita, Tamura, & Tanaka, 1994; Tamura, Kaneko, & Fujita, 2005). The aim of including the luminance layer is to alleviate the influence of such IT neurons on the evaluation of shape processing by the model. This layer codes luminance polarities (white and black) at each pixel on lower-resolution images:

$$u_1(x,y) = \begin{cases} (v'(x,y) - 0.6)/0.4 & \text{if } v'(x,y) \geq 0.6 \\ 0 & \text{else} \end{cases}$$
$$u_2(x,y) = \begin{cases} (0.4 - v'(x,y))/0.4 & \text{if } v'(x,y) < 0.4 \\ 0 & \text{else} \end{cases},$$

where $v'(x, y)$ indicates the intensity of the lower-resolution image at the position (x, y), and $u_1(x, y)$ and $u_2(x, y)$ represent respectively the whiteness and blackness of $v'(x, y)$ in a graded manner.

Figure 3: *Examples of object images for consecutive views used to train the S3 layer.*

2.3 Neuron Data. The neuron data for comparison with the results of our model were obtained from the experiments described in Tamura et al. (2004, 2005). Using a multi-probe electrode, extracellular spikes were recorded from neurons (n = 497) in the dorsal part of anterior IT (cytoarchitectonic area TE) of the right hemisphere of four anesthetized Japanese monkeys (*Macaca fuscata*). Neural responses to complex shapes were examined by using 64 visual stimuli (Figure 4). The stimuli were individually presented for one second at the center of the receptive field with 10 repetitions for each recording site. The response rate was calculated by computing the mean firing rate during stimulus presentation, subtracting the spontaneous firing rate, and truncating negative values. In this study, we only consider excitatory neural responses in comparison to non-negative model outputs.

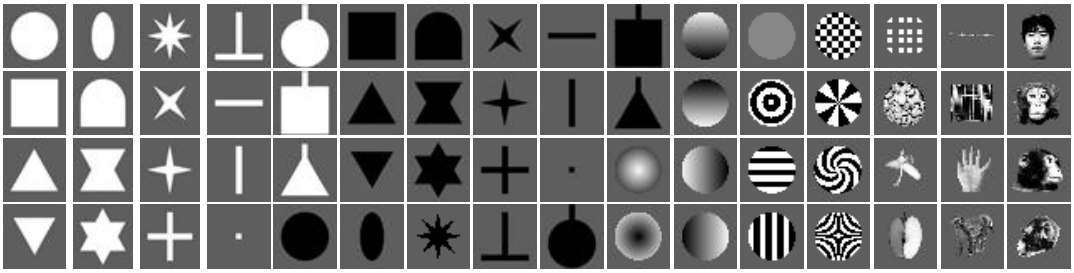

Figure 4: *A set of 64 stimulus images used to examine responses of IT neurons (Tamura et al., 2004, 2005).*

2.4 Self-organizing Maps. The SOM was implemented with the following batch-learning algorithm (Kohonen, 1988):

$$W_{ia} = \sum_j (MH)_{aj} V_{ij} / \sum_j (MH)_{aj} ,$$

where $\mathbf{W} = (W_{ia})$, $\mathbf{M} = (M_{ab})$, and $\mathbf{H} = (H_{bj})$ are basis functions arranged on a two-dimensional topographic map, neighborhood functions, and coefficients (map outputs), respectively, as in the TNMF. Unlike the TNMF, **H** is determined by a winner-take-all mechanism, and there is no non-negative constraint on **W**. We used the same Gaussian neighborhood function as in the TNMF analysis with an annealing method that gradually narrows neighborhood radius. The initial state was set to a

two-dimensional map obtained by PCA (Kohonen, 1988). After training, we further modified SOM outputs by convolving with neighborhood functions **M** (σ = 1) to blur the all-or-nothing (0, 1) outputs along the map.

2.5 Sparseness Measure. We measured the sparseness of the model representation. The sparseness of a data vector **x** with entries x_i is defined as follows (Hoyer, 2004):

$$sparseness(\mathbf{x}) = \frac{\sqrt{n} - (\sum_i |x_i|) / \sqrt{\sum_i x_i^2}}{\sqrt{n} - 1},$$

where n is the dimensionality of **x**. The sparseness ranges from 0 to 1. The value zero means all entries of **x** are equal, while the value one means only a single entry of **x** is non-zero and all the other entries are zeros like a winner-take-all representation. Intermediate values represent other cases between the two extremes.

3 Results

3.1 Basic Properties of TNMF. How do the added cooperation weights in the TNMF affect the representation and coding efficiency when compared to the NMF? Basically the added neighborhood cooperation induces a regularization condition for the set of learned basis functions. This regularization is useful, if the amount of training data is small compared to the latent data complexity. The summation of basis functions over local neighbors leads the network to form new basis functions that are superpositions of input vectors. Therefore, if this a *priori* assumption on the latent data structure is correct, the network can learn faster and better generalizable representations from less training data.

We tested this prediction with a toy data set. We prepared input vectors each of which was generated by a Gaussian component underlying one dimensional space as in a population coding model (Pouget, Zemel, & Dayan, 2000). Specifically, 32 input neurons were arrayed on a line, and their activity distribution on the array formed a Gaussian hill with Poisson noise (Figure 5A):

$$v_i = Poisson(\exp(-\|q_i - \mu\|^2 / 2s^2) / (\sqrt{2\pi} s)),$$

where v_i (i=1, …,32) is an activity of i-th input neuron placed at q_i = (i/32) on the linear array. The function *Poisson* (*) returns a non-negative integer (count) under Poisson noise with mean * count. The variables μ and s are the peak position and the radius of the Gaussian component respectively. The radius s was set to 1/16 (0.0625). The position μ was random, but ranged in 0-0.4 and 0.6-1.0 for training data, and in 0.45-0.55 for test data. Thus, the test data did not contain input patterns in the training data.

The TNMF with a one-dimensional map and the NMF were first applied to the training data (sample size = 400). The test data (sample size = 50) were then fed into the learned model and are reconstructed by the models. We evaluated similarity between the test inputs and their reconstructions by the TNMF or NMF model (1), based on Pearson's correlation coefficient.

We also evaluated similarity between Gaussian components of the test inputs and the reconstructions by the TNMF or NMF model (2), to show how the model learns the latent input structure. These evaluations were performed with changing the number of basis functions of each model.

The results are summarized in Figure 5B. Clearly, the TNMF results surpassed the NMF results in certain optimal model sizes. Moreover, the TNMF model consistently gave better reconstructions for latent Gaussian components (2) than test inputs themselves (1), while the NMF model did not show such a generalizing capability. The reconstructions by the NMF were unstable. This is because the NMF model likely became trapped at a local optimum in an ill-posed problem (note that optimal solution is selected from same number of multiple solutions starting from random initial conditions for both TNMF and NMF).

If TNMF basis functions appropriately capture latent features, their outputs will be sparser than when they do not (provided that the occurrence distribution of latent features is sparse. Figure 5C plots the sparseness (see Methods) of model outputs (vectorized **H**) to the test data against the number of basis functions. In almost every case, TNMF outputs are sparser than those of NMF for each number of basis functions.

Figure 5D shows examples of learned basis functions (columns of **WM**) by the TNMF using 32 basis functions. Each basis function forms a Gaussian-like hill on the input array, and its peak position is smoothly changed with the map topography. More noteworthy is that the middle basis function on the map is elevated around 0.5 on the linear input array coordinate, the missing area of the Gaussian component in the training data.

Although, in this toy problem, only a single Gaussian component appeared in each

input, the TNMF can learn proper latent features from training inputs, each of which is generated by multiple Gaussian components. The TNMF can represent such multi-peaked inputs by multi-peaked outputs on a map, while the SOM derives only single-peaked outputs.

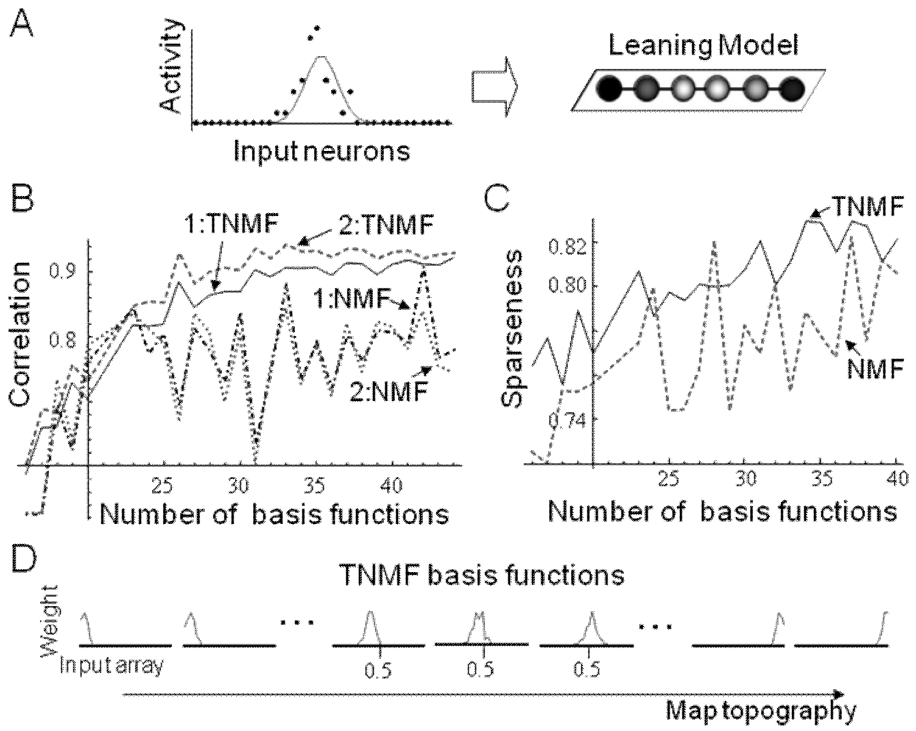

Figure 5: *A toy experiment to illustrate the computational advantage of TNMF over NMF. (A) The left graph shows an example of input data. The input neurons are arrayed in a line and activated by a Gaussian component on the input array with Poisson noise. The position of the Gaussian component ranged in 0-0.4 and 0.6-1.0 for training data, and ranged in 0.45-0.55 for test data. The map architecture of the TNMF was one dimensional as in the right diagram. (B) The correlation between test inputs and their reconstructions by the TNMF or NMF model, as a function of the number of basis functions (1:TNMF, 1:NMF), and that of between the latent Gaussian components and the reconstructions (2:TNMF, 2:NMF). (C) The sparseness of model outputs (vectorized* **H***) to test data. (D) Examples of learned basis functions (columns of* **WM***) by the TNMF using 32 basis functions. Each plot shows weights of each basis function to input neurons, and these plots form a row in order of the one dimensional map topography of the TNMF.*

3.2 TNMF for Rotating Views. We then examined the interpolation property in a situation similar to the toy experiment, except for using real object images. As an input data set, we prepared C2 outputs of the hierarchical model responding to successive views of an object (Figure 6A; 0 to 180 degrees with an interval of 5 degrees). We used the outputs for 0 to 80 degrees and those for 105 to 180 degrees as a training data set, and those for 85 to 100 degrees as a test data set. We then applied the TNMF and NMF to the training data to let them ~~to~~ learn basis functions, and to the test data to evaluate the reconstruction performance. We constructed the TNMF map topography to be a circular ring; we set positions of basis functions to be evenly arranged in the (0, 1) interval, and defined the distance between x and y in the interval as $\min(x-y, x-y-1, x-y+1)$.

Figure 6A shows the input data ($\mathbf{V}=[\mathbf{V}_{training}, \mathbf{V}_{test}]$) and reconstructions ($\mathbf{WMH}_{test}$) for the test data ($\mathbf{V}_{test}$) by the TNMF and NMF with 38 basis functions, visualized by principal component analysis (PCA). In this figure, the input data points are circularly distributed in the PC space as corresponding to the object rotation. Here, we only show the data points of model reconstruction for the test data set, since we see a nearly complete overlap for the training data set to the input in both models. The reconstruction data points of the TNMF are closer to the target points of the test data than those of the NMF. In this case, the mean of Pearson's correlations between test input vectors and their reconstruction vectors was 0.893 for the TNMF and 0.866 for the NMF. The reconstruction data points of the NMF are distributed on the straight line joining points for views of 80 and 105 degrees at the ends of the view changes in the training data. On the contrary, the TNMF reconstruction points run out of the straight line, but follow~~s~~ tangential lines around the view ends of the training data points, as with the test data points. The difference suggests that while the NMF reconstructs the test data by linear interpolation within the training data vectors, the TNMF reconstructs them with nontrivial features capturing non-linear continuity underlying the object images.

Figure 6B shows the mean of correlations between test input vectors and their reconstruction vectors of the TNMF and NMF against the number of basis functions. The mean correlations for the TNMF exceed those for the NMF when the number of basis functions was 15 or larger. This result is similar to the result for the population coded artificial dataset in that the TNMF shows better reconstructions in certain optimal model sizes, and further indicates that the unique interpolation property of the TNMF

also works in a realistic situation. Note that we choose the relative size of sigma in matrix **M** as inversely changing with the number of basis functions. The plot in Figure 6B therefore demonstrates the dependency of the reconstruction error on the chosen value of σ. We also note that the NMF has smaller reconstruction errors than the TNMF for the number of basis function smaller than 15. This is because model parameters of the NMF have a larger degree of freedom than those of the TNMF under the same number of basis functions, and therefore shows only smaller reconstruction errors when the TNMF does not exhibit its distinctive characteristics.

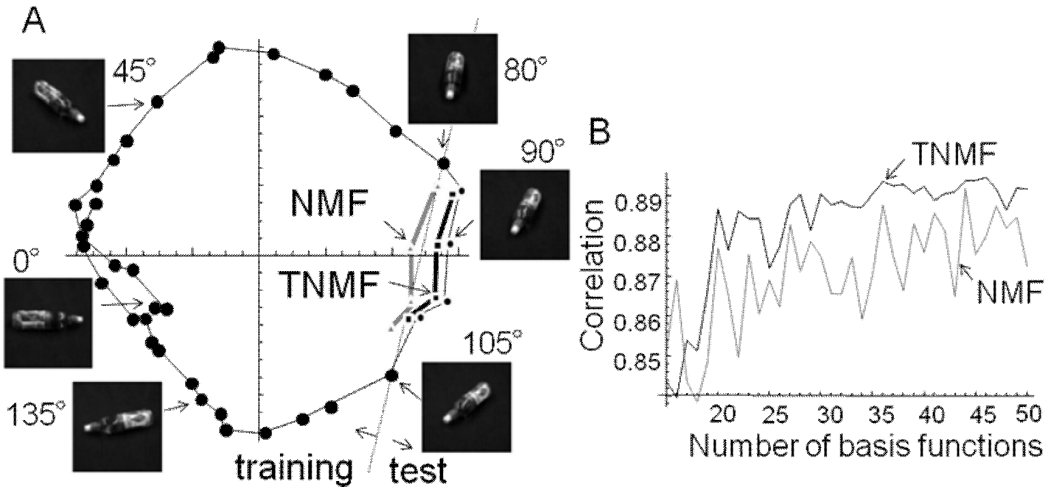

Figure 6: *(A) A PCA visualization for C2 outputs responding to successive views of an object, and their reconstructions obtained by TNMF and NMF analyses for test data. The views in the 0-to-80 and 105-to-180 degrees were the training images. The views in the 85-to-100 degrees were the test images. The map of TNMF was constructed to form a circular ring. In the PC space, the C2 output vectors are represented by circular points connected with black thin lines where lines indicate view changes. The reconstruction vectors of the TNMF and NMF are represented by square points with black thick lines and triangle points with gray thick lines respectively. (B) Mean of correlations between the test data and their reconstructions of the TNMF (black) and NMF (gray) against the number of basis functions.*

3.3 Evaluation of Topological Preservation. To evaluate the functional advantages of the unique interpolation property of TNMF over conventional NMF, we analyzed the degree of topological preservation for view angle parameters when model inputs are affected by noise (see below for details). Specifically, we assessed the consistency between the order of view angles of input images and the spatial relations of the final output vectors. It is beneficial for higher layers to acquire object representations that best preserves view point topology.

As in the previous analysis, the training images were views of an object chosen from 0 to 80 and 105 to 180 degrees with an interval of 5 degrees. The test images were views of the same object in the 90 to 95 degrees with an interval of 1 degree (see Figure 7A). Then, C2 outputs for the test images were modified by adding Poisson noise with the mean of 1 in a multiplicative fashion:

$$v_i' = v_i \times Poisson(1)$$

where v_i is the response of an i-th C2 neuron, and v'_i is the noise-added response.

The TNMF and NMF models were trained based on the C2 outputs of the training images. Then, their performance of topological preservation was tested based on noise-added C2 outputs. The map topology of the TNMF was set to be circular as in the previous analysis.

For each of the TNMF and NMF models, we matched a test output vector with other test output vectors. Specifically, the nearest test output vector with the minimum Euclidean distance was selected for each test output vector. If a view angle for a test output vector is adjacent to the view angle for the nearest one (e.g., 92 and 93 degrees), it means that the model preserves the view topology in its object representation. We refer to this situation as "hit". We calculated the hit rate by 100 repetitions of the test phase generating the noise-added C2 outputs, and by 20 repetitions of the training phase changing the initial conditions of the learning model.

The hit rate for each example object is plotted against the number of basis functions (Figure 7B). The peak hit rate of the TNMF significantly surpassed that of the NMF for the object 1, 2 and 3. The hit rates directly calculated on the noise-added C2 outputs for the object 1, 2 and 3 were 0.42, 0.38 and 0.52 respectively, all of which were lower than the peak hit rates for both TNMF and NMF. These results provide evidence that the TNMF can preserve the view topology more robustly than the NMF for simpler training conditions.

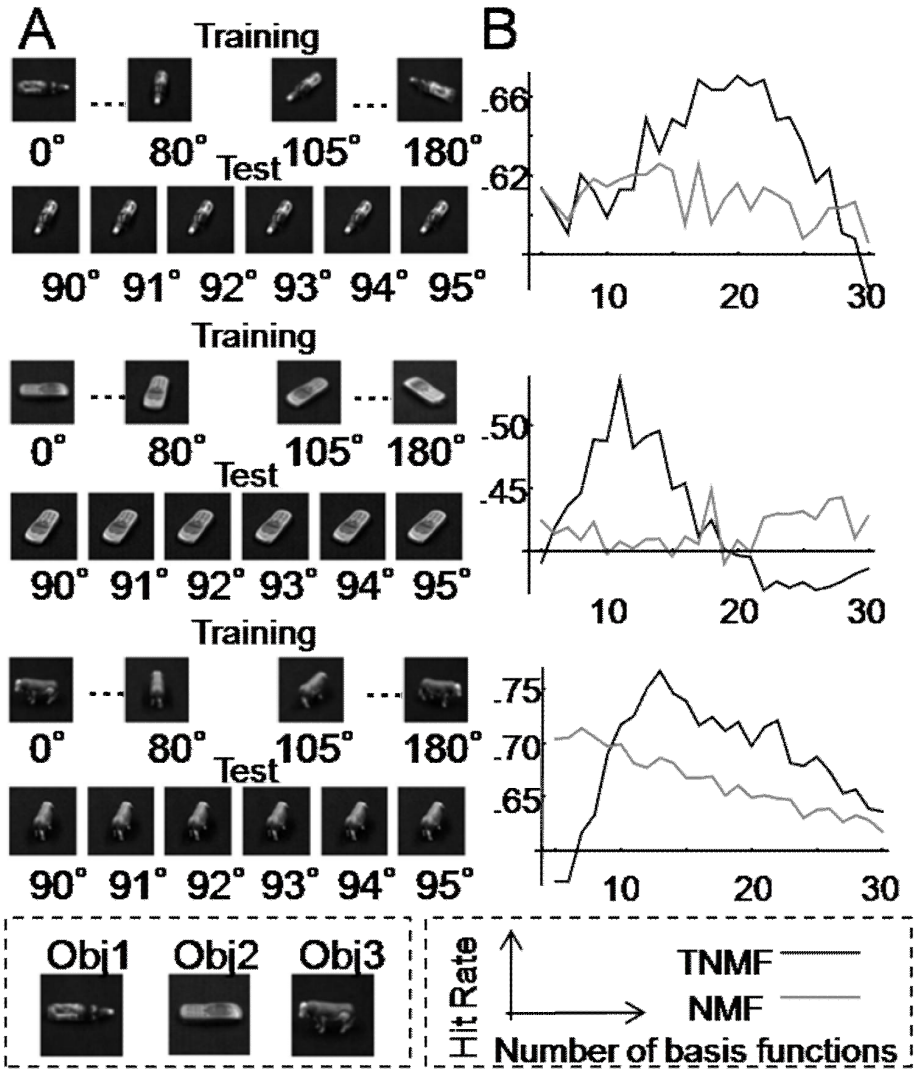

Figure 7: *View angle topological preservation of the TNMF and the NMF models when inputs of the model are affected by noise. The map topology of the TNMF was set to be circular as in Figure 6. (A) Object images for training and testing the models (train: 0 to 80 and 105 to 180 degrees with an interval of 5 degrees; test: 90 to 95 degrees with an interval of 1 degree). (B) The hit rate indicated by the vertical axes is a measure for a consistency between the order of view angles of input images and the spatial relations of final output vectors. The hit rate for each object is plotted against the number of basis functions, where the object is shown in the left side in the same row of the figure.*

3.4 Response Properties of S3 Neurons. Next, we applied the TNMF with a 32×32 square map to the hierarchical model to train the highest layer (S3 layer). The image set for training the S3 layer consists of a set of 250 grey-level photographs (50 objects × 5 views, see Figure 3 for example images). Figure 8A-F shows the map of the S3 layer responses (rows of **H**) to images of three objects, each successively rotated in depth. Spots of different colors indicate the location of the strongest responses to five viewing angles of each object (upper photographs) underlined by bars with the corresponding colors. The distribution of response peaks exhibits several conspicuous features. Each object image elicits multiple, spatially separate, peaks of activity. Response peaks to one view often abut response peaks to other views. Thus, an object with various viewing angles elicits activity in clustered patches. Importantly, activity peaks continuously shift their position within many of these patches (e.g., regions indicated by arrows) as objects are rotated in depth. For example in Figure 8A-D, the cup elicits two gradually shifted patches, the duck elicits one largely changing patch, and the car and the animal elicit several continuously changing patches with different robustness. Other patches do not show this gradual shift or comprise responses to only one or two views.

Each response peak appears to represent a component feature or a part of the image. For example, region 1 was activated by three different views of a cup with an elaborate surface pattern (Figure 8A). These responses disappeared when stimuli were switched to images of a cup with nearly the same shape but a much simpler surface pattern (Figure 8E, dashed circle). We conclude from these results that the responses at region 1 are evoked by the texture pattern on the surface of the cup shown in A. Similarly, masking the left half of the images in Figure 8A eliminated the activities at region 2 evoked by the entire image of the cup (Figure 8F, dashed circle). Even if images of the entire cup were provided to the model, region 2 did not respond when the cup was rotated and the handle was occluded; the patch at region 2 consists of yellow, orange, and pink, but lacks purple and indigo (Figure 8A, D). Therefore, activity at region 2 is evoked by the handle of the cup.

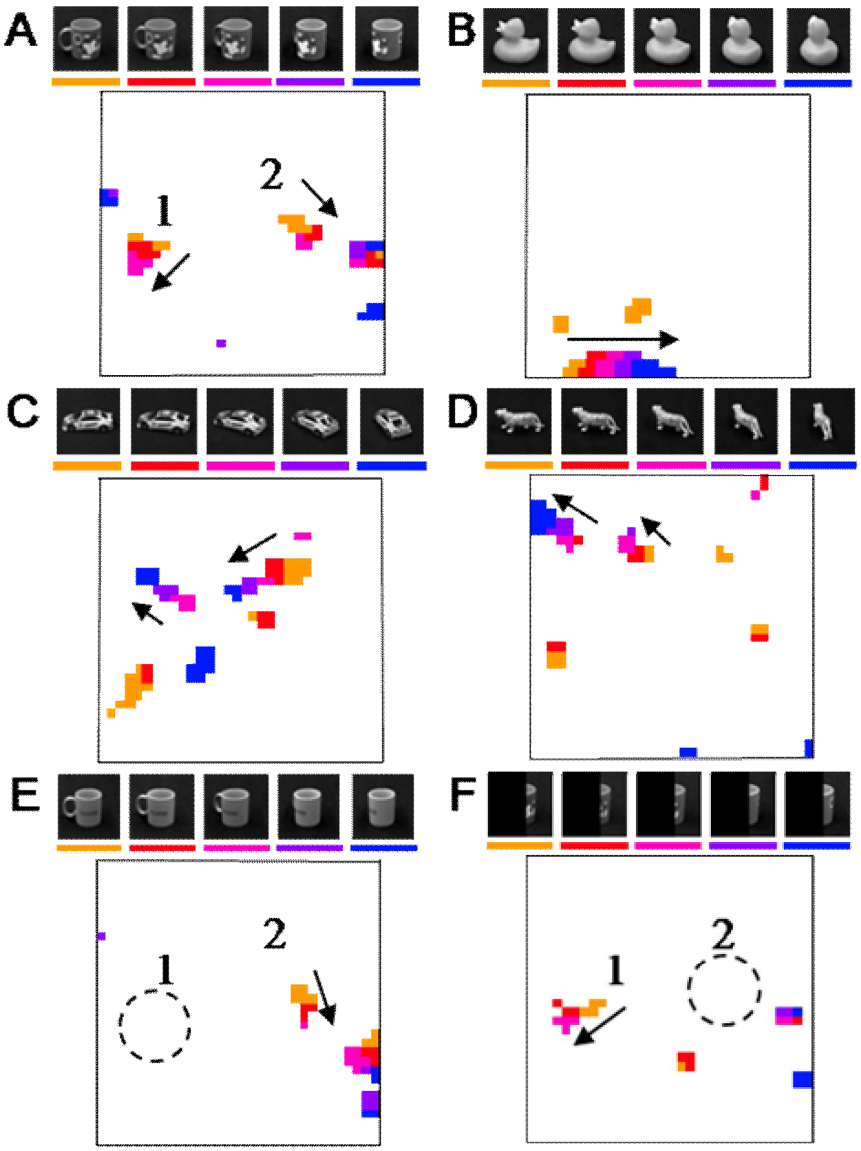

Figure 8: *Object representations (***H***) in the S3 layer trained with the TNMF. Active neurons defined by a common threshold are colored for a corresponding trigger image, but only for the highest activation. If the presented image is changed along the arrow in the top of A or B, some active spots (e.g., regions indicated by arrows) continuously shift their position along the arrows in the map. These results reflect the parts-based and topographic properties of the map.*

We then quantitatively evaluated how the active patches are distributed. First, we computed positions of activity peaks for each map response (each row of **H**). A peak is defined by the maximal element that locally dominates its surroundings, where we only select local maxima, that are larger than 1/10 of the overall global maximum. Figure 9A shows the histogram of the number of peaks for each map response. This figure indicates that about three peaks on average are invoked by a stimulus under a Poisson-like distribution.

Next, we evaluated how far the response peaks continuously shift their position as objects are rotated. First, we chose a set of stimulus pairs, where each pair consists of views of an object being different by $d\theta$ degree. Then for each stimulus pair, the minimum distance was measured between each peak responding to one view and peaks responding to another view. Figure 9B shows the distributions of the minimum peak distance for each set of stimulus pairs whose angle distance ($d\theta$) is 15°, 30°, 45°, or 60° (solid, dash, dot-dash, or dot lines respectively), and for other random stimulus pairs (thick solid line). As can be seen, the peak distance distribution gradually shifts with the angle distance for stimulus pairs. Note that the distance distribution for random pairs also has a tendency to shift toward considerably shorter distance. This is because we used object sets each of which consists of similar objects, for obtaining the stimulus images. These results reveal that certain number of activity peaks continuously shift their position with object rotation.

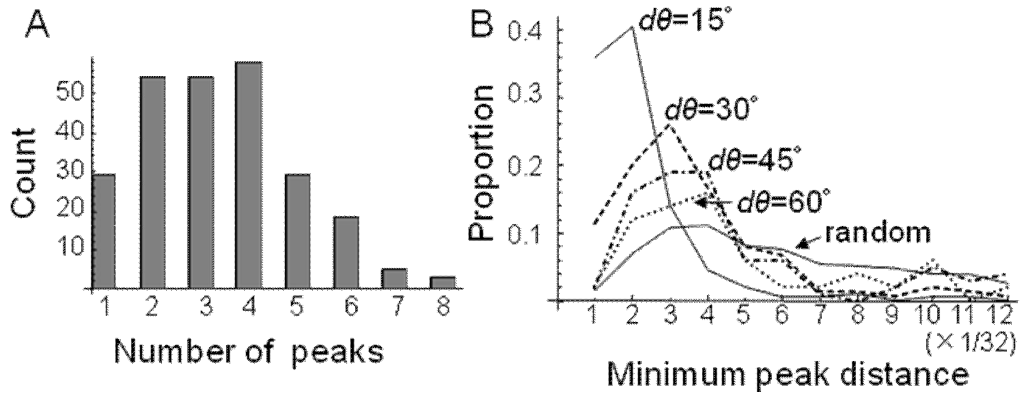

Figure 9: *(A) The histogram of the number of peaks for each map response (each row of* **H***). (B) The distributions of the minimum peak distance for each set of stimulus pairs whose view angle distance is dθ, and for other random stimulus pairs. The minimum peak distance is defined for each stimulus pair as a minimum distance between each peak responding to one view and peaks responding to another view.*

3.5 Basis Functions of S3 Neurons. The response map in the S3 layer is parts-based and locally topographic in a similar fashion as in IT cortex. When we look at the structure of the S3 basis functions convolved with neighborhood functions (columns of **WM**) (Figure 10), adjacent S3 neurons have similar connection weights to C2 neurons (r = 0.89 ± 0.04 for all adjacent pairs, and r = 0.15 ± 0.15 for all pairs), indicating continuous encoding. For each S3 neuron, each of 5×10 arrays in Figure 10, where C2 neurons with same feature type are arranged on the retinotopic location of their receptive fields, connection weights are spatially localized. Therefore, S3 neurons code spatially localized object features, not a holistic pattern of objects. Moreover, connection weights to C2 neurons were spread over a wide area of the 5×10 arrays, consistent with S3 neurons building more complex features than C2 neurons.

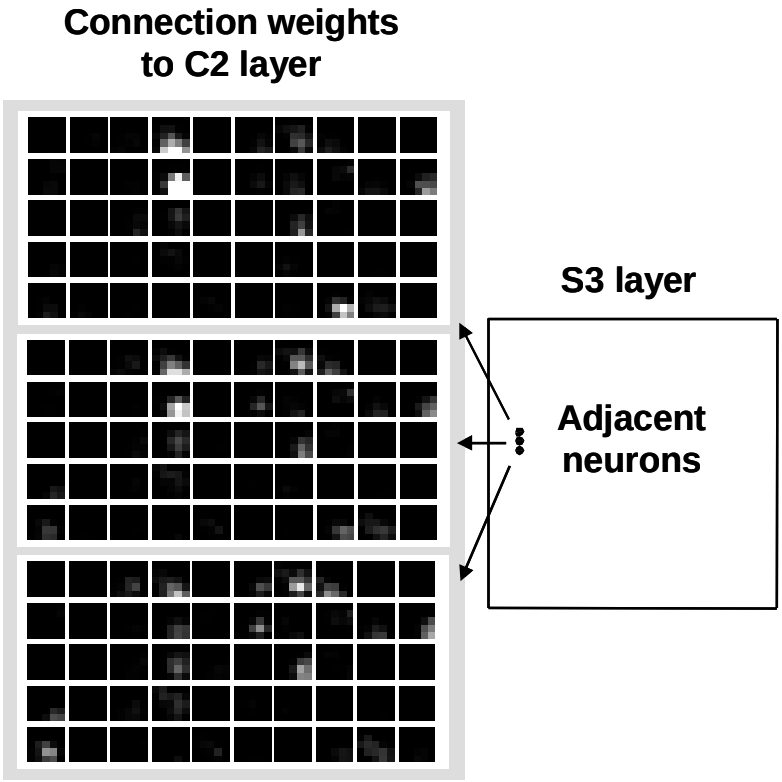

Figure 10: *Connection weights of example adjacent S3 neurons to C2 layer (columns of* **WM***). For each S3 neuron, each of 5×10 arrays represents connection weights to C2 neurons that code each of 50 feature types. The coordinate within each array corresponds to the retinotopy of receptive fields of C2 neurons. Brighter entries indicate stronger magnitudes of the connection weights.*

3.6 Sparseness of S3 Representation. The original NMF has an ability to derive a localized or sparse representation. However, the NMF sometimes derives more global representation than desirable, and some extended models had been proposed to impose explicit sparseness constraints on the NMF (Li, Hou, Zhang, & Cheng, 2001; Hoyer, 2004; Heiler, & Schn□rr, 2006). To clarify if additional sparsity constraints could be beneficial for our TNMF setting, we measured the sparseness (see Methods) of the obtained map components.

In the study by Hoyer (2004), the sparseness of each row of **H** or each column of **W** was fixed to 0.75-0.85 for yielding a local representation. In Heiler et al. (2006), the sparseness was imposed to be greater than 0.6 for a strong sparseness constraint. In our experiments, the sparseness of each column of **H** was 0.84 ± 0.07, and the sparseness of each row of **WM** was 0.68 ± 0.07 (The basis functions in Figure 10 show 0.83, 0.79, and 0.74 sparseness in order, from the top). So, at least in our dataset, the TNMF can derive a comparatively sufficient sparse representation.

3.7 Comparison with IT neurons. In order to examine the biological plausibility of the TNMF learning and the hierarchical model, we compared neuronal responses predicted by the model with those of monkey IT neurons to the 64 object images shown in Figure 4. Importantly, the test figures used for IT neurons (Figure 4) were not presented to the model during training.

For each IT neuron, we selected a model neuron from each layer (C1, C2 and S3) that has the most similar stimulus selectivity:

$$\mathrm{Best}(i, X) = \arg\max_{j}(\mathrm{cor}(\mathbf{z}_i^{(\mathrm{IT})}, \mathbf{z}_j^{X})),$$

where Best(i,X) is the index of the best-fit model neuron in the layer X to the i-th IT neuron, and the vectors of $\mathbf{z}_i^{(\mathrm{IT})}$ and $\mathbf{z}_j^{(X)}$ are, respectively, response patterns of the i-th IT and the j-th model neuron to the 64 images. The function cor(*,*) calculates the Pearson's correlation coefficient. For convenience we define the "best-fit correlation" for each IT neuron:

$$\mathrm{BestCor}(i, X) = \mathrm{cor}(\mathbf{z}_i^{(\mathrm{IT})}, \mathbf{z}_{\mathrm{Best}(i,X)}^{X}).$$

Figure 11A-D show four examples of IT neurons, with their most similar counterparts in the C1, C2, and S3 layers. In the examples A and B, the S3 model neurons display more similar selectivity to the IT neurons than the C2 and C1 model neurons, showing more specific selectivity to star shapes (A) or concentric circles (B). In the example C, both S3 and C2 neurons show selectivity similar to that of the IT neuron preferring shapes with vertical or horizontal lines, while the best-fitting C1 shows poor similarity. In the example D, all S3, C2 and C1 neuron respond selectively to shapes with a small protrusion and a base as a target IT neuron. This example also implies that the higher layers still preserve resolution of coded features despite the increasing of receptive field size. Figure 11 E and F show luminance-contrast selective IT neurons, with the most similar counterparts in the luminance layer (see Methods). Although the luminance layer is strongly simplified, the luminance layer provides better reconstruction to the luminance-contrast selective IT neurons than other shape selective layers.

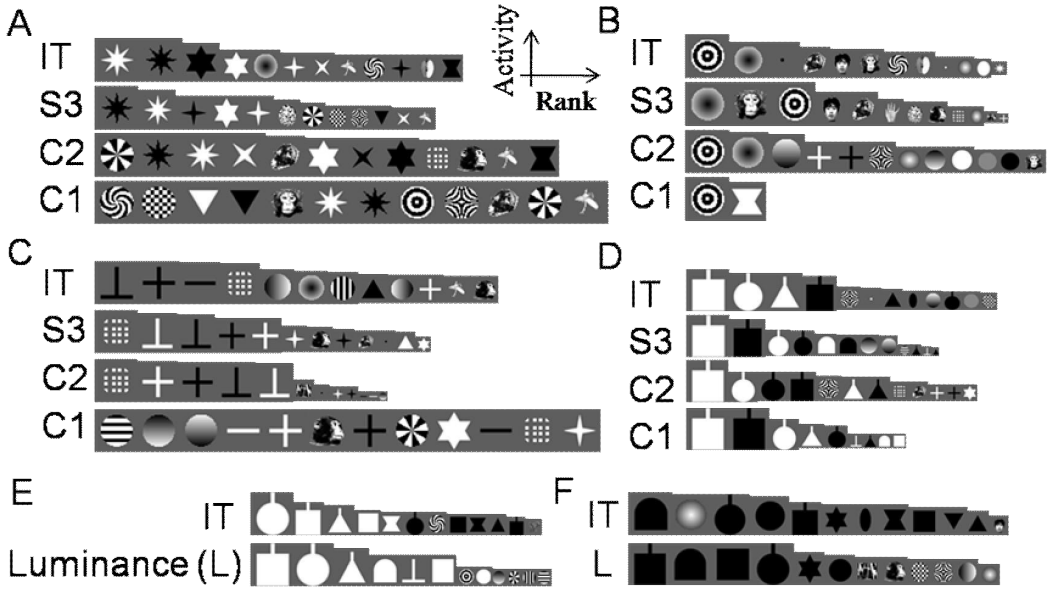

Figure 11: *Examples of selected model neurons of each layer (A-D: from the C1, C2 and S3 layers; E-F: from the Luminance layer) displaying the most similar stimulus selectivity to the IT neurons. Each image sequence indicates the selectivity pattern of each neuron where image size indicates the response magnitude normalized by the maximum and ordered descending.*

We then measured how well the model neurons in different layers reconstruct the stimulus selectivity of IT neurons. The best-fit correlation coefficient used to select the best model neurons is not an appropriate index for comparing different layers, because the response variability across neurons may differ among the model layers. For example, even in cases where model neurons randomly respond to stimuli, if the number of such neurons increases, the best-fit correlation will stochastically increase.

We therefore computed a probabilistic index that measures, for each IT neuron (i), the probability of obtaining the best-fit correlation of each target layer (X) or higher from that layer with a random stimulus-permutation (X^{permute}):

$$P_i = \int_{\text{BestCor}(i,X)}^{1} \Pr[\text{BestCor}(i, X^{\text{permute}}) = r]\, dr \ ,$$

where Pr[*] is the occurrence probability of an event *. Random stimulus permutation was performed by shuffling the stimulus order of the original model responses. Stimulus permutation was applied at the layer level, rather than the single neuron level, because the latter increases the response variability of the model layer.

Such surrogate layers were generated through many repetitions, and the probabilistic index was calculated as the percentile of the original best-fit correlation for a set of surrogate best-fit correlations to each IT neuron. A lower probabilistic index indicates a better reconstruction of responses of an IT neuron. For a population comparison, we defined whether the IT neuron was reconstructed to an arbitrary threshold level of the probabilistic index. We then counted the number of IT neurons out of 497 IT neurons that met two thresholds, 0.1 (strict threshold) and 0.2 (moderate threshold) of the probabilistic index. The higher the number, the more similar object representation of the model layer is to that of IT cortex. Although the evaluation depends on the threshold, it is sufficient and critical to select an appropriate threshold which yields most dissociable results between different target layers.

Figure 12 shows the number of well-reconstructed neurons for different model layers. First, we evaluated a random layer that is a set of IT neural responses with random stimulus-permutation. The performance of the random layer was worse than the other layers with preserved stimulus information. The number for the luminance layer (see Methods) was much higher than that for the random layer, but lower than that for the other layers described below. The C1, C2 and S3 layers were evaluated with incorporating neurons of the luminance layer. This feature alleviates the influence of

luminance-contrast selective IT neurons on the evaluation of shape processing. If an IT neuron exhibits strong selectivity for luminance, the best-fit model neuron was selected from the luminance layer. The number of IT neurons reconstructed by the C2 layer was significantly higher than that by the C1 layer under the strict threshold (Figure 12a) ($p < 0.05$, binomial test). The S3 layer trained by the TNMF exhibited a significantly higher reconstructed number than the C2 score under the moderate threshold (Figure 12b) ($p < 0.05$, binomial test). Thus, higher layers performed better in reconstructing responses of IT neurons than lower layers.

Last, we compared the performance of SOM and NMF for training the S3 layer with the performance of the TNMF (see Figure 12, two bottom bars). The number of basis functions for the TNMF, SOM or NMF analysis was chosen to have an optimal score (TNMF: 32×32, SOM: 24×24, NMF: 1000). Although no significant difference was found between the TNMF- and the NMF-trained layers in this evaluation, the SOM-trained layer displayed a considerably lower number of well-reconstructed neurons than the layers trained by the TNMF or the NMF under both strict and moderate threshold ($p < 0.01$: binomial test). The results indicate the significance of the parts-based property for a biologically plausible model.

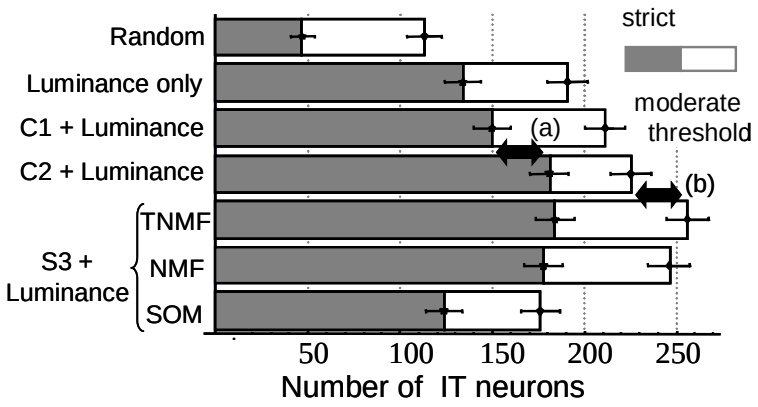

Figure 12: *The number of neurons among 497 IT neurons that are well-reconstructed by each model layer with criteria of strict and moderate thresholds of a probabilistic index that measures the degree of the reconstruction for each IT neuron. The gray and white bars are the number of well-reconstructed neurons assessed with strict and moderate thresholds, respectively. Error bars indicate the standard deviation based on population proportion. Double-headed arrows indicate significant difference between the pointed values ($p < 0.05$; binomial test).*

4 Discussion

In this study, we propose a learning model, TNMF, that accounts for both the parts-based representation and the locally topographic organization of IT in a unified framework. The initial analysis showed the generalizing capability of the TNMF relevant to the topographic extension. Next, we constructed a multi-layer model for visual processing in the ventral pathway, and applied the TNMF to training of the highest layer. The generalizing capability of the TNMF with topographic-induced non-linear interpolation was also confirmed based on hierarchical model responses expressing continuous change of an image. Especially, we showed that the TNMF can generalize objects with more robust topological preservation for a view angle parameter than the NMF. We then showed that the TNMF qualitatively captures the parts-based and topographic properties. Comparison of model neurons with IT neurons at the single neuron level revealed that the higher layers in the model contain more neurons with similar stimulus selectivity to IT neurons than the lower layers. In the highest layer, S3, the TNMF and NMF trained the model neurons to be similar to IT neurons equally well, whereas no positive effect was found with SOM training, indicating the significance of the parts-based representation.

4.1 Structural Features of TNMF. The TNMF is a generative model. A generative model is based on an assumption that observed inputs are generated from a latent structure with noise, and refines the approximation of the latent structure to better represent the inputs. The TNMF as well as the NMF assume that observed inputs are generated by combinations of parts-based elements with noise. An important component in generative models is backward signals used for calculating differences or errors between the actual and reconstructed inputs. In the cortex, forward and backward processing can be performed by interactions between cortical regions and/or between layers. Rao and Ballard (1999) have designed a generative model assuming cortical inter-layer interactions. In the TNMF learning rule, there is also an operation that measures errors between inputs and reconstructions: $(V_{ij}/(WMH)_{ij})$. If the brain uses a learning rule similar to the TNMF, the activity of input neuron i (V_{ij}) might be suppressed by the activities of output neurons (**H**) via the internal networks (the i-th row of **WM**) in a multiplicative fashion to calculate residual errors.

A principal feature of the TNMF is the non-negative constraint. In the original NMF, the non-negative constraint prevents mutual cancellation between basis functions, and yields parts-based representations (Lee & Seung, 1999). The non-negative constraint is

also critical in the TNMF. If the non-negative constraint is not present, the optimal basis functions for the TNMF objective function (**V** ← **WMH**) is $\mathbf{W'M}^{-1}$ where **W**' are the optimal basis functions for the objective function without the neighborhood functions **M** (**V** ← **WH**). This circumstance means that basis functions **W** negate the learning effect of the neighborhood functions **M**, while preserving the degree of freedom in the data reconstruction.

Non-negativity of the NMF has been related to the network properties such as firing rate representation and signed synaptic weight (Lee & Seung, 1999). However, one might argue that there are ubiquitous inhibitory interactions in cortex, which could be modeled as negative entries in the basis **W**. The NMF and TNMF algorithms provide an alternative hypothesis that inhibitory interactions play a role in performing division processes in the iterative dynamics, while non-negative synaptic weights build the basis **W**. Specifically, the division processes can be found in the operation that calculate residual errors ($R_{ij}=V_{ij}/(WMH)_{ij}$), and in the subsequent competitive normalization ($\Sigma_j(MH)_{aj}*R_{ij}/\Sigma_j(MH)_{aj}$) for updating **W**, and that for updating **H**.

Another important component of the TNMF is the set of neighborhood functions **M**. An equivalent of the neighborhood functions in the brain might be intra-area horizontal connections within cortex. Cortical horizontal connections include local recurrent axons forming a dense halo around the origin and long-range horizontal axons with patchy terminal arborization (for IT, see Fujita & Fujita, 1996; Tanigawa, Wang, & Fujita, 2005). The neighborhood functions of the model may be more closely related to local recurrent axons because these axons project in a radially symmetric fashion, similar to the circular neighborhood functions in the TNMF. Long-range horizontal axons might contribute to other functional roles in visual processing, although the rule that dictates their highly irregular, seemingly specific, connections is still unknown (Fujita, 2002). The range of local recurrent axons is larger in IT than in V1 (Tanigawa, Wang, & Fujita, 2005). In the TNMF, the range of neighborhood functions (σ) controls the continuity and resolution of features over the map. If inputs deviate or are more sparsely distributed from their latent continuous structure, it is better to use a larger neighborhood range for accurately capturing the latent continuous structure. Therefore, the range of local recurrent axons in each cortical region may be correlated with the intrinsic difficulty of extracting the latent continuous structure of the underlying inputs.

4.2 Topographic Representations by Other Learning Models. In addition to TNMF, computational models have been proposed to derive distributed input representations with a topographic organization. Among them is a topographic

extension of Independent Component Analysis (ICA) (Hyvärinen & Hoyer, 2001). The classic ICA approximates input signals by linear combinations of signal components that are independent of each other. The topographic ICA relaxes the assumption of the independence among components, and instead assumes that components from adjacent basis functions on a topographic map are dependent on each other under the second-order correlation (correlation on squared components). However, this model still imposes a strong assumption that there should be no first-order correlation between components (normal correlation). This is not the case for IT neurons that display a modest degree of first-order correlation between neighboring neurons (Fujita, Tanaka, Ito, & Cheng, 1992; Gawne & Richmond, 1993; Gochin, Colombo, Dorfman, Gerstein, & Gross, 1994; Wang, Fujita, & Murayama, 2000; Tamura, Kaneko, & Fujita, 2005). Another model is a multi-winner SOM (Schulz & Reggia, 2004, 2005) that narrows the competition range in the SOM. In this model, a region within a competition range has approximately one active peak (winner neuron) regardless of inputs, and thus the total number of active peaks on the topographic map is almost constant across inputs. However, optical imaging of activity in IT reveals that the number of active cortical patches increases with the complexity of the presented stimulus (Tsunoda, Yamane, Nishizaki, & Tanifuji, 2001).

Experimental evidence for locally continuous representation in IT is available only for face-responding columns (Wang, Tanifuji, & Tanaka, 1998). Given this paucity of evidence, Wada and his colleagues (2004) view IT as organized in a globally continuous and locally distributed manner. They proposed an extension of the SOM, and explained the patchy organization and the positive but low first-order correlation between neighboring IT neurons. However, because the model did not consider the parts-based representation of objects, activated neurons were restricted to a single columnar region.

4.3 Role of Topographic Organization. The topographic organization of cortex may reflect the minimization of wiring length in cortical networks (Hubel & Wiesel, 1963; Koulakov & Chklovskii, 2001). Does the topographic organization play any computational or functional role beyond this reason for wiring economy? One possible role would be to allow a population decoding along a topographic map and handle non-negative responses on the map in a population-coding framework (Pouget, Zemel, & Dayan, 2000). In particular, pooling neural responses along a topographic map and estimating the response peak on the map with interpolation would provide more reliable input information than the activity of single neurons. This estimation requires a population-coding framework that permits multiple response peaks (Zemel, Dayan, &

Pouget, 1998; Pasupathy & Connor, 2002; Sahani & Dayan, 2003). Another possible role would be that non-negative responses on the topographic map can be pooled along the map with an OR-operation to yield higher response invariance, as in the C1 and C2 layers. For example, the S3 layer trained by the TNMF can be naturally extended to include a "C3" layer in which neurons will show more view-invariant responses. These post-processing functions cannot be naturally established without topographically organized representations, or on representations with negative values.

4.4 Biological Plausibility. Model neurons generated by the SOM poorly reproduced the response properties of IT neurons. Their performance was worse than model neurons generated with the NMF and the C1, C2, S3 layers of the TNMF (see Figure 12). A possible reason for this is that features emerging with the SOM were holistic patterns specialized for individual training images. In contrast, most IT neurons respond to partial features of input images (Tanaka, Saito, Fukada, & Moriya, 1991). The S3 layer trained with the TNMF and the NMF parallels IT neuronal responses better than the C1 and C2 layers. Thus, the TNMF and the NMF generate generalized higher-level parts-based features in a similar way as individual IT neurons encode object features. Theoretically, the features emerging with the TNMF are different from those with the NMF in that the TNMF considers the continuity of the learnt features over the map, and extracts the latent continuous structure underlying training inputs in an interpolating manner. No significant difference was found just for the comparison in Figure 12. There is still the strong difference with regard to topography and population coding. Further studies will be required for a more sensitive evaluation of the continuous interpolation in TNMF learning.

In conclusion, we propose a new learning rule that can successfully capture two features of the higher object representation in primate visual cortex, parts-based representation and locally topographic organization.

Acknowledgments

This work is supported by grants from the Ministry of Education, Culture, Sports, Science and Technology (MEXT) to MW (17022015) and IF (170220025), and Core Research of Evolutional Science and Technology (CREST) of the Japan Science and Technology Agency to IF.

References

Buchsbaum, G. & Bloch, O. (2002). Color categories revealed by non-negative matrix factorization of Munsell color spectra. *Vision Research*, 42, 559–563.

Cho, Y. C. & Choi, S. J. (2005). Nonnegative features of spectro-temporal sounds for classification. *Pattern Recognition Letters*, 26, 1327–1336.

Durbin, R. & Mitchison, G. (1990). A dimension reduction framework for understanding cortical maps. *Nature*, 343, 644–647.

Fujita, I. (2002). The inferior temporal cortex: architecture, computation, and representation. *Journal of Neurocytology*, 31, 359–371.

Fujita, I. & Fujita, T. (1996). Intrinsic connections in the macaque inferior temporal cortex. *Journal of Comparative Neurology*, 368, 467–486.

Fujita, I., Tanaka, K., Ito, M., & Cheng, K. (1992). Columns for visual features of objects in monkey inferotemporal cortex. *Nature*, 360, 343–346.

Fukushima, K. (1980). Neocognitron: a self organizing neural network model for a mechanism of pattern recognition unaffected by shift in position. *Biological Cybernetics*, 36, 193–202.

Gawne, T. J., & Richmond, B.J. (1993). How independent are the messages carried by adjacent inferior temporal cortical neurons? *Journal of Neuroscience*, 13, 2758–2771.

Gochin, P. M., Colombo, M., Dorfman, G.. A., Gerstein, G. L., & Gross, C. G.. (1994). Neural ensemble coding in inferior temporal cortex. *Journal of Neurophysiology*, 71, 2325–2337.

Gross, C. G. (1994). How inferior temporal cortex became a visual area. *Cerebral Cortex*, 5, 455-469.

Heiler, M. & Schn□rr, C. (2006). Learning sparse representations by non-negative matrix factorization and sequential cone programming. *Journal of Machine Learning Research*, 7, 1385–1407.

Hubel, D. H. & Wiesel, T. N. (1962). Receptive fields, binocular interaction and functional architecture in the cat's visual cortex. *Journal of Physiology*, 160, 106–154.

Hubel, D. H. & Wiesel, T. N. (1963). Shape and arrangement of columns in cat's striate cortex. *Journal of Physiology*, 165, 559–568.

Hoyer, P. O. (2004). Non-negative matrix factorization with sparseness constraints. *Journal of Machine Learning Research*, 5, 1457–1469.

Hyvärinen, A., Hoyer, P. O., & Inki, M. (2001). Topographic independent component

analysis. *Neural Computation*, 13, 1527–1558.

Ito, M., Fujita, I., Tamura, H., & Tanaka, K. (1994). Processing of contrast polarity of visual images in inferotemporal cortex of the macaque monkey. *Cerebral Cortex*. 5,499–508.

Kohonen, T (1998). Self-Organization and Associative Memory. *Springer-Verlag, Berlin*.

Koulakov, A. A. & Chklovskii, D. B. (2001). Orientation preference patterns in mammalian visual cortex: a wire length minimization approach. *Neuron*, 29, 519–527.

Lee, D. D. & Seung, H. S. (1999). Learning the parts of objects by non-negative matrix factorization. *Nature*, 401, 788–791.

Lee, D. D. & Seung, H. S. (2001). Algorithms for non-negative matrix factorization. *In Advances in Neural Information Processing Systems*, Volume 13, pp. 556–562. The MIT Press.

Li, S. Z., Hou, X., Zhang, H., & Cheng, Q. (2001). Learning spatially localized parts-based representations. *Proc. IEEE Conf. on Computer Vision and Pattern Recognition (CVPR), USA*, I, 207–212.

Mishkin, M., Ungerleider, E., & Macko, K. A. (1983). Object vision and spatial vision: two cortical pathways. *Trends in Neuroscience*, 6, 414–417.

Obermayer, K., Ritter, H., & Schulten, K. (1990). A Principle for the Formation of the Spatial Structure of Cortical Feature Maps. *Proc. Natl. Acad. Sci. USA*, 87, 8345–8349.

Pasupathy A. & Connor, C. E. (2002). Population coding of shape in area V4. *Nature Neuroscience*, 5, 1332–1338.

Pouget, A.., Dayan, P., & Zemel, R. (2000). Information processing with population codes. *Nature Review Neuroscience*, 1, 125–132.

Rao, R. P. N. & Ballard, D. (1999). Predictive coding in the visual cortex: a functional interpretation of some extra-classical receptive-field effects. *Nature Neuroscience*, 2, 79–87.

Riesenhuber, M. & Poggio, T. (1999). Hierarchical models of object recognition in cortex. *Nature Neuroscience*, 2, 1019–1025.

Sahani, M. & Dayan, P. (2003). Doubly distributional population codes: simultaneous representation of uncertainty and multiplicity. *Neural Computation*, 15, 2255–2279.

Schulz, R. & Reggia, J. A. (2004). Temporally asymmetric learning supports sequence processing in multi-winner self-organizing maps. *Neural Computation*, 16, 535–561.

Schulz, R. & Reggia, J. A. (2005). Mirror symmetric topographic maps can arise from activity-dependent synaptic changes. *Neural Computation*, 17, 1059–1083.

Serre, T., Kouh, M., Cadieu, C., Knoblich, U., Kreiman, G., & Poggio, T. (2005). A theory of object recognition: computations and circuits in the feedforward path of the ventral stream in primate visual cortex. *CBCL Paper 259/AI Memo 2005–036*, Massachusetts Institute of Technology, Cambridge, MA.

Serre, T. (2006). Learning a dictionary of shape components in visual cortex: Comparison with neurons, humans and machines. *Ph.D. thesis*, Massachusetts Institute of Technology, Cambridge, MA.Swindale, N. V. (1991).

Swindale, N. V., 1991. Coverage and the design of striate cortex. *Biological cybernetics*. 65, 415–424.

Tamura, H., Kaneko, H., & Fujita, I. (2005). Quantitative analysis of functional clustering of neurons in the macaque inferior temporal cortex. *Neuroscience Research*, 52, 311–322.

Tamura, H., Kaneko, H., Kawasaki, K., & Fujita, I. (2004). Presumed inhibitory neurons in the macaque inferior temporal cortex: visual response properties and functional interactions with adjacent neurons. *Journal of Neurophysiology*, 91, 2782–2796.

Tanaka, K. (2003). Columns for complex visual object features in the inferotemporal cortex: clustering of cells with similar but slightly different stimulus selectivities. *Cerebral Cortex*, 13, 90–99.

Tanaka, K., Saito, H., Fukada, Y., & Moriya, M. (1991). Coding visual images of objects in the inferotemporal cortex of the macaque monkey. *Journal of Neurophysiology*, 66, 170–189.

Tanigawa, H., Wang, Q. X., & Fujita, I. (2005). Organization of horizontal axons in the inferior temporal cortex and primary visual cortex. *Cerebral Cortex*, 15, 1887–1899.

Tsunoda, K., Yamane, Y., Nishizaki, M., & Tanifuji, M. (2001). Complex objects are represented in macaque inferotemporal cortex by the combination of feature columns. *Nature Neuroscience*, 4, 832–838.

Wada, K., Kurata, K., & Okada, M. (2004). Self-organization of globally continuous and locally distributed information representation. *Neural Networks*, 17, 1039-1049.

Wang, G., Tanifuji, M., & Tanaka, K. (1998). Functional architecture in monkey inferotemporal cortex revealed by in vivo optical imaging. *Neuroscience Research*, 32, 33–46.

Wang, Y., Fujita, I., & Murayama, Y. (2000). Neuronal mechanisms of selectivity for object features revealed by blocking inhibition in inferotemporal cortex. *Nature Neuroscience*, 3, 807-813.

Wersing, H. & Körner, E. (2003). Learning optimized features for hierarchical models of invariant object recognition. *Neural Computation*, 15, 1559–1588.

Xu, W., Liu, X., & Gong Y. (2003). Document-clustering based on non-negative matrix factorization. *In Proceedings of SIGIR 03*, Toronto, pp. 267–273.

Yu, H., Farley, B., Jin, D., & Sur, M. (2005). The coordinated mapping of visual space and response features in visual cortex. *Neuron*, 47, 267–280.

Zemel, R. S., Dayan, P., & Pouget, A.. (1998). Probabilistic interpretation of population codes. *Neural Computation*, 10, 403–430.

Zoccolan, D., Kouh, M., Poggio, T. & DiCarlo, J. J. (2007). Trade-off between object selectivity and tolerance in monkey inferotemporal cortex. *Journal of Neuroscience*, 27, 12292–12307.