A mixture of sparse coding models explaining properties of face neurons related to holistic and parts-based processing

Haruo Hosoya [1, ¤, *], Aapo Hyvärinen[2, 3]

1 Cognitive Mechanisms Laboratories, ATR International, Kyoto, Japan
2 Department of Computer Science and HIIT, University of Helsinki, Helsinki, Finland
3 Gatsby Computational Neuroscience Unit, University College London, UK

¤2–2–2 Hikaridai, Keihanna Science City, Kyoto, Japan, 619–0288
*** E-mail (corresponding author): hosoya@atr.jp**

Abstract

Experimental studies have revealed evidence of both parts-based and holistic representations of objects and faces in the primate visual system. However, it is still a mystery how such seemingly contradictory types of processing can coexist within a single system. Here, we propose a novel theory called mixture of sparse coding models, inspired by the formation of category-specific subregions in the inferotemporal (IT) cortex. We developed a hierarchical network that constructed a mixture of two sparse coding submodels on top of a simple Gabor analysis. The submodels were each trained with face or non-face object images, which resulted in separate representations of facial parts and object parts. Importantly, the latent variables were estimated by Bayesian inference to model evoked neural activities, which had a top-down explaining-away effect that enabled recognition of an individual part to depend strongly on the category of the whole input. We show that this explaining-away effect was indeed crucial for the units in the face submodel to exhibit significant selectivity to face images compared to object images. Furthermore, the model explained, qualitatively and quantitatively, several tuning properties to facial features found in the middle patch of face processing in the macaque IT cortex as documented by Freiwald, Tsao, and Livingstone (2009). These included, in particular, tuning to only a small number of facial features that were often related to geometrically large parts like face outline and hair, preference and anti-preference of extreme facial features (e.g., very large/small inter-eye distance), and reduction of the gain of feature tuning for partial face stimuli compared to whole face stimuli. Thus, we hypothesize

that the coding principle of facial features in the middle patch of face processing in the macaque IT cortex may be closely related to mixture of sparse coding models.

Introduction

The variety of objects that we see everyday in this world is overwhelming. How our visual system deals with such complexity in representing objects is a long-standing problem. Classical psychology on object representation has often debated on the two opposing hypotheses: parts-based and holistic processing [1]. While the parts-based processing hypothesis assumes that an object is represented as a combination of individual parts, the holistic processing hypothesis postulates that an object is perceived as a whole. Experimental studies have revealed evidence of both types of processing in behavior [1, 2] and in neural activities in higher visual areas [2–5]. Although these data favored holistic representation for faces and parts-based representation for non-face objects [1, 2, 5], neither category seemed to completely rely on only one of the processing types. However, if we assume that the brain has both kinds of computation, then how could a single system reconcile the two seemingly contradictory types of processing? And how would it be related to the computation in the inferotemporal (IT) cortex?

In this study, we propose a novel theoretical framework, called mixture of sparse coding models, to study the computational principles underlying face and object processing in the IT cortex. Similarly to a classical mixture of Gaussians [6], our model describes data coming from a fixed number of categories, where each category is defined by a sparse coding model [7]. Mixture of sparse coding models was motivated by the following three observations. First, such a clustering model is in line with the general fact that IT neurons form specialized subregions for important categories such as faces, body parts, and objects [8–10]. Second, sparse coding often brings about parts-based feature representations when applied to an image set of a specific category (cf. [11, 12]). Third, when multiple sparse coding models are combined in a mixture and Bayesian inference is performed, the emerging computation is surprising: not only recognition of the whole object depends on the individual parts, but also recognition of a part depends on the whole. This is in fact a Bayesian explaining-away effect: an input image is first independently interpreted by each sparse coding submodel, but then the mixture model adopts the best interpretation and dismisses the other interpretations. For example, even if an input image contains a potential facial feature (e.g., a half-moon-like shape ◡), that feature would not be recognized as an actual facial feature (e.g., a mouth) if it appears in a non-face object (Figure 1A).

To elucidate the relevance of our theory to computation in the IT cortex, we constructed a concrete hierarchical network that started with a simple fixed Gabor-analysis stage and proceeded to a mixture of two sparse coding submodels. The submodels were each trained with natural face or non-face object images by a sparse-coding learning algorithm, resulting in separate dictionaries of facial

parts or object parts, respectively. Then, we found that our model face cells exhibited prominent selectivity to face images compared to non-face object images and that this selectivity was crucially dependent on the above-mentioned explaining-away effect. Furthermore, these model cells explained well a number of response properties of face neurons in a region of the macaque IT cortex called the face middle patch, documented by Freiwald et al. [4]. In particular, our model face cells tended to (1) be tuned to only a small number of facial features, often related to geometrically large parts such as face outline and hair, (2) prefer one extreme for a particular facial feature while anti-prefering the other extreme, and (3) reduce the gain of tuning when a partial face was presented compared to a whole face. We quantified these properties and compared these with the experimental data at the population level [4]; the result showed a good match. Thus, we propose the hypothesis that regions of the IT cortex representing objects or faces may employ a computational principle similar to mixture of sparse coding models.

Results

Mixture of sparse coding models

We assume an observed variable $\mathbf{x} : \mathcal{R}^D$, a (discrete) hidden variable $k : \{1, 2, \ldots, K\}$, and K hidden variables $\mathbf{y}^h : \mathcal{R}^M$ $(h = 1, 2, \ldots, K)$. Intuitively, the variable $\mathbf{x}$ represents an input image (in fact, outputs from Gabor analysis in our hierarchical model), the variable k represents the index of an image class (submodel), and each variable $\mathbf{y}^h$ represents features for the class h.

We define the generative process of these variables as follows (Figure 1B). First, an image class k is drawn from a pre-fixed prior $\pi_h : [0, 1]$ (where $\sum_h \pi_h = 1$):

$$P(k) = \pi_k \tag{1}$$

We call k the generating class. Next, features $\mathbf{y}^k$ for the class k are drawn from the Laplace distribution with mean $\mathbf{b}^k : \mathcal{R}^M$ and a pre-fixed standard deviation λ (common for all dimensions)

$$P(\mathbf{y}^k \mid k) = \mathcal{L}(\mathbf{y}^k \mid \mathbf{b}^k, \lambda) = \prod_m \frac{1}{2\lambda} \exp\left(-\frac{|y_m^k - b_m^k|}{\lambda}\right) \tag{2}$$

and an observed image $\mathbf{x}$ is generated from the features $\mathbf{y}^k$ by transforming it by the basis matrix $\mathbf{A}^k : \mathcal{R}^{D \times M}$, with a Gaussian noise of a pre-fixed variance σ^2 added:

$$P(\mathbf{x} \mid \mathbf{y}^k, k) = \mathcal{N}(\mathbf{x} \mid \mathbf{A}^k \mathbf{y}^k, \sigma^2 I) \tag{3}$$

Here, $\mathbf{A}^k$ and $\mathbf{b}^k$ are model parameters estimated from data (see next section). Features $\mathbf{y}^h$ for each non-generating class $h \neq k$ are drawn from the zero-mean Laplacian

$$P(\mathbf{y}^h \mid k) = \mathcal{L}(\mathbf{y}^h \mid 0, \lambda) \tag{4}$$

Figure 1. (A) Cartoon face and boat. Note that the mouth of the face and the base of the boat are the same shapes. (B) The graphical diagram for a mixture of sparse coding models. The variable k is first drawn from its prior, then each variable $\mathbf{y}^h$ is draw from a Laplace distribution depending on whether $h = k$ or not, and finally the variable $\mathbf{x}$ is generated from a Gaussian distribution depending on $\mathbf{y}^k$. (Note that, until k is determined, $\mathbf{x}$ is dependent on k and all of $\mathbf{y}^1, \mathbf{y}^2, \ldots, \mathbf{y}^K$.) (C) An energy detector model. (D) A mixture of two sparse coding models for faces and objects, built on top of an energy detector bank.

and never used for generating $\mathbf{x}$. Altogether, the model distribution is rewritten as follows:

$$P(\mathbf{x}, \mathbf{y}^1, \mathbf{y}^2, \ldots, \mathbf{y}^K, k) = \mathcal{N}(\mathbf{x} \mid \mathbf{A}^k \mathbf{y}^k, \sigma^2 I) \mathcal{L}(\mathbf{y}^k \mid \mathbf{b}^k, \lambda) \left[\prod_{h \neq k} \mathcal{L}(\mathbf{y}^h \mid 0, \lambda) \right] \pi_k \tag{5}$$

Since data are generated from the mixture of K distributions each of which is a combination of a Laplacian and a Gaussian similar to the classical sparse coding model [13], we call the above framework mixture of sparse coding models.

However, we depart from standard formulation of mixture models or sparse coding in two ways, motivated for modeling face neurons. First, since the feature variable $\mathbf{y}^h$ for the non-generating classes $h \neq k$ are unused for generating $\mathbf{x}$, a standard formulation would simply drop the factor (4), leaving $\mathbf{y}^h$ unconstrained. However, our goal here is to model the responses of all (face or object) neurons for all stimuli (faces or objects). In fact, actual face neurons are normally strongly activated by face stimuli, but are deactivated by non-face stimuli, which is why our model uses a zero mean for non-generating feature variables. Second, the classical sparse coding uses a zero-mean prior [13], which is suitable for natural image patch inputs since their mean is zero (blank image) and this evokes no response like V1 neurons. However, the mean of face images is not zero and such mean face image usually elicits non-zero responses of actual face neurons. Therefore our model uses a prior with potentially non-zero mean $\mathbf{b}^k$ on the feature variable $\mathbf{y}^k$ for the generating class.

Given an input $\mathbf{x}$, how do we infer the hidden variables $\mathbf{y}^h$? Since evoked response values of neurons that are experimentally reported are usually the firing rates averaged over trials, we model these quantities as posterior expectations of the hidden variables. Since exact computation of those values would be too slow, we use the following approximation (see the derivation in the section on Approximating posterior in Methods).

1. For each image class k, compute the MAP (maximum a posteriori) estimates of the feature variables $\mathbf{y}^1, \mathbf{y}^2, \ldots, \mathbf{y}^K$, conditioned on the class k:

$$(\hat{\mathbf{y}}^1(k), \hat{\mathbf{y}}^2(k), \ldots, \hat{\mathbf{y}}^K(k)) = \underset{\mathbf{y}^1, \mathbf{y}^2, \ldots, \mathbf{y}^K}{\operatorname{argmax}} P(\mathbf{y}^1, \mathbf{y}^2, \ldots, \mathbf{y}^K, k \mid \mathbf{x}) \tag{6}$$

2. Compute the approximate posterior probability of each image class k:

$$r_k = \frac{P(\hat{\mathbf{y}}^1(k), \hat{\mathbf{y}}^2(k), \ldots, \hat{\mathbf{y}}^K(k), k \mid \mathbf{x})}{\sum_h P(\hat{\mathbf{y}}^1(h), \hat{\mathbf{y}}^2(h), \ldots, \hat{\mathbf{y}}^K(h), h \mid \mathbf{x})} \tag{7}$$

3. Compute the approximate posterior expectation of each feature variable k:

$$\hat{\mathbf{y}}^k = \sum_h r_h \hat{\mathbf{y}}^k(h) \tag{8}$$

Note that, in equation (6), the feature variables for non-selected classes are always exactly zero:

$$\hat{\mathbf{y}}^h(k) = 0 \quad \text{for} \quad h \neq k. \tag{9}$$

Therefore, even though an alternative approach would be to model neural responses by the MAP estimates of feature variable for the best image class, this may be too radical since responses becoming absolutely zero are a little unnatural.

The following is a more concrete version of the above algorithm (derived using the model definition (5) and the property (9)).

1. For each k: $\hat{\mathbf{y}}^k \leftarrow \mathrm{argmax}_{\mathbf{y}^k} L_k(\mathbf{y}^k \mid \mathbf{x})$ where $L_k(\mathbf{y}^k \mid \mathbf{x}) = -\frac{1}{2\sigma^2}\|\mathbf{x} - \mathbf{A}^k\mathbf{y}^k\|^2 - \frac{1}{\lambda}\sum_m |y^k_m - b^k_m|$.

2. For each k: $r_k \leftarrow \frac{\pi_k \exp(L_k(\hat{\mathbf{y}}^k|\mathbf{x}))}{\sum_h \pi_h \exp(L_h(\hat{\mathbf{y}}^h|\mathbf{x}))}$

3. For each k: $\hat{\mathbf{y}}^k \leftarrow r_k \hat{\mathbf{y}}^k$

Note that, in step 1, the feature variable for each class is estimated in a similar way to the usual sparse coding inference. Those estimated values are, in step 3, multiplied by the probability indicating how well each class interprets the input (calculated in step 2). Therefore, even if the input contains a feature that can potentially activate variables of submodel k, they will be eventually deactivated if the submodel does not interpret the whole input well compared to the other submodels.

Hierarchical model for face and object areas

To study face and object processing in the IT cortex, we applied the mixture of sparse coding models as the top layer in a multi-layer network, analyzing the output of a fixed bank of standard energy detector models (Figure 1D). The energy model received an image of 64×64 pixels and each energy detector computed the squared norm of the outputs from two Gabor filters applied to the image (Figure 1C). The two filters had the same center position, orientation, and spatial frequency, but had phases different by 90°. The entire bank of energy detectors had all combinations of 10×10 center positions (in a grid layout), 8 orientations, and 3 frequencies; thus the output of this stage had a total of 2400

dimensions. Those outputs were then sent to the mixture model, which had two sparse coding submodels, called face submodel and object submodel. The feature variable of each submodel had 400 dimensions; we call each dimension a model neuron or a unit. After the inference computation of the mixture model as described in the last section, the resulting value of each unit was passed to the smooth half-rectification function $h(a) = \log(1 + \exp(a))$ to produce non-negative values for the comparison with neural responses. (See the section on Model details in Methods.) In the actual visual cortex, inputs to IT areas are presumably computed between V1 and V4 and this computation must be much more complex than the energy detector bank in our model. However, some important aspects should still be reflected by this simple operation since a large number of V4 neurons are known to be orientation-selective [14]; moreover, this simple assumption was sufficient to reproduce certain response properties of face neurons as shown in what follows.

The mixture model was trained with face and object images processed by the energy detector stage. Classical mixture models are usually trained with an unsupervised learning method [6]. However, such learning is generally not easy and not our main interest here since we focus on inference (i.e., on computation of evoked responses, not on learning or plasticity). Therefore we simplified learning here by explicitly using class labels, either "face" or "object," to train each submodel separately with a sparse coding method. More concretely, we used publicly available face and object image datasets in which the faces or objects were properly aligned within each image frame [12,15,16] (see the section on Data preprocessing in Methods). After processing the images with the energy detectors, we first subtracted, from each data, the dimension along the mean of all (face and object) data:

$$\mathbf{x} \leftarrow \mathbf{x} - \frac{\bar{\mathbf{x}}\bar{\mathbf{x}}^{\mathsf{T}}\mathbf{x}}{\|\bar{\mathbf{x}}\|^2} \tag{10}$$

Although this operation was not quite essential, this had the effect of a (linear) form of contrast normalization suppressing a part of inputs with prominently strong signals; in fact, without this operation, some elements of $\mathbf{b}^k$ (estimated below) became outrageously large. Then, for each image class k, we learned the basis matrix $\mathbf{A}^k$ and the mean activities $\mathbf{b}^k$ by a sparse coding of the corresponding dataset. Although we could have directly estimated these by the classical sparse coding method [7], we instead used our previously developed approach [17] in the following two steps:

1. perform strong dimension reduction using principal component analysis (PCA) [18] from 2400 to 100 dimensions while whitening;

2. apply overcomplete independent component analysis (ICA) [19] to estimate 400 components from 100 dimensions.

The first step is a minor modification of a standard preprocessing used in any classical sparse coding or ICA methods. However, we have previously discovered that a drastic reduction of input dimensions has an effect of spatial pooling [18] and thereby produces much larger basis patterns than without it [17]. Indeed,

in the present case, we later show that weaker dimension reduction resulted in representations of overly small features, which led to a loss of discriminative power. In the second step, we used the overcomplete ICA as an approximation of sparse coding [13], where we adopted the score matching method for efficient computation [19].

Formally, if we write $\mathbf{d}^k$ for the vector of top 100 eigenvalues sorted in descending order, $\mathbf{E}^k$ for the matrix of the corresponding (row) eigenvectors, and $\mathbf{R}^k$ for the weight matrix estimated by the overcomplete ICA, then the estimated filter matrix is written as

$$\mathbf{W}^k = \mathbf{R}^k \operatorname{diag}(\mathbf{d}^k)^{-1/2}\mathbf{E}^k. \tag{11}$$

Finally, we calculated the basis matrix $\mathbf{A}^k = (\mathbf{W}^k)^{\#}$ (# is the pseudo inverse) and $\mathbf{b}^k = \mathbf{W}^k\bar{\mathbf{x}}^k$ (where $\bar{\mathbf{x}}^k$ is the mean of all data of class k). Note that the signs of the filter vectors obtained from ICA are arbitrary; for the present purpose, we adjusted each sign so that all elements of $\mathbf{b}^k$ are non-negative. For simplicity, we fixed $\pi_k = 1/K$ (therefore $P(k)$ is uniform).

Basis representations

The basis matrix $\mathbf{A}^k$ of each submodel defines its internal representation and each column vector of the matrix exposes the specific feature represented by each unit. Figure 2(A) shows the basis vectors of three example units in the face submodel. Each unit is visualized as a set of ellipses corresponding to the energy detectors, where their underlying Gabor filters have the indicated center positions (in the visual field coordinates), orientations, and spatial frequencies (inversely proportional to the size of the ellipse). The color of the ellipse indicates the weight value normalized by the maximal weight value. For readability, we show only the ellipses corresponding to the maximal positive (excitatory) weight and the minimal negative (inhibitory) weight at each location. Although this visualization approach may seem a bit too radical, it did not lose much information: we confirmed by visual inspection that the local weight patterns for most units had only one positive peak and one negative peak at each position and frequency and the patterns of orientation integration did not have notable changes across frequencies. In Figure 2, we can see that unit #1 represented a face outline either on the left (excitatory) or on the right (inhibitory); unit #2 represented mainly eyes (excitatory); unit #3 mainly represented a mouth (excitatory) and nose (weakly inhibitory). Figure 3 shows the basis vectors of 32 randomly selected units from (A) the face submodel and (B) the object submodel. The representations in these two submodels were qualitatively different: face units represented local facial features (i.e., facial parts like outline, eye, nose, and mouth) and object units represented local object features.

Selectivity to faces

As mentioned before, due to the Bayesian explaining-away effect in the mixture model, model face units exhibited selectivity to face images and object units to

Figure 2. The basis representations of three sample model face units. Each panel depicts the weighting pattern (basis vector) from a face unit to energy detectors by a set of ellipses, where each ellipse corresponds to the energy detector at the indicated x-y position, orientation, and frequency (inverse of the ellipse size); see the top right legend. The color shows the normalized weight value (color bar). Only the maximum positive and the minimum negative weights are shown at each position for readability.

Figure 3. The basis representations of (A) 32 example model face units and (B) 32 example model object units.

object images. We measured the responses of our model units to natural face and object images that were separate from the training images. The left panel of Figure 4(A) shows the responses (the feature variable $\hat{\mathbf{y}}^k$ in step 3 of the inference algorithm) of the face units (top) and object units (bottom) to face images, where the images were sorted by the response magnitudes, separately for each unit. The right panel similarly shows the responses of the same units to object images. We can see that the face units were prominently responsive to many face images while indifferent to non-face object images; the object units had the opposite property. Such vivid selectivities disappeared when the mixture computation was removed. Figure 4(B) shows the analogous responses of the face and object units immediately after performing sparse coding (the feature variable $\hat{\mathbf{y}}^k$ in step 1); the face units became almost equally responsive to object images to face images. To gain more insight into the underlying computations, see the distributions of face posterior probabilities (i.e., the approximate posterior probabilities of the face class; the value r_1 in step 2) for face and object images in Figure 4(C): faces and objects were clearly discriminated. Note that the response of each unit representing a part was modulated by the discrimination result of the whole image, which produced the face selectivity.

Figure 4. (A) The responses of model face units (1–400) and model object units (401–800) to face images (left) and object images (right). The images are sorted by response magnitudes (color bar) for each unit. (B) The responses in the case of removing mixture computation. (C) The distribution of face posteriors (the approximate posterior probabilities of the face class) for face image inputs and for object image inputs.

Figure 5. The tuning curves (red) of the model face units shown in Figure 2 to 19 feature parameters of cartoon faces. The mean (blue) as well as the maximum and minimum (green) of the tuning curves estimated from surrogate data are also shown (see the section on Simulation details in Methods).

Explaining face tuning properties

We next turn our attention to tuning properties to facial features. We particularly targeted the experiment conducted by Freiwald et al. [4] on the region in monkey IT cortex called the face middle patch. The experiment used cartoon face stimuli for which facial features were controlled by 19 feature parameters, each ranging from -5 to $+5$. The authors recorded responses of a neuron in the face middle patch while presenting a number of cartoon face stimuli whose feature parameters were randomly varied. Then, for each feature parameter, they estimated a tuning curve by taking the average of the responses to the stimuli that had a particular value while varying other parameters ("full variation"). We simulated the same experiment and analysis on our model (see the section on Simulation details in Methods).

To illustrate tuning to facial features in our model, Figure 5 shows the tuning curves of the face units in Figure 2 to all 19 feature parameters. Each unit was significantly tuned to one to nine feature parameters (where significance was defined in terms of surrogate data; see Methods). Some tunings clearly reflected the corresponding parts in the basis representations. Unit#1 was tuned only to the face direction, preferring the left as opposed to the right. Unit#2 mainly showed tuning to eye-related features, in particular, preferring narrower inter-eye distances and larger irises. Unit#3 mainly showed tuning to mouth- and nose-related features, in particular, preferring smily mouths and longer noses.

Even in the whole population, most units were significantly tuned to only a small number of features similarly to the experiment [4]. Figure 6(A) shows the distribution of the numbers of tuned features per unit, which were on average 3.6 and substantially smaller than 19, the total number of features. The face neurons in the monkey face middle patch were also tuned to only a small number of features, i.e., 2.6 on average [4, Figure 3c] (replotted in red boxes in Figure 6(A)). Figure 6(B) shows the distribution of the numbers of significantly tuned units per feature. The distribution strongly emphasizes geometrically large parts, i.e., face aspect ratio, face direction, feature assembly height, and inter-eye distance. The shape of the distribution has a good match with the experimental result [4, Figure 3d] (replotted in Figure 6(B)), though iris size seems much more represented in the monkey case.

A prominent property of the experimentally obtained tuning curves was preference or anti-preference of extreme facial features [4]; our model reproduced this property as well. For example, Figure 5 shows that many tuning curves

Figure 6. (A) The distribution of numbers of significantly tuned units for each feature parameter, overlaid with a replot (red boxes) of [4, Fig. 3d]. (B) The distribution of numbers of significantly tuned features per unit, overlaid with a replot of [4, Fig. 3c].

Figure 7. (A) All significant tuning curves of all model face units sorted by the peak parameter value. Each tuning curve (row) here was mean-subtracted and divided by the maximum. (B) The distributions of peak parameter values (top) and of trough parameter values (bottom). The extremity preference index of each distribution is shown above. The overlaid red boxes are replots of [4, Fig. 4a] averaged over three monkeys. (C) The distribution of minimal values of the significant tuning curves peaked at $+5$ and the flipped tuning curves peaked at -5, overlaid with a averaged replot of [4, Fig. 4d]. (D) The average of the tuning curves for each minimal value in (C) (with the same color).

were maximum or minimum at one of the extreme values (-5 or $+5$). For the entire population, Figure 7(A) shows all significant tuning curves of all face units, sorted by the peak feature values. To quantify this, Figure 7(B) shows the distributions of peak and trough feature values; the extremity preference index (the ratio of the average number of peaks in the extreme values to the number of peaks in the non-extreme values) was 9.1 and the extremity anti-preference index (analogously defined for troughs) was 12.0. These indicate that the tendency of preference or anti-preference of extreme features generally held for the population. This result is in good agreement with the monkey experiment [4], which also reported distributions of peak and trough values that were biased to the extreme values [4, Fig. 4a] (the extremity preference indices were 7.0, 5.5, and 7.1, and the extremity anti-preference indices were 12.6, 13.7, and 12.1 for three monkeys; the average distribution is replotted in Figure 7(B)).

In addition, the experimental study even observed monotonic tuning curves [4], which were also found in our model as in Figure 5. To quantify this for the population, Figure 7(C) shows the distribution of minimal values of the significant tuning curves preferring value $+5$ pooled together with the tuning curves preferring value -5 that have then been flipped; the distribution has a clear peak at value -5. Further, for each minimal value in Figure 7(C), the average of the tuning curves (normalized by the maximum response) with that minimal value is given in Figure 7(D); the averaged tuning curve for minimal value -5 has a monotonic shape. These indicate that tuning curves preferring one extreme value tended to anti-prefer the other extreme value and be monotonic. This result is consistent with the experimental data, which also showed a distribution of minimal values that was peaked at -5 [4, Fig. 4d]

Figure 8. (A) Full-variation versus single-variation tuning curves. (B) Full-variation versus partial face tuning curves. (C) Single-variation versus partial face tuning curves. (D) Single-variation versus partial face tuning curves in the case of removing mixture computation. (E) The distributions of face posteriors for the full variation, the single variation, and the partial face conditions.

(replotted in Figure 7(C)) and a monotonic averaged tuning curve corresponding to minimal value -5 [4, Fig. 4d, inset]. We discuss later why the model face units acquired such extremity preferences.

We have explained the face selectivity property as a form of holistic processing in the mixture model. The experimental study offered another, somewhat more direct example of holistic processing by using partial face stimuli [4]. In this, two kinds of tuning curves were estimated in addition to the one used so far ("full variation"), namely, the responses to full cartoon faces where one feature was varied and the other were fixed to standard ones ("single variation") and the responses to partial faces where only one feature was presented and varied ("partial face"). Again, we simulated the same experiments in our model (see the section on Simulation details in Methods). Figure 8 compares tuning curves in (A) full variation vs. single variation, (B) full variation vs. partial face, and (C) single variation vs. partial face. Overall, the shapes of the tunings were similar for all three kinds (average correlation 0.94 to 0.95). However, the gain of each tuning function (the slope of the fitted linear function) tended to drop after the removal of most of facial features (Figure 8C); the average gain ratio was 2.0, which was close to 2.2, the experimentally reported number [4, Fig. 6c]. This effect was because partial faces looked less face-like than full faces: Figure 8E shows lower face posteriors for the partial face condition than the full variation condition. Indeed, such drop was weakened when the mixture computation was removed: the average gain ratio was 1.5 when the same comparison was made for the responses of model face units without the mixture computation, i.e., using only step 1 in the inference algorithm (Figure 8D). In addition to these, note that the tunings curves in full variation were slightly reduced compared to those in single variation (Figure 8A and B); a similar tendency can be observed in the experimental result [4, Fig. 6c]. This reduction in the model was because the face images used in the single variation condition took standard feature values for most parameters and such face images looked more face-like than others (giving slightly larger face posteriors than the full variation condition; Figure 8E).

Interaction between feature parameters was limited, though present. For each pair of feature parameters, a 2D tuning was estimated by averaging the responses to a pair of parameter values while varying the remaining parameters. Then, the 2D tuning for a pair of parameters was compared to another 2D tuning predicted by the sum of two (full-variation) 1D tunings for the same parameters

Figure 9. The distributions of correlation coefficients between 2D tuning functions and additive (blue) or multiplicative predictors (yellow).

Figure 10. The distributions of (A) the number of tuned features per unit (cf. Figure 6A), (B) the number of tuned units per feature (cf. Figure 6B), and (C) the peak (top) and the trough (bottom) feature values (cf. Figure 7B), in different model variations. The color of each curve indicates the model variation (see legend).

or by the product of these. The distributions of correlation coefficients are given in Figure 9; the averages were both 0.90, which was similar to the experimental result (averages 0.88 and 0.89) [4, Figure 5b].

Control simulations

How much do our results depend on the exact form of model? To address this question, we modified the original model in various ways and conducted the same analysis.

First, we already showed that, when we omitted the mixture computation and simply used a sparse coding model of face images, the model units were deprived of selectivities to faces vs. objects (Figure 4). However, tuning properties to facial features did not change much. Figure 10 shows that the distributions of the number of tuned features per unit, of the number of tuned units per feature, of the peak feature values, and of the trough feature values for the modified model (cyan curves) are all similar to the original model (blue curves). Therefore, while the selectivities were from the mixture model, the tuning properties were produced by the sparse coding.

Next, we varied the strength of dimension reduction of the outputs of the energy detector bank before performing sparse coding learning (the original model reduced the dimensionality from 2400 to 100). Three observations were made. First, consistently with our previous observation in our V2 model [17, 18], overall feature sizes tended to decrease while the reduced dimensionality was increased. Figure 12 shows example face and object units in the case of 300 reduced dimensions; compare these with Figure 3. (When we further increased the reduced dimensionality, we obtained quite a few units with globally shaped, somewhat noisy basis representations. These seemed to be a kind of "junk units" that are commonly produced when the amount of data is insufficient compared to the input dimensionality.) Second, as the reduced dimensionality increased, face posteriors (as in Figure 4C) were substantially decreased for face images (Figure 11); the face images could barely be discriminated in the case of 300

Figure 11. The distribution of face posteriors for face images (solid curve) or for object images (broken curve) in different model variations (cf. Figure 4C). The color of each curve indicates the model variation (see legend).

Figure 12. The basis representations of 32 example model units from (A) the face submodel and (B) the object submodel, in the network trained with 300 reduced dimensions.

reduced dimensions. Meanwhile, face posteriors remained low for object images. This seemed to happen because the object submodel now learned to represent spatially very small and generic features so that it could give sufficiently good interpretations not only to object images but also to face images. This justified our model construction approach that performs strong dimension reduction before sparse coding learning. Third, Figure 10A–B shows that the number of tuned features per unit and the number of tuned units per feature decreased in the case of 300 reduced dimensions (red curve). This was due to the weakened selectivity rather than the size decrease of feature representations since the effect disappeared when the mixture computation was omitted (yellow curve).

Finally, we varied the number of units in each submodel, but this hardly made any difference in the results.

Discussion

In this study, we proposed a novel theory called mixture of sparse coding models for investigating the computational principles underlying face and object processing in the IT cortex. In this model, several submodels are employed where each submodel has its own sparse feature representation. For a given input, while each submodel attempts to interpret the input by its code set, the best interpretation explains away the input, dismissing the explanation offered by the other submodel. As a concrete network model, we built a mixture of two sparse coding submodels on top of an energy detector bank and separately trained each submodel by face images or non-face object images (Figure 1). We used probabilistic (Bayesian) inference of hidden variables to model evoked neuronal responses. The model face units in the resulting network not only exhibited significant selectivity to face images (Figure 4), but also explained qualitatively and quantitatively tuning properties of face neurons to facial features (Figures 5 to 9) as reported for the face middle patch, a particular subregion in the macaque IT cortex [4]. Thus, computation in this cortical region might be somehow related to mixture of sparse coding models.

While sparse coding produced parts-based representations in each submodel (Figures 2 and 3), the mixture model produced an explaining-away effect that led to holistic processing. This combination was key to simultaneous explanation of two important neural properties: tuning to a small number of facial features and face selectivity. That is, although the former property could be explained by sparse coding alone (Figure 10), the latter could not (Figure 4) presumably since facial parts could accidentally be similar to object parts. However, when the sparse coding submodels for faces and objects were combined in the mixture model, the individual face units could be activated only if the whole input was interpreted as a face. A similar explanation holds for the gain reduction observed when the presented faces were partial (Figure 8).

Among the reported properties of face neurons in the monkey IT cortex, preferences to extreme features (in particular, monotonic tuning curves) were considered as a surprising property [4] since they were rather different from more typical bell-like shapes such as orientation and frequency tunings. We showed that our model explained quite well such extremity preferences (Figure 7). It is intriguing why our model face units had such property. First, we would like to point out that the facial features discussed here are mostly related to positions of facial parts and such features can be relatively easily encoded by a linear function of an image. This is not the case, however, for orientations and frequencies since encoding these seem to require a much more complicated nonlinear function, perhaps naturally leading to units with bell-like tunings. Second, we could speculate that the extremity preferences may be really necessary due to the statistical structure of natural face images, irrelevant to any particular details of our model. Indeed, even when we perform a very basic statistical analysis of principal components of face images (so-called eigenfaces, e.g., [20]), they look like linear representations of certain facial features, maximal in one extreme and minimal in the other extreme. However, this seems to be a rather deep question and fully answering it is beyond the scope of this study.

The results shown here relied on all computational components in mixture of sparse coding models, including computation of posterior probabilities in each sparse coding submodel and suppressive operations based on the computed posteriors. Since these computations seem to be difficult to implement only with simple feedforward processing, it would be interesting whether or not similar results could be reproduced by a deep (feedforward) neural network model [21–24]. Note that, although face-selective units have been discovered in some models [23, 24], no tuning properties to facial features like here have been reported yet (except for tuning to head orientation [24]). We particularly wonder whether the face-selective units in such models represent facial parts, which can be impossible to recognize correctly without any surrounding context in the case where the input image does not contain enough detail of the facial part, e.g., Figure 1(A). While it is mathematically true that such nonlinear context-dependent computation could also be arbitrarily well approximated by a feedforward model, whether this can be achieved by a network optimized for image classification needs some experiment. In any case, however, we think that top-down feedback processing as formulated in our model would be a simpler

and biologically more natural way of performing such computation.

Since we trained each submodel of our mixture model separately by face or object images, our learning algorithm was supervised, implicitly using class labels ("face" or "object"). This choice was primarily for simplification to avoid the generally complicated problem of unsupervised learning of a mixture model. So we do not claim by any means that face and object representations in the IT cortex should be learned exactly in this way. Nonetheless, the existence of such teaching signals may not be a totally unreasonable assumption in the actual neural system. In particular, since faces can be detected by a rather simple operation [25, 26], some kind of innate mechanism would easily be imaginable. This may also be related to the well-known fact that infant monkeys and humans can recognize faces immediately after eye opening [27, 28].

Sparse coding was originally developed to explain receptive field properties of V1 simple cells in terms of local statistics of natural images [7, 29], following Barlow's efficient coding hypothesis [30]. The theory was subsequently extended to explain other properties of V1 complex cells [31–33] and V2 cells [17, 34, 35], though few studies pursued a similar approach to investigate higher visual areas prior to our study here. On the other hand, sparse-coding-like models have also been used in computer vision for feature representation learning, including the classical study of ICA of face images [20]. Since global facial features were reported as the resulting basis set [20], it was once argued that parts-based representations require the non-negativity constraint [11]. However, it seems that such completely global ICA features may have been due to some kind overlearning and, indeed, local feature representations were obtained when we used enough data as above (Figure 3; we also confirmed the case with raw images). Another relevant formalism is mixture of ICA models [36]. Although the idea is somewhat similar, their full rank assumption on the basis matrix and the lack of Gaussian noise terms make it inappropriate in our case because the strong dimension reduction was essential for ensuring the image discriminability (Figure 11).

Our model presented here is not meant to explain all properties of face neurons. Indeed, the properties explained here are a part of known properties of face neurons in the middle patch, which is in turn a part of the face network in the monkey IT cortex [8, 37, 38]. In the middle patch, face neurons are also tuned to contrast polarities between facial parts [39]. In more anterior patches, face neurons are tuned to viewing angles in a mirror-symmetric manner or invariant to viewing angles but selective to identities [40]. Further, all these neurons are invariant to shift and size transformation as usual for IT neurons [40]. Explaining any of these properties seems to require a substantial extension of our current model and is thus left for future research. Finally, since most detailed and reliable experimental data on the IT cortex concerns face processing, we hope that the principles, such as presented here, found in face processing could serve to elucidate principles of general visual object processing.

Methods

Approximating posterior

Given an input x, we intend to compute the posterior expectations of each $\mathbf{y}^h$:

$$\mathcal{E}\left[\mathbf{y}_k \mid \mathbf{x}\right] = \sum_k \int \int \cdots \int \mathbf{y}^k P(\mathbf{y}^1, \mathbf{y}^2, \ldots, \mathbf{y}^K, k \mid \mathbf{x}) d\mathbf{y}^1 d\mathbf{y}^2 \cdots d\mathbf{y}^K \quad (12)$$

Direct computation of this value is not easy. Note, however, that, from the definition of the model (equation 5), the posterior distribution has a single strong peak for each class k, with variances more or less similar across all classes. Therefore we approximate the posterior probability by

$$P(\mathbf{y}^1, \mathbf{y}^2, \ldots, \mathbf{y}^K, k \mid \mathbf{x}) \approx \delta(\mathbf{y}^1 = \hat{\mathbf{y}}^1(k), \mathbf{y}^2 = \hat{\mathbf{y}}^2(k), \ldots, \mathbf{y}^K = \hat{\mathbf{y}}^K(k)) r_k \quad (13)$$

where $\hat{\mathbf{y}}^h(k)$ is the MAP estimate of y^h when the selected image class is k (equation 6) and r_k is the relative peak posterior probability for the class k (equation 7). Here, $\delta(\cdot)$ is the delta function that takes infinity for the specified input value and zero for other values. Substituting the approximation (13) into equation (12) yields equation (8).

Data preprocessing

As a face image dataset, we used a version of Labeled Faces in Wild (LFW) [15] where face alignment was already performed using an algorithm called "deep funneling" [16]. By this alignment, faces had a more or less similar position, size, and (upright) posture across images. The dataset consisted of about 13,000 images in total. Each image was converted to gray scale, cropped to the central square region containing only the facial parts and hairs, and resized to 64×64 pixels. Since many images still contained some background, they were further passed to a disk-like filter, which retained the image region within 30 pixels from the center and gradually faded the region away from this circular area. Finally, the pixel values were standardized to zero mean and unit variance per image.

As an object image dataset, we used Caltech101 [12]. We removed four image categories containing human and animal face images (Faces, Faces_easy, Cougar_face, and Dalmetian). The objects within the images were already aligned. The dataset consisted of about 8,000 images in total. Like face images, each image was converted to gray scale, cropped to square, resized to 64×64 pixels, passed to the above mentioned disk-like filter, and standardized per image.

For each class, we reserved a thousand images for selectivity test and used the rest for model training.

Model details

Our hierarchical model began with a bank of Gabor filters. The filters had all combinations of 10×10 center locations (arranged in a square grid within

64×64 pixels), 8 orientations (at 22.5° interval), 3 frequencies (0.25, 0.17, and 0.13 cycles/picels), and 2 phases (0° and 90°). The Euclidean norm of each Gabor filter with frequency f was set to $f^{1.15}$ (following $1/f$ spectrum of natural images) and the Gaussian width and length were both set to $0.4/f$.

Simulation details

Cartoon face images were created by using the method described by Freiwald et al. [4]. Each face image was drawn as a linear combination of 7 facial parts (outline, hair, eye pair, iris pair, eyebrows, nose, and mouth). The facial parts were controlled by 19 feature parameters: (1) face aspect ratio (round to long), (2) face direction (left to right), (3) feature assembly height (up to down), (4) hair length (short to long), (5) hair thickness (thin to thick), (6) eyebrow slant (angry to worried), (7) eyebrow width (short to long), (8) eyebrow height (up to down), (9) inter-eye distance (narrow to wide), (10) eye eccentricity (long to round), (11) eye size (small to large), (12) iris size (small to large), (13) gaze direction (11 x-y positions), (14) nose base (narrow to wide), (15) nose altitude (short to long), (16) mouth-nose distance (short to long), (17) mouth size (narrow to wide), (18) mouth top (smily to frowny), and (19) mouth bottom (closed to open). Note that the first three parameters globally affected the actual geometry of all the facial parts, while the rest locally determined only the relevant facial part.

Following the method in the same study [4], we estimated three kinds of tuning curves: (1) full variation, (2) single variation, and (3) partial face. For full variation, a set of 5000 cartoon face images were generated while the 19 parameters were randomly varied. For each unit and each feature parameter, a tuning curve at each feature value was estimated as the average of the unit responses to the cartoon face images for which the feature parameter took that value. The tuning curve was then smoothed by a Gaussian kernel with unit variance. To determine the significance of each tuning curve, 5000 surrogate tuning curves were generated by destroying the correspondences between the stimuli and the responses. Then, a tuning curve was regarded significant if (1) its maximum was at least 25% greater than its minimum and (2) its heterogeneity exceeded 99.9% of those of the surrogates, where the heterogeneity of a tuning curve was defined as the negative entropy when the values in the curve were taken as relative probabilities.

For single variation, a tuning curve for a feature parameter at each value was estimated as the response to a cartoon face image for which the feature parameter took that value and the other were fixed to standard values. The standard parameter values were obtained by a manual adjustment with the stimuli used in the experiment [4, Suppl. Fig. 1]. For partial face, cartoon face images with only one facial part (hair, outline, eyebrows, eyes, nose, mouth, or irises) were created. Each tuning curve for each feature parameter was obtained similarly to single variation, except that only the relevant facial part was present in the stimulus.

References

1. Tanaka JW, Farah MJ. Parts and wholes in face recognition. The Quarterly journal of experimental psychology. 1993;46A(2):225–245.

2. McKone E, Kanwisher N, Duchaine BC. Can generic expertise explain special processing for faces? Trends in cognitive sciences. 2007 Jan;11(1):8–15.

3. Tsunoda K, Yamane Y, Nishizaki M, Tanifuji M. Complex objects are represented in macaque inferotemporal cortex by the combination of feature columns. Nature Neuroscience. 2001 Aug;4(8):832–838.

4. Freiwald WA, Tsao DY, Livingstone MS. A face feature space in the macaque temporal lobe. Nature Neuroscience. 2009 Aug;12(9):1187–1196.

5. Schiltz C, Rossion B. Faces are represented holistically in the human occipito-temporal cortex. NeuroImage. 2006 Sep;32(3):1385–1394.

6. Bishop CM. Pattern recognition and machine learning (information science and statistics). Springer; 2006.

7. Olshausen BA, Field DJ. Emergence of simple-cell receptive field properties by learning a sparse code for natural images. Nature. 1996;381(6583):607–609.

8. Tsao DY, Freiwald WA, Tootell R, Livingstone MS. A cortical region consisting entirely of face-selective cells. Science. 2006;311(5761):670–674.

9. Kanwisher N, McDermott J, Chun MM. The fusiform face area: a module in human extrastriate cortex specialized for face perception. The Journal of Neuroscience. 1997 Jun;17(11):4302–4311.

10. Popivanov ID, Schyns PG, Vogels R. Stimulus features coded by single neurons of a macaque body category selective patch. Proceedings of the National Academy of Sciences. 2016 Apr;113(17):E2450–E2459.

11. Lee DD, Seung HS. Learning the parts of objects by non-negative matrix factorization. Nature. 1999 Oct;401(6755):788–791.

12. Fei-Fei L, Fergus R, Perona P. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. Computer Vision and Image Understanding. 2007 Apr;106(1):59–70.

13. Olshausen BA, Field DJ. Sparse coding with an overcomplete basis set: a strategy employed by V1? Vision Research. 1997 Dec;37(23):3311–3325.

14. Desimone R, Schein SJ. Visual properties of neurons in area V4 of the macaque: sensitivity to stimulus form. Journal of Neurophysiology. 1987;57(3):835–868.

15. Huang GB, Ramesh M, Berg T, Learned-Miller E. Labeled faces in the wild: A database for studying face recognition in unconstrained environments. University of Massachusetts, Amherst; 2007. 07-49.

16. Huang G, Mattar M, Lee H. Learning to align from scratch. Advances in neural information processing systems. 2012;p. 764–772.

17. Hosoya H, Hyvärinen A. A Hierarchical Statistical Model of Natural Images Explains Tuning Properties in V2. Journal of Neuroscience. 2015 Jul;35(29):10412–10428.

18. Hosoya H, Hyvärinen A. Learning Visual Spatial Pooling by Strong PCA Dimension Reduction. Neural Computation. 2016;28:1249–1263.

19. Hyvärinen A. Estimation of Non-Normalized Statistical Models by Score Matching. The Journal of Machine Learning Research. 2005 Apr;6:695–709.

20. Bartlett MS, Movellan JR, Sejnowski TJ. Face recognition by independent component analysis. IEEE Transactions on Neural Networks. 2002 Nov;13(6):1450–1464.

21. Serre T, Oliva A, Poggio T. A feedforward architecture accounts for rapid categorization. Proceedings of the National Academy of Sciences. 2007 Apr;104(15):6424–6429.

22. Krizhevsky A, Sutskever I, Hinton GE. ImageNet classification with deep convolutional neural networks. In: Advances in neural information processing systems; 2012. p. 1097–1105.

23. Yamins DLK, Hong H, Cadieu CF, Solomon EA, Seibert D, DiCarlo JJ. Performance-optimized hierarchical models predict neural responses in higher visual cortex. Proceedings of the National Academy of Sciences. 2014 Jun;111(23):8619–8624.

24. Farzmahdi A, Rajaei K, Ghodrati M, Ebrahimpour R, Khaligh-Razavi SM. A specialized face-processing model inspired by the organization of monkey face patches explains several face-specific phenomena observed in humans. Scientific Reports. 2016;6:25025.

25. Sinha P. Qualitative representations for recognition. Workshop on Biologically Motivated Computer Vision (Lecture Notes in Computer Science). 2002;2525:249–262.

26. Viola P, Jones MJ. Robust real-time face detection. International Journal of Computer Vision. 2004;57(2):137–154.

27. Sugita Y. Face perception in monkeys reared with no exposure to faces. Proceedings of the National Academy of Sciences. 2008 Jan;105(1):394–398.

28. Morton J, Johnson MH. CONSPEC and CONLERN: a two-process theory of infant face recognition. Psychological review. 1991;98(2):164–181.

29. van Hateren JH, van der Schaaf A. Independent component filters of natural images compared with simple cells in primary visual cortex. Proceedings of the Royal Society B: Biological Sciences. 1998 Mar;265(1394):359–366.

30. Barlow HB. Possible principles underlying the transformation of sensory messages. Sensory communication. 1961;p. 217–234.

31. Hyvärinen A, Hoyer P. Emergence of phase-and shift-invariant features by decomposition of natural images into independent feature subspaces. Neural Computation. 2000;12(7):1705–1720.

32. Karklin Y, Lewicki MS. Emergence of complex cell properties by learning to generalize in natural scenes. Nature. 2009 Jan;457(7225):83–86.

33. Schwartz O, Simoncelli EP. Natural signal statistics and sensory gain control. Nature Neuroscience. 2001;4(8):819–825.

34. Hosoya H. Multinomial Bayesian learning for modeling classical and nonclassical receptive field properties. Neural Computation. 2012 Aug;24(8):2119–2150.

35. Gutmann MU, Hyvärinen A. A three-layer model of natural image statistics. Journal of Physiology-Paris. 2013;107(5):369–398.

36. Lee TW, Lewicki MS. ICA mixture models for unsupervised classification of non-Gaussian classes and automatic context switching in blind signal separation. IEEE Transactions on Pattern Analysis and Machine Intelligence. 2000;22(10):1078–1–90.

37. Moeller S, Freiwald WA, Tsao DY. Patches with Links: A Unified System for Processing Faces in the Macaque Temporal Lobe. Science. 2008 Jun;320(5881):1355–1359.

38. Grimaldi P, Saleem KS, Tsao D. Anatomical Connections of the Functionally Defined "Face Patches" in the Macaque Monkey. Neuron. 2016 Jun;90(6):1325–1342.

39. Ohayon S, Freiwald WA, Tsao DY. What Makes a Cell Face Selective? The Importance of Contrast. Neuron. 2012 May;74(3):567–581.

40. Freiwald WA, Tsao DY. Functional compartmentalization and viewpoint generalization within the macaque face-processing system. Science. 2010 Nov;330(6005):845–851.

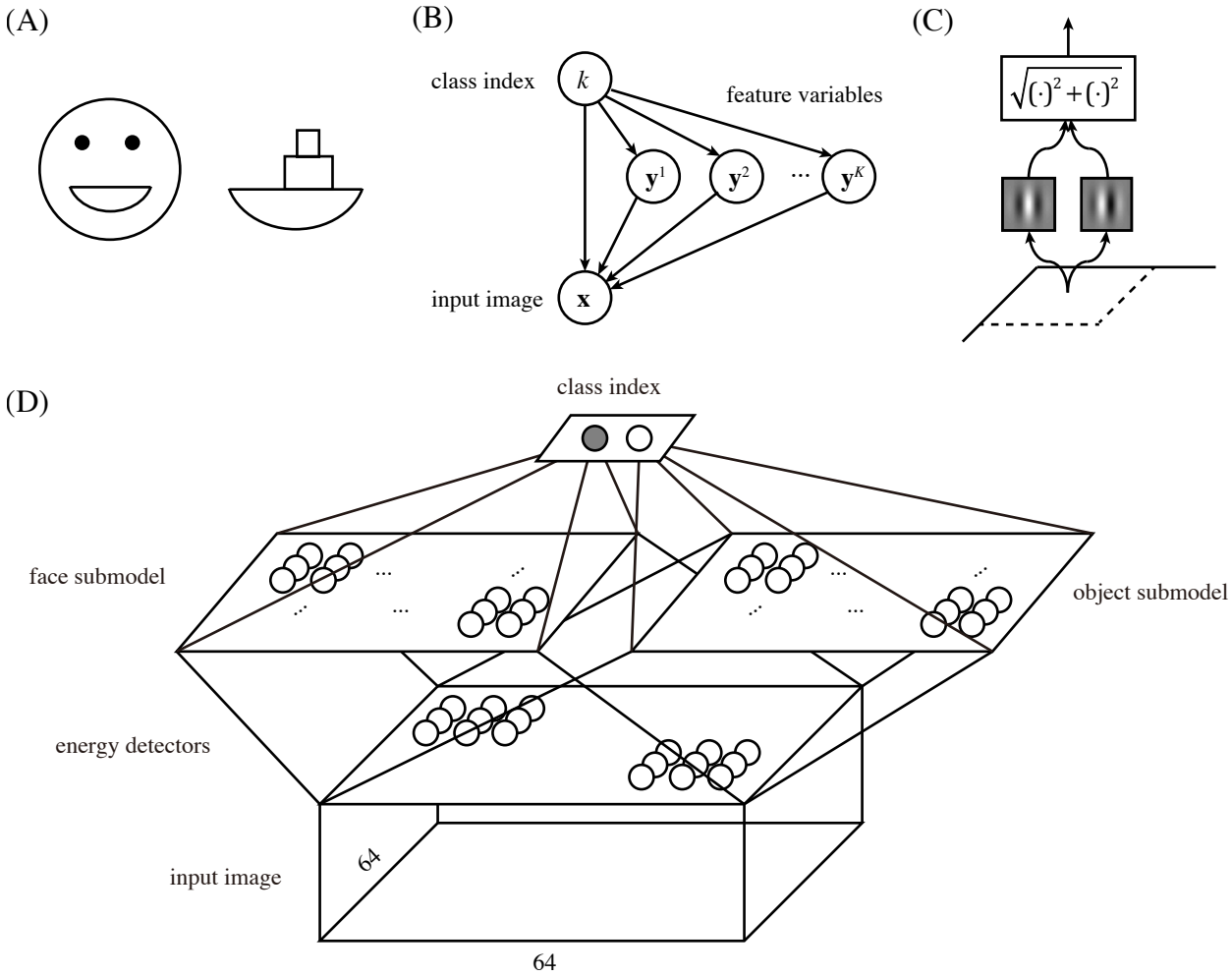

Figure 1

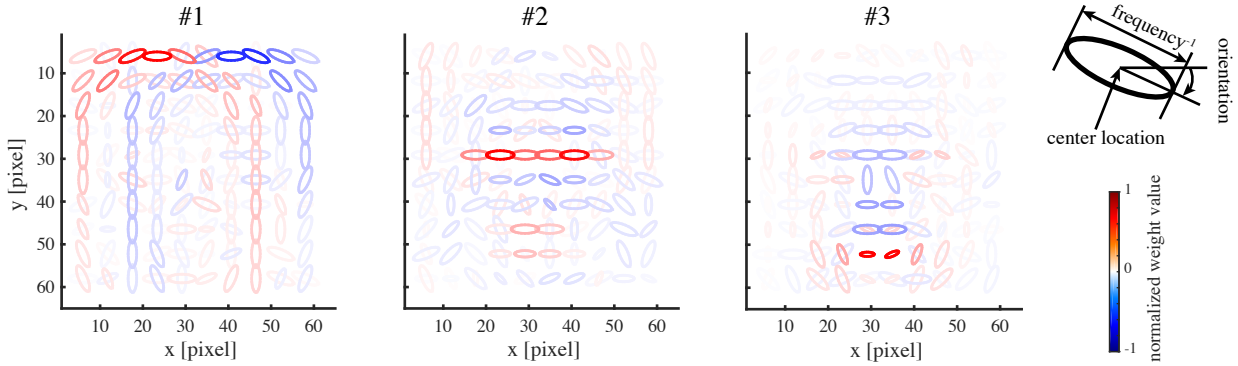

Figure 2

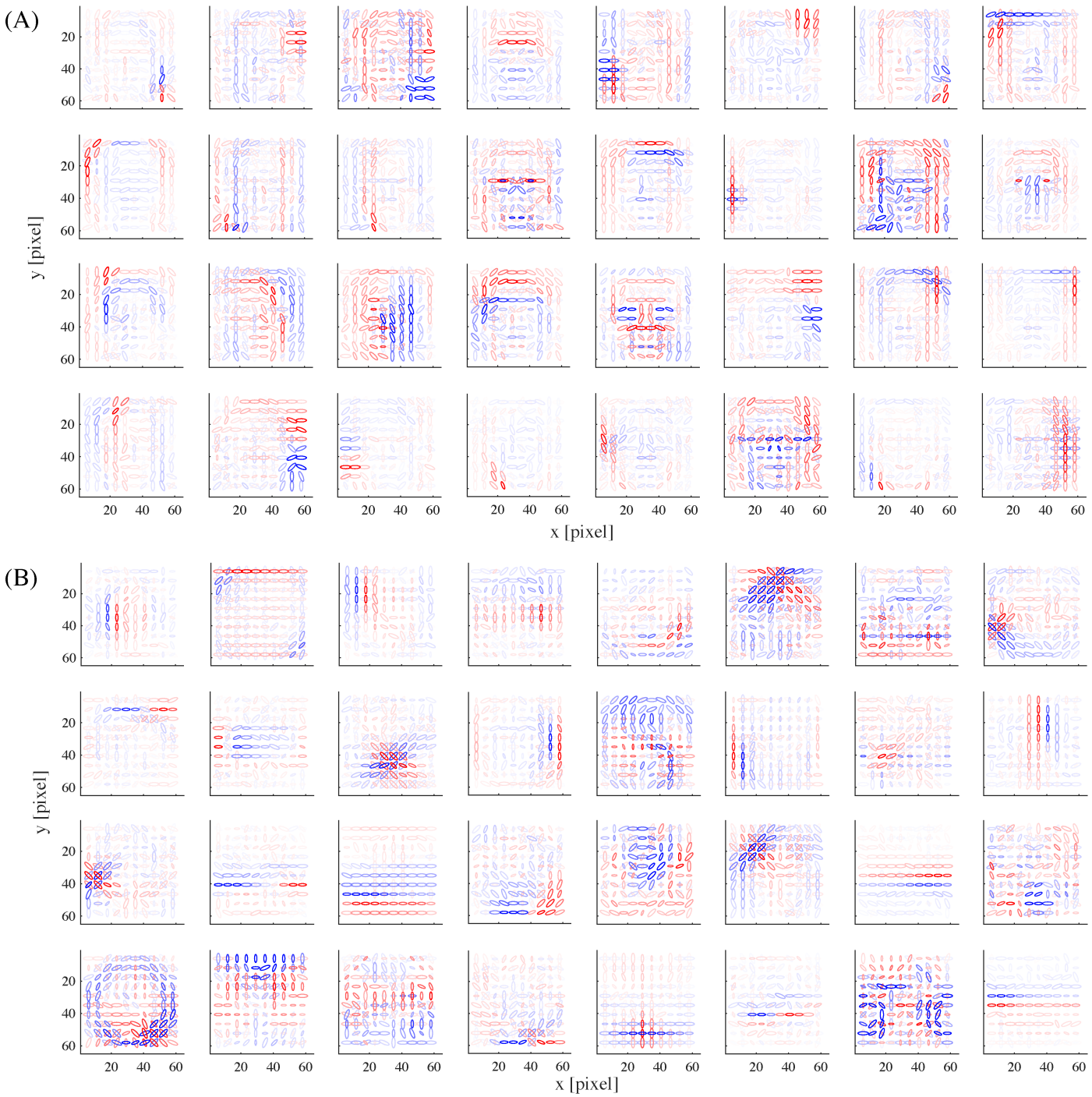

Figure 3

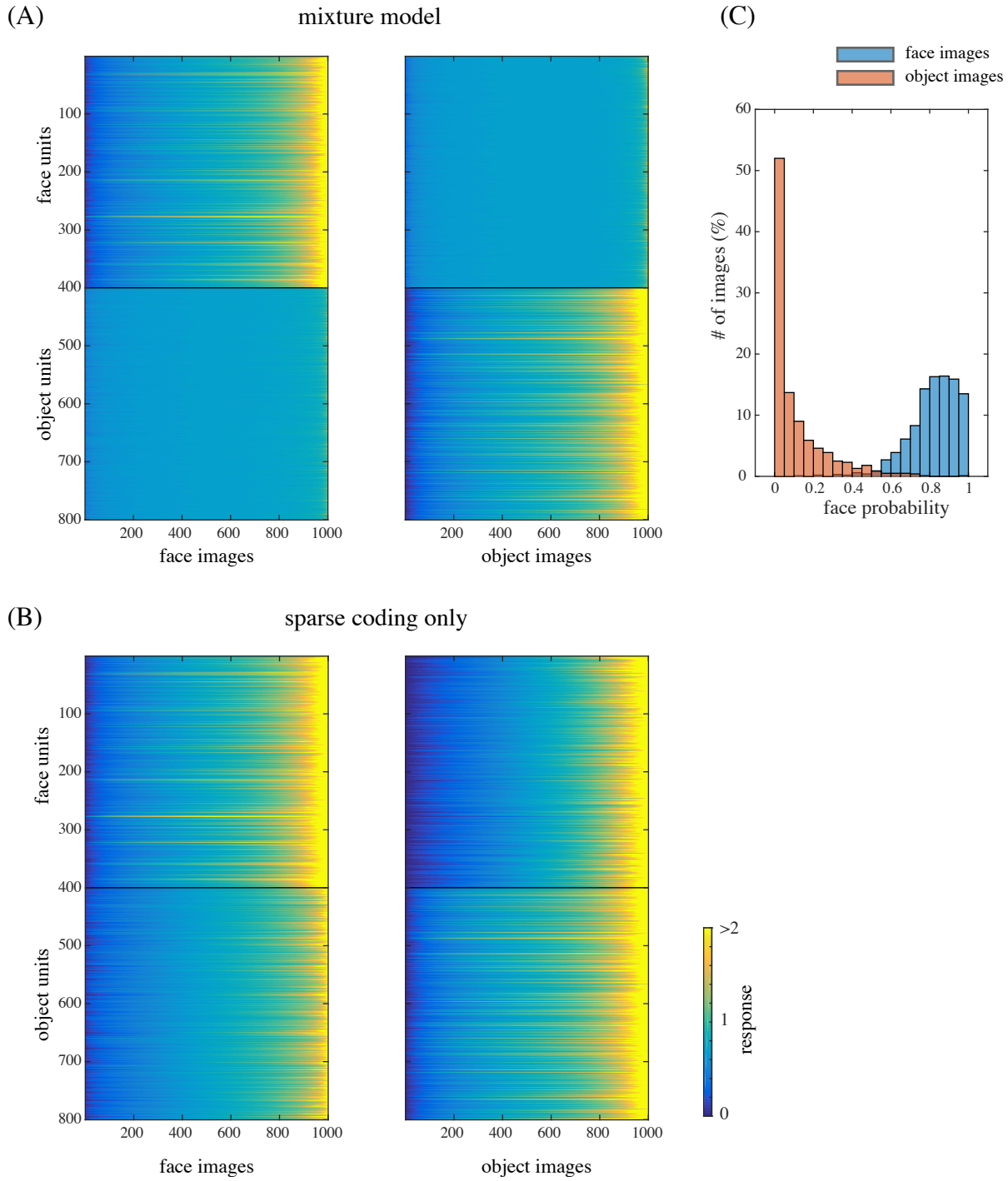
(A)
mixture model
face units
object units
face images
object images
(B)
sparse coding only
face units
object units
face images
object images
>2
1
0
response
(C)
face images
object images
# of images (%)
face probability

Figure 4

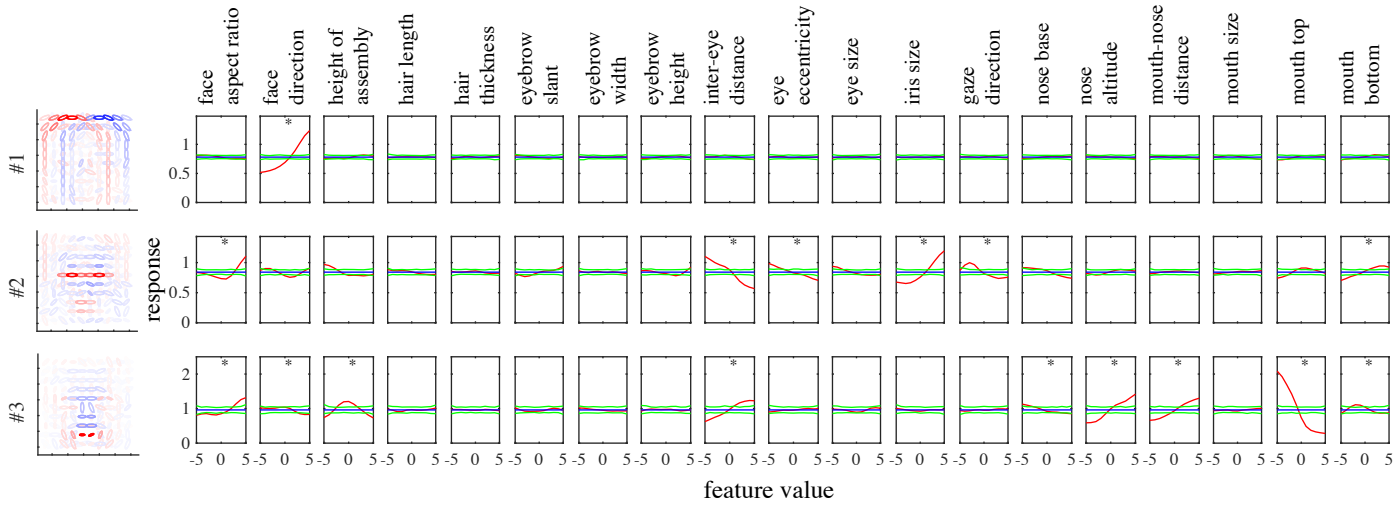

Figure 5

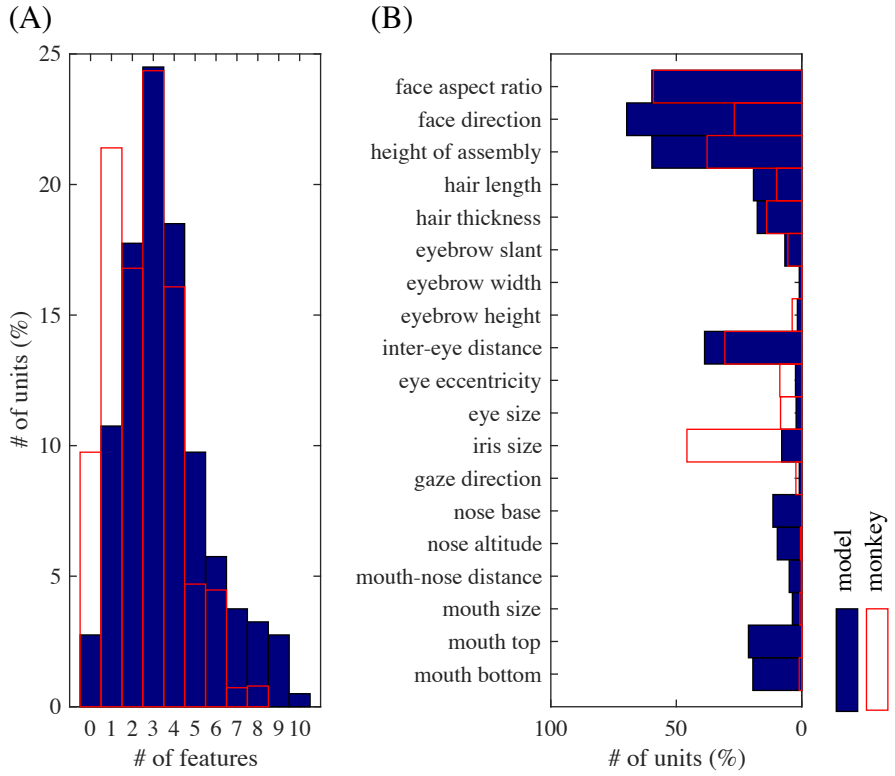

Figure 6

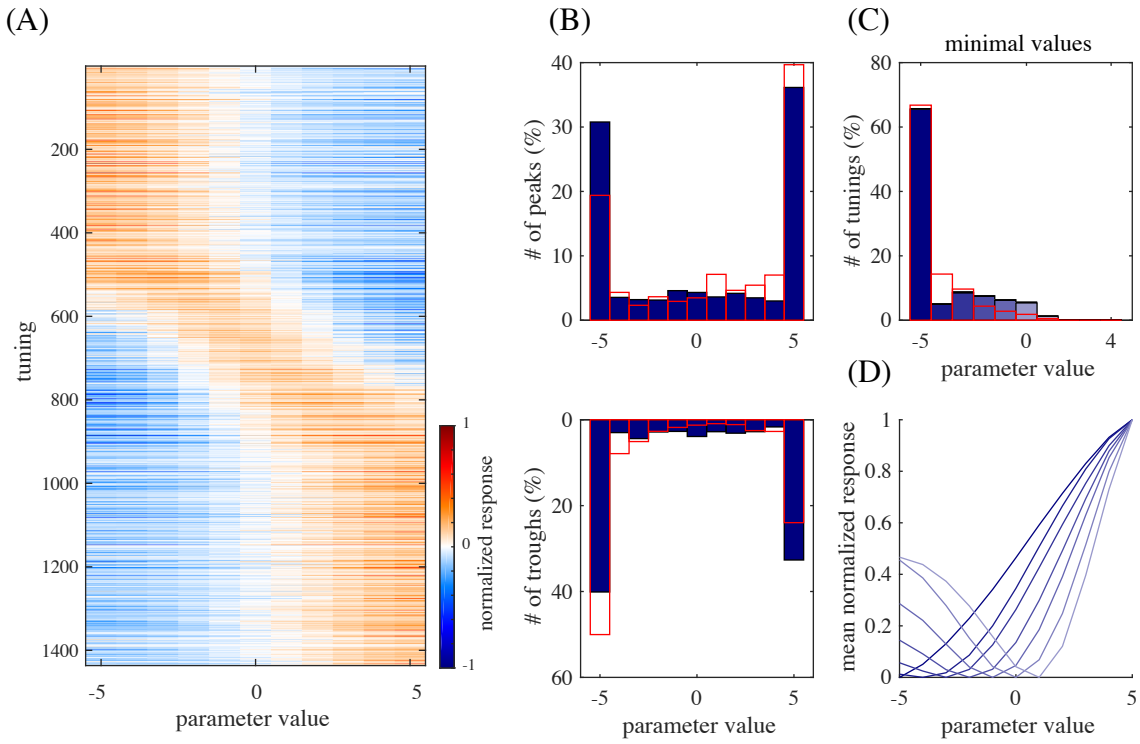
(A)
tuning
200
400
600
800
1000
1200
1400
-5
0
5
parameter value
1
0
-1
normalized response
(B)
# of peaks (%)
40
30
20
10
0
-5
0
5
# of troughs (%)
0
20
40
60
-5
0
5
parameter value
(C)
minimal values
# of tunings (%)
80
60
40
20
0
-5
0
4
parameter value
(D)
mean normalized response
1
0.8
0.6
0.4
0.2
0
-5
0
5
parameter value

Figure 7

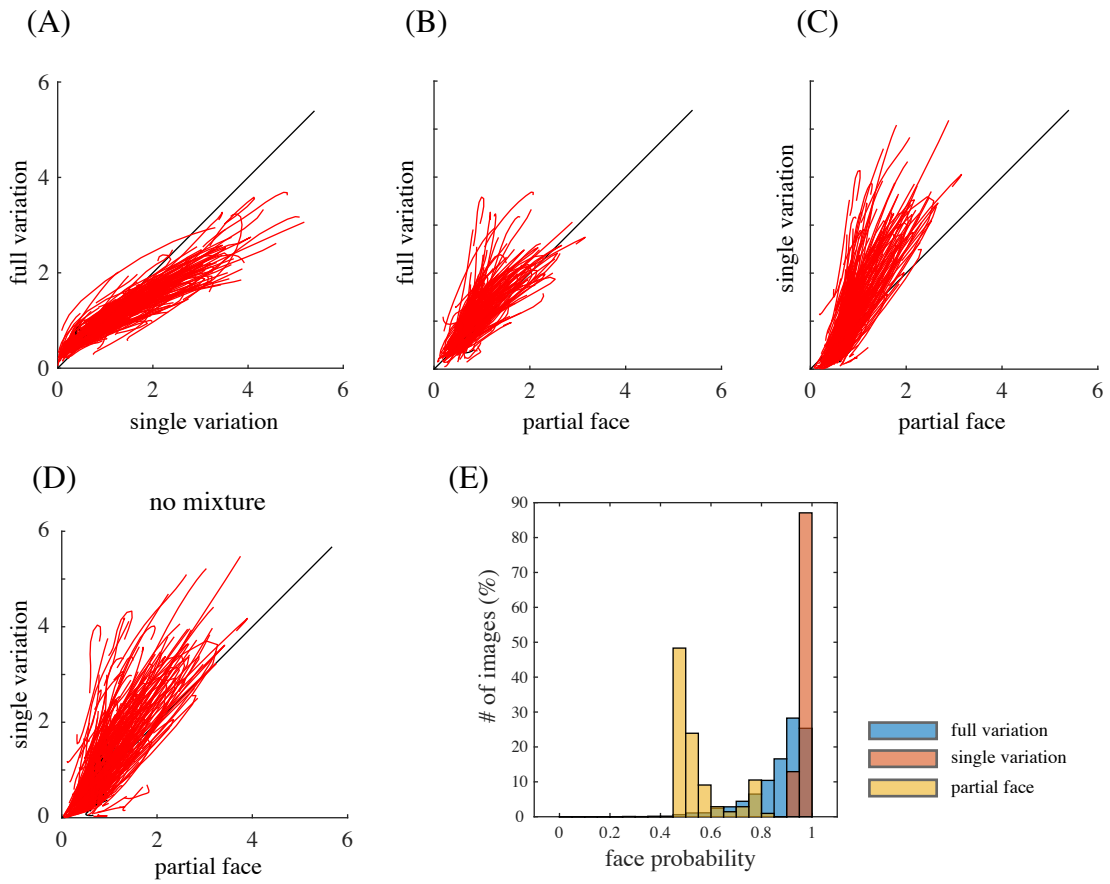

Figure 8

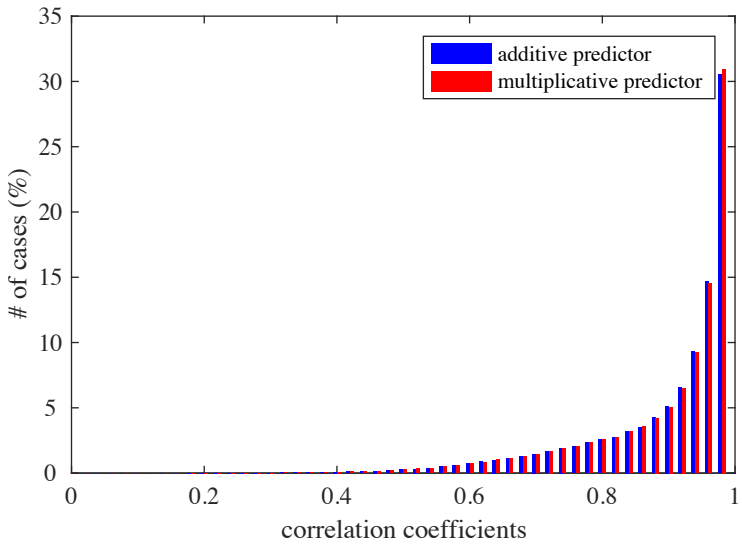

Figure 9

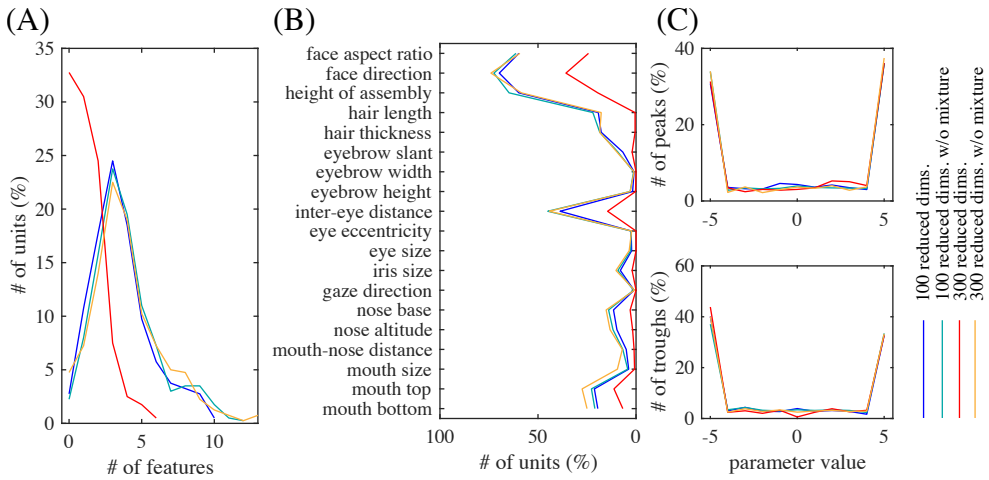

Figure 10

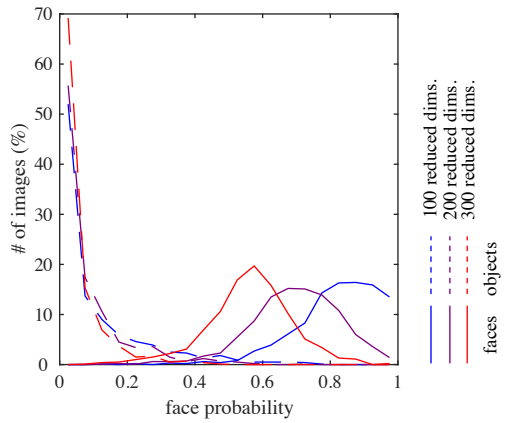

Figure 11

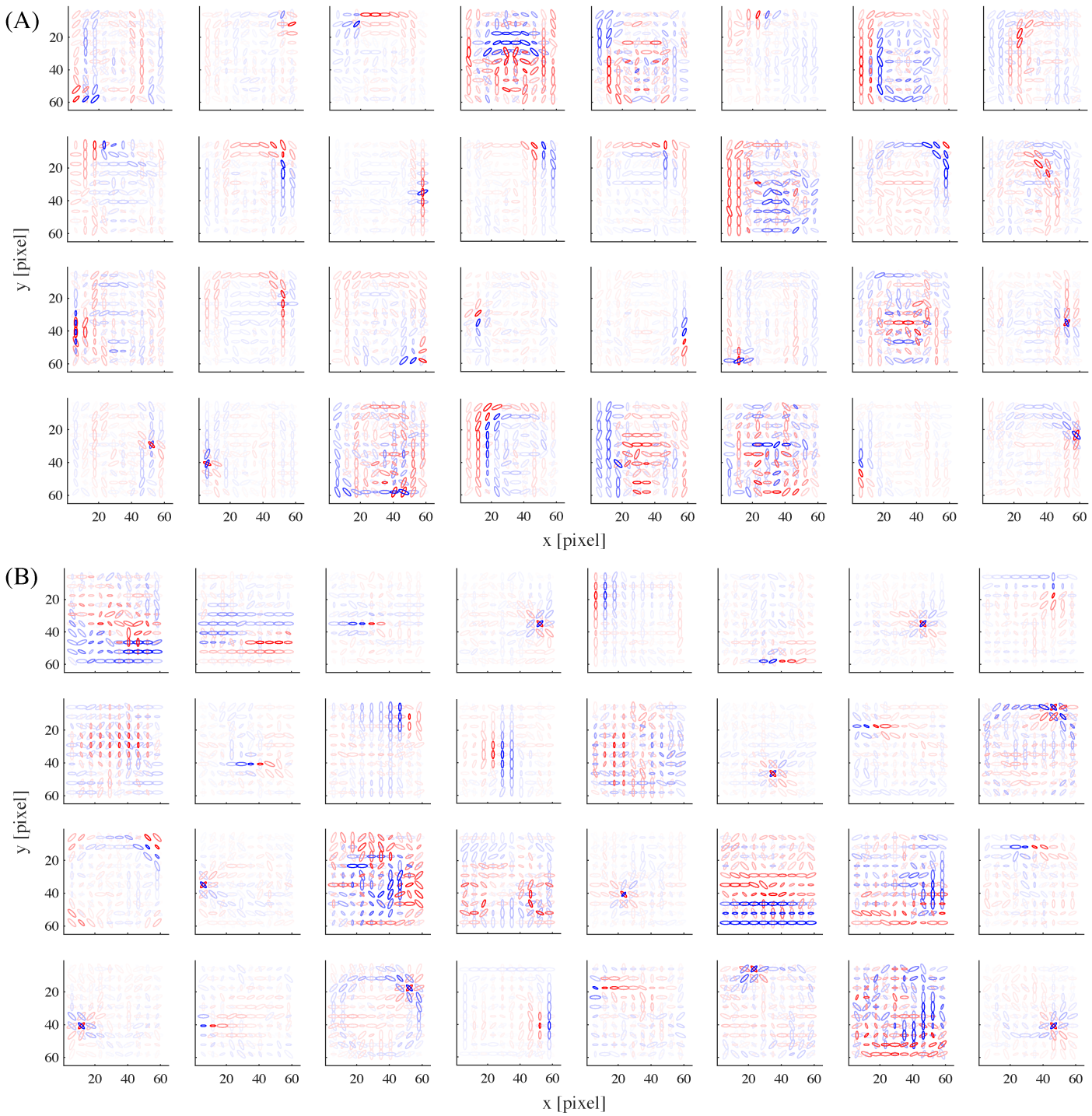

Figure 12