In Press, Neural Computation

Face Representations via Tensorfaces of Various Complexities

Sidney R. Lehky[1,6], Anh Huy Phan[2,3], Andrzej Cichocki[2,3,4,5], Keiji Tanaka[1]

[1]Cognitive Brain Mapping Laboratory
RIKEN Center for Brain Science
Wako-shi, Saitama, Japan

[2]Advanced Brain Signal Processing Laboratory
RIKEN Brain Science Institute
Wako-shi, Saitama, Japan

[3]Center for Computational and Data-Intensive Science and Engineering
Skolkovo Institute of Science and Technology
Moscow, Moskovskaya, Russia

[4]Systems Research Institute
Polish Academy of Sciences
Warsaw, Mazowieckie, Poland

[5]Nicolaus Copernicus University
Toruń, Kujawsko-pomorskie, Poland

[6]Computational Neurobiology Laboratory
The Salk Institute
La Jolla, California, USA

Running title: Face Representations of Various Complexities

Corresponding author:

Sidney R. Lehky
Cognitive Brain Mapping Laboratory
RIKEN Center for Brain Science
Hirosawa 2-1
Wako-shi, Saitama 351-0198
Japan

sidney.lehky@riken.jp

Abstract

Neurons selective for faces exist in humans and monkeys. However, characteristics of face cell receptive fields are poorly understood. In this theoretical study we explore the effects of complexity, defined as algorithmic information (Kolmogorov complexity) and logical depth, on possible ways that face cells may be organized. We use tensor decompositions to decompose faces into a set of components, called tensorfaces, and their associated weights, which can be interpreted as model face cells and their firing rates. These tensorfaces form a high-dimensional representation space in which each tensorface forms an axis of the space. A distinctive feature of the decomposition algorithm is the ability to specify tensorface complexity. We found low-complexity tensorfaces have blob-like appearances crudely approximating faces while high-complexity tensorfaces appear clearly face-like. Low-complexity tensorfaces require a larger population to reach a criterion face reconstruction error than medium- or high-complexity tensorfaces, and thus are inefficient by that criterion. On the other hand, low-complexity tensorfaces generalize better when representing statistically novel faces, which are faces falling beyond the distribution of face description parameters found in the tensorface training set. The degree to which face representations are parts-based or global forms a continuum as a function of tensorface complexity, with low- and medium tensorfaces being more parts-based. Given the computational load imposed in creating high-complexity face cells (in the form of algorithmic information and logical depth), and in the absence of a compelling advantage to using high-complexity cells, we suggest face representations consist of a mixture of low- and medium-complexity face cells.

Introduction

The ability to recognize individual faces and to interpret facial expressions is central to human social interactions, as well as the social interactions of some non-human primates (Leopold & Rhodes, 2010; Parr, 2011; Parr, Winslow, Hopkins, & de Waal, 2000). Neurons whose responses are selective for faces have been demonstrated in humans and non-human primates, both neurophysiologically and through fMRI (Duchaine & Yovel, 2015; Freiwald, Duchaine, & Yovel, 2016; Haxby, Hoffman, & Gobbini, 2000; Kanwisher & Yovel, 2006; Nestor, Plaut, & Behrmann, 2016; Parr, Hecht, Barks, Preuss, & Votaw, 2009; Tsao, 2014; Tsao & Livingstone, 2008). How those neurons are used to represent face is a matter of extensive research.

Neurophysiological evidence indicates that faces can be encoded using a neural population code, with each face represented by a point within a high-dimensional face response space (Chang & Tsao, 2017; Eifuku, De Souza, Tamura, Nishijo, & Ono, 2004; Rolls & Tovée, 1995; Young & Yamane, 1992). Each neuron forms an axis of the neural face space. Neural responses within the high-dimensional response space can be visualized through dimensional reduction techniques such as multidimensional scaling (MDS) or principal components analysis (PCA). The dimensionality of face space has been estimated psychophysically to be on the order of 100 (Meytlis & Sirovich, 2007; Sirovich & Meytlis, 2009). Within an axis-based face space the average face may have special status as defining the origin of the face space coordinate system (Leopold, Bondar, & Giese, 2006; Leopold, O'Toole, Vetter, & Blanz, 2001; Rhodes & Jeffery, 2006; Tsao & Freiwald, 2006; Wilson, Loffler, & Wilkinson, 2002), though this remains controversial.

The use of axis-based high-dimensional neural response spaces has become commonplace for interpreting neural data, not just for describing faces responses but also for describing neural responses to object stimuli in general. Those using an axis-based approach based to characterize neurophysiological object responses (non-face) include MDS studies (Kayaert, Biederman, & Vogels, 2005; Kiani, Esteky, Mirpour, & Tanaka, 2007; Lehky & Sereno, 2007; Murata, Gallese, Luppino, Kaseda, & Sakata, 2000; Op de Beeck, Wagemans, & Vogels, 2001; Romero, Van Dromme, & Janssen, 2013; Sereno & Lehky, 2018; Sereno, Sereno, & Lehky, 2014), as well as those based on PCA (Baldassi et al., 2013; Chang & Tsao, 2017). This axis-based approach can be extended to interpreting fMRI data for objects, in this case using each voxel as an axis for the high-dimensional response space (Bracci & Op de Beeck, 2016; Connolly et al., 2012; Kravitz, Peng, & Baker, 2011; Kriegeskorte et al., 2008).

There are two perspectives on the development of face processing circuitry in temporal cortex. The first is that there are face-specific neural processes that are hard-wired (domain specificity) (Kanwisher, 2000; McKone, Kanwisher, & Duchaine, 2007; Tsao & Livingstone, 2008; Yovel & Kanwisher, 2004). The second is that temporal cortex can also acquire processing for different classes of non-face stimuli through experience (expertise) (Cowell & Cottrell, 2013; Gauthier, Behrmann, & Tarr, 1999; Gauthier, Skudlarski, Gore, & Anderson, 2000; Gauthier & Tarr, 1997; Tong, Joyce, & Cottrell, 2008; P. Wang, Gauthier, & Cottrell, 2016). For the purposes of this study we remain agnostic between these possibilities, focusing on the face representations themselves and not their development.

A neural face space is defined by the properties of the individual neurons that constitute the axes of the space (plus possible interactions within the face cell population if the face space is nonlinear). Therefore, a central task in characterizing face space is to characterize those individual neurons. As with high-level representations of objects in general, the complexity of face representations at the population level reflects the complexity in the organization of individual face cell receptive fields. Given the complexity of face cell organization, a fruitful approach is to constrain the possibilities of what aspects of facial features are important to face cells. An interesting example of this sort of analysis is given by Freiwald, Tsao, and Livingstone (2009) for monkey inferotemporal cortex, based on the geometry of facial features and parts/whole organization using simple cartoon face stimuli. In contrast, we have hesitations concerning the conclusions of Chang and Tsao (2017) that face space corresponds to one unique linear space that they have discovered. We believe that other linear face spaces are also consistent with their data under their mathematical analysis methods, as we will consider in the Discussion section.

Here we suggest that "image complexity" may be a novel way to characterize face representations, where complexity is given a well-defined mathematical definition. We approach the issue of face complexity theoretically by using a mathematical technique based on tensor decomposition (Bro, 1997; Cichocki et al., 2015; Favier & de Almeida, 2014; Kolda & Bader, 2009; Rabanser, Shchur, & Günnemann, 2017; Sidiropoulos et al., 2017), that allows us to vary the complexity of the face cells that constitute the encoding dimensions. Complexity as used here is defined as Kolmogorov complexity, also known as algorithmic information (Adriaans, 2019; Cover & Thomas, 2006; Grünwald &

Vitányi, 2008a, 2008b; Li & Vitányi, 2008), as well as another complexity measure called logical depth (Bennett, 1988, 1994; Zenil, Delahaye, & Gaucherel, 2012). Comparing properties of face representations with different complexities is the central focus of this study.

Tensor analysis decomposes faces into a set of components, called tensorfaces. Under the algorithm used here, the original faces can be reconstructed by a weighted linear sum of the tensorfaces (under other tensor algorithms the mixing can be multilinear). A set of components and their associated weights can be thought of as model face cells and their firing rates. This tensor decomposition is analogous to reconstructing faces using a weighted linear sum of components derived from principal components analysis (PCA) (Turk & Pentland, 1991), or a weighted linear sum of components derived from independent components analysis (ICA) (Bartlett, Movellan, & Sejnowski, 2002; Bartlett & Sejnowski, 1997), or a weighted linear sum of components derived from nonnegative matrix factorization (NMF) (Y. Wang, Jia, Hu, & Turk, 2005), among other possibilities. These decomposition algorithms differ based on what constraint is applied to the decomposition. PCA produces components subject to the constraint that they are orthogonal, ICA that they are statistically independent, and NMF that they are non-negative. Another member of this genre of decomposing faces into linear components is active appearance modeling (AAM) (Cootes, Edwards, & Taylor, 2001; G. J. Edwards, Cootes, & Taylor, 1998), as used by Chang and Tsao (2017). AAM similar to PCA except that fiducial markers are placed on the face images by hand to help with aligning features during decomposition (and thus this is not an automatic algorithm). In this study the constraint we place on the face decomposition is that the components

have fixed complexity.

Tensor decomposition is not a single algorithm but a category of algorithms. The term “tensorface” was originated by Vasilescu and Terzopoulos (2002) for a particular nonlinear (multilinear) tensorface decomposition algorithm (see also Vasilescu & Terzopoulos, 2002; Vasilescu & Terzopoulos, 2003, 2005, 2011). We use a different tensor algorithm to linearly decompose faces (Phan, Cichocki, Tichavský, Zdunek, & Lehky, 2013), one that is, as mentioned, constrained to produce tensorfaces with specified image complexity. Each tensorface can be visualized as a matrix of pixels, and the rank of that matrix serves as the direct proxy of image complexity when running the algorithm. (Rank is defined as the maximum number of linearly independent columns or rows in a matrix.) Matrix rank is the input parameter specified for the algorithm to specify the face complexity we want, while Kolmogorov complexity and logical depth are calculated from the output tensorfaces after the algorithm is run.

We are not advocating the algorithm used here as a specific model of biological face cells and we are not interested in creating a canonical face space (as we believe such an effort is premature). Rather, we are interested in exploring the concept of “complexity” in face representations in general using this algorithm as an example, with hopes that this concept will prove useful in future investigations of biological face processing. We create tensorfaces with specified complexity by adding a rank constraint to a tensor decomposition algorithm. Creating other face representations with specified complexity could also be done by adding rank constraints to other decomposition algorithms not based on tensor algorithms. An example of this is PCA decomposition with a rank constraint added (Yu, 2016). We confine ourselves here to issues of basic

face representation and do not attempt to categorize different views of individual faces, as we are not creating a full face recognition model.

Methods

i. Face stimulus set

Synthetic colored faces were generated using FaceGen software (Singular Inversions, Inc.; facegen.com). Some details of the FaceGen algorithms are discussed in Blanz and Vetter (1999). The face set included equal numbers of males and females, and equal numbers from the four racial groups provided by the software: African, East Asian, European, and South Asian. As we included color in our consideration of facial representations, we wanted to have different skin tones in the face sample. Example faces are shown in Figure 1. Within each racial group, we generated faces with random shape, color, and texture parameters using the "Generate" button in the software control panel. This automatic, random generation of faces sometimes led to unnatural looking faces, which either were rejected from inclusion in the face set or had their parameters manually tweaked. Faces had zero rotation. The illumination angle was 0^o azimuth and 0^o elevation. Tensor decomposition was carried out on a sample set usually consisting of 128 faces (examples shown in Figure 2a). The resulting tensorfaces were tested by using them to reconstruct a different set of faces, a test set containing 40 faces (Figure 2b).

For this initial study of tensorface complexity we have kept the face sample set simple, all front facing with identical illumination. The multiway nature of tensor decompositions would allow inclusion of additional image parameters as additional dimensions to the input tensor containing the sample face set. For example, representations of rotated faces (changes in viewpoint) are an important aspect of face

identification (Fang, Murray, & He, 2007; Freiwald & Tsao, 2010; Jiang, Blanz, & O'Toole, 2006; Natu et al., 2010; Noudoost & Esteky, 2013; Perrett et al., 1991; Perrett et al., 1985; Ramírez, Cichy, Allefeld, & Haynes, 2014). Face rotation in depth (azimuth) could be added as a fifth dimension to the current four-dimensional input tensor (x spatial dimensions, y spatial dimension, color, and different individuals), and analogously for additional image parameters.

ii. Tensor decomposition algorithm background

We computed face components using tensor methods rather than the matrix methods used in PCA, ICA, and NMF. PCA and other matrix techniques can only deal with 2D data. That means each face image must be unfolded or *vectorized* into one long 1D vector. Then the vectors for the individual faces are placed together to form the columns of a 2D matrix, which serves as the input to PCA (Figure 3a). In contrast, tensor methods can be applied to data with an arbitrarily large number of parameter dimensions. Therefore, images do not need to be vectorized, and each pixel within the image retains its spatial context during the decomposition process (Figure 3b). Here we did tensor decompositions of 4D face data structures, which included two spatial dimensions for each face, color as the third dimension, and different individuals as the fourth dimension. While PCA and other matrix methods use linear algebra, tensor methods use multilinear algebra that allows consideration of multiple parameter dimensions concurrently. While we used a multilinear algorithm to decompose faces into a set of components and weights, the faces were reconstructed linearly as the weighted sum of the components.

iii. Tensor decomposition algorithm

Matlab code and example face files are available at

https://github.com/slehky/tensorfaces-neco .

A. MATRIX OPERATORS

The tensor decomposition algorithm used here uses the Kronecker, Khatri-Rao and Hadamard products between two matrices as well as Hadamard division. The properties and applications of those matrix operators have been reviewed by Van Loan (2000) as well as Liu and Trenkler (2008) and are included in the Matlab toolbox software of Bader, Kolda, and others (2017) and Phan (2018). Here we briefly these operators before describing the algorithm.

a. The Kronecker product $\otimes$ of the matrix $A \in \mathbb{M}^{m,n}$ and the matrix $B \in \mathbb{M}^{p,q}$ is defined as:

$$A \otimes B = \begin{bmatrix} a_{11}B & \dots & a_{1n}B \\ \vdots & & \vdots \\ a_{m1}B & \dots & a_{mn}B \end{bmatrix} \tag{1}$$

For example,

$$\begin{bmatrix} a & b & c \\ d & e & f \end{bmatrix} \otimes \begin{bmatrix} g & h & i \\ j & k & l \end{bmatrix} = \left[\begin{array}{ccc|ccc|ccc} ag & ah & ai & bg & bh & bi & cg & ch & ci \\ aj & ak & al & bj & bk & bl & cj & ck & cl \\ \hline dg & dh & di & eg & eh & ei & fg & fh & fi \\ dj & dk & dl & ej & ek & el & fj & fk & fl \end{array}\right] \tag{2}$$

The Kronecker product is the generalization to matrices of the vector outer product. It is sometimes called the *tensor product*.

b. The Khatri-Rao product $\odot$ of the matrix $A \in \mathbb{M}^{m,n}$ and the matrix $B \in \mathbb{M}^{p,n}$ is defined as the Kronecker product between corresponding columns of the two matrices:

$$A \odot B = \begin{bmatrix} a_1 \otimes b_1, & a_2 \otimes b_2, & \dots & a_n \otimes b_n \end{bmatrix} \tag{3}$$

where a_n and b_n are the nth column vectors. The Khatri-Rao product is only defined if the matrices have the same number of columns. For example,

$$\begin{bmatrix} a & b & c \\ d & e & f \end{bmatrix} \odot \begin{bmatrix} g & h & i \\ j & k & l \end{bmatrix} = \begin{bmatrix} ag & bh & ci \\ aj & bk & cj \\ dg & eh & fi \\ dj & ek & fl \end{bmatrix} \tag{4}$$

c. The Hadamard product $\circledast$ between two matrices $A \in \mathbb{M}^{m,n}$ and $B \in \mathbb{M}^{m,n}$ is defined as the element-wise multiplication between them:

$$A \circledast B = \left[A\right]_{ij} \left[B\right]_{ij} \tag{5}$$

for all $1 \le i \le m$, $1 \le j \le n$. The Hadamard product is only defined if the two matrices have the same dimension. For example,

$$\begin{bmatrix} a & b & c \\ d & e & f \end{bmatrix} \circledast \begin{bmatrix} g & h & i \\ j & k & l \end{bmatrix} = \begin{bmatrix} ag & bh & ci \\ dj & ek & fl \end{bmatrix} \tag{6}$$

Hadamard division $\oslash$ is defined analogously as element-wise division between two matrices.

B. THE MODEL

The tensor decomposition algorithm we use is described by Phan et al. (2013). We have not made any changes to it, but do present it in more detail here. The algorithm is a variant of the CANDECOMP/PARAFAC (CP) algorithm (Carroll & Chang, 1970; Harshman, 1970). It falls into the category of structured or constrained CP incorporating a PARALIND algorithm (Bro, Harshman, Sidiropoulos, & Lundy, 2009). Constrained CP algorithms have been reviewed by Favier and de Almeida (2014). Although this algorithm is derived from CP, it is not based on an outer product sum of rank-1

components as is done by CP. Rather, the decomposition is based on a Kronecker product between two tensors, namely a components tensor and a weights tensor. The structured CP algorithm used here can be viewed in some sense as intermediate between two commonly used tensor decomposition models, the conventional CP model (Carroll & Chang, 1970; Harshman, 1970) and the Tucker model (Tucker, 1966), and incorporates aspects of both. The reason for using a structured CP model in this study rather than either the conventional CP or Tucker models are briefly outlined in Phan et al. (2013).

Consider a data tensor $\mathcal{Y}$ of size $I_1 \times I_2 \times \cdots \times I_N$. Our aim is to represent this tensor by multiple basis components (tensorfaces) in which the components were specified to have various levels of complex structures. In our case we are dealing with a 4-way tensor ($N = 4$) with size 200 (pixels) × 200 (pixels) × 3 (color channels) × 128 (individuals), which represents 128 colored face images concatenated into a single data structure. All calculations are performed with the color channels converted from RGB to CIE 1976 L*A*B color space, which approximates human color vision more closely.

The tensor decomposition algorithm we use factors the tensor $\mathcal{Y}$ into a sum of components (basis patterns) and mixing weights:

$$\mathcal{Y} \approx \sum_{p=1}^{P} \mathcal{X}_p \otimes \mathcal{A}_p \tag{7}$$

where $\otimes$ denotes the generalized Kronecker product, $\mathcal{X}_p$ are components (tensorfaces in our case) and $\mathcal{A}_p$ the associated coefficient tensors (weights), for $p = 1,2,\ldots,P$ (P =number of patterns). Unlike matrix decompositions such as PCA, ICA, and NMF where the weight for each component must be a scalar, tensor decompositions can allow weights

to be a higher-order tensor, allowing multilinear mixing during reconstruction (Vasilescu & Terzopoulos, 2002). However in this model we arranged the algorithm such that the weights tensor is order-1 and rank-1, thereby making the weight for each component scalar and the mixing linear. The components tensor $\mathcal{X}_p$ is a higher-rank tensor. Although the face reconstruction is linear here, the decomposition of the input face tensor $\mathcal{Y}$ itself into weights $\mathcal{A}_p$ and components $\mathcal{X}_p$ is multilinear. Various decomposition algorithms can be carried out subject to different constraints on $\mathcal{X}_p$, such as orthogonality (PCA), statistical independence (ICA), or non-negativity (NMF), as well as possible constraints on $\mathcal{A}_p$ such as sparseness. Here the constraint was on the tensor rank of $\mathcal{X}_p$, where we take rank to be a measure of the complexity of the tensorface patterns. As the number of components P is limited, the decomposition will only be approximately equal to the original data $\mathcal{Y}$.

For our model, the weights $\mathcal{A}_p$ are of size $J_{p1} \times J_{p2} \times \ldots \times J_{pM}$, with their order (dimensionality) given by M. Within the algorithm we defined $\mathcal{A}_p$ to be order $M = 1$, and thus $\mathcal{A}_p$ is represented by a $n \times 1$ vector, where n is number of face images in the input set (typically n=128). The patterns $\mathcal{X}_p$ are of size $K_{p1} \times K_{p2} \times \ldots \times K_{pL}$, with their order given by L. $\mathcal{X}_p$ are of order $L = 3$, and form $m \times m \times 3$ sized tensors where m is the size of the input image in pixels, in our case always 200 pixels.

$\mathcal{A}_p$ and $\mathcal{X}_p$ are rank-S_p and rank-R_p tensors, respectively. The rank of $\mathcal{A}_p$ is always $S_p = 1$. The rank of $\mathcal{X}_p$ is set over the range $R_p = 2$ to $R_p = 32$ for different runs

of the tensor decomposition algorithm. Examining the effects of changing R_p (changing the complexity of tensorfaces) is a central concern of this study.

The subscript p for different tensorface patterns is included for generality, but we hold both the order and the rank of both $\mathcal{A}_p$ and $\boldsymbol{\mathcal{X}}_p$ constant for all p. Notably the rank of $\boldsymbol{\mathcal{X}}_p$ is constant for the entire population of tensorfaces during a single run. Although we had the option to set the rank of each tensorface individually, we do not do so here.

In implementing the model, $\mathcal{A}_p$ and $\boldsymbol{\mathcal{X}}_p$ can be expressed as sets of matrices $\mathbf{U}^{(m)}$ and $\mathbf{V}^{(l)}$ through *canonical polyadic* decomposition (CPD) (Carroll & Chang, 1970; Harshman, 1970) of $\boldsymbol{\mathcal{A}}_p$ and $\boldsymbol{\mathcal{X}}_p$ (Figure 4):

$$\boldsymbol{\mathcal{A}}_p = \boldsymbol{\mathcal{I}} \times_1 \mathbf{U}_p^{(1)} \times_2 \mathbf{U}_p^{(2)} \cdots \times_M \mathbf{U}_p^{(M)} \tag{8}$$

$$\boldsymbol{\mathcal{X}}_p = \boldsymbol{\mathcal{I}} \times_1 \mathbf{V}_p^{(1)} \times_2 \mathbf{V}_p^{(2)} \cdots \times_L \mathbf{V}_p^{(L)} \tag{9}$$

where $\times_n$ is tensor matrix multiplication along the *nth* mode (dimension), $\boldsymbol{\mathcal{I}}$ is a tensor with ones along the superdiagonal, the superscripts indicate the dimension number, and the subscripts the pattern number. The sizes of the matrices were $\mathbf{U}^{(m)} \in \mathbb{R}^{J_{p_m} \times S_p}$ and $\mathbf{V}^{(l)} \in \mathbb{R}^{K_{p_l} \times R_p}$. For $\mathcal{A}_p$, which has order $M_p = 1$ and rank $S_p = 1$, $\mathbf{U}^m$ reduces to a single 128×1 vector. For $\boldsymbol{\mathcal{X}}_p$, which has order $L_p = 3$ and rank R_p as variably defined, there were three matrices in which the number of rows was set equal to image dimensions and the number of columns equal to tensorface rank. Assuming rank $R_p = 8$ as an example, the sizes of the three matrices associated with each pattern were 200×8, 200×8, and $3 \times$

8. It is here that the rank constraint enters explicitly into the calculations. This model of $\mathcal{Y}$ is equivalent to a CP decomposition with total rank $T = \sum_{p=1}^{P} R_p S_p$.

The tensor decompositions in Eqs. (7)-(9) are particular cases of Kronecker tensor decomposition (KTD), and also constitute a generalized model of block term decomposition (BTD) (De Lathauwer, 2008a, 2008b; Sorber, Van Barel, & De Lathauwer, 2013). If all $\mathcal{A}_p$ are of order-1, i.e., $M = 1$, as was the case here, then the above model is simplified into the rank-R_p ∘ rank-1 BTD (Sorber et al., 2013).

In order to derive the algorithm that updates the factor matrices of the basis patterns and the weight tensors, we rewrite the tensor decomposition in Eq. (7) with rank constraints in Eq. (8) and Eq. (9), in the form of the CP decomposition.

Lemma 1. *The decomposition in Eqs.* (7)-(9) *is equivalent to a structured canonical polyadic decomposition*:

$$\mathcal{Y} \approx \mathcal{I} \times_1 \mathbf{W}^{(1)} \times_2 \mathbf{W}^{(2)} \cdots \times_N \mathbf{W}^{(N)} \tag{10}$$

where the factor matrices $\mathbf{W}^{(n)}$ *are given by*

$$\mathbf{W}^{(n)} = \begin{cases} \tilde{\mathbf{V}}^{(n)} \mathbf{Q}_X & \text{n=1,2,…,L} \\ \tilde{\mathbf{U}}^{(n)} \mathbf{Q}_A & \text{n=L+1,…,N} \end{cases} \tag{11}$$

$$\tilde{\mathbf{U}}^{(n)} = \left[\mathbf{U}_1^{(n)}, \mathbf{U}_2^{(n)}, \ldots, \mathbf{U}_P^{(n)} \right] \tag{12}$$

$$\tilde{\mathbf{V}}^{(n)} = \left[\mathbf{V}_1^{(n)}, \mathbf{V}_2^{(n)}, \ldots, \mathbf{V}_P^{(n)} \right] \tag{13}$$

$$\mathbf{Q}_X = \mathtt{blkdiag}\left(\mathbf{I}_{R_1} \otimes \mathbf{1}_{S_1}^T, \mathbf{I}_{R_2} \otimes \mathbf{1}_{S_2}^T, \ldots, \mathbf{I}_{R_P} \otimes \mathbf{1}_{S_P}^T \right) \tag{14}$$

$$\mathbf{Q}_A = \mathtt{blkdiag}\left(\mathbf{1}_{R_1}^T \otimes \mathbf{I}_{S_1}, \mathbf{1}_{R_2}^T \otimes \mathbf{I}_{S_2}, \ldots, \mathbf{1}_{R_P}^T \otimes \mathbf{I}_{S_P} \right) \tag{15}$$

In Eqs. (14) and (15), $\otimes$ is the Kronecker product, defined in Eq. (1), $\mathbf{I}$ is the tensor with ones along the superdiagonal, $\mathbf{1}$ is a vector of ones, and T is the matrix transpose operator.

In this decomposition, due to properties of the Kronecker product each component (column) of $\mathbf{U}_p^{(n)}$ was replicated S_p times in $\mathbf{W}^{(n)}$ for $n \leq L$, and each component of $\mathbf{V}_p^{(n)}$ was replicated R_p times in $\mathbf{W}^{(n)}$ for $n > L$. Such behavior is related to the rank-overlap problem (the decomposition creates multiple identical components) which often exists in real-world signals such as chemical data, flow injection analysis (FIA) data (Bro, 1998; Bro et al., 2009), or spectral tensors of EEG signals (Phan et al., 2013). However, in our case this does not lead to the creation of multiple identical tensorfaces $\boldsymbol{\mathcal{X}}_p$ because each $\boldsymbol{\mathcal{X}}_p$ is the result of combining all factor matrices $\mathbf{V}_p^{(n)}$.

The structured CPD in Lemma 1 is a particular case of parallel factor analysis (CANDECOMP/PARAFAC) (Carroll & Chang, 1970; Harshman, 1970) with linearly dependent loadings (PARALIND) (Bro et al., 2009) in which the dependency matrices (Bro et al., 2009) are fixed and given in Lemma 1. Discussions on the uniqueness of the CPD with linearly dependent loadings can be found in (Guo, Miron, Brie, & Stegeman, 2012; Stegeman & Lam, 2012).

C. ALGORITHM

We us an alternating least squares (ALS) algorithm to learn the approximate factorization of $\boldsymbol{\mathcal{Y}}$ into $\boldsymbol{\mathcal{A}}_p$ and $\boldsymbol{\mathcal{X}}_p$. The ALS algorithm is applied to the structured CPD in Lemma 1 in order to iteratively update $\tilde{\mathbf{U}}^{(n)}$ and $\tilde{\mathbf{V}}^{(n)}$:

$$\tilde{\mathbf{U}}^{(n)} \leftarrow \mathbf{G}_n \mathbf{Q}_L^T \left(\mathbf{Q}_L \Gamma_n \mathbf{Q}_L^T \right)^{-1}, \quad \left(n = 1,2,\ldots,S \right) \tag{16}$$

$$\tilde{\mathbf{V}}^{(n)} \leftarrow \mathbf{G}_n \mathbf{Q}_M^T \left(\mathbf{Q}_M \Gamma_n \mathbf{Q}_M^T \right)^{-1}, \quad \left(n = S+1,2,\ldots,N \right) \tag{17}$$

where

$$\mathbf{G}_n = \mathbf{Y}_{(n)} \left(\mathbf{W}^{(N)} \odot \ldots \odot \mathbf{W}^{(n+1)} \odot \mathbf{W}^{(n-1)} \odot \ldots \odot \mathbf{W}^{(1)} \right) \tag{18}$$

$$\Gamma_n = \left(\mathbf{W}^{(1)T} \mathbf{W}^{(1)} \right) \circledast \ldots \circledast \left(\mathbf{W}^{(n-1)T} \mathbf{W}^{(n-1)} \right) \circledast \left(\mathbf{W}^{(n+1)T} \mathbf{W}^{(n+1)} \right) \circledast \ldots \circledast \left(\mathbf{W}^{(N)T} \mathbf{W}^{(N)} \right) \tag{19}$$

and $\odot$ and $\circledast$ denote the Khatri-Rao product (Eq. (3)) and Hadamard product (Eq. (5)) respectively.

Updating $\tilde{\mathbf{U}}^{(n)}$ and $\tilde{\mathbf{V}}^{(n)}$ in turn allows updates of $\boldsymbol{\mathcal{A}}_p$ and $\boldsymbol{\mathcal{X}}_p$ through Eqs. (8) and (9). ALS acts to iteratively adjust the factors $\boldsymbol{\mathcal{A}}_p$ and $\boldsymbol{\mathcal{X}}_p$ in Eq. (7) so as to minimize the Frobenius error between the original data tensor $\boldsymbol{\mathcal{Y}}$ and the reconstructed data tensor $\hat{\boldsymbol{\mathcal{Y}}}$, $Error = \left\| \boldsymbol{\mathcal{Y}} - \hat{\boldsymbol{\mathcal{Y}}} \right\|_F$. $\hat{\boldsymbol{\mathcal{Y}}}$ is calculated from the estimated $\boldsymbol{\mathcal{A}}_p$ and $\boldsymbol{\mathcal{X}}_p$ during each iteration. The ALS algorithm updates each parameter sequentially, in contrast to error minimization using a gradient descent algorithm, which updates all parameters simultaneously. The error minimization loop is begun by initializing $\boldsymbol{\mathcal{A}}_p$ and $\boldsymbol{\mathcal{X}}_p$ using the singular value decomposition (SVD) of $\boldsymbol{\mathcal{Y}}$. SVD is performed on a matrix in which each column is formed by vectorizing a face image (creating a vector with 200 × 200 × 3 pixels), with the number of columns equal to the number of images (128 images). The left SVD vector is saved to a tensor with image dimensions, then approximated by a low rank tensor using CANDECOMP/PARAFAC, and finally assigned as initialization of $\boldsymbol{\mathcal{X}}_p$. $\boldsymbol{\mathcal{A}}_p$ is initialized using the right SVD vector.

Although we did not impose non-negativity constraints, they could be included using the iterative algorithm below (Cichocki, Zdunek, Phan, & Amari, 2009; Lantéri, Soummer, & Aime, 1999; Lee & Seung, 1999; Lin, 2007):

$$\begin{aligned}\tilde{\mathbf{U}}^{(n)} &= \tilde{\mathbf{U}}^{(n)} \circledast \left(\mathbf{G}_n \mathbf{Q}_L^T\right) \oslash \left(\tilde{\mathbf{U}}^{(n)} \mathbf{Q}_L \Gamma_n \mathbf{Q}_L^T\right), \qquad \left(n = 1,2,\ldots,S\right) \\ \tilde{\mathbf{V}}^{(n)} &= \tilde{\mathbf{V}}^{(n)} \circledast \left(\mathbf{G}_n \mathbf{Q}_M^T\right) \oslash \left(\tilde{\mathbf{V}}^{(n)} \mathbf{Q}_M \Gamma_n \mathbf{Q}_M^T\right), \qquad \left(n = S+1,2,\ldots,N\right)\end{aligned} \tag{20}$$

where $\oslash$ denotes (element-wise) Hadamard division.

iv. Reconstruction error

We measure the error between original faces and faces reconstructed from a set of tensorface components. Error is calculated as the Frobenius norm (Euclidean matrix norm) of the pixel-wise difference between the original face and reconstructed face, divided by the Frobenius norm of the original face:

$$Err = \frac{\sqrt{\sum_{i=1}^{n}\sum_{j=1}^{m}\left(a_{ij} - \hat{a}_{ij}\right)^2}}{\sqrt{\sum_{i=1}^{n}\sum_{j=1}^{m} a_{ij}^{\ 2}}} \tag{21}$$

Reconstructions and reconstruction errors are meant to illustrate the amount of information contained in the tensorfaces and associated weights, and are not intended to imply that the brain reconstitutes face pixel maps somewhere along the visual pathways.

v. Displaying tensorfaces

The pixel values of the tensorfaces produced by the tensor decomposition algorithm generally extend beyond the range of values allowed by the *L*a*b* color space, as the decomposition was not constrained to fit requirements of the color space. The *L* (luminance) channel allows values on the range 0–100, while the *a* (red-green opponent)

and *b* (blue-yellow opponent) channels both allow values on the range -100–100. For display purposes, each tensorface was individually normalized to fill out the allowable values of the color space. The L channel was separately normalized, while the *a* and *b* channels were jointly normalized so as not to affect the color balance between the two. After the *L*a*b* color space normalization, the tensorface was converted to RGB color space for display.

vi. Kolmogorov complexity (algorithmic information)

The Kolmogorov complexity of a pattern, or equivalently the algorithmic information it contains, is the length of the shortest algorithm required to reproduce it (Grünwald & Vitányi, 2008a, 2008b; Li & Vitányi, 2008). In other words, the complexity of a pattern is the size of the most compressed description of the pattern. The concept of Kolmogorov complexity was independently introduced by Solomonoff (1964), Kolmogorov (1965), and Chaitin (1969), and is sometimes known as Kolmogorov-Chaitin-Solomonoff (KCS) complexity.

To illustrate the difference between algorithmic information and Shannon information, consider a communications channel in which only two messages are possible, either Face A or Face B. Whenever one of those faces is transmitted, the Shannon information is one bit because there are only two possibilities. However, the algorithmic information transmitted is vastly higher because it requires many bits to form a complete description of the face.

While the definition of Kolmogorov complexity is straightforward, actually determining its value is problematic as there is no systematic way to determine the most compact description of a pattern. In other words, Kolmogorov complexity is

uncomputable (no algorithm exists). Because Kolmogorov complexity is uncomputable, we use lossless compression algorithms to approximate an upper bound to complexity of the tensorfaces (Ruffini, 2017).

Here we base our estimate of the Kolmogorov complexity of tensorfaces on the file size of the tensorface images after they underwent a lossless compression. That is done by saving a tensorface image in PNG image format and noting the number of bits in the saved file. The PNG image format uses an efficient, lossless compression algorithm called DEFLATE, which is based on the Lempel-Ziv algorithm (Lempel & Ziv, 1976; Ruffini, 2017) together with Huffman coding. To further compress the tensorface files beyond the standard PNG format, we use the program ImageOptim (imageoptim.com), which ran an additional set of compression algorithms, also based on DEFLATE, that are more efficient but too time consuming for ordinary use, combining the results of those compression algorithms. The algorithms included Zopfli, PGNOUT, OptiPGN, AdvPGN, and PGNCrush. Using ImageOptim reduces tensorface file sizes beyond the standard PNG compression by an amount depending on tensorface rank, ranging on average from 19% for rank=2 tensorfaces to 9% for rank=32 tensorfaces.

The number of bits in the compressed image was then normalized by the number of pixels in the image, giving an estimate of Kolmogorov complexity as bits/pixel for the compressed image. While all tensorface images had identical file sizes when uncompressed and initially had 24 bits/pixel, some tensorfaces were more compressible than others, reflecting image complexity.

After setting the desired rank of the tensor decomposition, the face sample set was decomposed into 100 components and then the decomposition was replicated ten times to

produce a total of 1000 tensorfaces. Kolmogorov complexity was averaged over those 1000 tensorfaces. The same set of tensorfaces was used in calculations of logical depth, power spectra, and globality described below.

In this analysis the tensor decomposition algorithm provides us with model face receptive fields (tensorfaces). Such model receptive fields are presented as images of receptive fields, analogous to the way that V1 Gabor receptive fields are presented as images of the receptive fields. Having access to such receptive field images makes it feasible to employ the mathematical methods for evaluating algorithmic complexity. On the other hand, in an experimental neurophysiological situation, producing images of face cell receptive fields is problematic because of the intractability with finding optimal face stimuli given undefined spatial nonlinearities in the receptive fields. We will discuss nonlinearities in face cells further below.

vii. Logical depth

Logical depth is another way to measure the complexity of tensorfaces. In the present context, logical depth is the duration of computational time required to restore an image back to its original state after it has been maximally compressed in a lossless manner. The concept of logical depth was originated by (Bennett, 1988, 1994) and has previously been applied to the characterization of images by Zenil et al. (2012). The basic idea is that objects which "contain internal evidence of a nontrivial causal history" (Bennett, 1988) have complex structure that require more computational time to reconstitute from their shortest descriptions (maximally compressed states) than objects without complex structure.

While Kolmogorov complexity can be thought of as measuring complexity in

terms of space (the length of the shortest description of an object), logical depth measures complexity in terms of time (the number of computational steps required to reconstruct the object from that shortest description). An important difference between the two is that Kolmogorov complexity considers both structured states and random states to be complex, but logical depth only considers structured states as complex while treating both trivial and random states as non-complex. Thus, as pointed out by Zenil et al. (2012), logical depth may lie closer to our intuitive concept of complexity than Kolmogorov complexity.

To measure logical depth, we first compress the tensorface images by running the program dzip within Matlab (The Mathworks Inc., Natick, MA). Then the image is uncompressed using dunzip, and the elapsed time to perform the uncompression was measured using the Matlab tic-toc timer function. The uncompression time is measured 1000 times for each tensorface and then averaged. Timing is measured with no user applications running aside from Matlab, with WiFi and Bluetooth turned off, and nothing attached to any of the computer ports.

Dzip implements the DEFLATE lossless compression algorithm. Both dzip and dunzip are available for download from the Matlab File Exchange: www.mathworks.com/matlabcentral/fileexchange/8899-rapid-lossless-data-compression-of-numerical-or-string-variables .

viii. Power spectra

We calculated the 2D spatial frequency power spectra of tensorfaces having different levels of complexity. The tensorfaces were first converted from color to grayscale images. The 2D spectra were then transformed to 1D by performing rotational

averaging (averaging spectral power over all orientations in the images).

ix. Globality index

We define the globality of a tensorface component as the fraction of the face it covers. This is the number of pixels in a tensorface divided by the average number of pixels in a face (averaged over all faces in the sample). The number of pixels in the faces is simple to determine, as the faces are on a black background and easy to segment. The number of pixels in a tensorface is more difficult as the tensorfaces had a continuum of values that could blend in with the background. Including all tensorface pixels that differed just a tiny bit from the background would greatly inflated the size of the tensorfaces and therefore their globality.

We therefore follow the following procedure to exclude small pixel values from the globality calculations and isolate the high-activity regions of the tensorfaces. First, we convert the tensorfaces to grayscale and subtracted the background, leaving the tensorfaces on a black background. Then we set a grey threshold level and exclude pixels below that level. The threshold is set using Otsu's method (Otsu, 1979), which minimizes the intraclass variance of the pixels above and below threshold (Matlab command graythresh in the Image Processing Toolbox). The grayscale tensorface is then binarized based on that threshold level, with pixels above threshold set to white and those below threshold set to black. This thresholding typically leaves the high-activity tensorface regions as a set of disjoint white patches. To create a unitary tensorface region for purposes of globality calculations, all the individual white patches are enclosed by their convex hull (Matlab command convhull). The interior of this convex hull constitutes the high-activity region of the tensorface. Finally, the area of a tensorface enclosed by the

convex hull, measured in pixels, is divided by the area of the face. The resulting fraction is the globality index of the tensorface.

x. Selectivity and Sparseness

We use kurtosis as a measure of both the selectivity of single tensorfaces and also the sparseness of populations of tensorfaces. Kurtosis is a measure of the shape of a probability distribution, in this case the distribution of tensorface responses to stimuli. A high kurtosis distribution, corresponding to high selectivity or high sparseness, emphasizes the peak and tails of the distribution with less probability in between. A low kurtosis distribution, corresponding to low selectivity or low sparseness, has a flatter distribution. By tensorface "response" to a stimulus we mean the weight associated with that tensorface when reconstructing the stimulus image.

Cell selectivity is based on the probability distribution of the responses of a single cell (single tensorface) when presented with a set of stimuli over time. Selectivity has also been called "lifetime sparseness" of single neurons (Willmore & Tolhurst, 2001). Population sparseness is based on the probability distribution of the simultaneous responses of a population of cells to a single stimulus (using the terminology of (Lehky, Kiani, Esteky, & Tanaka, 2011; Lehky, Sejnowski, & Desimone, 2005; Lehky & Sereno, 2007)). In the work presented here, responses could take both positive and negative values, which we interpret as deviations from a spontaneous level of activity, and probability distributions were roughly symmetrical.

The equation for kurtosis is:

$$kurtosis = \frac{\sum_{i=1}^{n}\left(r_i - \bar{r}\right)^4}{\left(n-1\right)s^4} - 3 \tag{22}$$

For single-cell responses, r_i refers to the response of the neuron to the *i*th stimulus, and n refers to the number of stimuli. For population responses, r_i refers to the response of the *i*th cell in the population to a single particular stimulus and n refers to the number of cells in the population. Mean response is indicated by $\bar{r}$, and the standard deviation of the responses is given by s. Subtracting three scales values so that a normal distribution has a reduced kurtosis of zero.

Kurtosis has previously been used as a measure of selectivity and sparseness in the theoretical literature (for example, Bell & Sejnowski, 1997; Olshausen & Field, 1996; Simoncelli & Olshausen, 2001). Kurtosis has also been used in the experimental literature for extrastriate cortex, including (Lehky et al., 2011; Lehky et al., 2005; Lehky & Sereno, 2007; Tolhurst, Smyth, & Thompson, 2009).

xi. Multidimensional scaling

Multidimensional scaling (MDS) (Hout, Papesh, & Goldinger, 2013) is used to visualize the face space produced by tensor decomposition. The MDS analysis is based on the tensorface weights that allow reconstruction of the faces in the sample set, after the faces have been decomposed into a set of 100 tensorfaces. We examined responses (weights) of a population of 100 tensorfaces to each of the 128 faces in the sample face set, as well as the responses of those tensorfaces to the average face calculated from the 128 faces. Thus, in total, we have population responses for 129 faces. These faces form 129 points in a 100-dimensional face space defined by the tensorface population. As the relative

positions of faces in the high-dimensional face space cannot be visualized, we use MDS to reduce the dimensionality of the face space down to two dimensions while maintaining approximate relative positions. While MDS is useful for low-dimensional visualization, the MDS algorithm has nonlinearities within it and should not be relied to produce a quantitatively accurate depiction of biological face space.

The responses of the tensorface population to a single face forms a *response vector* with a length of 100 elements, defining the position of that face in face space. The first step in performing MDS is to calculate the distances between response vectors for all 129 faces, forming a 129x129 distance matrix. A Euclidean distance metric is used. The distance matrix is then fed into the cmdscale command in the Matlab Statistics and Machine Learning Toolbox, which performs the MDS.

Results

Appearance of tensorfaces

The tensor decomposition algorithm was applied to a set of 128 sample faces (examples shown in Figure 2a), producing tensorface components. Shown are the resulting low complexity tensorfaces (rank=2, Figure 5), medium complexity tensorfaces (rank=8, Figure 6), and high complexity tensorfaces (rank=32, Figure 7). These are all shown as 200x200 pixel images. The number of tensorface components created by the algorithm was set by a parameter, and here we show examples of a decomposition of the face set into 40 components. The qualitative appearance of the components did not change as we varied the number of components over the range 5-100.

An expanded view of example tensorfaces at the three complexity levels is shown in Figure 8. As the complexity increases, the face representation progresses from crude

blobs to a clear face-like appearance.

For comparison, eigenfaces resulting from a PCA decomposition of the sample face set are shown in Figure 9. They most closely resemble the high complexity tensorfaces. The eigenfaces have rank=142, so they are more complex than any of the tensorfaces we created. Applying ICA to the sample faces produced components that qualitatively resembled the eigenfaces and which were also highly complex, with the same rank=142.

Reconstructing faces using tensorfaces

The tensorfaces were used to reconstruct a set of test faces (Figure 3b), which were different from the sample faces used to create the tensorfaces. Although the tensor decomposition algorithm used to create the tensorfaces is nonlinear, reconstructing faces from a population of tensorfaces is a linear process. These face reconstructions are used to graphically illustrate how much information is available in the tensorfaces for representing faces, and does not imply that the brain reconstitutes face bitmaps somewhere along the visual pathways.

Face reconstructions are shown in Figure 10a (reconstructed using low complexity tensorfaces), Figure 10b (reconstructed using medium complexity faces), and Figure 10c (reconstructed using high complexity tensorfaces). In all three cases the reconstructions are subjectively comparable, showing that even the blob-like, low complexity tensorfaces are capable of providing a good face representation.

Reconstruction errors are plotted as a function of the number of components in Figure 11a. Reconstruction error is the normalized Euclidean pixel-wise distance between original and reconstructed images. Not surprisingly, performance improved as tensorface

population size increased. There was a trade-off between tensorface complexity and the population size required to reach a criterion error level. A large population of low-complexity tensorfaces can match the performance of a smaller population of high-complexity tensorfaces.

Reconstruction error is plotted as a function of complexity in Figure 11b (holding the tensorface population size constant at 100). Error is large for low-complexity tensorfaces, with the error dropping greatly going to medium complexity but then staying approximately constant with further increases in complexity. There is in fact a slight rise in reconstruction error at high complexities. That is because error is being measured here on a test set of faces different from the sample set used to create the tensorfaces, and high-complexity tensorfaces have a poorer ability to generalize to new stimuli. Generalization will be further discussed below.

Computational complexity of tensorfaces

We have been measuring complexity in terms of the rank of the matrix of pixel values representing a tensorface image. The algorithm allows specification of the desired tensorface rank resulting from the decomposition process. Matrix rank is the minimum number of column vectors that can be used to generate all the columns in the matrix (or equivalently it can be done in terms of rows rather than columns). For example, a tensorface with rank=8 means that all 200 columns in the tensorface image can be generated by different linear combinations of just 8 column vectors. A matrix with a larger rank requires a larger basis set of vectors to define it and is therefore more complex.

A standard way to measure complexity within computational theory is

Kolmogorov complexity, also known as algorithmic information (Grünwald & Vitányi, 2008b; Li & Vitányi, 2008). As described in the Methods section, we operationally define Kolmogorov complexity as the number of bits per pixel required to store the tensorface image after undergoing maximal lossless compression. A more complex image requires a larger file size. The relationship between complexity measured as matrix rank and Kolmogorov complexity is plotted in Figure 12a. We see Kolmogorov complexity correlates with complexity measured by rank. A second measure of computational complexity is logical depth, the time duration of computations required to uncompress a compressed tensorface (Bennett, 1988, 1994; Zenil et al., 2012). The logical depth of tensorfaces as a function of tensorface rank is plotted in Figure 12b.

Both Kolmogorov complexity and logical depth provide similar estimates of the relative complexity of different tensorfaces. Tensorfaces that compress to a small file size (low Kolmogorov complexity) take less computational time to uncompress (small logical depth). Tensorfaces that produce large compressed file sizes (high Kolmogorov complexity) take more computational time to uncompress (large logical depth).

Note the large standard deviations in Figure 12. For each plotted point, even though all tensorfaces had identical matrix ranks, the resulting values for Kolmogorov complexity and logical depth were spread out over a broad range. The relation between tensorface rank and the two complexity measurements is therefore statistical rather than deterministic.

In addition to characterizing tensorfaces by their complexity as defined by computational theory, we can also characterize them using concepts from signal processing theory. The average spatial frequency power spectra of tensorfaces at rank=2,

8, and 32 are plotted in Figure 13a. We see that as tensorface complexity increases the spectral power at high spatial frequencies also increases (Figure 13b). Computational complexity of tensorface receptive fields (Figure 12a,b) correlates strongly with their Fourier power at high spatial frequencies.

Selectivity and sparseness of tensorfaces

Selectivity and sparseness of neural responses to stimuli are major concerns in neural coding theory. Under our terminology, selectivity is a function of the statistical distribution of responses of a single neuron to a large set of stimuli presented sequentially (Lehky et al., 2005). Sparseness is a function of the statistical distribution of responses over a population of neurons when simultaneously presented with a single stimulus. Here we quantify both selectivity and sparseness by calculating kurtosis of the appropriate probability distribution.

Tensorface selectivity is plotted as a function of rank in Figure 14a. Although there is a lot of variability for different tensorfaces, the median value of selectivity (kurtosis) is close to zero, independent of tensorface complexity (rank). That means that as one presents many faces to a particular tensorface, responses tend to be Gaussian distributed, as Gaussians have zero reduced kurtosis. Population sparseness of tensorfaces is plotted as a function of rank in Figure 14b. Although there is higher population sparseness with very low complexities, for medium and high complexities the sparseness settles down to values of around 1.0.

Sparseness and selectivity of tensorfaces are lower than reported in monkey inferotemporal cortex (Lehky et al., 2011), with single-unit selectivity = 1.88 and population sparseness = 9.61 (as measured by kurtosis). One reason tensorfaces have

lower sparseness values and lower selectivity values is because responses of tensorfaces do not include a threshold nonlinearity for response rates. In real neurons response rates can't have negative values. That causes the probability distribution of response rates to be skewed to the right, leading to higher values sparseness and selectivity values. Without that threshold nonlinearity, the response probability distributions of tensorfaces are closer to Gaussian and the sparseness and selectivity values are therefore smaller.

Another factor reducing model values of sparseness and selectivity is that tensorfaces are linear filters, as are all the face decompositions mentioned earlier (PCA, ICA, NMF, AAF), whereas biological face cells are nonlinear spatial filters, as is the case for inferotemporal object representations in general (K. Tanaka, 1996). By being nonlinear spatial filters, we mean that different portions of the receptive field sum nonlinearly to produce the total response of a neuron to an object stimulus. As the nature of the spatial nonlinearities within face cells and object cells are unknown we can't quantify their contributions to sparseness and selectivity. Spatial nonlinearities will be discussed further below.

Tensorfaces: local or global representations

We define the globality of a tensorface as the fraction of the face covered by that tensorface. Thus, local and global representations formed endpoints on a continuum rather than a dichotomy. Within that continuum, we observe tensorfaces that can be strongly local or strongly global (Figure 15a), and also everything in between.

To measure globality, we threshold the tensorfaces to include only the envelope (convex hull) of the areas that gave a "strong" activation, as described in the Methods section. Examples of such thresholded tensorfaces are shown in Figure 15b, with the

high-activation region outlined in a black line. Only the high-activation region was used in calculations of globality.

Globality as a function of tensorface complexity is plotted in Figure 15c. More complex tensorfaces have greater globality on average, although there is large variability in globality across a tensorface population. This relationship breaks down at the lowest values of rank. The discrepancy at low ranks appears to be an artifact of the methodology we are using to calculate globality. Some low-rank tensorfaces form bilaterally symmetric pairs of blobs at the left and right edges of the face. When those two widely separated blobs are enclosed in an envelope to define the high-activation region of the tensorface that inflates the area covered by the tensorface, thereby increasing the globality measure as we calculate it.

Generalization to statistically novel categories of faces

When previously examining face reconstructions based on tensorfaces (Figures 10 and 11), we used a set of test faces (Figure 3b) that closely resembled the original sample set (Figure 3a). Even though the two sets contained different individuals, they are both drawn from the same statistical distribution of face parameters in the face generating software and thus are statistically non-novel. As used here, the "statistically non-novel/statistically novel" distinction is based purely on the physical characteristics of faces and not cognitive and semantic factors involved.

In Figure 16 we examine what happens when we reconstruct statistically novel faces that are radically different from those used to create the tensorfaces. Yoda (Figure 16ai) is a face we can instantly perceive without a period of training, yet it is unlikely from an evolutionary perspective that we would have developed face cells specifically

tuned to handle that stimulus. Presented here are reconstructions of Yoda using tensorface populations of different complexities that were created using human faces. All the tensorface reconstructions are of poor quality, as nothing resembling this stimulus was part of the face sample set used to create the tensorfaces. However, subjectively, it looks as if the low-complexity reconstruction is better than the high-complexity one. The high-complexity reconstruction appears to be overconstrained to resemble the faces in the training set.

Measuring reconstruction error, we can see the reconstruction error of Yoda does indeed get worse as tensorface complexity increases (Figure 16aii, dashed line). That trend is the opposite of what we saw for the reconstruction of the test face set, where reconstruction error decreased with greater tensorface complexity (Figure 16aii, solid line, repeated from Figure 11b). Figure 16bi show the reconstruction of another statistically novel face far beyond the bounds of what was included in the sample face set, the face of a chimpanzee. As with Yoda, we see in Figure 16bii that reconstruction error increases with tensorface complexity, the opposite of what occurs when reconstructing the test face set.

Although high-complexity tensorfaces produce the best reconstructions of faces that are statistically non-novel, they have a reduced ability to generalize to faces in statistically novel faces. For statistically novel faces, the low-complexity tensorfaces produce the best reconstructions. The lower ability to generalize as tensorface complexity increases cannot be explained by changes in the selectivity of tensorfaces. We see in Figure 14a that tensorface selectivity remains constant (and low) regardless of tensorface complexity.

Representation of the average face

There is evidence indicating that the representation of the average face forms the origin of a high-dimensional face space (Leopold et al., 2001; Rhodes & Jeffery, 2006; Tsao & Freiwald, 2006; Wilson et al., 2002). Using multidimensional scaling (MDS) based on a Euclidean distance metric, we examined the location of the average face in a face space formed by 100 tensorfaces. This set of faces thus formed 129 points (128 sample faces plus the average face) in a 100-dimensional face space. The MDS analysis are based on tensorfaces with rank=8, but other rank values performed similarly.

The result of MDS analysis is given in Figure 17. It shows that the faces of different racial groups and genders cluster into different regions of face space. Furthermore, we see that the face space formed by the tensorfaces places the representation of the average face at the origin of the face space. Note that the representation of the average face at the origin is due to activity across a population of tensorfaces. There is no individual tensorface that specifically represents the average face.

Cross-stimulation of tensorfaces by non-face stimuli

A complete and autonomous face processing system should reject non-face inputs. Therefore, now we look at reconstruction of a non-face object by tensorfaces. Reconstruction of a melon is shown in Figure 18a. The reconstruction obviously fails, producing an enormous reconstruction error (Figure 18b). Going beyond just the magnitude of the error, the organization of the errors gives the reconstructed melon the shape of a face, although with melon texture on it.

Representing faces is essentially the only thing that tensorfaces are capable of.

Any object presented to that face representation system will be interpreted as a face. In contrast, the ideal representation of a non-face object produced by a specialized face representation system should be null, no response.

The spurious reconstruction of non-face objects as faces by the tensorface population occurs because response magnitudes of tensorfaces are similar to face and non-face objects (Figure 18c). This cross-stimulation of tensorfaces by face and non-face stimuli appears to be the result of the low stimulus selectivity of tensorfaces seen in Figure 14a. As will be discussed further below, a possible solution to this cross-stimulation problem would be to have a nonlinearity associated with the linear tensorface receptive fields that would filter out non-face stimuli from being processed (Figure 18d).

Discussion

Based on measures of algorithmic information (Kolmogorov complexity) we show here that low-complexity and high-complexity faces have different properties, and therefore that complexity can be a way of constraining possible ways that face space is organized. Just as Shannon information has proven useful for understanding processing in early vision (Barlow, 1961; Field, 1994), we suggest that Kolmogorov complexity and related measures such as logical depth may prove useful in providing a framework for studying high-level vision, including face recognition and object recognition in general.

Cover and Thomas (2006), in their textbook on information theory, state that "we consider Kolmogorov complexity to be more fundamental than Shannon entropy". Kolmogorov complexity is associated with concepts from computational theory (Turing machines), while Shannon entropy is a statistical theory not derived from computational theory. Both Shannon entropy and Kolmogorov complexity can be used as measures of

efficient coding, indicating how compressed a representation can be. In addition to considering compression from the statistical perspective of Shannon entropy (Barlow, 1961; Field, 1994; Olshausen & Field, 2004) it can also be considered from the algorithmic perspective of Kolmogorov complexity (Adriaans, 2007; Chater & Vitányi, 2003; Feldman, 2016). A critical difference between the two types of information is that Shannon entropy is defined probabilistically in terms of a distribution over an ensemble of symbols, without any connection to the structure of individual messages, while Kolmogorov complexity is a deterministic concept measuring information of a single entity (message) by itself in isolation (Grünwald & Vitányi, 2008a). While assigning probabilities to repetitive low-level structures (for example, V1 Gabor receptive fields) is clearly reasonable within the framework of Shannon entropy, assigning such probabilities to high-level structures that are essentially unique (for example, inferotemporal receptive fields) may be problematic (Chater & Vitányi, 2003). On the other hand, as a non-probabilistic computation, Kolmogorov complexity can assign a measure of information content to an individual high-level structure purely in terms of its internal structure.

Receptive field complexity appears to increase as one ascends through the hierarchy of visual cortical areas. Although this impression is not yet confirmed through neurophysiological measurements of Kolmogorov complexity, the ventral stream deep learning model of Güçlü and van Gerven (2015) reinforces this perception of increased complexity, as it shows a monotonic increase in Kolmogorov complexity as a function of the layer in the network. While Kolmogorov complexity has not been measured experimentally, sparseness has been. It is well established that visual representations are sparse (Dan, Atick, & Reid, 1996; Lehky et al., 2011; Lehky et al., 2005; Lehky &

Sereno, 2007; Pitkow & Meister, 2012; Rolls & Tovée, 1995; Vinje & Gallant, 2000; Willmore, Mazer, & Gallant, 2011). What is not well established is the gradient of how sparseness changes across different cortical areas, as such data is limited. Kolmogorov complexity and sparseness need not necessarily be correlated. Just because receptive field organization may appear highly complex does mean that there must be a correspondingly high level of sparseness (Figure 14). Indeed, a comparison of a low-level visual area (V1) (Lehky et al., 2005) and a high-level visual area (anterior inferotemporal cortex) (Lehky et al., 2011) shows only a very modest increase in sparseness (median kurtosis going from 0.84 to 1.88, measuring lifetime sparseness or what we call cell selectivity). The data of Willmore et al. (2011) indicates sparseness stays essentially the same going from V1 to V4. As Willmore et al. (2011) conclude, the data suggests that "maximization of lifetime sparseness is not a principle that determines the organization of visual cortex". In contrast, there appears to be a steadily and substantially increase in receptive field complexity across the cortical areas of the ventral visual hierarchy. In view of that, Kolmogorov complexity may be a more interesting parameter in high-level visual processing than sparseness.

Given these preliminary remarks on the general significance of using Kolmogorov complexity for characterizing visual receptive fields, we now turn to face processing specifically. The different complexities of tensorfaces we examine here demonstrate a range of possibilities that biological face cells could have. In particular, low and medium complexity face cells form feasible representations in addition to very high complexity representations, such as those formed by PCA eigenfaces or variants thereof (e.g., active appearance models, Chang and Tsao (2017)). Such high-complexity face representations

have in the past been suggested as forming the basis of biological face space. The actual complexity of biological face cells remains a question for future experimental studies.

We observe a trade-off between receptive field complexity and the population size necessary to reach a criterion error in reconstructing faces (Figure 11a). A large population of low-complexity tensorfaces is equivalent to a smaller population of high complexity tensorfaces. This trade-off can be observed for receptive fields in earlier cortical areas. For example, large populations of low-complexity Gabor functions in striate cortex can also accurately represent faces, and face identification can be performed using Gabor-based face representations without any face cells (e.g., Wiskott, Krüger, Kuiger, & von der Malsburg, 1997).

From the perspective of information contained in a population of tensorfaces as indicated by reconstruction error, there doesn't seem to be a benefit to using high-complexity face cells. Reconstruction error as a function of tensorface complexity does not decrease moving from medium- to high complexity tensorfaces (Figure 11b). Moreover, high-complexity face cells incur high computational costs to create, measured as Kolmogorov complexity or logical depth (Figure 12). On the other hand, low-complexity cells are inefficient in that they require larger population sizes to reach a criterion reconstruction error. The "sweet spot" for face representations may be at intermediate complexity, perhaps at about rank=8. Nevertheless, low-complexity face cells may balance their representational inefficiency with their increased ability to generalize to statistically novel faces (Figure 16). Thus there may be an advantage to having a mixture of low to medium complexity face cells, but not high-complexity face cells such as produced by PCA. Furthermore, not all face cells in a population need to

have the same level of complexity. That is another empirical question for future experimental work.

What might be the advantage of increased complexity of face representations in higher visual cortical areas (i.e., the creation of face cells)? The smaller population sizes allowed by more complex receptive fields means that face spaces with lower dimensionalities can be created (see Lehky, Kiani, Esteky, and Tanaka (2014) for a discussion of dimensionality). In other words, creating face representations with more complex receptive fields may be a dimensionality reduction technique. Lower-dimensional face spaces may make it easier to categorize faces (Plastria, De Bruyne, & Carrizosa, 2008). However, the benefits for creating more efficient face spaces using more complex receptive fields must be balanced with computational costs of the increased complexity as measured by Kolmogorov complexity and logical depth of receptive field spatial structure.

There is a high correlation between computational complexity (Figure 12) and spectral power at high spatial frequencies (Figure 13b). The link between computational complexity and spatial frequency provides additional motivation to characterize spatial frequency properties of face cells, expanding upon current physiological (Inagaki & Fujita, 2011; Rolls, Baylis, & Hasselmo, 1987) and psychophysical studies (Costen, Parker, & Craw, 1996; Gaspar, Sekuler, & Bennett, 2008; Näsänen, 1999). Nevertheless, tensorfaces with different complexities are not simply Fourier amplitude filtered versions of each other but have substantial differences in appearance (phase spectra). The spatial frequency content of a facial representation is not sufficient to completely characterize its complexity.

We worked with colored faces rather than the monochromatic faces as used in most studies of face coding. Color can be an important aspect of face identification (Nestor, Plaut, & Behrmann, 2013; J. Tanaka, Weiskopf, & Williams, 2001), particularly if the shape information is degraded or ambiguous (Choi, Ro, & Plataniotis, 2009; Yip & Sinha, 2002). We include joint shape/color sensitivity in the tensorfaces developed here. Responsiveness to both shape and color are found in the same face cells in the inferotemporal cortex of monkeys as measured neurophysiologically (R. Edwards, Xiao, Keysers, Földiák, & Perrett, 2003). However, there is also fMRI evidence for separate, parallel channels coding face shape and color (Lafer-Sousa & Conway, 2013; Lafer-Sousa, Conway, & Kanwisher, 2016).

A significant question is whether the representation of faces is global (holistic) or local (parts-based) (Behrmann, Richler, Avidan, & Kimchi, 2015; Maurer, Grand, & Mondloch, 2002; Piepers & Robbins, 2012; Richler, Palmeri, & Gauthier, 2012; J. Tanaka & Simonyi, 2016). We examined this issue by measuring a globality index for tensorfaces, defined as the average fraction of the face covered by the tensorfaces. Tensorfaces across a population exhibit a great deal of variability in their globality. Some tensorfaces are local and others are strongly global. On average, high complexity tensorfaces are more global than low complexity ones (Figure 15). Typically, a tensorface covers a sizeable fraction of a face, but not the entire face.

This variability in globality is consistent with both psychophysical (J. Tanaka & Simonyi, 2016) and neurophysiological reports (Freiwald et al., 2009), which conclude that face processing involves both global and parts-based processing. We have previously proposed such mixed and intermediate globality for inferotemporal object representations

in general, not just faces (Lehky & Tanaka, 2016), based on data from monkey neurophysiology showing sensitivity of neurons to a partial set of features but generally not the entire object (Fujita, Tanaka, Ito, & Cheng, 1992; Ito, Fujita, Tamura, & Tanaka, 1994; Ito, Tamura, Fujita, & Tanaka, 1995; Kobatake & Tanaka, 1994; K. Tanaka, Saito, Fukada, & Moriya, 1991; Yamane, Tsunoda, Matsumoto, Phillips, & Tanifuji, 2006).

Recently Chang and Tsao (2017) have reported that biological face space corresponds to one specific linear space that they have discovered. However, we believe that the linear face space they report is not uniquely defined under their mathematical data analyses. Rather, a variety of different face spaces are consistent with their data.

Approximate linear transforms (i.e. multiple linear regression) can be fit between face coefficients for various linear decompositions (PCA, ICA, NMF, our version of tensorfaces, etc.). Fits between the different linear face decompositions will be good provided they are each capable of doing acceptable reconstructions of faces (for example, under some psychophysical criterion for reconstruction error). If the neurophysiological data provides a good fit to face components from one linear decomposition, such as active appearance model (AAT) of Chang and Tsao (2017), then the data will also provide good fits to other linear face decompositions. Chang and Tsao (2017) have studied one pre-determined linear face decomposition, and since it happened to meet their criterion of goodness of fit, they did not continue to examine other possible decompositions.

For example, we investigated the transform between PCA coefficients ($PCAcoeff$) and tensor coefficients ($tensorCoeff$) for two linear face decompositions. This transform is given by $PCAcoeff' = tensorCoeff * b$, where $PCAcoeff$ and $tensorCoeff$ are

matrices of coefficients for a set of faces, one face per column, and $PCAcoeff'$ are estimated PCA coefficients. The coefficient b is given by $b = \text{pinv}\left(tensorCoeff\right) * PCAcoeff$, where pinv is the Moore-Penrose pseudoinverse operator performing a multiple linear regression, and $tensorcoeff$ has been augmented by a column of ones to include offsets. There were 128 faces as input, the tensor decomposition had 100 components, and we modeled the first 50 PCA components. We fit the model leaving one face left out for testing, repeating with a different face being left out. The results show that when comparing actual $PCAcoeff$ and estimated $PCAcoeff'$, the model accounts for a 0.985 fraction of the variance. This shows that it is possible to predict $PCAcoeff$ from $tensorCoeff$ with high accuracy**.** Therefore, the two linear face decompositions would each provide essentially the same fit to the neurophysiological data. The interchangeability between different linear face spaces means that if one wants to select a face model, it would have to be constrained based on some criterion other than overall goodness of fit of data to one single linear model in isolation, but perhaps based instead on comprehensive experimental characterizations of receptive fields of individual face cells across the population.

Furthermore, experimental face stimulus sets are limited in that they cover only a limited range of the faces that are possible. It is conceivable that observed face spaces such as Chang and Tsao (2017) are approximately linear at a local scale but that a more complete sampling of faces will reveal a nonlinear face space at a broader scale. Even color space at high-level visual cortex is a complicated, nonlinear space (Bohon, Hermann, Hansen, & Conway, 2016; Komatsu, Ideura, Kaji, & Yamane, 1992; Lehky &

Sejnowski, 1999), and there is no reason to expect that face space is not similarly complicated and nonlinear.

Underlying a possible nonlinear face space would be face cells themselves that act individually as spatially nonlinear filters. Reports of inferotemporal processing indicate that object representations, and in particular face representations, involve nonlinear spatial filters, as mentioned earlier (Owaki et al., 2018; K. Tanaka, 1996; Yamane et al., 2006). Those nonlinear spatial interactions are why we can't map face cell receptive fields with a simple stimulus spot as we do in striate cortex. The spurious reconstruction of non-face objects using linear components such as tensorfaces (Figure 18) and eigenfaces (Figure 2b in Tsao and Livingstone (2008)) also indicates a requirement to introduce some sort of nonlinearity in face cells.

Linear models of biological facial representations, including the particular implementation of tensorfaces used here, can reveal some significant aspects of face processing and thus can be useful in theoretical discussions, as long as the biological limitations of those models are kept firmly in sight. However, without nonlinearity they cannot be considered complete solutions. Nonlinearity is a central stumbling block in understanding biological face processing and object processing generally. One approach to introducing nonlinearity into face representations is illustrated by the nonlinear tensor modeling of Vasilescu and Terzopoulos (2011). However, there are a multitude of other possibilities, including the development of Kolmogorov-complexity constrained deep learning networks.

Overall, the results here suggest that spatial complexity of face cells is likely to be a significant factor, among others, in characterizing face space. Defining the complexity

of face representations may contribute to a more complete framework for guiding future research.

Acknowledgments

This research was supported by a grant to Dr. Keiji Tanaka from the Strategic Research Program for Brain Sciences of the Japan Agency for Medical Research and Development. It was also supported by grants to Dr. Andrzej Cichocki from the Ministry of Education and Science of the Russian Federation (grant 14.756.31.0001) and the Polish National Science Center (grant 2016/20/W/N24/00354). We thank Dr. Topi Tanskanen for comments on the manuscript.

References

Adriaans, P. (2007). *Learning as Data Compression.* Paper presented at the Third Conference on Computability in Europe: Computation and Logic in the Real World, Siena, Italy.

Adriaans, P. (2019). Information. In E. N. Zalta (Ed.), *The Stanford Encyclopedia of Philosophy (Spring 2019 Edition)*: https://plato.stanford.edu/archives/spr2019/entries/information/.

Bader, B. W., Kolda, T. G., & others. (2017). Matlab Tensor Toolbox (Version 3.0-dev): https://www.tensortoolbox.org.

Baldassi, C., Alemi-Neissi, A., Pagan, M., Dicarlo, J. J., Zecchina, R., & Zoccolan, D. (2013). Shape similarity, better than semantic membership, accounts for the structure of visual object representations in a population of monkey inferotemporal neurons. *PLoS Computational Biology, 9*, e1003167.

Barlow, H. B. (1961). Possible principles underlying the transformations of sensory messages. In W. Rosenblith (Ed.), *Sensory Communication* (pp. 217-234). Cambridge, MA: MIT Press.

Bartlett, M. S., Movellan, J. R., & Sejnowski, T. J. (2002). Face recognition by independent component analysis. *IEEE Transactions on Neural Networks, 13*, 1450 - 1464.

Bartlett, M. S., & Sejnowski, T. J. (1997). *Independent components of face images: A representation for face recognition.* Paper presented at the 4th Annual Joint Symposium on Neural Computation, Pasadena, CA.

Behrmann, M., Richler, J. J., Avidan, G., & Kimchi, R. (2015). Holistic Face Perception. In J. Wagemans (Ed.), *The Oxford Handbook of Perceptual Organization* (pp. 758-774). Oxford: Oxford University Press.

Bell, A. J., & Sejnowski, T. J. (1997). The "independent components" of natural scenes are edge filters. *Vision Research, 37*, 3327-3338.

Bennett, C. H. (1988). Logical depth and physical complexity. In R. Herken (Ed.), *The universal Turing machine– a half-century survey* (pp. 227-257). Oxford, England: Oxford University Press.

Bennett, C. H. (1994). Complexity in the universe. In J. J. Halliwell, J. Peres-Mercader & W. H. Zurek (Eds.), *Physical Origins of Time Asymmetry* (pp. 33-46). New York: Cambridge University Press.

Blanz, V., & Vetter, T. (1999). *A morphable model for the synthesis of 3D faces.* Paper presented at the Siggraph '99: Proceedings of the 26th annual conference on computer graphics and interactive techniques, Los Angeles.

Bohon, K. S., Hermann, K. L., Hansen, T., & Conway, B. R. (2016). Representation of Perceptual Color Space in Macaque Posterior Inferior Temporal Cortex (the V4 Complex). *eNeuro, 3*, ENEURO.0039-0016.2016.

Bracci, S., & Op de Beeck, H. (2016). Dissociations and Associations between Shape and Category Representations in the Two Visual Pathways. *Journal of Neuroscience, 36*, 432-444.

Bro, R. (1997). PARAFAC. Tutorial and applications. *Chemometrics and Intelligent Laboratory Systems 38*, 149-171.

Bro, R. (1998). *Multi-way Analysis in the Food Industry: Models, Algorithms, and Applications*. University of Amsterdam, Ph.D. thesis.

Bro, R., Harshman, R. A., Sidiropoulos, N. D., & Lundy, M. E. (2009). Modeling multi-way data with linearly dependent loadings. *Journal of Chemometrics, 23*, 324-340.

Carroll, J. D., & Chang, J. J. (1970). Analysis of individual differences in multidimensional scaling via an n-way generalization of Eckart–Young decomposition. *Psychometrika, 35*, 283-319.

Chaitin, G. (1969). On the length of programs for computing finite binary sequences: statistical considerations. *Journal of the Association of Computing Machinery, 16*, 145-159.

Chang, L., & Tsao, D. Y. (2017). The code for facial identity in the primate brain. *Cell, 169*, 1013-1028.

Chater, N., & Vitányi, P. (2003). Simplicity: a unifying principle in cognitive science? *Trends in Cognitive Sciences, 7*, 19-22.

Choi, J. Y., Ro, Y. M., & Plataniotis, K. N. (2009). Color face recognition for degraded face images. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics, 39*, 1217-1230.

Cichocki, A., Mandic, D., Phan, A.-H., Caiafa, C., Zhou, G., Zhao, Q., et al. (2015). Tensor Decompositions for Signal Processing Applications: From two-way to multiway component analysis. *IEEE Signal Processing Magazine, 32*, 145-163.

Cichocki, A., Zdunek, R., Phan, A.-H., & Amari, S.-I. (2009). *Nonnegative matrix and tensor factorizations: Applications to exploratory multi-way data analysis and blind source separation*. Chichester: Wiley.

Connolly, A. C., Guntupalli, J. S., Gors, J., Hanke, M., Halchenko, Y. O., Wu, Y. C., et al. (2012). The representation of biological classes in the human brain. *Journal of Neuroscience, 32*, 2608-2618.

Cootes, T. F., Edwards, G. J., & Taylor, C. J. (2001). Active Appearance Models. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 23*, 681-685.

Costen, N. P., Parker, D. M., & Craw, I. (1996). Effects of high-pass and low-pass spatial filtering on face identification. *Perception and Psychophysics, 58*, 602-612.

Cover, T. M., & Thomas, J. A. (2006). *Elements of Information Theory, Second Edition*. Hoboken, NJ, USA: John Wiley & Sons.

Cowell, R. A., & Cottrell, G. W. (2013). What evidence supports special processing for faces? A cautionary tale for fMRI interpretation. *Journal of Cognitive Neuroscience, 25*, 1777-1193.

Dan, Y., Atick, J. J., & Reid, R. C. (1996). Efficient coding of natural scenes in the lateral geniculate nucleus: experimental test of a computational theory. *Journal of Neuroscience, 16*, 3351-3362.

De Lathauwer, L. (2008a). Decompositions of a higher-order tensor in block terms – Part I: Lemmas for partitioned matrices. *SIAM Journal on Matrix Analysis and Applications, 30*, 1022-1032.

De Lathauwer, L. (2008b). Decompositions of a higher-order tensor in block terms – Part II: Definitions and uniqueness. *SIAM Journal on Matrix Analysis and Applications, 30*, 1033-1066.

Duchaine, B., & Yovel, G. (2015). A Revised Neural Framework for Face Processing. *Annual Review of Vision Science, 1*, 393-416.

Edwards, G. J., Cootes, T. F., & Taylor, C. J. (1998). *Face recognition using active appearance models.* Paper presented at the 5th European Conference on Computer Vision, Freiburg, Germany.

Edwards, R., Xiao, D., Keysers, C., Földiák, P., & Perrett, D. (2003). Color sensitivity of cells responsive to complex stimuli in the temporal cortex. *Journal of Neurophysiology, 90*, 1245-1256.

Eifuku, S., De Souza, W. C., Tamura, R., Nishijo, H., & Ono, T. (2004). Neuronal correlates of face identification in the monkey anterior temporal cortical areas. *Journal of Neurophysiology, 91*, 358-371.

Fang, F., Murray, S. O., & He, S. (2007). Duration-dependent fMRI adaptation and distributed viewer-centered face representation in human visual cortex. *Cerebral Cortex, 17*, 1402-1411.

Favier, G. r., & de Almeida, A. L. (2014). Overview of constrained PARAFAC models. *EURASIP Journal on Advances in Signal Processing, 2014*, 142.

Feldman, J. (2016). The simplicity principle in perception and cognition. *Wiley Interdisciplinary Review:Cognitive Science, 7*, 330-340.

Field, D. J. (1994). What is the goal of sensory coding? *Neural Computation, 6*, 559-601.

Freiwald, W. A., Duchaine, B., & Yovel, G. (2016). Face processing systems: from neurons to real-world social perception. *Annual Review of Neuroscience, 39*, 325-346.

Freiwald, W. A., & Tsao, D. Y. (2010). Functional compartmentalization and viewpoint generalization within the macaque face-processing system. *Science, 330*, 845-851.

Freiwald, W. A., Tsao, D. Y., & Livingstone, M. (2009). A face feature space in the macaque temporal lobe. *Nature Neuroscience, 12*, 1187-1196.

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

Gaspar, C., Sekuler, A. B., & Bennett, P. J. (2008). Spatial frequency tuning of upright and inverted face identification. *Vision Research, 48*, 2817-2826.

Gauthier, I., Behrmann, M., & Tarr, M. J. (1999). Can face recognition really be dissociated from object recognition? *Journal of Cognitive Neuroscience, 11*, 349-370.

Gauthier, I., Skudlarski, P., Gore, J. C., & Anderson, A. W. (2000). Expertise for cars and birds recruits brain areas involved in face recognition. *Nature Neuroscience, 3*, 191-197.

Gauthier, I., & Tarr, M. J. (1997). Becoming a "Greeble" expert: exploring mechanisms for face recognition. *Vision Research, 37*, 1673-1682.

Grünwald, P., & Vitányi, P. (2008a). Algorithmic information theory. https://arxiv.org/abs/0809.2754

Grünwald, P., & Vitányi, P. (2008b). Shannon information and Kolmogorov complexity. https://arxiv.org/abs/cs/0410002

Güçlü, U., & van Gerven, M. A. (2015). Deep neural networks reveal a gradient in the complexity of neural representations across the ventral stream. *Journal of Neuroscience, 35*, 10005-10014.

Guo, X., Miron, S., Brie, D., & Stegeman, A. (2012). Unimode and partial uniqueness conditions for candecomp/parafac of three-way arrays with linearly dependent loadings. *SIAM Journal on Matrix Analysis and Applications,, 33*, 111-129.

Harshman, R. A. (1970). Foundations of the PARAFAC procedure: Models and conditions for an explanatory multimodal factor analysis. *UCLA Working Papers in Phonetics, 16*, 1-84.

Haxby, J. V., Hoffman, E. A., & Gobbini, M. I. (2000). The distributed human neural system for face perception. *Trends in Cognitive Sciences, 4*, 223-233.

Hout, M. C., Papesh, M. H., & Goldinger, S. D. (2013). Multidimensional scaling. *Wiley Interdisciplinary Review:Cognitive Science, 4*, 93-103.

Inagaki, M., & Fujita, I. (2011). Reference frames for spatial frequency in face representation differ in the temporal visual cortex and amygdala. *Journal of Neuroscience, 31*, 10371-10379.

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, 14*, 499-508.

Ito, M., Tamura, H., Fujita, I., & Tanaka, K. (1995). Size and position invariance of neuronal responses in monkey inferotemporal cortex. *Journal of Neurophysiology, 73*, 218-226.

Jiang, F., Blanz, V., & O'Toole, A. J. (2006). Probing the visual representation of faces with adaptation: A view from the other side of the mean. *Psychological Science, 17*, 493-500.

Kanwisher, N. (2000). Domain specificity in face perception. *Nature Neuroscience, 3*, 759-763.

Kanwisher, N., & Yovel, G. (2006). The fusiform face area: a cortical region specialized for the perception of faces. *Philosophical Transactions of the Royal Society of London B Biological Sciences, 361*, 2109-2128.

Kayaert, G., Biederman, I., & Vogels, R. (2005). Representation of regular and irregular shapes in macaque inferotemporal cortex. *Cerebral Cortex, 15*, 1308-1321.

Kiani, R., Esteky, H., Mirpour, K., & Tanaka, K. (2007). Object category structure in response patterns of neuronal population in monkey inferior temporal cortex. *Journal of Neurophysiology, 97*, 4296-4309.

Kobatake, E., & Tanaka, K. (1994). Neuronal selectivities to complex object features in the ventral visual pathway of the macaque cerebral cortex. *Journal of Neurophysiology, 71*, 856-867.

Kolda, T. G., & Bader, B. W. (2009). Tensor Decompositions and Applications. *SIAM Review, 51*, 455-500.

Kolmogorov, A. N. (1965). Three Approaches to the Quantitative Definition of Information. *Problems of Information Transmission, 1*, 1-7.

Komatsu, H., Ideura, Y., Kaji, S., & Yamane, S. (1992). Color selectivity of neurons in the inferior temporal cortex of the awake macaque monkey. *Journal of Neuroscience, 12*, 408-424.

Kravitz, D. J., Peng, C. S., & Baker, C. I. (2011). Real-world scene representations in high-level visual cortex: it's the spaces more than the places. *Journal of Neuroscience, 31*, 7322-7333.

Kriegeskorte, N., Mur, M., Ruff, D. A., Kiani, R., Bodurka, J., Esteky, H., et al. (2008). Matching categorical object representations in inferior temporal cortex of man and monkey. *Neuron, 60*, 1126-1141.

Lafer-Sousa, R., & Conway, B. R. (2013). Parallel, multi-stage processing of colors, faces and shapes in macaque inferior temporal cortex. *Nature Neuroscience, 16*, 1870-1878.

Lafer-Sousa, R., Conway, B. R., & Kanwisher, N. (2016). Color-Biased Regions of the Ventral Visual Pathway Lie between Face- and Place-Selective Regions in Humans, as in Macaques. *Journal of Neuroscience, 36*, 1682-1697.

Lantéri, H., Soummer, R., & Aime, C. (1999). Comparison between ISRA and RLA algorithms: Use of a Wiener filter based stopping criterion. *Astronomy and Astrophysics Supplementary Series, 140*, 235-246.

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

Lehky, S. R., Kiani, R., Esteky, H., & Tanaka, K. (2011). Statistics of visual responses in primate inferotemporal cortex to object stimuli. *Journal of Neurophysiology, 106*, 1097-1117.

Lehky, S. R., Kiani, R., Esteky, H., & Tanaka, K. (2014). Dimensionality of object representations in monkey inferotemporal cortex. *Neural Computation, 26*, 2135-2162.

Lehky, S. R., & Sejnowski, T. J. (1999). Seeing white: Qualia in the context of decoding population codes. *Neural Computation, 11*, 1261-1280.

Lehky, S. R., Sejnowski, T. J., & Desimone, R. (2005). Selectivity and sparseness in the responses of striate complex cells. *Vision Research, 45*, 57-73.

Lehky, S. R., & Sereno, A. B. (2007). Comparison of shape encoding in primate dorsal and ventral visual pathways. *Journal of Neurophysiology, 97*, 307-319.

Lehky, S. R., & Tanaka, K. (2016). Neural representation for object recognition in inferotemporal cortex. *Current Opinion in Neurobiology, 37*, 23-35.

Lempel, A., & Ziv, J. (1976). On the complexity of finite sequences. *IEEE Transactions on Information Theory, 22*, 75-81.

Leopold, D. A., Bondar, I. V., & Giese, M. A. (2006). Norm-based face encoding by single neurons in the monkey inferotemporal cortex. *Nature, 442*, 572-575.

Leopold, D. A., O'Toole, A. J., Vetter, T., & Blanz, V. (2001). Prototype-referenced shape encoding revealed by high-level aftereffects. *Nature Neuroscience, 4*, 89-94.

Leopold, D. A., & Rhodes, G. (2010). A comparative view of face perception. *Journal of Comparative Psychology, 124*, 233-251.

Li, M., & Vitányi, P. (2008). *An Introduction to Kolmogorov Complexity and Its Applications* (3rd ed.). New York: Springer.

Lin, C. J. (2007). Projected gradient methods for non-negative matrix factorization. *Neural Computation, 19*, 2756–2779.

Liu, S., & Trenkler, G. (2008). Hadamard, Khatri-Rao, Kronecker and other matrix products. *International Journal of Information and Systems Sciences, 4*, 160-177.

Maurer, D., Grand, R. L., & Mondloch, C. J. (2002). The many faces of configural processing. *Trends in Cognitive Sciences, 6*, 255-260.

McKone, E., Kanwisher, N., & Duchaine, B. C. (2007). Can generic expertise explain special processing for faces? *Trends in Cognitive Sciences, 11*, 8-15.

Meytlis, M., & Sirovich, L. (2007). On the dimensionality of face space. *IEEE Transactions on Pattern Analysis and Machine Intelligence, 29*, 1262-1267.

Murata, A., Gallese, V., Luppino, G., Kaseda, M., & Sakata, H. (2000). Selectivity for the shape, size, and orientation of objects for grasping in neurons of monkey parietal area AIP. *Journal of Neurophysiology, 83*, 2580-2601.

Näsänen, R. (1999). Spatial frequency bandwidth used in the recognition of facial images. *Vision Research, 39*, 3824-3833.

Natu, V. S., Jiang, F., Narvekar, A., Keshvari, S., Blanz, V., & O'Toole, A. J. (2010). Dissociable neural patterns of facial identity across changes in viewpoint. *Journal of Cognitive Neuroscience, 22*, 1570-1582.

Nestor, A., Plaut, D. C., & Behrmann, M. (2013). Face-space architectures: evidence for the use of independent color-based features. *Psychological Science, 24*, 1294-1300.

Nestor, A., Plaut, D. C., & Behrmann, M. (2016). Feature-based face representations and image reconstruction from behavioral and neural data. *Proceedings of the National Academy of Sciences of the United States of America, 113*, 416-421.

Noudoost, B., & Esteky, H. (2013). Neuronal correlates of view representation revealed by face-view aftereffect. *Journal of Neuroscience, 33*, 5761-5772.

Olshausen, B. A., & Field, D. J. (1996). Emergence of simple-cell receptive field properties by learning a sparse code for natural images. *Nature, 381*, 607-609.

Olshausen, B. A., & Field, D. J. (2004). Sparse coding of sensory inputs. *Current Opinion in Neurobiology, 14*, 481-487.

Op de Beeck, H., Wagemans, J., & Vogels, R. (2001). Inferotemporal neurons represent low-dimensional configurations of parameterized shapes. *Nature Neuroscience, 4*, 1244-1252.

Otsu, N. (1979). A threshold selection method from gray-level histograms. *IEEE Transactions on Systems, Man and Cybernetics, 9*, 62-66.

Owaki, T., Vidal-Naquet, M., Nam, Y., Uchida, G., Sato, T., Câteau, H., et al. (2018). Searching for visual features that explain response variance of face neurons in inferior temporal cortex. *PLoS ONE, 13*, e0201192.

Parr, L. A. (2011). The evolution of face processing in primates. *Philosophical Transactions of the Royal Society of London B Biological Sciences, 366*, 1764-1777.

Parr, L. A., Hecht, E., Barks, S. K., Preuss, T. M., & Votaw, J. R. (2009). Face processing in the chimpanzee brain. *Current Biology, 19*, 50-53.

Parr, L. A., Winslow, J. T., Hopkins, W. D., & de Waal, F. B. (2000). Recognizing facial cues: individual discrimination by chimpanzees (Pan troglodytes) and rhesus monkeys (Macaca mulatta). *Journal of Comparative Psychology, 114*, 47-60.

Perrett, D. I., Oram, M. W., Harries, M. H., Bevan, R., Hietanen, J. K., Benson, P. J., et al. (1991). Viewer-centred and object-centred coding of heads in the macaque temporal cortex. *Experimental Brain Research, 86*, 159-173.

Perrett, D. I., Smith, P. A., Potter, D. D., Mistlin, A. J., Head, A. S., Milner, A. D., et al. (1985). Visual cells in the temporal cortex sensitive to face view and gaze direction. *Proceedings of the Royal Society of London. Series B: Biological Sciences, 223*, 293-317.

Phan, A.-H. (2018). Matlab TensorBox (Version 2018.08): https://faculty.skoltech.ru/people/anhhuyphan - tabs-4.

Phan, A.-H., Cichocki, A., Tichavský, P., Zdunek, R., & Lehky, S. R. (2013). *From basis components to complex structural patterns.* Paper presented at the 38th IEEE International Conference on Acoustics, Speech, and Signal Processing.

Piepers, D. W., & Robbins, R. A. (2012). A review and clarification of the terms "holistic," "configural," and "relational" in the face perception literature. *Frontiers in Psychology, 3*, 559.

Pitkow, X., & Meister, M. (2012). Decorrelation and efficient coding by retinal ganglion cells. *Nature Neuroscience, 15*, 628-635.

Plastria, F., De Bruyne, S., & Carrizosa, E. (2008). *Dimensionality Reduction for Classification.* Paper presented at the Advanced Data Mining and Applications. Lecture Notes in Computer Science Vol 5139.

Rabanser, S., Shchur, O., & Günnemann, O. (2017). Introduction to Tensor Decompositions and their Applications in Machine Learning. https://arxiv.org/abs/1711.10781

Ramírez, F. M., Cichy, R. M., Allefeld, C., & Haynes, J. D. (2014). The neural code for face orientation in the human fusiform face area. *Journal of Neuroscience, 34*, 12155-12167.

Rhodes, G., & Jeffery, L. (2006). Adaptive norm-based coding of facial identity. *Vision Research, 46*, 2977-2987.

Richler, J. J., Palmeri, T. J., & Gauthier, I. (2012). Meanings, mechanisms, and measures of holistic processing. *Frontiers in Psychology, 3*, 553.

Rolls, E. T., Baylis, G. C., & Hasselmo, M. E. (1987). The responses of neurons in the cortex in the superior temporal sulcus of the monkey to band-pass spatial frequency filtered faces. *Vision Research, 27*, 311-326.

Rolls, E. T., & Tovée, M. J. (1995). Sparseness of the neuronal representation of stimuli in the primate temporal visual cortex. *Journal of Neurophysiology, 73*, 713-726.

Romero, M. C., Van Dromme, I. C., & Janssen, P. (2013). The role of binocular disparity in stereoscopic images of objects in the macaque anterior intraparietal area. *PLoS ONE, 8*, e55340.

Ruffini, G. (2017). Lempel-Zif Complexity Reference. https://arxiv.org/abs/1707.09848

Sereno, A. B., & Lehky, S. R. (2018). Attention Effects on Neural Population Representations for Shape and Location Are Stronger in the Ventral than Dorsal Stream. *eNeuro, 5*, e0371-0317.2018.

Sereno, A. B., Sereno, M. E., & Lehky, S. R. (2014). Recovering stimulus locations using populations of eye-position modulated neurons in dorsal and ventral visual streams of non-human primates. *Frontiers in Integrative Neuroscience, 8*, 28.

Sidiropoulos, N. D., De Lathauwer, L., Fu, X., Huang, K., Papalexakis, E. E., & Faloutsos, C. (2017). Tensor Decomposition for Signal Processing and Machine Learning. *IEEE Transaction on Signal Processing, 65*, 3551-3582.

Simoncelli, E. P., & Olshausen, B. A. (2001). Natural image statistics and neural representation. *Annual Review of Neuroscience, 24*, 1193-1216.

Sirovich, L., & Meytlis, M. (2009). Symmetry, probability, and recognition in face space. *Proceedings of the National Academy of Sciences of the United States of America, 106*, 6895–6899.

Solomonoff, R. (1964). A formal theory of inductive inference. Part I. *Information and Control, 7*, 1-22.

Sorber, L., Van Barel, M., & De Lathauwer, L. (2013). Optimization-Based Algorithms for Tensor Decompositions: Canonical Polyadic Decomposition, Decomposition in Rank-$(L_r,L_r,1)$ Terms, and a New Generalization. *SIAM Journal on Optimization, 23*, 695–720.

Stegeman, A., & Lam, T. (2012). Improved uniqueness conditions for canonical tensor decompositions with linearly dependent loadings. *SIAM Journal on Matrix Analysis and Applications,, 33*, 1250–1271.

Tanaka, J., & Simonyi, D. (2016). The "parts and wholes" of face recognition: A review of the literature. *Quarterly Journal of Experimental Psychology, 69*, 1876-1889.

Tanaka, J., Weiskopf, D., & Williams, P. (2001). The role of color in high-level vision. *Trends in Cognitive Sciences, 5*, 211-215.

Tanaka, K. (1996). Inferotemporal cortex and object vision. *Annual Review of Neuroscience, 19*, 109-139.

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.

Tolhurst, D. J., Smyth, D., & Thompson, I. D. (2009). The sparseness of neuronal responses in ferret primary visual cortex. *Journal of Neuroscience, 29*, 2355-2370.

Tong, M. H., Joyce, C. A., & Cottrell, G. W. (2008). Why is the fusiform face area recruited for novel categories of expertise? A neurocomputational investigation. *Brain Research, 1202*, 14-24.

Tsao, D. Y. (2014). The macaque face patch system: A window into object representation. *Cold Spring Harbor Symposia on Quantitative Biology, 79*, 109-114.

Tsao, D. Y., & Freiwald, W. A. (2006). What's so special about the average face? *Trends in Cognitive Sciences, 10*, 391-393.

Tsao, D. Y., & Livingstone, M. S. (2008). Mechanisms of face perception. *Annual Review of Neuroscience, 31*, 411-437.

Tucker, L. R. (1966). Some mathematical notes on three-mode factor analysis. *Psychometrica, 31*, 279-311.

Turk, M., & Pentland, A. (1991). Eigenfaces for recognition. *Journal of Cognitive Neuroscience, 3*, 71-86.

Van Loan, C. F. (2000). The ubiquitous Kronecker product. *Journal of Computational and Applied Mathematics, 123*, 85-100.

Vasilescu, M. A., & Terzopoulos, D. (2002). *Multilinear analysis of image ensembles: TensorFaces.* Paper presented at the European Conference on Computer Vision, Copenhagen, Denmark.

Vasilescu, M. A., & Terzopoulos, D. (2003). *Multilinear Subspace Analysis of Image Ensembles.* Paper presented at the IEEE Conference on Computer Vision and Pattern Recognition, Madison, WI.

Vasilescu, M. A., & Terzopoulos, D. (2005). *Multilinear Independent Components Analysis.* Paper presented at the IEEE Conference on Computer Vision and Pattern Recognition, San Diego, CA.

Vasilescu, M. A., & Terzopoulos, D. (2011). *Multilinear Projection for Face Recognition via Canonical Decomposition.* Paper presented at the IEEE Conference on Automatic Face and Gesture Recognition, Santa Barbara, CA.

Vinje, W. E., & Gallant, J. L. (2000). Sparse coding and decorrelation in primary visual cortex during natural vision. *Science, 287*, 1273-1276.

Wang, P., Gauthier, I., & Cottrell, G. (2016). Are Face and Object Recognition Independent? A Neurocomputational Modeling Exploration. *Journal of Cognitive Neuroscience, 28*, 558-574.

Wang, Y., Jia, Y., Hu, C., & Turk, M. (2005). Non-negative matrix factorization framework for face recognition. *International Journal of Pattern Recognition and Artificial Intelligence, 19*, 495-511.

Willmore, B., Mazer, J. A., & Gallant, J. L. (2011). Sparse coding in striate and extrastriate visual cortex. *Journal of Neurophysiology, 105*, 2907-2919.

Willmore, B., & Tolhurst, D. J. (2001). Characterizing the sparseness of neural codes. *Network, 12*, 255-270.

Wilson, H. R., Loffler, G., & Wilkinson, F. (2002). Synthetic faces, face cubes, and the geometry of face space. *Vision Research, 42*, 2909-2923.

Wiskott, L., Krüger, N., Kuiger, N., & von der Malsburg, C. (1997). Face recognition by elastic bunch graph matching. *IEEE Transactions on Pattern Analysis and Machine Intelligence 19*, 775-779.

Yamane, Y., Tsunoda, K., Matsumoto, M., Phillips, A. N., & Tanifuji, M. (2006). Representation of the spatial relationship among object parts by neurons in macaque inferotemporal cortex. *Journal of Neurophysiology, 96*, 3147-3156.

Yip, A. W., & Sinha, P. (2002). Contribution of color to face recognition. *Perception, 31*, 995-1003.

Young, M. P., & Yamane, S. (1992). Sparse population coding of faces in the inferotemporal cortex. *Science, 256*, 1327-1331.

Yovel, G., & Kanwisher, N. (2004). Face perception: domain specific, not process specific. *Neuron, 44*, 889-998.

Yu, J. (2016). *Rank-constrained PCA for intrinsic images decomposition.* Paper presented at the IEEE Conference on Image Processing, Phoenix, AZ.

Zenil, H., Delahaye, J.-P., & Gaucherel, C. (2012). Image Characterization and Classification by Physical Complexity. *Complexity, 17*, 26-42.

Figure captions

Figure 1. Examples of different classes of faces included in the face sets.

Figure 2. Face sets used to examine the tensor decomposition algorithm. **a.** Sample set. Shows 64 out of 128 faces serving as input to the algorithm to create the tensorfaces. **b.** Test set. Contains a different set of faces to evaluate properties of the tensorfaces.

Figure 3. Comparison between matrix methods used in PCA and tensor methods. **a.** Matrix methods can only operate on 2D data. That requires faces to be unfolded into 1D vectors before being placed as columns in a 2D matrix. **b.** Tensor methods allow data structures with an indefinite number of dimensions. That means faces do not need to be vectorized, but can be stacked on top of each other retaining their 2D organization. Here we used a 4D data structure for faces, including two spatial dimensions, a color dimension, and a dimension representing faces of different individuals.

Figure 4. Illustration of the tensor decomposition equations. Order-3 tensors $\mathcal{Y} \in \mathbb{R}^{I_1 \times I_2 \times I_3}$ are shown here as examples. **a.** Block term decomposition (BTD) for P terms of Kronecker tensor products of $\mathcal{A}_p$ (weights) and $\mathcal{X}_p$ (tensorface patterns) (Eq. 7), where $\otimes$ is the Kronecker product and P is the number of tensorfaces in the decomposition. In general the algorithm allows tensor size for each term P to be set individually, as shown in the diagram, but in practice they were all set the same size. **b.** Rank-constrained BTD decomposition illustrated for a single term. $\mathcal{A}_p$ and $\mathcal{X}_p$ can each be expressed as a set of matrices $\mathbf{U}^m$ and $\mathbf{V}^{(I)}$ (indicated by small rectangles) (Eqs. 8 and 9). Setting the number of columns for those matrices equal to the desired rank values, S_p

and R_p respectively imposes the rank constraints of the decomposition. Rank of the $\mathcal{X}_p$ decomposition determines tensorface complexity. Rank of the $\mathcal{A}_p$ decomposition was always set to 1 here.

Figure 5. Tensorfaces with low complexity (rank=2).

Figure 6. Tensorfaces with medium complexity (rank=8).

Figure 7. Tensorfaces with high complexity (rank=32).

Figure 8. Example tensorfaces with different levels of complexity.

Figure 9. Eigenfaces resulting from PCA decomposition of the sample face set. This shows the first 64 eigenfaces out of 128. PCA calculated after converting from RGB to L*A*B color space. When vectorizing the faces, the three color channels were concatenated to form one long 1D vector for each face. The average face was not subtracted prior to performing PCA, so the first eigenface here is the average face.

Figure 10. Reconstructions of the face test set (Figure 3b) using tensorfaces. **a.** Reconstruction using tensorfaces with low complexity (rank=2, Figure 5). **b.** Reconstruction using tensorfaces with medium complexity (rank=8, Figure 6). **c.** Reconstruction using tensorfaces with high complexity (rank=32, Figure 7).

Figure 11. Plots of reconstruction errors. **a.** Mean reconstruction error as a function of the number of tensorface components, holding tensorface complexity (rank) constant. Mean was calculated over 128 faces in sample set. **b.** Mean reconstruction error as a function of tensorface complexity (rank), holding the number of tensorface components

constant. Mean and standard error calculated over 128 faces in sample set.

Figure 12. Complexity measurements of tensorface images. **a.** Relation between tensorface rank and mean Kolmogorov complexity. **b.** Relation between tensorface rank and mean logical depth. Means were calculated from 100 tensorfaces. Shaded area shows standard deviation.

Figure 13. Spatial frequency power spectra of tensorfaces. **a.** Power spectra for tensorfaces for different rank values. Geometrical means of power spectra for 100 tensorfaces shown. **b.** Power at a high spatial frequency (100 cycles/image) as a function of tensorface rank. Geometrical means and geometrical standard deviations plotted.

Figure 14. Median cell selectivity and population sparseness of tensorfaces as a function of tensorface rank. Responses are from 100 tensorfaces stimulated by the 128 faces in the sample set. Cell selectivity and population sparseness both calculated as kurtosis of the probability distribution of responses. Cell selectivity refers to sparseness of single neurons calculated to a set of stimuli presented over time (also called lifetime sparseness). Population sparseness refers to population response to a single stimulus. Shaded area shows interquartile range. **a.** Cell selectivity. **b.** Population sparseness.

Figure 15. Globality of tensorface representations. **a.** Example local and global tensorfaces. **b.** Examples showing tensorface "high-activation" regions (enclosed by black lines) used to define the area covered by a tensorface. The globality of a tensorface is defined as the area of the tensorface divided by the average area of a face. **c.** Plot of mean globality as a function of tensorface rank. Mean is calculated over 100 tensorfaces. Shaded area shows standard deviation.

Figure 16. Reconstruction of statistically novel faces. These faces are radically different from sample set (Figure 3a) used to create the tensorfaces. **ai.** Reconstructing Yoda using tensorfaces with different complexities. **aii.** Relative reconstruction error for Yoda as a function of tensorface complexity (rank) (blue line), as well as relative reconstruction error for the test face set (red line). The red line is duplicated from Figure 11b. Reconstruction error decreases as a function of tensorface complexity for familiar faces (red) but increases for the novel face (blue). Each line independently normalized so that the maximum equals one. **bi.** Reconstructing chimp using tensorfaces with different complexities. **bii.** Relative reconstruction error for chimp as a function of tensorface complexity (rank) (blue line), as well as relative reconstruction error for the test face set (red line). The red line is again duplicated from Figure 11b. As with Yoda, reconstruction error decreases as a function of tensorface complexity for familiar faces (red) but increases for the novel face (blue).

Figure 17. Average face is at the origin of the face space. **a.** Average face, based on 128 faces in sample set. **b.** Face space as derived by multidimensional scaling (MDS). Based on responses of a population of 100 tensorface cells (rank=8) to 128 face stimuli, as well as responses of those tensorfaces to the average face. MDS reduced the original 100-dimensional face space to a 2-dimensional approximation to allow visualization. Plot symbols show positions of individual faces in the face space, classified by race and gender. Black star shows the average face located at the origin of the face space.

Figure 18. Reconstruction of a non-face object (melon) using tensorfaces (rank=8). **a. i.** Original melon image. **ii.** Reconstructed melon using tensorfaces derived from human face sample set. **iii.** Ideal reconstruction of melon, which should be null for a specialized

face processing system. **b.** Reconstruction error for melon compared to other stimuli. Human faces response shows the mean and standard deviation for 128 faces in the sample set. Other response values are for a single stimulus image. **c.** Average response magnitudes of tensorfaces to face and non-face stimuli, which are very similar. This similarity leads tensorface populations to create spurious reconstructions of non-face objects. Shows mean responses of 100 tensorfaces of rank 8 to 512 faces and 512 non-face objects. **d.** To prevent spurious reconstructions of non-face stimuli, the face identification stage (tensorfaces) require a nonlinearity. Two possible organizations for such nonlinearity are: i. Sequential nonlinearity, with nonlinear face detector stage preceding linear face identification stage in separate neurons. ii. Integrated nonlinearity with nonlinear spatial interactions present within receptive fields of single face cells. In this case face detection and face identification occur concurrently within single face cells.

Figure 1

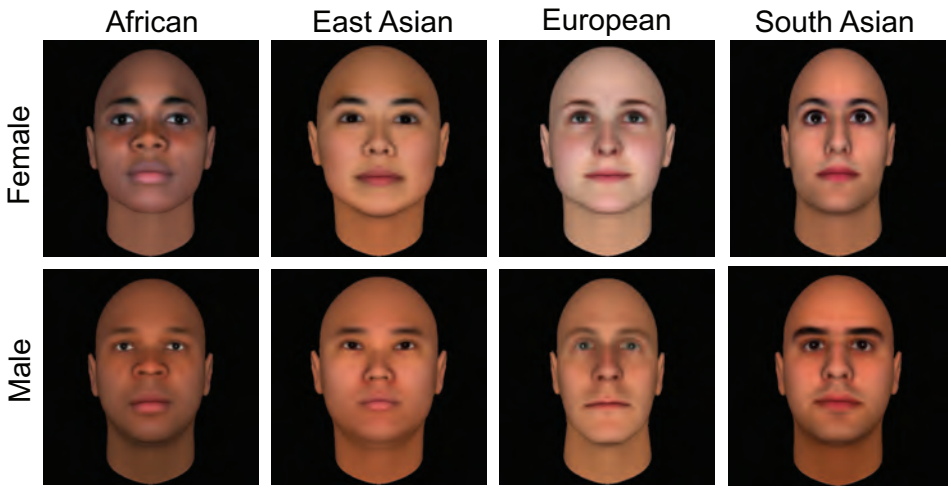
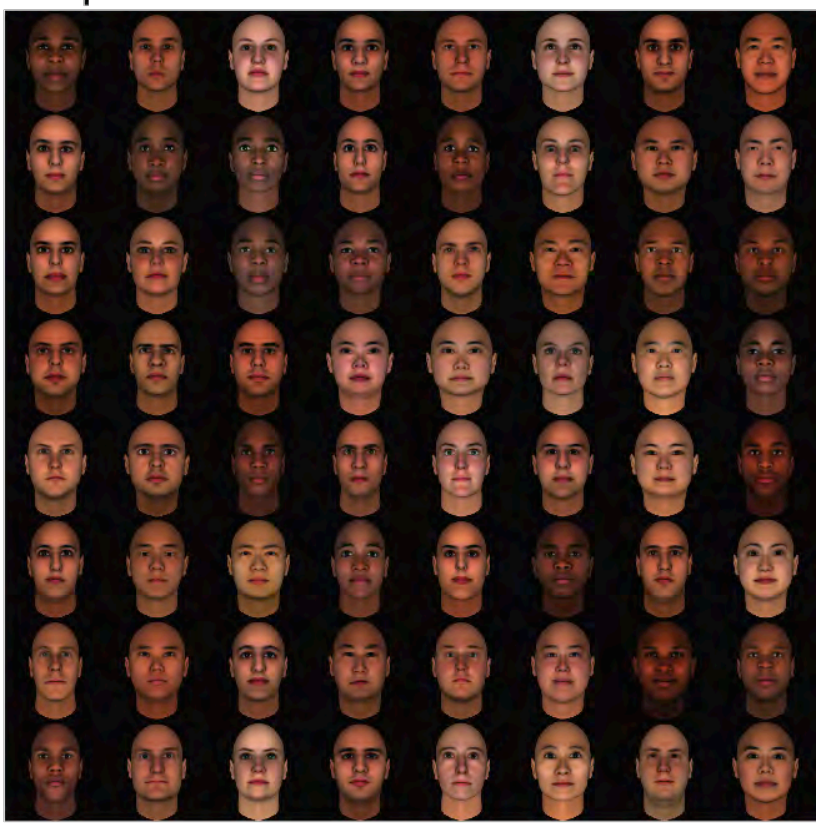


Figure 2

a. Sample set



b. Test set

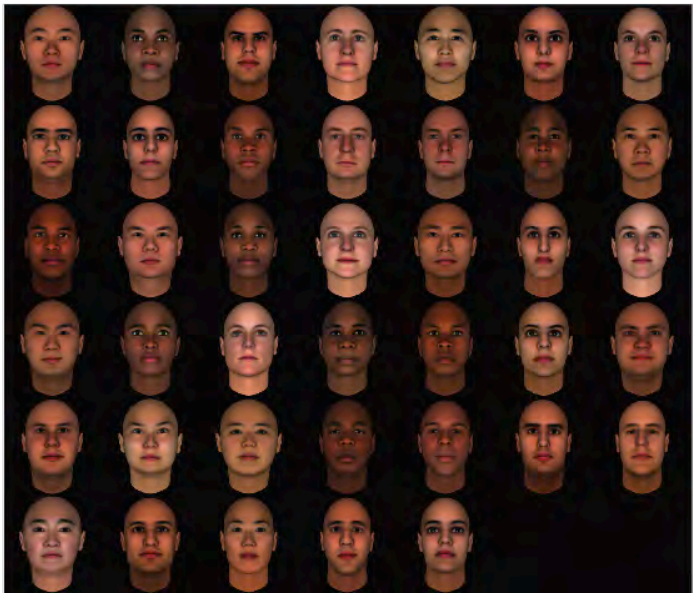


Figure 3

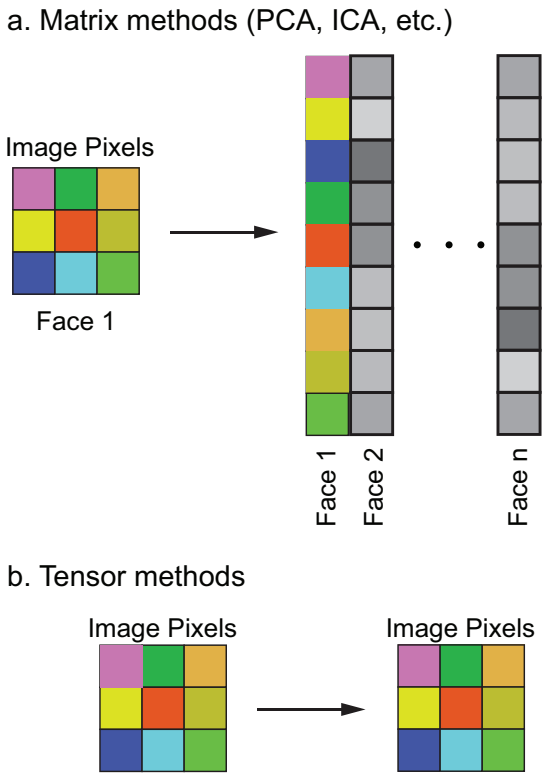

a. Matrix methods (PCA, ICA, etc.)
Image Pixels
Face 1
Face 1
Face 2
Face n
b. Tensor methods
Image Pixels
Image Pixels

Figure 4

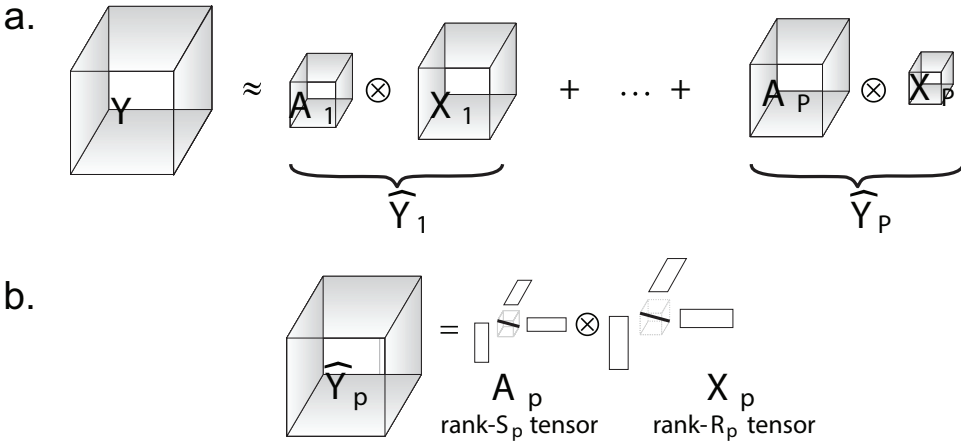


Figure 5

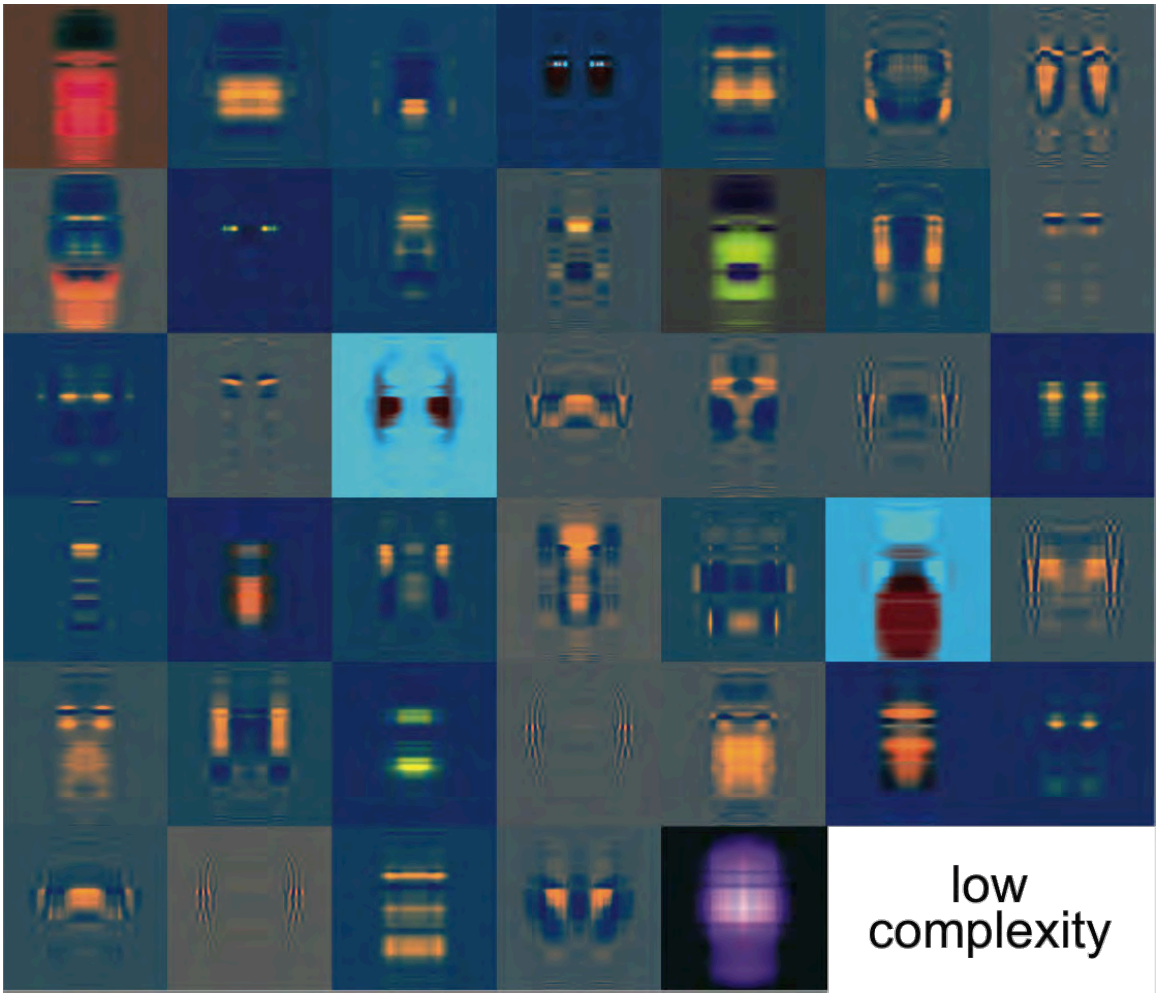


Figure 6

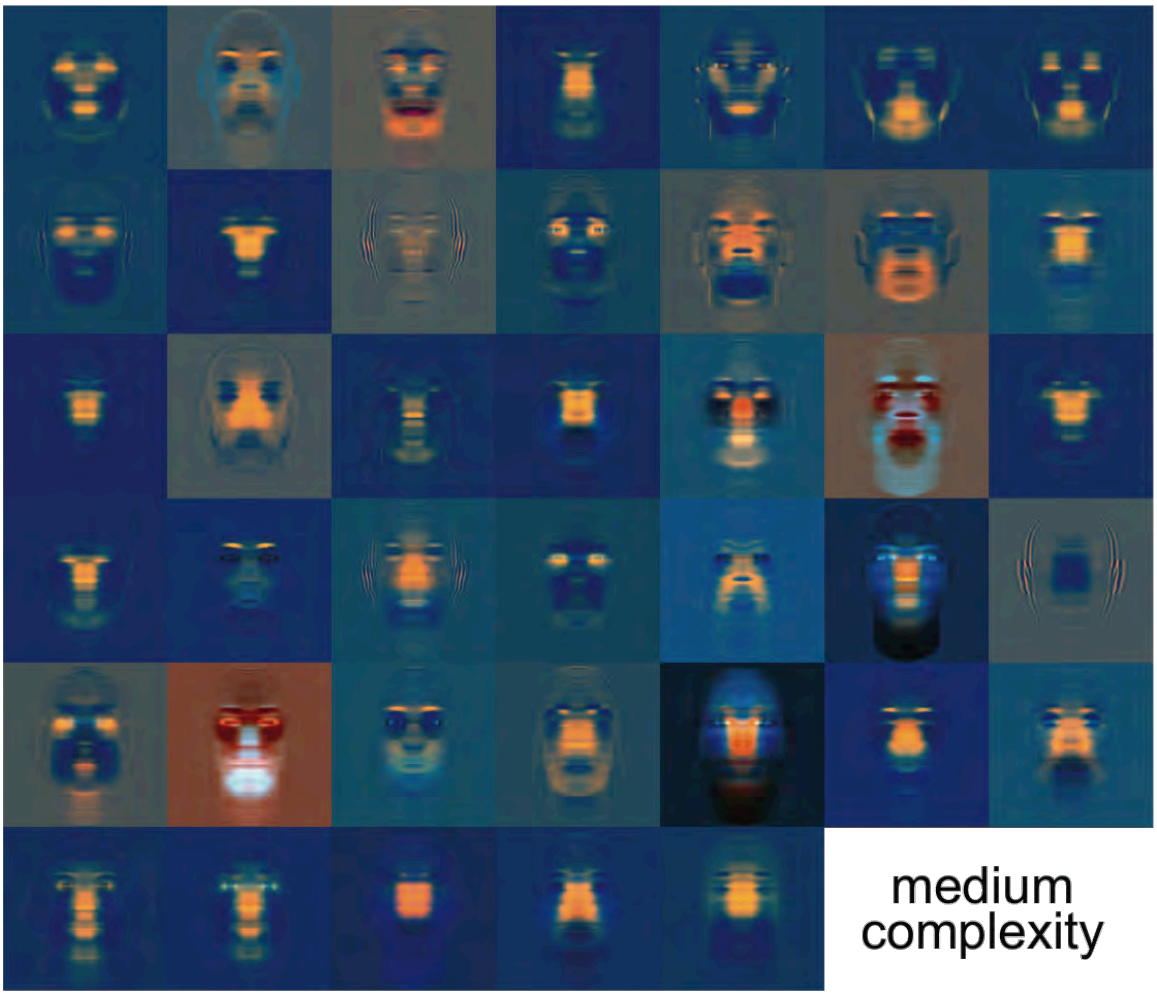


Figure 7

Figure 8

low complexity (rank=2) medium complexity (rank=8) high complexity (rank=32)

Figure 9

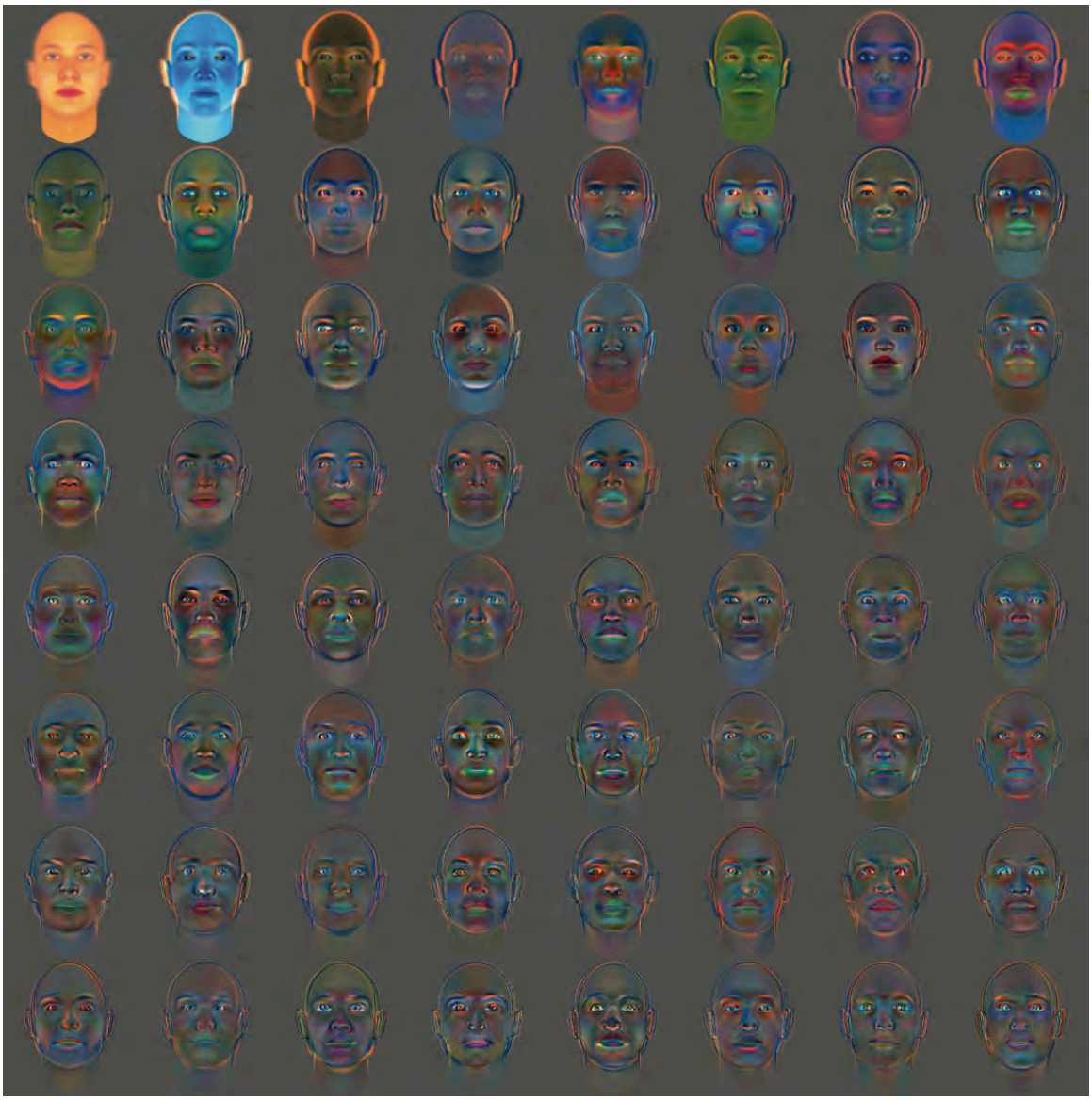


Figure 10

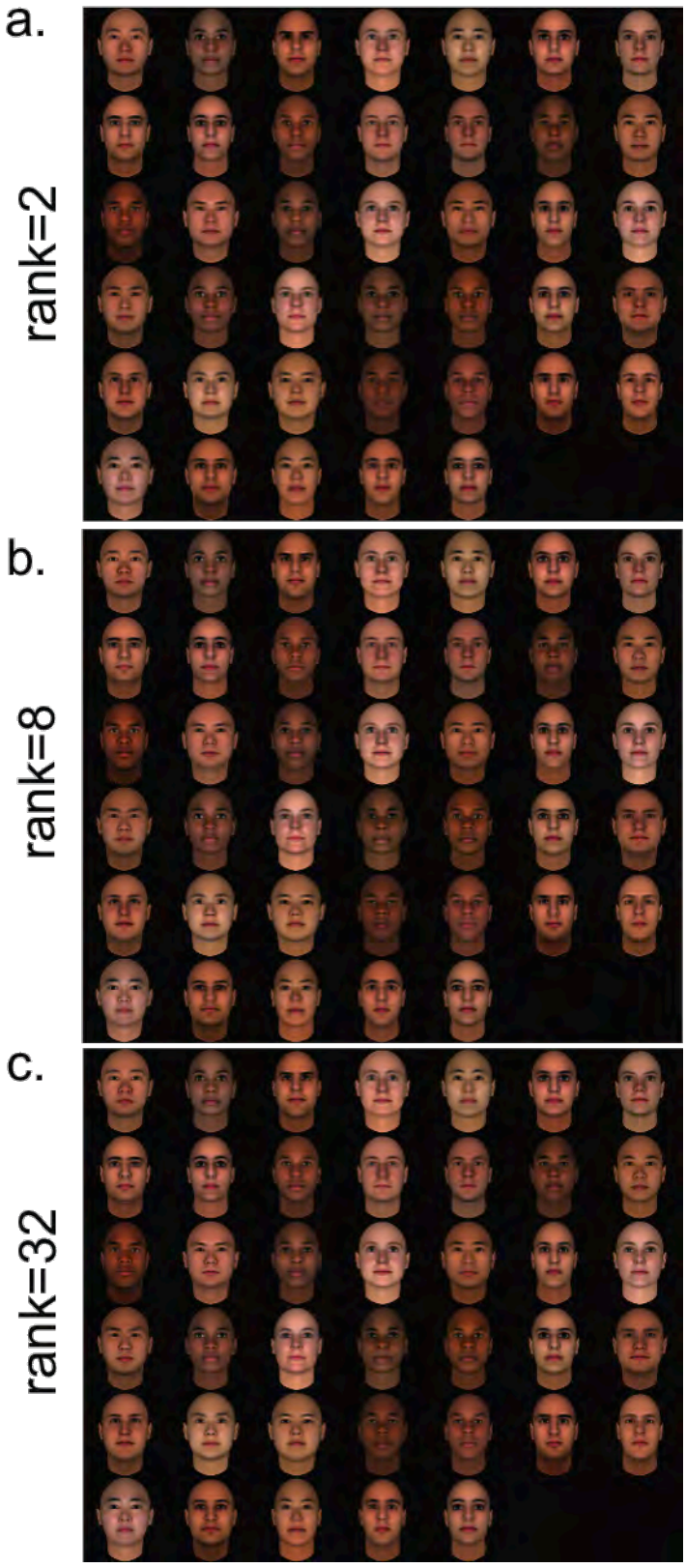


Figure 11

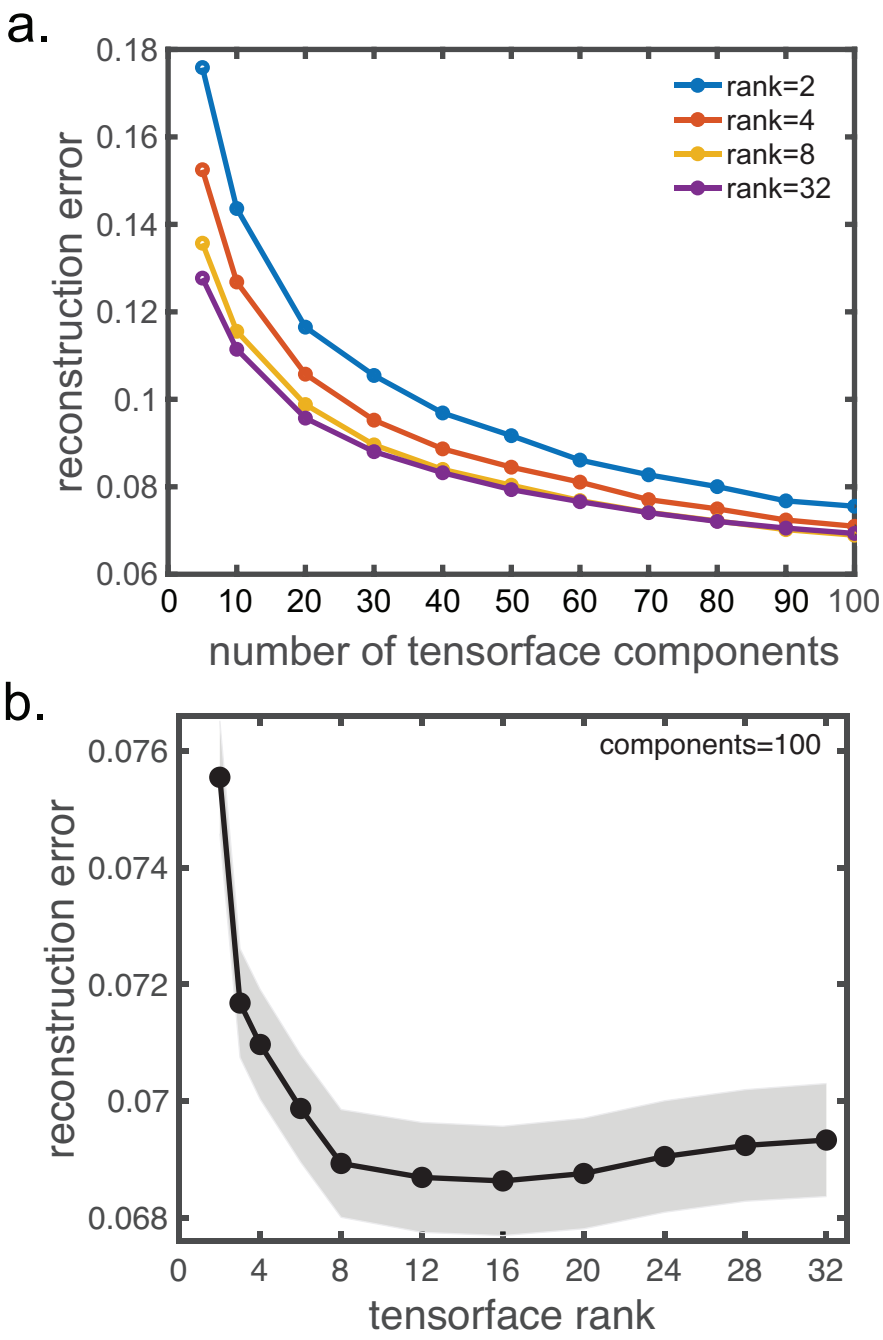

a.
0.18
0.16
0.14
0.12
0.1
0.08
0.06
reconstruction error
rank=2
rank=4
rank=8
rank=32
0
10
20
30
40
50
60
70
80
90
100
number of tensorface components
b.
0.076
0.074
0.072
0.07
0.068
reconstruction error
components=100
0
4
8
12
16
20
24
28
32
tensorface rank

Figure 12

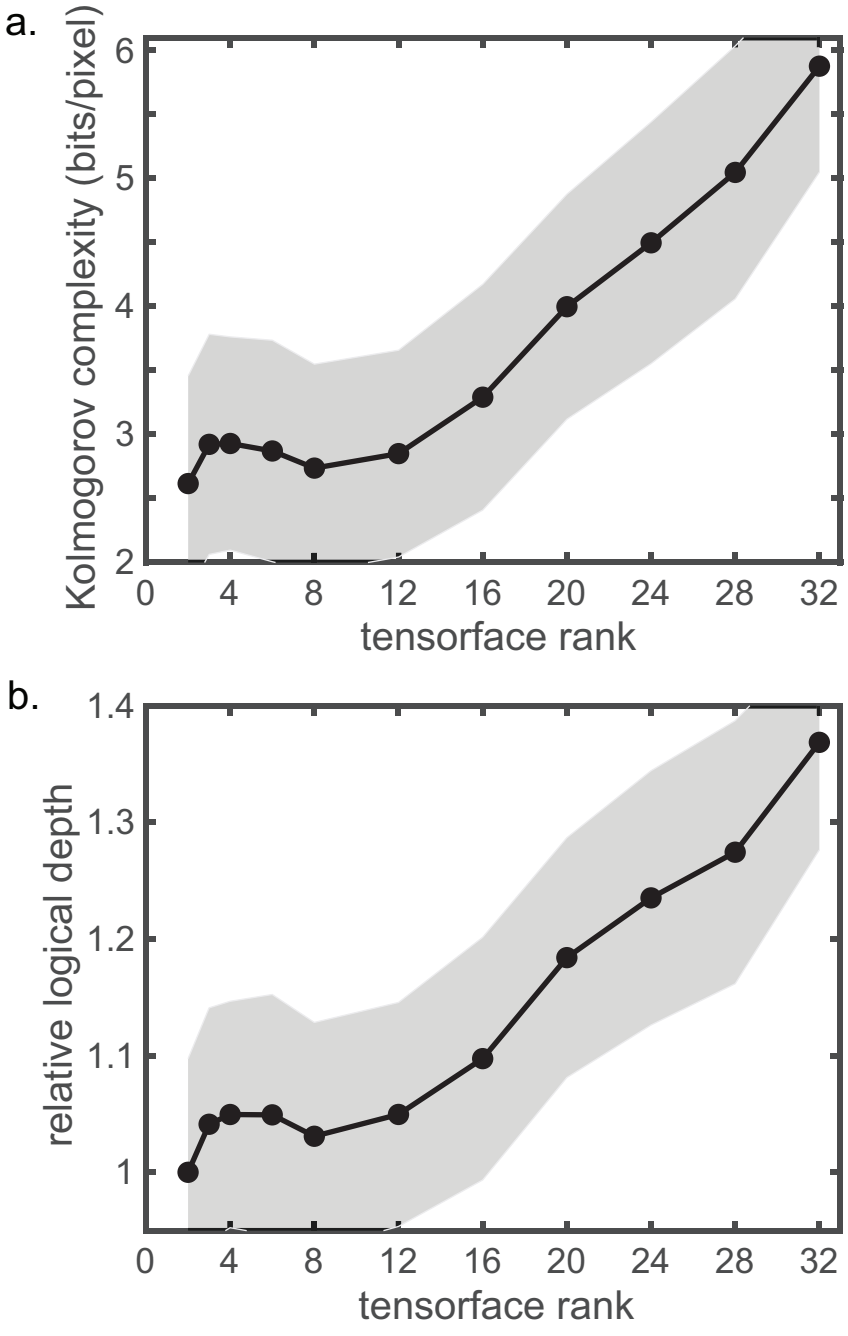

a.
Kolmogorov complexity (bits/pixel)
2
3
4
5
6
0
4
8
12
16
20
24
28
32
tensorface rank
b.
relative logical depth
1
1.1
1.2
1.3
1.4
0
4
8
12
16
20
24
28
32
tensorface rank

Figure 13

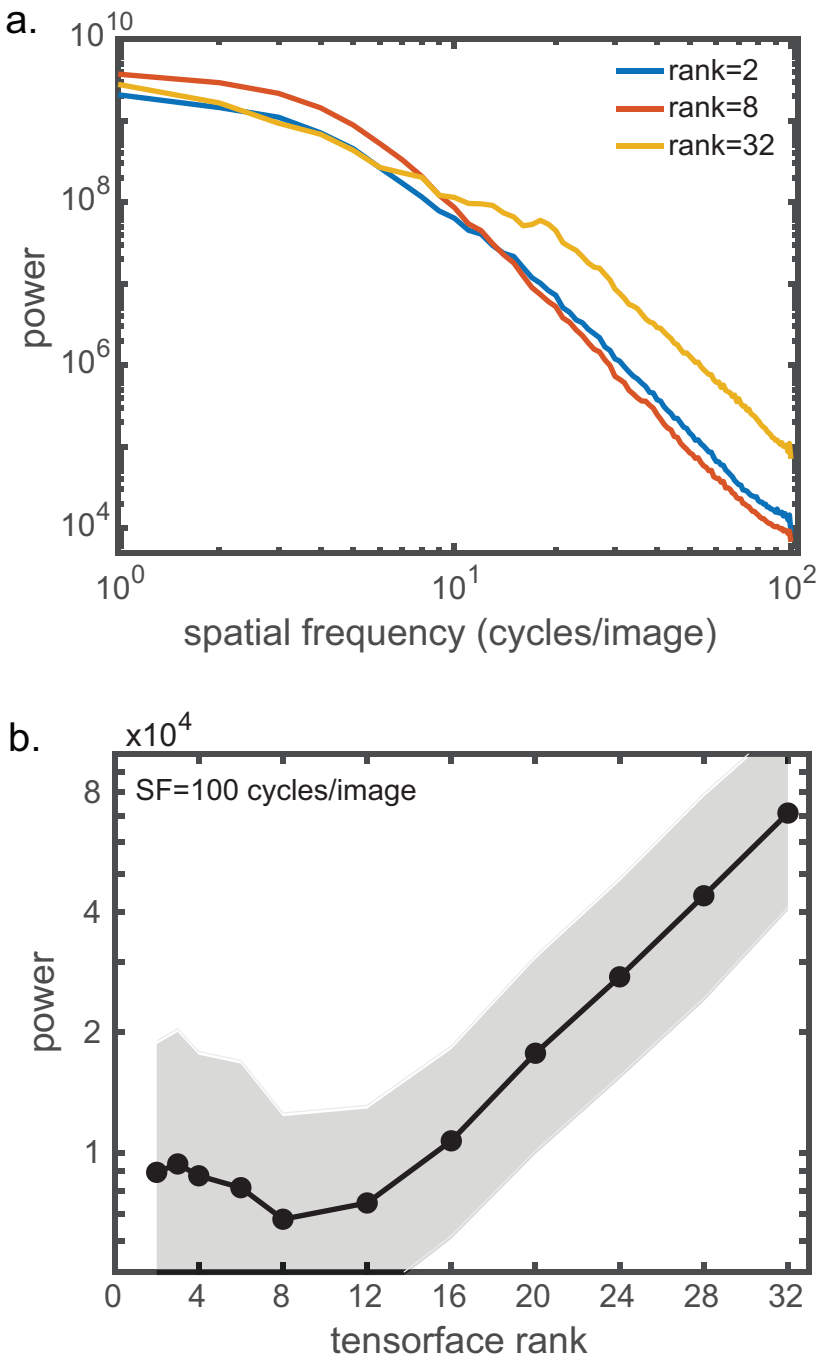

a.
rank=2
rank=8
rank=32
power
spatial frequency (cycles/image)
b.
x10^4
SF=100 cycles/image
power
tensorface rank

Figure 14

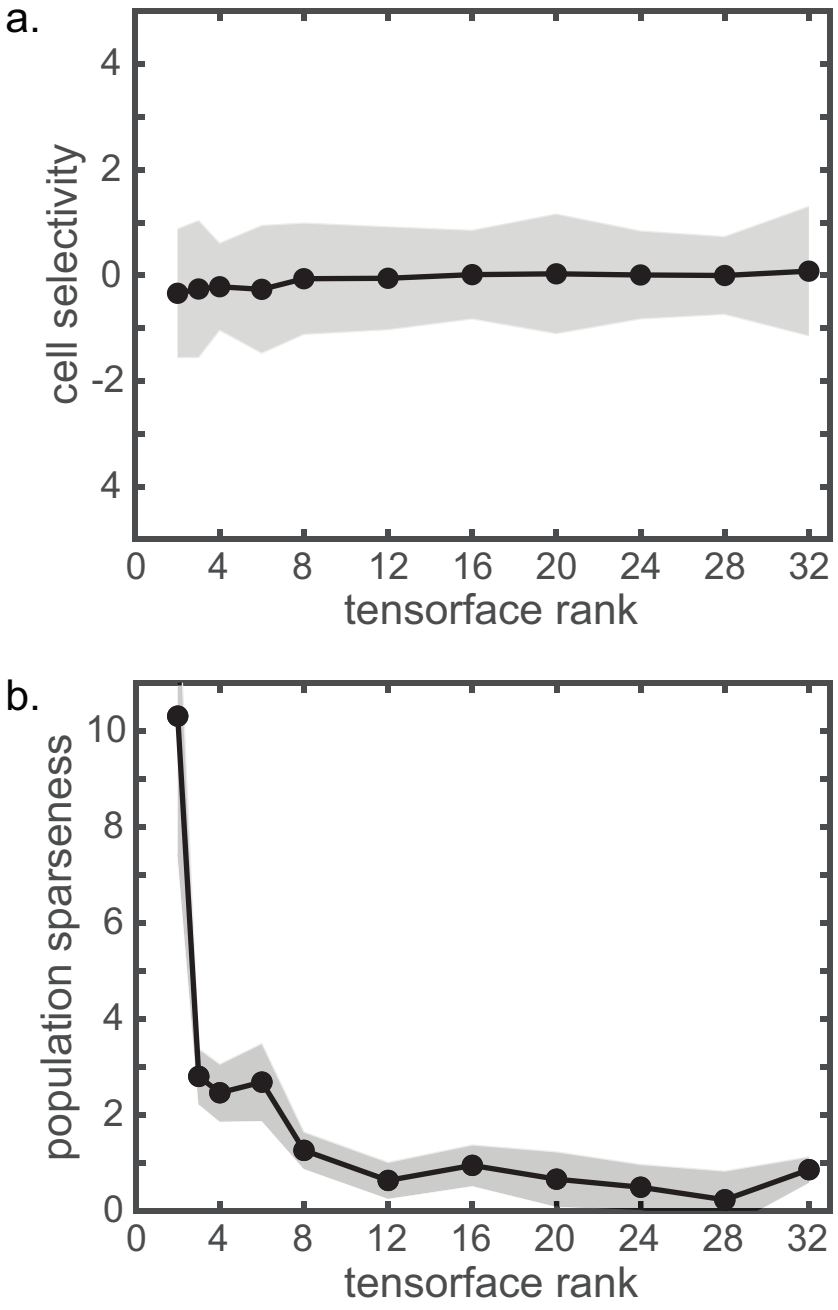

a.
cell selectivity
4
2
0
-2
4
0
4
8
12
16
20
24
28
32
tensorface rank
b.
population sparseness
10
8
6
4
2
0
0
4
8
12
16
20
24
28
32
tensorface rank

Figure 15

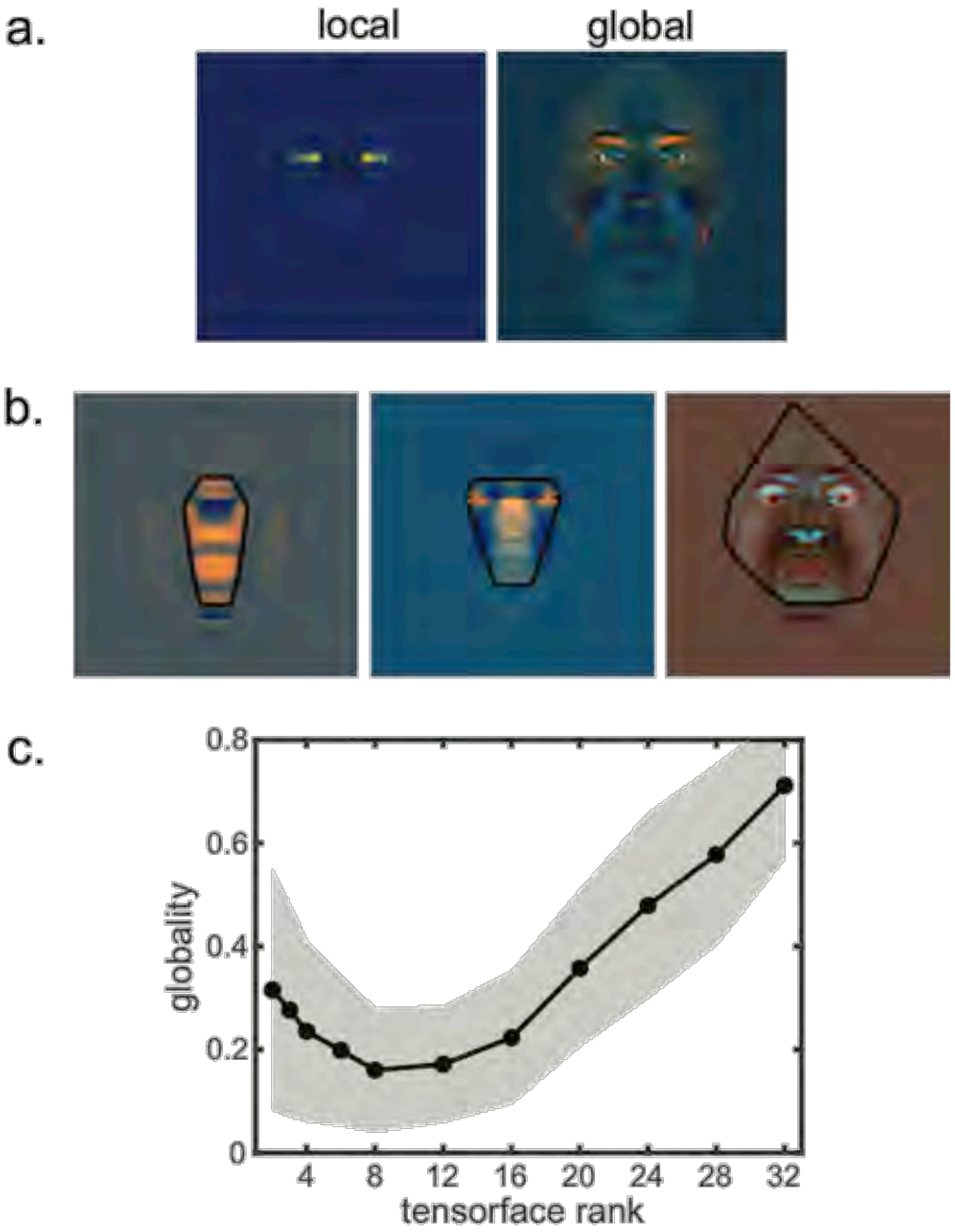

a.
local
global
b.
c.
0.8
0.6
0.4
0.2
0
globality
4
8
12
16
20
24
28
32
tensorface rank

Figure 16

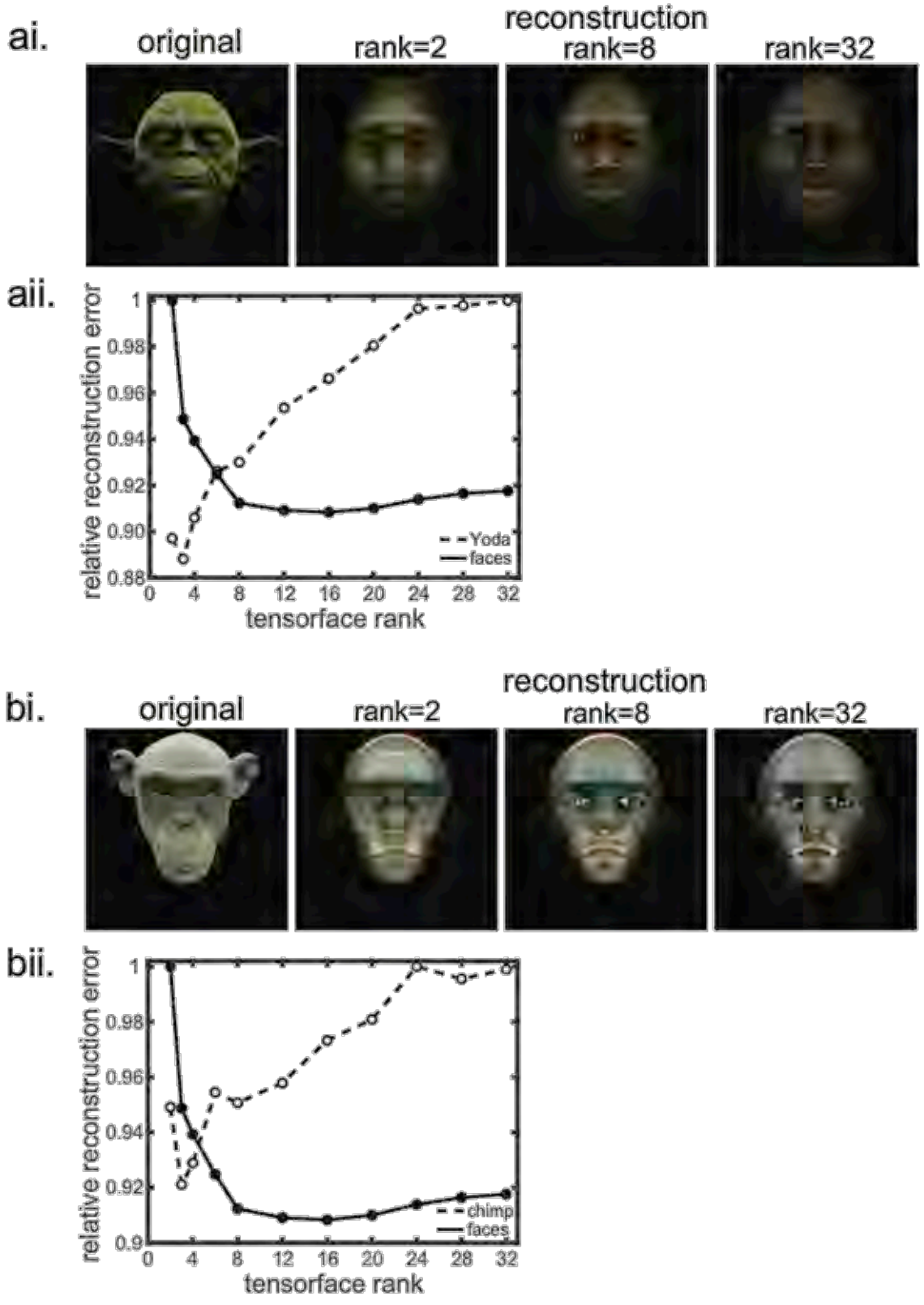

ai.
original
rank=2
reconstruction
rank=8
rank=32
aii.
relative reconstruction error
1
0.98
0.96
0.94
0.92
0.90
0.88
0
4
8
12
16
20
24
28
32
tensorface rank
Yoda
faces
bi.
original
rank=2
reconstruction
rank=8
rank=32
bii.
relative reconstruction error
1
0.98
0.96
0.94
0.92
0.9
0
4
8
12
16
20
24
28
32
tensorface rank
chimp
faces

Figure 17

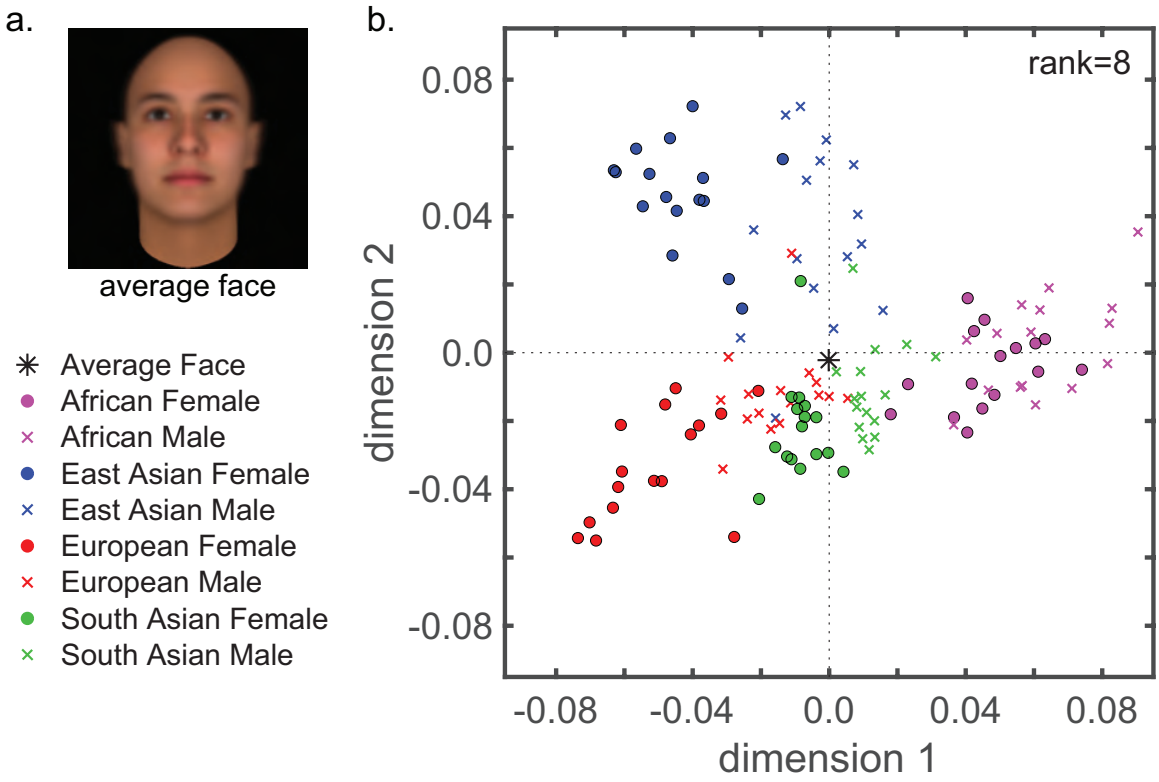
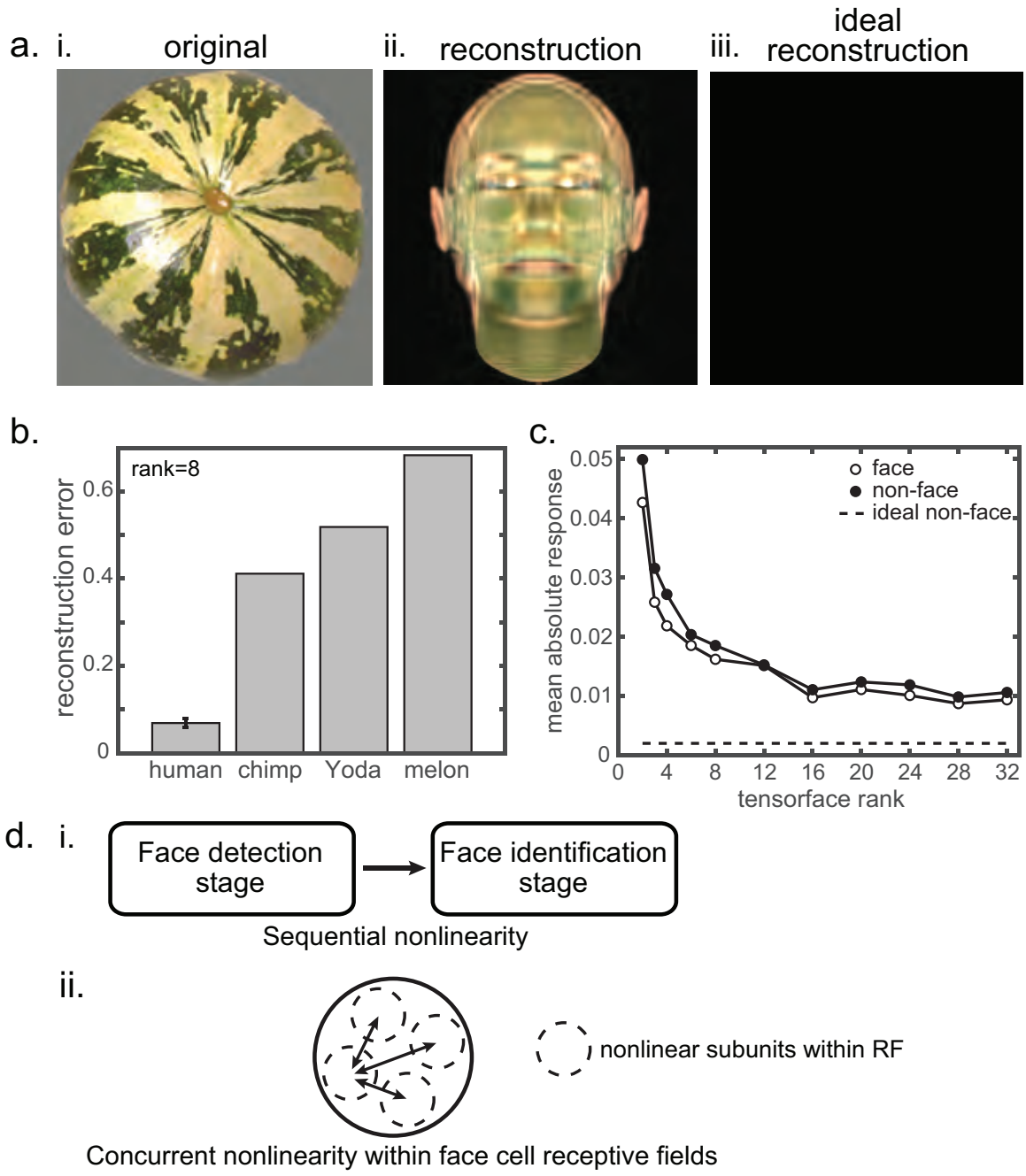

a.
average face
Average Face
African Female
African Male
East Asian Female
East Asian Male
European Female
European Male
South Asian Female
South Asian Male
b.
rank=8
0.08
0.04
0.0
-0.04
-0.08
dimension 2
-0.08
-0.04
0.0
0.04
0.08
dimension 1

Figure 18


a.
i. original
ii. reconstruction
iii. ideal reconstruction
b.
rank=8
reconstruction error
0
0.2
0.4
0.6
human
chimp
Yoda
melon
c.
face
non-face
ideal non-face
mean absolute response
0
0.01
0.02
0.03
0.04
0.05
0
4
8
12
16
20
24
28
32
tensorface rank
d.
i.
Face detection stage
Face identification stage
Sequential nonlinearity
ii.
nonlinear subunits within RF
Concurrent nonlinearity within face cell receptive fields