Supplementary Materials

Supplementary Methods

Solving the generalized eigenvalue problem. We use T to denote a metric tensors (interpreted as model-predicted perceptual thresholds in the main text), and $\boldsymbol{\epsilon}$ as a $p \times 1$ vector (we interpreted $\boldsymbol{\epsilon}$ as an image distortion in the main text). Rayleigh quotient $R(\boldsymbol{\epsilon})$ has the following expression:

$$R(\boldsymbol{\epsilon}) = \frac{\boldsymbol{\epsilon}^\top T \boldsymbol{\epsilon}}{\boldsymbol{\epsilon}^\top \boldsymbol{\epsilon}}. \tag{13}$$

To find an $\boldsymbol{\epsilon}$ that maximizes the Rayleigh quotient, we can make the following observation: assuming a fixed norm for $\boldsymbol{\epsilon}$, the denominator of $R(\boldsymbol{\epsilon})$ is a constant, and finding an $\boldsymbol{\epsilon}$ that maximizes the Rayleigh quotient is equivalent to finding an $\boldsymbol{\epsilon}$ that aligns with the principle direction of T, or the first Eigenvector direction (that corresponds to the largest Eigenvalue) of T.

The ratio of (squared) thresholds is in the form of a general Rayleigh Quotient, which can be expressed as:

$$G(\boldsymbol{\epsilon}) = \frac{\boldsymbol{\epsilon}^\top T_1 \boldsymbol{\epsilon}}{\boldsymbol{\epsilon}^\top T_2 \boldsymbol{\epsilon}}. \tag{14}$$

T_1 and T_2 are two full-rank $p \times p$ metric tensors, corresponding to two different models. To maximize or minimize this expression, we first rewrite it in the simpler form of Eq. 13. To do so, we define a new vector $\mathbf{w}$, such that $\boldsymbol{\epsilon} = T_2^{-1/2}\mathbf{w}$. Substituting $\mathbf{w}$ for $\boldsymbol{\epsilon}$, we obtain the following expression:

$$G(\mathbf{w}) = \frac{\mathbf{w}^\top (T_2^{-1/2})^\top T_1 T_2^{-1/2} \mathbf{w}}{\mathbf{w}^\top \mathbf{w}}. \tag{15}$$

The $\mathbf{w}$ that maximizes $G(\mathbf{w})$ would be the first eigenvector of the transformed metric tensor $(T_2^{-1/2})^\top T_1 T_2^{-1/2}$, such that $\left[(T_2^{-1/2})^\top T_1 T_2^{-1/2}\right] \mathbf{w}_1 = \lambda_1 \mathbf{w}_1$. Similarly, the minimal $\mathbf{w}$ is the last eigenvector. With this expression, we can find the corresponding $\boldsymbol{\epsilon}$ to the original generalized Eigenvalue problem, through

$$\begin{aligned}
& \left[(T_2^{-1/2})^\top T_1 T_2^{-1/2}\right] \mathbf{w}_1 = \lambda_1 \mathbf{w}_1 && (16)\\
\Longrightarrow \quad & \left[(T_2^{-1/2})^\top T_1\right] \boldsymbol{\epsilon}_1 = \lambda_1 T_2^{1/2} \boldsymbol{\epsilon}_1 \quad \text{(substitution)} && (17)\\
\Longrightarrow \quad & \left(T_2^{-1} T_1\right) \boldsymbol{\epsilon}_1 = \lambda_1 \boldsymbol{\epsilon}_1 \quad \text{(multiple } T_2^{-1/2} \text{ on both sides).} && (18)
\end{aligned}$$

And $\boldsymbol{\epsilon}_1$, the solution to the original generalized Eigenvalue problem, is the first eigenvector of matrix $T_2^{-1}T_1$. The expression would be similar if we look for the last Eigenvector instead of the first. Notice that the matrix $T_2^{-1}T_1$ is not always symmetric, so the right eigenvectors of this matrix are not necessarily orthogonal to each other.

Two representations that are equivalent in terms of their metric properties (Figure 1) In figure 1, we showed two example representations that have equivalent metric properties (as measured using the Fisher lower bound). Here, we present the calculation that shows the equivalence of the two representations. For this figure, stimulus $\mathbf{s}$ is assumed to be two-dimensional, and $\mathbf{s} = [s_1,\ s_2]^\top$.

First representation. We analyze a case where a two-dimensional stimulus space is represented by three neurons, the responses of which are assumed stochastic and independent (no correlation). The three neurons' response means and standard deviations are designed to be:

$$\begin{aligned}
\mu_1(\mathbf{s}) &= 2s_1, \quad \sigma_1(\mathbf{s}) = \sqrt{\mu_1(\mathbf{s})}; && (19)\\
\mu_2(\mathbf{s}) &= 3s_2, \quad \sigma_2(\mathbf{s}) = \sqrt{\mu_2(\mathbf{s})}; && (20)\\
\mu_3(\mathbf{s}) &= s_2, \quad \sigma_3(\mathbf{s}) = \sqrt{\mu_3(\mathbf{s})}. && (21)
\end{aligned}$$

Fisher discriminability (and thresholds) for the three neurons has the following form:

$$M_1(\mathbf{s}) = \begin{bmatrix} 2 & 0 & 0 \\ 0 & 3 & 1 \end{bmatrix} \begin{bmatrix} (2s_1)^{-1} & 0 & 0 \\ 0 & (3s_2)^{-1} & 0 \\ 0 & 0 & s_2^{-1} \end{bmatrix} \begin{bmatrix} 2 & 0 \\ 0 & 3 \\ 0 & 1 \end{bmatrix} = \begin{bmatrix} 2/s_1 & 0 \\ 0 & 4/s_2 \end{bmatrix}; \tag{22}$$

$$T_1(\mathbf{s}) = \begin{bmatrix} s_1/2 & 0 \\ 0 & s_2/4 \end{bmatrix}. \tag{23}$$

Second representation. In the second representation, we also assume three neurons stochastically (and independently) responding to the same stimulus space. The mean and standard deviation of the neural population are designed to be:

$$\mu_1(\mathbf{s}) = 2s_1^{1/2}, \quad \sigma_1(\mathbf{s}) = 1/2; \tag{24}$$

$$\mu_2(\mathbf{s}) = 2s_2^{1/2}, \quad \sigma_2(\mathbf{s}) = 1; \tag{25}$$

$$\mu_3(\mathbf{s}) = 3s_2, \quad \sigma_3(\mathbf{s}) = \sqrt{\mu_3(\mathbf{s})}. \tag{26}$$

Computing Fisher discrimnability and threshold for this new set of neurons give us:

$$M_2(\mathbf{s}) = \begin{bmatrix} s_1^{-1/2} & 0 & 0 \\ 0 & s_2^{-1/2} & 3 \end{bmatrix} \begin{bmatrix} 2 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & (3s_2)^{-1} \end{bmatrix} \begin{bmatrix} s_1^{-1/2} & 0 \\ 0 & s_2^{-1/2} \\ 0 & 3 \end{bmatrix} = \begin{bmatrix} 2/s_1 & 0 \\ 0 & 4/s_2 \end{bmatrix}; \tag{27}$$

$$T_2(\mathbf{s}) = \begin{bmatrix} s_1/2 & 0 \\ 0 & s_2/4 \end{bmatrix}. \tag{28}$$

Because the two sets of neurons' metric tensors have the same form, the two representations are equivalent in terms of their metric properties.

S1. Experimental cost of measuring high-dimensional human perceptual discriminability

Suppose an image patch consists of 100×100 pixels, and the total number of pixels is 10000. Conservatively, let us assume that the human metric tensor at a reference image has a rank 30 (30 linearly independent neurons contribute to perceiving the image), and to reliably recover the human metric tensor, we need an order of 30×10000 number of samples. Because each sample is a measurement of perceptual threshold along an image distortion. Conservatively, we assume that it takes around 2 minutes to estimate a single threshold. The total number of hours required to recover the human metric tensor can be computed as the following:

$$30 \times 10000 \times 2/60 = 10000 \;\text{ hours.} \tag{29}$$

So conservatively, it takes an order of 10,000 hours to estimate a human metric tensor for a single reference image.

S2. Fisher information

For simplicity of intuition, we first assumed that stimulus s only consists of a single dimension. Fisher Information measures the precision with which stimuli are encoded in noisy measurements. Fisher information is often used in the context of the Cramér-Rao bound in statistics and engineering literature, and it states an upper bound on the precision (inverse variance) attainable by an unbiased estimator. In perceptual literature, the square root of Fisher information provides a bound on perceptual precision (discriminability), and can be viewed as a generalization of "d-prime", the traditional metric of signal detection used in perceptual studies.

For a stimulus attribute s (a single dimension), the Fisher Information can be derived from a stochastic model of neural responses, $p(r|s)$, and can be written as:

$$M(s) = \sqrt{\mathbb{E}\left[\left(\frac{\partial \log p(r|s)}{\partial s}\right)^2 \middle| s\right]}. \tag{30}$$

The expression captures how much the neural response distribution changes when the stimulus is perturbed.

Intuitively, the square-root of Fisher information accounts for discriminability in two steps: first, we take a derivative of $p(r|s)$ with respect to s to account for how the stochastic representation varies with the stimulus. Second, to quantify *the extent of* change in the representation, we compute the expected squared norm of this derivative under the probability $p(r|s)$. This definition relies only on the differentiability of the measurement distribution with respect to s and some modest regularity conditions, but does not make assumptions regarding the form of the response density. Generally, both s and r can be vector-valued (as in the main text), but the intuition and interpretation remains the same.

S3. Comparing metric properties between deterministic and stochastic models

Because metric tensors can be computed for both types of models, they can be compared across the two model types. Conceptually, the metric tensors computed for deterministic and stochastic models are very similar, with the only difference being that metric tensors for stochastic models take the precision matrix of neural responses (the inverse of covariance matrix) into account. One caveat is that as the noise in a stochastic model decreases in amplitude, the covariance matrix does not converge to an identity matrix, and neither does the precision matrix. The consequence is that the stochastic metric tensors do not converge to the corresponding deterministic metric tensors (in terms of the size of the elliptical metric tensor representations). This caveat does not affect our model comparison method because we compare the orientations (and shape) of the metric tensors, and the sizes of the metric tensors do not affect our model comparison results.

S4. The effect of metric tensor size in the model comparison

We examine how scaling metric tensors affects the solution using one of the two objectives in the main text, stated as follows:

$$\boldsymbol{\epsilon}_1(\mathbf{s}) = \arg\max_{\boldsymbol{\epsilon}} \left[\frac{\boldsymbol{\epsilon}^\top T_{\mathbf{f}_1}(\mathbf{s})\boldsymbol{\epsilon}}{\boldsymbol{\epsilon}^\top T_{\mathbf{f}_2}(\mathbf{s})\boldsymbol{\epsilon}}\right]. \tag{31}$$

The solution to the above objective is the first eigenvector of matrix $T_{\mathbf{f}_2}^{-1}(\mathbf{s})T_{\mathbf{f}_1}(\mathbf{s})$. Suppose we scale both metric tensors with $T_{\mathbf{f}_1}^* = \lambda_1 T_{\mathbf{f}_1}$, and $T_{\mathbf{f}_2}^* = \lambda_2 T_{\mathbf{f}_2}$. Now, the solution to the scaled objective becomes the lead eigenvector of $T_{\mathbf{f}_2}^{*\,-1}(\mathbf{s})T_{\mathbf{f}_1}^*(\mathbf{s})$. Because $T_{\mathbf{f}_2}^{*\,-1}(\mathbf{s})T_{\mathbf{f}_1}^*(\mathbf{s}) = \frac{\lambda_2}{\lambda_1} T_{\mathbf{f}_2}^{-1}(\mathbf{s})T_{\mathbf{f}_1}(\mathbf{s})$, the lead eigenvector of the pre-scaled matrix maintains to be the solution.

S5. Transforming metric tensors

Here, we illustrate how to transform metric tensors using a deterministic neural model. Transforming stochastic neural models can be similarly computed. First, we use a differentiable map to transform a stimulus $\mathbf{s}$, and $\tilde{\mathbf{s}} = \mathbf{g}(\mathbf{s})$. $\mathbf{g}$ can be either linear (e.g. cropping an image) or nonlinear (e.g. the overall contrast of an image). Assuming neurons differentiably map from transformed stimulus $\tilde{\mathbf{s}}$ to different levels of responses, represented by $\mathbf{f}(\tilde{\mathbf{s}})$. We can compute the metric tensor of the map $\mathbf{f}$ as the following:

$$M_{\mathbf{f}}(\tilde{\mathbf{s}}) = J_{\mathbf{f}}^\top(\tilde{\mathbf{s}})J_{\mathbf{f}}(\tilde{\mathbf{s}}). \tag{32}$$

The map $\mathbf{f}[\mathbf{g}(\mathbf{s})]$ summarizes the map from the original stimulus space to neural responses. The metric tensor for the map $\mathbf{f}[\mathbf{g}(\mathbf{s})]$, $M_{\mathbf{fg}}(\mathbf{s})$ has the following form:

$$M_{\mathbf{fg}}(\mathbf{s}) = J_{\mathbf{g}}(\mathbf{s})^\top J_{\mathbf{f}}(\tilde{\mathbf{s}})^\top J_{\mathbf{f}}(\tilde{\mathbf{s}})J_{\mathbf{g}}(\mathbf{s}) \tag{33}$$

$$= J_{\mathbf{g}}(\mathbf{s})^\top M_{\mathbf{f}}(\tilde{\mathbf{s}})J_{\mathbf{g}}(\mathbf{s}). \tag{34}$$

S6. Low-rank metric tensor comparison

When either $M_{\mathbf{f}_1}(\mathbf{s})$ or $M_{\mathbf{f}_2}(\mathbf{s})$ is rank-deficient (or both), we can still find optimal perturbations based on the logic of our method. As an example, let us consider obtaining $\boldsymbol{\epsilon}_1$:

$$\boldsymbol{\epsilon}_1 = \arg\max_{\boldsymbol{\epsilon}} \left[\frac{\boldsymbol{\epsilon}^\top M_{\mathbf{f}_2}(\mathbf{s})\boldsymbol{\epsilon}}{\boldsymbol{\epsilon}^\top M_{\mathbf{f}_1}(\mathbf{s})\boldsymbol{\epsilon}}\right] \tag{35}$$

Because the metric tensors (especially $M_{\mathbf{f}_1}(\mathbf{s})$) are rank deficient, we cannot obtain a perturbation direction as we described in the main text. Also notice that we stated the objective in terms of discriminability instead of threshold, because the discriminability metric tensors may not be invertible. So alternatively, we describe how to find the optimal perturbation in the fashion of Equation 9.

To find the optimal perturbation, we projected $M_{\mathbf{f}_2}(\mathbf{s})$ to the nullspace of $M_{\mathbf{f}_1}(\mathbf{s})$. We find the lead eigenvector of the projections. Say V is a matrix whose each column is a length 1 orthogonal basis vector of the nullspace of $M_{\mathbf{f}_1}$. The eigenvectors of the projections can be found by performing eigendecomposition on the following matrix:

$$VV^\top M_{\mathbf{f}_2}(\mathbf{s})VV^\top. \tag{36}$$

Now we can find the lead eigenvector (that corresponds to the largest Eigenvalue of the above matrix) of the above matrix, which would be the solution $\boldsymbol{\epsilon}_1$.

$VV^\top$ is called a projection matrix. For example, one can project a vector x into a subspace spanned by the orthogonal column vectors of V, by computing $VV^\top x$. The projection of a matrix into a subspace can be intuitively understood when it is described in terms of the eigenvectors of the matrix. Say the eigenvalue decomposition of matrix M can be stated as $M = U\Sigma U^\top$, where Σ is a diagonal matrix, and the columns of U are orthonormal vectors that form the basis of M. The projection of the basis of M into the subpace spanned by V is $P = VV^\top U$. Finally, the projection of M into the subspace is given by $P\Sigma P^\top$.

S7. Additional details for chromaticity models

The chromatic diagram, or the xyY color space, assumes a fixed luminance level Y, the value of which we empirically estimated from MacAdam's data. The process of generating chromatic threshold predictions ([17]) takes two steps, a forward step, and a backward step. In the forward step, we sampled reference colors in the xy chromatic diagram, and mapped them to the space of cone responses. In the backward step, we used the fisher information matrix to compute threshold predictions in the space of cone responses, and map them back to the chromatic diagram.

Forward step. It takes two transforms to map reference colors sampled in the xyY space to the space of neural responses. First, we mapped the reference colors from the xyY space to the XYZ color space (same color space but with varied luminance levels) via a non-linear transform. To be specific, let $\mathbf{c}$ be a reference color in the xyY space (a three-dimensional vector with luminance level Y fixed): $\mathbf{c} = [x, y, Y]^\top$. The inverse of the nonlinear transform from XYZ to xyY space, denoted as f_1^{-1}, is more intuitively defined as normalizing the luminance levels of the XYZ space:

$$x = \frac{X}{X+Y+Z}, \qquad y = \frac{Y}{X+Y+Z}, \qquad Y = Y. \tag{37}$$

The inverse map $\mathbf{f}_1 : (x, y, Y) \to (X, Y, Z)$ is:

$$X = \frac{Yx}{y}, \qquad Y = Y, \qquad Z = \frac{Y(1-x-y)}{y}. \tag{38}$$

In the second step, we further projected the reference colors from the XYZ to the LMS space of cone responses via a linear map. We used $\boldsymbol{\alpha} = [\alpha_S,\, \alpha_M,\, \alpha_L]^\top$ to indicate the rates of cone responses. Furthermore, we used $\boldsymbol{\beta} = [\beta_S,\, \beta_M, \beta_L]$ (empirically estimated) to indicate the proportion of S, M and L cones in a (small region with the) retina, and $\beta_S + \beta_M + \beta_L = 1$. The transform from XYZ to LMS space is linear, and can be represented by a 3×3 matrix F_2, which empirically estimated in [17] as:

$$F_2 \begin{bmatrix} X \\ Y \\ Z \end{bmatrix} = \begin{bmatrix} \alpha_S \\ \alpha_M \\ \alpha_L \end{bmatrix}, \qquad \textit{and} \quad F_2 = \begin{bmatrix} 0.038\beta_S & -0.043\beta_S & 0.48\beta_S \\ -0.39\beta_M & 1.17\beta_M & 0.049\beta_M \\ 0.34\beta_L & 0.69\beta_L & -0.076\beta_L \end{bmatrix}. \tag{39}$$

Both $\mathbf{f}_1$ and F_2 are invertible transforms, and the forward computation can be summarized as:

$$F_2 \mathbf{f}_1(\mathbf{c}) = \boldsymbol{\alpha}. \tag{40}$$

Backward step. We mapped colors sampled from the xyY chromatic space to the space of cone responses. In the Backward step, we mapped the predicted perceptual thresholds from the space of cone responses back to the xyY chromatic space.

First, we assumed either independent Gaussian (same variance for all cone types) or Poisson noise to compute Fisher discriminability (the lower bound on Fisher information). This computation only involves the mean and precision matrix (the inverse of covariance matrix) of the cone responses.

We assumed the mean neural responses coincide with the three-dimensional light input $\boldsymbol{\alpha}$, and the derivative of the mean response (with respect to $\boldsymbol{\alpha}$) is the identity matrix $\mathbf{I}$. In the case of Poisson noise, each neuron's response variance is identical to the mean, and the Fisher discriminability $D(\boldsymbol{\alpha})$, as well as Fisher thresholds, which is the matrix inverse of discriminability, can be expressed as:

$$D(\boldsymbol{\alpha}) = \begin{bmatrix} \frac{1}{\alpha_S} & 0 & 0 \\ 0 & \frac{1}{\alpha_M} & 0 \\ 0 & 0 & \frac{1}{\alpha_L} \end{bmatrix}, \qquad \textit{and} \quad T(\boldsymbol{\alpha}) = D^{-1}(\boldsymbol{\alpha}) = \begin{bmatrix} \alpha_S & 0 & 0 \\ 0 & \alpha_M & 0 \\ 0 & 0 & \alpha_L \end{bmatrix}. \tag{41}$$

Predicted discriminability and threshold assuming independent Gaussian noise can be similarly computed.

To visualize thresholds in the xyY chromatic space, we transform $T(\boldsymbol{\alpha})$ via two stages. To transform mean neural response $\boldsymbol{\alpha}$ back to the xyY space, we compute $\mathbf{f}_1^{-1}(F_2^{-1}\boldsymbol{\alpha})$. To transform thresholds $T(\boldsymbol{\alpha})$ (a quadratic form), back to the chromatic space, has the following expression:

$$\tilde{T}(\boldsymbol{\alpha}) = \left[\nabla \mathbf{f}_1^{-1} F_2^{-1}\right] T(\boldsymbol{\alpha}) \left[\nabla \mathbf{f}_1^{-1} F_2^{-1}\right]^\top. \tag{42}$$

$\nabla \mathbf{f}_1^{-1}$ is the Jacobian of the map $\mathbf{f}_1^{-1}$ with respect to input $(F_2^{-1}\boldsymbol{\alpha})$, and $\nabla \mathbf{f}_1^{-1}$ has a form:

$$\nabla \mathbf{f}_1^{-1} = \begin{bmatrix} \frac{\partial x}{\partial X} & \frac{\partial x}{\partial Y} & \frac{\partial x}{\partial Z} \\ \frac{\partial y}{\partial X} & \frac{\partial y}{\partial Y} & \frac{\partial y}{\partial Z} \\ \frac{\partial Y}{\partial X} & \frac{\partial Y}{\partial Y} & \frac{\partial Y}{\partial Z} \end{bmatrix} = \begin{bmatrix} \frac{Y}{y} & -\frac{xY}{y^2} & \frac{x}{y} \\ 0 & 0 & 1 \\ -\frac{Y}{y} & -\frac{Y(1-x)}{y^2} & \frac{1-x-y}{y} \end{bmatrix}. \tag{43}$$

Because F_2^{-1} is a linear map, the Jacobian of which has a form $\nabla F_2^{-1} = F_2^{-1}$. The chromatic diagram lives in a two dimensional space (a hyperplane in the xyY color space), and the transformed metric tensor is a 3×3 matrix. To visualize perceptual thresholds in the two-dimensional xy space, as in [17], we extracted a 2×2 sub-matrix of $\tilde{T}(\boldsymbol{\alpha})$, $\tilde{T}(\boldsymbol{\alpha})_{\{i \le 2,\ j \le 2\}}$, that represents the transformed threshold matrix restricted to the xy hyperplane.

S8. Euclidean metric for matrix distances

Between each pair of measured and predicted metric tensors $T_m(\mathbf{s})$ and $T_p(\mathbf{s})$, we compared their distances, before summing up the distances across all pairs of such metric tensors in the chromatic diagram.

We used Frobenius norm to quantify distances between a measured metric tensor $T_m(\mathbf{s})$, and the corresponding prediction $T_p(\boldsymbol{\alpha})$ (from either a Gaussian, a Poisson, or a hybrid model). The Frobenius norm can be expressed as:

$$d_E\left[T_m(\mathbf{s}),\ T_p(\mathbf{s})\right] = ||T_m(\mathbf{s}) - T_p(\mathbf{s})|| = \sqrt{\text{trace}\left\{\left[T_m(\mathbf{s}) - T_p(\mathbf{s})\right]^\top \left[T_m(\mathbf{s}) - T_p(\mathbf{s})\right]\right\}}. \tag{44}$$

Because the space of symmetric positive definite matrices is not an Euclidean space, and some other choices of quantifying distances between such matrices could be explored. As an alternative metric, weh minimize (the sum of) the Frobenius norm of the difference between the square-root of the metric tensors, expressed as:

$$d_S\left[T_m(\mathbf{s}),\ T_p(\mathbf{s})\right] = ||T_m(\mathbf{s})^{1/2} - T_m(\mathbf{s})^{1/2}||. \tag{45}$$

The model comparison result conducted using the square-root distance is similar to the result shown in the main text (computed with Frobenius distance).

S9. Alternative set-up for model comparisons

In the main text, we find the image distortions that maximizes the ratio between two models' metric tensor predictions. Alternatively, we can maximize the difference between the metric tensor predictions. Here, we examine one such objective function:

$$\text{Find } \boldsymbol{\epsilon}^* \quad \text{s.t.} \quad \boldsymbol{\epsilon}^* = \arg\max_{\boldsymbol{\epsilon}} \left[\boldsymbol{\epsilon}^\top T_1 \boldsymbol{\epsilon} - \boldsymbol{\epsilon}^\top T_2 \boldsymbol{\epsilon}\right] \quad \text{and} \quad \boldsymbol{\epsilon}^\top \boldsymbol{\epsilon} = 1. \tag{46}$$

To find such an $\boldsymbol{\epsilon}$, we need to take a derivative of the objective function (denoted as $D(\boldsymbol{\epsilon})$) with respect to $\boldsymbol{\epsilon}$:

$$\mathcal{L}(\boldsymbol{\epsilon},\ \lambda) = \boldsymbol{\epsilon}^\top (T_1 - T_2)\, \boldsymbol{\epsilon} + \lambda(1 - \boldsymbol{\epsilon}^\top \boldsymbol{\epsilon}) \tag{47}$$

$$\frac{\partial \mathcal{L}(\boldsymbol{\epsilon})}{\partial \boldsymbol{\epsilon}} = 2\boldsymbol{\epsilon}^T (T_1 - T_2) - 2\lambda \boldsymbol{\epsilon}^\top = 0, \quad \text{and } (T_1 - T_2) \text{ is symmetric.} \tag{48}$$

$$\frac{\partial \mathcal{L}(\boldsymbol{\epsilon})}{\partial \lambda} = 1 - \boldsymbol{\epsilon}^\top \boldsymbol{\epsilon} = 0. \tag{49}$$

So any solution needs to satisfy $\boldsymbol{\epsilon}^\top (T_1 - T_2) = \lambda \boldsymbol{\epsilon}^\top$. The solution that maximizes $\boldsymbol{\epsilon}^\top (T_1 - T_2) \boldsymbol{\epsilon}$ would be the one that maximizes $\lambda \boldsymbol{\epsilon}^\top \boldsymbol{\epsilon}$. So the solution is the largest, when λ is the largest eigenvalue of matrix $(T_1 - T_2)$, and $\boldsymbol{\epsilon}$ is the first eigenvector of $(T_1 - T_2)$. The ratio setting of the optimization does not depend on the scaling of each metric tensor (size of the metric tensor ellipsoids). But the solution to the difference computation depends on the scaling of the metric tensors.

S10. Comparing between n models (n > 2)

To compare between $n > 2$ number of metric tensors, a seemingly plausible approach is to pair-wise compare between these metric tensors, and choose the one that wins the most comparisons. However, this approach does not guarantee that there is a winning model, and may produce conflicting results. For example, to compare among three metric tensors $\{M_{\mathbf{f}_1}(\mathbf{s}), M_{\mathbf{f}_2}(\mathbf{s}), M_{\mathbf{f}_3}(\mathbf{s})\}$, we find two perturbation directions for each pair of metric tensors as described in previous section. Different perturbation directions were used to compare between different mode pairs, which could result in cyclic comparisons, e.g. $\mathbf{f}_1 \succ \mathbf{f}_2$, $\mathbf{f}_2 \succ \mathbf{f}_3$, and $\mathbf{f}_3 \succ \mathbf{f}_1$.

To compare between $n > 2$ number of metric tensors, we may take a different generalization that shares the same advantage of comparing between 2 metric tensors – a *single-shot* perceptual experiment can be used to select among n model predictions. Previously, we find perturbation directions $\boldsymbol{\epsilon}$ that maximizes the ratio between two metric tensor predictions. For more than two metric tensors, we find a set of perturbation directions, each of which minimizes the ratio between one metric tensor's prediction, $\boldsymbol{\epsilon}^\top M_{\mathbf{f}_i}(\mathbf{s})\boldsymbol{\epsilon}$, and the average predictions from all other metric tensors, $1/(n-1)\sum_{k \neq i} \boldsymbol{\epsilon}^\top M_{\mathbf{f}_k}(\mathbf{s})\boldsymbol{\epsilon}$. Because the scaling factor $1/(n-1)$ does not affect the solution, we can omit it in the objective functions:

$$\text{Find} \quad \{\boldsymbol{\epsilon}_i\}_{i=1:n} \quad \text{s.t.} \quad \boldsymbol{\epsilon}_i = \arg\min_{\boldsymbol{\epsilon}} \left\{ \frac{\boldsymbol{\epsilon}^\top \left[\sum_{k\neq i} M_{\mathbf{f}_k}(\mathbf{s})\right] \boldsymbol{\epsilon}}{\boldsymbol{\epsilon}^\top M_{\mathbf{f}_i}(\mathbf{s}) \boldsymbol{\epsilon}} \right\}. \tag{50}$$

The sum of symmetric and positive definite matrices is still symmetric and positive definite, so the objective functions described above remain generalized eigenvalue problems, and the closed-form solution of $\boldsymbol{\epsilon}^{(i)}$ is the last eigenvector (that corresponds to the smallest Eigenvalue) of the matrix $M_{\mathbf{f}_i}(\mathbf{s})^{-1}\left[\sum_{k\neq i} M_{\mathbf{f}_k}(\mathbf{s})\right]$.