Neural Networks 121 (2020) 122–131

Contents lists available at ScienceDirect

Neural Networks

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

How are response properties in the middle temporal area related to inference on visual motion patterns?

Omid Rezai [a], Lucas Stoffl [b], Bryan Tripp [a,*]

[a] University of Waterloo, Canada
[b] Technical University of Munich, Germany

ARTICLE INFO

Article history:
Received 1 November 2018
Received in revised form 4 August 2019
Accepted 22 August 2019
Available online 4 September 2019

Keywords:
Middle temporal area
Representation
Deep networks
Sensitivity analysis
Representational similarity

ABSTRACT

Neurons in the primate middle temporal area (MT) respond to moving stimuli, with strong tuning for motion speed and direction. These responses have been characterized in detail, but the functional significance of these details (e.g. shapes and widths of speed tuning curves) is unclear, because they cannot be selectively manipulated. To estimate their functional significance, we used a detailed model of MT population responses as input to convolutional networks that performed sophisticated motion processing tasks (visual odometry and gesture recognition). We manipulated the distributions of speed and direction tuning widths, and studied the effects on task performance. We also studied performance with random linear mixtures of the responses, and with responses that had the same representational dissimilarity as the model populations, but were otherwise randomized. The width of speed and direction tuning both affected task performance, despite the networks having been optimized individually for each tuning variation, but the specific effects were different in each task. Random linear mixing improved performance of the odometry task, but not the gesture recognition task. Randomizing the responses while maintaining representational dissimilarity resulted in poor odometry performance. In summary, despite full optimization of the deep networks in each case, each manipulation of the representation affected performance of sophisticated visual tasks. Representation properties such as tuning width and representational similarity have been studied extensively from other perspectives, but this work provides new insight into their possible roles in sophisticated visual inference.

1. Introduction

The visual responses of neurons in different areas of the visual cortex have been studied extensively, providing a detailed view of many relationships between visual representations and stimulus properties. Complementing these experiments, information theory has been used to understand how tuning curve widths affect the encoding of stimulus information by neuron populations (Zhang & Sejnowski, 1999). However, the significance of response properties with respect to the outputs (rather than the inputs) of the visual cortex has been less studied.

In the primate middle temporal area (MT), many neurons respond strongly to visual motion, with robust tuning for motion speed and direction. Tuning curves and other response properties have been extensively characterized. Microstimulation studies have confirmed the role of MT cells in motion perception; microstimulation biases animals' judgements towards the direction of motion encoded by the stimulated neurons (Salzman, Britten, & Newsome, 1990; Salzman, Murasugi, Britten, & Newsome, 1992). Furthermore, lesion studies in monkeys have confirmed the role of MT in smooth pursuit eye movements (Newsome, Wurtz, Dürsteler, & Mikami, 1985). Trial-to-trial variability in MT responses is also correlated with motion perception decisions (Smith, Chang'an, & Cook, 2011). Furthermore, tuning properties have been linked with perception and visually guided action. For example, preferred speeds of MT neurons are slower for smaller stimuli, which accounts for human perception of smaller stimuli as moving faster (Boyraz & Treue, 2011). Other stimulus manipulations affect pursuit eye movements in a way that is consistent with their effects on MT neuron tuning (Lisberger, 2010).

However, a potential limitation of these studies is that they involve decoding the same low-level variables that form the domain of the tuning curves, whereas much more complex inferences can also be made from visual motion patterns (e.g. Johansson, 1973; Warren & Rushton, 2009). MT projects strongly to several other cortical areas (Markov et al., 2014), suggesting that MT representations may have a variety of roles in perception and visually guided behaviour. The significance of MT

[*] Corresponding author.
E-mail address: bptripp@uwaterloo.ca (B. Tripp).

https://doi.org/10.1016/j.neunet.2019.08.027
0893-6080/© 2019 Elsevier Ltd. All rights reserved.

tuning widths, and other properties of the representation, with respect to complex visual inferences is unclear.

In this study, we embedded models of MT activity within models that performed sophisticated inference, to estimate the potential contributions of MT representation properties in such tasks. Specifically, we embedded an MT model within convolutional networks that perform visual odometry (i.e. egomotion from video) and gesture recognition. We then varied properties of the representation to estimate the relevance of these properties to sophisticated motion processing. The results in Fig. 6 have been presented previously (Rezai, Boyraz Jentsch, & Tripp, 2018).

One property of the representations that we varied was tuning curve widths. A tuning curve describes a neuron's mean spike rate as a function of some experimental variable. Although electrophysiology experiments typically measure tuning curves in single dimensions (for practical reasons), individual neurons in a given area are typically sensitive to multiple stimulus dimensions (DeAngelis & Uka, 2003). Tuning curves have been extensively measured in neurophysiology for many decades, but new details and insights continue to emerge, e.g. related to their dynamics (Ringach, Hawken, & Shapley, 1997), statistics (Wang & Movshon, 2016), and modulation by attention (Treue, 2001). Their significance has also been widely studied in theoretical work, often from the perspective of their effect on the amount of stimulus information encoded by a population of noisy neurons (Butts & Goldman, 2006; Zhang & Sejnowski, 1999). However, this perspective may not completely address the functional significance of tuning curves in the brain. Other theoretical work deals more directly with the use of tuning curves as a basis for computation rather than stimulus reconstruction. In particular, Eliasmith and Anderson (2003) showed that different sets of tuning curves support robust computation of different functions (via multi-linear regression). This gives additional insight into the roles of tuning curves in supporting feature transformations. However, multilinear regression is a simplified model of computation in a single connection from one population to another, rather than computation in a more complex network. In this study, we extend this view by studying the effects of tuning width in deeper networks that compute relatively complex and naturalistic functions.

Our results suggest that tuning width is functionally significant, even in deeper networks that perform complex tasks, and that the optimal tuning curves are task-dependent. In separate simulations, we also found that representational similarity (Kriegeskorte, 2009) does not fully account for the functional significance of neural responses. In each case, the networks were retrained on the modified representations, but the representation details affected task performance.

2. Methods

2.1. Model of population activity in the middle temporal area

We used our previous empirical model of area MT population responses (Rezai et al., 2018) (see Fig. 1).

Briefly, the model uses computer-vision methods to calculate various fields from video input, including optic flow, disparity, and local contrast fields. A number of feature maps are created from these fields, corresponding to different stimulus tuning. The number of feature maps varies somewhat between networks, as described below. For example, a specific instance of the MT model might have 64 13 × 13-pixel feature maps, in which case each feature map would model 13^2 MT neurons with the same feature selectivity, tiled over visual space. To create each feature map, we combined tuning curves for speed, direction, disparity, etc. from the literature. The tuning curves were calculated as pixel-wise functions of the flow, disparity, and contrast fields. Specifically, the neurons' responses were $r = [A \int_x \int_y K_{x,y} g_s g_\theta g_d + B]_+$, where $[]+$ denotes half-wave rectification, B is the background firing rate (spikes/s), A = maximum firing rate − background firing rate, $K_{x,y}$ is the spatial receptive field, g_s is a speed-tuning function that is also a function of local contrast, g_θ is a direction-tuning function, and g_d is a binocular-disparity tuning function. Our original model also included an attention component, which we omitted here. Of particular interest in the present study are the speed and direction tuning functions. The speed-tuning function (from Nover, Anderson, & DeAngelis, 2005) is,

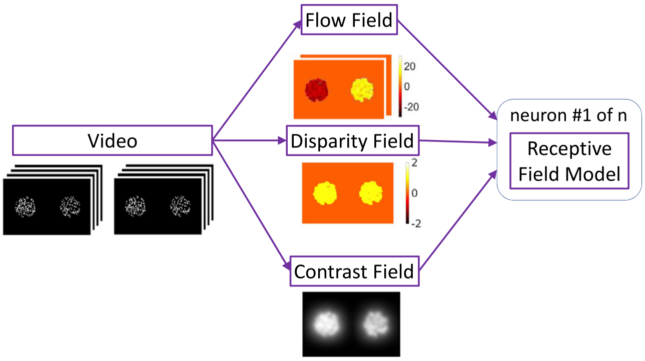

Fig. 1. Overview of the MT model. The model uses computer-vision methods to calculate flow, disparity, and contrast fields, and tuning curves from the primate electrophysiology literature to estimate the MT population response from these fields.
Source: Adapted with permission from Rezai et al. (2018).

$$g_s = \exp\left(-\frac{[\log(q(s,c))]^2}{2\sigma_s^2}\right), \tag{1}$$

where,

$$q(s,c) = \frac{s + s_0}{s_p(c) + s_0}, \tag{2}$$

s is motion speed, and s_p is the preferred speed. The tuning curve has parameters s_0 (offset) and σ_s (width). Preferred speed is a function of contrast c (Rezai et al., 2018). The direction-tuning function is

$$g_\theta = \exp\left(\frac{\cos(\theta - \theta_p) - 1}{\sigma_\theta}\right) + a_n \exp\left(\frac{\cos(\theta - \theta_p - \pi) - 1}{\sigma_\theta}\right), \tag{3}$$

where θ is motion direction, θ_p, σ_θ, and a_n are the preferred direction, direction width, and relative amplitude in null direction (i.e. 180 degrees away from preferred direction), respectively.

Spatial receptive fields were then modelled by combining responses across pixels, using difference-of-Gaussians kernels. For each feature map, we drew tuning-curve parameters from distributions that were modelled on data from the electrophysiology literature. The model reproduces some MT response properties that have not appeared in previous models (e.g. local rather than global pattern-motion integration within a receptive field; Majaj, Carandini, & Movshon, 2007), and generally reproduces MT response properties more closely than previous models. It approximates dynamics of component and pattern selectivity, but this aspect of the model was omitted in this study. In summary, the model produces an approximation of an MT population response (in spikes/s) to video input, in the same form as a multi-channel convolutional-network layer.

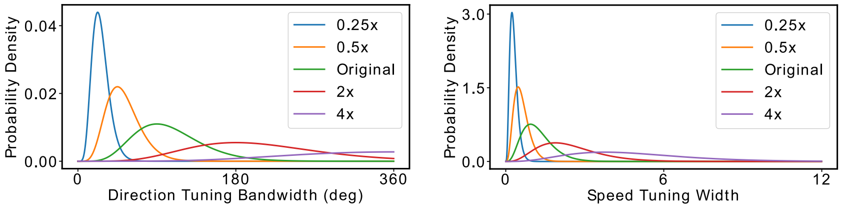

Fig. 2. Left: Gamma distributions for drawing direction tuning bandwidths where the parameters of the original distribution were shape = 7.32 and scale = 14.20. Note that the direction tuning bandwidths were truncated at 360°. Right: Gamma distributions for drawing speed tuning widths where the parameters of the original distribution were shape = 4.36 and scale = 0.28.

3. Perturbations of the MT model representation

Using the MT population model as a baseline representation of visual motion, we explored several variations of this representation, described below. We trained multiple deep networks independently, using each of these variations as input.

3.0.1. *Changes in speed and direction tuning width*

The issue of optimal neuronal tuning widths has received much attention in the literature (Brown & Bäcker, 2006), particularly around whether sharp (Barlow, 1972; Zhang & Sejnowski, 1999) or broad (Baldi & Heiligenberg, 1988; Eurich & Schwegler, 1997; Georgopoulos, Schwartz, & Kettner, 1986; Hinton, McClelland, Rumelhart, et al., 1984) tuning curves increase encoding accuracy. In contrast, the present study addresses the role of tuning width in complex and naturalistic visual inference.

We focused on two features of MT response, the direction-tuning bandwidth and speed-tuning width. A large percentage of MT neurons are sensitive to both direction and speed. In the MT model, the widths of both speed and direction-tuning curves are drawn from Gamma distributions. These are a family of continuous distributions over $[0, \infty)$, which include exponential distributions as a special case (see Lehky, Kiani, Esteky, and Tanaka (2011) for an example of the use of Gamma distributions in the neuroscience literature). Previously, we found that Gamma distributions fit both MT speed and direction-tuning width histograms from the electrophysiology literature better than a number of other common distributions, according to the Akaike Information Criterion (Rezai et al., 2018). Gamma distributions have two parameters, the shape and the scale. We experimented with variations in tuning-curve widths by changing the scale parameters. Specifically, we experimented with 0.25, 0.5, 1, 2, and 4 times the original scale. Fig. 2 depicts these Gamma distributions for both direction tuning bandwidths and speed tuning widths. For direction tuning, we truncated the distributions at 360°. We also experimented with eliminating each of these tuning dimensions entirely.

3.0.2. *Random linear recombinations*

Several studies (e.g. Schrimpf et al., 2018) have used linear regression to approximate neural responses from model responses. In these studies, the quality of linear reconstruction is taken to reflect the similarity of the model and neural representation. The rationale is that correspondences between individual model and biological neurons cannot be expected, but a given biological neuron response should resemble some linear combination of model neuron responses, if the model responses belong to the same family. Random linear mixing of responses has little effect on linear reconstruction, but we wondered whether it could affect performance of complex tasks. To test this, we passed MT population responses through random 1 × 1 kernels before input to the odometry and gesture networks. Tuning of these random combinations was qualitatively different than tuning of MT model neurons. For example, tuning for speed and velocity was not separable (see Fig. 3). Because we used 1 × 1 kernels, the responses remained spatially localized, and were only mixed in feature space.

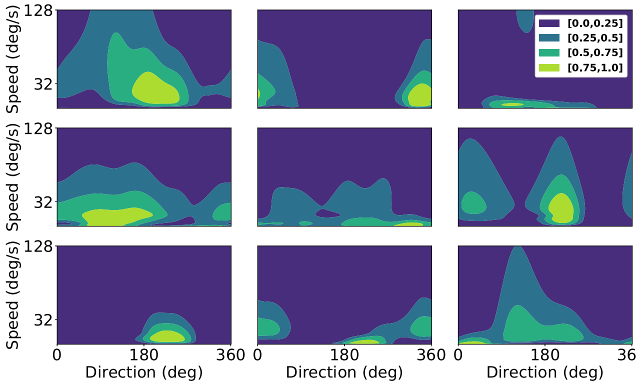

Fig. 3. Examples of normalized speed vs. direction-tuning curves for nine random linear recombinations of MT model responses. In contrast with the tuning of the MT model neurons, these mixed responses are not separable in speed and direction.

3.0.3. *Random responses with given representational similarity*

Representational similarity analysis (RSA) (Kriegeskorte, 2008) is widely used to characterize and compare neural representations. It consists of calculating a representational dissimilarity matrix (RDM), which is typically simply one minus the matrix of correlations between population responses to different stimuli, i.e.

$$RDM = 1 - R, \tag{4}$$

where R is the correlation matrix. If two stimuli evoke highly correlated population responses, this suggests that the recorded population makes little distinction between them. RSA allows comparison of different representation modalities, such as electrophysiology data, functional imaging data, and model data. For example, if RDMs of a model and an electrophysiology dataset are similar, this suggests that the model and the recorded neuron population make similar distinctions between the stimuli. RSA has been used to compare neurobiological representations with representations in deep networks (Cadieu et al., 2014; Khaligh-Razavi & Kriegeskorte, 2014; Vries et al., 2018).

Visual representations can be viewed as intermediate processing steps toward visual perception or visually guided action. In this context, we wondered how much the RDM of a representation determines how useful the representation is for certain visual tasks. To test this, we created population responses matrices with RDMs that closely matched those of our MT model, but which were otherwise random. We then trained odometry networks using these RDM-matched responses as input, and compared their performance to odometry networks with the actual MT model as input. This process required the RDM of a model population, over a full dataset. To make this tractable, we used a smaller MT population (10,816 units in total) and a subset of the odometry dataset (18,000 training and 2048 validation sequences). The RDMs were therefore 20,048 × 20,048. A dataset of similar size for the gesture recognition task yielded poor performance (36% classification accuracy), so we did not analyse the gesture task in this way. These RDMs were exact, because they were based on the responses of the entire MT model, whereas only a sample of the relevant population is available in an electrophysiology study.

We experimented with two methods of generating RDM-matching population response matrices (10,816 by 20,048 entries). First, we began with the original MT-model responses, and changed population response vectors for individual stimuli repeatedly by small amounts. Each change of a population response to a given stimulus was within the null space of the gradients of the correlations with responses to other stimuli. This approach did not work well however, apparently due to accumulation of numerical errors. The step directions were uncorrelated (as the null space changed at each step), so they accumulated poorly, and many steps were needed to make substantial changes.

Accurate results were reliably obtained with a different method. We began with a random response matrix R^R, and defined the cost function $C = \sum_{i,j} r(R^R_{i,:}, R^R_{j,:}) - r(R^{MT}_{i,:}, R^{MT}_{j,:})$, where R^{MT} is the MT-model response matrix, and r is the correlation coefficient. We then calculated the gradient of C with respect to the elements of R^R, and minimized the cost using the Adam algorithm (Kingma & Ba, 2014), an adaptive variant of gradient-descent. We optimized each response matrix by optimizing 1000 random sub-matrices, each consisting of 300 stimuli. This reliably resulted in close matches between R^{MT} and the optimized R^R. To reiterate, once we had calculated R^{MT}, the RDM of an MT model's responses, this procedure allowed us to create new, random responses with the same RDM. This procedure produced new (random, but RDM-matched) responses for a set of 20,048 stimuli, which was large enough for training and validation of an odometry network with the new responses as inputs. This in turn let us assess whether fixing the RDM determined task performance, or whether other aspects of the responses (which were randomized by this procedure) were also important.

Fig. 4 illustrates some differences between linear mixing and RDM matching in a simple one-dimensional example. In general, a matching RDM does not imply good linear reconstructions of the original responses, and good linear reconstructions do not imply matching RDM. Experimenting with other simplified population models, consisting of Gaussian tuning on vector fields of various dimensions, we found that RDM could be closely matched with a wide variety of populations. However, linear reconstruction of held-out samples from the RDM-matching responses was generally poor, although it tended to improve somewhat with wider tuning and larger receptive fields.

3.1. Visual tasks and deep networks

MT model activity was used as input to convolutional networks that performed sophisticated visual tasks. We tested how the above perturbations of the MT-model representation affected performance of two sophisticated visual motion-processing tasks, a visual odometry task and a gesture recognition task. In each case, the networks only received motion, disparity, and contrast information. However, other information is useful in these tasks as well, for example some gestures can be recognized from still images. Our networks were therefore somewhat impaired in these tasks, in order to isolate the role of visual motion representation.

The goal of the visual odometry task was to estimate self-motion velocity from video. We used a photorealistic synthetic dataset that we had developed previously (Rezai et al., 2018). The dataset is well suited to provide input to the MT model, as it has stereo video with a biologically realistic stereo baseline, and a high frame rate. It is also large enough for supervised learning with deep networks. The dataset was created in Unreal Engine 4, using the Modular Neighbourhood Pack, which contains a model of a residential neighbourhood with houses, cars, and streets, surrounded by a natural landscape of grass and trees (see example frame in Fig. 5). The dataset has 84,000 short six-frame stereo videos in which the camera moves along curvilinear paths. For each video, it has ground-truth antero-posterior, medio-lateral, and rotational velocities. We used 75,000 sequences for training and held out 9000 additional sequences for validation.

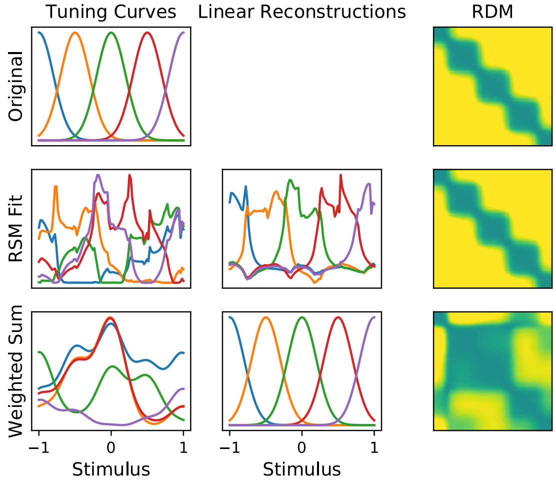

Fig. 4. Illustration of differences between RDM matching of responses and linear recombination of responses with a small one-dimensional population model. Tuning curves of the original model responses are shown in the top-left panel, and the RDM is shown at the top right. Yellow indicates high dissimilarity. The axes correspond to the same range of 1D stimulus values as the horizontal axis in the top-left panel, in the same order. Middle row: responses (left) that have essentially the same RDM (right) lead to poor linear reconstruction (centre). Bottom row: random linear mix of original responses (left). These lead to good linear reconstruction of the original responses (centre), and non-matching RDM (right). (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

Fig. 5. An example stereo video frame from the odometry dataset.

The network structure used for the odometry task is shown in Table 1. The MT model responses provided input to the network. As described above, these were calculated from video. The flow, disparity, and contrast fields were averaged over the six frames of each video, approximating the low-pass properties of MT neurons (Bair & Koch, 1996). Batch normalization was also used after all layers of the network (except the output layer) to reduce overfitting and speed up training. The CNN was implemented in Keras (Chollet, 2015) with TensorFlow (Abadi et al., 2016) back end. The mean-square error of self-motion estimates was minimized using the Adam algorithm (Kingma & Ba, 2014).

The gesture recognition task was based on the 20BN-JESTER dataset, which was developed by TwentyBN (Toronto, Canada). This dataset consists of about 150,000 short videos sequences in

Table 1
CNN architecture for the visual odometry task. The structure was based very roughly on the structure of the primate dorsal visual stream, with area MT corresponding to the input, and the two convolutional layers corresponding respectively to the middle superior temporal area and the ventral intraparietal area, which has been linked to coding of heading direction (Bremmer, 2005).

Layer	# Kernels	Kernel size	Shape	Pool	Nonlinearity
Conv-1	128	9 × 9	6 × 6	None	ReLU
Conv-2	128	9 × 9	6 × 6	2 × 2	ReLU
Dense			1024		ReLU
Output			3		None

which people perform hand gestures from 25 different categories (e.g. thumbs-up, swipe left).

The network structure used for the gesture recognition task is shown in Table 2. Since the frame rate of the 20BN-JESTER dataset was already comparable to the temporal range of MT, unlike the higher frame rate of the odometry dataset, we did not feed the sequence-average as input to the gesture networks. Instead, we chose a twelve-frame window from each sequence where the average flow was maximum compared to any other window. Therefore, the most motion-informative part of the sequence was captured while keeping the input sequence small enough so a mini-batch could be fit on GPU memory during training. Because the input was a sequence we used a long-short-term-memory (LSTM) layer instead of a dense layer after the final convolutional layer.

When we used the simplified difference-of-Gaussians (DoGs) kernels as the receptive fields (RFs) of the MT neurons, the gesture networks overfitted after 3 or 4 epochs with high validation loss. Therefore, instead of using DoGs, we added three parallel sparse convolutional layers to the beginning of these networks. The networks received pixel-wise non-linear functions of flow and contrast fields as input. These functions, which we refer to as tuning feature maps, were calculated using our MT model. The sparsity of the three parallel layers meant that each channel of the MT layer (Table 2) almost exclusively connected to one of the tuning feature maps through each parallel layer. In other words, each channel of MT layer was connected to three tuning feature maps via three kernels that corresponded, respectively, to the classical centre RF, direction-selective surround and non-direction-selective surround (Cui, Liu, Khawaja, Pack, & Butts, 2013). These kernels were learned during the training phase and constrained to be either non-negative (centre RFs) or non-positive (surrounds).

To create random linear mixing of MT responses on the gesture recognition task, we added a convolutional layer with 1 × 1 kernels right after the MT layer (Table 2). These 1 × 1 kernels were randomly initialized and not allowed to change during training.

These network models lack many physiological details, such as spiking and lateral interactions. It is not practical to avoid this limitation, because most of the missing physiological details have not been incorporated into functionally sophisticated models.

3.2. Fisher Information and optimal linear estimation

Past work has considered the significance of tuning-curve width in terms of information theory. The Fisher information is the inverse of the least possible variance of an unbiased estimator (Dayan & Abbott, 2001). If neurons exhibit independent Poisson variability, the Fisher information is (Dayan & Abbott, 2001),

$$I_f = T \sum_{i=1}^{N} \frac{(r_i'(s))^2}{r_i(s)}, \tag{5}$$

Table 2
CNN architecture for the gesture recognition task. There were three parallel sparse convolutional layers (RF-1, RF-2, RF-3) in the network that constituted the centre and surround RFs of MT layer (see text). MT layer had 64 channels where the activity was computed by adding the output of RF-1, RF-2 and RF-3, as well as 64 bias values, and passing the result through the rectified linear units (ReLUs). Compared to the odometry network, this network replaces the Dense layer before the output with an additional convolutional layer and a LSTM layer, which was important for integrating information over larger numbers of frames. This network incorporated a more realistic model of MT receptive fields, based on Cui et al. (2013), which improved performance in this task.

Layer	# Kernels	Kernel size	Shape	Pool	Nonlinearity
RF-1	64	15 × 15	12 × 76 × 76	None	None
RF-2	64	15 × 15	12 × 76 × 76	None	None
RF-3	64	15 × 15	12 × 76 × 76	None	None
MT			12 × 76 × 76	None	ReLU
Conv-1	64	15 × 15	12 × 76 × 76	6 × 6	ReLU
Conv-2	64	9 × 9	12 × 12 × 12	None	ReLU
Conv-3	64	9 × 9	12 × 12 × 12	3 × 3	ReLU
LSTM			256		ReLU
Output			27		Softmax

where $r_i(s)$ is the ith tuning curve, N is the number of neurons, and T is the time window (Poisson noise is independent over time, so information accumulates over time). We calculate the Fisher information of our MT population models to contrast it with task performance, as it is unclear how these are related. Fisher information is related to recovering a stimulus property, and in the standard model we use, it is assumed that the main barrier is independent Poisson noise in each neuron. In contrast, task performance in our network relies on inference, and the key barrier is reliance on visual cues that may be subtle and variable. In this context, independent neuron-level noise (such as Dropout) may mildly degrade performance, but it can also play an important regularizing role. In our models, errors in the estimation of motion speed and direction are potentially a more problematic source of noise that is correlated across all the neurons. However, this noise of noise is injected before the tuning curves, so their shapes do not affect sensitivity to it.

Others (Eliasmith & Anderson, 2003) have studied the effect of tuning curve width on the accuracy of optimal linear estimates (Salinas & Abbott, 1994) of a represented variable,

$$\hat{s}_{OLE} = (R^T R)^{-1} R^T s, \tag{6}$$

where R is a matrix of responses for different neurons and stimuli) and also of optimal linear estimation of different functions of a represented variable,

$$\hat{f}(s)_{OLE} = (R^T R)^{-1} R^T f(s). \tag{7}$$

The latter is related to our deep network models, in that the first convolutional layer after the MT model performs a linear mapping that extracts some unknown function of the represented variables. To relate our models to this past work, we test the accuracy of optimal linear estimates of speed and direction from our MT models. For the best-performing MT model populations in each task, we also plot the principal components of the speed-direction tuning curves, which span the space of functions that can be computed robustly, i.e. with low sensitivity to noise.

The Fisher information and optimal linear estimate analyses used a slightly simplified model where we removed the contrast dependency from speed tuning. Specifically, in this simplified version preferred speeds were drawn from a log uniform distribution that we had modelled based on Nover et al. (2005) (as opposed to calculating the preferred speeds as a function of contrast).

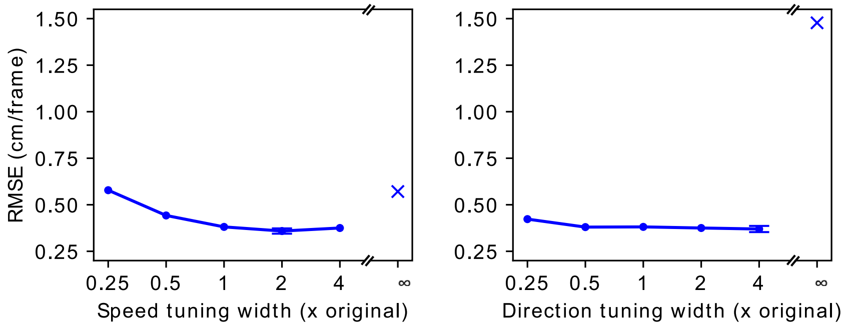

Fig. 6. Root-mean-square error of odometry predictions on held-out validation data, as a function of speed tuning width (left) and direction tuning width (right). The standard deviation of the targets was 1.54. On the horizontal axis, 1 means the distribution is unchanged, 2 means the scale parameter of the Gamma distribution is 2 × its original value, etc. The error bars indicate +/− 2SD for repeated training with different random initialization.

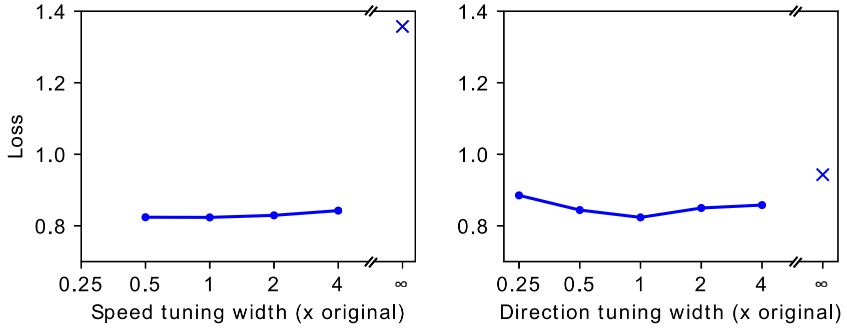

Fig. 7. Validation loss of gesture recognition networks as a function of speed-tuning width (left) and direction-tuning width (right). The horizontal axis is as in Fig. 6. For example, 1 indicates the original speed-tuning width distribution, and 2 indicates that the scale parameter of the distribution was increased by a factor of two. The points on the right (marked with ×) indicate infinite tuning width (i.e. no sensitivity).

4. Results

4.1. Sensitivity of task performance to speed and direction tuning width

Fig. 6 shows how performance of the visual odometry task is affected by the distributions of speed and direction tuning widths. Importantly, the networks were trained independently with each set of tuning curves. The ∞ symbol refers to not having any selectivity for either speed (left panel) or direction (right panel) in the model (i.e., bandwidth is infinite). The root-mean-square error (RMSE) was 53% higher when speed-tuning widths were narrowed to 0.25 times their original range, and 6% lower when they were increased to twice their original range. Eliminating speed tuning completely (the point on the right of the plot) resulted in 50% higher error than the original model.

In contrast, RMSE was only 11% higher when direction-tuning widths were narrowed to 0.25 times their normal range, and 3% lower at best (4 × normal width). However, RMSE was 289% higher when direction tuning was eliminated (1.48, similar to the standard deviation of the targets, which was 1.54).

We tested whether performance improvements with broader tuning were significant, using t-tests with Bonferroni correction for multiple comparisons. To improve power, we created and trained three additional networks with new MT population models that had the best-performing speed and direction-tuning widths (same tuning distributions; different random samples from these distributions). Mean absolute errors with 2 × speed tuning widths were significantly lower than all other cases ($\alpha <$ .05). The average RMSE of the 4 × direction-tuning width populations was lower than other cases, although the 0.5 ×, 1 ×, and 2 × means differed by less than five percent. Among the 0.25 ×, 0.5 ×, 1 ×, and 2 × direction-tuning variations, only the 0.25 × and 0.5 × had significantly higher mean absolute errors ($\alpha <$ .05) than the 4 × errors.

Fig. 7 shows how the distributions of speed and direction tuning widths affect performance of the gesture recognition task. The loss increased moderately when tuning widths were increased to 4x their original range. The loss increased more substantially when either speed or direction tuning were eliminated. Classification accuracy dropped from 75% to 60% when speed tuning was eliminated, and to 70% when direction tuning was eliminated. These results (both loss and classification accuracy) are in different units than the odometry results, but qualitative comparisons are possible. In contrast with odometry, broader tuning did not improve gesture recognition performance. Elimination of direction tuning had a larger impact on odometry performance, while elimination of speed tuning had a larger impact on gesture recognition performance.

4.2. Sensitivity of Fisher information to tuning width

Fig. 8 plots Fisher information (assuming independent Poisson variability in each neuron) for the MT layers of our models. Fisher information declines monotonically with an increase of tuning curve width in a single dimension. This is qualitatively consistent with the monotonic increase in error with speed tuning width that we found in the gesture recognition task. However, it is inconsistent with the other task effects. Specifically, odometry performance is best with broad direction tuning, and the other relationships are non-monotonic. Taken together, the effects of tuning width on Fisher information and task performance have little in common.

Notably, in addition to decreasing information about speed, increasing speed-tuning width also indirectly increases informa-

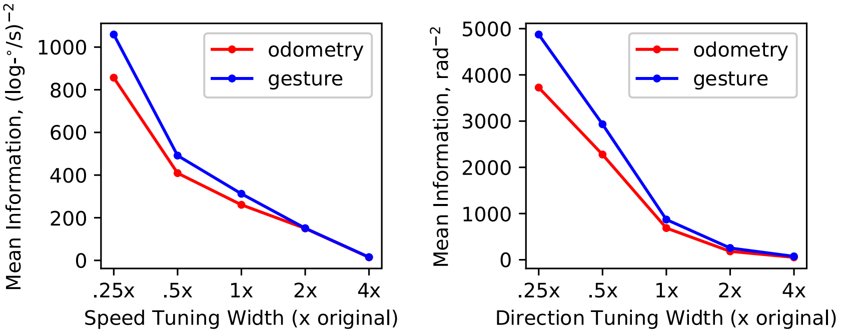

Fig. 8. Left, Fisher information about log-speed, for MT model populations with different speed-tuning widths. The red and blue lines correspond to the odometry and gesture recognition tasks, respectively. These are slightly different because the Fisher information is calculated as a weighted average over the actual speed distributions that appeared in these tasks. The task-specific direction distributions were ignored, because direction tuning was statistically uniform. Right, Fisher information about direction, for MT model populations with different direction-tuning widths. (For interpretation of the references to colour in this figure legend, the reader is referred to the web version of this article.)

tion about direction, simply because wider tuning curves increase mean spike rates. For this reason, task performance as a function of speed-tuning width could potentially be well correlated with a linear combination of Fisher information about speed and direction. However, this would be coincidental, because Fisher information depends on independent noise around the tuning curves, whereas our model responses have no such noise source. Many deep networks use Dropout, which is a kind of independent noise, but this is typically turned off at inference time.

4.3. *Effect of tuning width on optimal linear estimation*

Previous work has also examined the effects of tuning-curve shapes on optimal linear decoding of stimulus properties in the presence of noise. Deep networks are more powerful than linear estimators, but each of their layers includes a linear map, so the effects of tuning-curve shapes on optimal linear decoding could conceivably be related to task effects. However, decoding of both log-speed and direction was quite accurate for all of the populations. For example, using the Moore–Penrose pseudoinverse, with a regularization parameter equivalent to additive Gaussian noise of five spikes/s, root-mean-squared log-speed decoding error was $< .01$ log-○/s for all populations, and root-mean-squared direction decoding error was $< .001$ radians for all populations.

Importantly, the linear maps in our deep networks do not explicitly decode direction and velocity. However, they do compute new visual features that are functions of direction and velocity. Tuning-curve width affects the functions that can be robustly computed from a neural population (Eliasmith & Anderson, 2003). In particular, the functions that can be computed with the least sensitivity to noise are in the space of the large principal components of the tuning curves, corresponding to large singular values a matrix with tuning curves as rows. Fig. 9 plots singular values of these matrices, with different perturbations of tuning-curve width. Wider tuning curves produce a few large principal components, corresponding to a small space of functions that can be decoded very robustly. Fig. 10 shows the largest principal components of the best-performing populations for each task. The best principal components for the odometry task (left) tend to be fairly separable in speed and direction, whereas the best principal components for gesture recognition (right) are somewhat more complex functions of speed and direction.

4.4. *Sensitivity to linear recombination and RDM-maintaining perturbations*

Fig. 11 shows an example of random responses with a RDM that closely matches that of an MT-model response to the odometry dataset.

Table 3
Task performance with linear mixing of the MT-model responses, and random responses with the same RDM as the MT model. The RDM matching procedure was not performed with the gesture task due to the large size of the required correlation matrix.

	Odometry RMSE (mean +/− SD cm/frame)	Gesture Correct Classification (mean +/− SD %)
Baseline	0.45 +/− 0.00	75.82 +/− 0.25
Linear Recombination	0.42 +/− 0.01	75.00 +/− 0.19
RDM Match	0.66 +/− 0.00	–
Spatially shuffled	0.51 +/− 0.03	–

Table 3 shows the results of experiments with random linear mixing of the MT-model responses, and with randomized RDM-matching responses (root-mean-square difference with original RDM less than 0.0086 in each case). The means and standard deviations are over three independently trained networks in each case. With random linear mixing, odometry performance improved, but gesture performance was slightly worse. Using random responses with the same RDM led to substantially worse performance of the odometry task.

The RDM is insensitive to the spatial organization of the representation, but spatial organization could be an important factor in both deep networks and the brain, because individual neurons tend to receive spatially localized input. In our network, the kernels of the MST layer were fairly large, but did not span the whole MT layer. So the worse performance we found using random responses with the same RDM could have been due to loss of spatial organization of the representation. To test this, we performed a control experiment in which the network was trained on a spatially shuffled version of the MT model representation (i.e. each multi-channel pixel was moved to a new random location). This resulted in slightly worse performance than baseline (RMSE 0.51 cm/frame as opposed to 0.45 cm/frame), but much better performance than the full RDM-maintaining randomization (RMSE 0.66 cm/frame). This suggests that reduced performance in the latter case is not primarily due to a simple loss of spatial organization.

5. Discussion

Tuning-curve width affects how much stimulus information is encoded in the presence of noise (Zhang & Sejnowski, 1999). It also affects the functions that can be decoded from a population with diverse tuning (Eliasmith & Anderson, 2003). Using linear regression (a simple model of synaptic integration), lower-frequency functions can be decoded more accurately from wider

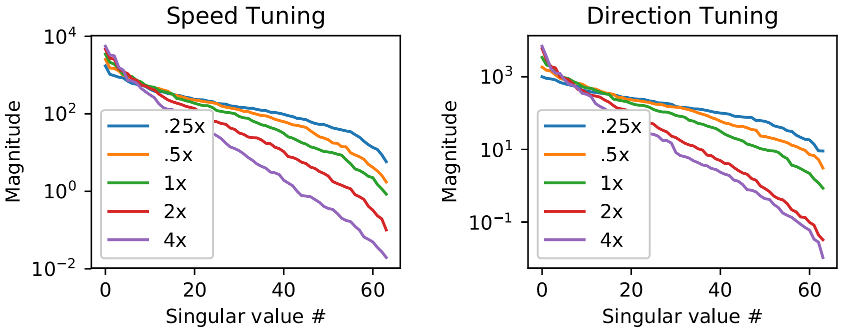

Fig. 9. Singular values of matrices of MT population model tuning curves. Left, Singular values of populations with different speed-tuning widths. Right, Singular values of populations with different direction-tuning widths.

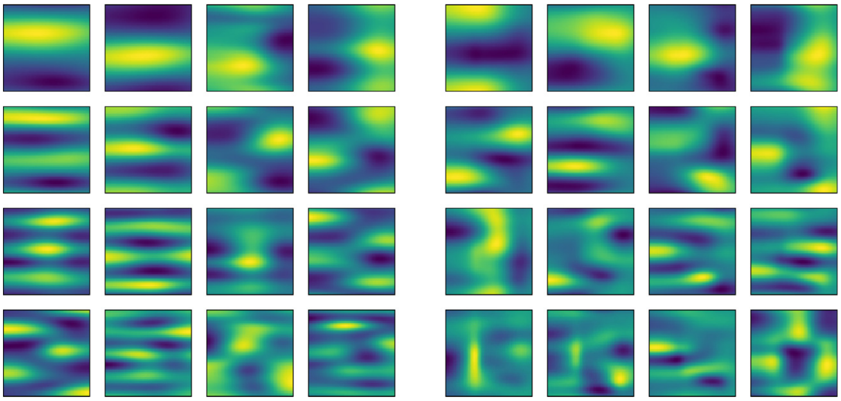

Fig. 10. Left, The first 16 principal components of the tuning curves of the best-performing MT model population for the odometry task (beginning from the top-left). The horizontal axes are log-speed, from −2 to 4 log-∘/s; the vertical axes are direction, from 0 to 2π radians. Right, As on the left, but for the best-performing model population for gesture recognition.

tuning curves, and higher-frequency functions can be decoded more accurately from narrower tuning curves. In the naturalistic scenarios studied here, the networks do not explicitly decode functions that are easily expressed in terms of their spatial frequency with respect to stimulus properties. However, we found that broader tuning was beneficial for visual odometry, but not for gesture recognition, which may reflect implicit decoding of lower frequency and higher-frequency functions at certain stages of these two networks, respectively. The fact that tuning width had different effects on different visual tasks in this study supports the idea that optimal tuning properties in MT may reflect a compromise between different functional roles (Tadin, 2015).

Random linear mixing of the tuning curves improved performance on the odometry task, but not the gesture task. This is consistent with the tuning width results, as linear mixing generally increased the effective tuning curve widths. However, additionally, the mixed tuning curves were non-separable in the speed and direction dimensions. It is notable that this did not substantially impair performance in either task.

We also optimized random responses to match the representational similarity of the MT model, to test whether representational similarity alone could account for task performance. However, this was not the case in our experiment. The RDM-matched random responses resulted in distinctly worse performance of the odometry task than the original responses.

It is somewhat surprising that performance was affected at all by these manipulations, because the network parameters were optimized individually around each representation. Deep networks can perform both of these tasks with video (rather than an MT representation) as input. We previously trained deep networks on the odometry task, with several frames of video as input (Rezai et al., 2018). We found that intermediate layers of high-performance networks exhibited speed and direction tuning, however the tuning statistics were quite different from those of area MT. In general, deep networks are fairly robust to large differences in representation. However, we found here that details of the representation of visual motion can affect their ability to perform complex tasks. There may be a manifold of high-performing representations that allows for certain large differences but not others.

The details of motion representation in MT have been studied extensively in electrophysiology experiments. However, details such as the distribution of tuning widths are difficult to manipulate independently in experiments, so their relationship with visual function must be studied in models. Models have previously been used to study the impact of tuning on motion velocity estimation (Boyraz & Treue, 2011) and smooth pursuit (Lisberger, 2010). Here we have extended this line of work, in models that perform two sophisticated and naturalistic motion-processing tasks with reasonable accuracy.

A limitation of our study is that the responses of our MT model probably differ substantially from those of real MT. Among MT models, ours addresses a relatively thorough list of MT response phenomena. It closely reproduces stimulus-parameter tuning from the literature, and at the population level, it incorporates a number of distributions of tuning properties from the literature into the population response (Rezai et al., 2018). However, the model assumes that MT neurons are completely insensitive to other stimulus parameters, except insofar as they cause errors in the estimation of velocity, disparity, and contrast fields. Despite limitations of any particular model, the question of the functional significance of tuning properties can only be addressed with a model, because tuning properties cannot be individually manipulated in animals.

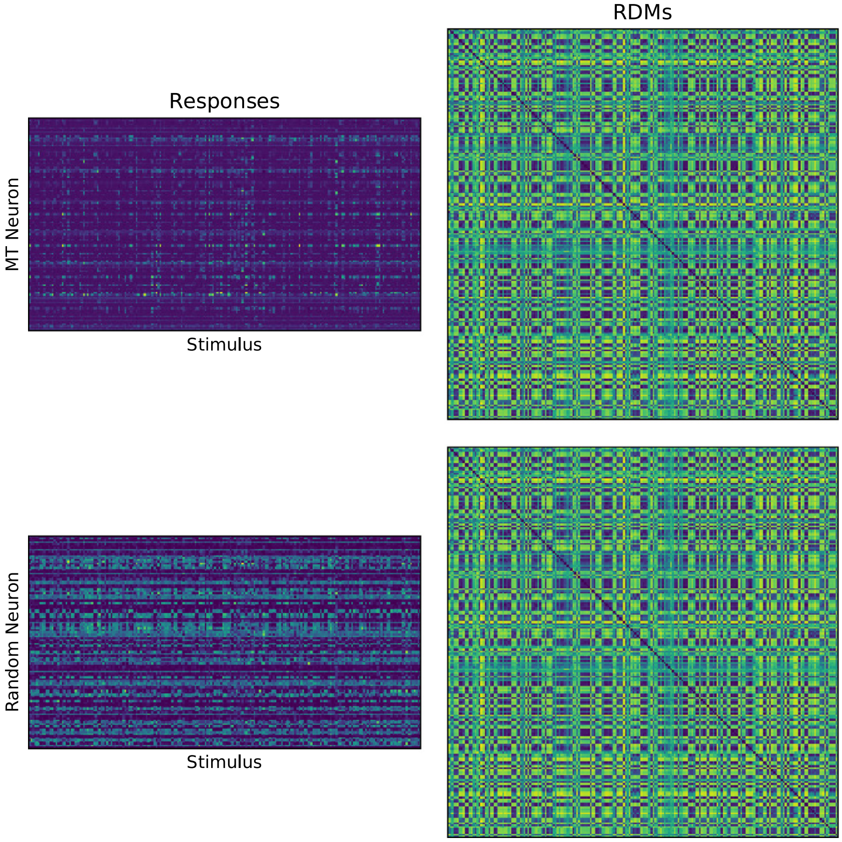

Fig. 11. An example of a random population response to the small version of the odometry dataset, that is optimized to closely match the RDM of an MT-model response to this dataset. Spike-rate responses are plotted on the left, and RDMs on the right. The top row is from the MT model and the bottom row is randomized. The root-mean-square difference between the RDMs is 0.0085. The full matrices are very large, so only every 100th neuron and stimulus are plotted.

Despite these limitations, our study suggests that tuning width is relevant to sophisticated visual inference, that the optimal widths are task-dependent, and that they differ from those that maximize Fisher information about the corresponding variables. They also suggest that neither linear reconstruction quality nor representational similarity fully account for the task performance associated with a representation.

It is unclear whether these observations can provide insights for learning of representations in the brain or in artificial systems. However, in light of the task relevance of tuning width, perhaps representation learning can be somehow decomposed into learning a tuning space, and learning parameters of the tuning width distribution in the space. This might potentially be more data-efficient than independently learning each feature.

Acknowledgments

This work was supported by the Natural Sciences and Engineering Research Council of Canada, Discovery Grant 418331.

References

Abadi, M., Agarwal, A., Barham, P., Brevdo, E., Chen, Z., Citro, C., et al. (2016). TensorFlow: Large-scale machine learning on heterogeneous distributed systems. *OSDI*, *16*, 265–283.

Bair, W., & Koch, C. (1996). Temporal precision of spike trains in extrastriate cortex of the behaving macaque monkey. *Neural Computation*, *8*(6), 1185–1202. http://dx.doi.org/10.1162/neco.1996.8.6.1185, URL http://www.mitpressjournals.org/doi/abs/10.1162/neco.1996.8.6.1185.

Baldi, P., & Heiligenberg, W. (1988). How sensory maps could enhance resolution through ordered arrangements of broadly tuned receivers. *Biological Cybernetics*, *59*(4–5), 313–318.

Barlow, H. B. (1972). Single units and sensation: a neuron doctrine for perceptual psychology? *Perception*, *1*(4), 371–394.

Boyraz, P., & Treue, S. (2011). Misperceptions of speed are accounted for by the responses of neurons in macaque cortical area MT. *Journal of Neurophysiology*, *105*(3), 1199–1211. http://dx.doi.org/10.1152/jn.00213.2010, http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3074420{&}tool=pmcentrez{&}rendertype=abstract.

Bremmer, F. (2005). Navigation in space - the role of the macaque ventral intraparietal area. *Journal of Physiology*, *566*(Pt 1), 29–35. http://dx.doi.org/10.1113/jphysiol.2005.082552, URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=1464721{&}tool=pmcentrez{&}rendertype=abstract.

Brown, W., & Bäcker, A. (2006). Optimal neuronal tuning for finite stimulus spaces. *Neural Computation*, *18*(7), 1511–1526.

Butts, D. A., & Goldman, M. S. (2006). Tuning curves, neuronal variability, and sensory coding. *PLoS Biology*, *4*(4), 639–646. http://dx.doi.org/10.1371/journal.pbio.0040092.

Cadieu, C. F., Hong, H., Yamins, D. L. K., Pinto, N., Ardila, D., Solomon, E. A., et al. (2014). Deep neural networks rival the representation of primate IT cortex for core visual object recognition. *PLoS Computational Biology*, *10*(12), http://dx.doi.org/10.1371/journal.pcbi.1003963, arXiv:1406.3284.

Chollet, F. (2015). Keras. URL https://keras.io/.

Cui, Y., Liu, L. D., Khawaja, F. A., Pack, C. C., & Butts, D. A. (2013). Diverse suppressive influences in area MT and selectivity to complex motion features. *Journal of Neuroscience*, *33*(42), 16715–16728.

Dayan, P., & Abbott, L. F. (2001). *Theoretical neuroscience: computational and mathematical modeling of neural systems*. MIT press.

DeAngelis, G. C., & Uka, T. (2003). Coding of horizontal disparity and velocity by MT neurons in the alert macaque. *Journal of Neurophysiology*, (2), 1094–1111. http://dx.doi.org/10.1152/jn.00717.2002, URL http://www.ncbi.nlm.nih.gov/pubmed/12574483.

Eliasmith, C., & Anderson, C. (2003). *Neural engineering: Computation, representation, and dynamics in neurobiological systems*. MIT Press.

Eurich, C. W., & Schwegler, H. (1997). Coarse coding: calculation of the resolution achieved by a population of large receptive field neurons. *Biological Cybernetics*, *76*(5), 357–363.

Georgopoulos, A. P., Schwartz, A. B., & Kettner, R. E. (1986). Neuronal population coding of movement direction. *Science*, *233*(4771), 1416–1419.

Hinton, G. E., McClelland, J. L., Rumelhart, D. E., et al. (1984). *Distributed representations*.

Johansson, G. (1973). Visual perception of biological motion and a model for its analysis. *Perception & Psychophysics*, *14*(2), 201–211. http://dx.doi.org/10.3758/BF03212378, arXiv:19433921.

Khaligh-Razavi, S. M., & Kriegeskorte, N. (2014). Deep supervised, but not unsupervised, models may explain IT cortical representation. *PLoS Computational Biology*, *10*(11), http://dx.doi.org/10.1371/journal.pcbi.1003915.

Kingma, D., & Ba, J. (2014). Adam: A method for stochastic optimization. arXiv:1412.6980 [cs]. 1–15, URL http://arxiv.org/abs/1412.6980%5Cnhttp://www.arxiv.org/pdf/1412.6980.pdf.

Kriegeskorte, N. (2008). Representational similarity analysis – connecting the branches of systems neuroscience. *Frontiers in Systems Neuroscience*, (November), 1–28. http://dx.doi.org/10.3389/neuro.06.004.2008, URL http://journal.frontiersin.org/article/10.3389/neuro.06.004.2008/abstract.

Kriegeskorte, N. (2009). Relating population-code representations between man, monkey, and computational models. *Frontiers in Neuroscience*, *3*(DEC), 363–373. http://dx.doi.org/10.3389/neuro.01.035.2009.

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*(3), 1097–1117. http://dx.doi.org/10.1152/jn.00990.2010, URL http://www.ncbi.nlm.nih.gov/pubmed/21562200.

Lisberger, S. G. (2010). Visual guidance of smooth-pursuit eye movements: sensation, action, and what happens in between. *Neuron*, *66*(4), 477–491. http://dx.doi.org/10.1016/j.neuron.2010.03.027, http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=2887486{&}tool=pmcentrez{&}rendertype=abstract.

Majaj, N. J., Carandini, M., & Movshon, J. A. (2007). Motion integration by neurons in macaque MT is local, not global. *Journal of Neuroscience*, *27*(2), 366–370. http://dx.doi.org/10.1523/JNEUROSCI.3183-06.2007, URL http://www.jneurosci.org/content/27/2/366.long.

Markov, N., Ercsey-Ravasz, M., Ribeiro Gomes, a. R., Lamy, C., Magrou, L., Vezoli, J., et al. (2014). A weighted and directed interareal connectivity matrix for macaque cerebral cortex. *Cerebral Cortex*, *24*(1), 17–36. http://dx.doi.org/10.1093/cercor/bhs270, URL http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=3862262{&}tool=pmcentrez{&}rendertype=abstract.

Newsome, W., Wurtz, R., Dürsteler, M., & Mikami, a. (1985). Deficits in visual motion processing following ibotenic acid lesions of the middle temporal visual area of the macaque monkey. *Journal of Neuroscience*, *5*(3), 825–840, URL http://www.ncbi.nlm.nih.gov/pubmed/3973698.

Nover, H., Anderson, C. H., & DeAngelis, G. C. (2005). A logarithmic, scale-invariant representation of speed in macaque middle temporal area accounts for speed discrimination performance. *Journal of Neuroscience*, *25*(43), 10049–10060. http://dx.doi.org/10.1523/JNEUROSCI.1661-05.2005, URL http://www.ncbi.nlm.nih.gov/pubmed/16251454.

Rezai, O., Boyraz Jentsch, P., & Tripp, B. (2018). A video-driven model of response statistics in the primate middle temporal area. *Neural Networks*, *108*, 424–441. http://dx.doi.org/10.1016/j.neunet.2018.09.004, URL https://linkinghub.elsevier.com/retrieve/pii/S0893608018302661.

Ringach, D. L., Hawken, M. J., & Shapley, R. (1997). Dynamics of orientation tuning in macaque primary visual cortex. *Nature*, *387*(6630), 281.

Salinas, E., & Abbott, L. F. (1994). Vector reconstruction from firing rates. *Journal of Computational Neuroscience*, *1*, 89–107.

Salzman, D. C., Britten, K. H., & Newsome, W. T. (1990). Cortical microstimulation influences perceptual judgements of motion direction. *Nature*, *346*, 174–177. http://dx.doi.org/10.1016/0021-9797(80)90501-9, http://www.ncbi.nlm.nih.gov/pubmed/16290341%5Cnhttp://link.springer.com/10.1007/BF01177222%5Cnhttp://linkinghub.elsevier.com/retrieve/pii/0021979780905019%5Cnhttp://scitation.aip.org/content/aip/journal/pof2/10/9/10.1063/1.869740%5Cnhttp://www.sciencedire.

Salzman, D. C., Murasugi, C. M., Britten, K. H., & Newsome, W. T. (1992). Microstimulation in visual area mt: Effects on direction discrimination performance. *Journal of Neuroscience*, *12*(6), 2331–2355.

Schrimpf, M., Kubilius, J., Hong, H., Majaj, N. J., Rajalingham, R., Issa, E. B., et al. (2018). Brain-score: Which artificial neural network for object recognition is most brain-like? bioRxiv, 407007,

Smith, J. E., Chang'an, A., & Cook, E. P. (2011). The functional link between area MT neural fluctuations and detection of a brief motion stimulus. *Journal of Neuroscience*, *31*(38), 13458–13468.

Tadin, D. (2015). Suppressive mechanisms in visual motion processing: From perception to intelligence. *Vision Research*, *115*, 58–70. http://dx.doi.org/10.1016/j.visres.2015.08.005.

Treue, S. (2001). Neural correlates of attention in primate visual cortex. *Trends in Neurosciences*, *24*(5), 295–300.

Vries, S. E. J. D., Lecoq, J., Buice, M. A., Peter, A., Ocker, G. K., Oliver, M., et al. (2018). A large-scale, standardized physiological survey reveals higher order coding throughout the mouse visual cortex, bioRxiv, http://dx.doi.org/10.1101/359513.

Wang, H. X., & Movshon, J. (2016). Properties of pattern and component direction-selective cells in area MT of the macaque. *Journal of Neurophysiology*, 74.2/OO9. http://dx.doi.org/10.1152/jn.00639.2014.

Warren, P. A., & Rushton, S. K. (2009). Optic flow processing for the assessment of object movement during ego movement. *Current Biology*, *19*(18), 1555–1560. http://dx.doi.org/10.1016/j.cub.2009.07.057, URL http://www.ncbi.nlm.nih.gov/pubmed/19699091.

Zhang, K., & Sejnowski, T. J. (1999). Neuronal tuning: To sharpen or broaden? *Neural Computation*, *11*(1), 75–84. http://dx.doi.org/10.1162/089976699300016809, arXiv:arXiv:1011.1669v3.