Applying 3D Convolutional Neural Networks to Human Psychophysics

Dan Birman
Stanford University
Gardner Lab, Dept. of Psychology
dbirman@stanford.edu

Dylan Cable
Stanford University
Gardner Lab, Dept. of Psychology
dcable@stanford.edu

Steeve Laquitaine
Stanford University
Gardner Lab, Dept. of Psychology
steeve@stanford.edu

Abstract

Despite the radical simplicity of convolutional neural networks some researchers have found direct correlates between network layer properties and actual neuron responses. In model systems such as the macaque [1] there are clear analogies between the early visual cortex layers (V1-V4) and the properties of a trained convolutional network. We plan to explore this interesting dynamic in the context of motion. Our goal is to modernize an older model of the visual stream dedicated to motion [2]. The original model explicitly coded the features selected for at each layer, based on the known anatomical properties of macaque V1 and MT. In contrast, we plan to build a generic convolutional neural net architecture, which will be trained to discriminate examples of motion. Because our architecture is more generic it will fail to precisely model the known anatomy of V1. This leaves open the possibility that during training the network will 'learn' similar features, such as simple and complex cell receptive fields. Our goal in building this model is to develop a model system within which we can test other interesting questions, such as whether a convolutional architecture will develop similar behavioral asymmetries to the actual human (and monkey) visual systems.

Introduction

As cognitive neuroscientists one of our core goals is to understand human behavior. Because we can't easily probe neural signals in human brains we often to turn to model systems such as monkeys or rodents as sources of data to understand behavior. We think of these model systems as implementing the same *computations* while using different *algorithms* and neural *implementations*. For example, to understand how humans process information about motion researchers have created a number of simple tasks involving moving dots. Using these stimuli humans can discriminate various features motion speed, coherence, contrast, etc. How do we do this? Monkeys are also able to perform this task and offer a wealth of information about what neurons in different cortical areas might be doing and the limits of neural representations [3]. One issue with monkeys is that we aren't quite sure that they are behaving identically to humans--they show very distinct training patterns and in some ways their behavior looks nothing like human behavior [4]. Because of these issues cognitive neuroscientists have begun to think of model systems as pieces of a puzzle, best interpreted in light of other pieces--such as evidence from other model systems [5]. Despite their radical architectural differences artificial neural networks can be trained to perform the exact same behavioral tasks that we are interested in studying in humans [1]. They act in this sense as another model system that can give us insights into how information might be represented in the human brain.

In this study we designed a motion network (MotionNet) with architectural similarities to the human visual stream up to human area hMT+ (areas MT/MST in the monkey). Our neural network was trained to perform direction discrimination on video clips of random dot displays at full contrast, full coherence, uniform speed and dot number, and low noise. We show that our model has characteristics that are qualitatively similar to humans, such as poor motion discrimination under adverse conditions of low contrast, low coherence, and high noise. We also show that our model has quantitative similarity to humans, in that its ability to discriminate untrained features of the stimulus tracks human performance. We believe that models like MotionNet are a powerful tool to understand the human visual system and can give us precise and testable predictions about how the brain may represent motion.

Methods

We trained our network on a psychophysical task in which white dots undergo random translational motion on a black background. We put together basic functions to generate an unlimited number of translational random dot displays. All of these stimuli can have arbitrary contrast (e.g. Michelson contrast, difference between luminance intensities in the dots and background, compared to the total luminance),

Table 1: Training Dataset Parameters

Video Size	16 Frames x 64 pixels x 64 pixels
Angle Directions [0-360]	0, 45, 90, 135, 180, 225, 270, 315
Coherence [0-100%]	100%
Velocity [0-inf]	3 pixels / frame
Number of dots [0-inf]	15
Dot radius [0-inf]	2 pixels (std of gaussian dot)
Contrast [0-100%]	100%
Noise [0-255]	4 (std of additive gaussian noise)
Number Train Examples	1600
Number Validation Examples	200
Number Test Examples	200

speed, number of dots, etc. This parameterization allowed us to generate an unlimited set of training examples on the fly. One advantage of having tools to generate online data is that we can build batches without relying on a heavy memory load, allowing us to train our dataset on an arbitrary number of examples with no overhead for loading and saving data. This also allows us to generate far more variability in our input dataset, similar to jittering of image data.

We set up a four layer convolutional neural network, modeled after the human visual system (Fig. 1). To that end we have a layer corresponding to each cortical and sub-cortical region spanning the retina to MT. Although this is far from a perfect analogy to cortical computations we introduced several constraints that will allow us some anatomical similarity to cortex. First, we enforced the first three layers to be 2D convolutions (across both spatial dimensions), whereas the final layer (MT) was a 3D convolution (across both spatial dimensions and time). Our full model parameters are shown in Table 2. While it is true that LGN, and V1 have interesting temporal dynamics, for the behavior that we are interested in, we can think of these layers as making static computations on individual frames. Second, we used spatial batch normalization as a form of divisive normalization. Divisive normalization is a property of local sets of neurons in cortex whereby they inhibit each other: a feature of the brain that limits explosive response rates. Our hope is that although these constraints are not explicitly anatomically correct, they will nevertheless lead to the MotionNet learning a set of features analogous to the human/monkey visual system. We trained the neural network to do motion direction discrimination on stimuli with constant contrast, coherence, number of dots, noise, and velocity (Table 1).

Our network was able to achieve near perfect cross validated classification accuracy (95.6%). We chose to train for direction discrimination because it is a simple task and the architecture responsible for performing the task is well understood in the monkey brain [2]. V1 neurons are thought to be edge detectors that yield positional information about the stimuli, and MT neurons differentiate V1 behavior in order to obtain motion information.

Feature Inversion

We computed feature and class inversion according to the following algorithm:

```
img = random initial image
for i iterations:
    forward pass img
    compute gradient relative only to one
      class/feature
    img += dImg * stepsize
```

Training parameters: learning Rate: $1e^{-3}$, L2-regularization on W matrices: $1e^{-3}$, batch size 20				
Name	***Layer***	***Input Space***	***Output Space***	***Params***
	ZeroPad3D: 0x1x1	16x64x64	16x66x66	0
LGN	Conv3D: 4x1x3x3 ReLU, BatchNorm	16x66x66	16x64x64	40
	MaxPool3D: 0x2x2	16x64x64	16x32x32	0
	ZeroPad3D: 0x1x1	16x32x32	16x34x34	0
V1s	Conv3D: 4x1x3x3 ReLU, BatchNorm	16x34x34	16x32x32	40
	MaxPool3D: 2x2x2	16x32x32	8x16x16	0
	ZeroPad3D: 0x1x1	8x16x16	8x18x18	0
V1c	Conv3D: 4x1x3x3 ReLU, BatchNorm	8x18x18	8x16x16	40
MT	Conv3D: 4x5x3x3	8x16x16	4x14x14	184
LIP	Dense Fully Connected	4x4x14x14	8	3144

Table 2: Model Parameters. Our model included four convolutional layers and a single dense readout layer. The full architecture is shown above with details about parameter numbers and kernel sizes for each of the layers.

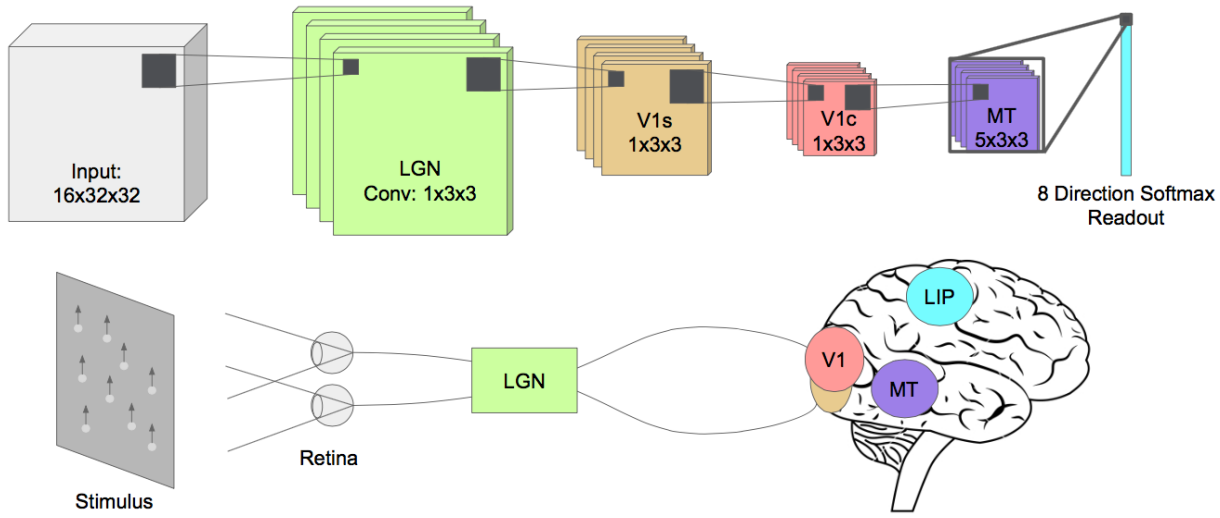

Figure 1: Architecture. MotionNet has four convolutional layers, one per synapse in the human visual system. Each layer was assigned four features, sufficient to drive near perfect discrimination of 8-direction translational motion, but also enough to force the model to encode multiple directions into each feature.

Psychometric Discrimination Functions

Human volunteers (n=3, two female one male) were paid to perform approximately 2000 trials of a contrast discrimination and a motion discrimination task. Our experiment was approved by the Stanford IRB and all subjects gave informed written consent. Each subject was shown two random dot displays and asked which display had higher contrast or motion coherence. Data was collected at different contrast and motion coherence differences. A maximum likelihood fit was found for a Weibull function of the form:

$$(1-\gamma-\lambda)*\left(1-e^{-x/T^{\beta}}\right)+\gamma$$

Where T is the threshold, β is the slope, λ is the lapse rate, and γ is chance performance. To compute psychophysical performance for the model we generated a dataset with 100 examples each of a base contrast (10%) or coherence (20%) and a test contrast (10-100%) or coherence (20-100%). We read out layer activations from the trained model at each convolutional output layer (see Information Readout) giving us data in the same format as for the humans.

Information Readout

The information types that we tried reading out were speed, contrast, number of dots, coherence, and noise. For each layer of the network and information type, we generated N = 2400 train examples, N = 240 validation examples, and N = 240 test examples. We then fed each example through the network, and took the output at the layer of interest. We then averaged the output across space and time, meaning that we have one feature for each filter (which is less than 10 for each layer). The logic for averaging across space and time is that a given filter is measuring the same information regardless of its spatiotemporal position. We subsequently trained a linear regression with this reduced dimension activation as input and the parameter of interest as output. For example, if we were trying to decode speed, the output for the regression would be speed. This linear regression was trained on the train set, a regularization parameter was chosen using the validation set, and performance was measured using the test set.

Code

All code, dataset generating, and analysis scripts are in our repository: https://github.com/dbirman/motnet. Our convolutional neural network is built using Keras with a mixture of custom code and existing pull requests to implement 3D convolutions. Keras is a wrapper package that simplifies the implementation for Theano [6].

Results

We measured the similarity of our model to humans using three qualitative and quantitative measures. (1) We looked at the features learned by the model, (2) the psychophysical performance of the model on discrimination tasks, and (3) the information representation of individual model layers.

MotionNet Features

Our features at layer MT are videos that cannot be reproduced in print, but can be found on our website,

along with the features at all of the other layers at http://gru.stanford.edu/doku.php/deepmotion. One drawback of 3x3 convolutions is that we do not get much out of visualizing the weights directly; rather, it is through feature inversion that we can gain insight into the behavior of our features (see Methods). We can observe that our early layers (LGN, V1s) developed a preference for very strong edges. This makes sense since tracking the position of edges is central to the task of detecting motion. Furthermore, each MT neuron seems to have preferences over motion directions. However, one significant difference between our model and the human neural network is that our MT learned a much more distributed representation of motion direction. In humans, each MT neuron has one preferred direction of motion, whereas our MT neurons seem to be encoding something more complicated than a singular motion direction. This is due, in part, to the fact that we only have four neurons in our MT. Having very few MT neurons was a design choice that we made to force the network to use all available features in its representations.

Results: Psychophysics

Next, we wanted to test whether MotionNet would perform similarly to humans on psychophysical tests. We tested this in two ways: first, by looking at the tuning of our model to the motion stimulus parameters (Fig. 2). Second, we looked at how we could read out information from our model (Fig. 3 & Fig. 4).

In Figure 2 we varied the stimulus parameters to understand where our model performed optimally. We found that, as expected, our model was tuned to the parameters of the training stimulus set. Performance dropped off whenever any parameter extended too far from the optimum. This is qualitatively similar to what we see in human psychophysics data. To test this quantitatively we examined linear readouts from our model output layers.

We performed model readout in two ways. First we tried to simply reconstruct the input stimulus parameter as a linear readout from each layer's activations, the results from this analysis are shown in Figure 3. We found that the low-level parameters of the stimulus such as number of dots, contrast, and noise were all trivial to read out from our LGN layer and progressively more difficult to read out as we progressed further into the model. Layer MT was relatively invariant to these features. In contrast we found that parameters related to motion like speed and coherence could only be read out from the later layers. This makes sense in comparison to the human cortex, where motion cannot be easily read out from area V1, while it can be easily read out from area MT. We were surprised to see that speed could be decoded from layer V1c, but we interpret this as an effect of very high speeds where the dots would start to "jump" due to our low temporal resolution.

Our second approach to model readout was to explicitly compare the model's discrimination of stimulus features with human discrimination. We chose to use contrast and coherence rather than motion discrimination itself because these are a low-level

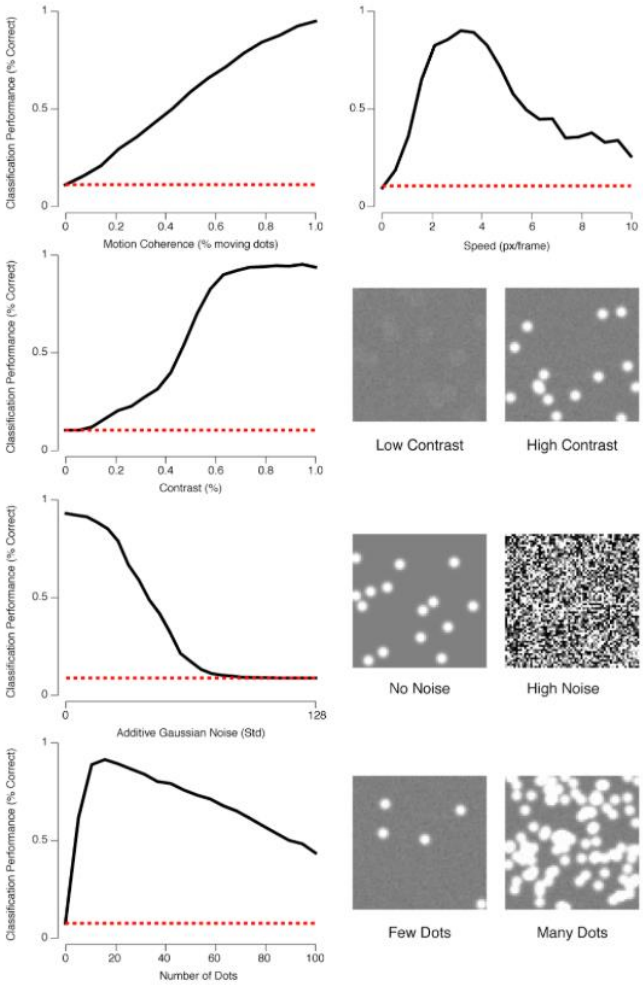

Figure 2. Validation Performance. We validated our model by looking at its performance on untrained ranges of the stimulus features. As expected we found that the model was "tuned" to the parameters of the training dataset, with performance dropping off as parameters extended too far away from the optimum.

stimulus feature and a high level one, respectively, that we know humans can easily discriminate. For this experiment, we varied the each stimulus property's

difference between a baseline stimulus and a test stimulus. The task was to discern which stimulus has the higher contrast or motion coherence. Our model was able to perform this task and the results are shown

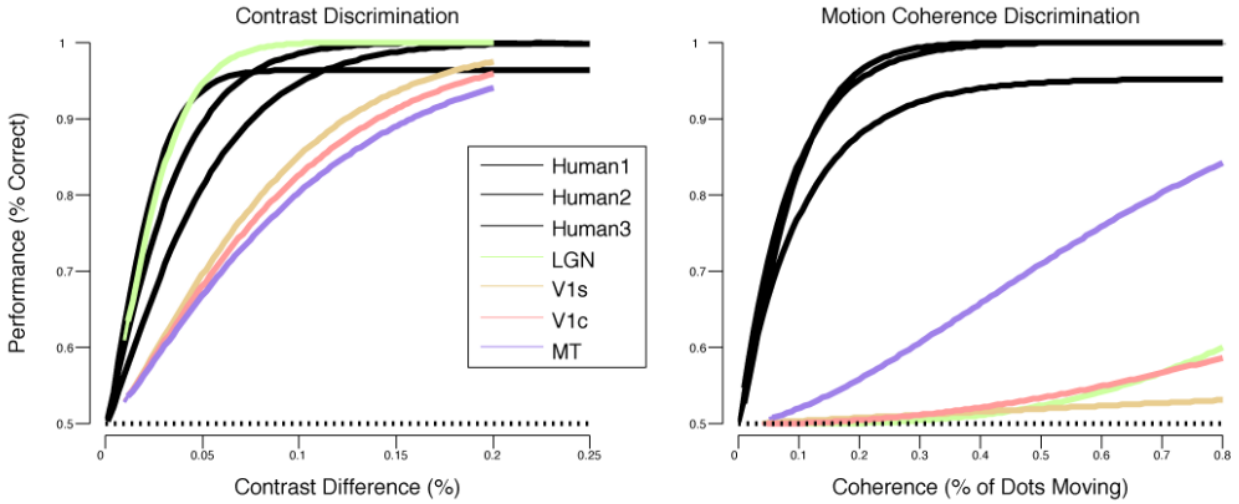

Figure 3. Human vs. Model Psychometric Functions. We fit weibull functions to performance data obtained for both humans and our model. We found that only layer LGN of our model outputs sufficient information to do a linear readout of contrast at the same accuracy as human performance. None of our model layers contained sufficient information to do a linear readout of coherence at human performance and only MT showed discrimination performance approaching human performance. Motion coherence though is inherently non-linear (i.e. coherence depends on the direction of motion) and we believe a more involved readout may be sufficient to achieve human-level discrimination.

in Figure 3. As the difference increased, this task becomes easier. Similar to the humans, for both contrast and coherence discrimination, the computer exhibits a smooth increase from chance to perfect performance. For contrast, performance looked markedly similar for both humans and robots, which is impressive since the parameters of our model are so different from biological settings. Coherence, on the other hand, was more difficult to read out of the neural network. This is because we used a linear readout model. However, the true process of reading out coherence information from MT is inherently nonlinear. Consequently, if we were to train a more involved readout procedure, we might be able readout coherence from MT.

Conclusions

Our results show that, although anatomically very different from human visual cortex, an artificial neural network nevertheless has many qualitative similarities to humans. In our goal of understanding precisely how human vision represents motion information, a tool where we have explicit access to every computation and every input and output is a huge strength. Normally at best we can observe brain activity and make small tweaks--although we are always at risk of overinterpreting our results, in particular when behavioral similarities between model systems turn out to be overstated [3]. The greatest advantage of our MotionNet as a model system is its ability to be iterated over in fast steps. The model can be trained in a matter of minutes and the validation and analyses can be run within a few days--this fast turnaround means that hypotheses about the human visual system can be implemented and tested at speeds order of magnitudes faster than in traditional model systems. Ultimately we see artificial neural networks as a model system that adds to our existing models. One of our goals for future work with MotionNet is to see whether our layer activations show correspondence with the activation of layers in the human and monkey visual system. For example, one approach based on the the work of Yamins et al. [1] would involve trying to predict human fMRI BOLD activity as a weighted sum of a small number of units in our MotionNet model. By seeing which layers are most predictive of human BOLD activity we can infer what kinds of representations the human brain stores in neural activity within individual voxels. These kinds of approaches are difficult to do in other model systems because of the lack of available data--but with MotionNet we can generate arbitrary amounts of data from many different models in rapid iterations. Ultimately the strength of artificial neural networks as a model system will depend on their success in generating new insights about the human visual system.

Acknowledgments

Thank you to the CS231n staff and Lane McIntosh for comments on previous iterations of this work.

References

1. Yamins, D. L., Hong, H., Cadieu, C. F., Solomon, E. A., Seibert, D., & DiCarlo, J. J. (2014). Performance-optimized hierarchical models predict neural responses in higher visual cortex. *Proceedings of the National Academy of Sciences*, *111*(23), 8619-8624.
2. Heeger, D. J., Simoncelli, E. P., & Movshon, J. A. (1996). Computational models of cortical visual processing. *Proceedings of the National Academy of Sciences*, *93*(2), 623-627.
3. Britten, K. H., Shadlen, M. N., Newsome, W. T., & Movshon, J. A. (1993). Responses of neurons in macaque MT to stochastic motion signals. *Visual neuroscience*, *10*(06), 1157-1169.
4. Birman, D., & Gardner, J. L. (2016). Parietal and prefrontal: categorical differences? *Nature neuroscience*, *19*(1), 5-7.
5. Churchland, A. K., & Abbott, L. F. (2016). Conceptual and technical advances define a key moment for theoretical neuroscience. *Nature neuroscience*, *19*(3), 348-349.
6. J. Bergstra, O. Breuleux, F. Bastien, P. Lamblin, R. Pascanu, G. Desjardins, J. Turian, D. Warde-Farley and Y. Bengio. “Theano: A CPU and GPU Math Expression Compiler”. *Proceedings of the Python for Scientific Computing Conference (SciPy) 2010. June 30 - July 3, Austin, TX*