Biologically Motivated Object Recognition

Peter Henderson, Dhirendra Singh

Mcgill University, Montreal, Canada

Abstract

Here we modify the HMAX object recognition system of Serre et al. with an end-stopped filter to try and improve the accuracy of the model and to further the parallels in the algorithm to the actual processing of the visual cortex for learning object representations.

Keywords: Biologically Inspired Object Recognition, HMAX, Cortical Keypoints

1. Introduction

Object Recognition is a fundamentally complex problem in Computer Vision as well as Machine Learning. Various approaches have been taken in attempts to match the precision of the human object recognition system, but few have come close to succeeding. Two distinct method tracks have been created in these attempts: feature-based methods and appearance-based methods. Appearance based methods generally use a large database of images to find the closest match to the presented image. These generally perform well with little position or lighting invariance, if the object exists in the database, but perform poorly in variant conditions. A more successful approach has been in the category of feature-based methods. Speeded-Up Robust Features (SURF) are used in a model from Bay et al.[1] which incorporates Hessian matrix based features to learn representations of images. Though this model is fast and relatively robust in ignoring variance in position and scale, it still does not match the accuracy of the human visual system.

As the state of the art algorithms in both the previously mentioned tracks of methodologies still cannot compete with the accuracy and speed of the human object recognition system, the question turns to how does the human visual system work and how can it be recreated in software. With the work of

Hubel and Wiesel [2] in finding the role of simple and complex receptive fields in the visual cortex, the layer based feature extraction used by the human brain has become increasingly clear and mathematical models representing these receptive fields have been formed.[3] Based on these models, several algorithms have been developed attempting to mimic the early stages of feature extraction in the visual cortex. Though some remain skeptical that biologically inspired systems are the correct approach, it is undeniable that cortical based methods outperform the state of the art in object recognition in the current state of affairs.[4]

1.1. Algorithms Based on Simple and Complex Cells

As Hubel and Wiesel showed[2], the initial stages of feature extraction in the visual cortex happen in a feed-forward (for the most part) network of neurons which are composed of several different types of cells which form receptive fields. More prominently, the cells which many modern algorithms draw for inspiration from are simple and complex cells. Simple cells distinguish contrast in light (for example, a dark spot surrounded by light, or visa versa). A series of these simple cells feed into one cell, called a complex cell, which has an activation function represented by a line of contrast angled in a certain direction. The activation functions of these cells are called receptive fields. A visualization of simple and complex cell receptive fields can be seen in Figure 1.

Marr and Hildreth theorized that the feature extraction occurring in the simple cell stage of the visual cortex could be mimicked by the Laplacian of the Gaussian convolved with the image. The purpose of this convolution and the simple cells in general, they showed, was for edge detection.[5] Duagman et al. and Jones et al. later showed that Gabor Filters actually better represented the activation function of the simple cell and convolution of these filters with the image would achieve better results than the LoG in edge detection. [6, 3]

Several algorithms, generally referred to as HMAX algorithms, stemmed from these papers. They used the Gabor filters which representing the activation functions of simple cells and applied them for use in Object Recognition by creating feature representations of images. Most notably, Serre et al.[7, 8] developed a largely successful algorithm mimicking the feedforward structure of simple and complex receptive fields in the brain for use in object recognition. In their algorithm they use four layers of feature extraction followed by a simple learner (they compare an SVM with gentleBoost). The first layer is

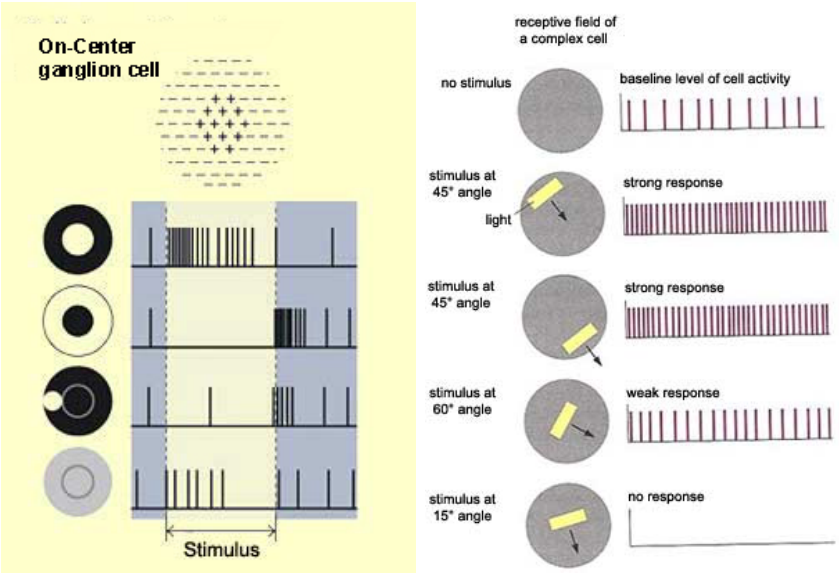

Figure 1: On the left is a depiction of a simple cell, where the cell is activated if the center of its field is much brighter than the outer part of the field. On the right, is a complex cell receptive field. The complex cell can be seen as a line of simple cells and is activated by a line of contrast at a particular angle in the receptive field. Both images were taken from: `http://thebrain.mcgill.ca/flash/i/i_02/i_02_cl/i_02_cl_vis/i_02_cl_vis.html`

a series of convolutions of the image with Gabor Filters of different angles and sizes. The second layer, called the Max-Pooling C1 layer, mimics the pooling action of complex cells in the visual cortex by taking the convolutions that have the largest (maximum) response. The following two layers, which they refer to as S2/C2 are supposed to represent another layer of simple and complex cells and follow the generalization which occurs further on the primate visual cortex for learning, but in reality, the implementations stray a little from the biological representation. In the S2, they essentially take random patches of the image and test the feature representations against them using a radial basis function (akin to Euclidean Distance) to learn feature representations of an image (similar to PCA). Then in the C2 layer, the global maxima are taken again to be the final representations of the total image.

1.2. End-Stopped Cells and Work Based on Them

While Serre et al. only took into account the Simple and Complex cells in the visual cortex to learn feature representations of an image, the visual

cortex actually continues the feedforward approach of feature extraction in the form of hypercomplex cells. [9, 10] These cells have a deeper representation of edges in the image allowing for the detection of curves and corners. While complex cells essentially allow for the detection of lines at different angles, hypercomplex cells - or more specifically, end-stopped cells - have an activation function which is only stimulated at a certain length of line (a line in this case is actually a straight patch of contrasting brightness). [11, 12] Not much work has been done in the field of object recognition using mathematical representations of these end-stopped cells until recently. Initially, Terzić et al. used a mathematical representation of these end-stopped cells - put forward by Heitger et al.[11] - to extract features for hand gesture and pose recognition.[13] More recently however, they have fairly successfully combined the features found from end-stopped cell representations with SURF features for fast real-time Object Recognition.[14] Additionally, the same group has put forward attempts to incorporate the full range of inputs used by the visual cortex for real time object recognition. This includes Optical Flow and Motion Analysis.[15]

2. Technical Background

2.1. Simple Cell and Gabor Filter Representation

As previously mentioned, Hubel and Wiesel showed evidence that in the early stages of the visual system, simple cells have an activation function based on the contrast of the image patch provided to them (see Figure 1).[2]. Each simple cell corresponds to a relative part of the image plane (i.e. the simple cells activated by the bottom part of an image are lined up together relative to the part of the image they are processing) and the neurons have alternating columns from the left and right eyes corresponding to the same part of the image being processed. As an aside, this is due to the presence of stereo processing related to these fields.

2.1.1. Gabor Filters

Marčelja et al.[16] noted that Gabor filters were extremely similar to the structure of the simple cell activation function along one axis. Daugman et al. later generalized this observation to a 2D space[6] to allow for convolution with a 2D image - as occurs in the visual cortex. Jones et al. follow Daugman's generalization with experimental proof that 2D Gabor Filters

accurately represent the activation function of simple cells in the visual cortex. They explain, that 2D Gabor filters can be formed by "bivariate elliptic Gaussians modulated by sinusoidal plane waves".[3] The resulting equation looks like the following:

$$G(x, y; \lambda, \theta, \phi, \gamma) = \exp\left(-\frac{x'^2 + \gamma^2 y'^2}{2\sigma^2}\right)\cos\left(2\pi\frac{x'}{\lambda} + \phi\right) \tag{1}$$

where

$$x' = x\cos\theta + y\sin\theta$$
$$y' = -x\sin\theta + y\cos\theta$$

and

ϕ: Phase offset
θ: Orientation of filter
λ: Spatial Frequency (Wavelength of the sinusoidal component)

By comparing this function to the average distribution of 2D spectral response profiles from simple cells in 14 different cats, Jones et al. confirmed that the Gabor Filter representation had a relatively low error rate to the real responses of simple cells, and as such can effectively be used to mimic the simple cell responses in a synthetic vision system. By convolving a given image at several different angles and using several different filter sizes, the full range of simple cells can be reproduced in fairly accurate fashion, as seen in the first layer of the HMAX algorithm of Serre et al. [7, 8] The values for σ, λ, ϕ, and the various filter sizes used by Serre et al. were taken from studies of real "parafoveal simple cells" in the V1 layer of the visual cortex and as such hold a deep basis in biological vision.[17, 18]

2.2. Complex Cells and Max-Pooling Representation

In the visual cortex, a series of simple cells feed forward into a complex cell. This can be visualized as a series of simple circular contrast thresholds being added together to form a line at a certain angle (see Figure 2). As a model representation, Serre et al.[7, 8] accurately portray this sort of pooling operation by taking the maximal responses from convolutions with filters of a certain size range. For example, two filters of size $7x7$ and $9x9$ would create a complex cell grid of $8x8$ and the maximal responses within each of the two filters would be mapped into this grid. This operation and its parameters

(mainly the overlap of the cell grid section when dividing the image into patches representing the activation area of a single complex or simple cell) were taken from experimental results presented in past research on the feline visual cortex, as mentioned earlier.[19]

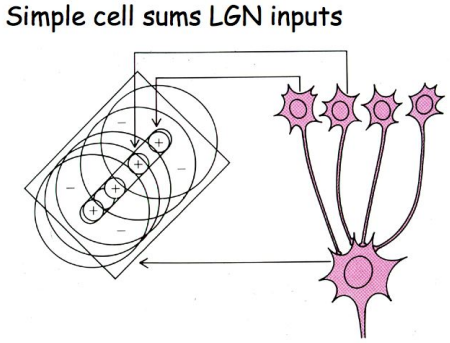

Figure 2: A series of simple cells actually feed into a complex cell creating a line of contrast. This is a max-pooling activation function. Image taken from: `http://www.cns.nyu.edu/~david/courses/perception/lecturenotes/V1/lgn-V1.html`

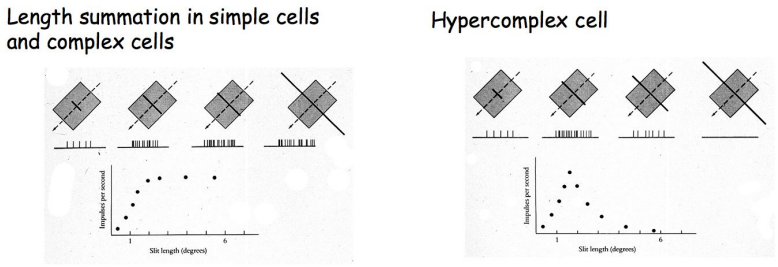

Figure 3: On the left is the activation function of a complex cell, at a given angle of line the cell is activated until the entire receptive field is filled with the line, then the response rate levels off. On the right is an end-stopped cell. The response rate reaches a peak at a certain length of line and then decreases past that length. Images taken from: `http://www.cns.nyu.edu/~david/courses/perception/lecturenotes/V1/lgn-V1.html`

2.3. End-Stopped Cells and Mathematical Representation

Further on in the visual cortex, complex cells feed into hypercomplex cells, but more specifically end-stopped cells. These cells have an activation

threshold that is acute to only a particular length of line or edge (see Figure 3). By performing a pooling function from a collection of these cells, a neuron may be formed which is activated by curves and corners (see Figure 4).

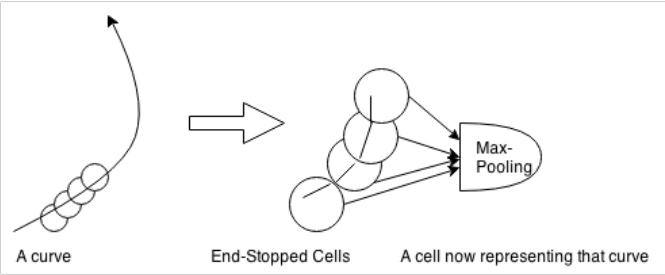

Figure 4: A series of end-stopped cells can make up a curve as depicted here, these cells can pool into another cell which then represents a certain curve as a whole. Image created for the purposes of this paper.

Heitger et al. explain that there are two forms of end-stopped cells: double-stopped and single-stopped.[11] Double-stopped cells generally have suppression on both ends of the cell. In other words, the line segment which they detect doesn't extend to either side of the receptive field but is rather solely in the middle of it. Single-stopped cells, on the other hand, have suppression on one end of the line. So, in these cells, one end of the line can extend out of the range of the receptive field, but the other end of line has to be of a certain length in the range of the receptive field. Rodrigues et al.[20], using the original Heitger et al. derivation, show that Single-Stopped and Double-Stopped cells, respectively, can be depicted as:

$$\mathrm{S}_{s,i}(x,y) = [C_{s,i}(x + dS_i, y - dC_i) - C_{s,i}(x - dS_i, y + dC_i)]^+ \quad (2)$$

$$\begin{aligned} \mathrm{D}_{s,i}(x,y) = [C_{s,i}(x,y) - \tfrac{1}{2}C_{s,i}(x + 2dS_i, y - 2dC_i) \\ -\frac{1}{2}C_{s,i}(x - 2dS_i, y + 2dC_i)]^+ \end{aligned} \quad (3)$$

where

$$C_{s,i}(x,y) = [R^E_{s,i}(x,y)^2 + R^O_{s,i}(x,y)^2]^{\frac{1}{2}}$$

$R^E_{s,i}(x,y)$: The convolution of images with Gabor filters of even ϕ values
$R^O_{s,i}(x,y)$: The convolution of images with Gabor filters of odd ϕ values
d: A scaling of the original Gabor filter scale, Rodrigues et al. use $d = .6s$
$[.]^+$: Denotes the suppression of negative values
$C_i = cos\theta_i$
$S_i = sin\theta_i$
s : scale-band of filter
i : orientation of filter

The C-operator, as Heitger refers to it, is supposed to be an energy function representing the complex cells of the visual cortex (in the Serre et al. paper, this is presented by the Max-Pooling C1 layer). This is made up of an average of the convolutions of the Gabor Filters with the image in question. Terzić et al. later simplify this equation to a convolution kernel derived from Dirac functions[14], given for Double-Stopped cells as the following:

$$k^D_{\lambda,\theta} = \delta(x,y) - \frac{1}{2}[\delta(x - 2ds, y + 2dc) + \delta(x + 2ds, y - dc)] \tag{4}$$

where
$ds = 0.6\lambda sin\theta$
$dc = 0.6\lambda cos\theta$
λ and θ are the values used in the Gabor filter convolutions

This is a much simpler way of viewing the convolution kernel and is the one we use for the implementation in this paper. The derivation of this can be seen in the Heitger et al. and Terzić et al. papers, but the idea behind it is to take the existing outputs of complex cells and add a function which inhibits the edges at both ends (hence here the edges are inhibited by a function of the theta of the angle of convolution).

3. Methodology

3.1. Feature Extraction

In an effort to mimic the visual cortex's feature extraction process for object recognition, we use a feedforward network of convolutions with kernels representing simple cells and end-Stopped cells mixed with the max-pooling action of complex cell models (see Figure 5). To narrow down the feature set further, we use the technique put forth by Serre et al. [8] in using a radial

basis function to create a set of N feature representations which can compose each image.

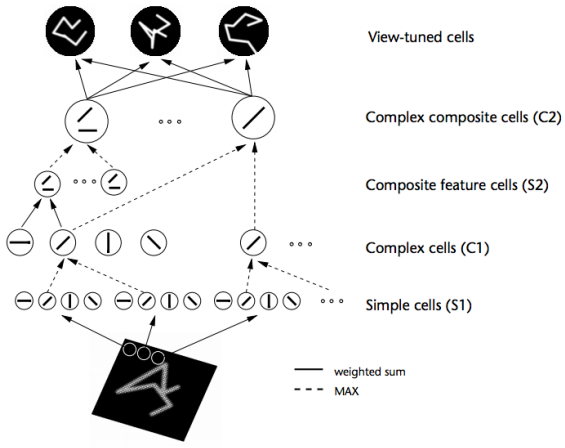

Figure 5: Here the hierarchical model of convolutions and filters can be seen. End-stopped cells are not featured here but can be represented in this hierarchical model as pooling corners and curves. Image taken from [21]

3.1.1. S1

For the S1 layer, we convolve the Gabor filter presented in Eq. 1 with the images. Various filter sizes and scales are used. To mimic the visual cortex further, we use hyperparameters presented by Serre et al. [8], which were collected in studies on the simple cells of feline visual cortices. A table of these values can be seen in Figure 6. The resulting convolutions are passed to the following Max-Pooling layer, representing the Complex Cells of the V1 layer of the visual cortex.

3.1.2. C1

The C1 layer of the algorithm takes the maximal responses from the related filtered images and pools them together. This was discussed earlier in the technical background section. This is the same process which Serre et al. use and the related values can be seen in Figure 6.

3.1.3. ES1

The ES1 layer is a layer of end-stopped cell convolutions. For simplicity, we only use the double-stopped cell representation seen in Eq. 4. Terzić et

C_1 layer			S_1 layer		
Scale band $\mathcal{S}$	Spatial pooling grid ($N_S \times N_S$)	Overlap $\Delta_{\mathcal{S}}$	filter size s	Gabor σ	Gabor λ
Band 1	8×8	4	7×7 9×9	2.8 3.6	3.5 4.6
Band 2	10×10	5	11×11 13×13	4.5 5.4	5.6 6.8
Band 3	12×12	6	15×15 17×17	6.3 7.3	7.9 9.1
Band 4	14×14	7	19×19 21×21	8.2 9.2	10.3 11.5
Band 5	16×16	8	23×23 25×25	10.2 11.3	12.7 14.1
Band 6	18×18	9	27×27 29×29	12.3 13.4	15.4 16.8
Band 7	20×20	10	31×31 33×33	14.6 15.8	18.2 19.7
Band 8	22×22	11	35×35 37×37	17.0 18.2	21.2 22.8

Figure 6: All the hyperparameters used in the C1 and S1 layers of the HMAX algorithm. These were taken from the Serre et al. paper[8] due to the fact that they were collected experimentally by testing the responses of actual S1 and C1 cells in feline visual cortices. This provides for a deeper entrenchment in the biological basis for this model.

al. and also avoid the single-stopped cell representation due to slowing down of the algorithm and complexity related to the number of features and layers of processing. The kernel is convolved with the afferent C1 layer outputs. This is parallel to multiple complex cells feeding into the end-stopped cells in the visual cortex in the beginning stage of curvature detection.

3.1.4. C2

In our C2 layer, we again use the same Max-Pooling operation as in C1 to narrow down the features presented and create a feature set related to curves and corners. We use the same pooling operation as in C1 in the Serre et al. HMAX algorithm. As mentioned earlier, multiple end-stopped cells can pool into one cell to form an activation function related to a particular curve or corner (see Figure 4). The C2 layer is parallel to the pooling cell seen in this representation.

3.1.5. S2

To improve the feature set further, we use the technique of Serre et al. in their S2 layer. In this layer, each unit, or neuron, acts as a radial basis function (RBF). The properties of these functions bare a resemblance to the response properties in the primate inferotemporal cortex. Poggio et al. have shown that this functionality is key in generalization used for the vi-

sual learning process and hence is an ideal biologically inspired method for generalizing the current feature set. [22] From Serre et al., the response is modeled by the following, where X is an image patch from the pooled C2 layer images at a particular scale band.

$$r = exp(-\beta||X - P_i||^2) \tag{5}$$

Here, β is the "sharpness of the TUNING and P_i is one of the N features (center of the RBF units) learned during training".[8] We use Serre et al.'s learning processes, using the RBF to converge on a "feature dictionary". Initially each, P_i is initialized to a random feature representation and is updated by Eq. 5 throughout the learning process. This can be thought of as a biologically inspired version of Principal Component Analysis, where the image is being broken into a feature set made of N "prototypes", P_i.

3.1.6. C3

In this final Max-Pooling layer, we take the global maximum response over all scales and positions from the S2 units. This global maximum is determined by matching each S2 features against a stored prototype P_i described in the learning process taken from Serre et al. The best matching S2 feature for each scale and position remains, resulting in a total of N C3 features where N matches the number of prototypes described in the learning stage of the S2 layer.

3.2. Learning and Classification

In this final stage of the algorithm, we train the algorithm on a set of images labeled by an object class. The images are put through the feature extraction stages and the C3 features are run through an SVM to learn feature representations for each object. Though an SVM isn't biologically inspired, the purpose of this project was to develop a feature extraction method based on the visual cortex, so a simple learner was used. A discussion of biologically inspired learners that would perhaps be more fitting can be found in the Future Work section.

3.2.1. Dataset

For training and testing, we selected four of the imagesets from the Caltech101 database for comparison against current techniques. As in past papers [8, 23] we use the motorcycle, face-front, cars, and airplane image sets

for learning and testing recognition. We convert them to greyscale and resize all images to 128 x 128 pixels, as in past papers, for better accuracy and simplicity. We used several techniques as in Serre et al. to run our algorithm.

First, we use a binary classifier. So for each imageset we create a set of 40 positive training samples and 50 negative training samples. We then divide the new set into 10 random folds, training on 9 folds and testing on one. The accuracies are then averaged for the final result.

For the ideal case, a multiclass system, we combined the 5 imagesets each with a given label and then used the same 10-fold validation scheme to obtain average accuracies for the model.

3.2.2. Learner and Testing

In the binary case, we used a linear SVM classifier, while for the multiclass case we used a multiclass SVM with a polynomial kernel.

4. Results

The results can be seen for the binary classification cases as well as for the multiclass case in Table 1. In the table we also add the results of Serre et al. on the datasets provided in [8] for comparison. For the multiclass case, the Serre et al. results are for 102 labels, so cannot directly be compared against our results which are only for 5 classes.

Imageset	**Average Accuracy (Our Model)**	**Average Accuracy (Serre et al.[8])**
Motorcycle	93.1	97.4
Face-Front	89.7	98.1
Cars	93.6	99.8
Airplane	92.2	94.9
Multiclass	67.47 (5 classes)	45.14 (102 classes)

Table 1: The average accuracies in percentages when using an SVM for our algorithm vs. the Serre et al. implementation. Note: For the multiclass accuracies, we used 5 classes, while Serre used 102 classes.

5. Conclusions

Our results do not differ greatly from the Serre et al. paper for the binary classification case. They are slightly less than the reported results of their

algorithm, probably due to the fact that we did not spend much time tuning the SVM learner. Additionally, for such a simple classification case, where you only determine if an image is of a particular object or not, perhaps honing the feature set with end-stopped filters is not necessary and results in a loss of necessary specific information. Our more specific method of feature extraction, which begins to highlight curves and corners through end-stopped cell methods, results in the keypoints which might hamper the process for binary classification, but would improve results in a multiclass system.

For the multiclass case, our algorithm seemed to outperform the Serre et al. algorithm. However, it must be noted that Serre et al. ran trainined and tested their algorithm on 102 classes while we trained/tested ours only on 5. As a result, the given accuracies are skewed to favor ours. For a slightly better indicator of comparing the two algorithms, we take the error percentage from random guessing. For 102 classes, random guessing should yield close to 1% accuracy. As a result, the difference from random for the Serre et al. algorithm is $\sim$ 43%. For our implementation with end-stopped filtering, random guessing would yield $\sim$ 20% accuracy, as a result our algorithm yielded a difference of $\sim$ 47%. The Δ from chance between our implementation and the Serre et al. implementation seems to be nearly negligible based on this comparison, and can probably be attributed to the difference in the number of classes.

Nonetheless, our implementation still performed nearly as well as the Serre et al. algorithm (due to the similarity of the two), and shows that following the model of a visual cortex system could in fact lead to realistic and usable results at accuracies in the 90% range. Our implementation seemed to perform better on images with more pronounced corners (for example, airplanes and motorcycles), which is probably an indication that the end-stopped features are creating a sort of outline based on corners for the objects.

6. Future Work

There are several issues with biologically inspired vision systems, which must be addressed for them to be viable in real time applications such as robotics. The first of these is the speed of learning and recognition. The feed-forward convolution system put forth by Serre et al. and modified here is much too slow for a real-time recognition. However, Terzić et al. [14] show that a different approach can be taken for feature extraction that, with parallelization, can be used for real time recognition. They show that you

can combine the Gaussian filter (Eq. 1) with the Double-Stopped kernel (Eq. 4) and an inhibition kernel to create a filter which quickly derives keypoints similar to the ones achieved through the multilayer system described here. Though they are missing the S2/C3 layer we present here, and Serre et al. present as S2/C2, they achieve achieve binary classification results in real time in the range of $\sim 70\%$. This is a much lower accuracy than both our method and Serre et al.'s method, however.

Future work which we would pursue would be to apply this "fast cortical keypoint"[14] method for early feature extraction and then to still attempt to learn feature representations of the images as in the S2/C3 layer. This way the accuracy would be improved and the performance wouldn't be affected by the convolution and Max-Pooling layers presented in S1/C1 and ES1/C2 here. Additionally, parallelization with CUDA or OpenCL could vastly improve the possibility for real time object recognition with our model.

A second problem with this current system is that though object recognition in the binary case can achieve results in the range of 90%, the multiclass application presented by Serre et al. and in this paper, still lag behind by a large percentage. Perhaps a better classifier could be used rather than an SVM for the multiclass approach, one that is more suitable for the Biologically-Inspired learning model. One such learning model is the Fuzzy ARTMAP presented by Carpenter et al. [24] This neural network model uses Adaptive Resonance Theory in an effort to reproduce the human learning process. Carpenter's later work [25] - as well as work from a past Comp 558 project - shows that the Fuzzy ARTMAP modified for object recognition can outperform a linear SVM by a large margin.

However, perhaps another approach which would be worth pursuing would be to use the feed-forward network to learn feature representations of objects using semi-supervised Deep Learning techniques as taken from Natural Language Processing methods shown by Socher et al. [26] In this method, they use a Recursive Neural Network to "parse" a segmented image and match a description to the image composition. They do this by creating a tree of the segmented parts and attempting to label parts in the tree using a reconstruction error. Though this is not directly the problem of object recognition, it is possible to modify this system to try and segment an image into sections in a similar unsupervised manner and then to use Terzić et al.'s fast cortical features to learn representations at each node in the parsed image tree. This is analogous to learning word vector representations for sentiment analysis or natural language parsing. This would perhaps perform much faster and more

accurately for the multiclass case and trying to locate an object accurately in a cluttered scene. Additionally, from this, an unsupervised system could be created where a program could learn representations of objects though it does not have labels for them.

Returning to our own method, further work to be done includes: parallelizing the system, applying a window based method to try and locate an object in a scene, and attempting to learn a representation of the image without the time consuming patch based methodology of the S2/C3 layer. It may be worth removing the final layer to try and maximize processing time if the cost of accuracy is only a few percentage points, so this is something we would like to test. Additionally, following on the work of Terzić et al. in incorporating other features from video instead of still images, including optical flow, to truly mimic a full human object recognition system would be the ideal next step in creating a truly human-like visual system.[15]

References

[1] H. Bay, A. Ess, T. Tuytelaars, L. V. Gool, Speeded-up robust features (surf), Computer Vision and Image Understanding 110 (2008) 346 – 359. Similarity Matching in Computer Vision and Multimedia.

[2] D. H. Hubel, T. N. Wiesel, Receptive fields, binocular interaction and functional architecture in the cat's visual cortex, The Journal of physiology 160 (1962) 106.

[3] J. P. Jones, L. A. Palmer, An evaluation of the two-dimensional gabor filter model of simple receptive fields in cat striate cortex, Journal of Neurophysiology 58 (1987) 1233–1258.

[4] N. Pinto, D. D. Cox, J. J. DiCarlo, Why is real-world visual object recognition hard?, PLoS Comput Biol 4 (2008) e27.

[5] E. Hildreth, Theory of edge detection, Proceedings of Royal Society of London 207 (1980) 9.

[6] J. G. Daugman, et al., Uncertainty relation for resolution in space, spatial frequency, and orientation optimized by two-dimensional visual cortical filters, Optical Society of America, Journal, A: Optics and Image Science 2 (1985) 1160–1169.

[7] T. Serre, L. Wolf, T. Poggio, Object recognition with features inspired by visual cortex, in: Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on, volume 2, pp. 994–1000 vol. 2.

[8] T. Serre, L. Wolf, S. Bileschi, M. Riesenhuber, T. Poggio, Robust object recognition with cortex-like mechanisms, Pattern Analysis and Machine Intelligence, IEEE Transactions on 29 (2007) 411–426.

[9] D. H. Hubel, T. N. Wiesel, Receptive fields and functional architecture in two nonstriate visual areas (18 and 19) of the cat., Journal of neurophysiology (1965).

[10] H. Kato, P. Bishop, G. Orban, Hypercomplex and simple/complex cell classifications in cat striate cortex., Journal of Neurophysiology 41 (1978) 1071–1095.

[11] F. Heitger, L. Rosenthaler, R. Von Der Heydt, E. Peterhans, O. Kübler, Simulation of neural contour mechanisms: from simple to end-stopped cells, Vision research 32 (1992) 963–981.

[12] A. Dobbins, S. W. Zucker, M. S. Cynader, Endstopped neurons in the visual cortex as a substrate for calculating curvature, Nature 329 (1987) 438–441.

[13] M. Farrajota, M. Saleiro, K. Terzić, J. Rodrigues, J. du Buf, Multi-scale cortical keypoints for realtime hand tracking and gesture recognition, in: Proc. 1st Int. Workshop on Cognitive Assistive Systems, IROS 2012, Vilamoura, pp. 9–15. ISBN 978-972-8822-26-2.

[14] K. Terzić, J. M. Rodrigues, J. H. du Buf, Real-time object recognition based on cortical multi-scale keypoints, in: Pattern Recognition and Image Analysis, Springer, 2013, pp. 314–321.

[15] K. Terzić, D. Lobato, M. Saleiro, J. Martins, M. Farrajota, J. Rodrigues, J. du Buf, Biological models for active vision: Towards a unified architecture, in: M. Chen, B. Leibe, B. Neumann (Eds.), ICVS 2013, LNCS, volume 7963, Springer, St. Petersburg, Russia, 2013, pp. 113–122.

[16] S. Marčelja, Mathematical description of the responses of simple cortical cells*, JOSA 70 (1980) 1297–1300.

[17] R. L. De Valois, D. G. Albrecht, L. G. Thorell, Spatial frequency selectivity of cells in macaque visual cortex, Vision research 22 (1982) 545–559.

[18] R. L. De Valois, E. William Yund, N. Hepler, The orientation and direction selectivity of cells in macaque visual cortex, Vision research 22 (1982) 531–544.

[19] T. Serre, M. Riesenhuber, Realistic modeling of simple and complex cell tuning in the hmax model, and implications for invariant object recognition in cortex. Massachusetts Institute of Technology, Cambridge, MA, Technical Report, CBCL Paper 239/AI Memo 2004-017, 2004.

[20] J. Rodrigues, J. H. du Buf, Multi-scale keypoints in v1 and beyond: object segregation, scale selection, saliency maps and face detection, BioSystems 86 (2006) 75–90.

[21] M. Riesenhuber, T. Poggio, Hierarchical models of object recognition in cortex, Nature neuroscience 2 (1999) 1019–1025.

[22] T. Poggio, E. Bizzi, Generalization in vision and motor control, Nature 431 (2004) 768–774.

[23] R. Fergus, P. Perona, A. Zisserman, Object class recognition by unsupervised scale-invariant learning, in: Computer Vision and Pattern Recognition, 2003. Proceedings. 2003 IEEE Computer Society Conference on, volume 2, IEEE, pp. II–264.

[24] G. A. Carpenter, S. Grossberg, N. Markuzon, J. H. Reynolds, D. B. Rosen, Fuzzy artmap: A neural network architecture for incremental supervised learning of analog multidimensional maps, Neural Networks, IEEE Transactions on 3 (1992) 698–713.

[25] G. A. Carpenter, W. D. Ross, Art-emap: A neural network architecture for object recognition by evidence accumulation, Neural Networks, IEEE Transactions on 6 (1995) 805–818.

[26] R. Socher, C. C. Lin, A. Ng, C. Manning, Parsing natural scenes and natural language with recursive neural networks, in: Proceedings of the 28th International Conference on Machine Learning (ICML-11), pp. 129–136.