LA-UR- 12-00096

Approved for public release;
distribution is unlimited.

Title:	Combining Multiple Visual Processing Streams for Locating and Classifying Objects
Author(s):	DM Paiton, SP Brumby GT Kenyon, GJ Kunde KD Peterson, MI Ham PF Schultz, JS George
Intended for:	IEEE Southwest Symposium on Image Analysis and Interpretation 2012

Los Alamos
NATIONAL LABORATORY
EST. 1943

Form 836 (7/06)

Combining Multiple Visual Processing Streams for Locating and Classifying Objects

DM Paiton[1,2], SP Brumby[1], GT Kenyon[1,2], GJ Kunde[1,2], KD Peterson[2], MI Ham[1], PF Schultz[2], JS George[1]

[1]Los Alamos National Laboratory
Los Alamos, New Mexico, USA
paiton@lanl.gov, jsg@lanl.gov

[2]New Mexico Consortium
Los Alamos, New Mexico, USA

***Abstract*—Automated, invariant object detection has proven a substantial challenge to the artificial intelligence research community. In computer vision, many different benchmarks have been established using whole image classification based on datasets that are too small to eliminate statistical artifacts. As an alternative, we used a new dataset consisting of ~66GB of compressed high-definition aerial video, which we employed for both object classification and localization. Our algorithm mimics the processing pathways seen in the primate visual cortex by independently evaluating color/texture, shape/form, and motion. We then combine the data using a clustering technique to produce a final output in the form of labeled bounding boxes around the objects of interest in the video. The output format required a degree of localization that is not seen in whole-image classification problems. Our results are evaluated qualitatively and quantitatively using a scoring metric that assessed the overlap between our detections and the ground-truth.**

Keywords- NeoVision2; computer vision; visual cortex; viewpoint invariance; localization; classification; motion energy; optic flow; online k-means; lateral interactions; object detection

I. INTRODUCTION

The primate visual cortex has inspired computer vision models with state of the art performance in object identification tasks [1-6]. These models were primarily tested using whole-image classification on still image datasets [8,9], which contained statistical biases resulting from their small size. Although new datasets attempt to address these issues [7], the generality of previous results have been called into question [10-12]. It is desirable to compare performance evaluations on a well-annotated dataset that lacks the biases found in standard image sets. DARPA has provided a set that we believe is a step towards this goal for its NeoVision2 grand challenge. The set is a high-definition video sequence recorded from a helicopter that flew over the greater Los Angeles, California area with a ground sampling distance of 25-40 pixels/meter. The set is divided up into two parts: a training portion which contains 131 sequences totaling 47 minutes of video and a testing portion containing 125 sequences totaling 45 minutes of video. The objective of the challenge was to identify, localize, and label objects of interest within the video. Detections are recorded in the form of bounding boxes, with an associated confidence value for the object label. The objects we focused on were Cars, Planes, People, and Cyclists. A total of ten categories were labeled in ground truth data, annotated by human analysts.

As with many other computer vision models, we chose to base our algorithms on the only known system that has solved the problem: the visual cortex. The primate visual cortex utilizes different processing streams in order to evaluate motion, form, color, and texture. These streams are then combined to create a perception of the visual scene [13]. We emulate this process by independently evaluating texture/color cues, object shape, and motion. We then combine the data into a single coherent output using a clustering algorithm.

II. TEXTURE AND COLOR – PETA-SCALE ARTIFICIAL NEURAL NETWORK (PANN)

Semi-supervised Object Detection using Sparse Generative Cortical Models

Observations of sparse patterns of neuronal activation in the visual cortex have led to models based on sparse image-patch representations using adaptive, over-complete image feature dictionaries learned from data [6,14]. These models are generative, allowing reconstruction of the input image, and are compatible with hierarchical models of cortex (extending standard HMAX approaches [5]). They can also drive many-category classification of image patches for object detection within a large image frame.

For the NeoVision2 challenge, we modified PANN, our high-performance model [15] to learn a sparsifying over-complete color/texture feature dictionary for the dataset (see Fig. 1a). Our retinal model down-samples the input frame to remove video compression artifacts and reduce computational expense. Our primary visual cortex (V1) S-cell layer uses our learned dictionary to build a local sparse representation (corresponding to a cortical column), using a greedy matching pursuit algorithm. Our S-cell columns are very sparse, with typically <5% of local feature detectors active in any given column. However, they still allow for good reconstruction of the input image, in distinction to standard HMAX approaches (see Fig. 1b,c). Our V1 C-cell layer applies a local max pooling operation, producing a translation-tolerant representation of image patches. Even after pooling, the C-cell columns are quite sparse. Note that our model does not use frame differencing .This allows us to detect objects that are stationary within the frame.

Figure 1. From left to right: (a) Learned over-complete color/texture dictionary for the helicopter dataset. Our dictionary captures orientation edge structures and non-oriented color features. (b) Example image patch from a training sequence. (c) Reconstruction of image patch using local sparse representations over the learned dictionary.

For our object detection and classification step, we used a multi-category generative model based on k-means clustering of the sparse C-cell column responses. We trained this model in a semi-supervised way, allowing the image background to divide up into unlabeled categories that, on inspection, appear strongly correlated with naturally occurring background scene object categories, including tree foliage, grass, pavement, water, and beach (see Fig 2). Setting the number of background categories is a meta-learning task and we typically used 30 background categories. We then augmented this set of background categories with the target categories learned using the same sparsifying dictionaries for labeled image patches (i.e. we used supervised learning for target categories). The final image patch classifier algorithm is a Euclidean (L2) minimum distance classifier in this multi-category space of category mean vectors. This multi-category classifier is a small component of the overall computation time (which is dominated by the formation of sparse representations in V1 S-cell columns, and produces whole scene visualizations that provide additional contextual cues for object classification (e.g. allows correlation of cars with roads or boats with water). The utilization of these clues is discussed later.

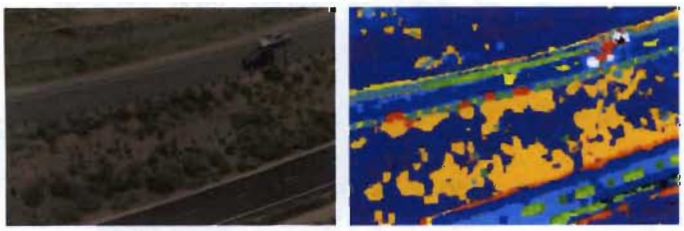

Figure 2. (a) Final car detections (purple boxes) for a training sequence. (b) Internal model of the frame, detecting cars (white pixels) as well as lane markings (yellow), grass (purple), bushes (orange), and road (blues). Detection of background features can be used to enhance detection of targets.

III. Form and Shape – PetaVision

Lateral Interactions based on Object-Distractor Difference (ODD) kernels

ODD kernels are intended to represent lateral interactions between cortical neurons located at the same cortical layer. Rather than learning a dictionary of features whose complexity increases as one travels up a cortical hierarchy, our implementation of lateral interactions uses a very simple set of features, corresponding to edge detectors spanning eight orientations between 0 and 180 degrees. The activation of these feature detectors is modulated by extensive lateral interactions based on co-occurrence of edges [16].

The technology underlying ODD kernels was developed in PetaVision, a massively parallel neural simulation toolbox for conducting high-performance simulations of large networks. The model was motivated and guided by a psychophysical object detection task employing computer-generated images. As depicted in Fig. 3, ODD kernels act to suppress edge features belonging to the background clutter while preserving edge features belonging to the target object.

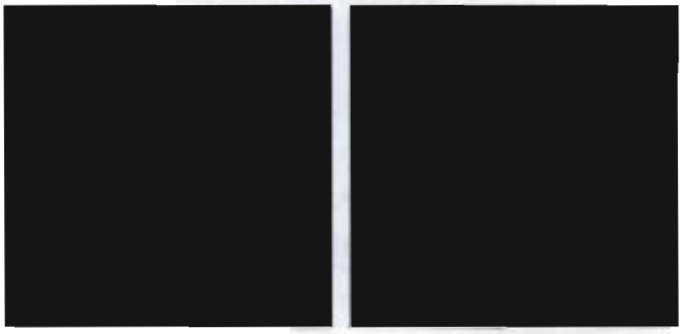

Figure 3. Representative example of ODD lateral-interaction kernels applied to computer-generated images. (a) The original image. (b) The processed image. Long, smooth contours tend to be preserved while short contour segments are suppressed.

For the above task, ODD lateral interaction kernels were trained by computing the co-occurrence matrix as a function of the distance and relative orientation between all pairs of edge features belonging to the target object and subtracting the co-occurrence matrix computed between all pairs of edges belonging to the background clutter. This difference was then normalized by the sum of the two co-occurrence matrices so that the elements of the ODD kernel were always between -1 and +1. Fig. 4 shows co-occurrence matrices and the resulting ODD kernel.

Figure 4. ODD Kernels trained to preserve smooth contours against background clutter. (a) Co-occurrence matrix obtained from all pairs of edges belonging to 10,000 target objects. (b) Co-occurrence matrix for edge pairs belonging to background clutter. (c) ODD kernel computed from the normalized difference between target and background co-occurrrence matrices.

The effective result from PetaVision was a preservation of edges associated with objects of interest. The algorithm was trained using ground-truth bounding boxes supplied by DARPA and distractor patches pulled from the image. ODD kernels were applied using standard convolution techniques to calculate lateral support for each feature. New ODD kernels were trained after each convolution to further reduce undesired edges and account for changes in the co-occurrence statistics. Fig. 5 shows that the number of active neurons decreased by two orders of magnitude with just four iterations of the kernels.

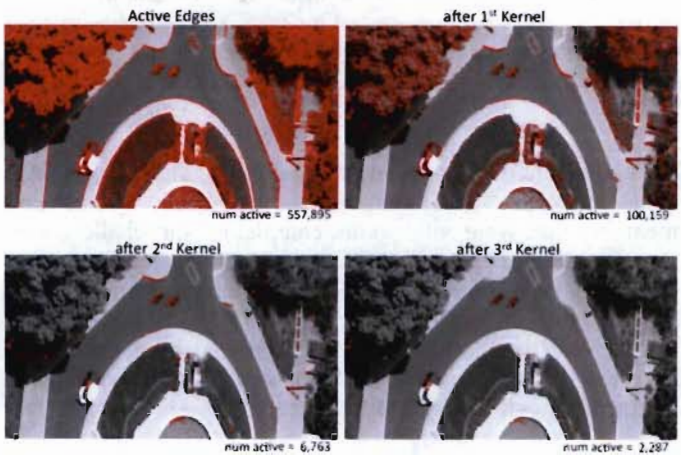

Figure 5. ODD Cyclist-distractor kernels applied to the NeoVision2 Tower dataset. "num active" indicates the number of active neurons in each snap shot. The tower dataset is from a still-camera, unlike the helicopter dataset which is explored in the rest of this document.

IV. MOTION – PETAMOTION

Most models of the mammalian motion-processing stream are based on motion-energy filters that mathematically describe the response properties of direction-sensitive V1 neurons but lack a clear physiological interpretation. Here, we implemented a more physiological model of V1 direction-selective cells that uses connection weights, realistic conduction delays, and separate excitatory and inhibitory channels [18]. Our neural network model exhibits similar responses to the mathematically defined motion-energy filters, but admits a more direct interpretation in terms of the underlying physiology and anatomy. The model, termed PetaMotion, was implemented using the PetaVision, toolbox previously described. Fig. 6 shows a simplified diagram of the neural processing pathway.

Our V1 neurons derived their spatiotemporal filter properties from their synaptic input weights, tuned such that the center frequency of the temporal band (ω_{t0}) divided by the center frequency of the spatial frequency band (ω_{x0}) give the velocity matched by the V1 cell. We implemented the spatial filter using a Gaussian-shaped excitatory connection surrounded by inhibitory Gaussian shaped flanks. To generate the temporal tuning properties of our V1 model, we used variable delays on retinal output (for simplicity, the LGN was omitted). The weights were calculated as 3D difference of Gaussians in X, Y, and time (1), where time refers to delayed retinal output (Fig. 7). We tune the response to a given velocity, equal to a displacement in X per time step by rotating with respect to the X and T axes. The Fourier transform of the V1 filter weights gives two Gaussian shaped regions oriented along a line whose slope is equal to the tuned velocity.

$$w(x,y,t) = e^{-\frac{(x')^2}{2\sigma_x^2}} e^{-\frac{y^2}{2\sigma_y^2}} e^{-\frac{(t')^2}{2\sigma_t^2}} - \frac{1}{2} e^{-\frac{(x'-x_0)^2}{2\sigma_x^2}} e^{-\frac{y^2}{2\sigma_y^2}} e^{-\frac{(t')^2}{2\sigma_t^2}} - \frac{1}{2} e^{-\frac{(x'+x_0)^2}{2\sigma_x^2}} e^{-\frac{y^2}{2\sigma_y^2}} e^{-\frac{(t')^2}{2\sigma_t^2}}$$
$$x' = x\cos\theta - t\sin\theta \; ; \; t' = t\cos\theta + x\sin\theta \qquad (1)$$

The V1 neuronal output was fed into a neural network implementation of the cortical area MT to calculate pattern motion utilizing the connection weights between the retina and simple cells [19]. The algorithm was MPI accelerated, with each frame broken into 8 regions that operated in parallel.

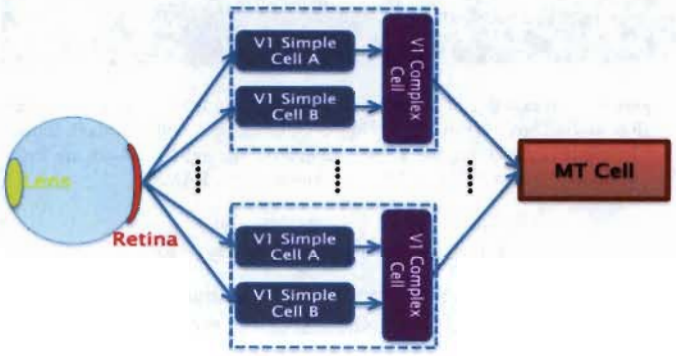

Figure 6. V1 cells act as spatio-temporal filters that respond to a specific optic flow velocity and direction. MT cells use the outputs of several V1 complex cells, tuned to different speeds and directions, to measure pattern velocity.

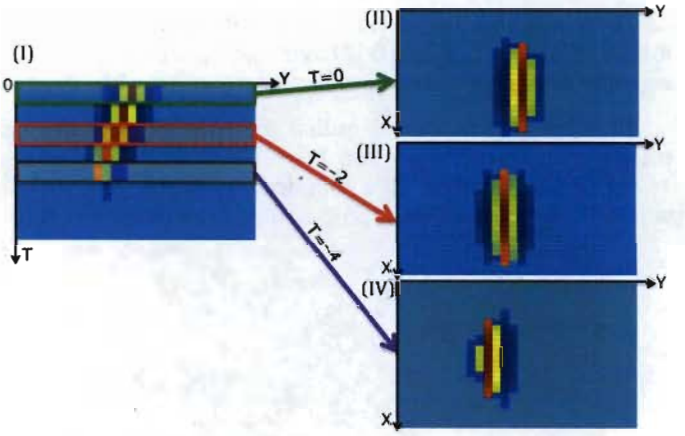

Figure 7. V1 simple cell weights plotted on the X, Y and T axes. (The time (T)-axis corresponds to delayed retinal output.) (I) Weight values plotted vs. Y and T. (II) Weights plotted on X-Y for time = 0. (III) Weights plotted on X-Y for time = -2. (IV) Weights plotted on X-Y for time = -4.

V. COLLECTIVE OUTPUT

PANN's primary output was in the form of rasterized PNG images where the pixel values represented the different categories. PetaMotion and PetaVision both created neuron activity files. For each frame in the video sequence, PANN gave a single output, while PetaMotion and PetaVision gave outputs per object of interest. The final step was to combine this information into a coherent and intelligible output.

This was accomplished using an adaptation of the DBSCAN algorithm [17]. The three primary inputs were formatted as points in a matrix that was the size of the video frame. Our algorithm formed clusters based on hits that were within a close proximity and of the same label. We then found

the minimum volume enclosing ellipsoid to get the appropriate size and orientation of the bounding box. Motion detections, which did not have a label, would modify confidence values. We also used contextual clues in the form of background labels

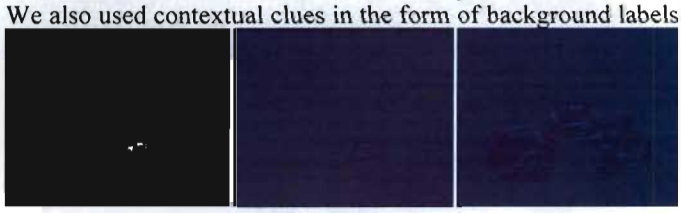

Figure 8. An example of the clustering algorithm output. (a) Original image with bounding box overlayed. (b) Bounding box output within a blank frame. (c) Example of clustering for the bounding box. The light blue lines are from PetaVision and the red dots are from PANN.

VI. Results and Discussion

Our algorithm was scored using qualitative measures as well as the DARPA established scoring metric. Their metric followed the equation:

$$\text{Overlap\_Ratio} = \frac{G_i^{(t)} \cap D_i^{(t)}}{G_i^{(T)} \cup D_i^{(t)}} \quad (2)$$

Where $G_i^{(t)}$ denotes the ith ground-truth object in frame t and $D_i^{(t)}$ denotes the ith detected object in frame t. From (2) you can conclude that the overall score will have a range from minus infinity to 1, where 0 is having no detections at all. This equation heavily penalizes for false detections (see Fig. 9).

In 1,994 frames, our algorithm scored a -3.38 in the Car category. In this category we had 2613 false negatives and 16733 false positives at 20% confidence. Our true positive rate was 0.39 and the number of false positives per frame was 8.39.

Figure 9. Example output. Every car has a detection on it with one false positive. Although this is a very desirable outcome, it did not score well due to the false positive.

VII. Concusion

In this paper we outline a method for detecting and localizing objects in areal video. The original incentive for the project was the DARPA NeoVision2 grand challenge. There were several other teams who participated in the challenge, who will hopefully soon publish comparable results.

Our neuromorphic algorithms emulated the visual processing streams found in the primate cortex. These streams emphasize texture/color, form and motion. We then combined the data stream outputs using a known clustering algorithm. from PANN to modify the confidence. For example, it is unlikely to see a car surrounded by sand, so car hits with sand background were given a lower confidence.

Acknowledgment

We would like to acknowledge our sponsors: DARPA, DOE, and NSF. In addition, we want to thank several team members that were vital to us completing the challenge on time: T Achler, D Moody, J Galbraith, E Raby, and Zhengping Ji. C Rasmussen helped develop the PetaVision framework. Finally, much of the work on PANN and PetaVision done prior to the NeoVision2 challenge was supported by a Laboratory Directed Research and Development (LDRD) project: Synthetic Cogniti led by LM Bettencourt.

References

[1] D.H. Hubel, T.N. Wiesel, "Receptive fields and functional architecture of monkey striate cortex", *J. Physiol.* (Lond.) 195, 215-243 (1968).

[2] K. Fukushima: Neocognitron, "A self-organizing neural network model for a mechanism of pattern recognition unaffected by shift in position", *Biological Cybernetics*, 36(4), pp. 193-202, April 1980.

[3] Riesenhuber, M. & Poggio, T., "Hierarchical Models of Object Recognition in Cortex", *Nature Neuroscience 2*: 1019-1025, 1999.

[4] T. Serre, A. Oliva and T. Poggio. "A feedforward architecture accounts for rapid categorization", *PNAS*, 104(15), p.6424-6429, 2007.

[5] T. Serre, L. Wolf, S. Bileschi, M. Riesenhuber, and T. Poggio, "Object recognition with cortex-like mechanisms", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 29(3), pp. 411-426, 2007.

[6] B.A. Olshausen and D.J. Field. "Sparse coding with an overcomplete basis set: a strategy employed by V1?", *VISR*, 1997.

[7] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. "ImageNet: a large-scale hierarchical image database". *CVPR*, pp. 710-719, 2009.

[8] Torralba, A. Fergus, R. and Freeman, W.T. "80 million tiny images: a large data set for nonparametric object and scene recognition. *IEEE TPAMI*, 30(11):1958-1970, 2008.

[9] G. Griffin, A. Holub, and P. Perona. "Caltech-256 object category dataset." Technical report, CalTech, 2007.

[10] W. Landecker, SP Brumby, M. Thomure, C Rinauldo, GT Kenyon, L Bettencourt, and M Mitchell. "Visualizing classification decisions of hierarchical models of cortex", *COSYNE*, Salt Lake City, Utah, 2010.

[11] N. Pinto, D.D. Cox, and J.J. DiCarlo. "Why is Real-World Visual Object Recognition Hard?" *PLoS Comp. Bio*, 2008

[12] N. Pinto, J.J. DiCarlo, D.D. Cox. "Establishing good benchmarks and baselines for face recognition." *ECCV*, 2008.

[13] DC Van Essen and JL Gallant, "Neural mechanisms of form and motion processing in the primate visual system." *Neuron*, Volume 13, Issue 1, Pages 1-10, July 1994.

[14] M Zeiler, G Taylor, R Fergus. "Adaptive deconvolutional networks for mid and high level feature learning." *IEEE ICCV*, Barcelona, 2011.

[15] SP Brumby, GT Kenyon, W Landecker, C Rasmussen, S Swaminarayan, and L Bettencourt. "Large-scale functional models of the visual cortex for remote sensing", 38th *IEEE AIPR*, Washington DC, Oct 2009.

[16] Gintuatas et al. "Model cortical association fields account for the time course and dependence on target complexity of human contour perception." *PLoS Comput. Biol.* 7(10), 2011.

[17] M Ester, HP Kriegel, J Sander, X Xu. "A density-based algorithm for discovering clusters in large spatial databases with noise". *Int'l Conf. on Knowledge Discovery in Databases & Data Mining*, Portland, OR, 1996.

[18] EH Adelson and JR Bergen. "Spatiotemporal energy models for the perception of motion." *J. of the Opt. Soc. Of America.* A 2.2 (1985): 284.

[19] Eero P. Simoncelli and David J. Heeger. "A model of neuronal responses in visual area MT." Vision Research 38.5 (1998): 743.