A Hardware Accelerated Multilevel Visual Classifier for Embedded Visual-Assist Systems

Matthew Cotter
Penn State University

Siddharth Advani
Penn State University

Jack Sampson
Penn State University

Kevin Irick
SiliconScapes LLC

Vijaykrishnan Narayanan
Penn State University

ABSTRACT

Embedded visual assist systems are emerging as increasingly viable tools for aiding visually impaired persons in their day-to-day life activities. Novel wearable devices with imaging capabilities will be uniquely positioned to assist visually impaired in activities such as grocery shopping. However, supporting such time-sensitive applications on embedded platforms requires an intelligent trade-off between accuracy and computational efficiency.

In order to maximize their utility in real-world scenarios, visual classifiers often need to recognize objects within large sets of object classes that are both diverse and deep. In a grocery market, simultaneously recognizing the appearance of people, shopping carts, and pasta is an example of a common diverse object classification task. Moreover, a useful visual-aid system would need deep classification capability to distinguish among the many styles and brands of pasta to direct attention to a particular box. Exemplar Support Vector Machines (ESVMs) provide a means of achieving this specificity, but are resource intensive as computation increases rapidly with the number of classes to be recognized. To maintain scalability without sacrificing accuracy, we examine the use of a biologically-inspired classifier (HMAX) as a front-end filter that can narrow the set of ESVMs to be evaluated. We show that a hierarchical classifier combining HMAX and ESVM performs better than either of the two individually. We achieve 12% improvement in accuracy over HMAX and 4% improvement over ESVM while reducing computational overhead of evaluating all possible exemplars.

1. INTRODUCTION

Vision assist systems (VAS) are of high utility and demand across many domains such as retail, security, automotive. Recent work has shown how wearable devices like Google Glass can be used to assist users in cognitive decline [1]. A key component in such systems is detecting important regions of interest (RoI) from the scene and then classifying them accurately in real-time. For example, a VAS designated to help a visually impaired person cross an intersection would require the system to (1) detect objects of interest; (2) classify those objects as impeding car, traffic light, people, obstructions; and (3) generate navigation plans for the user. Given the scheduled tasks at hand, such a system imposes stringent constraints on latency, accuracy, and energy; all of which are difficult for general purpose architectures to satisfy simultaneously. Recently, hardware accelerators have shown significant ability to meet these constraints albeit in a restricted environment [2, 3, 4].

The aim of this paper is to provide a scalable visual accelerator framework flexible enough to be applied to various domains. We focus on the application of visual recognition in a retail space to highlight the efficacy of our design proposal. The key contributions of this paper are:

- First, we discuss the trade-off in using a single level object classification module in large scale real-world applications. We also show the disadvantages of using a fine grained SVM-based approach to tackle the same problem.
- We propose a multi-level hierarchical approach towards identifying objects of different shapes, sizes and variety.
- We then evaluate our design on a significantly large dataset that captures the variance seen in a real world scenario.
- Finally, we suggest ways to enhance our visual assist pipeline in terms of robustness, accuracy and data dependence.

2. RELATED WORK

There has been considerable activity in the hardware accelerator space with regard to building ubiquitous real-time learning machines. In [5], the authors propose an architecture for Convolutional Neural Networks (CNNs) and Deep Neural Networks (DNNs) that minimizes memory transfers thus achieving high throughput with small area, power and energy footprint. In [6], a multi-FPGA emulation platform is evaluated for the purpose of detecting and recognizing objects of interest using Saliency and HMAX respectively. In [7], the authors propose a multi-core system with a set of customized functional units tightly coupled to the pipeline to speed-up computer vision algorithms. The impact of using a server that uses a heterogeneous architecture to provide low power, high throughput and application specific accelerators for large-scale recognition is studied in [8].

Different kinds of datasets have been used to benchmark the performance of these computer vision models. The Caltech dataset consisting of 101 categories is a popular dataset used for evaluating various object recognition models [9]. When evaluated on this dataset, the HMAX model was able to achieve 56% accuracy. While the Caltech 101 dataset is diverse in terms of the number of

ICCAD 2014 San Jose, California USA

978-1-4799-6278-5/14/$31.00 ©2014 IEEE

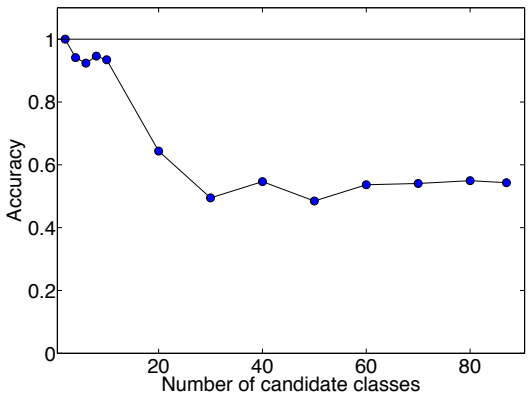

Figure 1: Feature-based Classification becomes more difficult as the number of candidate classes increases

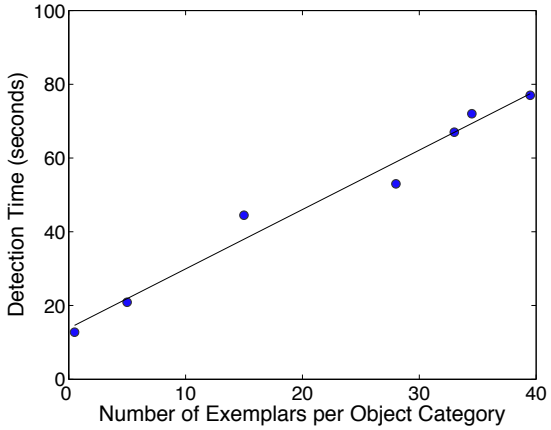

Figure 2: Object Identification takes longer as number of Exemplars increases

categories, it is not deep enough. For example, there are around 41 different birds and animals but when it comes to other categories, it is hard to create a super-group for them. In the recent past, datasets like PASCAL Visual Object Classes (VOC) Challenge [10] and ImageNet [11] have become immensely popular too.

3. MOTIVATION

The human brain is in a continuous process of adaptive learning based on the world it perceives. While the theory behind the human brain is still an active thrust of scientific research, neuroscientists are slowly pushing the envelope in understanding the actual workings of this intricate part of the human anatomy. The human visual system (HVS) is considered one of the better-understood pieces in this massive network of neuronal activity. Computational algorithms that model simple and complex cells of the visual cortex have been developed and mapped to computers, thus making them capable of emulating humans in processing visual information. Although many cognitive tasks have been successfully applied to real-time systems in the recent past, the resources required to accomplish this are very high compared to that used by the human brain. Thus, as the gap between state-of-the art neuromorphic visual models and the way humans perceive the world gradually diminishes, the power and resources needed to actualize these models in real-time dynamic autonomous systems become an increasingly difficult challenge.

The HMAX model is one such hierarchical system based on the ventral stream of the human visual cortex [12]. This model builds a "bag of features" representation of an object by projecting an image-pyramid through a feed-forward system of alternating layers of simple "S" cells and complex "C" cells. The image pyramid structure provides scale invariance to the extracted features. The simple cells in the hierarchy are responsible for recognizing key aspects or features of the visual data. The cells of the S1 layer are tuned to respond to edges tuned to a variety of orientations. The cells of the C1 layer pool these responses across neighboring scales. The S2 layer – the heart of the model – processes the results of the C1 layer through a pre-learned dictionary of feature prototypes which are far more descriptive than the simple edge features detected by the S1 layer. Finally, the responses to each dictionary feature are pooled across scales and orientations to produce the final feature vector.

These features can then be used to train any desired classification scheme for recognition of novel objects that appear during operation. Regardless of the scheme used, the ability of a classifier to discriminate amongst classes is reduced as the number of candidate classes increases. In a real-world system, the number of object classes that may need to be identified is enormous, and despite showing reasonable classification using HMAX features, the level of overall accuracy leaves much to be desired in the context of a day-to-day visual assist system where accuracies are expected to be very high. Figure 1 shows the trade-off between accuracy and scalability for the HMAX model. There is a steep roll-off in accuracy after 10 classes and then the model seems to stabilize after 30 classes. Even though HMAX may not be able to hit the exact target class with sufficient accuracy, the resulting classification of HMAX features can significantly narrow the field of potential targets.

The Exemplar SVM (ESVM) model uses the Histogram of Oriented Gradients (HOG) feature as a building block. HOG is based on gradient computation followed by a non-linear weighted voting operation based on the gradient magnitude. Votes are accumulated into orientation bins over local spatial cells. These cells are grouped into blocks over which contrast normalization is carried out. The final HOG descriptor is a vector of all components of the normalized cell responses from all of the blocks in the detection window [13]. Each exemplar is essentially built by combining a number of these HOG features together. The model is then formed by training a separate linear SVM for each exemplar (representation) in the training set. Each of these ESVMs is thus defined by a single positive instance and numerous negatives [14]. An ensemble of exemplars per category can then be used to classify or detect a particular object. This ensemble can at times get extremely large. Being able to classify a wide number of classes would require to pool over the entire ensemble, which would entail a lot of time and/or resources. Figure 2 shows the trade-off between the time taken to classify/detect an object and the number of exemplars per object category.

While there is still considerable research being carried out in the computer vision fraternity to build more scalable and robust models, our focus here is to design a visual assist system that is able to achieve high accuracy while still being able to classify a large number of classes efficiently. We propose an architecture that marries the highly parallelizable HMAX with the immensely detailed ESVM to establish a pipeline that can accurately classify a significantly large number of objects set in a real world scenario with a high degree of accuracy.

4. SYSTEM ARCHITECTURE

Figure 3 illustrates the proposed hierarchical classification system. RoI extraction is performed by a collection of segmentation engines that are appropriate for the application domain. In the case of visual assistance in grocery environments, RoI extraction may be performed by saliency [15, 16, 17], objectness [18], and symmetry [19]. A fixed number of RoIs are selected and prepared for processing by the classification cascade. The selection scheme is application-specific, however a useful scheme might be to select the RoI that has the highest information-bearing potential once classified. For example, an RoI that exhibits a tall bottle-like silhouette may be more informative, once classified, than an object that has a short box silhouette. Under the assumption that there are fewer locations in a grocery store that contain tall bottle-like objects versus short box objects, the selection scheme would choose the RoI of the former in an effort to allow the classifier to quickly discriminate the scene location or context. Once selected, an RoI undergoes pre-processing in preparation for evaluation by the first stage classifier. In the proposed architecture, the first stage classifier, HMAX, expects a fixed size window as input. To accommodate this, the pre-processing stage performs image resizing as necessary.

HMAX performs the first stage classification and the reader is encouraged to review [3] for details on the micro-architecture of a digital accelerator. The primary architectural component in the HMAX accelerator is a bank of streaming correlation engines that accelerate the expensive S2 stage. In the S2 stage, a multi-scale and multi-feature pyramid of the input image is correlated with a large set – roughly 5000 – templates to produce an N-dimensional feature descriptor of the input image. This descriptor is subsequently classified using a Regularized Least Squares (RLS) approach. Acceleration in the S2 stage is achieved by concurrently correlating each template with several scale and feature images within the pyramid. The correlation accelerator engine is duplicated to trade-off classification latency and on-chip resource utilization.

Once the C2 stage produces the N-dimensional feature descriptor, a RLS classifier produces a C-dimensional score vector, where C equals the number of classes under consideration. Each entry C_i in the vector represents the likelihood that the descriptor, and the image from which it was derived, belongs to the i^{th} class. The assignment of these scores requires that an offline learned coefficient matrix of dimension CxN is multiplied by the N-dimensional descriptor. In total, C dot products between two N-dimensional vectors are performed.

The second stage classifier consists of a bank of Support Vector Machines. This work follows the ESVM classification approach that achieves high specificity at the expense of execution latency. For each class under consideration, the ESVM learning process identifies E exemplary instances of the class in the training set. These exemplars are representative instances of the class and collectively define the positive models of the class. All other training instances not belonging to the class under consideration, represent the negative models. Determining if a novel object belongs to a given class requires that for each exemplar in the class, a distance function be computed between the object and the exemplar's single positive instance and its set of negative instances. In this work we compute the L2 norm distance *d* between normalized HOG representations of the input image and exemplar models. We employ vector norm accelerators to compute:

$$d = \sqrt{\sum_{i=1}^{N}(a_i - p_i)^2} - \sum_{n}^{n \in Negatives} \sqrt{\sum_{i=1}^{N}(a_i - n_i)^2} \qquad (1)$$

where *a* is the feature representation of the test object to be classified, *p* is the feature representation of the positive exemplar, *n* is the feature representation of the negative exemplar and *N* is the dimension of the feature.

5. RESULTS

5.1 Methodology

We evaluated our hierarchical object recognition model on a significantly large and deep dataset consisting of images of products one would typically come across in a grocery store. 87 products categorized under eight shapes were listed. After we generated this list, we used the Microsoft Bing Search API to download all the images. Each product class was manually pruned to remove images that did not represent a good example of the given product. Any image that either did not contain the example class, or contained the example class with another example class (i.e. if an image of Tide Laundry detergent also had an image of Gain detergent) was not included. Post-pruning, approximately 7700 objects were available to train HMAX and Exemplar SVM independently. Table 1 shows the breakdown of different categories used in our evaluations. Figure 4 illustrates examples of exemplars pertaining to each category. Row 1 shows exemplars from shape category box, condimental, jug and bottle, while Row 2 shows exemplars from shape category packet, bar, can and jar.

Shape	Products
Bar	3
Bottle	8
Box	21
Can	23
Condimental	7
Jar	10
Jug	9
Packet	6
Total	**87**

Table 1: Grocery Database

5.2 Evaluation

To ensure that the training and test datasets were separate and distinct, the test images consisted of rectangular RoIs manually designated in images captured from two different grocery stores. These bounding boxes were manually annotated using the LabelMe tool [20]. Approximately 800 images were used in the testing phase. Figures 5 and 6 highlight the accuracies obtained using HMAX and ESVM respectively. The total accuracy (total correct/total tested) for HMAX is around 47% while that for ESVM is around 55%. Figure 7 provides the accuracy curve for HMAX-top-K where K is number of categories that HMAX provides with a confidence that the correct category is one of them. We choose K=11 (HMAX includes the right category in the 11 categories sent to ESVM 80% of the time). This relaxes the computational overhead on ESVM considerably since now ESVM needs to evaluate exemplars from 11 of the 87 categories. Figure 8 shows the accuracies obtained when using this joint-classifier approach. We obtain a total accuracy of around 59% in this case which is around 12% more than HMAX and around 4% more than ESVM. Note that the performance of EVSM can be improved as more exemplars are extracted from a more expansive dataset. Here, the pre-classification performance of HMAX ensures that the increase in exemplars is not detrimental to the runtime performance of the system.

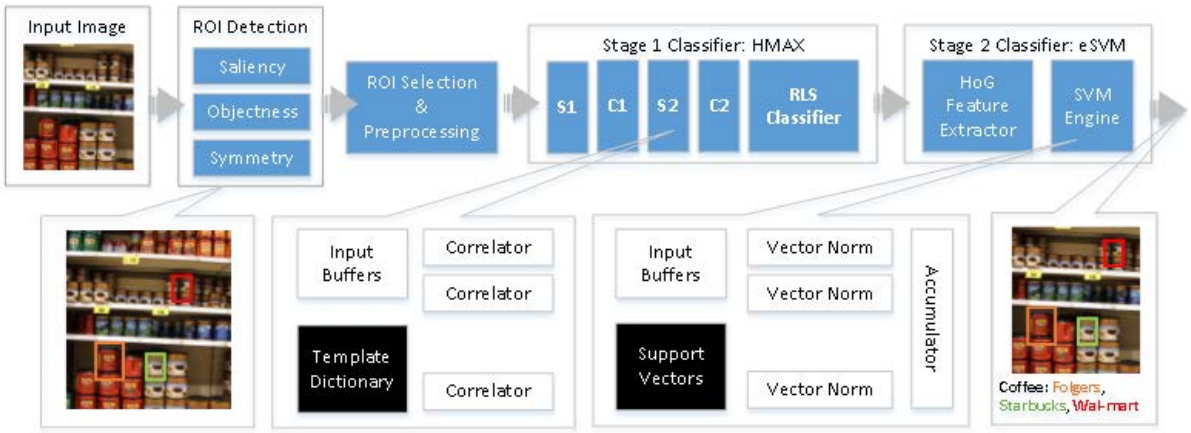

Figure 3: System Architecture

Figure 4: Exemplars from Training Dataset

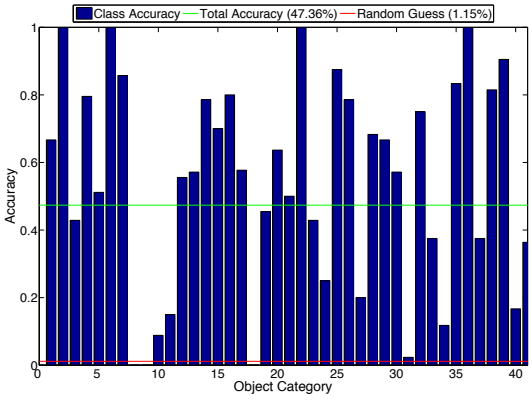

Figure 5: HMAX Performance on Test Dataset

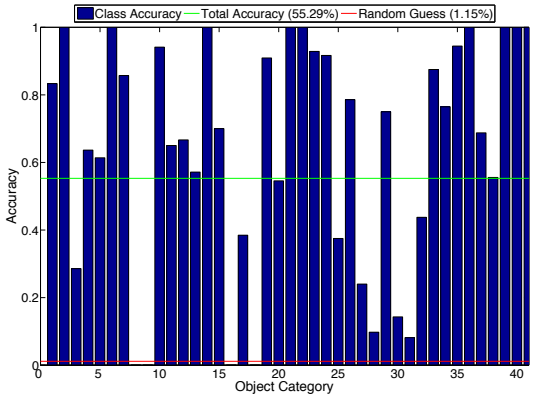

Figure 6: Exemplar Performance on Test Dataset

6. CONCLUSION

In conclusion, we show that multi-level filtering is 12% better than HMAX and 4% better than ESVM individually. This multi-level approach not only reduces computational overheads on ESVM, but also allows for an opportunity to trade-off between resource utilization and accuracy. Future work entails looking at a three-tier approach to be able to distinguish between varieties of a particular product. Evaluation of pre-filters based on feature extractors other than HMAX must be explored as well. In addition, our evaluations on ESVM are conservative since we cap the number of exemplars per product category to 40 in order to reduce the training overheads. We also use a simple raw detection score to classify the category rather than a more sophisticated normalization scheme across categories. Ways to improve the performance of ESVM to achieve better accuracies are thus necessary but beyond the focus of this paper. To the best of our knowledge, this is the first work that has used a multi-level classification approach when confronted with a real-world dataset that consists of deep and diverse object classes.

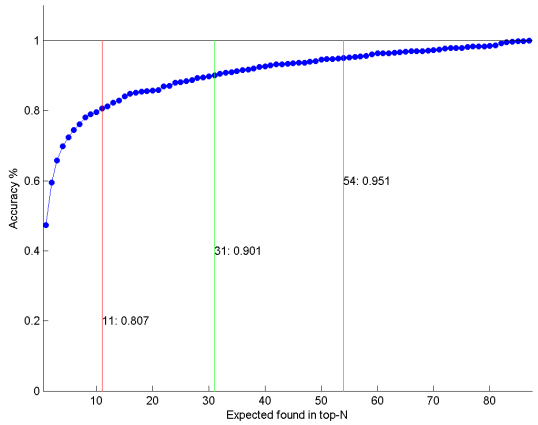

Figure 7: HMAX Top K

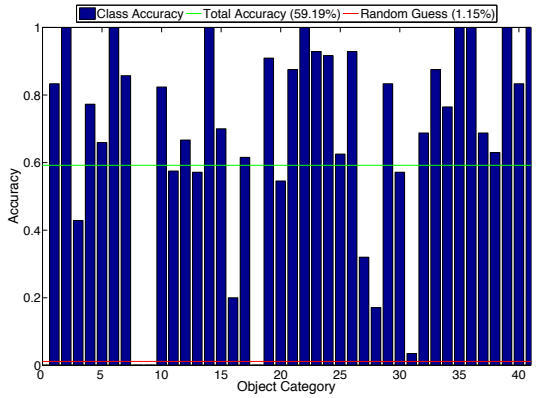

Figure 8: HMAX-Exemplar Performance on Test Dataset

7. ACKNOWLEDGEMENTS

This work is supported in part by NSF Expeditions: Visual Cortex on Silicon CCF 1317560, Office of Naval Research under grant number N00024-12-D-6404 DO0194 and SBIR grant N00014-14-P-1183. The work is also supported by a gift from Intel and infrastructure provided by NSF Award 1205618.

8. REFERENCES

[1] K. Ha et al. Towards wearable cognitive assistance. In *Int. Conf. on Mobile Systems, Applications, and Services*, MobiSys, pages 68–81, 2014.

[2] S. Bae et al. An FPGA Implementation of Information Theoretic Visual-Saliency System and Its Optimization. *International Symposium on Field-Programmable Custom Computing Machines*, pages 41–48, May 2011.

[3] A.A Maashri et al. Accelerating neuromorphic vision algorithms for recognition. In *Design Automation Conference*, page 579, 2012.

[4] J. Sabarad et al. A reconfigurable accelerator for neuromorphic object recognition. In *ASP-DAC*, pages 813–818, 2012.

[5] T. Chen, J. Wang, Y. Chen, and O. Temam. DianNao : A Small-Footprint High-Throughput Accelerator for Ubiquitous Machine-Learning. In *International Conference on Architectural Support for Programming Languages and Operating Systems*, 2014.

[6] S. Kestur et al. Emulating Mammalian Vision on Reconfigurable Hardware. In *Int. Symp. on Field-Programmable Custom Computing Machines*, April 2012.

[7] J. Clemons, A Pellegrini, S. Savarese, and T. Austin. Eva: An efficient vision architecture for mobile systems. In *Compilers, Architecture and Synthesis for Embedded Systems (CASES), Int. Conf. on*, pages 1–10, Sep 2013.

[8] R. Iyer et al. Cogniserve: Heterogeneous server architecture for large-scale recognition. *Micro, IEEE*, May 2011.

[9] L. Fei-Fei, R. Fergus, and P. Perona. Learning generative visual models from few training examples: An incremental bayesian approach tested on 101 object categories. In *Computer Vision and Pattern Recognition Workshop, Conf. on*, pages 178–178, June 2004.

[10] M. Everingham, L. Van Gool, C. K. I. Williams, J. Winn, and A. Zisserman. The pascal visual object classes (voc) challenge. *International Journal of Computer Vision*, 88(2):303–338, June 2010.

[11] J. Deng et al. Imagenet: A large-scale hierarchical image database. In *Computer Vision and Pattern Recognition, IEEE Conference on*, pages 248–255, June 2009.

[12] J. Mutch and D. G. Lowe. Object class recognition and localization using sparse features with limited receptive fields. *International Journal of Computer Vision*, 2008.

[13] N. Dalal and B. Triggs. Histograms of oriented gradients for human detection. In *Computer Vision and Pattern Recognition, 2005. CVPR 2005. IEEE Computer Society Conference on*, volume 1, pages 886–893 vol. 1, June 2005.

[14] T. Malisiewicz, A Gupta, and AA Efros. Ensemble of exemplar-svms for object detection and beyond. In *Computer Vision (ICCV), 2011 IEEE International Conference on*, pages 89–96, Nov 2011.

[15] N. D Bruce and J. K Tsotsos. Saliency, Attention, and Visual Search: An Information Theoretic Approach. *Journal of Vision*, 9(3):5.1–24, January 2009.

[16] L. Itti, C. Koch, and E. Niebur. A model of saliency-based visual attention for rapid scene analysis. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 20(11):1254–1259, Nov 1998.

[17] T. Judd, K. Ehinger, F. Durand, and A. Torralba. Learning to Predict Where Humans Look. *2009 IEEE 12th Int. Conf. on Comput. Vision*, pages 2106–2113, Sep 2009.

[18] M-M Cheng, Z. Zhang, W-Y Lin, and P. H. Torr. BING: Binarized normed gradients for objectness estimation at 300fps. In *IEEE CVPR*, 2014.

[19] J. Liu and Y. Liu. Grasp recurring patterns from a single view. In *Computer Vision and Pattern Recognition (CVPR), 2013 IEEE Conference on*, pages 2003–2010, June 2013.

[20] B. Russell, A. Torralba, K. Murphy, and W. Freeman. Labelme: A database and web-based tool for image annotation. *Int. J. Comput. Vision*, May 2008.