Exploration of Hardware Acceleration for a Neuromorphic Visual Classification System

Ikenna J. Okafor, McNair Scholar
The Pennsylvania State University

McNair Faculty Research Advisors:
Kevin M. Irick, Ph.D
Research Associate
Department of Computer Science and Engineering
College of Engineering
The Pennsylvania State University

Vijaykrishnan Narayanan, Ph.D
Distinguished Professor of Computer Science & Engineering and Electrical Engineering
Department of Computer Science and Engineering
College of Engineering
The Pennsylvania State University

<u>*Abstract*</u>

Neuromorphic visual perception algorithms have become increasingly popular as they enable a wide array of vision based applications. HMAX is an example of a neuromorphic visual feature extraction algorithm that has been shown to perform well for large scale object and scene recognition tasks. While the accuracy of HMAX is considerable, its high computational latency makes it prohibitive for many real time applications. Hardware acceleration is a widely accepted technique for mitigating the computational latency of complex algorithms and has been investigated for HMAX specifically. However, prior investigations of hardware accelerated HMAX have not produced latencies that are suitable for large-scale real-time classification. Using a holistic approach, this work proposes both algorithmic optimizations and hardware customization techniques to accelerate HMAX beyond current state-of-art implementations. Results show confirmation of a future version of HMAX with potentially improved execution time, while still performing at a reasonable accuracy.

Introduction

Neuromorphic vision algorithms is a popular topic within computer vision. These algorithms mimic the way the mammalian visual cortex processes visual stimuli and have fostered a wide range of applications [1,2,3]. For example, utilizing the efficiency and robustness of neuromorphic algorithms, a visual prosthesis device can be engineered to augment the quality of life for visually impaired persons. A key task in such a system is object identification which relies on visual features to distinguish object classes. HMAX is one such neuromorphic algorithm that extracts visual features from an image for use in classification. HMAX is able to achieve considerable accuracy; however this accuracy comes at a high computational cost [4].

HMAX Model

HMAX is a four stage feature extraction model derived as an extension of the algorithm developed by Mutch & Lowe [5]. The model uses a combination of template matching, and maximum response collection to extract the edge-based features that are present within an image.

S0 Stage

This first stage acts as a preprocessing stage that creates a multiscale image pyramid from the original input image. This process is necessary to enable scale invariant feature extraction for objects that may appear at arbitrary sizes. In this work we extract features from twelve image scales.

S1 Stage

The S1 stage then takes all twelve scales and convolves it with an 11x11 Gabor filter to detect edges present within the image. The Gabor filter is described by Eq (1) where X = xcos(θ) + ysin(θ) and Y= -xcos(θ) + ysin(θ), x and y varies between -5 and 5, θ varies between 0 and π, and the wavelength (λ), width (σ), and aspect ratio (γ) are 5.6, 4.6, and 0.3, respectively.

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

C1 stage

The C1 stage utilizes a 10x10x2 3D max filter to find maximum responses from the Gabor convolution for each orientation across two different scales.

S2 stage

This stage of the HMAX model then takes a dictionary of prototypes and correlates every applicable prototype to the output of the C1 stage. Eq (2) describes this stage. The numerator of Eq (2) details the correlation of each C1 output, *X*, against a prototype, *P*, and, the accumulation of those responses across orientations. The denominator denotes the calculation of the normalization patch, where x_i is a single C1 output (n={4,8,12,16}; m=12). After the accumulation across orientations is finished pixel wise division is done with the normalization patch.

$$R(X, P) = \frac{X.P}{\sqrt{\sum x_i^2 - \frac{\left(\sum x_i^2\right)^2}{n^2 * m}}} \tag{2}$$

C2 stage

The C2 stage does a global pooling of all the S2 stage output data, by removing all spatial information and recording maximum responses across two sets of scales. Once these four stages have been completed the information is then passed to a classifier to detect the object within the image.

The Computational Cost of HMAX

The main bottleneck of the HMAX algorithm is in the S2 stage where the output from the C1 stage is correlated with the template dictionary [4]. The dictionary contains over five thousand prototypes which must be spatially correlated with each C1 output. Spatial correlation can be a costly process for a large number of inputs, as spatial correlation requires numerous multiplications and accumulations. But correlation in the Fourier Domain costs less, as those multiplications and accumulations now become single point-wise multiplications [6]. This project demonstrates using frequency correlation in the S2 stage to reduce execution time while still maintaining the baseline accuracy. In addition, this project also explores adding the spatial information originally removed in C2 stage, as adding spatial information along with a feature vector has shown to improve classification accuracy in other areas [7].

Methodology

The HMAX algorithm, written in C++ by Jim Mutch, served as the foundation of the experimentation. The project consisted of seven experiments total. The purpose of the first five experiments was to confirm our hypothesis of doing correlation in the Fourier domain, while still maintaining the baseline accuracy. The sixth and seventh experiment were used to evaluate the accuracy from adding spatial information to the final C2 vector.

Experiment 1: Fourier-A

The objective of this experiment was to perform a Fourier transform of the C1 output to do frequency correlation in the S2 stage, but then return back to the spatial domain to normalize the data as in the original algorithm. This information was then sent to the C2 stage for the global pooling.

Experiments 2-4: Fourier-B, Fourier-C, Fourier-D, & Fourier-E

These experiments again used frequency correlation in the S2 stage, but remained in the Fourier domain for the C2 stage. The data was normalized beforehand using an approximated normalization patch. This was done to test whether we could still achieve a reasonable accuracy without the calculated normalization patch. Four experiments were done to find the best value, which represents the maximum response from pooling in the Fourier domain. In *Fourier-B*, the final C2 vector was composed of the average power of each max Fourier coefficient [8]. *Fourier-C*’s final C2 vector was composed of the peak magnitude. *Fourier-D* and *Fourier-E*’s maximum response was composed of the max real and max imaginary value respectively.

Experiments 6-7: Spatial A and Spatial B

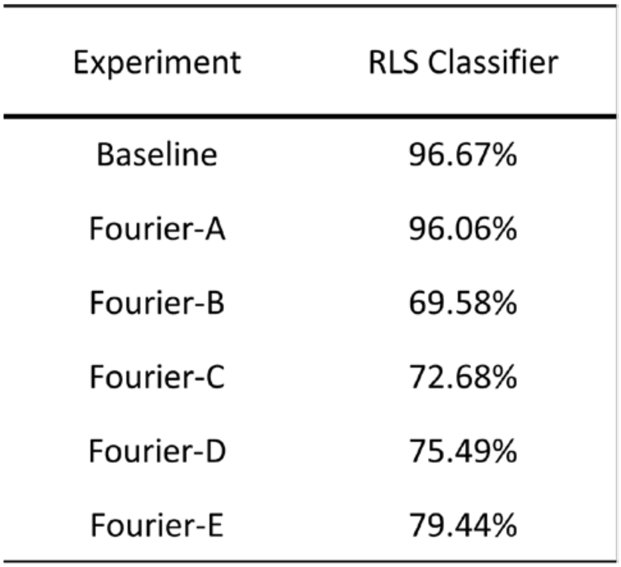
These two experiments were conducted to evaluate adding spatial information to the C2 output to improve classification accuracy. This was done in two ways using time-domain correlation. In experiment *Spatial A*, a linear combination of the (x,y) coordinate and the scale size was given along with the max response. In experiment *Spatial B*, the (x,y) coordinate pair was given along with the max response to evaluate the accuracy. The linear combination technique was done to minimize the amount of information given to the classifier, while still adding spatial info for better accuracy.

Ten classes of images from a dataset of grocery images were used for evaluation, with approximately 1200 images used for training, and 300 images used for testing. For each experiment the accuracy was recorded for comparison against the baseline HMAX implementation.

Three different classifiers were used for evaluation of experiments six and seven. This was done to gather data on the type of classifier which would yield the highest accuracy. The RLS and the SVM-linear classifier was used to evaluate the accuracy from using a linear classifier. The SVM-RBF was used to evaluate the accuracy from using a non-linear classifier. Since the purpose of the first five experiments was to confirm our hypothesis of doing the template correlation in the Fourier domain, classification was only done with the RLS classifier. After finishing the experiments, the next phase of the project was to model a hardware implementation with the information gathered from Experiments 1-5. This was done to observe a reasonable decrease in execution time from using frequency correlation in the S2 stage.

Results

Table 1 shows the results from the experiments *Fourier-A, Fourier-B, Fourier-C, Fourier-D, and Fourier-E.* The table shows the accuracies recorded from each individual experiment conducted using the RLS classifier. *Baseline* refers to the baseline HMAX accuracy for the ten classes of images.



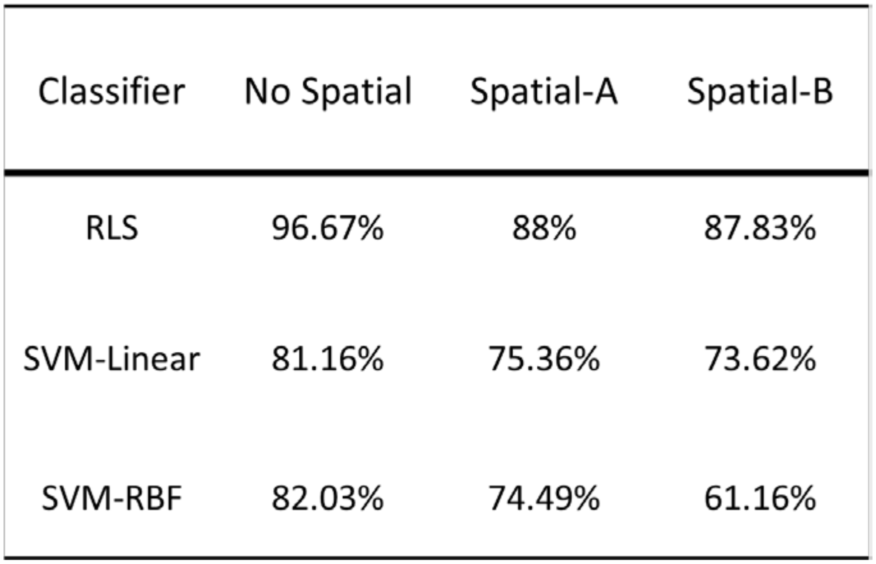
Experiment	RLS Classifier
Baseline	96.67%
Fourier-A	96.06%
Fourier-B	69.58%
Fourier-C	72.68%
Fourier-D	75.49%
Fourier-E	79.44%

Table 1. From the results it shows that we can achieve comparable accuracy to the baseline Accuracy using the architecture of Fourier-A

Table 2 shows the results from the experiments *Spatial-A and Spatial-B.* The table shows the accuracies recorded from each individual experiment conducted using a specific classifier. *No Spatial Data* refers to the baseline HMAX accuracy for the ten classes of images.

Discussion

With the results from experiments one through five, it can be seen that we can perform the S2 stage of HMAX in the Fourier domain, while still achieving considerable accuracy by doing the C2 stage in the time-domain. However, the low accuracies in experiments two through five may be attributed to the approximated normalization values used for these experiments. More favorable results may have been achieved if we had instead used the same normalization patch as in experiment *Fourier-A.* As for adding spatial info to the final C2 vector, in all cases, doing so decreased the accuracy considerably. This was most likely due to not enough variance between the data for any classifier to adequately make distinctions between different images.



Classifier	No Spatial	Spatial-A	Spatial-B
RLS	96.67%	88%	87.83%
SVM-Linear	81.16%	75.36%	73.62%
SVM-RBF	82.03%	74.49%	61.16%

Table 2. Adding spatial info to the final C2 vector in all cases hindered the classification accuracy.

Theoretical Architecture

A high-level architecture of the modified HMAX model was developed using the data gathered from Experiments 1-5. Equation 3 and Table 3 describe a model for the baseline HMAX architecture, as well as the modified architecture to conduct correlation in the Fourier-Domain. *S2(Baseline)* denotes the S2 stage with the baseline HMAX architecture and *S2(Fourier-A)* denotes the S2 stage with the modified Fourier HMAX architecture. The purpose of this model was primarily to observe the speed in the S2 stage from doing correlation in the Fourier domain. Therefore, the cost of going into and out of the Fourier domain was not included.

$$StageLatency = \frac{(\#OutputPixels * LatencyPerPixel + FSizeLatency) * (ZSize)}{ClockFrequency} \tag{3}$$

Stage	#OutputPixels	LatencyPerPixel (cycles)	FSizeLatency (cycles)	ZSize
S1	60516	241	0	12
C1	2209	99	0	12
S2(Baseline)	1936	511	21296	5120
S2(Fourier-A)	2209	1	24299	5120
C2(Baseline)	5120	11615	0	5120
C2(Fourier-A)	5120	13253	0	5120

Table 3. Listing of latency and output values for both the baseline and the modified architecture

Figures 1 and 2 detail a block diagram of architecture from using the modifications from Experiment *Fourier-A*. Theoretical execution times were drawn from both models using Eq (3) and Table 3 at a 100 MHz clock frequency,

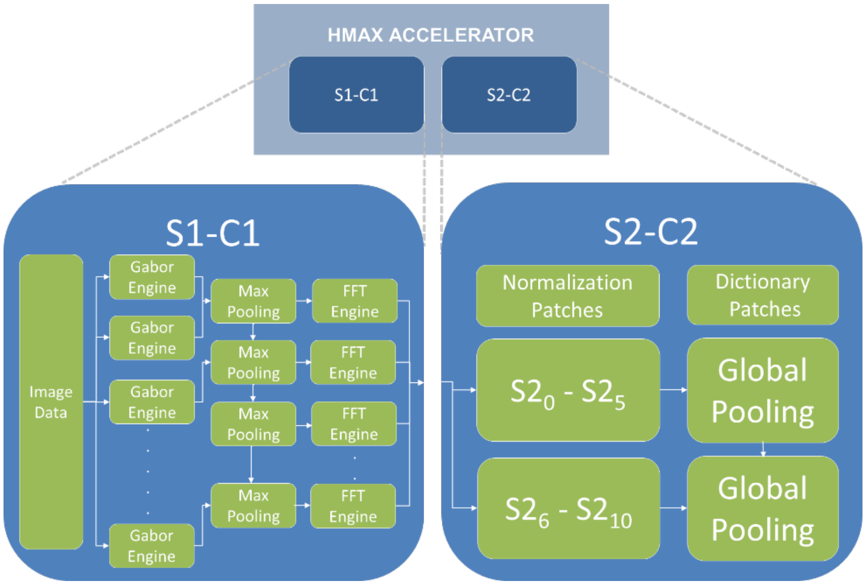


Figure 1. Depicts an overview of the S1-C1 stage and the S2-C2 stages. The S1-C1 stage consists of multiple Gabor engines each with their own max pooling unit

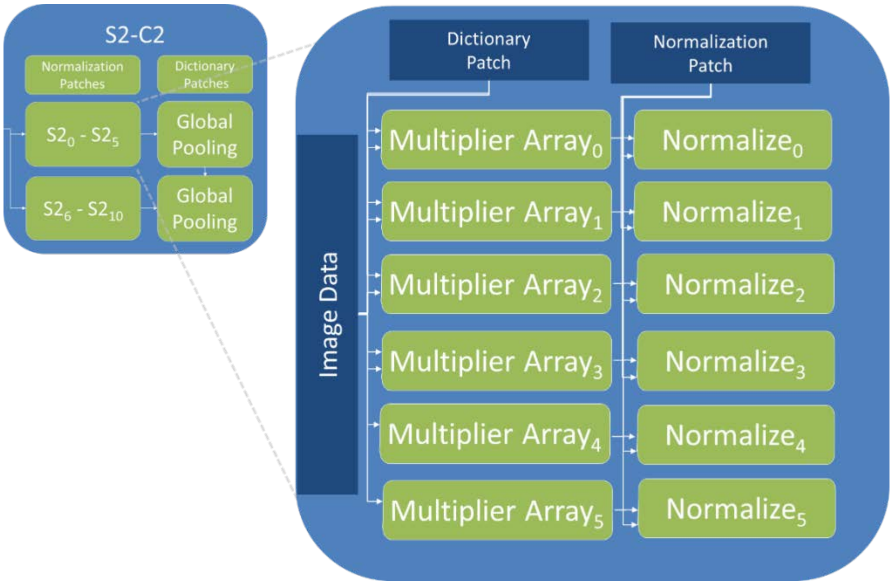
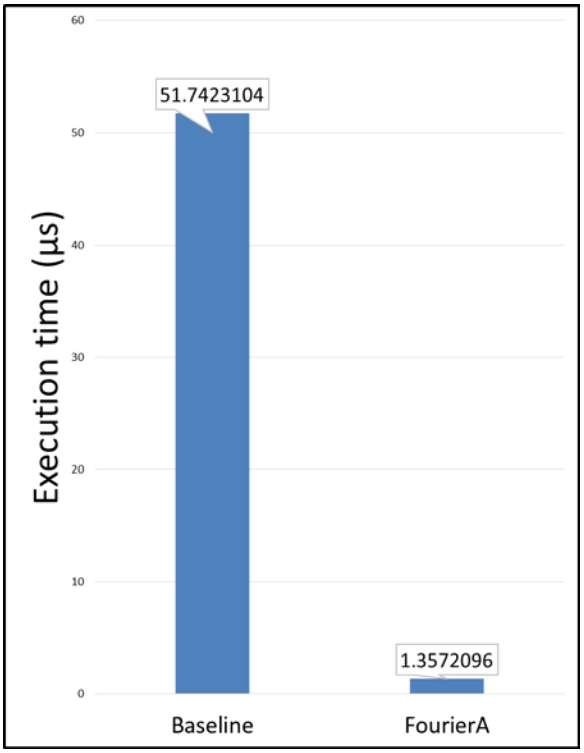


Figure 2. Depicts the S2 stage architecture, with several multiplier arrays

Memory resources were also recorded for both models. As can be noted from figures 3 and 4, Model *Fourier-A* has a 50x speedup versus the *Baseline* model, but on the other hand uses far more memory. This increase in memory consumption is due to extra padding needed for frequency correlation at various scales [6]. An alternative to this approach would be to convert the prototypes to the Fourier domain online, but this would require more DSP resources.



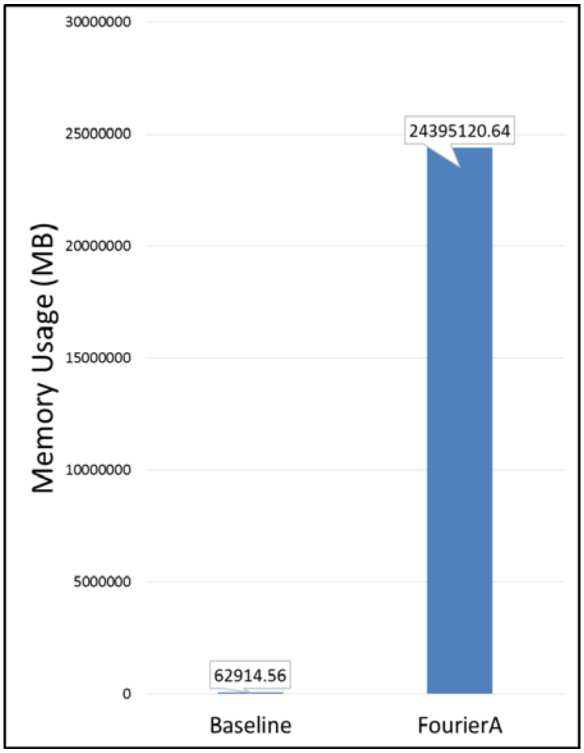


Figure 3. Graph shows a 50x Speedup from using frequency correlation

Figure 4. Graph shows a 387x increase in memory resources consumed

Future Work

The S2 stage currently uses a general dictionary composed of templates collected using a variety of photographs from the natural world. Using this dictionary, HMAX can perform at a significant accuracy, but this accuracy was only for ten classes of images. As can be seen in previous work [9], HMAX has been known to top out in accuracy at around twenty classes of images. This is due to the edge features extracted from HMAX not containing enough variance for more than twenty classes of images. This lack of variance then hinders any classifier from being able to yield a decent accuracy. In order to compensate for this, future work will look at using a more customized dictionary in the S2 stage. Future work should encompass using a more specialized dictionary, whereby the patches are instead extracted from the dataset being tested on.

In addition to a customized dictionary, future work will also investigate the effects of zero padding the image before doing a Fourier transform. Because of the amount of zero padding that needs to be done at different scales, the size of pre-converted patch coefficients is rather large. This extra padding was done to avoid wrap around error when converting to the Fourier domain [10]. However, the error produced from neglecting to zero pad the image may be negligible and may still yield satisfactory results. If the findings are true, then the amount of storage needed for the template coefficients will be cut down to one-fourth the original size.

The theoretical architecture developed does not account for the cost of going into and out of the Fourier domain. At the front end, the cost to convert the C1 data to the Fourier domain is negligible. However at the backend, that 50x speed up would be lost from having to convert all the data from the S2 stage back to the spatial domain. This is motivation for looking at schemes to extract viable features from the Fourier domain, making the large number of inverse Fourier transforms unnecessary.

Conclusion

In this paper, we describe various methods to increase both the accuracy and the execution time of the state-of-art HMAX feature extractor. We developed several experiments in software to test our theories of improving accuracy using spatial info, and conducting correlation in the Fourier domain. Results show that adding spatial information to the final C2 vector hinders the classification accuracy. However, HMAX can be accelerated using the Fourier domain for correlation while still maintaining the baseline accuracy.

REFERENCES

[1] P. Cerri, et al. (2011). Computer vision at the Hyundai autonomous challenge. International IEEE Conference on Intelligent Transportation Systems, pp. 777-783.

[2] R. T. Collins, et al., (2000). A system for video surveillance and monitoring: VSAM final report, Technical report CMU-RI-TR-00-12.

[3] L. Matthies, et al. (2007). Computer vision on mars. International Journal of Computer Vision, 75(1), 67–92.

[4] A. Maashri, A., M. Cotter, N. Chandramoorthy, M. DeBole, C. L.Yu, V. Narayanan, & C. Chakrabarti, (2013). Hardware Acceleration for Neuromorphic Vision Algorithms. Journal of Signal Processing Systems, 70(2), 163-175.

[5] J. Mutch, & D. G. Lowe, (2006, June). Multiclass object recognition with sparse, localized features. In Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on (Vol. 1, pp. 11-18). IEEE.

[6] R. Bracewell, "Pentagram Notation for Cross Correlation." The Fourier Transform and Its Applications. New York: McGraw-Hill, pp. 46 and 243, 1965.

[7] S. Lazebnik, S. Cordelia, P. Jean. "Beyond bags of features: Spatial pyramid matching for recognizing natural scene categories." Computer Vision and Pattern Recognition, 2006 IEEE Computer Society Conference on. Vol. 2. IEEE, 2006

[8] S. Vaseghi, Advanced digital signal processing and noise reduction. John Wiley & Sons, 2008 (Second Edition, pp. 272)

[9] M. Cotter, S. Advani, J. Sampson, K. Irick, & V. Narayanan, (2014, November). A hardware accelerated multilevel visual classifier for embedded visual-assist systems. In Proceedings of the 2014 IEEE/ACM International Conference on Computer-Aided Design (pp. 96-100). IEEE Press.

[10] R. C. Gonzalez, and Richard E. Woods. "Digital image processing (2007) (3rd edition pp. 263)