

MULTI-LEVEL MODELING OF OSCILLATOR BASED COMPUTATION

by

Zachary D. Hull

B.S. Computer Engineering Technology, University of Pittsburgh, Johnstown, PA, 2015

Submitted to the Graduate Faculty of

Swanson School of Engineering in partial fulfillment

of the requirements for the degree of

Master of Science in Computer Engineering

University of Pittsburgh

2018

UNIVERSITY OF PITTSBURGH
SWANSON SCHOOL OF ENGINEERING

This thesis was presented

by

Zachary D. Hull

It was defended on

July 23, 2018

and approved by

Bruce R. Childers, PhD, Professor, Department of Computer Science

Samuel Dickerson, PhD, Assistant Professor, Department of Electrical & Computer Engineering

Amro El-Jaroudi, PhD, Associate Professor, Department of Electrical & Computer Engineering

Thesis Advisor: Donald M. Chiarulli, PhD, Professor Emeritus, Department of Computer Science

Copyright © by Zachary D. Hull

2018

MULTI-LEVEL MODELING OF OSCILLATOR BASED COMPUTATION

Zachary D. Hull, M.S.

University of Pittsburgh, 2018

Emerging device technologies for nano-oscillators have inspired research in the use of oscillators to perform mathematical operations based on non-Boolean computations using the coupling behavior of an oscillator cluster, rather than CMOS logic gates. For example, circuits using coupled oscillators can be created to measure the Degree of Match (DOM) between two vectors. Coupled oscillators synchronize through a range of frequencies, called the locking region, depending on coupling strength. The output behavior of coupled oscillators with a DOM detector in the locking range has been shown to be the Euclidean distance squared, where larger DOM voltages correspond to more similar vectors. The convolution of two vectors can be calculated using three DOM oscillator clusters based on the algebraic expansion of Euclidean distance squared.

Because the nano-oscillator devices have not matured enough to build large systems, it is important to design models of coupled oscillator behavior. Modeling oscillators is required across a hierarchy consisting of device models, circuit and logic models, and system models in order to support the entire scope of design abstractions. Device models are created by trial and error matching of the model to measured data. The system level model is then created by a polynomial fit to the output of a SPICE simulation of coupled oscillator circuits built using the

circuit model. The result is a closed-form system level mathematical model usable in MATLAB and C++.

This thesis presents a study of three models that span the discussed hierarchy: one STO model created based on two different circuit models with different detector characteristics, a VO₂ oscillator model, and a generic parameterized model used to evaluate variations in oscillator parameters. All versions were tested at each level of the hierarchy and the results compared to control values to verify the models. The device level model was compared to the empirical data. The circuit level models were used to calculate DOM and convolution; these calculations were compared to MATLAB calculations. The system level models were tested in an image processing pipeline (IPP) and the accuracy of these models was compared to conventional floating-point calculations.

TABLE OF CONTENTS

PREFACE.....	XI
1.0 INTRODUCTION.....	1
1.1 PURPOSE OF STUDY.....	3
1.2 STATEMENT OF WORK.....	3
1.3 CONTRIBUTION.....	4
1.4 THESIS ORGANIZATION.....	5
2.0 BACKGROUND	6
3.0 APPROACH.....	10
3.1 HOW OSCILLATORS COMPUTE.....	10
3.1.1 Oscillator Based Degree of Match (DOM) Computation.....	10
3.1.2 Generic Parameterized Oscillator Model	12
3.1.3 Convolution with Coupled Oscillators	14
3.2 MODELING OSCILLATOR COMPUTATION.....	15
3.2.1 Physical Level Model for STO and VO₂	15
3.2.2 VO₂ Device Level Model.....	16
3.3 CIRCUIT LEVEL MODEL	19
3.3.1 STO Circuit Level Model	19
3.3.2 VO₂ Circuit Level Model.....	20
3.4 SYSTEM LEVEL MODEL	23

3.4.1	System Level Model Generation Methodology	23
3.4.2	STO System Level Model	25
3.4.3	VO ₂ System Level Model.....	33
3.5	C++ OSCILLATOR MODEL LIBRARY.....	37
4.0	EXPERIMENTAL DESIGN.....	38
4.1	VERIFICATION OF SYSTEM LEVEL MODEL.....	38
4.1.1	Discrete Cosine Transform	39
4.1.2	Inverse Discrete Cosine Transform.....	41
4.1.3	IPP Performance Evaluation	41
4.2	TEST PROCEDURE.....	43
5.0	RESULTS	45
5.1	STO IMAGE PROCESSING PIPELINE	45
5.2	VO ₂ IMAGE PROCESSING PIPELINE	47
5.3	IMAGE PROCESSING PIPELINE SIMULATIONS WITH GENERIC OSCILLATOR MODEL	49
5.1	PARAMETRIC ANALYSIS.....	50
6.0	CONCLUSION.....	53
	BIBLIOGRAPHY.....	55

LIST OF TABLES

Table 1. Summary Tables from IPP for STO Model 1 compared to Conventional Floating-Point calculations.	46
Table 2. Summary Tables from IPP for STO Model 2 compared to Conventional Floating-Point calculations.	47
Table 3. Summary Table of IPP using system level VO ₂ model using NeoVision Tower dataset	48
Table 4. WNMOTDA and F1 Scores using Tower dataset for three VO ₂ models: 1) Mean Coefficient values from MATLAB simulations 2) Lower Bound Coefficients from 95% confidence interval 3) upper bound coefficient from 95% confidence interval	48
Table 5. Summary Tables and pipeline scores for Generic Oscillator Model with baseline parameters and Conventional Floating-Point. Three datasets were tested to compare the two pipeline models. a) NeoVision Tower Scene 15 (750 Frames). b) DARPA Helicopter Scene	49

LIST OF FIGURES

Figure 1. Coupled Oscillator Circuit with DOM detector output [20]	11
Figure 2. DOM simulation data of three coupled oscillators [20]. A curve fit to the simulation points is shown in (a) and an inverted cross-section with L_2^2 plot is shown with the simulation data in (b).	12
Figure 3. Plot of 3-Oscillator simulations with oscillator parameters [20].....	13
Figure 4. Oscillator-Based Convolution using three coupled oscillator clusters for A and B terms A terms and B terms	15
Figure 5. VO ₂ Relaxation Oscillator Model [21].....	17
Figure 6. VO ₂ SPICE Model compared to hardware data	18
Figure 7. Voltage response of VO ₂ model compared to hardware measurements	18
Figure 8. Surface plot of DOM data from STO circuit level model	19
Figure 9. Four-Coupled Oscillator Circuit with DOM detector output	20
Figure 10. Two VO ₂ Oscillator circuit with resistive coupling. a) Shows the output voltage of both oscillators showing phase locking in the locking region and b) shows the output frequency of both oscillator devices and the locking region	22
Figure 11. VO ₂ DOM Response Surface	22
Figure 12. Model Generalization Methodology.....	25
Figure 13. Three STO Curve-Fit using Squared Terms.....	26
Figure 14. Three STO Curve-Fit using Squared and Linear Terms	26
Figure 15. Three STO Curve-Fit with discharged capacitor using squared terms.....	27
Figure 16. Three STO Curve-Fit using discharged using squared and linear terms.....	27
Figure 17. Degree of Match results for STO Model 1. This plot compares the generic oscillator model, STO model 1, and MATLAB Euclidean distance	29

Figure 18. Degree of Match results for STO Model 2. The plot compares the generic oscillator model, STO model 2, and MATLAB Euclidean distance	29
Figure 19. DOM calculation for STO Model 1 system level model with 95% Confidence Interval	30
Figure 20. DOM calculation for STO Model 2 system level model with 95% Confidence Interval	31
Figure 21. Model Convolution compared to MATLAB convolution. This plot shows the generic oscillator model, STO model 1, and MATLAB convolution	32
Figure 22. Convolution results for STO Model 2. The plot compares model convolution to MATLAB convolution for the generic model and STO model 2.....	32
Figure 23. Polynomial Curve-Fit to VO ₂ DOM data using squared and linear terms in the fit equation.....	33
Figure 24. DOM calculation comparing three methods for calculating Euclidean Distance squared: 1) Generic Oscillator model with baseline parameters 2) VO ₂ System level model 3) MATLAB	34
Figure 25. DOM calculation for VO ₂ system level model with 95% Confidence Interval	35
Figure 26. Convolution results for VO ₂ system level model compared to MATLAB convolution with 95% Confidence Interval	36
Figure 27. Block Diagram of Image Processing Pipeline.....	39
Figure 28. Sample Image Frames from the NeoVision Tower (top), DARPA Helicopter (middle), and DARPA Vivid (bottom) datasets	44
Figure 29. Sample Input and Output image frame from the IPP. The image on the right shows the input image to the IPP. The image on the left shows the bounding boxes for the identified salient regions as well as the regions that were classified by the CNN.....	44
Figure 30. Parametric Analysis of CA, LR, and N for the NeoVision Tower, DARPA Helicopter, and DARPA Vivid datasets. a) CA sweep from 0% to 10% b) LR sweep from 0 to 0.1 c) N sweep from 0% to 1%	51

PREFACE

I would like to thank my advisor, Dr. Donald M. Chiarulli, for his support and guidance. I would like to thank Dr. Bruce R. Childers, Dr. Samuel Dickerson, and Dr. Amro El-Jaroudi for their support and help in my thesis defense. I would like to extend a special acknowledgement to the memory of Dr. Steven P. Levitan who helped me start my graduate research.

A special acknowledgment goes to Gyorgy Csaba of the University of Notre Dame for his help in providing me with STO model data. Also, I would also like to acknowledge Nikhil Shukla and Matthew Jerry of the University of Notre Dame for the help in providing VO₂ data.

I would like to thank my parents, Michael and Pauline Hull, for their love, support, and motivation to pursue my goals.

I especially want to thank my mentor, Mrs. Karen Bowman, for always encouraging me to give my best effort ever time. Without her, the stupendous fabric would have yielded to the pressure of its own weight.

1.0 INTRODUCTION

The standard technology used for computation has been logic gates designed using complimentary metal-oxide semiconductors (CMOS). However, emerging technologies, specifically nano-oscillators, have inspired research in new computation paradigms utilizing the coupling behavior of nano-oscillators rather than CMOS devices. The non-linear behavior of these devices can be exploited to perform useful computation depending on coupling schemes such as nearest neighbor mesh, star, and random configurations. In this thesis, nano-oscillators are electrically coupled in a star-configuration and driven by a voltage input. When the frequencies of individual oscillators match or are near-similar, the oscillators lock in phase which drives the voltage at the common node. The common node is a measure of the amount of coupling among all oscillators in the cluster that is input to a Degree of Match (DOM) detector. In this circuit configuration, input vectors that match are at a maximum DOM voltage while input vectors that are not similar decrease in voltage. The use of coupled nano-oscillator arrays may prove to be faster and more power efficient than CMOS technologies.

Two such emerging nano-oscillator technologies are Spin-Torque Oscillators (STOs) and Vanadium Dioxide (VO_2) oscillators. The two technologies exhibit oscillatory behavior by exploiting different internal physics. STOs use magnetic layers and magnetic field line coupling while VO_2 devices behave with insulator to metal transitions and vice-versa with direct electrical coupling.

In order to use these devices, the question of how to design and build circuits using these devices has been the focus of recent research. At this time, it is difficult to build large oscillator clusters using these devices, so the demonstration of the technology in large designs depends on simulation models. By modeling oscillators in a hierarchy, the complexity of the lower level models can be hidden from the upper levels. In this thesis, we use five levels of modeling in the hierarchy modeled after CMOS VLSI design: physical, device, circuit, logic, and system. Physical models measure the physics the device by typically using electromagnetic simulators or direct measurements of the device in hardware. Device models describes the pin-to-pin behavior of individual devices. Circuit models form the basic computational unit to be used to create specialized systems. In this thesis, there is no distinction between the circuit and logic levels since all coupled oscillator circuits are tested with a DOM detector. System models are high-level models that describe the behavior of nano-oscillator devices through mathematical equations or program code.

This thesis focuses on modeling the STO, VO₂, and generic parameterized oscillator technologies through all levels of the design hierarchy. The physical model is empirical data measured from devices in hardware. The data for VO₂ devices was provided by Nikhil Shukla and Matthew Jerry from the University of Notre Dame. The device model is a SPICE model based on the measured data from the physical model. The device level for VO₂ oscillators is used in a circuit level model consisting of a coupled oscillator network with a DOM detector circuit output. The circuit level model data for STOs was simulated and provided by Gyorgy Csaba from the University of Notre Dame. The data collected from the circuit level model is used to generate a closed-form system level oscillator model that can be used in MATLAB, Verilog, and C code. The system level model is tested in an image processing pipeline to evaluate and

compare the computational performance of the nano-oscillator devices to a conventional floating-point pipeline configuration. The generic parameterized oscillator model is tested in the pipeline in a parametric analysis of three oscillator parameters to test the sensitivity of the oscillator model.

1.1 PURPOSE OF STUDY

The purpose of this thesis consists of three parts. The first goal of this thesis is to create a system level model of STOs and device, circuit, and system level models for VO₂ oscillators. The second goal is to verify the circuit and system models against MATLAB and C++ and to test system level model in image processing pipeline (IPP). The third goal is to perform a study to analyze the impact of oscillator parameters of the system level model in the IPP.

1.2 STATEMENT OF WORK

The following steps were performed in order to achieve the goals previously listed.

1. Create a device model in SPICE of a VO₂ oscillator using data collected from hardware provided by Nikhil Shukla and Matthew Jerry from University of Notre Dame.
2. Use device model to build a circuit model of a four-coupled VO₂ oscillator cluster including a Degree of Match detector in virtuoso. This forms a basic computational block of coupled oscillator logic.

3. Simulate the four-coupled circuit model to create a 4-Dimensional surface of DOM values for the entire range of voltage inputs.
4. Create a system level model by curve-fitting the circuit model data to create closed-form polynomial model for use in high level modeling tools such as MATLAB, Verilog, and C. Encode this model in a software library including functions for DOM and convolution.
5. Repeat 4 using STO circuit level model data provided by Gyorgy Csaba from the University of Notre Dame.
6. Verify system level model by comparing the DOM and convolution library functions to standard functions in MATLAB.
7. Test image processing pipeline using system level STO model and system level VO₂ model.
8. Use a parametrized model that captures three key model parameters including coupling asymmetry, locking range, and output noise.
9. Perform a parametric analysis of the three oscillator model parameters to measure the impact of each parameter the performance of the image processing pipeline.

1.3 CONTRIBUTION

- Creation of a hierarchical set of oscillator models for use in designing large-scale nano-oscillator based computational systems. These models will capture the behavior of oscillator clusters spanning the physical level model to the system level model.
- Verification of oscillator models in an IPP by showing that the system level models performed similarly to a conventional floating-point pipeline configuration.

- Analyses of oscillator model parameters on the IPP showing the performance impact of the variation of the three parameters.

1.4 THESIS ORGANIZATION

This thesis is organized into six chapters. Chapter 2 discusses background information including the motivation for oscillator research is discussed and the various physical domains of nano-oscillator devices. Research using coupled oscillators in associative memories for pattern matching is discussed as well as research in hierarchical modeling. Also, research in modeling techniques similar to those used in this thesis is discussed. Chapter 3 discusses the approach and methods used to create each oscillator model in the hierarchy. Chapter 4, experimental design, explains the verification methods used for each level of the hierarchy. Chapter 5 presents the results for the oscillator models in the IPP. Chapter 6 discusses the conclusions of this thesis.

2.0 BACKGROUND

Research in coupled oscillators was first inspired by Christian Huygens in 1673. While studying pendulum clocks, Huygens discovered that two pendulum clocks hung from a common object will synchronize either in phase or exactly out of phase [1]. Nano-oscillators that are electrically coupled also exhibit this phenomenon. In recent years, there has been research in a variety of fields including magnetics [2], neuro computing [3], and quantum computing [4]. In [2], Shi *et al.* used spin torque nano-oscillators for information carriers by controlling precession in skyrmions. In [3], Datta *et al.* used coupled relaxation oscillators in brain-inspired, neuromorphic models. In [4], Alexander *et al.* used a single optical parametric oscillator to create a computationally universal continuous-variable cluster state that enables universal quantum computing.

A cluster of coupled oscillators can be configured to lock in either phase or frequency. In [5], Gupta and Buckwalter used phase-locked coupled oscillators to achieve automatic self-steering for beam-forming arrays. A coupled oscillator array and coupled-phase-locked loop were used to implement a self-steering receiver. The magnitude that the coupled oscillators differ in phase was used to determine the beam direction as well as the necessary adjustments. As an alternative, Fang *et al.* [6] performed image segmentation using a two-dimensional array of coupled oscillators. In this network, each oscillator corresponded to an image pixel. Oscillators

corresponding to pixels of similar intensity tended to synchronize to the same frequency resulting in groups of oscillators locked to a different frequency.

Coupled oscillators have been shown to have the potential to be used in numerous computation applications. There has been research that focuses on using coupled oscillators as associative memory for pattern matching through synchronization [7] [8]. In [7], Nikonov *et al.* built a coupled oscillator associative memory array with the memorized patterns and test patterns encoded as the differences of frequencies of the oscillators. Test patterns similar to memorized patterns cause the oscillators to synchronize in phase indicating a match. Vodenicarevic *et al.* [8] proposed two counter-based protocols to detect and evaluate the output of a coupled oscillator based associative memory that showed comparable results to a variance measure and had strong resilience to noise. Coupled oscillators have also been shown to be able to compute convolution. Nikonov [9] demonstrated approximate convolution using coupled oscillator arrays based on synchronization and the resulting output voltage on the summation node.

Simulations of coupled oscillators with a DOM detector output showed behavior that closely resembled Euclidean distance squared. Following the research of Nikonov, Chiarulli *et al.* [10] presented a method of using coupled oscillators to calculate exact convolution using the DOM. The DOM circuit was created with a coupled oscillator network in a star configuration where the input to the circuit was two vectors of analog voltages representing image pixels. Each oscillator was controlled by the pairwise difference of these input voltages. The voltage on the coupled node was integrated to the output where the output was the DOM of the input vectors. In his thesis, Jennings [11] showed the use of DOM oscillator clusters in numerous signal processing algorithms, including convolution and discrete cosine transform. He then showed the impact of circuit parameters on these algorithms.

In order to develop large-scale systems, modeling must be done at multiple levels, starting at physical models up to system level models. One of the most recognizable model hierarchies is CMOS technology. At the lowest level, the physical characteristics of semiconductors can be used to design transistors. At the next level, pin-level circuit models of transistors can be configured into logic gates. At the top level, the behavior of collections of logic gates are used to generate system models used to design complex systems. Many modeling methods in research follow a similar hierarchical structure to CMOS technology. Ventrice *et al.* [12] developed a compact model of phase change memory (PCM) devices based on measurements collected from the PCM. This model accurately described key physical features of the PCM including bias and temperature dependence for use by other designers to research applications of PCM. Similar research by Cobley and Wright [13] in Phase-Change RAM devices have also developed parameterized models at the device level.

As emerging devices develop, a model hierarchy can be used to design systems before large scale hardware is available. At the lowest level, empirical data measured from a single device can be used as the basis of device models. In some cases, empirical data can extend higher up in the hierarchy than just the device level. For example, Lovin *et al.* [14] claimed that prior work with 6T SRAM cells focused too much on detailed circuit simulation. In order to be used by computer architects, empirical data of memory needed to be integrated to chip level simulators and also must be computationally efficient. To do this, the empirical models were created using information from the circuit simulation and algorithms of regression modeling [14]. Also, Puglisi *et al.* [15] made similar claims regarding RRAM where an empirical model was created from measurements in hardware and is validated from circuit simulations. At several levels of our hierarchy, empirical data from device measurements is used.

This thesis builds upon the work done by Jennings [11] by creating a hierarchy of oscillator models tied to oscillators in hardware, similar to [12 – 15]. A system model of STO [16] devices was created based on SPICE simulation data of a coupled oscillator circuit and a device model for VO₂ [17] devices was created using empirical data. Circuit models of these devices are created as SPICE device models designed to match the empirical data. Using SPICE model, circuit models for DOM circuits were implemented and measured data from these circuits were used to create a system level closed-form model. The system level models were used in an image processing pipeline to perform object detection and object classification using a convolutional neural network, which has been shown have low classification error [18] [19].

3.0 APPROACH

This chapter discusses the methods used to develop the models in the hierarchy. Although there are numerous configurations for oscillators to be coupled, this thesis focuses on star-coupled oscillator clusters with resistive coupling. Also, this thesis focuses on frequency-locking of oscillators rather than phase locking. This architecture is required for our DOM detector to function properly because this is the fundamental computation unit on which our systems are built. Then, each level of the hierarchy is discussed along with the verification of the model at that level. The final section is a listing of the elements of the C++ library of oscillator functions that make up the system model.

3.1 HOW OSCILLATORS COMPUTE

This section discusses how coupled oscillators can be used to compute the DOM and convolution of input vectors of analog voltages.

3.1.1 Oscillator Based Degree of Match (DOM) Computation

Figure 1 shows an array of coupled oscillators configured to compute the DOM. Each oscillator was connected in a star-configuration in a resistor array where each oscillator was controlled by

the pairwise analog difference in voltages between input vectors. The voltage at the coupling node was integrated at the output of the circuit which measures the degree to which the oscillators synchronized, or the DOM between the input vectors.

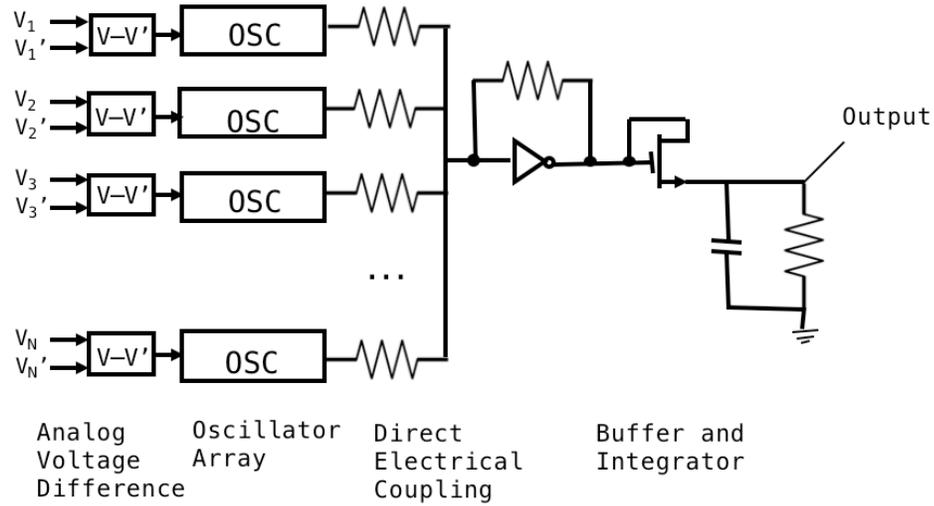


Figure 1. Coupled Oscillator Circuit with DOM detector output [20]

Research by Chiarulli *et al.* [10] has shown that the behavior of the DOM circuit is similar to Euclidean distance squared. In Figure 2a, the surface plot shows a curve-fit and data points generated from a three-coupled oscillator DOM circuit where the control voltage on two oscillators were swept over the operating region and the control voltage on the third was held constant. Figure 2b shows an inverted cross-section of Figure 2a where the solid line shows a curve-fit to the data points. This plot shows that the DOM behaves as a distance metric corresponding to Euclidean distance squared (L_2^2). Equation 1 shows the DOM equation of two vectors based on coupled oscillator circuits.

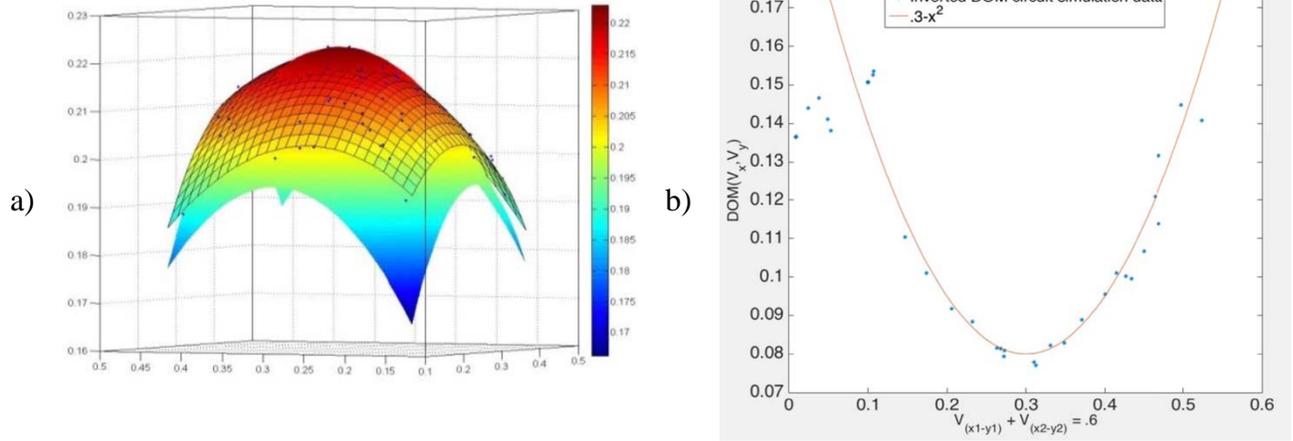


Figure 2. DOM simulation data of three coupled oscillators [20]. A curve fit to the simulation points is shown in (a) and an inverted cross-section with L_2^2 plot is shown with the simulation data in (b).

$$DOM(A,B) = L_2^2(A,B) = \sum_{i=1}^N (a_i - b_i)^2$$

Equation 1. Degree of Match

3.1.2 Generic Parameterized Oscillator Model

Equation 1 describes an ideal oscillator model. However, actual oscillator implementations exhibit variability from the ideal model. These variations can be described by three oscillator parameters: Coupling Asymmetry (CA), Locking Range (LR), and Noise (N). Figure 3 shows plots that illustrate the impact of these parameters.

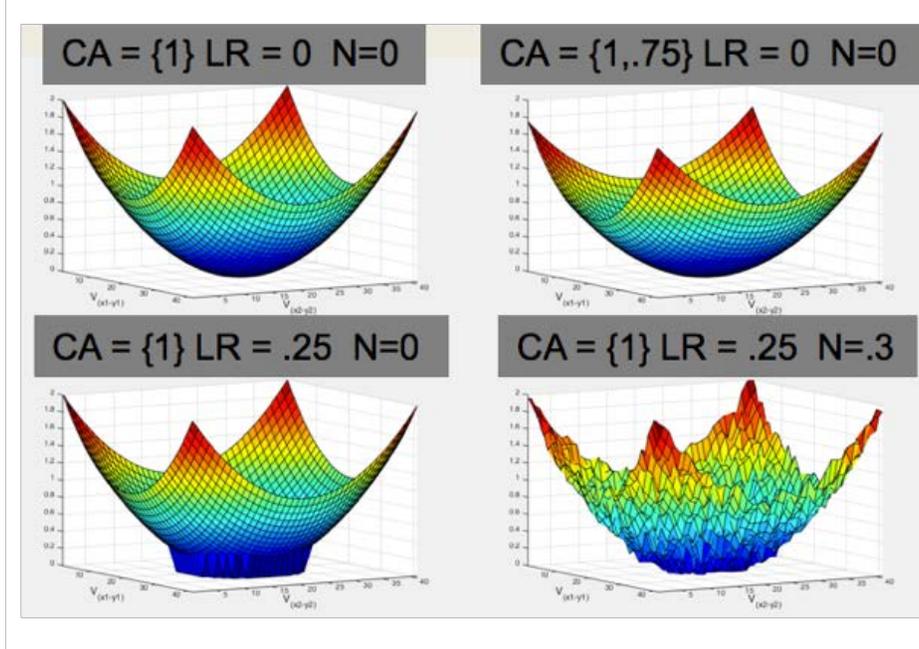


Figure 3. Plot of 3-Oscillator simulations with oscillator parameters [20]

CA describes the difference in coupling strength between each individual oscillator. LR represents a range of frequencies over which an oscillator cluster will synchronize. Depending on coupling strength, oscillator frequencies that are near similar can lock to the same frequency over a small range of input voltages making input vectors of small differences indistinguishable from other vectors of similarly small differences. Noise is a common parameter of any circuit. Output noise is the focus of this thesis where the noise is modeled by a Gaussian distribution of white noise. Equation 2 shows the DOM equation taking into account the three model parameters.

$$DOM(A,B) = \left[\begin{cases} \sum_{i=1}^N CA_i \times (a_i - b_i)^2, & L_2^2 > LR \\ 0, & L_2^2 \leq LR \end{cases} \right] + N$$

Equation 2. DOM equation with oscillator parameters

3.1.3 Convolution with Coupled Oscillators

The parabolic behavior of the DOM curve can be exploited to perform convolution, a common computational primitive in signal processing. Convolution is the summation of the dot product between two vectors, as defined in Equation 3, where a_i and b_i represent the i^{th} elements of two vectors A and B, each of length N.

$$y = \sum_{i=1}^N (a_i \cdot b_i)$$

Equation 3. Convolution of Two Vectors

Chiarulli *et al.* [10] showed that oscillator-based DOM can be used to calculate exact convolution. The convolution of two vectors can be calculated by the algebraic expansion of Euclidean Distance squared as shown in Equation 4. The oscillator convolution is performed using three clusters of oscillator arrays in parallel; one cluster for vectors A and B, one cluster for A and 0, and one cluster for 0 and B. The remaining -2 is divided out by hardware leaving the AB term. Figure 4 shows the block diagram of oscillator convolution.

$$\begin{aligned} L_2^2(A, B) &= (A-B)^2 = A^2 - 2AB + B^2 \\ -2AB &= (A-B)^2 - A^2 - B^2 \\ AB &= \frac{(A-B)^2 - A^2 - B^2}{-2} \\ conv(A, B) &= \frac{L_2^2(A, B) - L_2^2(A, 0) - L_2^2(0, B)}{-2} \end{aligned}$$

Equation 4. Convolution based on algebraic expansion of Euclidean distance squared

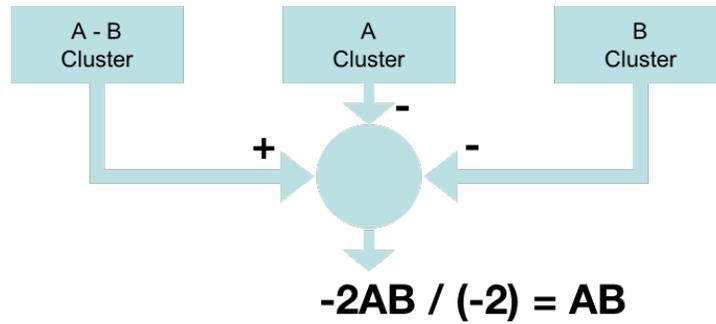


Figure 4. Oscillator-Based Convolution using three coupled oscillator clusters for A and B terms A terms and B terms

Based on DOM behavior, we have an important computational primitive and an important signal processing operation created from that computational primitive.

3.2 MODELING OSCILLATOR COMPUTATION

The physical level model aims to capture the pin-out behavior of individual nano-oscillator devices while the circuit level model aims to capture the behavior of coupled oscillator arrays. This section discusses the two nano-oscillator technologies, STO and VO₂, at the physical, device, and circuit levels.

3.2.1 Physical Level Model for STO and VO₂

Physical models are often build from electromagnetic simulations of the device physics of a nano-device. In our case, the physical level models for STO and VO₂ devices were based on measured data from devices in hardware. This level of the hierarchy was conducted by collaborators at the University of Notre Dame and the Pennsylvania State University. Gyorgy

Csaba used measurements of an STO device to design a device model. Nikhil Shukla and Matthew Jerry collected measurements of a VO₂ nano-oscillator device and provided the data to us.

3.2.2 VO₂ Device Level Model

The VO₂ device model used was based on the research of Maffezzoni *et al.* [21] and matched to measure device data obtained from the University of Notre Dame. Figure 5 shows the implementation of this model. The VO₂ oscillator is an Insulator-to-Metal (IMT) and Metal-to-Insulator (MIT) transition device. At low voltages, the VO₂ device is in the high-resistance (insulator) state. When the voltage approaches some critical value, the device transitions to the low-resistance (metal) state. The transitions were modeled using a driving-point equivalent circuit where the state is controlled by a voltage comparator. The output of the comparator is low (0V) or high (1V) which controls a voltage-controlled resistor switch which allows voltage to flow to the output.

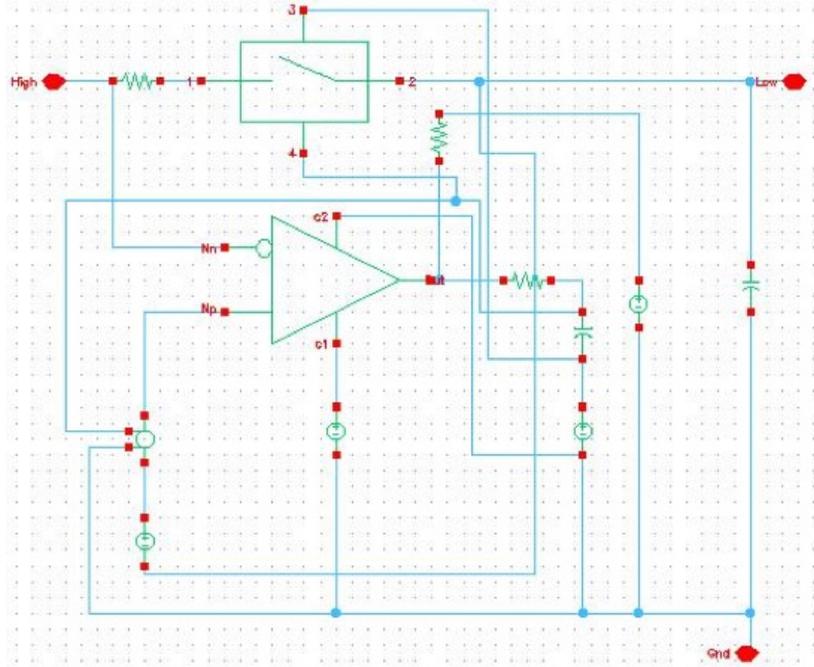


Figure 5. VO₂ Relaxation Oscillator Model [21]

Data collected from a single VO₂ device configured as a relaxation oscillator in hardware was provided by the University of Notre Dame. Using this data and the VO₂ device model configured as a relaxation oscillator, the model parameters were adjusted and internal capacitance was added until the model suitably matched the data provided through trial-and-error. To verify the model, the VO₂ device model and a transistor in series were used to create a test circuit. Figure 6 below shows the SPICE model frequency as a function of the drain-to-source current compared to the data. Figure 7 shows the voltage response of the SPICE model compared to the data. These figures show that our device model closely corresponded to the data.

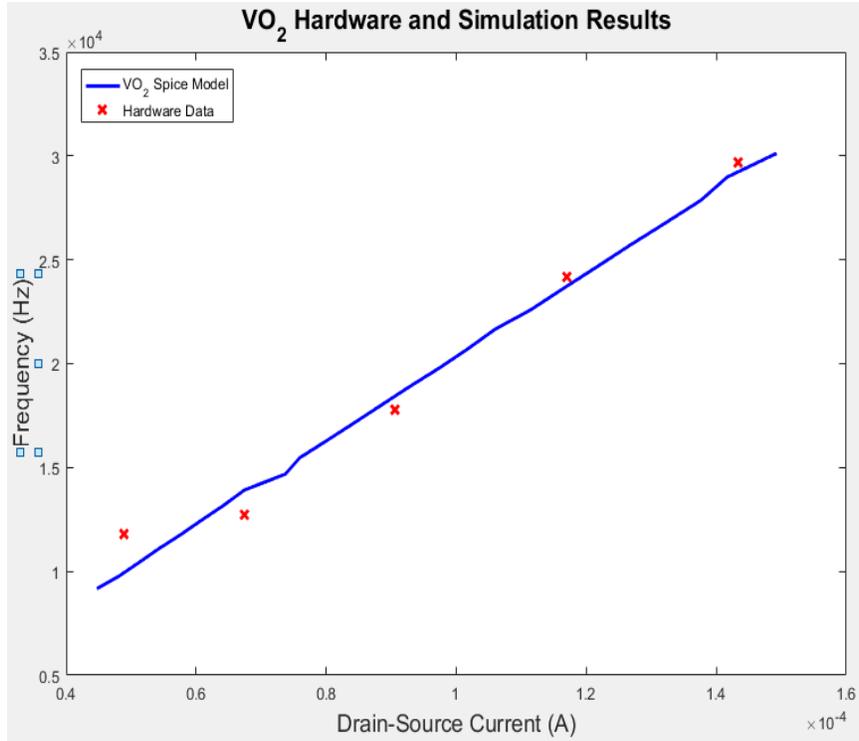


Figure 6. VO₂ SPICE Model compared to hardware data

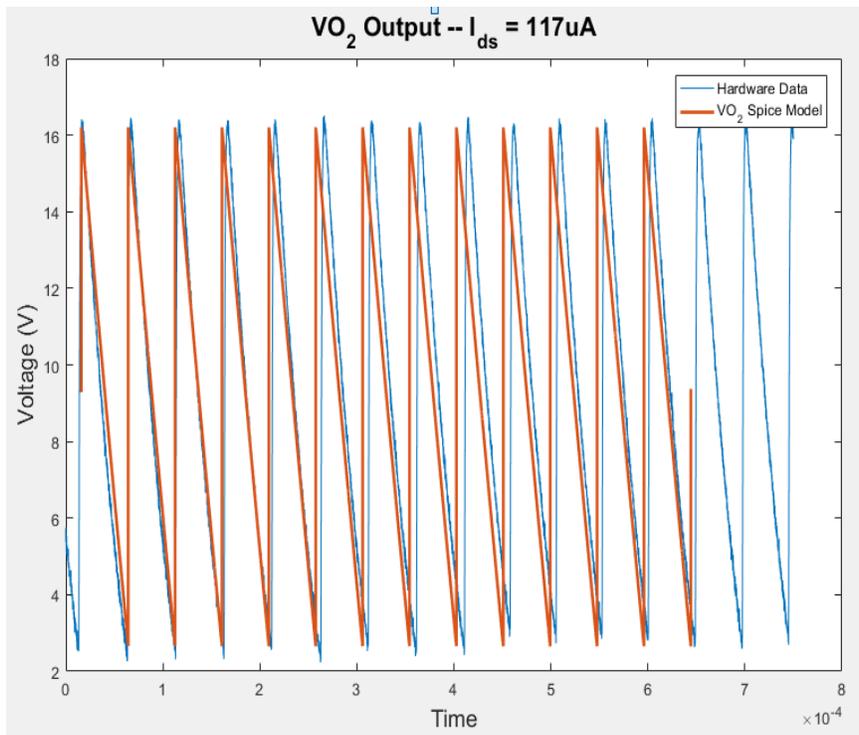


Figure 7. Voltage response of VO₂ model compared to hardware measurements

3.3 CIRCUIT LEVEL MODEL

The circuit level model uses the device level model to create a coupled oscillator cluster with a DOM detector circuit, which forms the basic computation unit. Therefore, there is no distinction between the circuit level and logic level of the model hierarchy because circuit models were always tested with DOM detectors. This section describes the DOM circuit models using STO and VO₂ oscillators.

3.3.1 STO Circuit Level Model

The STO circuit level model was developed and simulated by Gyorgy Csaba at the University of Notre Dame. Four STOs were coupled in a star configuration through a magnetic field line and the coupling node used as input to an integrator to produce a DOM response similar to the circuit shown in Figure 1. The circuits were simulated by holding the driving current of three oscillators constant while the driving current of a fourth oscillator was a time-dependent current sweep. Figure 8 shows a three-dimensional surface of the output from the circuit simulations.

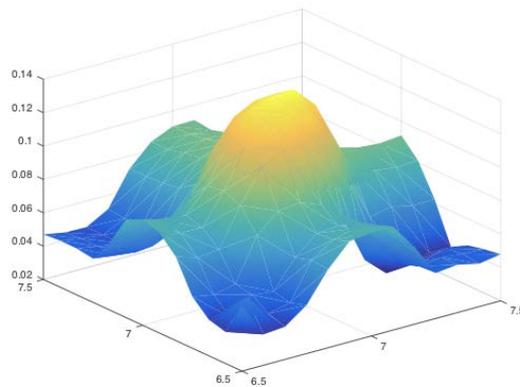


Figure 8. Surface plot of DOM data from STO circuit level model

3.3.2 VO₂ Circuit Level Model

Once the device model of the VO₂ was tuned and verified, a coupled oscillator circuit was created in Cadence Virtuoso with a Degree of Match detector circuit based on the DOM circuit in Figure 1. Figure 9 below shows the schematic of the four-coupled oscillator circuit.

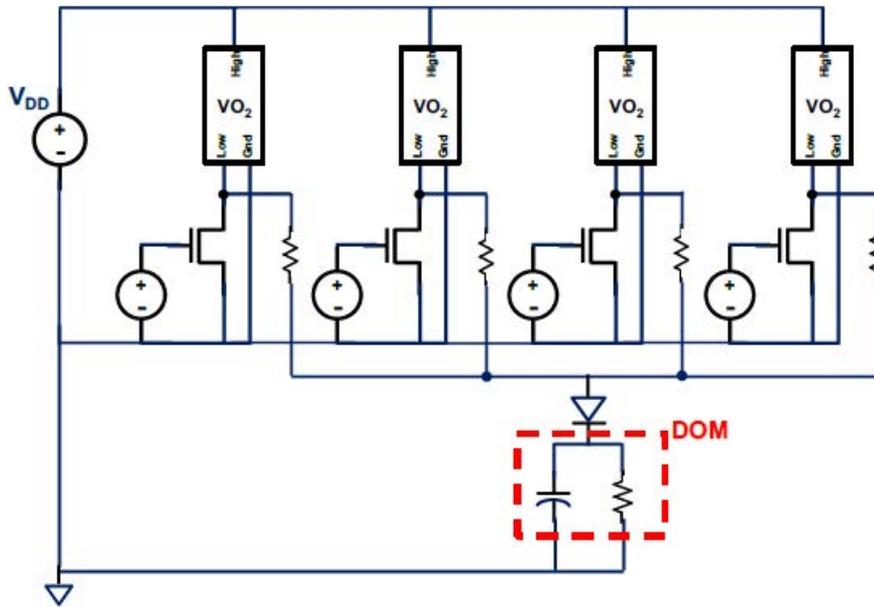


Figure 9. Four-Coupled Oscillator Circuit with DOM detector output

The DOM detector is an envelope detector that outputs the envelope of the coupled oscillator mixed signal. The frequencies of each oscillator were controlled using the gate voltage on the transistors. A critical component of this circuit was the coupling of the oscillators. An issue with using an IMT device as an oscillator was that the switching behavior was non-sinusoidal. A sinusoidal behavior was important for an accurate DOM response. Various configurations of resistive and capacitive coupling were used to generate sinusoidal behavior at the output of the circuit.

In order to perform curve-fitting, a 4-dimensional surface of VO₂ data was generated. This was done by holding the controlling voltages on three oscillators constant while the fourth oscillator was time-swept over a range of values. The constant voltages were swept using a parametric analysis for 11 points in the range over which the oscillators were locked. Once the simulations were completed, the DOM surface was created in MATLAB by smoothing the data collected by the circuit simulation

Resistive and capacitive coupling were used in a two-coupled oscillator circuit in order to find a configuration that has appropriate output behavior. The circuits were simulated holding one oscillator at a constant frequency while the second oscillator was swept over a range of frequencies. With capacitive coupling, the oscillators lock in frequency and do not lock in phase. Frequency only locking was not useful because the DOM detector used does not output parabolic behavior, which is critical to the DOM calculation. The oscillators lock in both phase and frequency with resistive coupling as shown in Figure 10 below. This gives the desired output response for DOM. The VO₂ DOM data was generated by the four-coupled oscillator circuit, shown in Figure 11, and was used to create a polynomial a curve-fit.

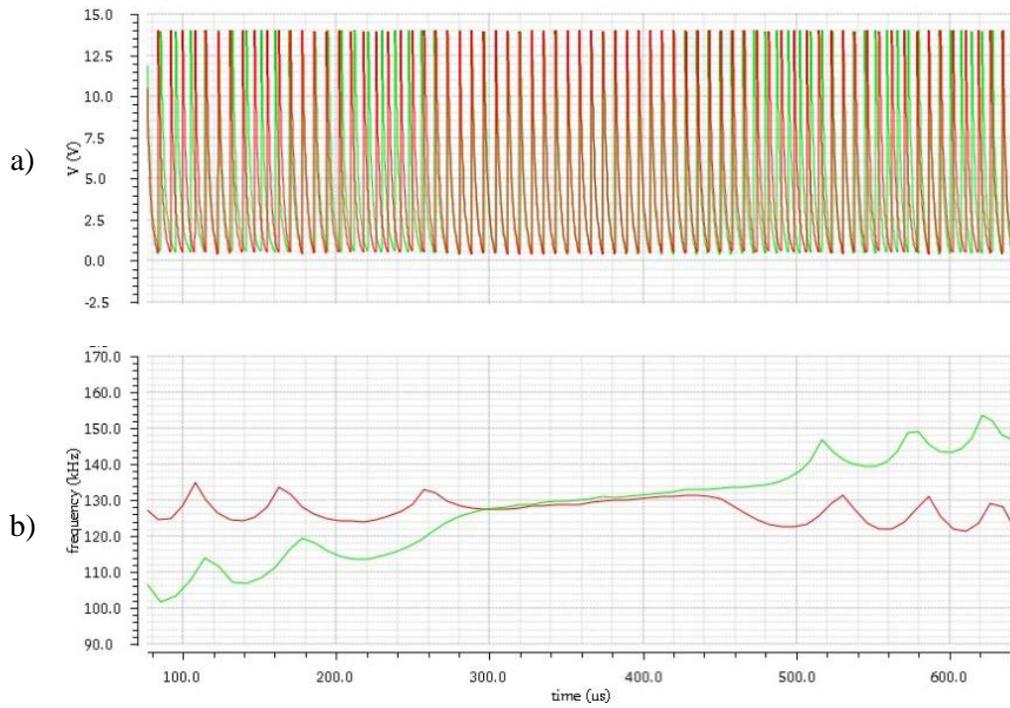


Figure 10. Two VO₂ Oscillator circuit with resistive coupling. a) Shows the output voltage of both oscillators showing phase locking in the locking region and b) shows the output frequency of both oscillator devices and the locking region

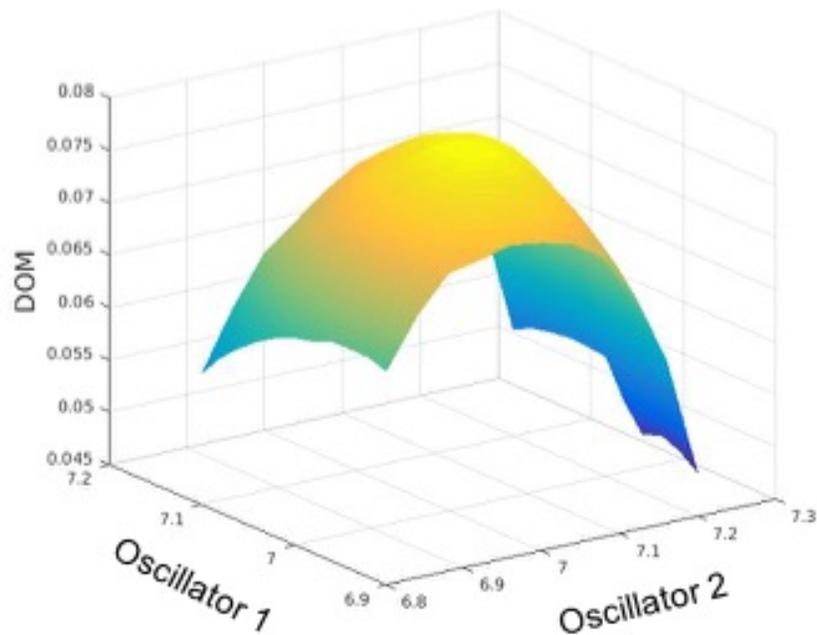


Figure 11. VO₂ DOM Response Surface

3.4 SYSTEM LEVEL MODEL

The system level model utilizes the data from the circuit level to create a mathematical representation of the behavior of four-coupled oscillator circuit. Based on the four-coupled oscillator circuit, we generalize the four-coupled oscillator arrays into larger array clusters. The system level models for STO and VO₂ oscillators are discussed in the following sections.

3.4.1 System Level Model Generation Methodology

The system level model is a polynomial curve-fit to the 4-Dimensional DOM surface for the STOs and VO₂ oscillators. The polynomial fit was performed by the polyfitn function in MATLAB set to generate a combination of squared and linear terms. Polyfitn calculates three metrics used to evaluate the goodness of the curve-fit: R², Adjusted R², and Root Mean Square Error (RMSE). R² is a measure of the goodness of fit between the curve-fit and the data as a percentage of how much variation is explained by the curve-fit model. The Adjusted R² is a modified R² calculation that takes into consideration the degrees of freedom in the model. Adjusted R² penalizes independent variables that do not improve the model which helps to prevent overfitting the model to the data. Curve-fit results with R² and Adjusted R² values closer to 1 are considered better than those with values closer to 0.

Using only squared terms, as in Equation 1, the curve-fit does not create a good enough fit to use for DOM. The linear terms can be used to create a more accurate curve-fit, because the linear terms cancel out in the process of calculating convolution, as shown in Equation 5. The curve-fit was performed over different regions in the 4-dimensional surface in order to find the best goodness of fit (Adjusted R²) value.

$$\begin{aligned}
conv(A,B) &= L_2^2(A,B) - L_2^2(A,0) - L_2^2(0,B) \\
conv(A,B) &= [(A-B)^2 + (A-B)] - [(A-0)^2 + (A-0)] - [(0-B)^2 + (0-B)] \\
conv(A,B) &= (A-B)^2 + A - B - (A-0)^2 - A - (0-B)^2 + B \\
conv(A,B) &= (A-B)^2 - (A-0)^2 - (0-B)^2 \\
conv(A,B) &= L_2^2(A,B) - L_2^2(A,0) - L_2^2(0,B)
\end{aligned}$$

Equation 5. Convolution using linear and squared curve-fit terms

The curve-fit model equation for a four-oscillator cluster was not sufficient for creating and simulating large clusters of oscillators. Figure 12 shows the steps used to generalize the four-oscillator curve-fit model to an N oscillator system level model. First, the original system level model equation was inverted by subtracting the original equation from the constant term. Next, the statistical mean and standard deviation for both the squared coefficients and linear coefficients were calculated. Two vectors of N coefficients were generated from a random normal distribution centered around the statistical mean with the calculated standard deviation for both the squared and linear terms. The final steps of the model derivation were simply factoring and normalization. For models generated using this method, the accuracy of any individual model can change because of a potentially large standard deviation. Models with a small standard deviation likely are more accurate as opposed to models with a large standard deviation.

1. Original Equation

$$\text{DOM_VO}_2 = 5.5367 + 2.9326x_1 - 1.0117x_2 - 1.0147x_3 - 1.0169x_4 - 6.0701x_1^2 - 82.2921x_2^2 - 81.445x_3^2 - 81.6023x_4^2$$
2. Invert (Max - DOM_VO₂)

$$= 5.5367 - (5.5367 + 2.9326x_1 - 1.0117x_2 - 1.0147x_3 - 1.0169x_4 - 6.0701x_1^2 - 82.2921x_2^2 - 81.445x_3^2 - 81.6023x_4^2)$$

$$= -2.9326x_1 + 1.0117x_2 + 1.0147x_3 + 1.0169x_4 + 6.0701x_1^2 + 82.2921x_2^2 + 81.445x_3^2 + 81.6023x_4^2$$
3. Calculate Mean and New Coefficient Vectors

$$\text{CL} = (-2.9326 + 1.0117 + 1.0147 + 1.0169) / 4 = 0.0277$$

$$\text{caL} = \text{normrnd}(\text{mean}(\text{CL}), \text{std2}(\text{CL}), n); \text{ \% generates array of length } n \text{ from normal distribution}$$

$$\text{CS} = (6.0701 + 82.2921 + 81.445 + 81.6023) / 4 = 62.8524$$

$$\text{caS} = \text{normrnd}(\text{mean}(\text{CS}), \text{std2}(\text{CS}), n); \text{ \% generates array of length } n \text{ from normal distribution}$$
4. Substitute average coefficients +/- CA and factor

$$\text{DOM_VO}_2 = 0.0277*(\text{caL}_1x_1 + \text{caL}_2x_2 + \text{caL}_3x_3 + \text{caL}_4x_4) + 62.8524*(\text{caS}_1x_1 + \text{caS}_2x_2 + \text{caS}_3x_3 + \text{caS}_4x_4)$$
5. Normalize Squared Terms to 1

$$\text{DOM_VO}_2\text{_model} = 4.4032e-4*(\text{caL}_1x_1 + \text{caL}_2x_2 + \text{caL}_3x_3 + \text{caL}_4x_4) + (\text{caS}_1x_1 + \text{caS}_2x_2 + \text{caS}_3x_3 + \text{caS}_4x_4)$$

Figure 12. Model Generalization Methodology

3.4.2 STO System Level Model

A polynomial fit to the STO data was performed using only the squared terms. However, there was some anomalous readings in the time-swept variable, oscillator 1, which led to a poor curve-fit. Figure 13 shows a curve-fit of STO model 1 using only oscillators 2, 3, and 4 using squared terms, which led to an improved curve-fit. To further improve the accuracy, a curve-fit was performed using squared and linear terms as shown in Figure 14. As stated previously, this does not affect the results of convolution.

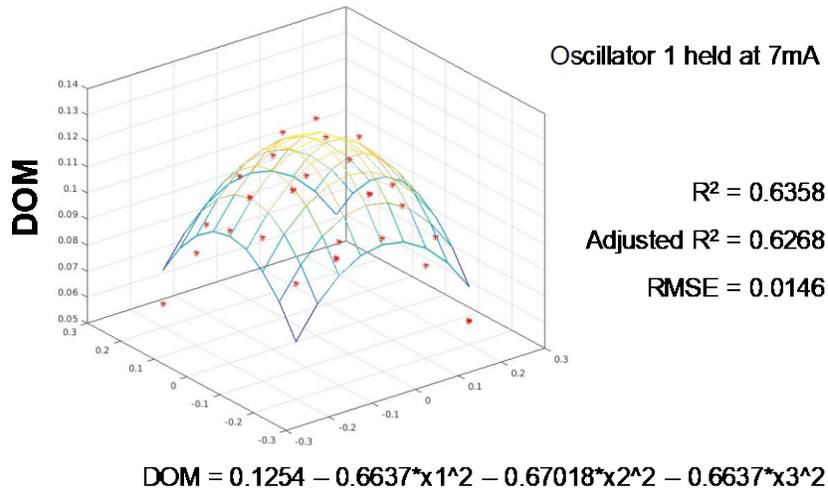


Figure 13. Three STO Curve-Fit using Squared Terms

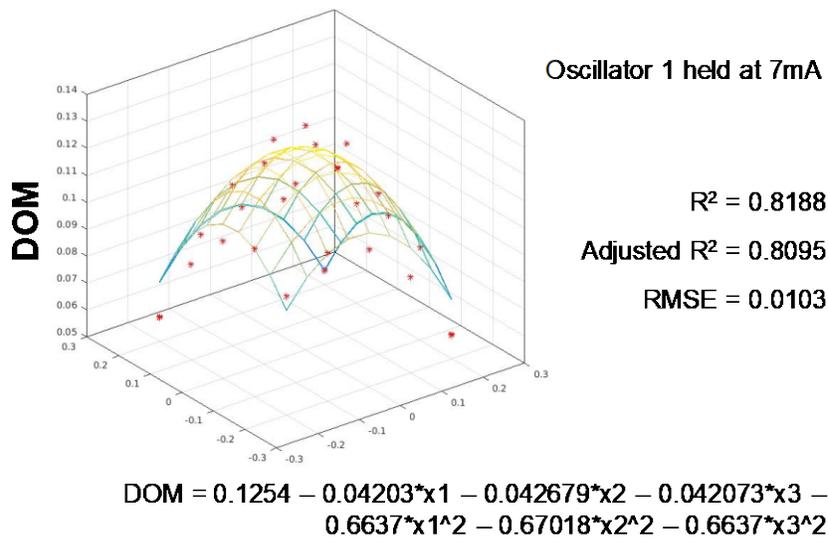


Figure 14. Three STO Curve-Fit using Squared and Linear Terms

A system model for STO model 2 was made based on data using a new detection method over a narrower range of values. This method uses new integrator circuitry where the output capacitor in the DOM circuit was fully discharged at the beginning of each matching step. Similar curve-fitting was performed as with STO model 1. Figure 15 shows the resulting curve-

fit using squared terms for oscillators 2, 3, and 4. As before, Figure 16 shows the curve-fit with squared and linear terms which resulted in better R^2 , Adjusted R^2 , and RMSE values.

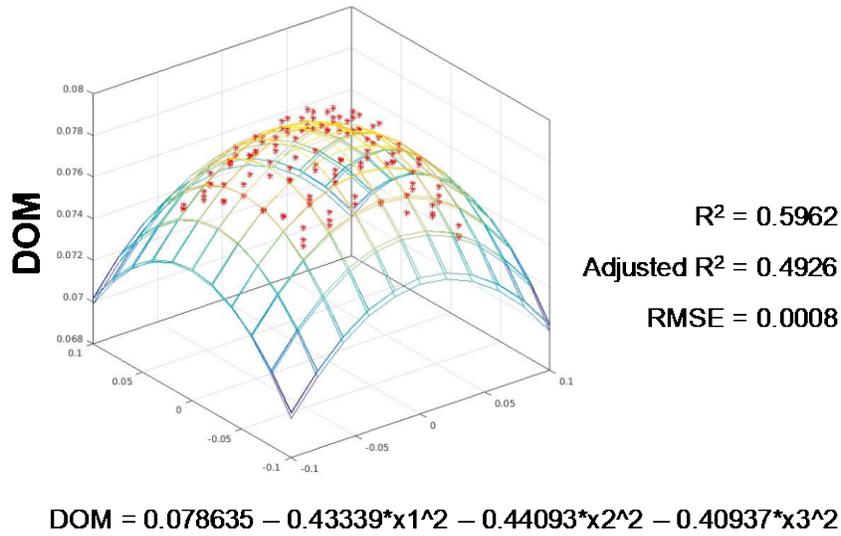


Figure 15. Three STO Curve-Fit with discharged capacitor using squared terms

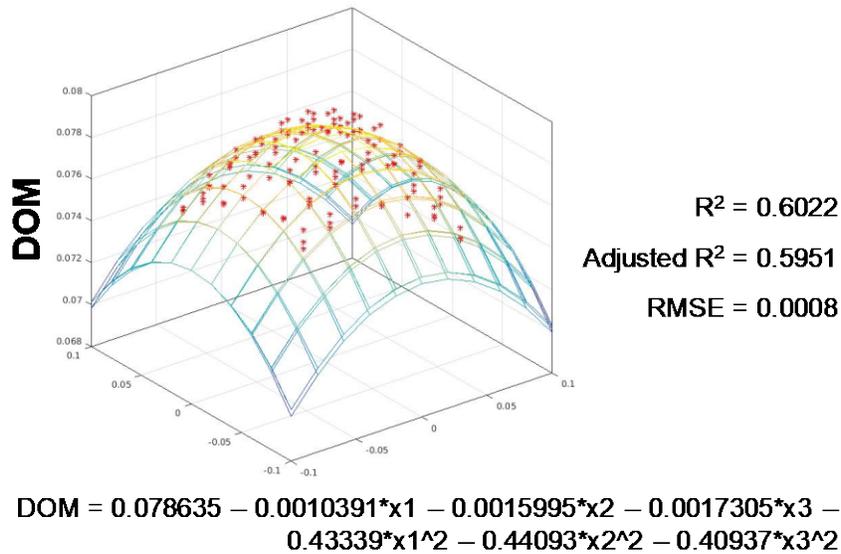


Figure 16. Three STO Curve-Fit using discharged using squared and linear terms

Both three-oscillator STO curve-fit models were generalized into a closed-form system model using the generalization method discussed above. The standard deviation of STO models

1 and 2 were 0.0037 and 0.0165, respectively. For these models, the standard deviation was small, and the system level models were likely to be accurate within a small margin of error compared to conventional floating-point calculations. Equations 6 and 7 show the closed-form system model used for STO model 1 and model 2, respectively.

$$DOM_STO_1 = 0.0635 * (caL_1x_1 + caL_2x_2 + caL_3x_3) + (caS_1x_1^2 + caS_2x_2^2 + caS_3x_3^2)$$

Equation 6. Generalized mathematical DOM model for STO model 1

$$DOM_STO_2 = 0.0034 * (caL_1x_1 + caL_2x_2 + caL_3x_3) + (caS_1x_1^2 + caS_2x_2^2 + caS_3x_3^2)$$

Equation 7. Generalized mathematical DOM model for STO model 2

The system level STO model 1 was tested in MATLAB to verify that the model could perform accurate computation. The first test was calculating the DOM between two vectors. Two vectors of 64 values were generated using a MATLAB script that, given a specific DOM value, outputs two random vectors constrained to result in that DOM. Figure 17 shows the results of performing the DOM for the set of STO model coefficients generated using Equation 6 and Figure 18 shows the DOM results for the set of STO model coefficients generated using Equation 7.

DOM Comparison of Generic Model, STO Model 1, and Matlab

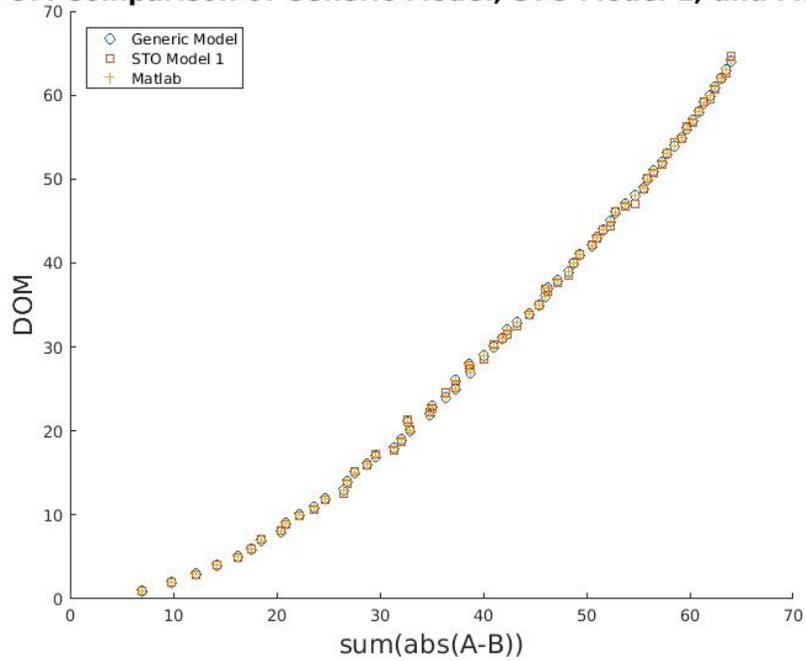


Figure 17. Degree of Match results for STO Model 1. This plot compares the generic oscillator model, STO model 1, and MATLAB Euclidean distance

DOM Comparison of Generic Model, STO Model 2, and Matlab

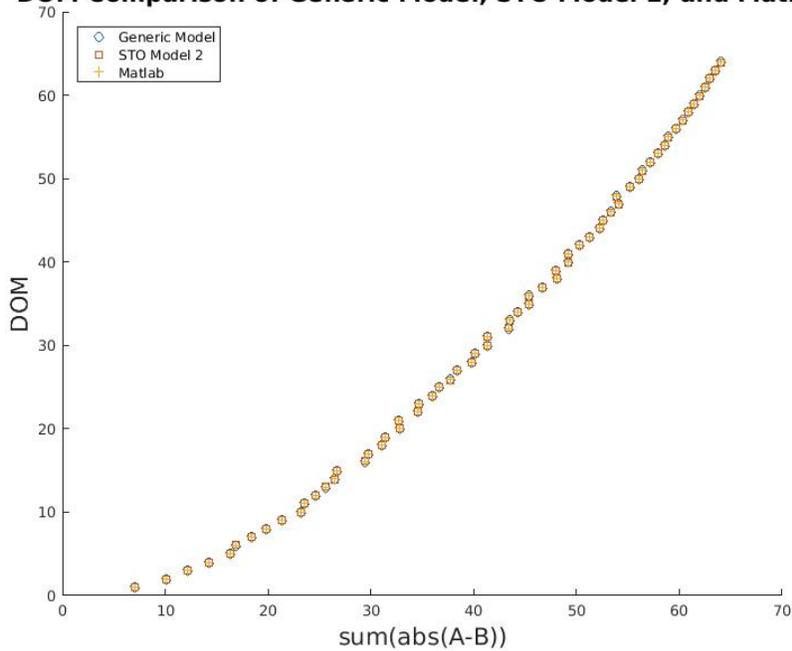


Figure 18. Degree of Match results for STO Model 2. The plot compares the generic oscillator model, STO model 2, and MATLAB Euclidean distance

Figures 17 and 18 verify the STO model for calculating the DOM between two vectors. To explore the impact of the standard deviation on DOM accuracy, a study of all possible models within a 95% confidence interval was conducted. Figures 19 and 20 below show a 95% confidence interval for the STO system models of 5000 STO models with model coefficients within two standard deviations of the mean for STO model 1 and model 2, respectively. Because of the small standard deviation in both models, the range of the 95% confidence interval was small and DOM values fall within a narrow range compared to conventional floating-point.

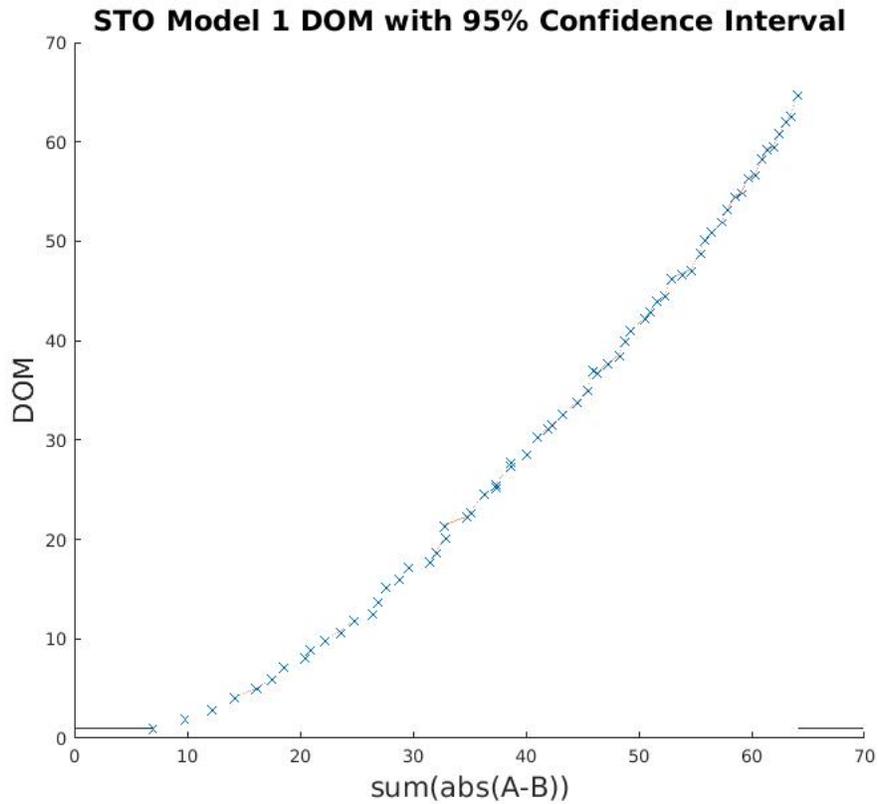


Figure 19. DOM calculation for STO Model 1 system level model with 95% Confidence Interval

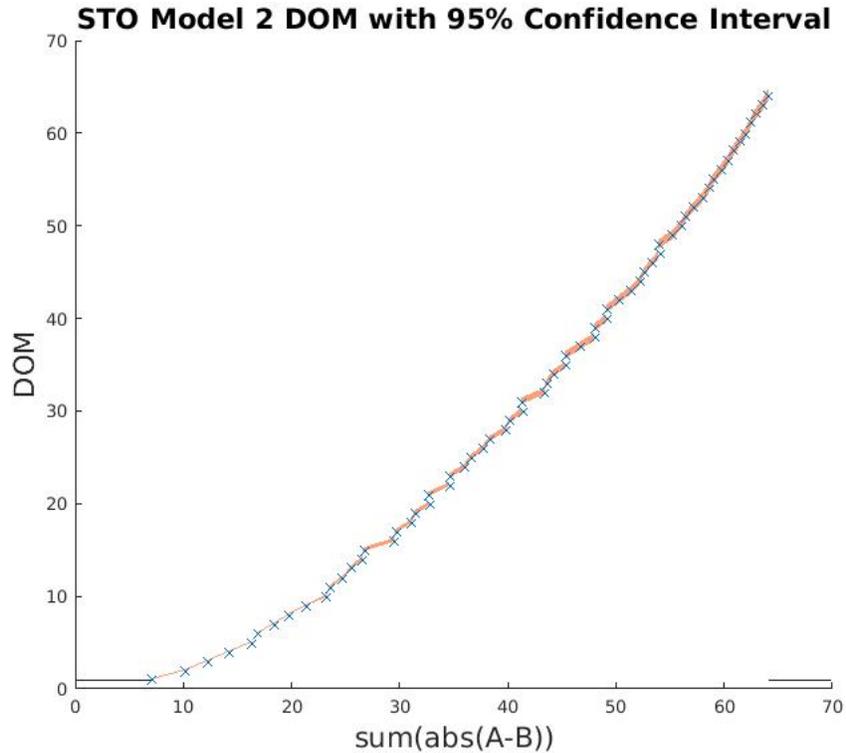


Figure 20. DOM calculation for STO Model 2 system level model with 95% Confidence Interval

The system level STO models were then tested to verify that accurate convolution can be calculated. Figures 21 and 22 below show the convolution of 64 different input vector pairs for a set of model coefficients with a 95% confidence interval and MATLAB convolution for STO models 1 and 2, respectively. As before, the 95% confidence interval has a small range indicating that for any set of coefficients generated, the accuracy was likely to be nearly equal to conventional floating-point. As shown in the figures, the plots closely correspond to MATLAB convolution for both STO model 1 and model 2.

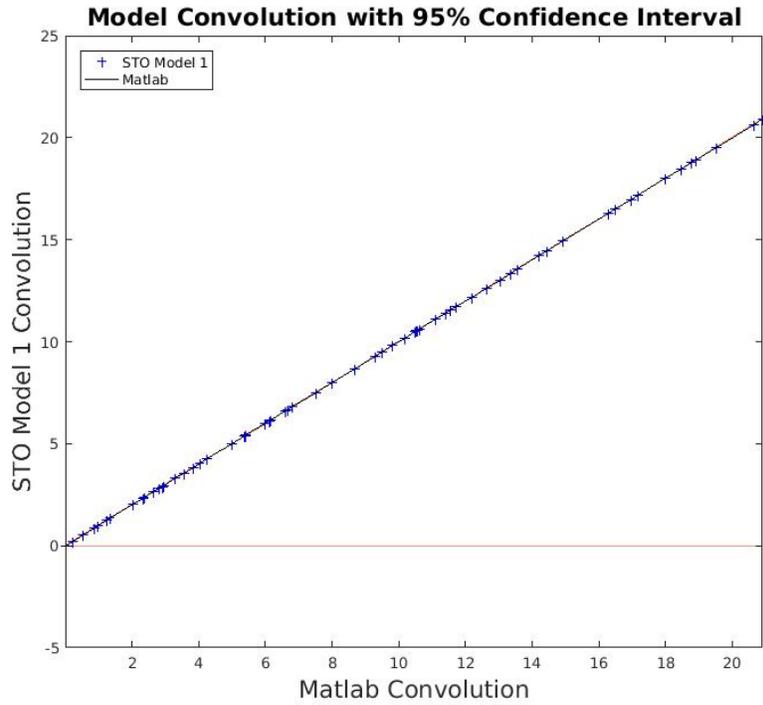


Figure 21. Model Convolution compared to MATLAB convolution. This plot shows the generic oscillator model, STO model 1, and MATLAB convolution

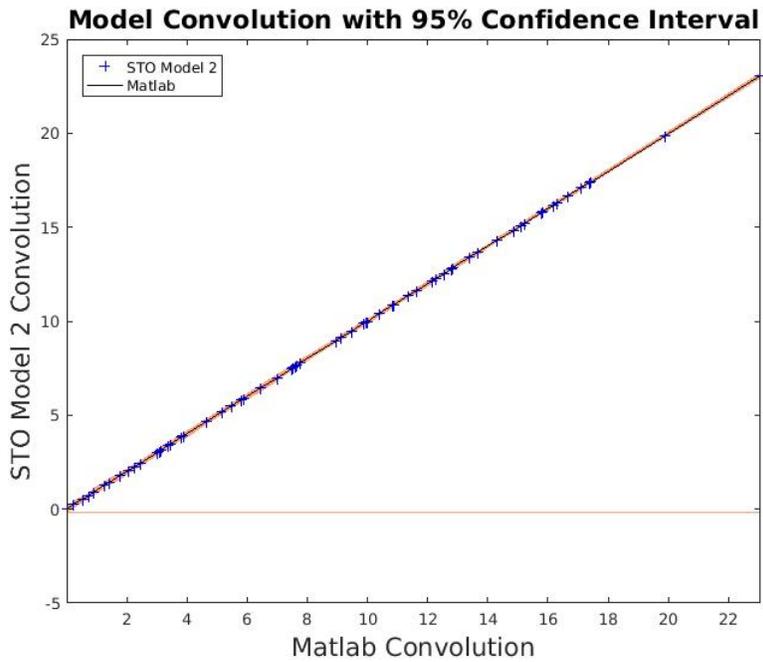


Figure 22. Convolution results for STO Model 2. The plot compares model convolution to MATLAB convolution for the generic model and STO model 2

3.4.3 VO₂ System Level Model

The system level model was created by a polynomial curve-fit for the VO₂ circuit model data. The curve-fit was generated over various lower and upper bounds on the input in order to find the most parabolic region of the four-coupled oscillator data. Figure 23 below shows the best curve-fit to the four-coupled oscillator data.

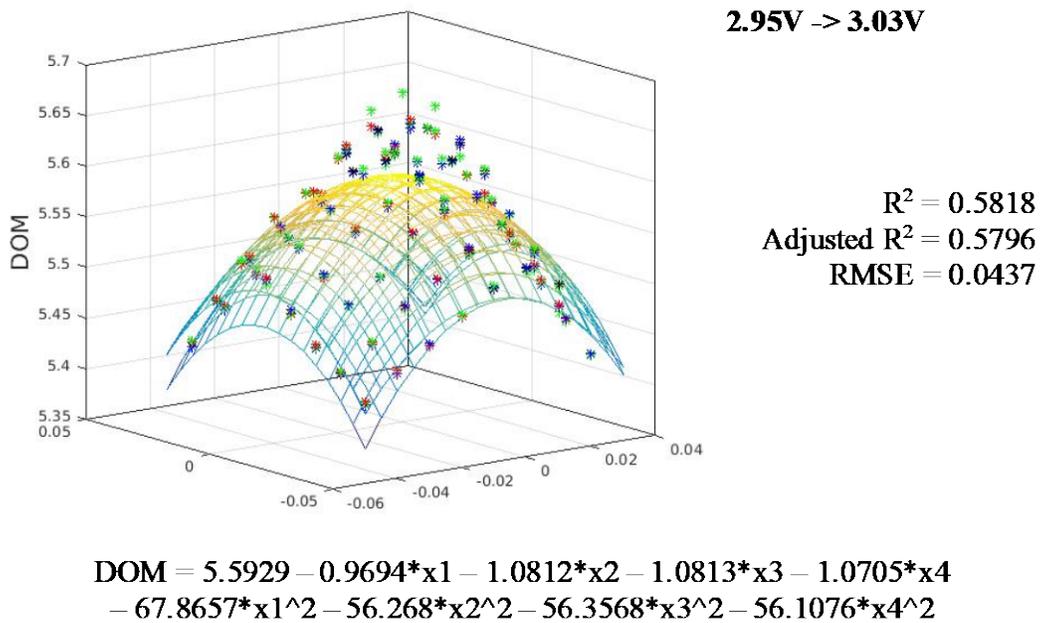


Figure 23. Polynomial Curve-Fit to VO₂ DOM data using squared and linear terms in the fit equation

A system level model for the four VO₂ oscillator curve-fit was generated using the same method use for STOs. The standard deviation of the VO₂ model was 5.8117 due to the coefficient of the time-swept oscillator. Because of the standard deviation of the VO₂ model, the accuracy of this model was not likely to fall within as small of a margin of error as the STO models. Equation 8 shows the system level model for VO₂ oscillators.

$$DOM_{VO_2} = 0.0178 * (caL_1x_1 + caL_2x_2 + caL_3x_3 + caL_4x_4) + (caS_1x_1^2 + caS_2x_2^2 + caS_3x_3^2 + caS_4x_4^2)$$

Equation 8. Generalized mathematical equation for VO₂ oscillators

The system level VO₂ model was tested in MATLAB in the same manner as the STO models for DOM calculations. The results of calculating DOM using the VO₂ model was shown in Figure 24 below where the DOM calculations for the VO₂ model are compared to the generic oscillator model and the calculation of Euclidean distance in MATLAB.

DOM Comparison of Generic Model, VO₂ Model, and Matlab

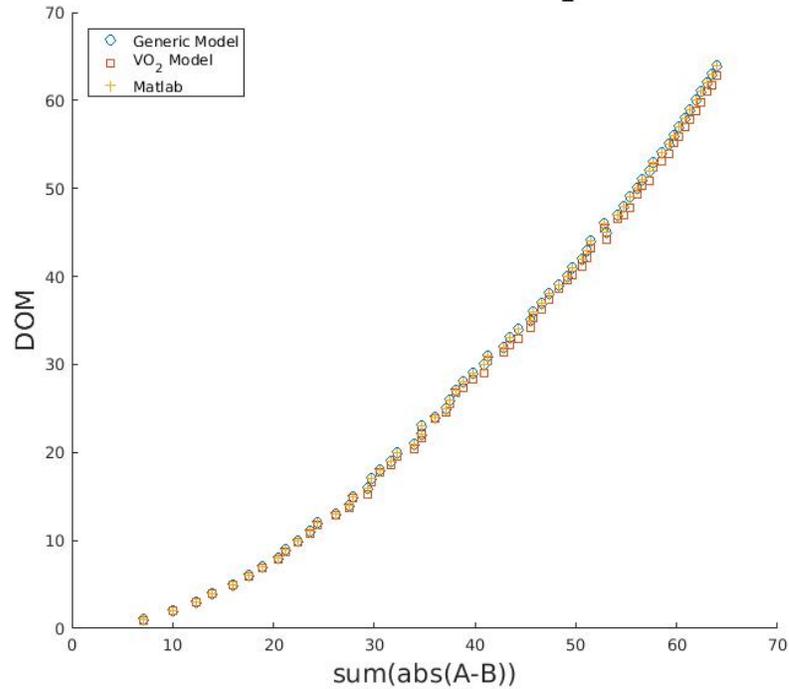


Figure 24. DOM calculation comparing three methods for calculating Euclidean Distance squared: 1) Generic Oscillator model with baseline parameters 2) VO₂ System level model 3) MATLAB

Figure 24 verifies the VO₂ model for calculating the DOM between two vectors. However, the system level VO₂ model contains a potentially significant standard deviation as explained above. This variability can cause different model coefficient vectors to result in the same DOM values. Because of the higher standard of individual models, a study of all possible

models within a 95% confidence interval was conducted to explore the impact on model accuracy. Figure 25 below illustrates shows a 95% confidence interval for the VO₂ system mode of 5000 VO₂ models with model coefficients within two standard deviations of the mean.

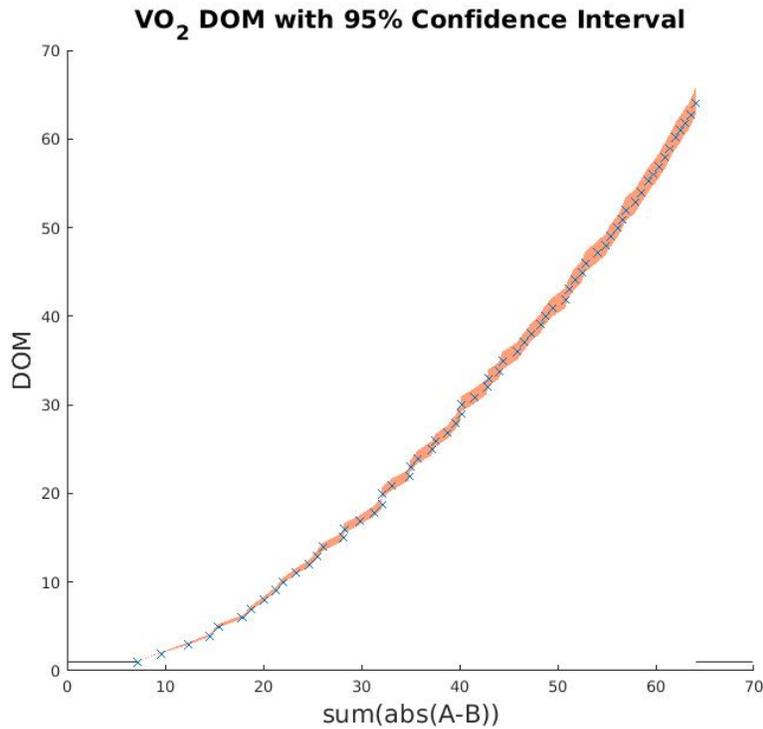


Figure 25. DOM calculation for VO₂ system level model with 95% Confidence Interval

The VO₂ system model was further tested by performing convolution calculations using the oscillator-based methodology. Figure 26 below shows the convolution results of 5000 simulations in MATLAB with at 95% confidence interval where the input vectors were held constant while the VO₂ model coefficients were regenerated each time.

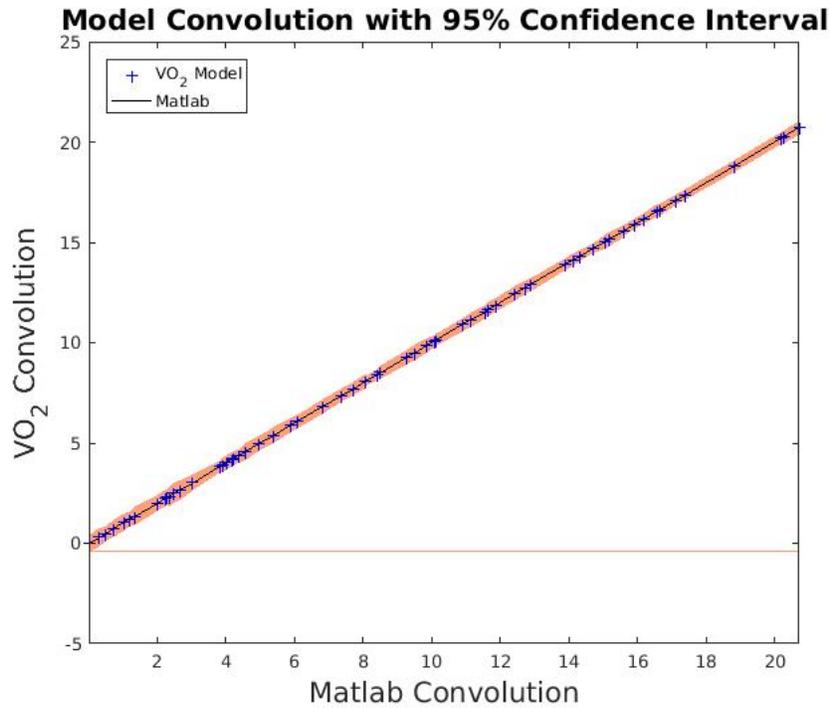


Figure 26. Convolution results for VO₂ system level model compared to MATLAB convolution with 95% Confidence Interval

The blue plotted points show the mean convolution result for all runs. These points very closely follow the expected convolution results. This verifies the ability of the system level VO₂ model to perform convolution. However, as before, the standard deviation of the squared coefficients in the system model impacts the accuracy of the convolution calculation. In Figure 26, a 95% confidence interval is plotted, representing the region that the convolution values will occur. For both the upper and lower bound, the coefficients that most closely resulting in each bound and the mean coefficient values were selected for testing in the IPP, discussed later.

3.5 C++ OSCILLATOR MODEL LIBRARY

A set of oscillator functions was created to test the oscillator models in the IPP in a C++ library and a MATLAB mex library. Below is a description of the elements contained in the oscillator library.

- An oscillator convolution function used by the IPP the uses oscillator DOM
- A DOM function used for the generic parameterized oscillator model containing the three oscillator parameters (CA, LR, and N)
- A DOM function used for hardware models (STO and VO₂) that uses both squared and linear terms
- A DCT function used by the IPP to perform 2-D DCT using oscillator convolution
- An IDCT function used by the IPP to perform 2-D IDCT using oscillator convolution

4.0 EXPERIMENTAL DESIGN

The purpose of the experiments was to verify the models. In the case of the device level model, empirical data was used to create the model which did not require verification. Since the circuit model was designed to generate a Euclidean distance squared response, the curve-fit the curve-fit process serves as a means to verify the model by using the R^2 and RMSE values of the curve-fit. This section discusses the IPP used to verify the system level model, the test procedure and test images. The IPP was tested with three sets of video sequences: NeoVision Tower [22], DARPA Helicopter [23], and DARPA Vivid [24].

4.1 VERIFICATION OF SYSTEM LEVEL MODEL

The system level model was used to test the performance of coupled oscillator DOM and convolution in an image processing pipeline. The image processing system used in this thesis was developed in collaboration with HRL Laboratories as part of the DARPA UPSIDE project [25]. The pipeline consists of two parts. First, the front-end saliency identifies salient areas or regions that may contain objects of interest. The front-end is a bottom-up spatial frequency analysis. It consists of a discrete cosine transform kernel, sign operation, and inverse discrete cosine transform kernel to identify salient regions in the image which were then extracted as image chips. Second, the back-end classifier classifies the image chips with object labels using a

convolutional neural network trained to recognize five classes of objects (car, truck, bus, cyclist, and person) with back propagation learning. Figure 27 below shows a block diagram of the IPP.

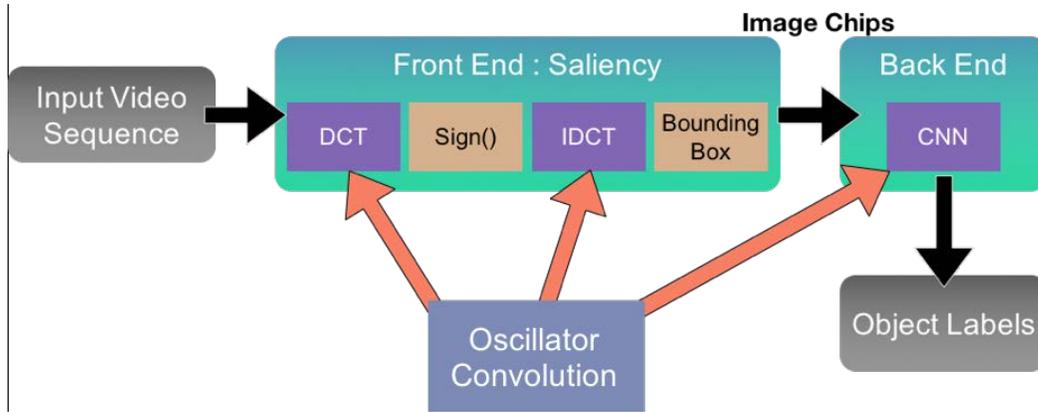


Figure 27. Block Diagram of Image Processing Pipeline

It is in the DCT, IDCT, and CNN where convolution operations occur and the oscillator model can be embedded into the pipeline. It is in these stages that the most computationally intensive operations occur and an oscillator-based accelerator can be used.

4.1.1 Discrete Cosine Transform

The DCT is an important function for image processing applications. The DCT is similar to the fast Fourier transform (FFT) and can be calculated using a convolution operation. Equation 9 below shows expression for DCT.

$$y(k) = w(k) \sum_{n=1}^N x(n) \cos\left(\frac{\pi}{2N}(2n-1)(k-1)\right), \quad k = 1, 2, \dots, N$$

where

$$w(k) = \begin{cases} \frac{1}{\sqrt{N}}, & k = 1 \\ \sqrt{\frac{2}{N}}, & 2 \leq k \leq N \end{cases}$$

Equation 9. Discrete cosine transform equation

Given Equation 9, the DCT can be calculated using DOM convolution, where $x(n)$ corresponds to input A and each cosine vector corresponds to input B.

In this pipeline, a 2-D DCT is used which consists of a series of convolutions between the input vector first along each row of the cosine coefficients and then along each column. Equation 10 shows 2-D DCT.

$$y(a, b) = w(a) * v(b) \sum_{m=0}^{M-1} \sum_{n=0}^{N-1} x(m, n) * \cos \frac{\pi * a * (2m + 1)}{2 * M} * \cos \frac{\pi * b * (2n + 1)}{2 * N}$$

where

$$\begin{aligned} 0 &\leq a \leq M - 1 \\ 0 &\leq b \leq N - 1 \end{aligned}$$

$$w(a) = \begin{cases} \frac{1}{\sqrt{M}}, & a = 0 \\ \sqrt{\frac{2}{M}}, & 1 \leq a \leq M - 1 \end{cases}$$

and

$$v(b) = \begin{cases} \frac{1}{\sqrt{N}}, & b = 0 \\ \sqrt{\frac{2}{N}}, & 1 \leq b \leq N - 1 \end{cases}$$

Equation 10. 2-D Discrete Cosine Transform

4.1.2 Inverse Discrete Cosine Transform

The IDCT returns a sequence of elements given a vector of discrete cosine transform values. The expression for IDCT is given in Equation 11 below.

$$y(k) = w(k) \sum_{n=1}^N x(n) \cos\left(\frac{\pi}{2N}(2n+1)k\right), \quad k = 1, 2, \dots, N$$

Equation 11. Inverse Discrete Cosine Transform Equation

where $w(k)$ is the same as Equation 9. Similar to DCT, IDCT can be calculated using DOM convolution. As with DCT, 2-D IDCT is used in this IPP as shown in Equation 12, where $w(a)$ and $v(b)$ are the same as in Equation 10.

$$x(m, n) = \sum_{a=0}^{M-1} \sum_{b=0}^{N-1} w(a) * v(b) * y(a, b) * \cos\frac{\pi * a * (2m + 1)}{2 * M} * \cos\frac{\pi * b * (2n + 1)}{2 * N}$$

where

$$\begin{aligned} 0 &\leq m \leq M - 1 \\ 0 &\leq n \leq N - 1 \end{aligned}$$

Equation 12. 2-D Inverse Discrete Cosine Transform

4.1.3 IPP Performance Evaluation

The image chips generated by the front end were input to the back end where the CNN attempted to classify each chip into a defined set of classes (i.e. car, truck, person, etc.). The image chips were processed by the neural network and the classification was compared to the ground truth. The image chips were then marked as a hit (true positive), miss (false negative), or false alarm (false positive). The IPP output was two scores: Weighted Normalized Multiple Object Thresholded Detection Accuracy (WMNOTDA) and F1. WNMOTDA accounts more for errors

in saliency because the WNMOTDA is calculated based on the NMOTDA which penalizes false detections, missed detections, and object fragmentation [26]. NMOTDA is calculated by Equation 14 for each class of objects.

$$NMOTDA = 1 - \frac{\text{false negatives} + \text{false positives}}{\text{class count}}$$

Equation 13. NMOTDA calculation used by IPP for each class

$$WNMOTDA = \sum_i^N \left(\frac{\text{count}_i}{\text{total objects}} * NMOTDA_i \right)$$

Equation 14. WNMOTDA calculation used by IPP

The F1 score accounts for accuracy in the CNN classification because it is based on the precision and recall. The F1 score is given by Equation 15.

$$F1 = 2 * \frac{\text{precision} * \text{recall}}{\text{precision} + \text{recall}}$$

Equation 15. F1 score for IPP test accuracy

where

$$\text{precision} = \frac{\text{true positive}}{\text{true positive} + \text{false positive}}$$

Equation 16. Precision Equation. Precision is the number of correctly classified objects divided by the number of objects classified to that class

and

$$\text{recall} = \frac{\text{true positive}}{\text{true positive} + \text{false negative}}$$

Equation 17. Recall equation. Recall is the number of objects correctly classified to a class divided by the total number of objects that should have been assigned to the class

4.2 TEST PROCEDURE

The image processing pipeline was configured with a generically trained convolutional neural network. The training set for NeoVision Tower dataset contained 45,000 image frames made up of over 571,000 image chips; the DARPA Helicopter dataset contained 22,700 image frames with over 47,700 image chips; the DARPA Vivid dataset contained 31,900 image frames with over 19,200 image chips. For the Tower dataset, the CNN was trained for only one epoch while forty and twenty epochs were needed to train Helicopter and Vivid, respectively.

Each dataset consisted of numerous video sequences for use in the IPP. The Tower dataset contains 50 video scenes each with 900 image frames. The Helicopter dataset also consisted of 50 video scenes each containing on average 435 image frames. The Vivid dataset contained 20 video scenes captured over two days. Each scene contained an average of 1,594 image frames. For testing the IPP with the Tower dataset, 750 images from Scene 15 were used to measure the performance of the oscillator models. The test set for the Helicopter Dataset was 450 images from Scene 001. The test set for the Vivid dataset contains 1557 images from Day 1 Scene 6 C009. All datasets contain five classes: Car, Truck, Bus, Bicycle, and Person. A sample image sequence for each data set is shown in Figure 28.

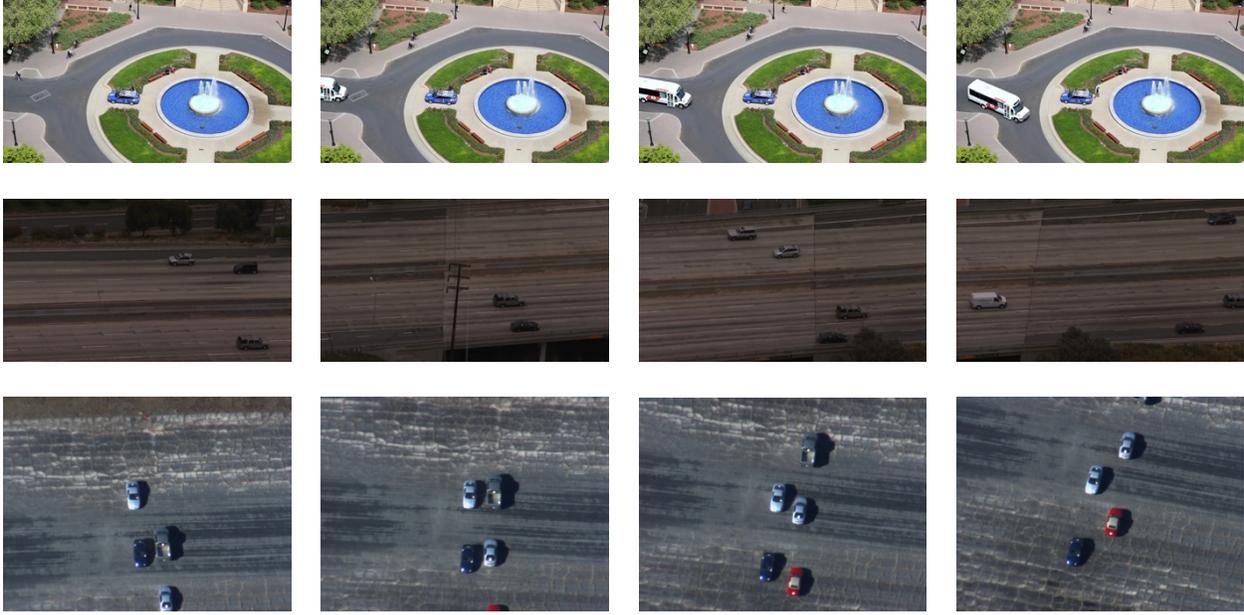


Figure 28. Sample Image Frames from the NeoVision Tower (top), DARPA Helicopter (middle), and DARPA Vivid (bottom) datasets

Figure 29 shows a sample image that has been processed by the image processing pipeline. The left image is the image input and the right is an image of the bounding box regions generated by the front-end. Each bounding box is color-coded to represent the class to which the bounding box is classified.



Figure 29. Sample Input and Output image frame from the IPP. The image on the right shows the input image to the IPP. The image on the left shows the bounding boxes for the identified salient regions as well as the regions that were classified by the CNN.

5.0 RESULTS

This chapter of the thesis shows the performance results of the IPP for STO and VO₂ oscillator models. STO models 1 and 2 are discussed first. Then, the VO₂ oscillator model performance is discussed along with an analysis of VO₂ variability. Finally, a parametric analysis of each oscillator parameter is discussed.

5.1 STO IMAGE PROCESSING PIPELINE

Once the system level models were verified in MATLAB, they were tested in the IPP. The results in Tables 1 and 2 are from IPP simulations using an older version of the pipeline for which we received trained CNN's for Tower, Helicopter and Vivid datasets. Table 1 below shows the summary tables, WNMOTDA, and F1 score for STO models 1 in the IPP for the three datasets compared to conventional floating-point calculations. These results were comparable to the conventional floating-point results.

Table 1. Summary Tables from IPP for STO Model 1 compared to Conventional Floating-Point calculations.

STO Model 1					Conventional Floating-Point				
Tower	Ground Truth	Hits	Misses	False Alarms	Tower	Ground Truth	Hits	Misses	False Alarms
a) Car	751	748	3	63	Car	751	743	8	61
Truck	0	0	0	0	Truck	0	0	0	1
Bus	390	0	390	0	Bus	390	0	390	0
Person	4687	2281	2406	940	Person	4687	2265	2422	918
Cyclist	572	396	176	37	Cyclist	572	428	144	44
WNMOTDA		F1			WNMOTDA		F1		
0.372656		0.605296			0.377031		0.606957		
b) Helicopter	Ground Truth	Hits	Misses	False Alarms	Helicopter	Ground Truth	Hits	Misses	False Alarms
Car	1982	324	1658	4	Car	1982	417	1565	8
Truck	55	0	55	0	Truck	55	0	55	0
Bus	0	0	0	0	Bus	0	0	0	0
Person	0	0	0	0	Person	0	0	0	0
Cyclist	0	0	0	0	Cyclist	0	0	0	0
WNMOTDA		F1			WNMOTDA		F1		
0.157094		0.485030			0.200785		0.552684		
c) Vivid	Ground Truth	Hits	Misses	False Alarms	Vivid	Ground Truth	Hits	Misses	False Alarms
Car	4131	189	3942	306	Car	4131	220	3911	299
Truck	2060	96	1964	102	Truck	2060	125	1935	101
Bus	0	0	0	0	Bus	0	0	0	0
Person	0	0	0	0	Person	0	0	0	0
Cyclist	0	0	0	0	Cyclist	0	0	0	0
WNMOTDA		F1			WNMOTDA		F1		
-0.019868		0.383580			-0.008884		0.439490		

Table 2 below shows the summary tables, WNMOTDA, and F1 score of STO model 2 compared to conventional floating-point for the three image datasets. This table shows that STO model 2 also was comparable to conventional floating-point calculations.

Table 2. Summary Tables from IPP for STO Model 2 compared to Conventional Floating-Point calculations.

STO Model 2					Conventional Floating-Point				
Tower	Ground Truth	Hits	Misses	False Alarms	Tower	Ground Truth	Hits	Misses	False Alarms
a) Car	751	748	3	70	Car	751	743	8	61
Truck	0	0	0	0	Truck	0	0	0	1
Bus	390	0	390	0	Bus	390	0	390	0
Person	4687	2285	2402	953	Person	4687	2265	2422	918
Cyclist	572	435	137	56	Cyclist	572	428	144	44
WNMOTDA		F1			WNMOTDA		F1		
0.373281		0.607343			0.377031		0.606957		
b) Heli-copter	Ground Truth	Hits	Misses	False Alarms	Heli-copter	Ground Truth	Hits	Misses	False Alarms
Car	1982	497	1485	12	Car	1982	417	1565	8
Truck	55	0	55	0	Truck	55	0	55	0
Bus	0	0	0	0	Bus	0	0	0	0
Person	0	0	0	0	Person	0	0	0	0
Cyclist	0	0	0	0	Cyclist	0	0	0	0
WNMOTDA		F1			WNMOTDA		F1		
0.238095		0.635956			0.200785		0.552684		
c) Vivid	Ground Truth	Hits	Misses	False Alarms	Vivid	Ground Truth	Hits	Misses	False Alarms
Car	4131	224	3907	312	Car	4131	220	3911	299
Truck	2060	111	1949	96	Truck	2060	125	1935	101
Bus	0	0	0	0	Bus	0	0	0	0
Person	0	0	0	0	Person	0	0	0	0
Cyclist	0	0	0	0	Cyclist	0	0	0	0
WNMOTDA		F1			WNMOTDA		F1		
-0.011791		0.425938			-0.008884		0.439490		

5.2 VO₂ IMAGE PROCESSING PIPELINE

Table 3 below shows the summary table for a pipeline using the VO₂ model for the mean case of coefficients, where the model coefficients were the mean coefficients generated from 5000 simulations using the system level model in Equation 8. When compared to the conventional

floating-point pipeline, the VO₂ model for the average case was comparable to conventional floating-point calculations. However, because of the standard deviation in the model, the pipeline performance can vary depending on the values of the coefficients generated.

Table 3. Summary Table of IPP using system level VO₂ model using NeoVision Tower dataset

VO ₂ Model Pipeline					Conventional Floating-Point				
Tower	Ground Truth	Hits	Misses	False Alarms	Tower	Ground Truth	Hits	Misses	False Alarms
Car	751	742	9	68	Car	751	743	8	61
Truck	0	0	0	0	Truck	0	0	0	1
Bus	390	0	390	0	Bus	390	0	390	0
Person	4687	2258	2429	911	Person	4687	2265	2422	918
Cyclist	572	429	143	47	Cyclist	572	428	144	44
WNMOTDA		F1			WNMOTDA		F1		
0.354063		0.585768			0.377031		0.606957		

As discussed earlier, the statistical variability in the VO₂ model can result in a range of convolution values. The IPP was tested using coefficients that corresponded to the upper and lower bounds of Figure 26 above. As expected, the performance of the pipeline decreases as the coefficients become farther from the mean coefficient case, as shown in Table 4 below.

Table 4. WNMOTDA and F1 Scores using Tower dataset for three VO₂ models: 1) Mean Coefficient values from MATLAB simulations 2) Lower Bound Coefficients from 95% confidence interval 3) upper bound coefficient from 95% confidence interval

Configuration	WNMOTDA	F1
Conventional Floating-Point	0.453923	0.704834
VO ₂ Corner Case Low	0.351094	0.581987
VO ₂ Mean	0.375469	0.605713
VO ₂ Corner Case High	0.352500	0.585768

5.3 IMAGE PROCESSING PIPELINE SIMULATIONS WITH GENERIC OSCILLATOR MODEL

The C++ oscillator model was embedded into the image processing pipeline code for DCT, IDCT, and convolution operations. The generic oscillator model with baseline parameters was used to test the performance of oscillator-based convolution compared to conventional floating-point calculations. Table 5 below shows the summary tables and the corresponding WNMOTDA and F1 scores for the generic oscillator model compared to conventional floating-point.

Table 5. Summary Tables and pipeline scores for Generic Oscillator Model with baseline parameters and Conventional Floating-Point. Three datasets were tested to compare the two pipeline models. a) NeoVision Tower Scene 15 (750 Frames). b) DARPA Helicopter Scene

Generic Oscillator with Baseline Parameters					Conventional Floating-Point				
Tower	Ground Truth	Hits	Misses	False Alarms	Tower	Ground Truth	Hits	Misses	False Alarms
a) Car	751	743	8	61	Car	751	743	8	61
Truck	0	0	0	1	Truck	0	0	0	1
Bus	390	0	390	0	Bus	390	0	390	0
Person	4687	2265	2422	918	Person	4687	2265	2422	918
Cyclist	572	428	144	44	Cyclist	572	428	144	44
WNMOTDA		F1			WNMOTDA		F1		
0.377031		0.606957			0.377031		0.606957		
b) Heli-copter	Ground Truth	Hits	Misses	False Alarms	Heli-copter	Ground Truth	Hits	Misses	False Alarms
Car	1982	417	1565	8	Car	1982	417	1565	8
Truck	55	0	55	0	Truck	55	0	55	0
Bus	0	0	0	0	Bus	0	0	0	0
Person	0	0	0	0	Person	0	0	0	0
Cyclist	0	0	0	0	Cyclist	0	0	0	0
WNMOTDA		F1			WNMOTDA		F1		
0.200785		0.552684			0.200785		0.552684		
c) Vivid	Ground Truth	Hits	Misses	False Alarms	Vivid	Ground Truth	Hits	Misses	False Alarms
Car	3819	530	3289	170	Car	3819	530	3289	170
Truck	1412	337	1075	241	Truck	1412	337	1075	241
Bus	0	0	0	0	Bus	0	0	0	0
Person	0	0	0	0	Person	0	0	0	0
Cyclist	0	0	0	0	Cyclist	0	0	0	0
WNMOTDA		F1			WNMOTDA		F1		
0.087173		0.676815			0.087173		0.676815		

The number of hits, misses, and false alarms for baseline parameters matches exactly to conventional floating-point calculations. The equivalence of the oscillator model and conventional floating-point algorithms was further demonstrated by the equivalent WNMOTDA and F1 scores.

5.4 PARAMETRIC ANALYSIS

A plot showing the F1 scores for a parametric analysis of each of the three oscillator parameters is shown in Figure 30 below. Figure 30a shows the impact of coupling asymmetry for the three datasets. The plot shows that as the coupling asymmetry varies from 0% to 10%, the performance of the pipeline was not severely affected. Figure 30b shows the performance as locking range varies from 0 to 0.1. This plot shows that locking region had a significant impact on pipeline performance, indicating that coupling strength between the oscillators was important to computation accuracy. Finally, Figure 30c shows the effect of output noise on pipeline performance. The plot shows that as output noise varies from 0% to 1%, accuracy of the pipeline decreases. This shows that, as with any analog computational circuit, it is important to minimize the effects of noise.

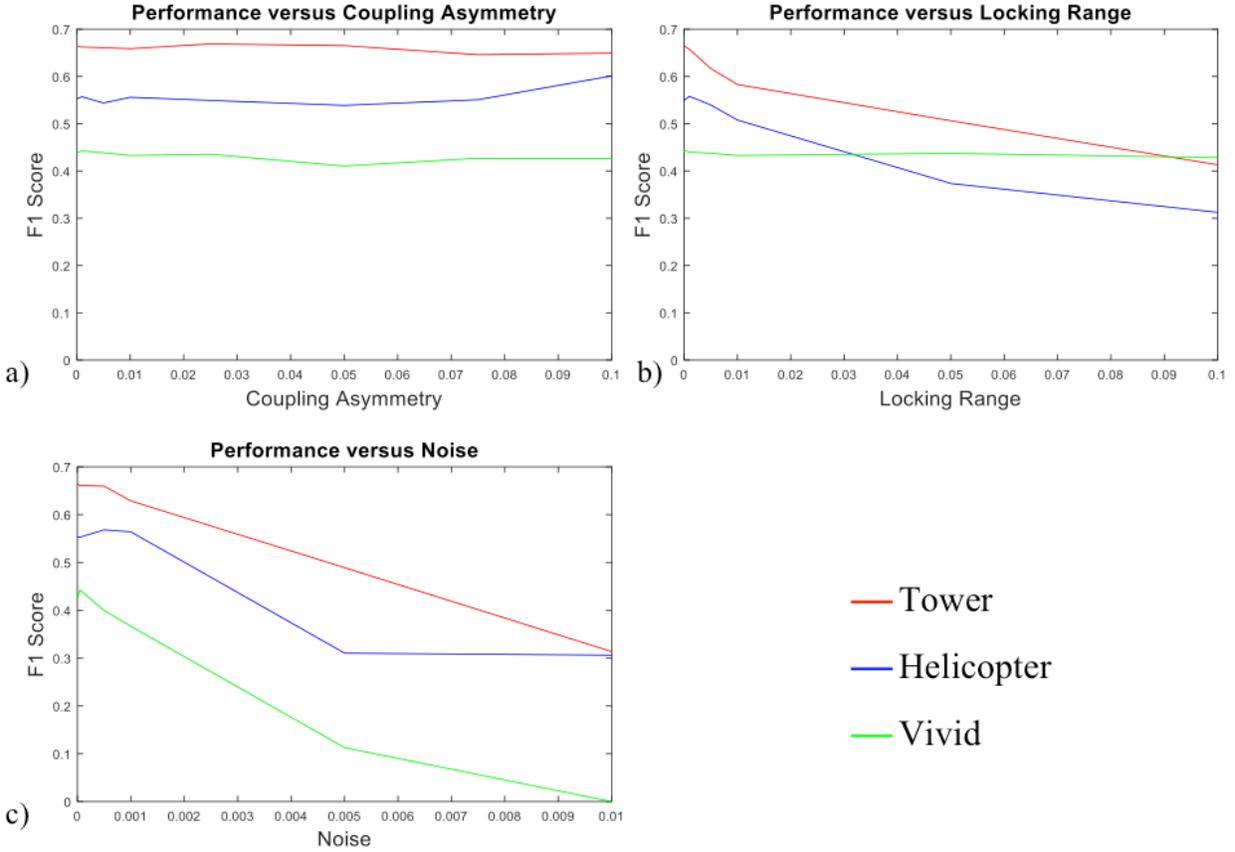


Figure 30. Parametric Analysis of CA, LR, and N for the NeoVision Tower, DARPA Helicopter, and DARPA Vivid datasets. a) CA sweep from 0% to 10% b) LR sweep from 0 to 0.1 c) N sweep from 0% to 1%

However, in the Vivid plot, the locking range did not impact pipeline performance. In this pipeline implementation, the locking range is represented as a fraction of the dynamic range of vector input to the DOM and is sensitive to input array sizes. This is acceptable for DOM operations in the front-end of the pipeline because the input array size does not change. However, in the back-end, the input sizes to convolution layers in the CNN were not constant. For this reason, the dynamic range of the Vivid dataset was significantly larger than the locking range does not significantly affect performance.

This parametric analysis shows three trends relating to the oscillator parameters. First, in designing coupled oscillator circuits, the coupling asymmetry does not pose a significant problem for slight differences in coupling strength. Second, the locking range needs to remain

small as coupled oscillator circuits with a small locking range performs more accurately than those with a large locking range. The locking range can be decreased by increasing the coupling coefficients, which means, in the case for the circuits in this thesis, increasing the resistor size. Third, noise affects the accuracy of pipeline performance as it would in any analog computational circuit. When designing coupled oscillator circuits, it is important to minimize the effects of noise as much as possible.

6.0 CONCLUSION

This thesis has shown the hierarchy of models for a generic parameterized oscillator, STO, and VO₂ nano-oscillator models. Oscillator models based on current nano-oscillator technologies were created and tested at all levels of the hierarchy. A device model of VO₂ oscillators was created based on data measured in hardware. A four-coupled oscillator circuit model was created using the VO₂ device model with a DOM detector circuit. This circuit was simulated by holding the controlling voltages on three of the oscillators constant while sweeping a fourth to create a 4-Dimensional surface of DOM values. A system model for STO and VO₂ nano-oscillator technologies using a polynomial curve-fit to circuit level data in order to generalize to larger oscillator clusters.

The system level models were verified by testing in MATLAB by comparing the Euclidean Distance Squared calculations and convolution calculations for the oscillator models and MATLAB calculations. In the case of the VO₂ oscillator model, the high standard deviation in the curve-fit model impacted the accuracy of the DOM and convolution calculations. In order to improve accuracy and decrease the standard deviation in model coefficients, better curve-fit models need to be generated either by adjusting coupling or modifying the coupled-circuit configuration.

The system level models were tested in the IPP. The mean VO₂ coefficient model and STO models performed comparable to the IPP using conventional floating-point calculations.

The upper and lower bounds of the VO₂ model were also tested in the IPP and the accuracy decreased compared to the mean VO₂ model. However, this difference was not significant compared to conventional floating-point which showed that the VO₂ system model could perform reliable convolution within a small margin of error.

The generic oscillator model with baseline parameters was shown to perform mathematically equivalent to conventional floating-point calculations in the IPP. A parametric analysis of each oscillator parameter (coupling asymmetry, locking range, and output noise) showed the impact of each parameter on the IPP performance. The IPP was least sensitive to the coupling asymmetry. Conversely, the IPP was more sensitive to locking range, indicating the importance of the coupling between the oscillators. The coupling needs to be adjusted so that the oscillators lock over a small range of frequencies in order to avoid significant impact of locking range. The pipeline performance also worsened for increasing noise. This was not a surprising observation considering noise should be minimized in any analog computation circuit.

The thesis has shown that the system level models created for current nano-oscillator technologies were shown to perform oscillator-based computation comparable to conventional floating-point calculations. Oscillator models can be implemented in image processing algorithms. Further research in these and other nano-oscillator technologies may lead to even more accurate computation models that can eventually be used to accelerate mathematically intensive operations in hardware.

BIBLIOGRAPHY

- [1] C. Huygens, "Horologium Oscillatorium," 1673.
- [2] X. Shi, H. Zhang, Y. Liao, L. Jin, T. Wen, X. Tang and Z. Zhong, "Skyrmions Based Spin-Torque Nano-Oscillator," *IEEE Magnetics Letters*, vol. 8, 2017.
- [3] S. Datta, N. Shukla, M. Cotter, A. Parihar and A. Raychowdhury, "Neuro Inspired Computing with Coupled Relaxation Oscillators," in *Design Automation Conference*, San Francisco, CA, USA, 2014.
- [4] R. N. Alexander, P. Wang, N. Sridhar, M. Chen, O. Pfister and N. C. Menicucci, "One-Way Quantum Computing with Arbitrarily Large Time-Frequency Continuous-Variable Cluster States from a Single Optical Parametric Oscillator," *Physical Review A*, vol. 94, no. 3, 2016.
- [5] A. K. Gupta, J. F. Buckwalter, "A Self-Steering Receiver Array Using Jointly Coupled Oscillators and Phased-Locked Loops," *IEEE Transactions on Microwave Theory and Techniques*, vol. 62, no. 3, pp. 631-644, 2014.
- [6] Y. Fang, M. J. Cotter, D. M. Chiarulli, S. P. Levitan, "Image Segmentation Using Frequency Locking of Coupled Oscillators," in *International Workshop on Cellular Nanoscale Networks and their Applications*, Notre Dame, IN, USA, 2014.
- [7] D. E. Nikonov, G. Csaba, W. Porod, T. Shibata, D. Voils, D. Hammerstrom, I. A. Young and G. I. Bourianoff, "Coupled-Oscillator Associative Memory Array Operation for Pattern Recognition," *IEEE Journal on Exploratory Solid-State Computation Devices and Circuits*, vol. 1, pp. 85-93, 2015.
- [8] D. Vodenicarevic, N. Locatelli, J. Grollier and D. Querlioz, "Synchronization Detection in Networks of Coupled Oscillators for Pattern Recognition," in *International Joint Conference on Neural Networks*, Vancouver, BC, Canada, 2016.
- [9] D. E. Nikonov, I. A. Young and G. I. Bourianoff, "Convolutional Networks for Image Processing by Coupled Oscillator Arrays," arXiv preprint arXiv:1409.4469, 2014.

- [10] D. M. Chiarulli, B. Jennings, Y. Fang, A. Seel and S. P. Levitan, "A Computation Primitive for Convolution based on Coupled Oscillator Arrays," in *IEEE Computer Society Annuyal Symposium on VLSI*, Montpellier, France, 2015.
- [11] B. B. Jennings, "An Analysis of Oscillator-Based Computations for Image Processing," Pittsburgh, PA, 2016.
- [12] D. Ventrice, P. Fantini, A. Redaelli, A. Pirovano, A. Benvenuti and F. Pellizzer, "A Phase Change Memory Compact Model for Multilevel Applications," *IEEE Electron Device Letters*, vol. 28, no. 11, pp. 973-975, 2007.
- [13] R. A. Cobley and C. D. Wright, "Parameterized SPICE model for a phase-change RAM device," *IEEE Transactions on Electron Devices*, vol. 53, no. 1, pp. 112-118, 2006.
- [14] K. Lovin, B. C. Lee, X. Liang, D. Brooks and G.-Y. Wei, "Empirical performance models for 3T1D memories," in *IEEE International Conference on Computer Design*, Lake Tahoe, CA, USA, 2009.
- [15] F. M. Puglisi, L. Larcher, G. Bersuker, A. Padovani and P. Pavan, "An Empirical Model for RRAM Resistance in Low- and High-Resistance States," *IEEE Electron Device Letters*, vol. 34, no. 3, pp. 387-389, 2013.
- [16] K. Yogendra, D. Fan, Y. Shim, M. Koo and K. Roy, "Computing with Coupled Spin Torque Nano Oscillators," in *Asia and South Pacific Design Automation Conference*, Macau, China, 2016.
- [17] N. Shukla, A. Parihar, M. Cotter, M. Barth, X. Li, N. Chandramoorthy, H. Paik, D. G. Schlom, V. Narayanan, A. Raychowdhury and S. Datta, "Pairwise Coupled Hybrid Vanadium Dioxide-MOSFET (HVFET) Oscillators for Non-Boolean Associative Computing," in *IEEE International Electron Devices Meeting*, San Francisco, CA, USA, 2014.
- [18] A. Krizhevsky, I. Sutskever and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," in *Neural Information Processing Systems*, 2012.
- [19] S. Lawrence, C. L. Giles, A. C. Tsoi and A. D. Back, "Face Recognition: A Convolutional Neural-Network Approach," *Transactions on Neural Networks*, vol. 8, no. 1, pp. 98-113, 1997.
- [20] Used by Permission from Donald M. Chiarulli
- [21] P. Maffezzoni, L. Daniel, N. Shukla, S. Datta, A. Raychowdhury, "Modeling and Simulation of Vanadium Dioxide Relaxation Oscillators," *IEEE Transactions on Circuits and Systems*, vol. 62, no. 9, pp. 2207-2215, 2015

[22] <http://ilab.usc.edu/neo2/dataset/tower/training/>

[23] <http://ilab.usc.edu/neo2/dataset/heli/training/>

[24] Personal Correspondence with HRL Laboratories

[25] Nigel Stepp, Yongqiang Cao, Shankar Rao, Praveen Pilly , DARPA UPSIDE Project (unpublished) , HRL Laboratories, *Malibu* , CA.

[26] D. Khosla, Y. Chen, and K. Kim, “A Neuromorphic System for Video Object Recognition,” *Frontiers in Computational Neuroscience*, vol. 8, pp. 147, PMC, Web, 2014.