Curriculum Vitae : Azman M. Yusof

Contact Information

azman {at} my1matrix {dot} org

Education

Doctor of Philosophy

Thesis: Enhancing Post-Harvest *Harumanis* Mango Sorting Process using Fourier Descriptor on a Customizable Vision System

Abstract: The use of vision systems in agriculture has been found to be very beneficial and is quickly becoming somewhat of a necessity for a modern farming system. Various agriculture activities from crop monitoring to post-harvest processes like cleaning and sorting, can be improved tremendously with the help of vision systems. An architecture for a fully-customized vision system implementation that can be used to enhance the sorting process of *harumanis* mango is proposed. The proposed architecture is presented using generic design components that are not dependent on any hardware platform or software library. This makes it very easy to reimplement the system and to customize it to meet any required cost-performance ratio. A hardware-centric approach is used to obtain faster processing time. Using a Field-Programmable Gate Array (FPGA) as additional processing element to handle basic image processing tasks provide more time for the main processing element to focus on high level vision algorithms like object classifications and tracking. A couple of implementations have been tested on different FPGA devices and are found to be more than capable of executing basic image processing tasks at 30 frame-per-second (fps), which is the normal frame rate of most imaging device. This shows that is can easily be inserted in an existing system with minimal modifications, if any. An open-source image processing library written in C (my1imgpro) has been developed and used as the base code for the software side of the proposed architecture. A custom connected-component labelling (CCL) algorithm and a custom border-following algorithm have been developed on top of that library. Both algorithms, along with other software components, have been developed to implement the software components of the proposed architecture. Shaped-based classification of harumanis is also proposed as a new classification method instead of weight and size. Fourier Descriptors (FD) are used as the shape descriptor for harumanis mango classification. Using k-Means Clustering algorithm, all 306 contours that have been successfully detected are classified into 5 categories based on the 24-term FDs generated using centroid distance data. Further analysis shows that the 5 categories can be finalized into 3 fundamental categories (elongated, normal, odd). These can potentially be used as a new standard by which harumanis mango is sorted and graded for commercial market. A fully software-based system using the proposed architecture have been implemented on a desktop computer. The same code have also been successfully compiled and executed on a Raspberry Pi 3 platform, which enables the system to be highly portable. Using live camera feed, both implementations are able to classify all available harumanis mango images into the proposed grades. The processing rate of this implementation implies that the sorting process can be done for a volume of about 15-tonnes per day (assuming 8-hour of daily operation time of a single conveyor line). This

indicates that a hardware-centric implementation should have a much better throughput than that.

Master of Science (2002)

M.Sc. by research at School of Electrical & Electronics Engineering, USM, specializing in Microelectronics (Integrated Circuit Design). Thesis submitted for evaluation in January 2001.

Thesis: Implementation of a Cascadable MLP Neural Processor with Sliding Feeder Technique

Abstract: The design of a 16-bit floating-point MLP neural processor is presented. It utilizes the sliding feeder technique, which reduces the complexity of common neural network interconnections. A new 16-bit floating-point data format is also introduced here. Its ability to match its 32-bit counterpart in calculating the MLP with BEP algorithm is quite remarkable. In a test to train a network to solve the linearly inseparable XOR logical function, the neural processor has successfully converge to an acceptable solution at the same number of training required by a processor using the standard single-precision value. The leading zero detection method has also been improved to save area consumption. The standard MLP with BEP algorithm has been restructured into an object-oriented type of algorithm. This is due to the fact that, instead of having all the data (weight, bias and node values) in a single memory heap, they are distributed among all the cascaded neural processors. This also enables the processor to accommodate the insertion of the sliding feeder technique. Some useful computer software – Code Generator, FPC Tool, and Neural Processor Simulator - has also been developed. All in all, they have contributed to the design, simulation and validation of the neural processor. The serial transmission circuit is only based on simple shift logic. The speed of the serial transmission only affects the network when data needs to be fed forward or backward between the layers. This is because, for data sliding, data transmission is done while the time consuming floating-point calculation takes place.

Bachelor of Engineering (1997)

B.Eng. (Hons.) at School of Electrical & Electronics Engineering, USM, specializing in Microelectronics (Integrated Circuit Design).

Final Year Project: Title: “Design of an Integrated Circuit Implementing the Discrete Wavelet Transform”. A system-level design mainly as a proof of concept. Sub-circuit simulations of the arithmetic unit (transistor-level) was done using pspice (MicroSim).

Selected Courses: Integrated Circuit Design, Microprocessor Systems, Software Engineering.

2026/02/09 22:23

Skills/Knowledge

Embedded Systems Development

Microprocessor/microcontroller board design, Embedded Linux (buildroot), 8085/8051 Assembly,

Verilog/VHDL for FPGA Synthesis

Computer Programming

C/C++, Java, Python, Linux (Bash) Shell scripts, Windows batch file, TCL Shell scripts

Microelectronic Design

Transistor-level design of CMOS/BiCMOS integrated circuits, full custom layout editing of integrated circuits

2026/02/09 22:23

Work Experience

Universiti Malaysia Perlis (UniMAP - formerly known as KUKUM)

Job	Lecturer
Duration	February 1, 2007 - March 31,2026

Kolej Universiti Kejuruteraan Utara Malaysia (KUKUM)

Job	Lecturer
Duration	May 15, 2002 - January 31, 2007

Sapura Technologies Sdn. Bhd.

Job	Systems Engineer
Duration	January 1, 2002 - May 31, 2002

SiRES Labs Sdn. Bhd.

Job	Chief Design Engineer
Duration	February 2, 2001 - September 31, 2001

Intel Technology Sdn. Bhd.

Job	Intern in the Design Automation Group
Duration	March 1, 2000 - May 31, 2000

Universiti Teknologi Petronas (UTP)

Job	Demonstrator (C++ Programming)
Duration	June 1998 - August 1998

Universiti Sains Malaysia (USM)

Job	**Research Personnel (IC Design & Testing)**
Duration	June 2000 - November 2000
Duration	September 1998 - December 1998
Duration	July 1997 - September 1997
Duration	January 1996 - June 1996
Job	**Tutor (C/C++ Programming)**
Duration	January 1999 - April 1999
Duration	July 1996 - October 1996

2026/02/09 22:23

Project Experience

Software Project (2023)

Development of a portable *harumanis* mango sorting system based on a proposed shape-based classification method (using Fourier Descriptor).

Software Project (2019)

Development of a simple software solution for testing image processing algorithm on conveyor-based vision system. It is available as an extension (my1imgproX) to the previously-developed image processing library (my1imgpro).

Software Project (2018)

Development on a basic digital electronics simulator (my1digitaljs) mainly used for teaching purposes. It is written in Javascript and meant to be run in a browser (making it cross-platform).

Soft-Core Development (2016)

An Implementation of Intel 8085-binary-compatible Microprocessor Core using Verilog. Simulated using ModelSim (Altera free version). Just to proof that this kind of project can be finished well within the time period of a bachelor's degree project (a.k.a. Final Year Project). The my1core85 code is also made available at CodeBerg.

Software Project (2014-2017)

Development of Vehicle Monitoring System software. The hardware module is developed by another team member. The software part consists of server-side code (API server) using PHP and client-side code (API client, Google Maps display) using HTML & Javascript. The API server is based on the my1apisrv code.

Software Project (2012-2014)

Software development for Wireless Sensor Network monitoring and data collection. A program (my1wsnbase) that extracts information from a WSN base node through serial port, stores data in sqlite database and acts as a simple web server (all written in C). The web server code uses the MIT-licensed mongoose (which is now a commercial software). This code is also made available at CodeBerg.

Software Project (2011-2012)

Development on an Intel 8085 microprocessor system simulation software (my1sim85) mainly used for teaching purposes. It uses wxWidgets (a cross-platform GUI library) which enables it to be compiled for both Linux and Win32. It is an open source project available at CodeBerg.

Software Project (2011)

Development of an open source software - my1asm85 - a cross assembler for Intel 8085 microprocessor (which is used in tertiary education related to electronics engineering). It is considered complete and currently updated per need basis. It is mainly used as the foundation for another project my1sim85. It is a cross platform - can be compiled for both Linux and Win32 - console application (no GUI).

Systems Development (2007-2008)

Development of Embedded Controller Systems based on ARM controller (running Linux) and Xilinx Spartan3E FPGA for Mobile Robot Platform. Development work was done both on Windows and Linux platform. Built cross-compilers on both platform. Used buildroot to build the base system. FPGA development for servo, camera (plus image processing) and motor controllers.

Internship @ INTEL (2000)

Development of design file format converter using PERL scripts (managed to write a Neural Network MLP-BEP implementation while learning Perl). Feasibility study on RAM generator (generated layout

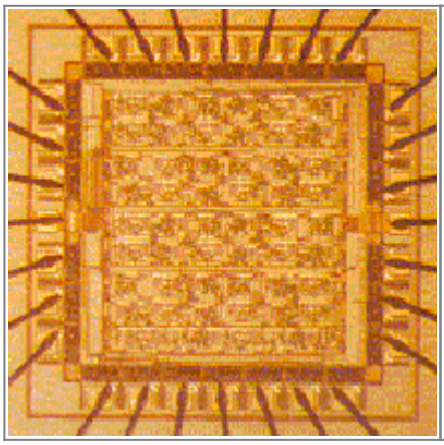
files were also analyzed, but not completed because of the short internship period).

Research Project (1998)

Temperature Dependence Analysis of a Digitally Controlled Oscillator (DCO) using BiCMOS circuits (In comparison with the same design using CMOS circuits. Some theoretical cause and effect were obtained.)

Research Project (1996)

Test & Development of an 8-bit CLA Adder using Novel Dynamic BiCMOS circuits (Transistor level design and simulation. Full-custom layout editing. Fabricated and tested. Photomicrograph of chip's layout are shown below.)

	My firstborn... this is my first chip (I did the design & the layout myself!). It's a working prototype of a novel dynamic 8-bit BiCMOS adder. I produced this during my industrial training. This was 'manually' done - the simulation input is 'handcrafted' netlist file and the layout was done using custom cells (no standard cell libraries!) without any auto-PNR. I also took this picture myself using a camera-fitted microscope (I forgot if that is actually what they call it - it was owned by another department).

Mini Project (1996)

Produced a liquid level control system using 8051 microcontroller.

Mini Project (1994)

Modeled a traffic light controller using an EPROM and a 555 timer.

2026/02/09 22:23

Publications

Journal of Physics: Conference Series 2019

Citation: Azman M. Yusof, Ali Yeon Md Shakaff, Saufiah A. Rahim, “Software solution for Testing

Image Processing Algorithm on Conveyor-Based Vision Systems", *Journal of Physics: Conference Series*, Volume 1372, https://iopscience.iop.org/article/10.1088/1742-6596/1372/1/012059

Abstract: Vision systems have been used in many applications that intends to reduce the need for human operators. This is especially true for tasks that are simple but repetitive in nature, which is largely applicable to most manufacturing and agriculture's post-harvest processes. Many such processes utilize conveyor-based systems where the objects being processed are placed on a conveyor belt that runs through multiple processing stations. Implementing a vision system to capture images of an object that is moving usually requires setting up an imaging device to a working conveyor system. Getting a working conveyor system to be ready can take some time and consequently delay development work on the vision system itself, especially those involving image processing algorithms. This paper proposes a software solution that can be used to expedite initial work on such systems. The solution is written in C and is therefore easily ported to any development machine. A basic image processing library has also been developed so that it does not depend on any development library or suite, which is usually huge in size. Thus, the solution can easily be compiled and run on embedded development boards like Raspberry Pi - for a more portable solution.

JTEC 2018

Citation: Azman M. Yusof, Ali Yeon Md Shakaff, Saufiah A. Rahim, "Implementation of a Hardware-centric Vision System Architecture", *Journal of Telecommunication, Electronic and Computer Engineering (JTEC)*, Volume 10, No 1-15, Pages 123-129, ISSN 2180-1843, eISSN 2289-8131, https://journal.utem.edu.my/index.php/jtec/article/view/4058

Abstract: Currently, most implementations of vision systems still heavily rely on software - computer algorithms run on general purpose microprocessors, like on personal computers. This is understandable since personal computers (PC) are readily available, and software implementations provide flexibility, especially when trying out various algorithms. The need to have real-time vision-based systems influenced developers and researchers towards hardware-based - or at least hardware-assisted - vision systems that are capable of processing huge amount of data from an imaging device in realtime (i.e. embedded vision system). Platforms like DSPs, GPUs and FPGAs are among the commonly used development platforms for a hardware-centric vision system, while ASIC implementations - tagged with a huge development cost - usually have the best performance. This paper compares various possible platforms that are readily available and can be used to develop hardware-centric vision systems. This includes DSPs, GPUs and FPGAs, with some insights on ASIC implementation. Consequently, two implementations of the proposed hardwarecentric vision system architecture are presented. Both implementations managed to process incoming image stream from camera module at 30 frames per second.

Neurocomputing 2014

Citation: Saufiah A. Rahim, Azman M. Yusof, Thomas Bräunl, "Genetically evolved action selection mechanism in a behavior-based system for target tracking", *Neurocomputing*, Volume 133, 10 June 2014, Pages 84-94, ISSN 0925-2312, https://dx.doi.org/10.1016/j.neucom.2013.11.028.

Abstract: The success of a behavior-based system relies largely on its Action Selection Mechanism (ASM) module, which is basically a behavior coordination method of either arbitration or command fusion type. Deciding on the right coordination method for ASM when executing a given mission in an

arbitrary environment can be a huge obstacle. Providing the system with some kind of Artificial Intelligence (AI) to deal with the dynamics of a given task would be highly recommended. In this paper, an evolutionary process has been employed in a behavior-based system to generate a suitable ASM based on a system's mission scenario. A Genetic Algorithm (GA) is used to train the weights of a Multi-layer Perceptron (MLP) feed-forward artificial neural network in identifying a suitable formulation of ASM. Implementation of such systems in a target tracking mission has shown positive results. Depending on the mission scenario, the evolved ASM can dynamically manage the coordination method in order to achieve the overall system objective.

ROVISP (2007) - International Conference on Robotics, Vision, Information and Signal Processing

ISBN: 978-983-43178-1-2

Citation: Azman M.Yusof and Thomas Braunl, “Using Xilinx ML310 Development Board as Test and Development Platform for FPGA-based Embedded Vision System”, pp. 350-353.

Abstract: FPGA-based embedded vision systems are currently the natural option for many vision system researchers and developers. The design and fabrication of a customized FPGA-based board for embedded vision system could be time-consuming and require proper verification of the functionality of the board itself, before using it for its actual task. As an alternative, a ready-for-testing system is usually available in the form of development board like the Xilinx ML310 Development Board. Other than the basic serial communication ports for desktop PC interface, it also has a 256MB DDR SDRAM available. This is very useful when building systems for image processing tasks. A simple custom interface board to a CMOS camera and an LCD display has been fabricated for this purpose. The advantages and the disadvantages of using this development board as a test and development platform will be discussed.

International Journal of Electronics (2001)

ISSN:0020-7217

Citation: S. M. Rezaul Hasan, Lim Chu Aun and Azman Yusof, “600 MHz BiCMOS digitally controlled oscillator for clock recovery & frequency synthesis PLLs”, *International Journal of Electronics*, Taylor & Francis, U.K., vol. 88, no. 5, pp. 529-541, May 2001.

Abstract: A 16-bit digitally controlled BiCMOS ring oscillator (DCO) is described. This BiCMOS DCO design provides improved frequency stability under thermal fluctuations. Simulations of a 5-stage DCO using 1 μm BiCMOS process parameters achieved a controllable frequency range of 90-640 MHz with a linear/quasi-linear range of around 300 MHz. A tiny test chip was fabricated using MOSIS Orbit 2 μm low-cost analogue CMOS process technology that provides a lateral NPN bipolar device option Monotone frequency gain (frequency vs control-word transfer function) with fine stepping (tuning) over several kHz was verified experimentally, thus auguring the prospect of accurate frequency lock in an all-digital phase-locked loop (ADPLL) application Worst-case jitter due to digital control transitions at pathological control-word boundaries for the BiCMOS DCO was observed to be less than 50 ps. This BiCMOS design would thus provide performance enhancement in PLL applications such as clock recovery and frequency synthesis.

IEEE National Symposium on Microelectronics (NSM2001)

Citation: Azman M. Yusof, Ali Yeon Md. Shakaff, Mohd. Noor Ahmad, “Implementation of a Cascadable Neural Processor for Pattern Classification Engine in an Artificial Taste Sensing System”, *2001 IEEE National Symposium on Microelectronics*, 12-13 Nov 2001, Awana Genting Highlands, Malaysia.

Abstract: An implementation of a 16-bit floating-point neuron-like neural processor, known as NEWRON, is presented. NEWRON implements the Multi-Layer Perceptron (MLP) neural network algorithm, with on-chip back-error propagation (BEP) training. It is primarily intended to be the prototype model for a real-time pattern classification engine in an artificial taste/smell sensing application, which is currently being developed. The use of 16-bit floating-point data format increases the dynamic range of calculated digital values. The standard MLP with BEP algorithm has been restructured into an object-oriented algorithm (OOA). In simple words, each NEWRON has a slightly modified algorithm that enables it to independently store and manage all parameters (weight, bias and node values) related to it. OOA also enables NEWRON to accommodate the insertion of the sliding feeder technique, which reduces the complexity of common neural network interconnections. Serial transmissions are used to exchange data between cascaded NEWRONs. The speed of the serial transmission does not affect the network because it is done while the time consuming floating-point calculation takes place.

Citation: Azman M. Yusof, “16-bit Floating-point Arithmetic Unit for Customized Data Processing Unit”, *2001 IEEE National Symposium on Microelectronics*, 12-13 Nov 2001, Awana Genting Highlands, Malaysia.

Abstract: The design of a floating-point arithmetic unit, to be used in a fully customized data processing unit, is presented. It has been designed for a full-custom neural network engine. A 16-bit floating-point data format, which is based on the IEEE 32-bit floating-point data format, has been adopted. Though it has limited accuracy compared to the latter, it offers a wider dynamic range compared to a fixed-point data of equivalent length. When implemented using minimum-sized devices in full-custom static logic cells, at 3.3V power supply, in a 1.2µ CMOS process, the 16-bit floating-point arithmetic unit is capable of around 8.3 MFLOPS (millions floating-point operations per second).

Great Lakes Symposium on VLSI (GLS98)

ISBN:0-8186-8409-7, pp. 71-71

Citation: Azman M. Yusof, S.M. Rezaul Hasan, Lim Chu Aun, “600 MHz Digitally Controlled BiCMOS Oscillator (DCO) for VLSI Signal Processing & Communication Applications”, *Great Lakes Symposium on VLSI ’98*, 19-24 Feb 1998, Lafayette, Louisiana.

Abstract: A 16-bit digitally controlled BiCMOS ring oscillator (DCO) is described. This BiCMOS DCO design provides improved frequency stability under thermal fluctuations compared to a CMOS DCO design. Simulations of a 5-stage DCO using a 1-um BiCMOS process parameters achieved a controllable frequency range of 90-640 MHz with a linear/quasi-linear range of around 300MHz. Monotone frequency gain (frequency vs control word transfer function) with fine stepping (tuning) in several KHz was verified. This augurs the prospect of accurate frequency lock in a BiCMOS all digital PLL (ADPLL) application in digital VLSI communication systems. Worstcase jitter due to digital control

transitions at pathological control word boundaries for the BiCMOS DCO was observed to be less than 50 ps, which is lower than that for the CMOS DCO.

2026/02/09 22:23

Reference

Available upon request

From:
https://azman.my1matrix.cc/ - **Azman @MY1**

Permanent link:
https://azman.my1matrix.cc/doku.php?id=archive:resume

Last update: **2026/04/05 00:57**