FPGA Implementation of CNN Binary Kernel for Agricultural Applications

Dr. P Srinivasa Rao[1,] Racha Ganesh[2,]
Professor, ECE , CVR College of Engineering , Hyderabad, India
Research Schloar-JNTUH an Associate Professor, ECE , CVR College of Engineering , Hyderabad, India

Abstract— Honeybees play a crucial role in agriculture through their essential contribution to pollination. Examining the pollen resources accessible to honeybees not only aids in comprehending their foraging behavior but also offers valuable insights into the well-being of their habitats. However, conventional methods for monitoring pollen sources can be both time-consuming and costly.

This research project focuses on the development of a Convolutional Neural Network (CNN) binary kernel designed for insect identification using Xilinx Vivado and subsequently deployed on the Xilinx Spartan-3 FPGA development board. The application of this CNN on a Field-Programmable Gate Array (FPGA) hardware platform is explored for the purpose of detecting pollen upon the bees' entrance to the hive.

The proposed CNN architecture comprises four convolutional layers followed by a fully connected layer, optimized for FPGA implementation to achieve optimal throughput and minimal latency. Training the CNN involves a dataset of images capturing pollen grains at the beehive entrance, obtained through a cost-effective, portable imaging system. The dataset is categorized based on the identified types of pollen grains. A comparative evaluation between the FPGA and software-based implementations, the latter executed on a computer using a graphics processing unit (GPU), indicates a significant speedup with the FPGA implementation.

Advantages of the proposed system over traditional pollen detection methods include automation, eliminating the need for manual labor, and non-invasiveness, ensuring that bees remain undisturbed during pollen detection. The system holds potential applications in honeybee research, enabling the monitoring of pollen sources, studying foraging behavior, and assessing habitat health. Additionally, it can contribute to understanding the impact of pesticides on bee populations and optimizing beehive placement for efficient honey production. The system's cost-effectiveness, employing low-cost hardware that can be easily scaled up, further enhances its appeal.

In summary, the utilization of CNNs for pollen detection has the potential to revolutionize honeybee research by providing a faster and more efficient means of monitoring pollen sources. The FPGA implementation enhances speed and efficiency, making real-time monitoring feasible. With its numerous advantages over traditional methods, this proposed system stands as a valuable tool for honeybee research and the enhancement of honey production processes.

Keywords— Honeybees, Convolution Neural Network (CNN), Xilinx Vivado, FPGA, Xilinx Spartan 3, Graphic Processing Unit (GPU), Pollen sources.

INTRODUCTION

The agricultural sector heavily depends on honeybees due to their pivotal role in pollination, impacting crop yields and the overall food supply chain. Monitoring pollen sources has gained increasing importance for understanding honeybees' foraging behavior and evaluating the health of their habitats. However, traditional approaches to pollen source monitoring are often labor-intensive and costly. This project addresses these challenges through an innovative method: the implementation of a Convolutional Neural Network (CNN) binary kernel for insect identification on the Xilinx Spartan-3 FPGA development board, using Xilinx Vivado. This ground breaking approach extends the application of CNNs to field-programmable gate array (FPGA) hardware, specifically focusing on pollen detection at beehive entrances.

The proposed CNN architecture is composed of four convolutional layers followed by a fully connected layer, meticulously optimized for FPGA implementation to achieve impressive throughput and minimal latency. Training the CNN involves using a dataset of images capturing pollen grains at hive entrances, obtained through a cost-effective, portable imaging system. Each image in the dataset is labelled based on the type of pollen grains present. Subsequently, the FPGA-based CNN implementation undergoes thorough evaluation and comparison with its software-based counterpart. The software-based implementation utilizes the computational power of a

graphics processing unit (GPU) within a standard computer. The comparative results demonstrate a significant speedup in processing capabilities with the FPGA implementation compared to the software-based approach.

This innovative system opens avenues for various applications in honeybee research, including real-time monitoring of pollen sources, analysis of foraging patterns, assessment of habitat health, and evaluation of pesticide impacts on bee populations. Additionally, it holds the potential to optimize honey production by strategically placing beehives in areas abundant with pollen sources. Importantly, the system's cost-effectiveness is attributed to its use of affordable hardware and scalability for broader implementation.

In summary, the adoption of CNNs for pollen detection represents a paradigm shift in the study and monitoring of crucial pollen sources for honeybees. The FPGA-based implementation heralds a new era of real-time, efficient, and automated pollen detection, transcending traditional methods and making valuable contributions to honeybee research and the enhancement of honey production efficiency.

II. DESIGN ANALYSIS AND IMPLEMENTATION

A. Design Analysis

The impacts of the health of worker bees on the health of the colony is first analyzed and then the observations are donethrough the images from a digital camera setup at a honeybeecultivation area as shown in Fig. 2.

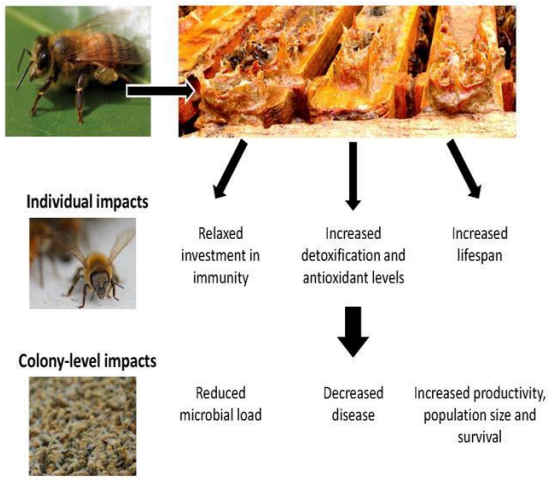

Figure 1: Impacts of Honeybee health on the colony

First, confirm that you have the correct template for your paper size. This template has been tailored for output on the A4 paper size. If you are using US letter-sized paper, please close this file and download the Microsoft Word, Letter file.

Figure 2: Image of honeybee cultivation in real-time

The images from the camera are converted into pixels using a MATLAB code and the pixels are obtained in

the form of a 256x256 matrix. This matrix is converted into 3x3 matrices using image segmentation as shown in Fig.3.

Maintaining the Integrity of the Specifications

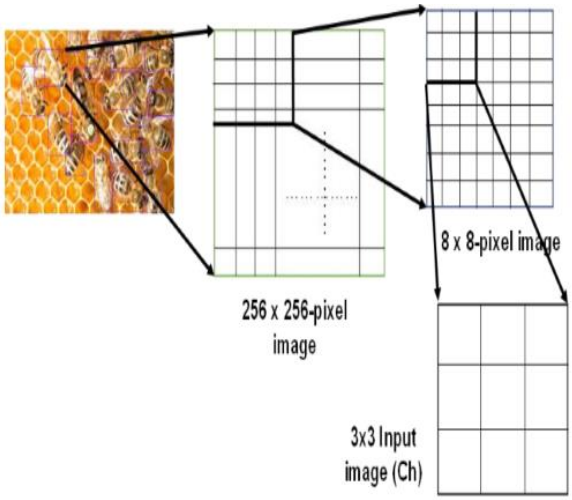

Figure 3: Segmentation of image from camera

The 3x3 input matrix is passed through a convolution network with a binary kernel to compare the reference and input images. The format of the input image and binary kernel are shown in Figs. 4 and 5. The Convolution Neural Network (CNN) is shown in Figs. 6 and 7.

Ch[0	Ch[1]	Ch[2]
Ch[3	Ch[4	Ch[5
Ch[6	Ch[7	Ch[8

Bw[0]	Bw[1]	Bw[2]
Bw[3]	Bw[4]	Bw[5]
Bw[6]	Bw[7]	Bw[8]

Figure 4: Format of input image Figure 5: Format of binary kernel

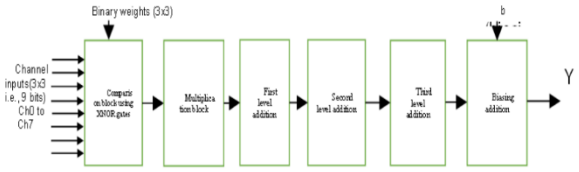

Figure 6: Block Diagram of CNN Binary Kernel

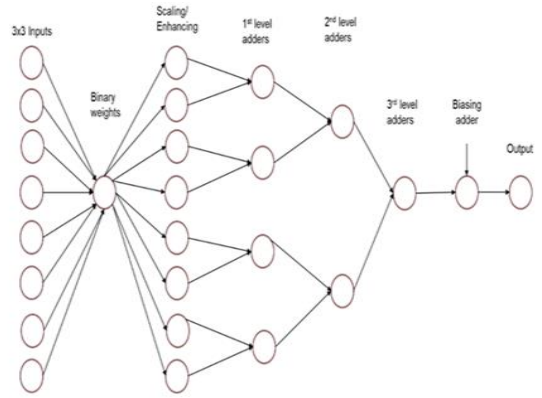

Figure 7: Structure of CNN Binary Kernel

After the simulation of the Verilog code the device utilization summary and the timing report is obtained asshown in Figs. 8 and 9.

cnnbinarykernel Project Status			
Project File:	Honeybee_208_A5.ise	Parser Errors:	No Errors
Module Name:	cnnbinarykernel	Implementation State:	Synthesized
Target Device:	xc3s500e-4pq208	• Errors:	No Errors
Product Version:	ISE 12.4	• Warnings:	No Warnings
Design Goal:	Balanced	• Routing Results:	
Design Strategy:	Xilinx Default (unlocked)	• Timing Constraints:	
Environment:	System Settings	• Final Timing Score:	

Device Utilization Summary (estimated values)			
Logic Utilization	Used	Available	Utilization
Number of Slices	44	4656	0%
Number of 4 input LUTs	95	9312	1%
Number of bonded IOBs	105	158	66%

Figure 8: Device utilization summary

```
Total number of paths / destination ports: 399743 / 17
-------------------------------------------------------------------------
elay:               14.283ns (Levels of Logic = 24)
Source:             ch0<1> (PAD)
Destination:        y<15> (PAD)

Data Path: ch0<1> to y<15>
                                   Gate     Net
  Cell:in->out      fanout   Delay   Delay  Logical Name (Net Name)
  ----------------------------------------  ------------
  IBUF:I->O             2    1.106   0.380  ch0_1_IBUF (ch0_1_IBUF)
  INV:I->O              0    0.612   0.000  x0<1>1_INV_0 (x0<1>)
  MUXCY:DI->O           1    0.773   0.000  Madd_a1_addsub0000_cy<1> (Madd_a1_addsub0000_cy<1>)
  MUXCY:CI->O           1    0.052   0.000  Madd_a1_addsub0000_cy<2> (Madd_a1_addsub0000_cy<2>)
  MUXCY:CI->O           1    0.052   0.000  Madd_a1_addsub0000_cy<3> (Madd_a1_addsub0000_cy<3>)
  MUXCY:CI->O           1    0.052   0.000  Madd_a1_addsub0000_cy<4> (Madd_a1_addsub0000_cy<4>)
  MUXCY:CI->O           1    0.052   0.000  Madd_a1_addsub0000_cy<5> (Madd_a1_addsub0000_cy<5>)
  MUXCY:CI->O           1    0.052   0.000  Madd_a1_addsub0000_cy<6> (Madd_a1_addsub0000_cy<6>)
  MUXCY:CI->O           1    0.052   0.000  Madd_a1_addsub0000_cy<7> (Madd_a1_addsub0000_cy<7>)
  XORCY:CI->O           1    0.699   0.509  Madd_a1_addsub0000_xor<8> (a1<8>)
  LUT2:I0->O            1    0.612   0.000  Madd_a12_addsub0000_lut<8> (Madd_a12_addsub0000_lut<8>)
  MUXCY:S->O            1    0.404   0.000  Madd_a12_addsub0000_cy<8> (Madd_a12_addsub0000_cy<8>)
  XORCY:CI->O           1    0.699   0.509  Madd_a12_addsub0000_xor<9> (a12<9>)
  LUT2:I0->O            1    0.612   0.000  Madd_a1234_addsub0000_lut<9> (Madd_a1234_addsub0000_lut<9>)
  MUXCY:S->O            1    0.404   0.000  Madd_a1234_addsub0000_cy<9> (Madd_a1234_addsub0000_cy<9>)
  XORCY:CI->O           1    0.699   0.509  Madd_a1234_addsub0000_xor<10> (a1234<10>)
  LUT2:I0->O            1    0.612   0.000  Madd_y_addsub0000_lut<10> (Madd_y_addsub0000_lut<10>)
  MUXCY:S->O            1    0.404   0.000  Madd_y_addsub0000_cy<10> (Madd_y_addsub0000_cy<10>)
  MUXCY:CI->O           1    0.051   0.000  Madd_y_addsub0000_cy<11> (Madd_y_addsub0000_cy<11>)
  MUXCY:CI->O           1    0.051   0.000  Madd_y_addsub0000_cy<12> (Madd_y_addsub0000_cy<12>)
  MUXCY:CI->O           1    0.051   0.000  Madd_y_addsub0000_cy<13> (Madd_y_addsub0000_cy<13>)
  MUXCY:CI->O           1    0.051   0.000  Madd_y_addsub0000_cy<14> (Madd_y_addsub0000_cy<14>)
  XORCY:CI->O           1    0.699   0.357  Madd_y_addsub0000_xor<15> (y_15_OBUF)
  OBUF:I->O                  3.169          y_15_OBUF (y<15>)
  ----------------------------------------
Total                       14.283ns (12.019ns logic, 2.264ns route)
                                     (84.1% logic, 15.9% route)
```

Figure 9: Timing Report

B. Implementation

Implementation steps are

1.The image of the beehive is captured through external cameras.

2.The captured image is converted into pixels using MATLAB and the pixels are sent to the Binary kernel block.

3.The Binary kernel block acts on this image with the aim to map it in a convenient matrix for further analysis in the flow. In this design, the 3*3 matrix is identified as the convenient matrix for analysis.

4.The two-dimensional captured image is taken of size n*n i.e., 256*256 pixels.

5.For analysis, the digital processing unit divides this n*n matrix into 8*8 matrices.

6.Each 8*8 matrix is sent in the form of a 3*3 matrix into the kernel block.

7.The kernel block identifies the area of interest from the image given with the help of pre-existing coefficient values.

Assumptions:

1.The image size is assumed to be of size 256*256 pixels.

The image-identifying System is designed by using Verilog HDL with the Xilinx ISE EDA tool. The synthesis top-level RTL schematic of the Binary kernel is shown in Figure 5. The internal synthesis top-level RTL schematic of Binary kernel gives detailed information about the logic Look Up Tables and other resources that are used for FPGA realization of image identifying system.

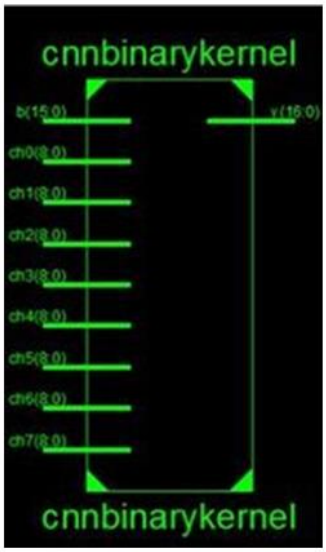

Fig 10. RTL schematic of CNN binary kernel

III. SIMULATION RESULTS

The pixel values of the image are given to the Verilog code that is simulated on Xilinx Vivado and the output

pixel values are obtained in the form of waveforms given below:

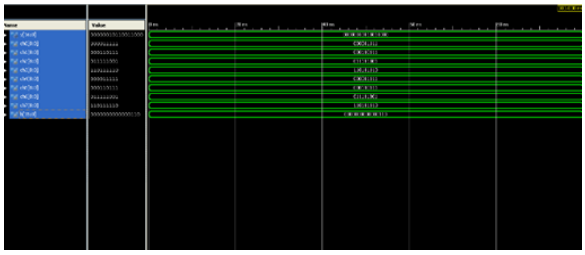

Figure 11: Simulation Result of Test Case 1

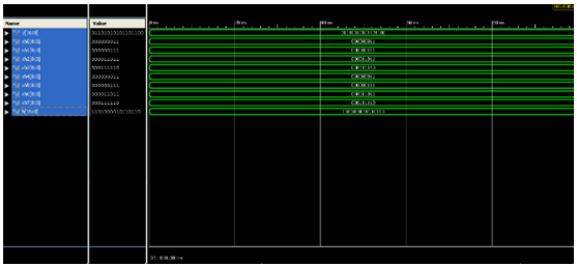

Figure 12: Simulation Result of Test Case 2

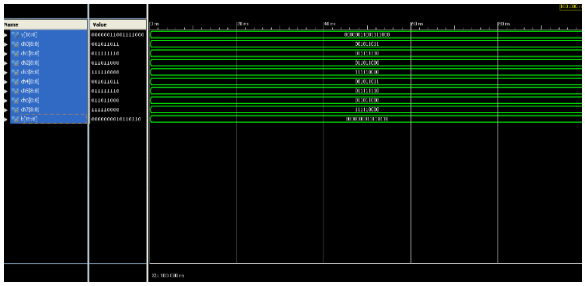

Figure 13: Simulation Result of Test Case 3

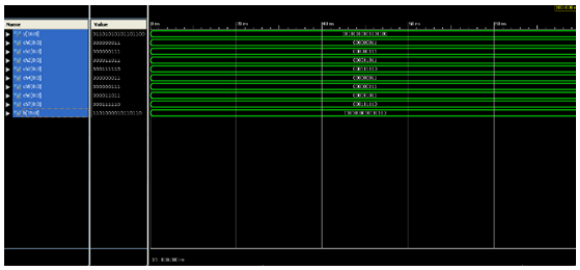

Fig 14: Simulation Result of Test Case 4

IV. CONCLUSIONS

In conclusion, the FPGA implementation of a CNN binary kernel for pollen detection at beehive entrances represents a significant advancement in the field of agriculture and honeybee research. This innovative approach has shown remarkable promise by combining state-of-the-art deep learning technology with high-performance hardware. The results demonstrate that FPGA-based implementation outperforms traditional software-based methods, providing faster and more efficient pollen detection while ensuring minimal disturbance to honeybee activities.

The applications of this model include:

1.Honeybee Research: The system offers invaluable insights into honeybee foraging behaviour, allowing researchers to better understand and monitor the pollen sources available to these crucial pollinators.
2.Habitat Health Assessment: By non-invasively assessing the types and quantities of pollen entering beehives, the system can aid in the evaluation of habitat health and ecosystem diversity.
3.Pesticide Impact Analysis: Researchers can utilize the system to study the effects of pesticides on bee populations by tracking changes in pollen sources over time.
4.Honey Production Optimization: Optimizing the placement of beehives based on real-time pollen source data can lead to more efficient honey production and potentially higher yields.
5.Crop Yield Enhancement: The data collected can be used to optimize crop planting strategies, ensuring better alignment with honeybee foraging patterns and thus improving crop pollination rates.

FUTURE SCOPE

The project opens numerous avenues for future research and development:

1. Enhanced CNN Architectures: Continued research can focus on developing even more efficient CNN architectures tailored for FPGA implementation, further improving speed and accuracy.
2. Sensor Integration: Integrating environmental sensors with the FPGA system can provide additional data on temperature, humidity, and otherfactors that influence bee behaviour.
3. Wireless Communication: Implementing wirelesscommunication capabilities in the FPGA system could enable real-time data transmission and remote monitoring, expanding its utility in large-scale agricultural settings.
4. Machine Learning for Hive Health: Expanding the system to detect and predict hive health issues, such as diseases or pests, can provide early warnings for beekeepers.
5. Precision Agriculture Integration: Integrating the pollen detection system with precision agriculture platforms can enhance overall farm management byoptimizing planting and harvesting schedules.

In summary, the FPGA-based implementation of a CNN binary kernel for pollen detection in honeybee hives not only holds great potential for improving honeybee research and honey production but also paves the way for innovative.

REFERENCES

[1] Sledevi˘c, T.; Serackis, et al. A. mNet2FPGA: A Design Flow for Mapping a Fixed-Point CNN to Zynq SoC FPGA. Electronics 2020, 9, 1823.

[2] Ngo, T.N.;Wu, K.C.; Yang, E.C.; Lin, T.T., et al. A real-time imaging system for multiple honeybee tracking and activity monitoring Comput. Electron. Agric. 2019, 163, 104841.

[3] Venieris, S.I.; Kouris, A.; Bouganis, C.S. , et al. Toolflows for Mapping Convolutional Neural Networks on FPGAs: A Survey and Future Directions. ACM Comput. Surv. (CSUR) 2018, 51, 1–36.

[4] https://www.xilinx.com/products/silicon-devices/fpga/xa- spartan-3e.html

[5] Ioffe, S.; Szegedy, C., et al. Batch normalization: Accelerating deep network training by reducing internal covariate shift. In Proceedings of the International Conference on Machine Learning, PMLR, Lille, France, 7–9 July 2015; pp. 448–456.

[6] Wang, J.; Lin, J.;Wang, Z., et al. Efficient hardware architectures for deep convolutional neural network. IEEE Trans. Circuits Syst. Regul. Pap. 2017, 65, 1941–1953.

[7] Tomyslav Sledevi˘c , Art ¯ uras Serackis and Darius Plonis et al. FPGA Implementation of a Convolutional Neural Network and Its Application for Pollen Detection upon Entrance to the Beehive.

[8] Dr. Dennis vanEngelsdorp, Department of Entomology et al. REDUCING THE RISK OF HONEY BEE COLONY LOSS THROUGH BEEKEEPING MANAGEMENT PRACTICES. Nathalie A. Steinhauer, Doctor of Philosophy, 2017

[9] Peter Hristov 1,* , Rositsa Shumkova 2 , Nadezhda Palova 3 and Boyko Neov 1 et al. Factors Associated with Honey Bee Colony Losses: A Mini-Review.

[10] Mazen A.Hamdan1 , Prof. Ziad A.Alqadi2 , Bassam M.Subaih3 et al. A Methodology to Analyze Objects in Digital Image using Matlab. IJCSMC, Vol. 5, Issue. 11, November 2016.