Raspberry Pi 4 / 3 B+ with Neural Compute Stick 2

Van Chau, Richard Chear, Abraham Alidadi, Timothy Calvert, and Mohamed El-Hadedy
Electrical and Computer Engineering Department, Cal Poly Pomona
3801 W Temple Ave, Pomona, CA 91768

***Abstract*—A study and implementation of Intel's Movidius Neural Compute Stick 2 with Raspberry Pi 4 and Raspberry 3. The two computers run facial detection software using the neural compute stick and compare the execution time, accuracy, and power consumed. A raspberry pi is an easy to use mini-computer that can easily interface with the AI structure neural stick device. It is found that the new model(raspberry 4) computes with faster execution speed as expected as well as a higher degree of accuracy in facial detection given a range of distorted and normal facial images.**

***Index Terms*—Raspberry Pi, Neural Compute Stick 2, OpenCV, OpenVINO, Raspberry Pi camera module**

I. Introduction

UNLIKE other biometric techniques such as fingerprint, signature, and iris recognition, facial recognition technology has the advantage of not requiring any direct collaboration with a subject. It is capable of identify human faces from a digital images, or video frames against a database of faces. This is done by pinpointing and measuring facial features from given image. In recent times, facial recognition has attracted a lot of attention in the field of machine learning as it opens to the door to countless applications.

A company named Movidius launched the world's first deep learning USB stick containing a Myriad 2 Vision Processor (VPU)[2]. Years later, after Intel bought Movidius, the company unveiled the second generation of the USB stick - Myriad X VPU. It is the first of its class to feature the Neural Compute Engine, a hardware accelerator for neural network deep-learning inferences. Being a newer innovation, Myriad X VPU does not have many supports for using with non-x86-64 architectures[2].

Intel then releases Intel Movidius Neural Compute Stick (NCS) for learning AI programming. It has been used on millions of smart security cameras, gesture-controlled drones, industrial machine vision equipment, and more. In 2018, the Intel Movidius Neural Compute Stick 2 (NCS) with more supports comes out; with computer vision and AI to your IoT. What looks like a standard USB thumb drive is built on the latest Intel Movidius Myriad X VPU, which features the neural compute engine - a dedicated hardware accelerator for deep neural network interfaces. With more compute cores than the original version and access to the Intel Distribution of OpenVINO toolkit, the Intel NCS2 has a 8x performance boost over the previous generation[1].

For this implementation, digital images are ran through the software and compare facial detection results between a Raspberry Pi 3 B+ and Raspberry Pi 4. This includes CPU execution time comparison, and accuracy percentage of detection of the two boards.

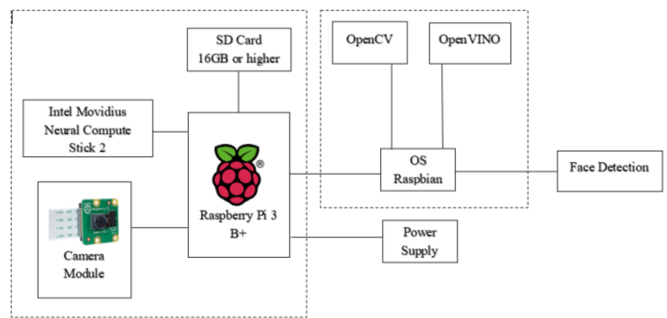

Fig. 1. Overall system block diagram

II. IMPLEMENTATION

A. Raspberry Pi Installation

A microSD card with at least 16GB will be needed for this implementation which will be used to download the Raspberry Pi OS. After the OS is downloaded, the default graphical environment is installed. Then the Raspberry Pi was connected to the Internet, and all the software was updated. Afterwards, the command 'sudo raspi-config' was ran in the terminal[3]. The camera must be enabled through the interface selection. After downloading and configuring all the setting, the raspberry Pi is rebooted. The Raspberry Pi camera module should successfully be accessible whenever the Raspberry Pi is booted up.

Once the terminal is opened again, run sudo modprobe bcm2835-v4l2 (note that's a lowercase L, not a 1), then confirm /dev/video0 now exists by navigating to /dev and running ls. The modprobe command runs each time, so it should be added to the startup.

Some of the applications that are compiled may run out of memory. This is due to the default swap partition size is 100 megabytes. The command 'sudo nano /etc/dphys-swapfile' is ran to solve this by increasing it. It is changed from 100 to 1024. Save, reboot and run free -h to confirm the swap size is increased. Finally, cmake is installed with sudo apt-get install cmake[3].

B. OpenVINO Installation

Intel provides adequate instructions for setting up their device. Installing to a folder on the desktop is preferable for simplicity. OpenVINO is downloaded for Raspberry Pi (.tgz file), then copied to /home/pi/Desktop and untar it with tar xvf filename.tgz[1].

The install scripts need to explicitly know where they are located, so in the OpenVINO folder, enter the /bin directory and open setupvars.sh in any text editor[2]. Replace with the full path to your OpenVINO folder.

The later scripts need this script loaded, so load setupvars.sh every time a terminal is opened. Close the terminal window and open it again to apply this change.

Next, set up the USB rules that will allow the NCS2 to work. First, add a user to the user group that the hardware requires with sudo usermod -a -G users.

Enter the install dependencies folder of the OpenVINO install, and install development rule folder. Now plug in the NCS2 and run dmesg, it should be able to be correctly detected at the end of the output.

III. DATA ANALYSIS

To analyze the efficiency and accuracy of facial recognition software, it is preferred to utilize both Raspberry 3 B+ and Raspberry 4. The execution times of the facial recognition programs on both devices are observed to compare their speed and accuracy. To achieve a better benchmark for the program, several images are used to test the performance consisting of different impairments. The quality and orientation of these images vary to provide an adequate test to the program's capabilities. Certain accessories, such as glasses, can often cause confusion for facial recognition software. Thus, said pictures are included in the tests.

A. Raspberry Pi 3 B+

For the default experiment, facial detection on single-face photos is examined. Without any disruptions, the 2592x1944 photo gives a 9.69-second execution time via Raspberry Pi 3 B+. Out of all the experimental photos, the undisturbed photo takes a much longer time than others. One possible speculation for this may be because of the compiled photos' pixels; there must be a proportional relationship between number of pixels (quality of the photos) to execution time.

For some unusual cases, photos of a mailbox, a camera, a Halloween mask, and fruits that eyes and a mouth are tested. Fruits, mailbox, and camera yield no successful results. However, the Halloween mask results in a successful detection. This is probably due to the apparent human facial features on the mask (Figure 3).

Single-face photos with embedded disruptions (e.g. masks, glasses, side-way views), and the unusual cases where features are not from a human (Figure 3 and Figure 5) are also tested. The masked face photo gives an accurate facial detection test; with 8.33-second execution time via Pi 3 B+. It should also be noted that the photo is 379x675. Some other blurred photos are tested; with 532x600 and 654x680, respectively. All of the above pictures' faces get successfully identified. This concludes that even with disruptions, NCS2 is able to detect humans' facial features. However, in later inspections, it's observed that there are some group photos where disruptions cause detection errors. As an example, a blurry group photo of 40+ people only gives about 20 successful detection. The proposed hypothesis for this error is the quality of the photos. With high quality photos, every exposed faces get recognized, whereas the pixelated photos get unpredictable errors. Execution time details are shown in Figure 4.

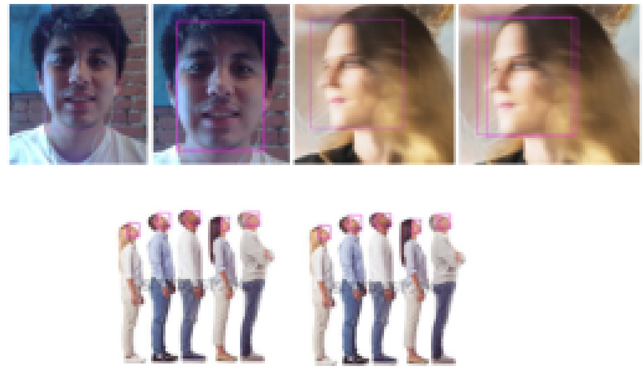

Fig. 2. Normal (Left - Pi 3 B+, Right - Pi 4) Blurred (Left - Pi 3 B+, Right - Pi 4) Line of people (Left - Pi 3 B+, Right - Pi 4)

Fig. 3. Halloween mask (Left-Raspberry Pi 4, Right-Raspberry Pi 3 B+

File No. and Pixel	Raspberry Pi 3 B+		Raspberry Pi 4	
	Real time (seconds)	Δ CPU temp. (after - idle)	Real time (seconds)	Δ CPU temp. (after - idle)
1	8.306	5.9	5.757	7.4
2	8.426	6.2	6.145	8.7
3	8.257	5.3	5.732	6.9
4	8.121	6.3	5.721	5.8
5	9.096	5.3	5.751	6.3
6	8.357	6.5	5.771	6.0
7	8.436	5.9	5.777	5.8
8	8.219	5.4	5.789	8.0
9	8.126	4.3	5.756	6.3
10	8.229	6.7	5.812	5.8
11	9.690	6.4	5.734	7.4
12	8.333	8.6	5.744	6.3
13	8.264	7.5	5.788	6.7
14	8.322	6	5.741	6.3

Fig. 4. Execution Time and Temperature's Difference using Raspberry Pi 3 B+ and Raspberry Pi 4

B. Raspberry Pi 4 B(2GB)

In the case for the raspberry Pi 4, the same photos are used from the Raspberry Pi 3 B+ tests. Large group photos that contains many faces doesn't seem to have any noticeable impact on execution time when compared to photos that only has one face or no human faces. The largest execution time out of all the photos has the largest resolution of 1920 x 1080 and an execution time of 6.145 s. This is possibly due to the fact that the Raspberry Pi 4 has more pixels to analyze and determined if those pixels are faces. The other photos that are used only had a resolution of no greater than 540x680, no where near the resolution of photo 2. In conjunction with the Raspberry Pi 3, the Pi 4 does not recognize the photos of objects that look like faces, such as the mailbox or fruit photos (Figure 5). Photos that have disruptions such as masks on a face are able to be detected. Meaning that the pre-trained model is accurate enough where it doesn't give a false positive and can detect faces that has visual disruptions, such as masks or glasses.

Fig. 5. Unusual cases (Left-fruit, Middle-mailbox, Right-camera

The Raspberry Pi 4 uses a Broadcom BCM2711 SoC with a 1.5 GHz 64-bit quad core ARM processor with 2 GB of ram[1]. The CPU on the Raspberry Pi 4 is .1 GHz faster than that of the Raspberry Pi 3 and also have 1 more GB of ram[1]. The extra ram and CPU clock speed yield noticeable results when benchmark the facial recognition program. From the tests, the overall average time to execute the facial recognition program is about 2.5 seconds faster.

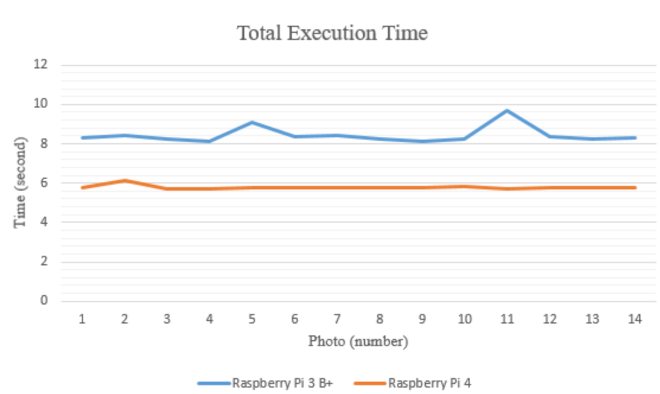

Fig. 6. Total execution time using Raspberry Pi 3 B+ and Pi 4

C. Work Comparison

Using the same NCS2 and experiments, the data table shows that Raspberry Pi 4 yields faster and more accurate results.

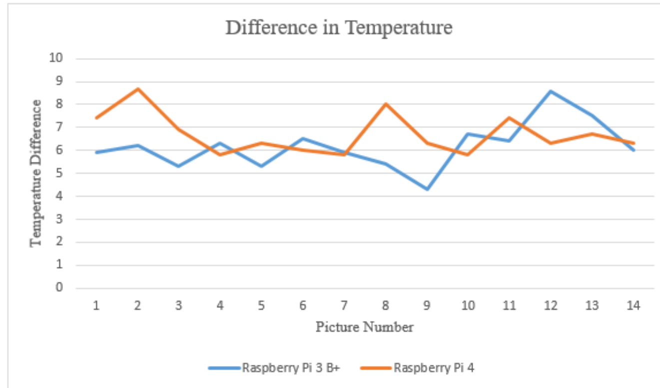

Fig. 7. Difference of temperature between Raspberry Pi 3 B+ and Pi 4 (AFTER - IDLE)

This makes sense as Raspberry Pi 4 sports a faster 1.5GHz clock speed processor than 1.4GHz on Raspberry Pi 3 B+[1]. Moreover, this Raspberry Pi 4 is a 2GB LPDDR4 SDRAM model, while Raspberry Pi 3 B+ is only the entry-level 1GB LPDDR2 SDRAM[1]. The difference in total execution time between Raspberry Pi 3 B+ and Raspberry Pi 4 is approximately 3 seconds, as shown in Figure 3. This is a tremendous increase in performance (Figure 6).

On top of having faster performance time, Raspberry Pi 4 also executes more accurate facial detection than Raspberry Pi 3 B+. Using the same blurred 40+ people image, Raspberry Pi 3 B+ only manages to detect 26 faces, while Raspberry Pi 4 detects a total of 48 faces, as shown in Figure 5. The deducted reason is that Raspberry Pi 4 has more memory than Raspberry Pi 3 B+; Raspberry Pi 4 is able to hold more pixels, hence it can detect more faces than the other Raspberry Pi.

Fig. 8. Facial detection via Raspberry Pi 3 B+ (Left) and Raspberry Pi 4 (Right)

Putting the difference in execution time to perspective,it's observed that there may be some correlation between the difference of temperature and execution time. If the Pi takes less time to execute tasks, then it makes the Pi heat up more. A table of the difference temperature of the two Pi are recorded (Figure 4). The average idle temperature of Raspberry Pi 3 is 43.5 (in Celcius), while Raspberry Pi 4's average idle temperature is 48.2 (in Celcius). This concludes that the faster the board runs, the higher the temperature the board may have. However, the difference in temperature (after-task to idle) does not seem to show any correlation (Figure 4). The two Pi's points in Figure 7 don't show any apparent elevation or decline.

IV. Conclusion

In this experiment, both the Raspberry Pi 3 B+ and Raspberry Pi 4 with the same Neural Compute Stick 2 are observed and compared with the use of facial detection. The experiment compared the execution time, temperature and speed of the two computers. From the results of the execution time and speed tests, it can clearly be seen that Raspberry Pi 4 is more adept in these ares. Comparing the outputs of these two computers also shows that the Raspberry Pi 4 yields more successful facial detection. However, it is important to note that the Raspberry 4 gets significantly hotter than the Raspberry 3 B+. It can also seen that both Pis were able to correctly detect single person photo; normal, blurred, with disruptions (masks, side-way view). Furthermore, when given the unusual photos (fruits, camera), both devices do not detect any human faces. Thus, it can be realized that the Raspberry Pi 3 B+'s interface is not completely inferior to the Raspberry 4. Although, in the case of the example photo of a blurred 40 person photo, Raspberry Pi 4 correctly detects more people than Raspberry Pi 3 B+. The most probable factor to causing this may be the memory difference between the two Pis. As such, the Raspberry 4 does conclusively have a better processing power, which can significantly impact results for certain applications.

References

[1] S. McManus and M. Cook, *Raspberry Pi For Dummies*, 4th ed. John Wiley and Sons, 2021.

[2] "Intel Movidius Neural Compute Stick." https://software.intel.com/content/www/us/en/develop/articles/intel-movidius-neural-compute-stick.html (accessed: Dec. 01, 2021).

[3] S. Depp. "FaceDectectionEdgeComputing." https://github.com/stevedepp/FaceDetectionEdgeComputing (accessed: 2021).