Overview of Use of Raspberry Pi in Implementation of Machine Learning and Image Processing

Savion Mario Sequeira
B.Tech Student,
Department of Computer Science and Engineering,
New Horizon College of Engineering, Bangalore, India

S. Shanmuga Priya
Senior Assistant Professor
Department of Computer Science and Engineering,
New Horizon College of Engineering, Bangalore, India

Abstract: **Raspberry Pi is a low cost credit sized computer that anyone can purchase and use. It is necessary to understand the usage of it and how to implement it with current technologies. This paper will focus on how one can implement Machine Learning techniques and Image processing effectively in a Raspberry Pi.**
Keywords: **Raspberry Pi, Machine Learning, Image Processing**

1. Introduction

There are three major concepts introduced in more detail in this paper: Raspberry Pi, Machine Learning and Image Processing.

Raspberry Pi, is a readily available low cost, credit sized computer that can be plugged into any monitor or TV and can be accessed using any keyboard and mouse with a USB connector. Raspberry Pi uses its own operating system named Raspbian OS. It is a free operating system that is based on a UNIX like operating system named Debian OS and is optimized for the Raspberry Pi hardware. The important thing to remember is all UNIX commands work in Raspbian OS. There are several models of Raspbian Pi available and Raspberry Pi 3 Model B+ is referenced here.

Machine Learning, is a rapidly growing field in today's world. Machine learning essentially is to detect patterns in data which have some meaning in context to the problem. Nowadays the world is engulfed by machine learning based technology, from search engines which figure out how to give the best possible results to advertisers figuring out how to present the best possible advertisements, filtering of emails and other endless solutions. There are many ways that machine learning can be implemented, and the main focus in this paper is on Object Detection through machine learning.

Image Processing, helps in showing how objects detection can be implemented in a raspberry pi. Hence it will involve the need to manipulate images to a certain extent. Image Processing is an approach to enhance and alter raw images received from cameras or sensors. The method used in this paper is Digital Image Processing.

The rest of the paper is organized as follows: section 2 gives requirement, section 3 throws insight on getting your Raspberry Pi ready, section 4 gives the details about how to train a basic model in desktop, section 5 gives an idea about how to run the application on Raspberry Pi and section 6 gives conclusion.

2. Requirements

There are several requirements that would be needed to implement this solution, listed as follows:

- *Hardware requirements*
  - Raspberry Pi 3: To run your trained model and program to perform object detection. The advantage of using a raspberry pi is that we would not need an entire computer or laptop to execute programs.
  - A camera module/webcam: The camera will be attached to the pi through which the pictures will be sent to the object detection program.
  - A desktop with or without a GPU: There are limitations while using a raspberry pi, one of them being it can't handle training or retraining of models. Hence using a desktop preferably with a GPU is the best to train models. The more powerful your desktop the faster you would be able to train your models.
- *Software Requirements*
  - TensorFlow: It is a free and open source library developed by Google. It can perform a wide range of tasks and can be used for Machine Learning applications.
  - OpenCV: OpenCV that stands for open source computer vision is a free library of programming functions. OpenCV will help us perform image processing.
  - Python: Python is like most languages used today a high level general purpose programming language. Python is suitable as a scripting language because it is directly compiled to byte code and executed. Python is used today because it has allowed us to write clear and logical applications for tasks of any size, this is due to its strong structuring constructs.[4]

- IDE: As for which IDE i.e., Integrated Development Environment one should use, there's no specific one. Anaconda and Visual Studio are two of my favourite IDE's and they will do the job.

- PyPI: The python package index is repository of software for the Python programming languages. PIP is the tool used to install the packages. PIP is a simple and elegant way of installing the dependencies using a single command line.

3. Getting your Raspberry Pi ready

In order to get your raspberry pi ready to run your machine learning applications, first and foremost work is to connect your raspberry pi to a monitor to be able to access the UI. The Pi supports different video, HDMI being the most common and obvious choice in this case. The bare minimum setup would require you to use a mouse and keyboard, both of them can be connected via the USB ports available on the Pi board. You will also need to connect a SD card to use with the Raspberry pi, you will need to flash your SD card, this is a process that involves flashing an operating system on to your SD card. This can be done from Windows, Linux or OS X , Linux being the easiest option out of the three. You can now connect your raspberry pi to the internet , wired or wirelessly. [1][2] Once it is setup all the above mentioned dependencies need to be installed. Firstly install Python since doing so will make your life way easier as then you can install the remaining dependencies using PIP. A suggestion in general would be to always create a virtual environment to work in, these applications require one to install a lot of dependencies and at times we can end up creating a mess of it and then trying to delete the ones that aren't required is a task. Hence by creating a virtual environment your dependencies will exist only within it and not affect your OS outside of it. Once the virtual environment is created install TensorFlow and OpenCV using pip commands. Based on the program and application, pip and the remaining packages are installed. Once installing all the dependencies is done, the raspberry pi is now ready and will be able to run machine learning applications. [3] discussed that Biomedical and anatomical data are made simple to acquire because of progress accomplished in computerizing picture division. More research and work on it has improved more viability to the extent the subject is concerned. A few tech- niques are utilized for therapeutic picture division, for example, Clustering strategies, Thresholding technique, Classifier, Region Growing, Deformable Model, Markov Random Model and so forth. This work has for the most part centered consideration around Clustering techniques, particularly k-implies what's more, fluffy c-implies grouping calculations. These calculations were joined together to concoct another technique called fluffy k-c-implies bunching calculation, which has a superior outco- me as far

as time usage. The calculations have been actualized and tried with Magnetic Resonance Image (MRI) pictures of Human cerebrum. The proposed strategy has expanded effectiveness and lessened emphasis when contrasted with different techniques. The nature of picture is assessed by figu- ring the proficiency as far as number of rounds and the time which the picture takes to make one emphasis. Results have been dissected and recorded. Some different strategies were surveyed and favorable circumstances and hindrances have been expressed as special to each. Terms which need to do with picture division have been characterized nearby with other grouping strategies.

4. Training a basic model in desktop

Once the raspberry pi is ready to run machine learning applications, it needs a trained model to do so and the training cannot be done on a raspberry pi, it's merely a tool to run your programs. Retraining a neural network for your project is the simplest way of implementing a machine learning application and since majority of students would be doing that for better accuracy. To give a small understanding of a neural network its basically trying to mimic the brain, in technical terms," A neural network is an interconnected assembly of simple processing elements, units or nodes, whose functionality is loosely based on the animal neuron. The processing ability of the network is stored in the interunit connection strengths, or weights, obtained by a process of adaptation to, or learning from, a set of training patterns."as defined in the book, "An Introduction to Neural Networks" by Kevin Gurney. My interpretation of a neural network would be trying to recreate a set of artificial neurons that interact with each other and can learn based on different criteria, this put together is a neural network. If you choose to retrain your neural network such as INCEPTION v2 or MOBILENET v2 you need first pick which model you want to retrain since INCEPTION for instance is an extremely heavy model and your desktop might be able to retrain and run the model but once imported to Raspberry Pi it is sure to crash the raspberry pi since a raspberry pi does not have the processing power to run such heavy models. Hence choosing a lighter model like MOBILENET would make more sense if your end goal is to be able to run it on a raspberry pi. Hence to reiterate using retrained CNN is perfectly fine if you can manage to make it light enough to be able to run smoothly on the pi. This is a very important thing to remember since most people manage to create perfectly running machine learning applications on their desktops but when importing the project to a miniature PC like the raspberry pi the program fails to run or takes too long to execute.

5. Running the application on Raspberry Pi

Once you have created a light CNN to run on your raspberry Pi, the last step would be to run the program on the raspberry pi as when it is connected to a power source. Let's take an

example to understand this better, imagine if you were going to be using a CCTV camera as the camera module which is usually connected at a height then it would be impractical to have to connect a monitor along with a keyboard and mouse and then run the program through the command prompt. What should happen is and when the raspberry pi is connected to a camera and an external power source it should immediately run the program. There are several ways to do it is to make changes to the cronfile. The cron is software which is present in Linux operating system which can be used to automate the any task that is scheduled. Crontab is a given list of commands that can be used to execute scheduled tasks at scheduled times. Open crontab -e from the command prompt while in root and @reboot specify the location of the file to be executed. Now whenever the raspberry pi is connected to a power source it will automatically run the program that was scheduled.

6. Conclusion

Firstly, a basic understanding of an application whether it works fine on your PC or laptop it is not guaranteed to work everywhere and understanding this is important since portability is a key factor for the success of any application. If it cannot be implemented everywhere then it is not a successful product. Hence making sure you have tuned out all the tiny mistakes and made the program as flexible as possible so that it worked everywhere is always a must. In regards to this application in particular remember that a raspberry pi as amazing it is, it's performance capabilities are limited. Maybe in the future there might be more powerful mini computers like the raspberry pi but as it stands now we have to make sure we have optimized our model as much as possible to make it run smoothly and quickly on the raspberry pi.

References

[1] Eben Upton and Gareth Halfacree ,"Raspberry Pi User Guide", WILEY Publications, 2012.

[2] Andrew Robinson and Mike Cook ,"Raspberry Pi Project", WILEY Publications, 2013, pp. 9-28.

[3] Christo Ananth, S.Aaron James, Anand Nayyar, S.Benjamin Arul, M.Jenish Dev, "Enhancing Segmentation Approaches from GC-OAAM and MTANN to FUZZY K-C-MEANS", Investigacion Clinica, Volume 59, No. 1, 2018,(129-138).

[4] Dave Kuhlman," A Python Book: Beginning Python, Advanced Python, and Python Exercises", Platypus Global Media,2013, pp 12-13.