**Badminton Service Fault Detector**

**Nvidia Jetson TX1**

***Achintya Dutta***

**Why this project?**

I did some research regarding the Nvidia Jetson TX1 and tried to come up with an original idea for the project. Whichever idea I came up with had either already been done before or somebody had already published a paper regarding it. Finally I came up with this project which is technically a brainchild between the two things I am most passionate about i.e. technology and badminton and I am proud to day that *this idea is completely original and I could not find anything similar anywhere on the internet.*

In sports like cricket, technology is already being used in extensive ways (DRS, spider cam etc.). But in sports like badminton the use of technology is still very limited.

**Problem Being Solved**

**A project is useless if it does not solve an existing problem.**

At the YONEX All England Open Badminton Championships 2018, a new service rule was implemented by the Badminton World Association:

*“The whole of the shuttle shall be below 1.15 metres from the surface of the court at the instant of being hit by the server's racket”*

This created quite a racket in the badminton community worldwide with several top ranked players such as Victor Axelson mocking the rule.

Therefore the rules of serving the shuttle are:

1. The shaft and the racket head of the server's racket at the instant of hitting the shuttle shall be pointing in a downward direction.
2. The whole of the shuttle shall be below 1.15 metres from the surface of the court at the instant of being hit by the server's racket.

These are essentially the two major rules which regularly create a lot of controversy at premier tournaments and top players regularly lodge complains against service judges.

My project using the Nvidia Jetson TX1 aims at keeping precisely these two conditions in check so that human error doesn’t influence the decision.

**What I Learnt About the Nvidia Jetson TX1**

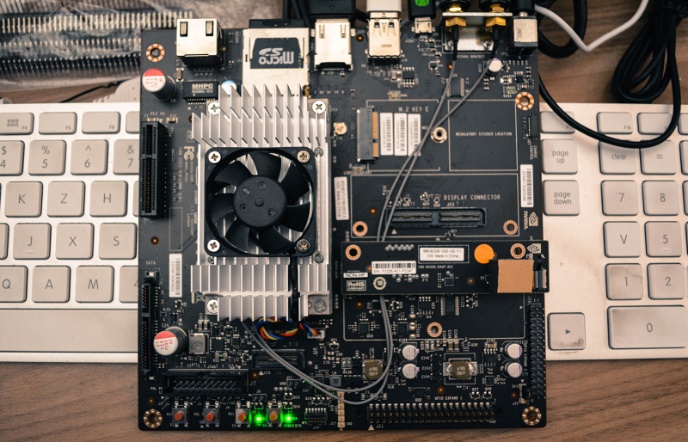
{I am not including the technical specifications as they can be looked up in its datasheet}

The Nvidia Jetson TX1 is essentially a developer kit which is basically a module constructed around a GPU which basically gives us higher computational power required for computer vision applications, deep learning and neural networking, machine learning algorithms and the specific one which I am using in my project i.e. Image Classification using computer vision.

*Difference between CPU and GPU*

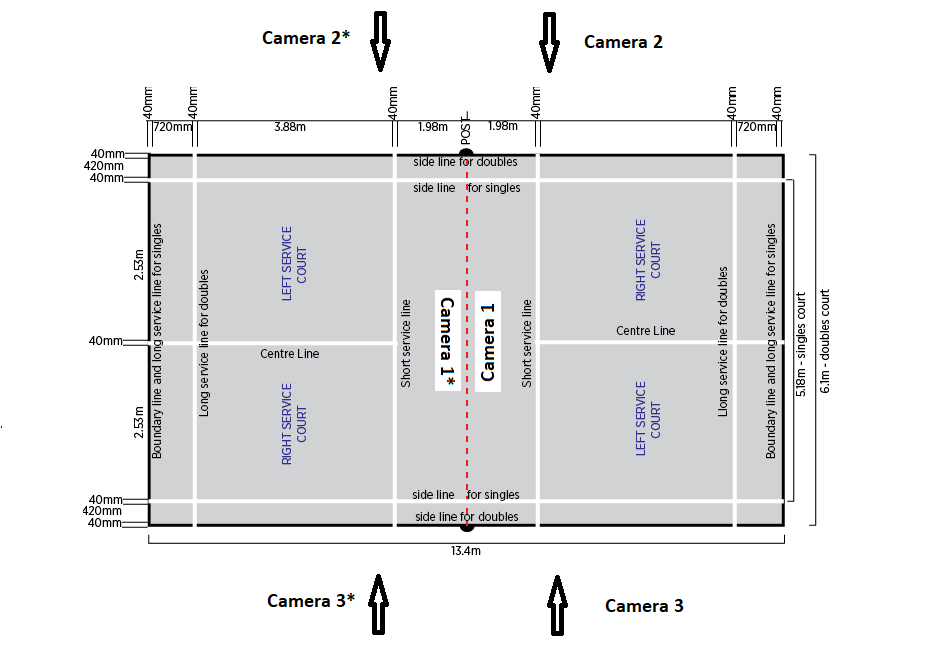
GPU stands for Graphics Processing Unit and its speciality is that it is exceptionally fast and efficient at certain tasks such as repetitive matrix multiplications etc. and hence makes more raw computational power available to the programmer and hence is used in areas such as 3D graphics, computer vision etc. as any picture can be expressed in pixels which are effectively matrices with three digit value elements representing the RBG or Red, Blue and Green values.

The Nvidia Jetson TX1 is a very powerful module and includes an Omnivision OV5693 5 MP camera module along with a display expansion header, DC power connector, antenna connector, USB2 micro, USB3 A, HDMI 2.0,SD card socket, Ethernet, power reset force recover and one user definable button, UART etc.



*The main feature of the Nvidia Jetson TX1 which I will be using is Computer Vision or object detection and processing to be precise i.e. I will be using it to detect the racket head and check its angle and height from the floor along with detecting the shuttle and checking its height from the floor.*

**Technical Design**



We would require a total of three cameras on each side as we have to cover all angles. All the cameras are to be placed at a height of 1.15 meters. The function of all the three cameras are defined in the next page. Two of them are essentially used in order to find the precise time of contact and find the distance of the shuttle from the ground as these cameras give the side view. The other camera is used to check the angle of the racket shaft in order to make sure that the racket head is pointing downwards.

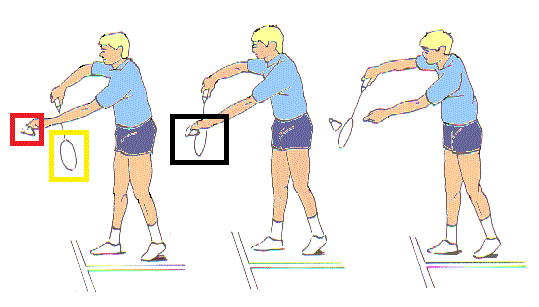
It has been explained in detail in the following pages.

Once all the computations regarding the rules have been done and the processor has been able to come to a conclusion whether it is a fault or not, we can simply attach an Arduino UNO through the USB port of the Nvidia Jetson TX1 and write a simple code for giving the output stream as fault or not interfering or simply blinking the red light if a fault is committed and not showing any light if it is a legal service.

**Functions of the Three Cameras**

**Camera 2**

This is the camera which gives us the side angle. This will help us determine the exact time‘t’ when the racket head is making contact with the shuttle.



This camera will first determine the racket head and shuttle separately using image classification and then store the time ‘t’ exactly when both the objects coincide giving us the time of contact. Moreover the feed from this camera is used to check whether the height of the shuttle is above or below 1.15 meters (explained ahead).

**Camera 3**

This camera is identical to camera 2 but is required in case the player serving is a left hander or for some reason the view of camera 2 is blocked. Moreover the instant of contact can be verified between the two cameras in case one of the two is unable to identity both the racket head and the shuttle.

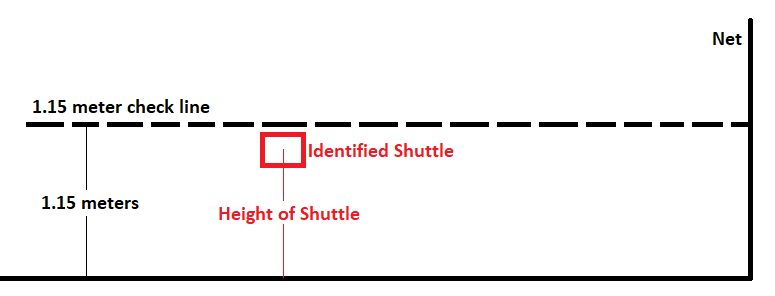
**Camera 1**

This camera is placed on the net. This is one of the advantages of having a camera instead of a human being. Humans have to sit outside the court and their decision might get affected by the angles of viewing. Whereas a small camera can be placed so that it has a straight line of sight. Once we have time ‘t’ i.e. as soon as we get the signal from camera 2 and camera 3 that there has been contact, it is at that moment that camera 1 has to capture the image.

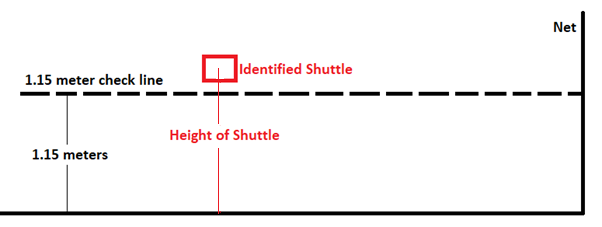
That moment would define the point of contact and it is at this moment that all the rules have to be verified.

***Checking Whether Shuttle is Below 1.15 meters***

This will be checked using the feed from camera 2 and camera 3. Both the cameras are placed at a height of exactly 1.15 meters. Therefore the central pixels of its feed represents 1.15 meters. Hence it will simply have to check if the shuttle which it has already identified, is completely below the central pixels of the picture matrix or not. If not it will give a signal which would call a fault.



LEGAL SERVICE

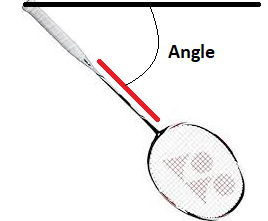


FAULT SERVICE

***Checking If Racket Head is Pointing in Downward Direction***

This is done using camera 1 as we would require the front view of the racket to check its angle.

The picture is clicked or frozen from the front view as soon as the signal is received from camera 1 or camera 2 that there has been contact. As soon as there is contact, we can check the angle of the identified racket by checking the shaft angle with a horizontal line. If the angle with the horizontal line is negative i.e. below the line, it is a legal service whereas if the angle with the horizontal line is positive, it is a fault service.

 Angle is Negative hence legal

If the racket head ends up making a positive angle even if it is very small, a fault will be called.

**………………………………………………………………………………………………………………………………………**

**Challenges of the Project**

* I studied the basics of computer vision but I am nowhere near good enough to write the algorithm for the above image classification tasks BUT I have done enough research to find examples where it has been used in different applications hence I can vouch that all the tasks mentioned above are possible and are not hypothetical.
* Occlusion i.e. if the racket/shuttle colour becomes same as its background colour, the system might have trouble detecting them.
* A player may intentionally hide the shuttle from the camera by finding its blind spots as every camera has a specific angle.
* A player might have a unique serving action which might cause unprecedented problems with viewing angles.
* A racket shaft is very thin hence we would require a very strong vision algorithm and a good camera to detect it with all the background noise.