Summer is over and it is time to get back to work.

The world leaders in robotics, drones, artificial intelligence and LED technologies are the industries in China. I just received an email from one of my suppliers. VEX is a Texas based company that makes robotics for the educational market. They announced that the price of all their products will be going up 10-15 percent this month. I suspect this is a result of the trade war, increased tariffs, between China and the USA. This will be hard for schools to adjust to. The other major supplier of robotics to schools is Makeblock, also from China.

Note: At present none of my hardware nor software comes from the USA. The latest hardware is from the EU and China. All the software is open source and available to anyone in the world with a computer.

Over the summer I have been reading about computer vision and it is a lot of mathematics. Think of a picture/image as a matrix of dots/pixels, like they advertise for TVs. A standard HD television/camera has 1920x1080 pixels. This means 2,073,600 dots. If the image is in colour, then each dot/pixel must also describe the colour for that dot. If we use the standard Red/Green/Blue (RGB) format and 8-bit depth (value between 0-255), then each dot/pixel will have three values to represent its RGB value.

The RGB values for red are 255,0,0: yellow is 255,255,0; and gold is 255,215,0. White is 255,255,255 and black is 0,0,0.

Therefore to compare two HD images the computer has to check 2,073,600 x 3 = 6,220,800 (approx 6 million) values. In computer vision software, an image is called a multi dimensional array. Think of a grid with width, height and depth.

Have you ever gone into a room where all the walls, floor and ceiling are the same colour? What does your brain do? It looks for anywhere where there is a difference. This is called edge recognition. We need a reference point. This is also how we recognize motion. Our brain identifies an edge mark as an item moves across our eyesight.

So if your computer program is designed for facial recognition, here are the initial steps:
- take a background/base shot
- take periodic shots and look for edge recognition (motion)

All humans have a general shape which includes shoulders, neck and head. All heads have a similar shape and spacing for eyes, nose and mouth.

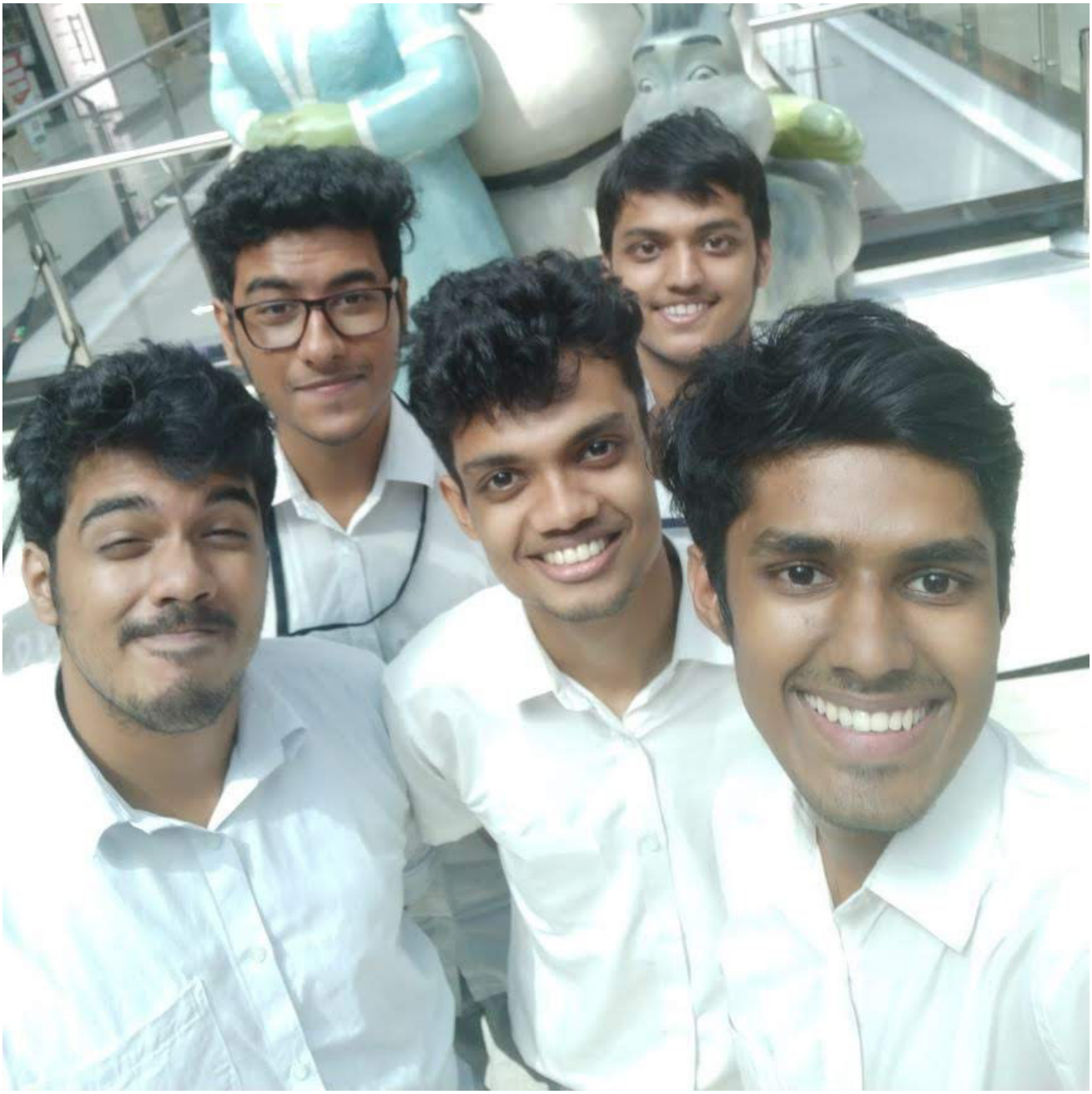
So if your program is designed for facial recognition, here are the next steps:
- from the edge, determine width and then crop (cut) the image
- compare to standard image of human
- if yes, crop the image to isolate the head

There are freely available databases of faces. IBM offers one called Flikr diversity in faces which contains almost two million faces and metadata (head/nose/etc length, gender, eye colour, etc). There is software that will review facebook profiles and build a database with the picture, name and other attributes.

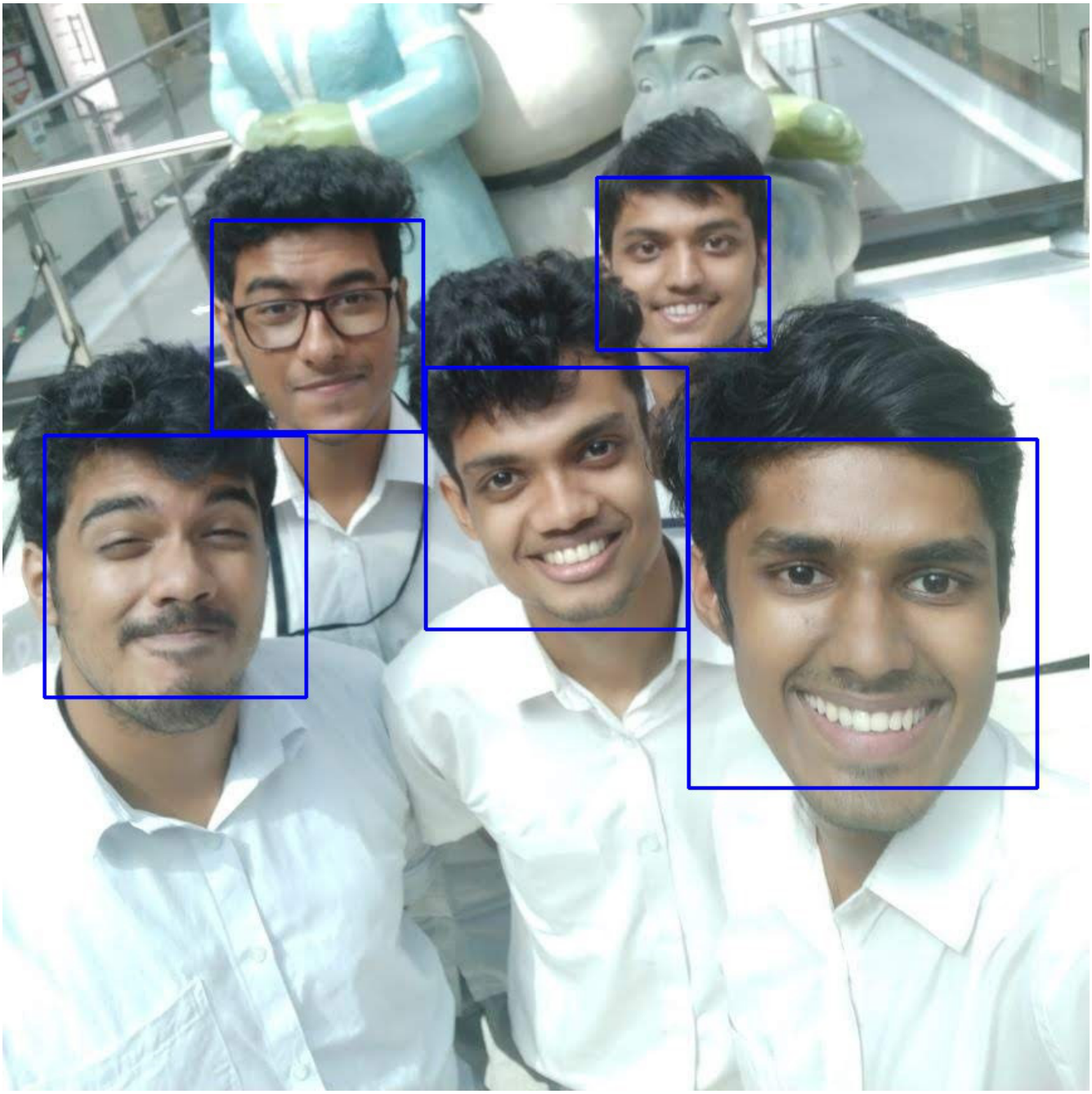
So if your program is designed for facial recognition, here are the final steps:
- calculate the metadata
- compare metadata and image to database

As you can expect the cost of the hardware must be really high. In my case the total cost was $75. The software was free. Since there is a world wide community of people working on the high level of mathematics required and developing this software, the hard part was the two weeks for the research to find the correct interlocking pieces of software. In the end, my program, to interface with all the different software modules, to detect the faces in an average picture was, wait for it, 26 lines long.

Here is my test input picture:



Here is the result of running the program, which took five seconds:



The program stores the co-ordinates of each rectangle, so that each face can to stored in a separate file or database entry. I am now modifying the program so that it will work with lower resolution pictures, such as from newsprint.