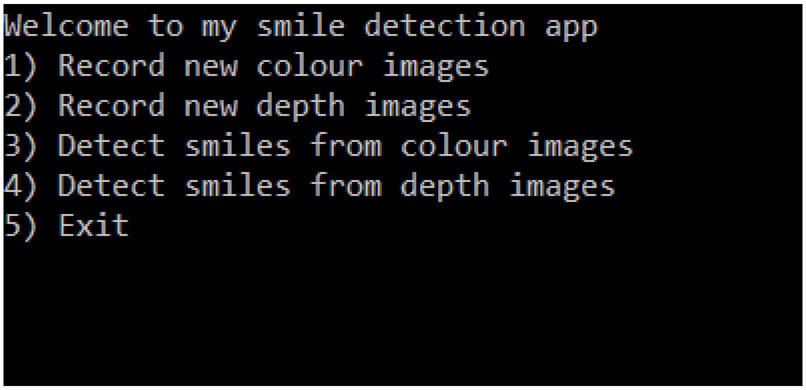
Unknown date | **Unknown author**

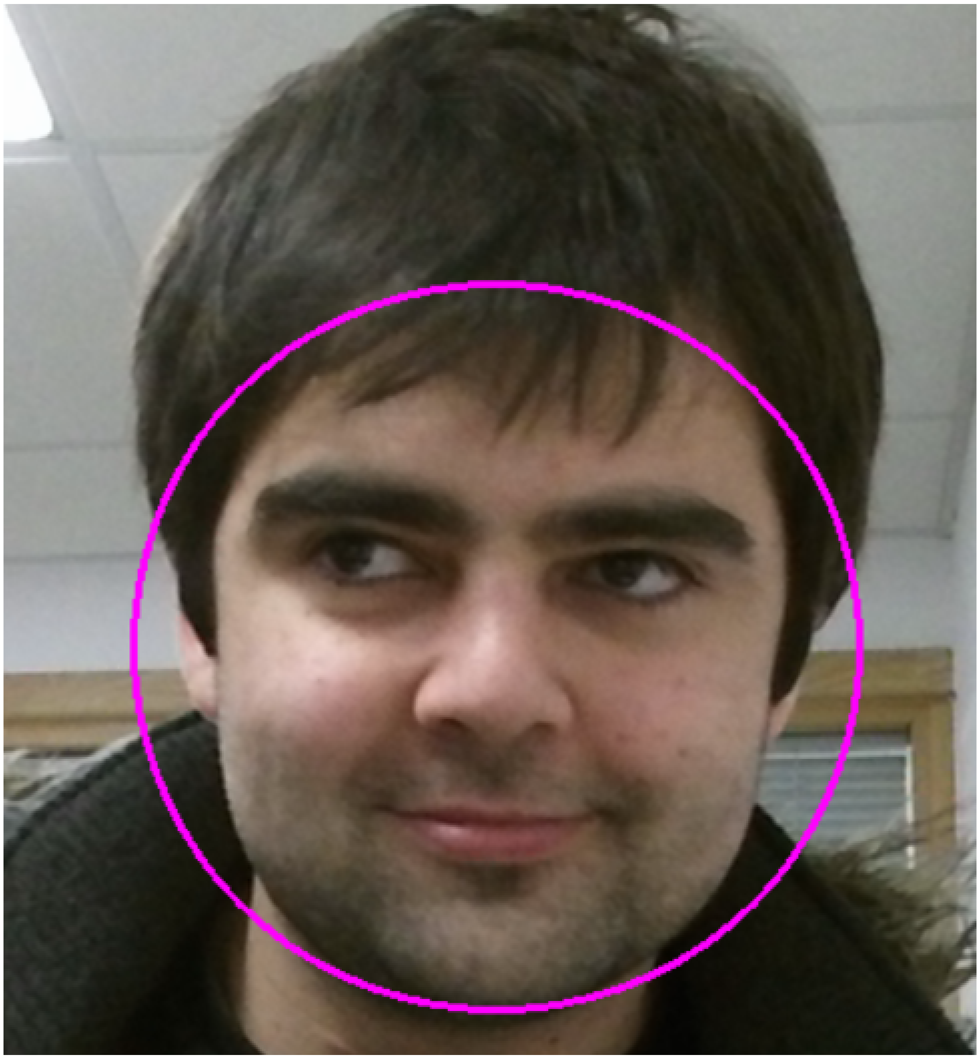
Share this:

Recently, I have finished a project based on detecting smiles through the Kinect 2.0 depth and colour data.

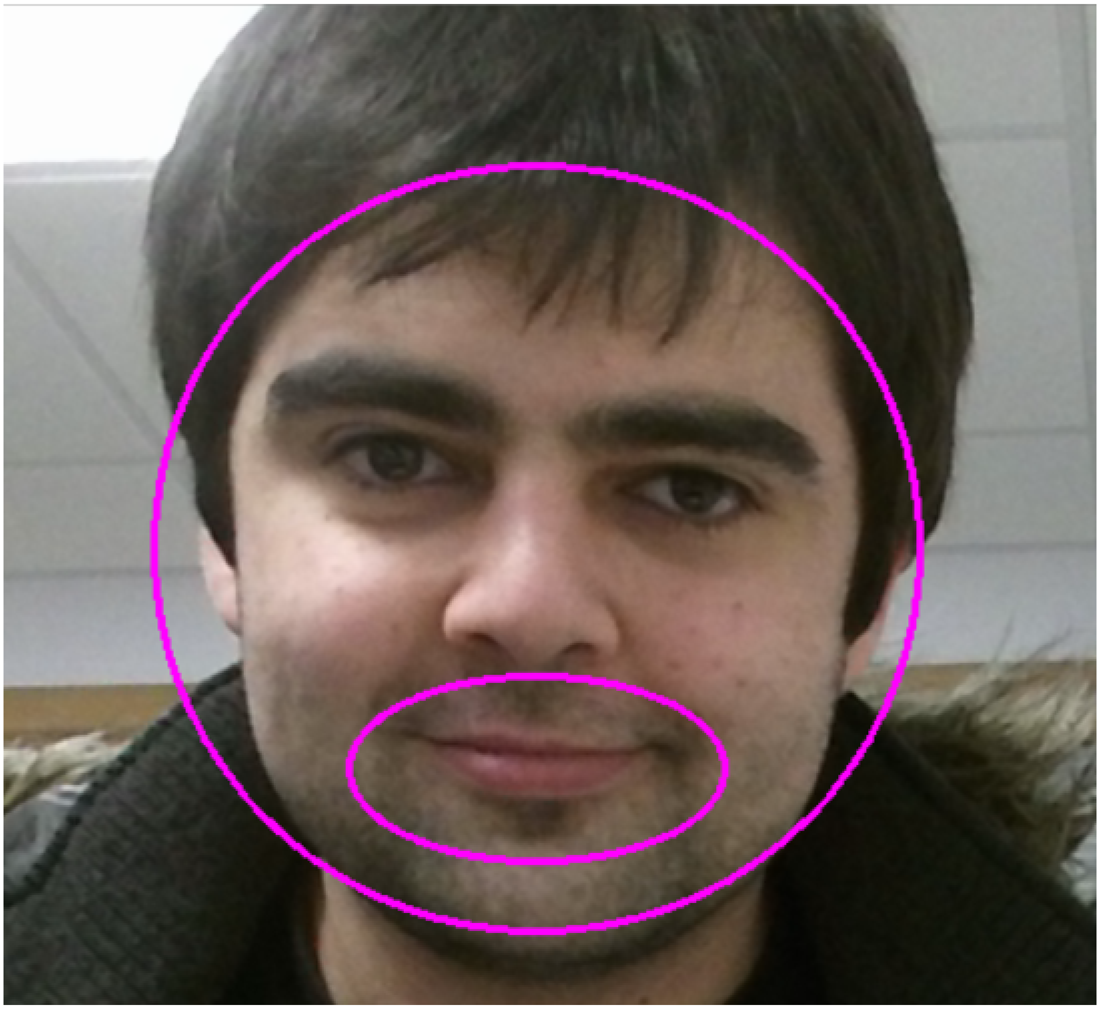


```
Welcome to my smile detection app
1) Record new colour images
2) Record new depth images
3) Detect smiles from colour images
4) Detect smiles from depth images
5) Exit
```

With this program, the user is able to plug in the Kinect 2 and record colour and depth images directly from it. My program is then able to apply OpenCV object detection (using detectMultiScale and the face classifier) to detect faces within each frame (as the less than complimentary picture shows below).



Once the face is found in a frame, the program will then focus in on the bottom half of the face and search in that region for a smile (using the provided smile haar classifier). The first attempt I made at this produced lots of false positives but after refining the criteria for a smile slightly (specifying how left/right it could be located in respect to the face), I managed to improve the accuracy significantly.



This is most of the work done. However it should be mentioned that creating code to extract the relevant data from the Kinect 2 and to convert it to the necessary format for OpenCV proved much trickier than expected. Moreover, getting Visual Studio to include all the relevant libraries and directories from both OpenCV and the Kinect was also a significant hurdle. E.g. As OpenCV 3 didn't actually have a released build for Visual Studio 2015, I had to use CMake to build all the files especially.

Having managed to implement face detection in python in the past with ease, the difficulty of applying the same function with C++ was surprisingly difficult (yet satisfying once it finally worked!).

If I had the time with this project, I was aiming to create my own haar classifier for the smile based on the positives and negatives recorded in the program and to, in general, allow the program to let you train a haar classifier in real time.

I'm looking forward to getting involved with similar projects in the near future, most likely involving Behaviour Recognition and dealing with various other forms of biometrics.

Viewed using Just Read