Dynamic Analysis of Facial Expressions using KINECT

Semester Project Report

Królikowski Paweł,
29.06.2011

1. Introduction

Face is one of the most acceptable biometrics, because it is one of the most common methods of identification that humans use in their visual interactions and acquisition of faces is non-intrusive. It is very challenging to develop face recognition techniques which can tolerate the effects of aging, facial expressions, slight variations in the imaging environment and variations in the pose of face with respect to camera.
In the 2D image domain, the effects of the expressions have a decisive influence on the performance. Hence, the utilization of 3D shape data becomes more and more popular in security applications.

Kinect can be viewed as a breakthrough in 3D data acquisition. It's a is a horizontal bar connected to a small base with a motorized pivot and is designed to be positioned lengthwise above or below the video display. The device features an "RGB camera, depth sensor and multi-array microphone running proprietary software", which provide full-body 3D motion capture, facial recognition and voice recognition capabilities.

One of the most interesting properties of Kinect is it's price - new sensor's price is around 150$, making the sensor extremely cheap compared to other solutions providing similar capabilities. In fact, it is the first device with 3D sensor available for average customer. This fact makes it interesting from developer/researcher point of view - any application developed on/for Kinect becomes immediately available for wide groups of people, instead of small number of people who possessing advanced 3d cameras.

2. Detailed introduction to Kinect

Described by Microsoft personnel as the primary innovation of Kinect, the software technology enables advanced gesture recognition, facial recognition and voice recognition. According to information supplied to retailers, Kinect is capable of simultaneously tracking up to six people, including two active players for motion analysis with a feature extraction of 20 joints per player.

However, from project's point of view only 2 features of Kinect are important: RGB & Depths cameras:

Kinect sensor outputs video at a frame rate of 30 Hz. The RGB video stream uses 8-bit VGA resolution (640 × 480 pixels) while the monochrome depth sensing video stream is in VGA resolution (640 × 480 pixels) with 11-bit depth, which provides 2,048 levels of sensitivity. The sensor has an angular field of view of 57°horizontally and 43° vertically which will constitute to one of the problems listed in following sections.

The depth sensor consists of an infrared laser projector combined with a monochrome CMOS sensor, which captures video data in 3D under any ambient light conditions. The sensing range of the depth sensor is adjustable, and the Kinect software is capable of automatically calibrating the sensor based on gameplay and the user's physical environment.

As a trivia, I can add that fastest selling consumer electronics device in the world, with 8 milion of devices sold over 60 days, with 133k/day average.

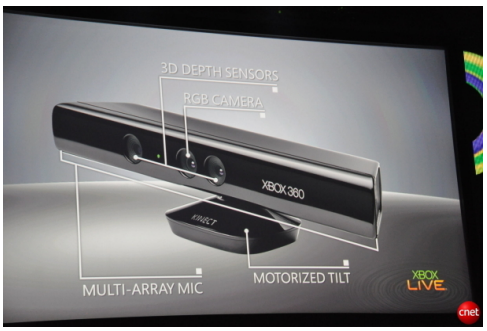

Figure.1. Kinect (source cnet.com)

3. **Drives issue**

Few dates:

- Kinect launched on November 4, 2010.
- On November 10, first Linux driver that allows the use of both the RGB camera and depth sensitivity functions of the device was developed.
- In December 2010, PrimeSense, whose depth sensing reference design Kinect is based on, released their own open source drivers along with motion tracking middleware called *NITE*.
- On June 16, 2011 Microsoft released a non-commercial Kinect software development kit for Windows. A commercial version is planned for a later release date.

Between 10 November and 16th June several different Open Source drivers were developed (OpenNI, LibfreeKinect, openKinect, etc.). None of them was officially supported by Microsoft, all of them delivered slightly different functionalities. At the moment of start of the project one of the best available drivers was chosen (OpenNI), but it has to be noted that results provided in this report should be considered as a results of software written using specific drivers & SDK. It's possible that some other drivers were, are or would result in different results, especially taking into consideration the speed of the development of Kinect connected software - since the beginning of the project two new version's of OpenNI were released, (with one scheduled in middle of July 2011) + Microsoft released it's own beta-SDK.

As of this moment it's unclear whether release of Microsoft SDK will results in changes in open source implementation of drivers. Attempts are being made to incorporate functions of both environments, but it's hard to predict future of Kinect's software.

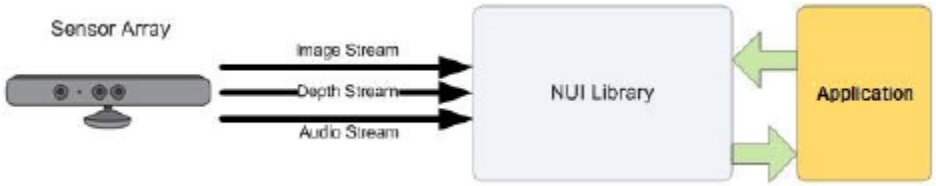

Figure 2. Hardware and software interaction with application
(source Microsoft Programming Guide for Kinect SDK)

4. Problem Statement / Objective

The goal of the project was to test possibility of using Kinect as a tool for dynamic analysis of facial expression in 3d. Because of the early stage of Kinect development and time constraints it was decided to track specific points on the face using specially prepared markers. After that, goal was to get 3D coordinate of each tracked point using depth camera.

5. Related work

Since the release of first open source driver community has made many attempts and applications utilizing Kinect sensor, such as controlling robots, creating games, tracking people, Minority Report style control glove, monitoring aged people, etc. However, to my knowledge no one has attempted and published results of successfully working facial expression recognition/analysis system.

6. Design & Implementation (could be separated)

a. Tools used

Whole application was developed in C++, using OpenNI library with modified "Avin's" PrimeSense Kinect drivers. OpenCV library was used for various image enhancements and display of the results.

b. Description of the idea

Kinect's output is a RGB map with 24 bits for each pixel, in 640x480 resolution, up to 30 FPS and Depth Map of 11 bits for each pixel with same resolution and FPS number. First step was aligning/calibrating both cameras.
After successful calibration it is possible to map any RGB pixel onto Depth Map and therefore getting it's depth -> distance from the sensor. With that data we're able to get points 3D coordinate.

After some research in available research articles several high-expression facial points were chosen. Each of them was marked with marker of specific colour. Application should be able to track them in real time, displaying coordinates in real time.

7. Issues

During development of the project several issues connected with Kinect have surface. I'll describe them in following sections.

First of them, and perhaps the most important was relative youth of Kinect itself, as well as it's Open Source drivers. Setting up the environment & developing with buggy, not well documented code was harsh at times. On the other hand, development process of whole field is very fast - lots of talented developers and companies are working on it and it can be assumed that in near future Kinect will get much more mature.

One of the encountered problems was calibration and alignment of both of the sensors. Both hardware and software of Kinect were supposed to support automated calibration. Unfortunately, this feature didn't work, requiring manual calibration.

Noise of RGB channel was really big. Up to 25% of the value of each colour can change at any time without reason -> tracking specific colour in RGB proved to be difficult. Also, the artifact of "moving stripes" on RGB stream appeared - horizontal areas of the picture were brighter than it's neighbours.

FPS manual specifies it can support up to 30 FPS. Up to seems to be key word, because I did not manage to get it over 22-23 FPS. Again, this might be a bad driver or my hardware issue, but I was unable to verify it.

Automatic brightness adjustment - Kinect hardware automatically adjusts brightness depending of number of objects and brightness of the scene. While this feature might be a good idea in XBOX gaming, seemingly random brightness changes were not desired in my project. At this point, none of the drivers offers feature of turning it off.

Field of view - the original goal of the Kinect was to enable controlling XBOX console with movement of whole body. Because of that, it was supposed to "see" as much of the scene as possible. Convex lens made it possible, but while it was good for gaming purposes, it's not best solution for facial recognition. Additionally, Depth sensor has a minimum distance of ~80 cms. This features makes it impossible to get high resolution of face image - f.e. maximum width of the face I was able to get was around 110 pixels, which constitutes to less than ¼ of Kinect's horizontal resolution.

Resolution & Quality of Depth Map - Depth map obtained by Kinect is not perfect. The properties of infrared light sensor make it sensitive to light condition changes & creates

a “shadow” behind scanned object - in some areas of the picture we cannot get the depth, as infrared light was stopped by some obstacle on the way. Additionally, although theoretical resolution of Depth should be around 1.5 mm at the distance of 80 cm, the real one is around 3-4.

8. Results & conclusion

The developed application was able to successfully obtain both RGB and Depth stream from Kinect. Both streams were converted to OpenCV data format & agligned/scaled. Application was theoretically able to detect any number of markers of required colour & tracks their position in real time. However, due to RGB camera constrains the required size of the markers was about 8x8 mm, which is too big for detailed expression analysis. Additionally, the results varied depending on the environment/lighting conditions on the scene. Under perfect lighting conditions it was able to successfully track up to 12 markers of the size 4x4 mm, however under average conditions & final "guaranteed" results was 6 markers of size ~8x8 mm.

The Kinect sensor indeed proved to be interesting and beneficial tool for 3D analysis, but my conclusion would be that at this stage it's not particularly useful for detailed face analysis, either static or dynamic. While features like gesture analysis or joint tracking, which are natively supported by Kinect do not require very high resolution, which makes them perfect for this sensor, face expressions need much more precision which is not (yet?) provided by Kinect. It might work for basic&clearly visible expressions like openning/closing mouth, nodding head, but these expression are easily recognizable using RGB stream solely, which make idea of using Kinect for it doubtful. On the other side, project proved that possibilities of Kinect are enormous, but most probably in other field applications.