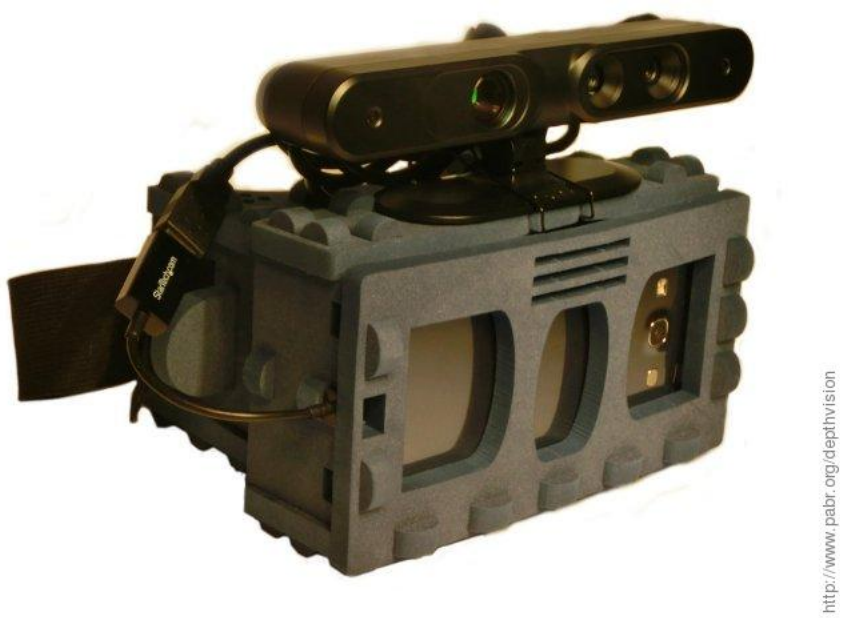
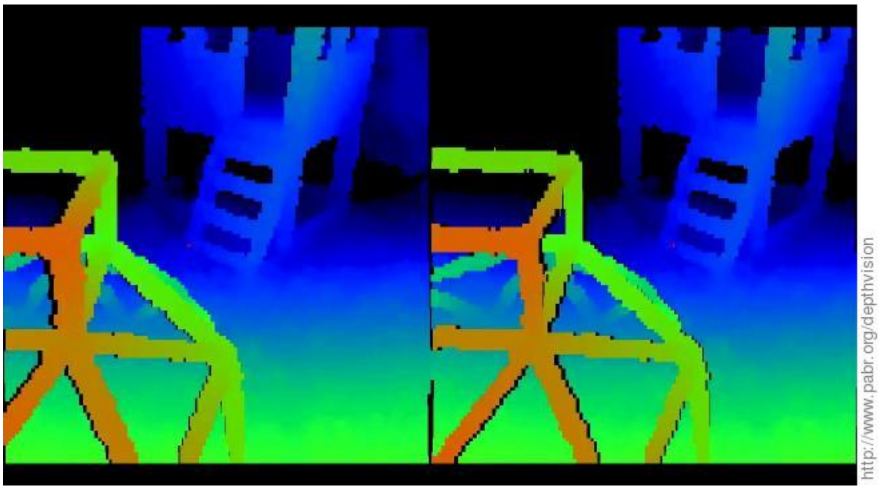
Seeing the world through a depth-sensing camera

In this project we attach a depth-sensing camera to a head-mounted stereoscopic display (think: Kinect + Google Cardboard). The resulting portable device allows users to see their surroundings in unusual ways.





READ THE HYPERTEXT VERSION HERE:
http://www.pabr.org/depthvision/depthvision.en.html

Revision History

1.0	2014-11-10	Initial release.

Table of Contents

1. Introduction

Is virtual reality finally coming of age, after decades of false starts and unmet expectations ? It is too early to tell, but there has certainly been exciting progress since we last investigated this topic in 2009 with [WXHMD]: Kinect, Google Glass, Oculus Rift, Google Cardboard, etc.

In this project we combine two technologies which have recently become affordable: depth-sensing cameras, and smartphone-based head-mounted displays. A Kinect-like depth-sensing camera is connected to the USB OTG port of an Android phone. Depth information is rendered stereoscopically with low latency. Possible rendering modes include:

- False-color depth map
- Magnified perspective (e.g. as if having eyes 50 centimeters apart)
- Third-person view (rendering from arbitrary viewpoints)
- Radar- or sonar-like overhead perspective.

Others have already used head-mounted cameras and displays, tethered to powerful computers, for purposes such as 3D modelling and motion tracking. Here we focus instead on low cost, portability and freedom of movement to provide an entertaining user experience. This is "alternate" reality rather than "virtual" or "augmented" reality.

2. Hardware

2.1. Tested configuration

This project begins with the realization that smartphones are now powerful enough to act as USB hosts for peripherals designed for desktop PCs, and simultaneously render real-time 3D graphics. Our implementation consists of:

- ASUS Xtion Pro Live: A depth-sensing camera very similar to the original Kinect for Xbox 360 (PrimeSense IR pattern projection technology).
- Samsung Galaxy S3: A relatively ancient (early 2012) Android smartphone with 1280x720 AMOLED display and USB OTG 2.0.
- Micro-B USB OTG cable: The ID pin in this cable tells the smartphone to act as a USB host rather than as a USB device.
- VR-Spektiv XL: A variant of Google Cardboard with adjustable lens distance and foam construction.

2.2. Alternatives

2.2.1. Depth-sensing camera

We selected the ASUS Xtion (instead of the Kinect) for its small size and plain USB connector. It is quite surprising that the Galaxy S3 is able to power it, as USB OTG is typically intended for low-power devices such as USB flash drives. The Xtion draws 300 mA in our application (streaming depth data at 160x120x30fps, no audio processing, no RGB video streaming).

Note: There are reports that the ASUS Xtion does not work well with USB 3.0, and that this can be worked around on Linux PCs by downgrading ports to USB 2.0 (ehci). Unfortunately this is not so easy to do on Android devices.

The Kinect for Xbox 360 is obviously the most natural alternative. Second-hand units should become massively available in the next few years as newer gaming consoles reach the market. Note that it has a proprietary connector which is essentially USB plus a 12 V input. When purchased separately as an accessory for the original Xbox 360, it reportedly comes with an external power supply which also serves as a splitter cable with a regular USB plug. Maximum current from the USB VBUS is 100 mA according to the USB descriptor. Therefore it should be possible to connect a Kinect to a low-power USB OTG 2.0 port.

Of course the Kinect v2 (for Xbox One) is an exciting alternative, but it requires a USB 3.0 port. The receptacle on the back of the sensor bar reportedly accepts a USB 3.0 Standard-B plug, but then an external 12 V supply must still be brought to soldering pads inside.

There are also a number of short-range depth camera which are intended primarily for gesture recognition. Note that unlike the Kinect, some of these cameras do not perform the stereo correlation work in hardware.

2.2.2. External power supply

To increase battery life, the camera can be powered externally through a USB Accessory Charging Adapter (ACA).

Ideally the ACA should also power the Android device. This cable we tested connects all VBUS lines together (micro-B plug for USB OTG host, Type-A receptacle for USB device, Type-A plug for power supply) but the Galaxy S3 fails to charge from it. The ID pin apparently does not have the magic resistor which triggers ACA functionality.

2.2.3. Android device

The Galaxy S3 performs well, but individual pixels are clearly visible at 1280x720 and the relatively small screen restricts the field of view.

Assuming an unlimited budget, we would use a Samsung Galaxy Note 4 (2560x1440 AMOLED display, 515 pixels per inch, USB 3.0). Its screen width is exactly twice the average human IPD; this yields a symmetrical field of view with Cardboard-like optics.

The Samsung Galaxy Note 3 also has an ideally sized screen, but with slightly lower resolution (1920x1080, 386 pixels per inch).

The Google Nexus 6 will have a slightly larger screen.

Of all these devices, only the Galaxy Note 3 has a USB 3.0 port (presumably required for Kinect v2).

Note that our proof-of-concept software requires a rooted device for direct access to USB peripherals.

2.2.4. Stereoscopic head mount

Google Cardboard brought smartphone-based head-mounted stereoscopic displays to the mainstream, but there are many alternatives, some of which predate Cardboard.

The main criteria are:

- Size (should match the size of the smartphone for optimal positioning)
- Glass vs resin lenses (glass is more durable)
- Biconvex vs plano-convex lenses (biconvex lenses are reportedly better)
- Focal length (shorter FL means wider field of view)
- Lens diameter (small lenses restrict the field of view)
- Lens mount (a conical piece can bring the lenses closer to the eye than a flat plate)

- Adjustable lens distance (should match the user's interpupillary distance)
- 3-point strap vs 2-point strap (an extra strap over the top of the head is recommended).

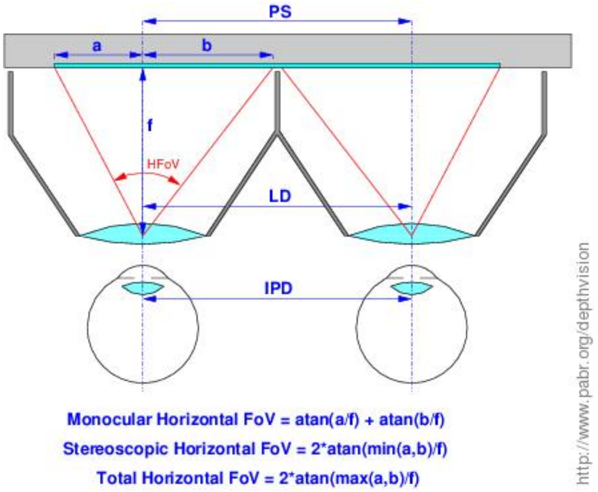
3. A note on HMD optics

It is easy to ruin a user's first VR experience with an improperly configured head-mounted display. Thus, it is a bit shocking that so many side-by-side stereoscopic software applications do not allow some crucial parameters to be adjusted.

At the very least, the following distances should match:

- Interpupillary distance (IPD): The distance between the centers of the pupils of the user (around 63 mm). The IPD is typically measured with the user looking at a faraway object; it might be appropriate to use a slightly lower value when rendering scenery that is very close.
- Lens distance (LD): The distance between the centers of the lenses.
- Projection separation (PS): The physical distance on the screen between the optical centers of the left and right 3D projections. As an approximation, this is the distance between the left and right renderings of a faraway object in the 3D scene. Unless the software can automatically determine the pixel density of the display, this must be adjusted manually.

  Many applications naively center the 3D projections in the middle of each half-screen. This will only yield proper separation if the screen width is exactly twice the user's IPD. For an IPD of 63 mm this implies a 5.7" 16:9 display. On a smaller screen, the perspective will be incorrect. On a larger screen, the excessive separation will cause eye strain.


PS
a
b
f
HFoV
LD
IPD
Monocular Horizontal FoV = atan(a/f) + atan(b/f)
Stereoscopic Horizontal FoV = 2*atan(min(a,b)/f)
Total Horizontal FoV = 2*atan(max(a,b)/f)
http://www.pabr.org/depthvision

Further, the focal length of the 3D projection should match that of the head mount, especially if motion tracking is used. This boils down to adjusting a 2D software scaling factor.

Finally, geometric distortion near the edges of the field of view should be corrected in software.

4. Software

The ASUS Xtion is supported by OpenNI2 [http://structure.io/openni], which can be easily cross-compiled for Android.

This quirk-and-dirty patch [depthvision-poc-1.patch] adds a few lines of C++ code to `OpenNI2/Samples/SimpleRead/main.cpp` so that the **SimpleRead** demo renders stereoscopic point clouds directly to `/dev/graphics/fb0`. It is only intended as a proof-of-concept and a way for others to get started. It is known to work on the Galaxy S3 with Android 4.3 (rooted).

```
$ export NDK_ROOT=/path/to/android-ndk-r9
$ cd OpenNI2/Packaging
$ ./ReleaseVersion.py android
$ ls AndroidBuild/obj/local/armeabi-v7a/
$ adb push AndroidBuild/obj/local/armeabi-v7a/ ...
```

5. Android app

Ideally this project should be converted into a regular Android app for non-rooted devices. Preliminary analysis suggests that it might be possible.

Bibliography

[WXHMD] *WXHMD - A Wireless Head-Mounted Display with embedded Linux* . http://www.pabr.org/wxhmd/doc/wxhmd.en.html .