

914 PC Bot Robot Head part 1

-c6jones720

I hit about the idea of making a robotic head with pan/tilt control and both binaural hearing and stereoscopic vision.

I had already tried programming stereoscopic vision systems in C++ using webcams and the Windows VFW API. I came across a problem that you couldn't connect two webcams to the same computer and have it access them simultaneously. Windows would only allow access to one device at a time and the switching between cameras took several seconds!

I decided to ditch USB webcams and instead use a TV tuner card connected to two colour analogue video cameras that I would time-multiplex. That way I would be in control of the switching and not the OS.

Now switching between analogue video signals is easy, but there is an issue that if you don't synchronise both video signals together (genlock) you lose video frames. Since all I had to hand were a pair of CMOS multiplexer chips and a few Picaxes it wasn't feasible to build a proper genlock circuit basically because the Picaxe processors simply would not have been fast enough.



So I went to the extreme of purchasing a Quad video multiplexer from Maplin's, it was a bargain since it only cost £9.97.

Now I wanted stereoscopic vision so my robot could perceive distance from the disparity between two cameras.



The computer tank on the 914 is too small to fit PCI cards into so the obvious choice of video input device would be an external USB device like the Hauppauge WinTV USB. I got an enclosure and wired it all up.



The nice thing is that the multiplexer and cameras use +12v which means you can power them easily from the 914 computer supply. I have capacity here for another two video cameras.

Here is the face of my robot:



Its got the two cameras, a light sensor, laser and an infrared transceiver for decoding signals from a remote control.

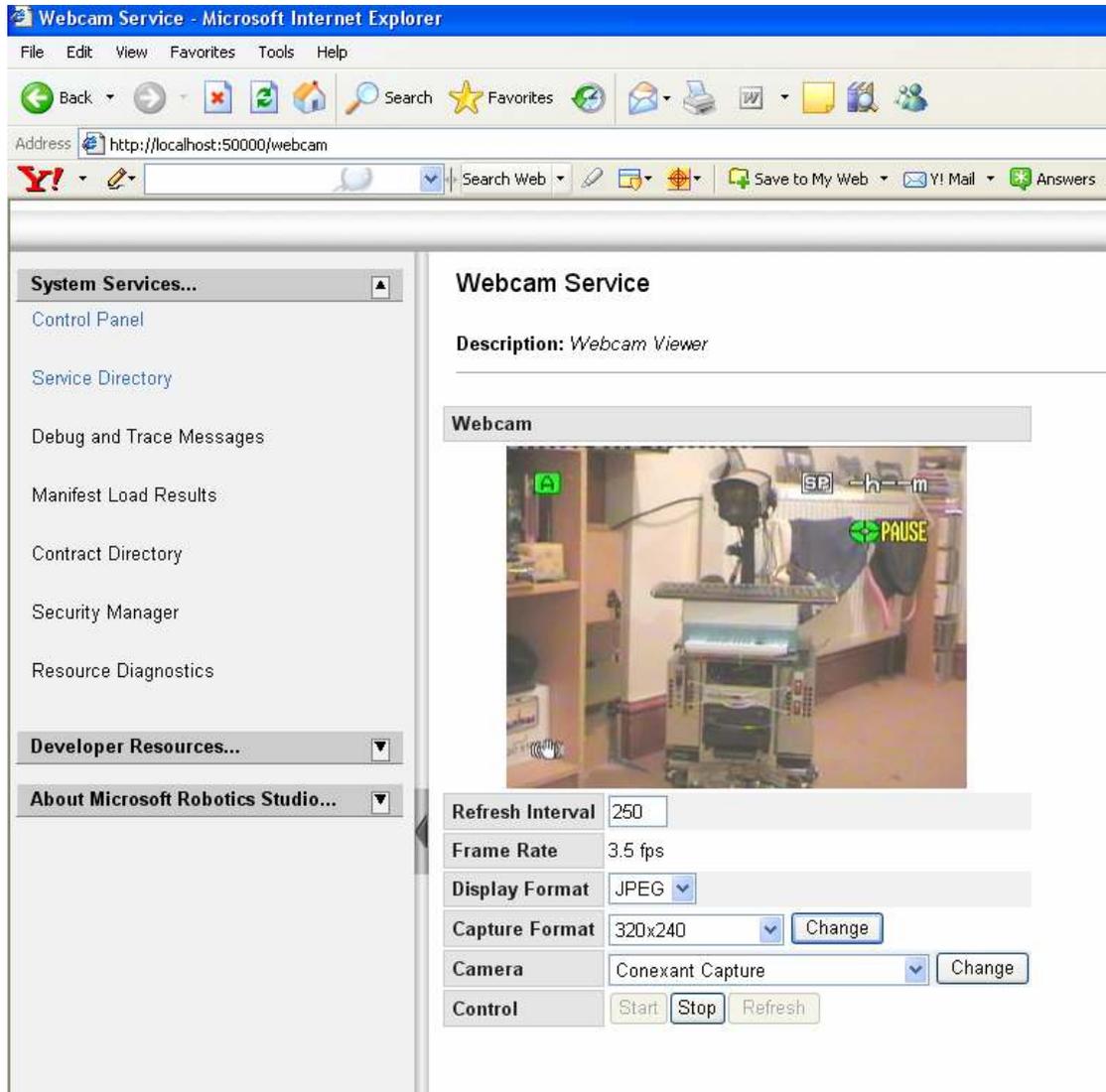
Microsoft Robotics Studio and Vision

I was going to write some vision software for the robot in Direct X and C#. I was going to use disparity to determine distance. It turns out that there's an even easier way to do it and that's to just use **Microsoft Robotics Studio!**

It seems that an awful lot of the video processing algorithms have already been written for you to the extent that there is already inbuilt support for face and gesture recognition. There is even a tutorial on ascertaining distance from a single camera using pinhole camera assumptions.

MSRS Basic Webcam service

There is a basic demo built into MSRS that allows you to access and use a video source connected to your computer. You can view the live video in a web browser or even include the video stream in your own .Net programs.



Accessing the video from your camera is as simple as typing in:

```
dsshost /p:50000 /m:"samples\config\WebCam.manifest.xml"
```

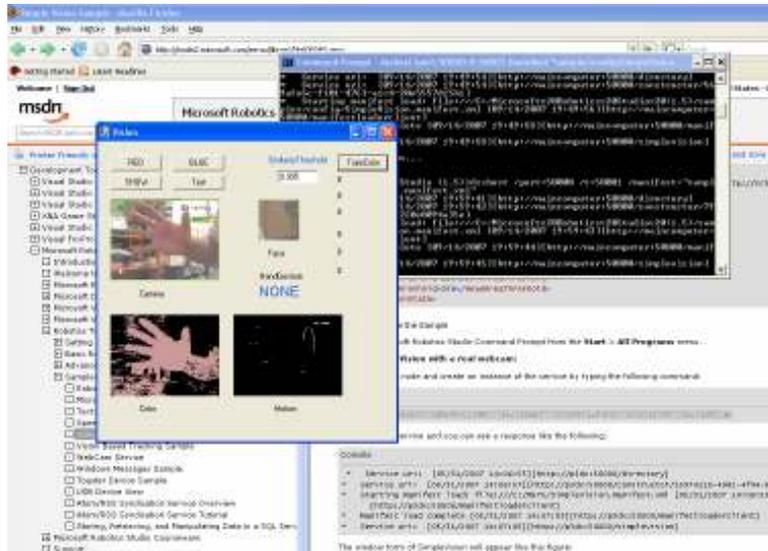
At the MSRS command prompt. The video is immediately viewable on your computer at the location:

<http://localhost:50000/webcam>

Certain web cams and video devices might run into difficulty but I found most problems can be resolved by deleting a special configuration file for the webcams (webcamconfig.xml).

MSRS Simple vision program

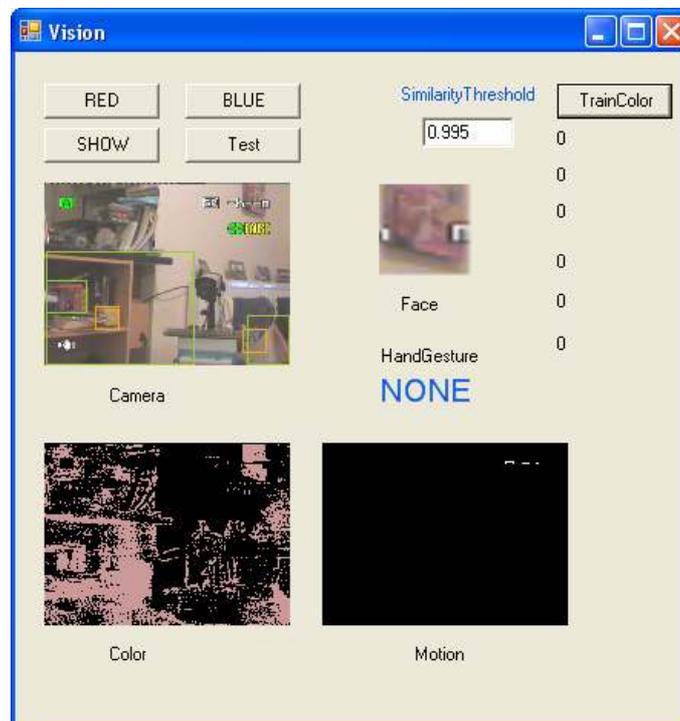
Even better than the webcam demo is a vision program that comes built into the package. The program simultaneously processes motion, colour face and hand gesture. The program comes with full C# source and is completely hackable.



You can spawn this program by typing:

```
dsshost /p:50000 /t:50001 /m:"samples\config\SimpleVision.manifest.xml"
```

At the MSRS command prompt.



I will be actively exploring MSRS and its built-in speech recognition when I look at the binaural part of my robot head.