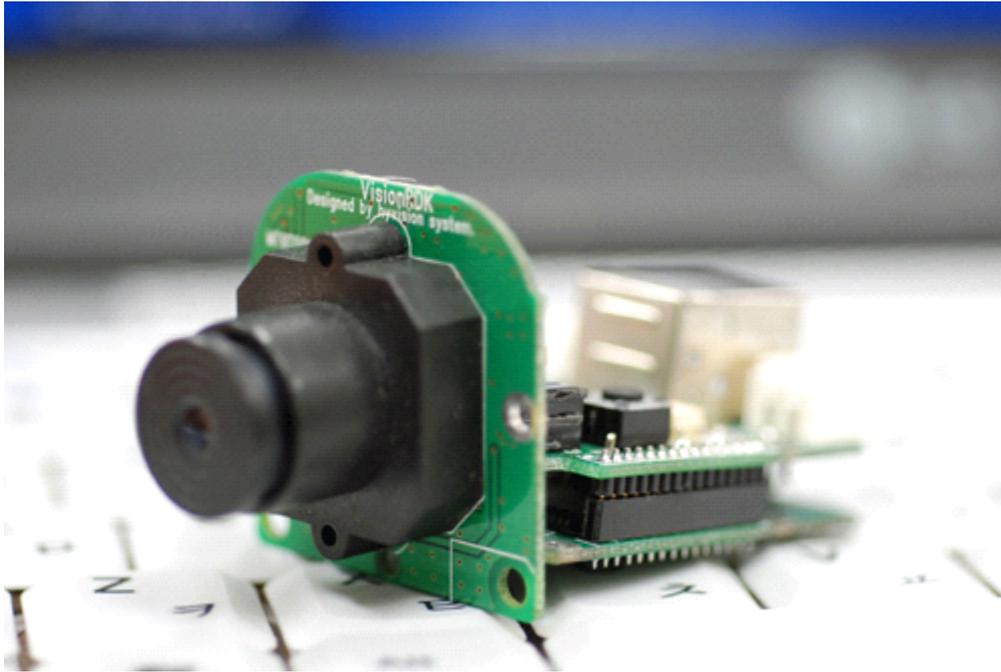


## HVR-2000 Series 이용하기

<http://deios.kr>

이번 시간에는 조금 더 재미있는 것을 해 보겠습니다.  
바로 요놈입니다.



바로 HyVision System의 HVR-2000 Series입니다.  
일반적인 웹캠이야 그냥 연결만 해주면 바로 써먹을 수 있었는데, 요놈은 조금 다릅니다.

USB드라이버야 당연히 깔아줘야 하는거고... 별도의 SDK를 이용해서 Cam에 접근하게 됩니다.

하이비전시스템에서 HVR SDK Interface Guide를 친절하게 작성해 주셔서 별 무리 없이 실습을 진행해 볼 수 있었습니다.

먼저 HVR SDK를 사용하기 위해서는 당연히 라이브러리 세팅을 해줘야겠죠? 필요한 파일은 너무나도 잘 아시겠지만 3가지입니다. 헤더파일, 라이브러리파일, DLL파일이겠죠?

```
include : HVRLibEx.h  
library : HVRLibV10.lib  
bin : HVRLibV10.dll, HVRcore.dll, HVRcv.dll, HVRhg.dll, USB20VisionDLL_V11.dll
```

적당히 잘 세팅해 줍니다.

설마 아직도 라이브러리 세팅법을 모르시는건 아니겠죠?

혹시... 단기 기억 상실증 등으로 잠깐 잊어버리셨다면 “[Visual Studio 2010에서 OpenCV 이용하기\(http://deios.kr/395\)](http://deios.kr/395)”를 참고해서 세팅해 주세요~

백문이불여일타~! 일단 코드부터 보고 시작하겠습니다.

```

// CamTest.cpp : 콘솔 응용 프로그램에 대한 진입점을 정의합니다.
#include "stdafx.h"
#include <cv.h>
#include <highgui.h>
#include "HVRLibEx.h"

int _tmain(int argc, _TCHAR* argv[])
{
    int nDrvNum;
    int nDrvInform[MAX_DEVICE];
    memset(nDrvInform, NULL, sizeof(nDrvInform));

    // 연결된 Camera를 초기화 하고 식별자를 구함
    nDrvNum = HVR_camDrvInit(&nDrvInform[0]);

    // Cam이 여러 개 연결되었을 경우 각각의 식별자를 구한다.
    for(int i = 0; i < nDrvNum; i++){
        printf("Device sequence=%3d, id=%x\n", i, nDrvInform[i]);
    }

    if(!HVR_ispResolutionCmd(VGA, HVR2300R, 0)){
        printf("해상도 설정에 실패하였습니다.\n");
        return 1;
    }

    // 센서 초기화
    if(!HVR_camSensorInit(HVR2300R, 0)){
        printf("센서가 초기화 되지 않았습니다.\n");
        return 1;
    }

    // Exposure Time(빛을 받아드리는 속도)를 메뉴얼로 설정한다.
    if(!HVR_ispSetExposureTime(5000, HVR2300R, 0))
        printf("HVR_ispSetExposureTime의 입력값이 Range를 벗어났거나, 정상동작
하지 않습니다.\n");

    BYTE *pRAWBuf;

    pRAWBuf = new BYTE[640*480];
    IplImage *image = cvCreateImage(cvSize(640,480),IPL_DEPTH_8U,1);
    IplImage *imageItp = cvCreateImage(cvSize(640,480),IPL_DEPTH_8U,3);

    cvNamedWindow("Test HVR2300R", 0 );
    cvResizeWindow("Test HVR2300R", 640, 480);

```

```

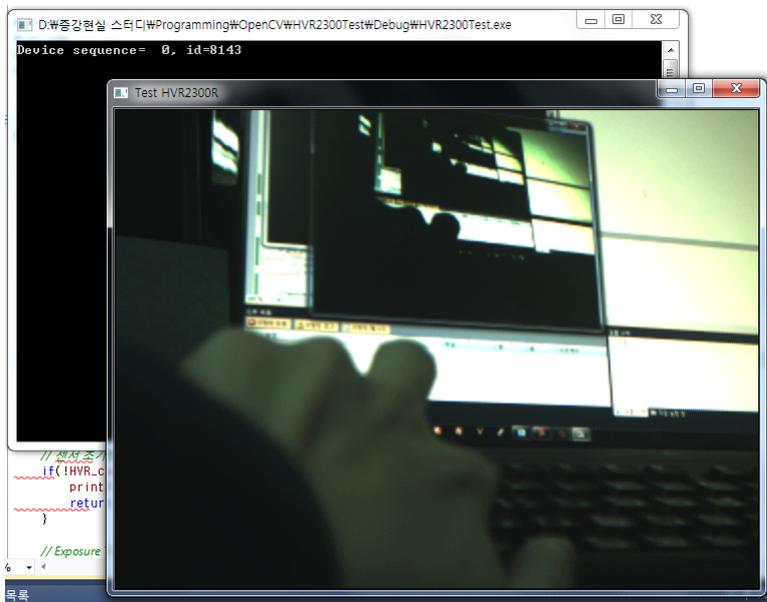
while(1){
    // Camera 영상 데이터를 받아온다
    switch(HVR_camGetImageData(640*480, pRAWBuf, 0)){
        case USB_DATA_PASS :
            image->imageData = (char*)pRAWBuf;
            cvCvtColor( image, imageItp, CV_BayerGB2BGR);
            cvShowImage( "Test HVR2300R", imageItp );
            break;
        case USB_DATA_FAIL :
            printf("USB_DATA_FAIL.\n");
            continue;
        case USB_BAND_HALT :
            printf("USB_BAND_HALT.\n");
            return 1;
    }
    if( cvWaitKey(10) >= 0 ) break;
}

delete pRAWBuf;
cvReleaseImage(&image);
cvReleaseImage(&imageItp);
cvDestroyWindow( "Test HVR2300R" );

return 0;
}

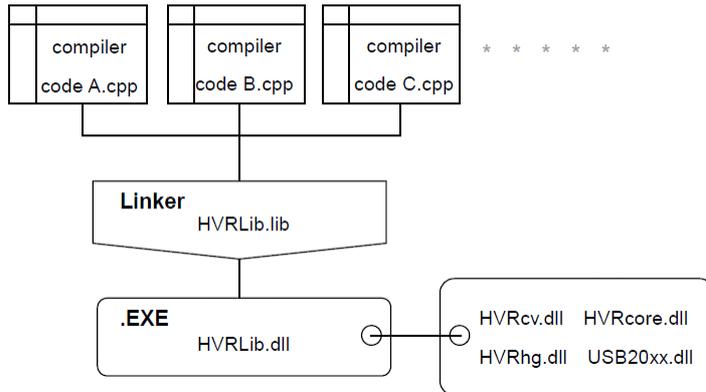
```

실행 화면부터 보여드릴게요~



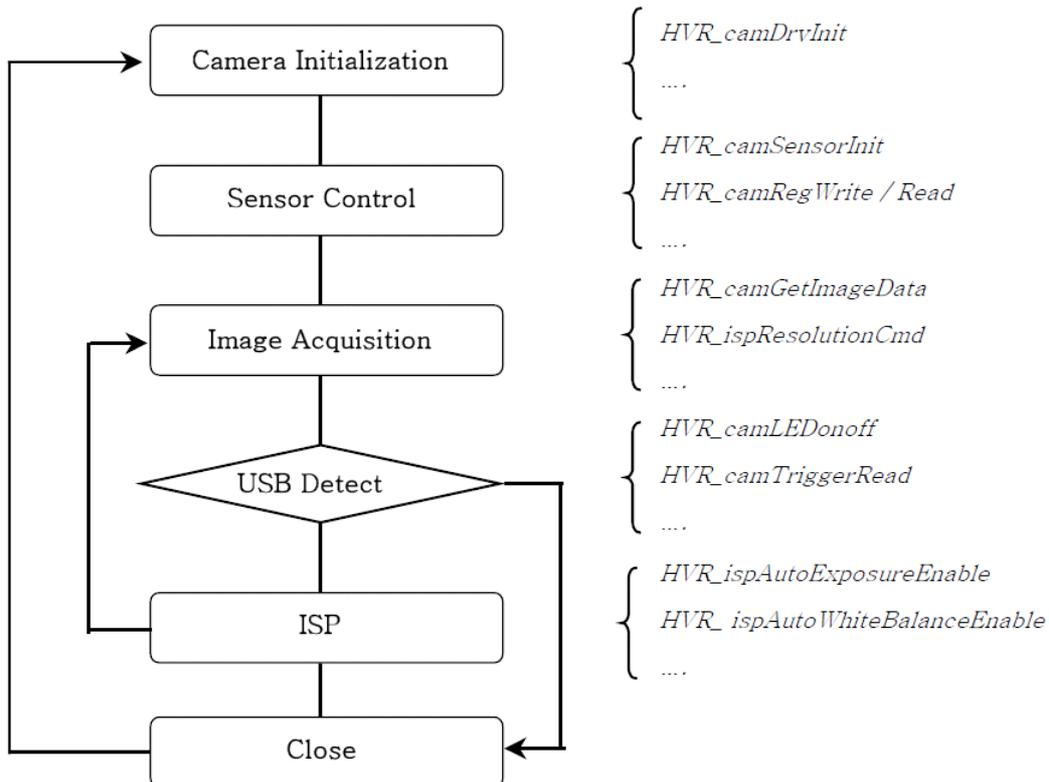
SDK 인터페이스 가이드와 위 코드만 본다면 너무나 쉽게 이해가 가실 겁니다. 때문에 저는 이 SDK 인터페이스 가이드에서 핵심이 되는 부분 몇 가지를 보여드리고 끝내도록 하겠습니다.

먼저 SDK 구조도입니다.



막상 이렇게 보니 그다지 핵심이랄 것도 없네요... 뭐 그냥 SDK 빌드를 저런식으로 했다는군요...

사실 이놈이 핵심입니다. 프로그래밍 흐름과 관련 함수를 대략적으로 보여주고 있네요... 위 코드랑 비교해 봐도 딱 보이구요~



짜구려 웹캠만 가지고 놀다가 조금 비싼 비전카메라 가지고 놀아보니 재미있네요~ ㅎㅎ