

～きょうびのOpenCVはどうよ～

1次元信号 $f(x)$ では x が定まれば $f(x)$ の値は定まります。しかし、2次元信号では x と y の値が定まらないと $f(x, y)$ の値を定めることはできません。

例えば 図 3.1 に示すように、

$f(x) = \sin x$ ならば、 $x = \pi/2$ で $f(x) = 1$ となりますが、 $f(x, y) = \sin(x + y)$ では $x = \pi/2$ だけでは $f(x, y) = \sin(\pi/2 + y) = \cos y$ となり、 y の値も与えなくては $f(x, y)$ の値を定めることはできません。

音声信号は1次元信号ですが、写真や絵画などの静止画像は2次元の信号です。つまり、音声信号では時間 t が定まれば、 $f(t)$ として関数値を定めることができますが、写真や絵画などの静止画像では、画面上の水平方向の位置 x と、垂直方向の位置 y とを定めなければ、関数値 $f(x, y)$ は定まりません。

The value of $f(x, y)$ at any point is gives the pixel value at that point of an image.

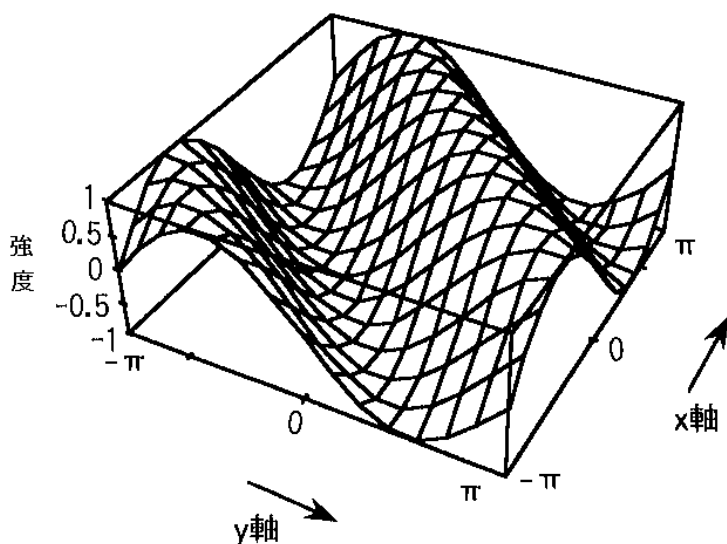


図 3.1 2次元信号 $\sin(x+y)$ two dimensional signal $\sin(x+y)$

0 環境の準備 ～ OpenCV を使う

0.1 OpenCV

よくつかわれるライブラリに OpenCV がある。さまざまな環境にポーティングされている。

本家 <http://opencv.org/>

2.x 系と 3.x 系がある。このテキストでは 2.x 系の現時点(2015.10)での最新版 2.4.11 と VS2013 を使う。

0.2 インストール Install

2つの方法があります。

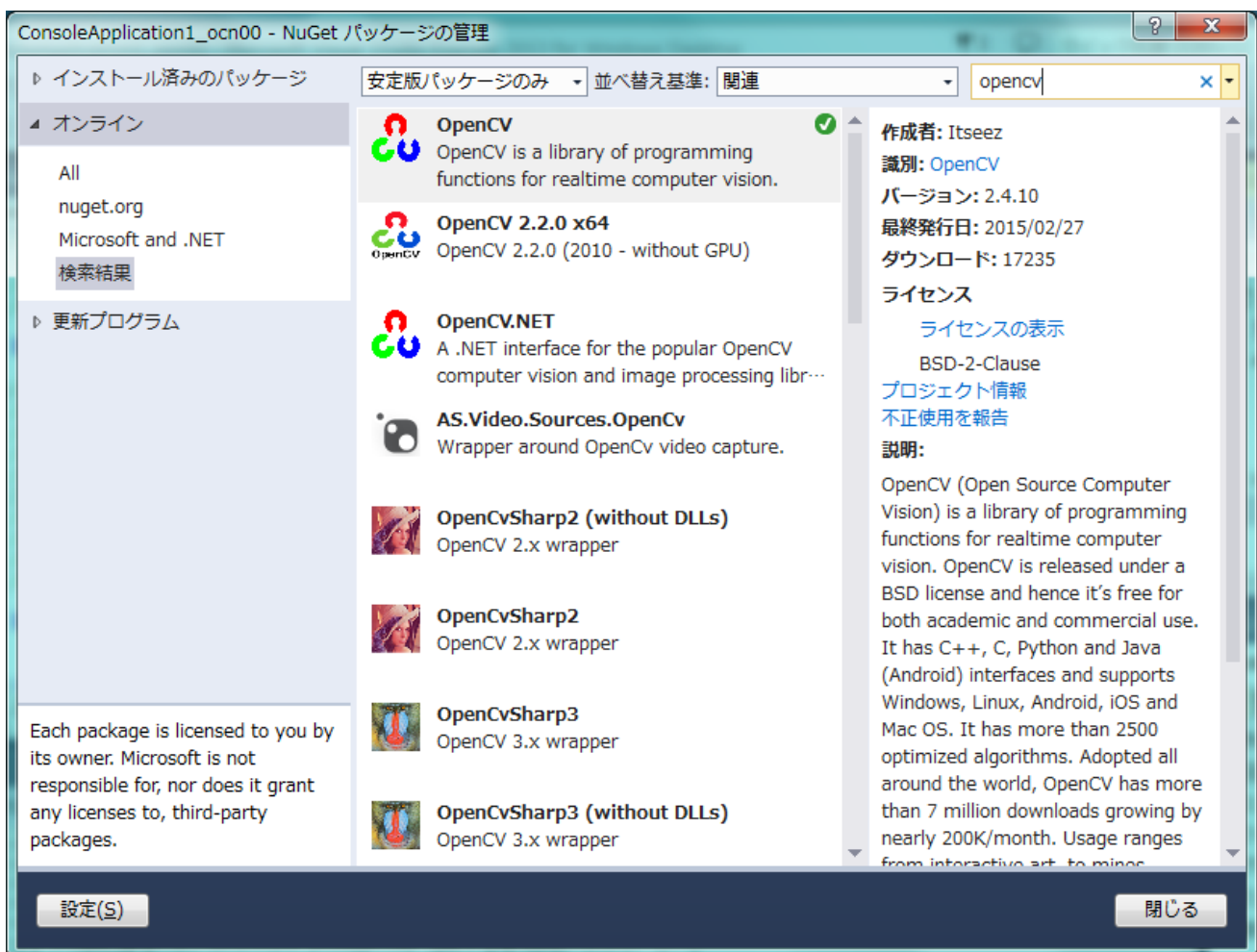
There are 2 ways of install

(a) VS2013 と NuGet を使う方法

VS2013 SP5

<http://www.microsoft.com/ja-jp/download/details.aspx?id=48131>

毎プロジェクトごと NuGet を使ってパッケージを組み込む。



(b) VS2013 と OpenCV を併せてインストールする方法

VS2013 を用意したあと、OpenCV 2.4.11 をダウンロードする。

Download

VS2013 SP5 同上

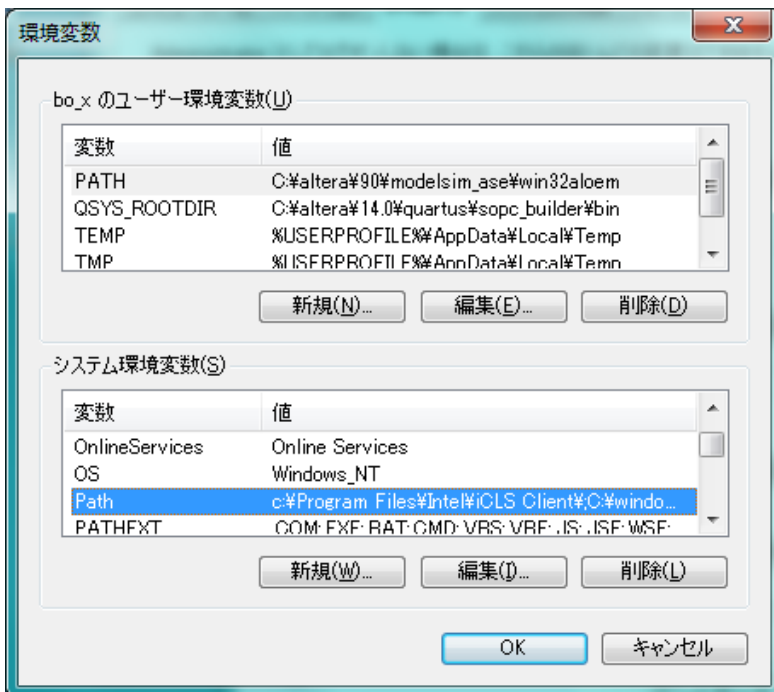
OpenCV2.4.11:

<http://opencv.org/downloads.html>

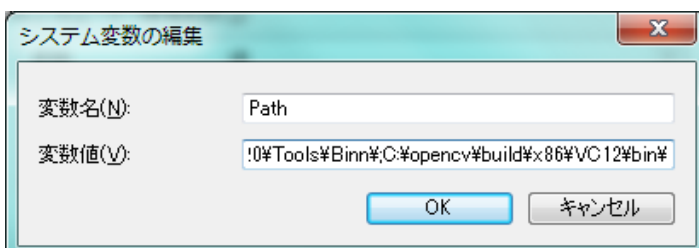
Choose “version2.4.11 OpenCV for Windows “

Unfreeze and Place C:\opencv\

Set Path (Add)



Add ;C:\opencv\build\x86\VC12\bin\

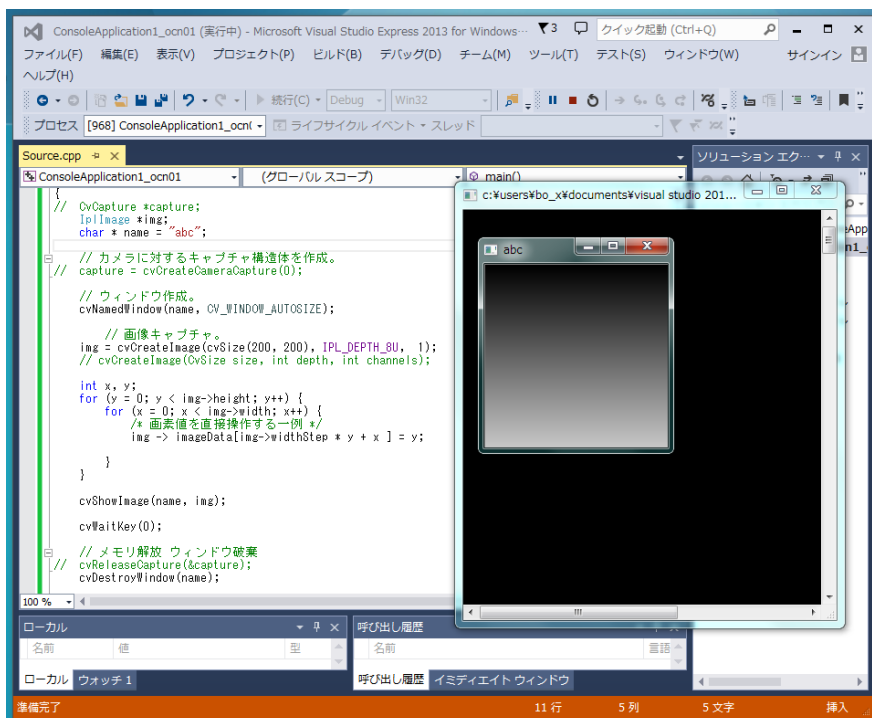


○OpenCV のサンプルプログラム

- ・イメージデータの扱い

```
#include "opencv2\opencv.hpp"
#include "opencv\highgui.h"
int main()
{
    IplImage *img;
    char * name = "abc";
    // ウィンドウ作成。
    cvNamedWindow(name, CV_WINDOW_AUTOSIZE);
    // 画像キャプチャ。
    img = cvCreateImage(cvSize(200, 200), IPL_DEPTH_8U, 1);
    // cvCreateImage(CvSize size, int depth, int channels);
    int x, y;
    for (y = 0; y < img->height; y++) {
        for (x = 0; x < img->width; x++) {
            /* 画素値を直接操作する一例 */
            img -> imageData[img->widthStep * y + x ] = y;      // B
        }
    }
    cvShowImage (name, img);
    cvWaitKey(0);
    // メモリ解放 ウィンドウ破棄
    cvDestroyWindow (name);

    return 0;
}
```



・カメラデバイスからのキャプチャ

```
#include "opencv2\opencv.hpp"
#include "opencv\highgui.h"

int main()
{
    CvCapture *capture;
    IplImage *frame;
    char * name = "camera";

    // カメラに対するキャプチャ構造体を作成。
    capture = cvCreateCameraCapture(0);
    // ウィンドウ作成。
    cvNamedWindow(name, CV_WINDOW_AUTOSIZE);

    while (cvWaitKey(10) == -1){
        // 画像キャプチャ。
        frame = cvQueryFrame(capture);
        cvShowImage(name, frame);
    }

    // メモリ解放 ウィンドウ破棄
    cvReleaseCapture(&capture);
    cvDestroyWindow(name);

    return 0;
}
```

