

# OpenCV Continues

# C++ Library

The standard C++ library is a collection of functions, constants, classes, objects and templates that extends the C++ language providing basic functionality to perform several tasks, like classes to interact with the operating system, data containers, manipulators to operate with them and algorithms commonly needed.

# Compiling OpenCV

```
gcc `pkg-config --cflags opencv` `pkg-config --libs opencv` -o my-opencv-prgm my-opencv-prgm.c
```

Reference -

<http://opencv.willowgarage.com/wiki/CompileOpenCVUsingLinux>

# Why OpenCV

Its Free

Matlab is just way too slow

OpenCV is lot more efficient

Impossible to develop applications in Matlab

# Coding with OpenCV

- We will be using OpenCV 2.3
- Always use `IplImage*`
- ```
typedef struct _IplImage {  
    int nChannels;  
    int depth;  
    int width;  
    int height;  
    int imageSize;  
    char* imageData;  
    int widthStep;  
} IplImage;
```

# Basic OpenCV code

Display an Image

```
#include "highgui.h"
int main( int argc, char** argv )
{
    IplImage* img = cvLoadImage( argv[1] ); //flags can be given
    cvNamedWindow( "Example1", CV_WINDOW_AUTOSIZE );
    cvShowImage( "Example1", img );
    cvWaitKey(0);
    cvReleaseImage( &img );
    cvDestroyWindow( "Example1" );
}
```

# A Simple transformation

```
void example( IplImage* image )
{
    cvNamedWindow( "Example4-in" );
    cvNamedWindow( "Example4-out" );
    cvShowImage( "Example4-in", image );
    IplImage* out = cvCreateImage(
        cvGetSize(image),IPL_DEPTH_8U,3);
    cvSmooth( image, out, CV_GAUSSIAN, 3, 3 );
    cvShowImage( "Example4-out", out );
    cvReleaseImage( &out );
    cvWaitKey( 0 );
    cvDestroyWindow( "Example4-in" );
    cvDestroyWindow( "Example4-out" );
}
```

# How to understand a function

`void cvSmooth`

( `const CvArr* src, CvArr* dst, int smoothtype = CV_GAUSSIAN, int param1 = 3, int param2 = 0, double param3 = 0, double param4 = 0`);

*Table 5-1. Types of smoothing operations*

| Smooth type      | Name                        | In place? | Nc  | Depth of src | Depth of dst                                | Brief description                                                                                              |
|------------------|-----------------------------|-----------|-----|--------------|---------------------------------------------|----------------------------------------------------------------------------------------------------------------|
| CV_BLUR          | Simple blur                 | Yes       | 1,3 | 8u, 32f      | 8u, 32f                                     | Sum over a <code>param1×param2</code> neighborhood with subsequent scaling by <code>1/(param1×param2)</code> . |
| CV_BLUR_NO_SCALE | Simple blur with no scaling | No        | 1   | 8u           | 16s (for 8u source) or 32f (for 32f source) | Sum over a <code>param1×param2</code> neighborhood.                                                            |
| CV_MEDIAN        | Median blur                 | No        | 1,3 | 8u           | 8u                                          | Find median over a <code>param1×param1</code> square neighborhood.                                             |
| CV_GAUSSIAN      | Gaussian blur               | Yes       | 1,3 | 8u, 32f      | 8u (for 8u source) or 32f (for 32f source)  | Sum over a <code>param1×param2</code> neighborhood.                                                            |
| CV_BILATERAL     | Bilateral filter            | No        | 1,3 | 8u           | 8u                                          | Apply bilateral 3-by-3 filtering with color <code>sigma=param1</code> and a space <code>sigma=param2</code> .  |



# Tips for Future

Read the OpenCV book

The most comprehensive introduction to OpenCV

Read many small tutorials easily available on net

Google stuff according to your needs

Learn C++ and move to OpenCV 2.3