

# OpenCV 3.0

Latest news and the Roadmap

Kirill Kornyakov, Itseez  
ICVS 2013



# Agenda

1. Introduction to OpenCV
2. Current state, latest news
3. OpenCV 3.0 plans and roadmap

# **Open**-source **C**omputer **V**sion Library

1. 2,500+ algorithms and functions
2. Cross-platform, portable API
3. Real-time performance
4. Liberal BSD license
5. Professionally developed



iOS

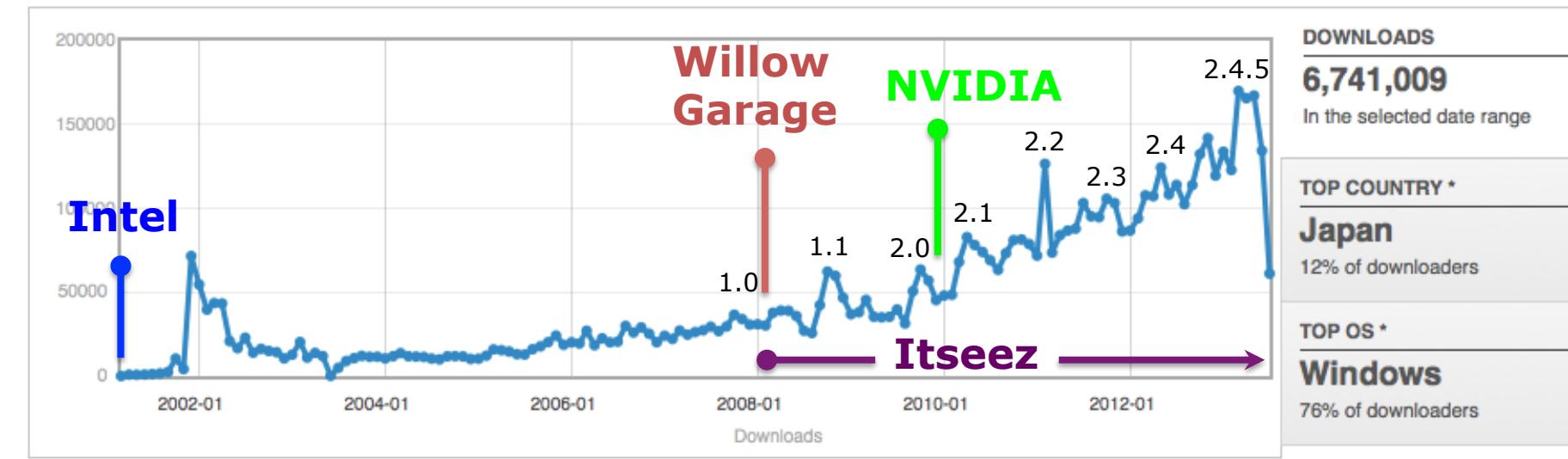
itseez A small graphic of a camera lens with a grid pattern.

# History

Brought to you by: akamaev, ashishkov, etalanin, garybradski, and 4 others

[Home](#) (Change File)

Date Range: 2001-03-15 to 2013-07-14

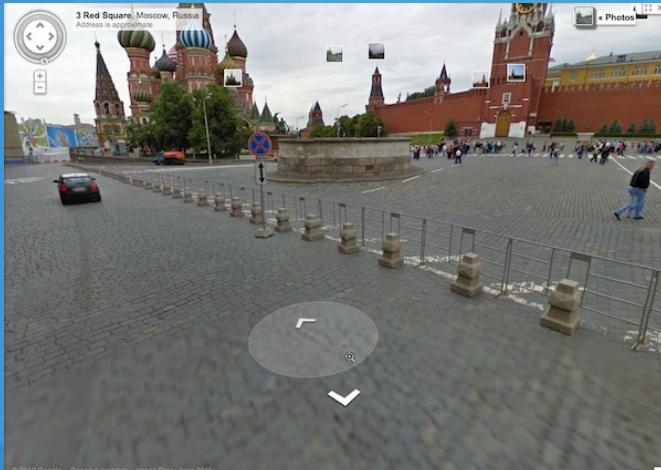


- Professionally maintained by Itseez
- GSoC, corporate contributions
- Contributors from all around the world



# Applications

- Street View Panorama (Google, other projects)
- Vision system of the PR2 robot (Willow Garage)
- Robots for Mars exploration (NASA)
- Quality control of the production of coins (China)



# Sample program

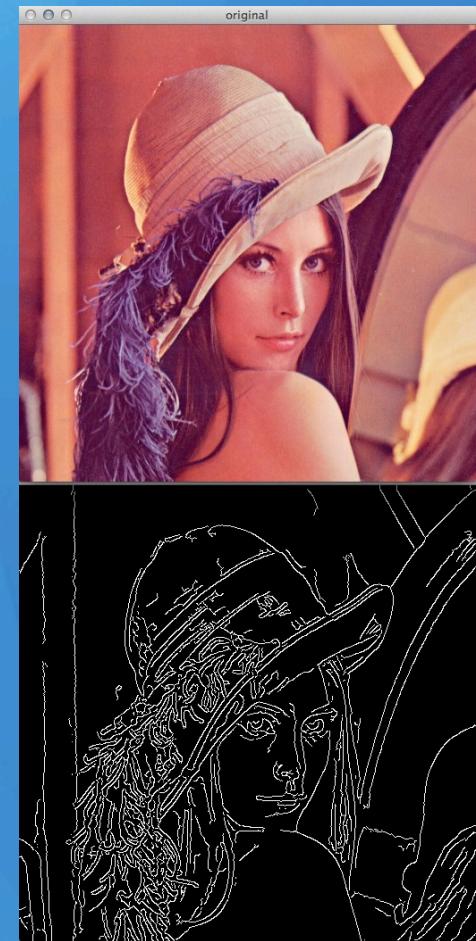
```
#include "opencv2/opencv.hpp"

using namespace cv;

int main(int argc, char** argv)
{
    Mat img, gray;
    img = imread(argv[1], 1);
    imshow("original", img);

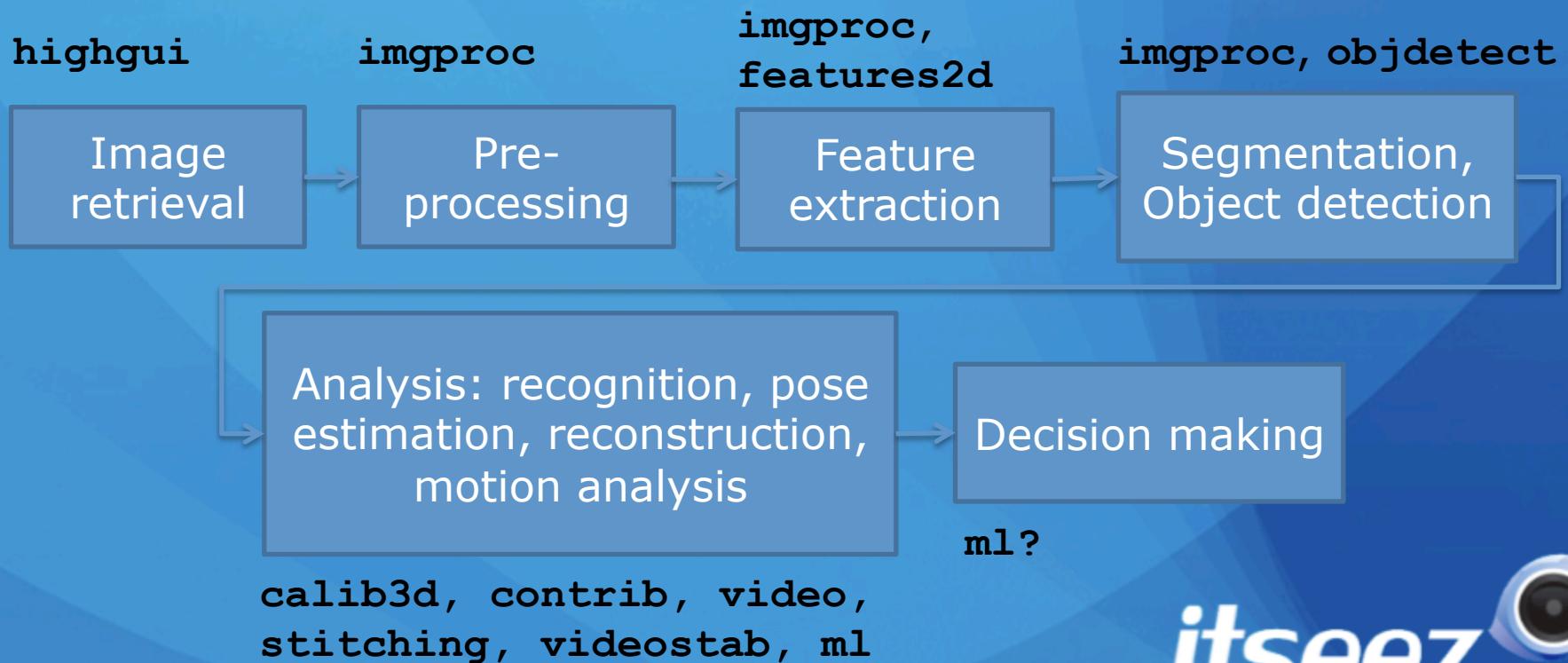
    cvtColor(img, gray, COLOR_BGR2GRAY);
    GaussianBlur(gray, gray, Size(7, 7), 1.5);
    Canny(gray, gray, 0, 50);

    imshow("edges", gray);
    waitKey();
    return 0;
}
```



# OpenCV in Apps

OpenCV – low-level library, providing you with building blocks for your applications.



# Agenda

1. Introduction to OpenCV
2. Current state, latest news
3. OpenCV 3.0 plans and roadmap

# Do you know that OpenCV

- Uses Git 'master' branch for 3.0 preparation. Use '2.4', if you don't want to be pre-alpha tester.
- Will drop C API support soon. C++ is much better.
- Maintains binary compatibility for minor releases.  
And we will likely adopt semantic versioning.

*Given a version MAJOR.MINOR.PATCH:*

- MAJOR – incompatible API changes*
  - MINOR – new functionality, that is backwards-compatible*
  - PATCH – backwards-compatible bug fixes.*
- Features mature infrastructure for regression, accuracy and performance testing.
  - Needs you support.

# C vs C++ API: Focus Detector

C

```
double calcGradients(const IplImage *src,
                      int aperture_size = 7)
{
    CvSize sz = cvGetSize(src);

    IplImage* img16_x = cvCreateImage(sz, IPL_DEPTH_16S, 1);
    IplImage* img16_y = cvCreateImage(sz, IPL_DEPTH_16S, 1);
    cvSobel(src, img16_x, 1, 0, aperture_size);
    cvSobel(src, img16_y, 0, 1, aperture_size);

    IplImage* imgF_x = cvCreateImage(sz, IPL_DEPTH_32F, 1);
    IplImage* imgF_y = cvCreateImage(sz, IPL_DEPTH_32F, 1);
    cvScale(img16_x, imgF_x);
    cvScale(img16_y, imgF_y);

    IplImage* magnitude = cvCreateImage(sz, IPL_DEPTH_32F, 1);
    cvCartToPolar(imgF_x, imgF_y, magnitude);
    double res = cvSum(magnitude).val[0];

    cvReleaseImage(&magnitude );
    cvReleaseImage(&imgF_x);
    cvReleaseImage(&imgF_y);
    cvReleaseImage(&img16_x);
    cvReleaseImage(&img16_y);

    return res;
}
```

C++

```
double contrast_measure(Mat& img)
{
    Mat dx, dy;

    Sobel(img, dx, 1, 0, 3, CV_32F);
    Sobel(img, dy, 0, 1, 3, CV_32F);
    magnitude(dx, dy, dx);

    return sum(dx)[0];
}
```

# Web resources

The image displays three separate browser windows side-by-side, all related to the OpenCV project:

- Left Window:** Shows the "OpenCV 2.4.4.0 documentation" page. It features the OpenCV logo, a "Table Of Contents" sidebar with links like "Welcome to opencv documentation", "Indices and tables", "Next topic", "This Page", and "WHAT'S NEW".
- Middle Window:** Shows the "Questions - OpenCV Q&A" page. It has a search bar, navigation buttons ("ALL", "UNANSWERED"), and a main list of questions. Some visible questions include:
  - "OpenCV C++ Drawing and Analyzing Line" (by Tomazi, 2 hours ago)
  - "how to understand bayes classifier training method?" (by franker, 3 hours ago)
  - "opencv execution" (by murnest, 3 hours ago)
  - "transform phase map to 3D" (by nmnm02003, 3 hours ago)
  - "Drawing a Ling from set of points" (by Tomazi, 6 hours ago)
  - "Open CV & C++" (by Tomazi, 6 hours ago)
- Right Window:** Shows the "answers.opencv.org" forum page. It includes a "Contributors" section with user profiles and a "Tags" section listing various topics with their counts, such as "AndroidOpenCV" (x 4), "C++.OpenCV" (x 4), "c++" (x 3), "Initialization" (x 2), "java" (x 2), "opencv" (x 2), "2d" (x 1), "3d" (x 1), and "accuracy" (x 1).

[opencv.org](http://opencv.org), [docs.opencv.org](http://docs.opencv.org), [answers.opencv.org](http://answers.opencv.org)

# Development infrastructure

The screenshot shows a desktop environment with four browser windows open:

- GitHub**: Shows the repository "itseez/opencv".
- OpenCV - Wiki**: Shows the OpenCV Wiki.
- BuildBot: Open Source Computer Vision**: Shows the BuildBot status.
- pullrequest.opencv.org**: Shows the "Active pull requests to itseez/opencv" page.

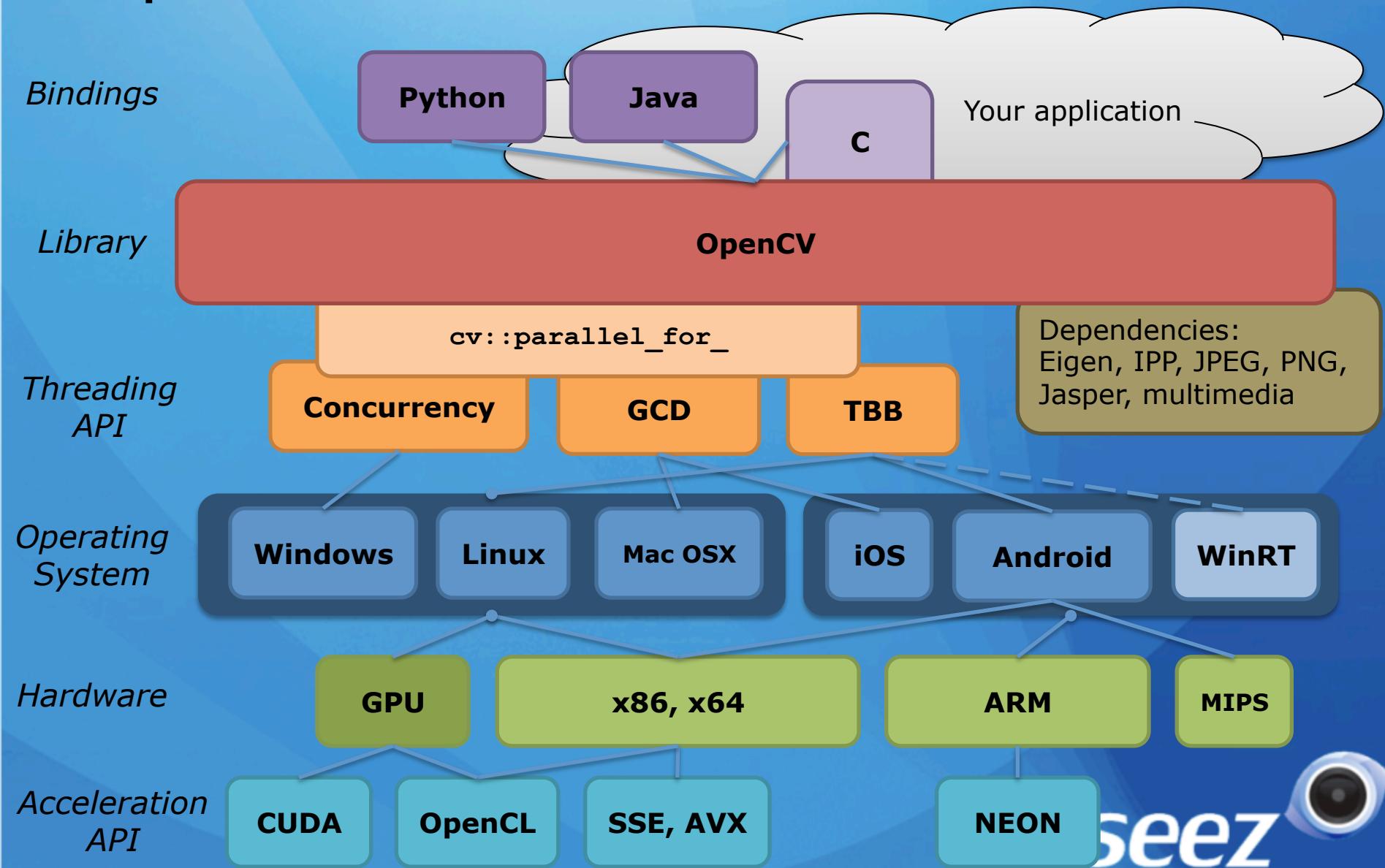
The "pullrequest.opencv.org" window displays the following table:

Total	16										
ID	Branch	Title and description	Win	Lin	Lin2	Mac	And	Doc	Author	Assigned to	Merge status
406	master	Fixed DynamicAdaptedFeatureDetector (#1334) Mono / .NET bindings hi, using IKVM ( <a href="http://www.ikvm.net/">http://www.ikvm.net/</a> ) i've created bindings for Mono / .NET using the existing java bindings, in order to get them built, IKVM needs to be in the system's PATH environment variable. i modified the java for desktop bindings a little so they do the System.loadLibrary call on their own. doing it the same way as in the java examples (letting the user call it) doesn't work in .NET due to different class loading behaviour. i'm fairly new to CMake, so any hints/help in that direction are appreciated - for example i've no clue what'd be the proper way to copy native dependencies (ikvm-native-win32) based on the current target platform (x86/x64). i'd like to see an output folder like this: mono/opencv-mono.dll – the mono bindings mono/IKVM.*.dll – managed assemblies needed by opencv-mono.dll mono/x86/opencv_java249.dll – native dependency needed by opencv-mono.dll mono/x86/ikvm-native-win32-x86.dll – native dependency needed by IKVM.Runtime.JNI.dll mono/x64... i've not yet written any tests yet. simply tried some basic examples which worked just fine. wanted to sort out above mentioned build issues first. thanks, elias	3042 fail	3055 fail	563 fail	2932 fail	1942 fail	2014 fail	ilysenkov	vpisarev	Unsuccessful
604	master	Merge pull request #102 from SpecLad authored 2 days ago ffmpeg_wrapper: build fix	2995 fail	3008 fail	516 fail	2885 fail	1895 fail	1967 fail	azeno	apavlenko	Unsuccessful
649	master	latency patch see <a href="http://code.opencv.org/issues/632">http://code.opencv.org/issues/632</a>	2912 fail	2917 fail	422 fail	2789 success	1801 fail	1873 success	eendebakpt	vpisarev	Unsuccessful
674	master	Bugfix to potential inaccuracy in function:polyfit 1.Modify the default datatype(CV_32F) to CV_64F. 2.Re-write the generation of X matrix, reducing the times of multiplication. The problem has been reported at <a href="http://code.opencv.org/issues/2887">http://code.opencv.org/issues/2887</a>	2987 success	3000 success	508 success	2877 success	1887 success	1959 success	chouilee	vpisarev	Under review
685	2.4	Fixed compatibility issues with libavcodec > 53.25.0 Post libavcodec 53.25.0 the enum CodecID was changed to AVCodecID causing compilation errors saying CodecID was not previously declared.	3083 success	3096 success	604 success	2973 success	1984 success	2056 success	shahibista	vpisarev	Under review
		Make TV-L1 rescaling flexible and add median filtering Previously the pyramid was done with a rescaling factor of 2 (implied by the use of pyrDown). This often leads to inferior results									

<https://github.com/itseez/opencv>, [pullrequest.opencv.org](http://pullrequest.opencv.org)



# OpenCV Environment



# Functionality overview

## Image Processing



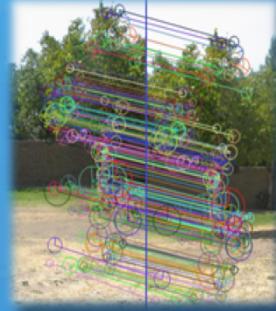
Filters



Transformations



Edges,  
contours

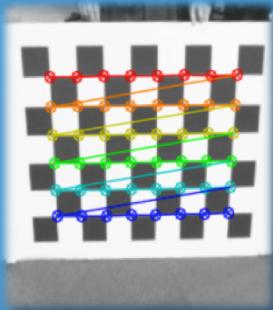


Robust  
features

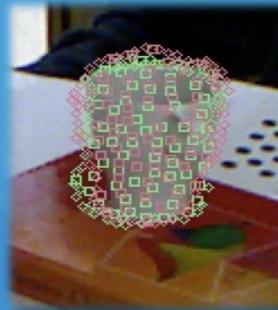


Segmentation

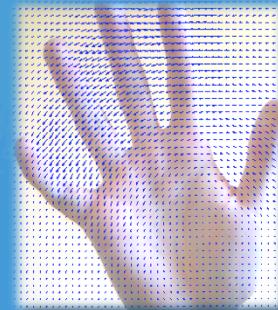
## Video, Stereo, 3D



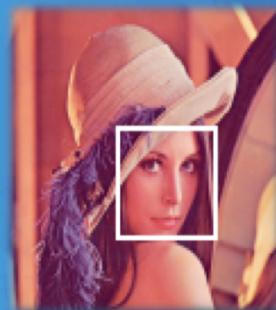
Calibration



Pose  
estimation



Optical Flow



Detection and  
recognition

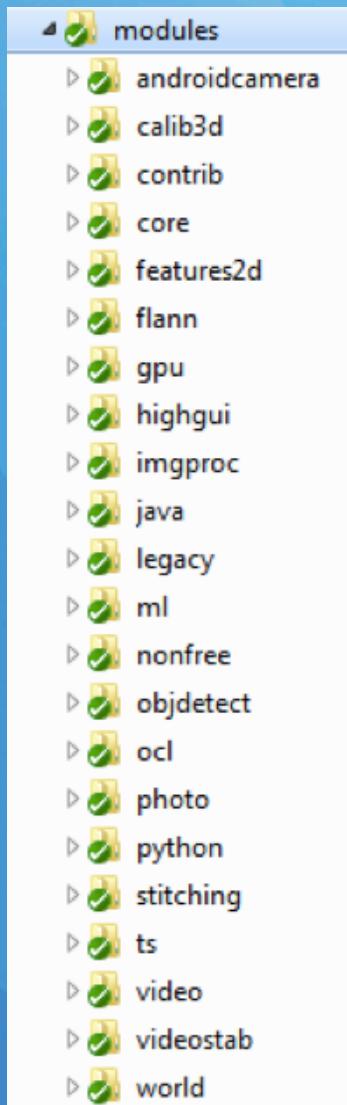


Depth

**itseez**



# Modules



## Algorithmic

- `core`, `imgproc`, `calib3d`, `video`,  
`ml`, `objdetect`, `features2d`
- `photo`, `stitching`, `videostab`, `superres`
- `contrib`, `legacy`, `nonfree`, `flann`

## GPU

- `gpu`, `ocl`

## Infrastructure

- `highgui`, `world`
- `python`, `java`
- `ts`, `androidcamera`

# Agenda

1. Introduction to OpenCV
2. Current state, latest news
3. OpenCV 3.0 plans and roadmap

# OpenCV Roadmap

- Closing 2.4 series
  - 2.4.7 in Oct
  - 2.4.8 in Feb (the last maintenance release)
- Starting 3.0 series
  - 3.0-alpha in Oct
  - 3.0-beta in Dec
  - 3.0 final in Feb

SUMMER

OF

CODE

2013

## General

1. Alex Leontiev, Generic **numerical optimization** module
2. Antonella Cascitelli, Long-term **optical tracking API** in OpenCV
3. Lluís Gómez i Bigordà, Implementation of Neumann & Matas algorithm for scene **text detection and recognition**
4. Digvijay Singh, Fast and Robust 1D and 2D **Barcode Detection** and Rectification
5. Juan Manuel Perez Rua, Implementation and Validation of the **Shape Context Descriptor**
6. Daniel Angelov, **Line Segment Detection**

## Bindings

1. Hilton Bristow, **MATLAB Code Generator** for OpenCV
2. Oli Wilkie, OpenCV for Android: **Augmented Reality Sample**
3. Abid Rahman, **Python Tutorials**



SUMMER

OF

CODE

2013

## 3D

1. Di YANG, SfM integration: **PTAM** implementation project
2. Ozan Tonkal, **Visualizer for SfM**
3. Gurpinder Singh Sandhu, **Hand Tracking with Kinect**

## Computational Photography

1. Fedor Morozov, **High Dynamic Range** imaging
2. Rahul Verma, Implementing Exemplar-Based (Patch Propagation) **Image Inpainting** in OpenCV
3. Takahito Aoto, **Photometric calibration** for imaging devices

# Major 3.0 updates

1. Lots of new functionality
2. API changes
3. Acceleration

# API changes in 3.0

*Migration should be smooth!*

- Mostly cleanings
  - Refined C++ API
  - Use `cv::Algorithm` everywhere
- API changes
  - C API will be marked as deprecated
  - Old Python API will be deprecated
  - Monstrous modules will be split into micromodules
  - Extra modules

# Extra modules

Possibility to add new modules  
without putting them into the OpenCV tree:

```
opencv/
  modules/
    core/
      include/, doc/, src/, test/, ...
      CMakeLists.txt
    imgproc
    ...
my_extra_modules/
  sfm/
    include/, doc/, src/, test/, ...
    CMakeLists.txt
  ...
```



Experimental or  
proprietary code.

```
$ cmake -D OPENCV_EXTRA_MODULES_PATH=~/my_extra_modules ...
```

# Acceleration in 3.0

- Sufficiently improved CUDA and OpenCL modules
  - Mobile CUDA support
  - Universal OpenCL binaries (CPU, GPU)
- Hardware Abstraction Layer (HAL)
  - IPP, FastCV-like low-level API to accelerate OpenCV on different HW
- Open-source NEON optimizations
  - iOS, Android, Embedded, but not yet confirmed

# Library's future

- More functionality
  - Flat architecture, modules with single responsibility
- Better performance
  - HW vendors are primary sponsors
- New platforms
  - WinRT, QNX (BlackBerry)
- New bindings
  - C# (official support), JavaScript, Haskell, D, ...

# Links

1. Home: [opencv.org](http://opencv.org)
2. Documentation: [docs.opencv.org](http://docs.opencv.org)
3. Q&A forum: [answers.opencv.org](http://answers.opencv.org)
4. Report issues: [code.opencv.org](http://code.opencv.org)
5. Develop: <https://github.com/Itseez/opencv>

# Thank you!

- Any questions?
- [kirill.kornyakov@itseez.com](mailto:kirill.kornyakov@itseez.com)