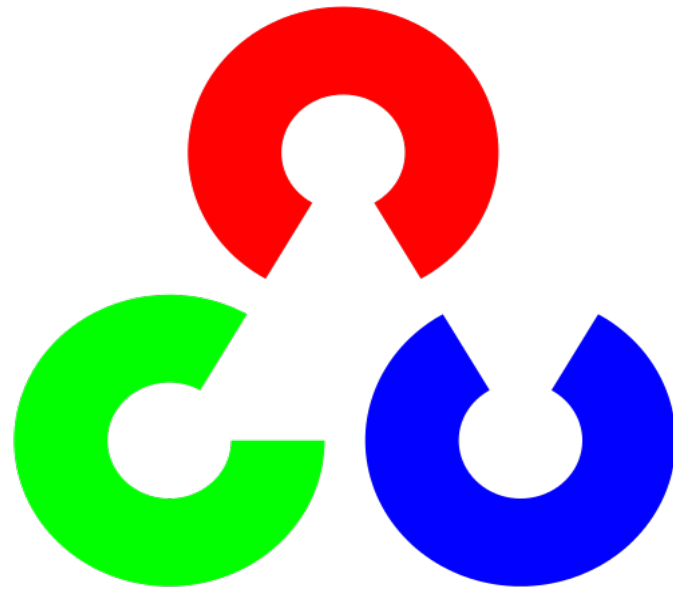


Introducción a OpenCV



OpenCV

Introducción a OpenCV

- 1. ¿Qué es OpenCV?**
- 2. Estructura**
- 3. Instalación de OpenCV**
- 4. Ejemplos de aplicación**

¿Qué es OpenCV?

- Librería de visión por computador.
- Orientada a visión en tiempo real.
- Muy popular, no sólo en ambientes académicos, sino también en la empresa.
- Usada y ampliada por investigadores de todo el mundo.
- En general muy bien optimizada.

¿Qué es OpenCV?

- Open source y libre (lic. BSD)
- Interfaces : C, C++, Python, JAVA...
- Linux, Windows, Mac, Android...
- Ahora versión 4.5.3. Coexiste con la versión 3.4.15

¿Qué es OpenCV?

- Documentación online:

<http://docs.opencv.org/>

- Tutoriales específicos para OpenCV con Python

https://docs.opencv.org/4.4.0/d6/d00/tutorial_py_root.html

- Búsqueda por Internet de ayuda específica

Introducción a OpenCV

Estructura

Miles de aplicaciones

General Image Processing Functions

Segmentation

Transforms

Machine Learning: • Detection, • Recognition

Geometric descriptors

Features

Tracking

Matrix Math

Image Pyramids

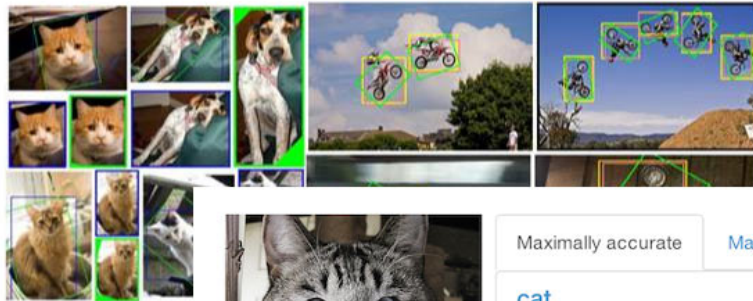
Camera calibration, Stereo, 3D

Utilities and Data Structures

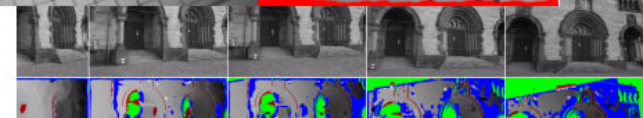
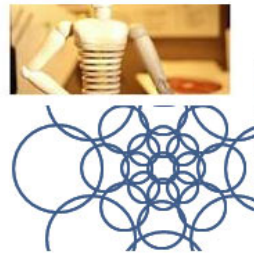
Fitting

G. Bradsky

Deep Learning



Maximally accurate	Maximally specific
cat	1.80727
domestic cat	1.74727
feline	1.72787
tabby	0.99133
domestic animal	0.78542



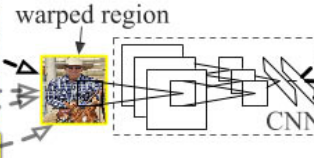
R-CNN: Regions with CNN features



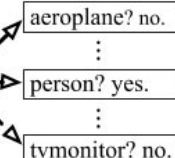
1. Input image



2. Extract region proposals (~2k)

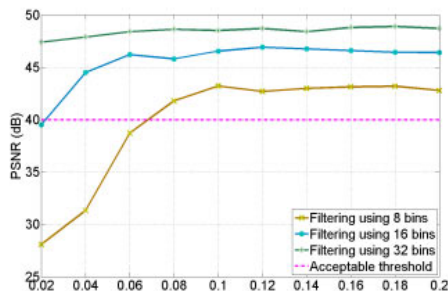
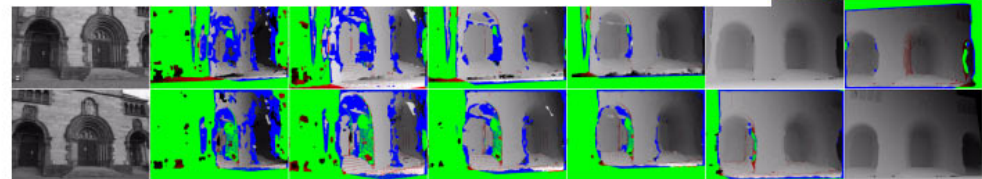


3. Compute CNN features



4. Classify regions

aeroplane? no.
:
person? yes.
:
tvmonitor? no.



Estructura modular

- Main modules:
 - core. **Core functionality**
 - imgproc. **Image processing**
 - imgcodecs. **Image file reading and writing**
 - videoio. **Video I/O**
 - highgui. **High-level GUI**
 - video. **Video Analysis**
 - calib3d. **Camera Calibration and 3D Reconstruction**
 - features2d. **2D Features Framework**
 - objdetect. **Object Detection**
 - dnn. **Deep Neural Network module**
 - ml. **Machine Learning**
 - flann. **Clustering and Search in Multi-Dimensional Spaces**
 - photo. **Computational Photography**
 - stitching. **Images stitching**
 - cudaarithm. **Operations on Matrices**
 - cudabgsegm. **Background Segmentation**
 - cudacodec. **Video Encoding/Decoding**
 - cudafeatures2d. **Feature Detection and Description**
 - cudafilters. **Image Filtering**
 - cudaimgproc. **Image Processing**
 - cudalegacy. **Legacy support**
 - cudaobjdetect. **Object Detection**
 - cudaoptflow. **Optical Flow**
 - cudastereo. **Stereo Correspondence**
 - cudawarping. **Image Warping**
 - cudev. **Device layer**
 - shape. **Shape Distance and Matching**
 - superres. **Super Resolution**
 - videostab. **Video Stabilization**
 - viz. **3D Visualizer**

Estructura modular

- Extra modules:
 - aruco. **ArUco Marker Detection**
 - bgsegm. **Improved Background-Foreground Segmentation Methods**
 - bioinspired. **Biologically inspired vision models and derivated tools**
 - ccalib. **Custom Calibration Pattern for 3D reconstruction**
 - cnn_3dobj. **3D object recognition and pose estimation API**
 - cvv. **GUI for Interactive Visual Debugging of Computer Vision Programs**
 - datasets. **Framework for working with different datasets**
 - dnn_modern. **Deep Learning Modern Module**
 - dpm. **Deformable Part-based Models**
 - face. **Face Recognition**
 - freetype. **Drawing UTF-8 strings with freetype/harfbuzz**
 - fuzzy. **Image processing based on fuzzy mathematics**
 - hdf. **Hierarchical Data Format I/O routines**
 - img_hash. **The module brings implementations of different image hashing algorithms.**
 - line_descriptor. **Binary descriptors for lines extracted from an image**
 - matlab. **MATLAB Bridge**
 - optflow. **Optical Flow Algorithms**
 - phase_unwrapping. **Phase Unwrapping API**
 - plot. **Plot function for Mat data**
 - reg. **Image Registration**
 - rgbd. **RGB-Depth Processing**
 - saliency. **Saliency API**
 - sfm. **Structure From Motion**
 - stereo. **Stereo Correspondance Algorithms**
 - structured_light. **Structured Light API**
 - surface_matching. **Surface Matching**
 - text. **Scene Text Detection and Recognition**
 - tracking. **Tracking API**
 - xfeatures2d. **Extra 2D Features Framework**
 - ximgproc. **Extended Image Processing**
 - xobjdetect. **Extended object detection**
 - xphoto. **Additional photo processing algorithms**

Introducción a OpenCV

Instalación de OpenCV

Instalación general

- Este método es válido para todos los sistemas operativos
- Instalaremos OpenCV utilizando el gestor de paquetes de Python 'pip'

`pip install opencv_python` (pip3 en Linux)

`pip install opencv-contrib-python` (pip3 en Linux)

- Si tenemos dos versiones de Python (Linux o Mac) podría ser necesario usar pip3.
- En el caso de Windows y Mac se hace en un terminal de Anaconda. En Linux en un terminal.

Instalación alternativa en Windows

- Bajarse las versiones compiladas en:
- <https://www.lfd.uci.edu/~gohlke/pythonlibs/>
- Hay que bajarse las últimas versiones de dos ficheros, uno correspondiente a **OpenCV** y otro a **Numpy**.
- Una vez descargados y desde un terminal de Anaconda se ejecuta:
 - `pip install numpy-1.21.2+mkl-cp38-cp38-win_amd64.whl`
 - `pip install opencv_python-4.5.3-cp38-cp38-win_amd64.whl`

Introducción a OpenCV

Ejemplos de aplicaciones

Ejemplos de procesamiento de imagen

- En la dirección:

https://docs.opencv.org/4.1.1/d2/d96/tutorial_py_table_of_contents_imgproc.html

podemos encontrar un buen número de aplicaciones básicas de procesamiento de imagen.

Ejemplos de visión artificial

- <https://youtu.be/tlC2O9T9jks?t=32>
- <https://youtu.be/m4qv9rK8k9s>
- En internet es posible encontrar múltiples ejemplos de reconocimiento y visión artificial.