



OpenCV

UNE INTRODUCTION

En quelques chiffres



- ▶ Téléchargements : > 9 millions
- ▶ Communauté : > 47 000 utilisateurs
- ▶ Nombre d'algorithmes : > 2500
- ▶ Première version : Juin 2000 (initialement intel, depuis 2008 : Willow Garage)
- ▶ Sources : Site web OpenCV + wikipedia (2016)

C'est quoi ?

- ▶ Une bibliothèque
 - ▶ Un ensemble de fonctionnalités
 - ▶ Des structures de données
- ▶ Un ensemble d'API (pas toujours complètes, attention !)
 - ▶ C / C++
 - ▶ Java
 - ▶ Python
- ▶ Une utilisation sous licence BSD (utilisation libre académique et commerciale)



Avantages / inconvénients



- ▶ +
 - ▶ Licence BSD
 - ▶ Beaucoup de fonctionnalités
 - ▶ Support de Linux, Windows, MacOS, iOS, Android
 - ▶ Accélération CUDA, OpenCL
 - ▶ ...
- ▶ -
 - ▶ Pas toujours simple
 - ▶ Parfois hétérogène
 - ▶ Priorités parfois ... opaques (choix de fonctionnalités)

Questions

- ▶ Vision par ordinateur ?
- ▶ Qu'attendez-vous d'une bibliothèque en terme de :
 - ▶ Traitement d'images ?
 - ▶ Traitement de vidéos ?
 - ▶ Algorithmes d'apprentissage ?
 - ▶ Calculs matriciels ?



Traitement d'images



- ▶ Fonctionnalités
 - ▶ Import/Export de nombreux format (BMP, JPEG, PNG, PNM, TIFF ...)
 - ▶ Histogrammes
 - ▶ Filtrage
 - ▶ Seuillage
 - ▶ Segmentation
 - ▶ Morphologie Mathématique
 - ▶ ...
- ▶ Pourquoi ? Quelles applications ?

Traitement de vidéos

- ▶ Fonctionnalités :
 - ▶ Import/Export de nombreux formats – lien avec bibliothèque extérieure
 - ▶ Détection de droites, de segment et de cercles par Transformée de Hough
 - ▶ Détection de visages par la méthode de Viola et Jones
 - ▶ Cascade de classifieurs
 - ▶ Mouvement
 - ▶ historique du mouvement
 - ▶ poursuite d'objets par mean-shift ou Camshift
 - ▶ détection de points d'intérêts
 - ▶ estimation de flux optique (Méthode de Lucas–Kanade)
 - ▶ Autres
 - ▶ triangulation de Delaunay, diagramme de Voronoi
 - ▶ enveloppe convexe
 - ▶ ajustement d'une ellipse à un ensemble de points par la méthode des moindres carrés
- ▶ Pourquoi ? Quelles applications ?



Algorithmes d'apprentissage



- ▶ Fonctionnalités :
 - ▶ K-means
 - ▶ AdaBoost et divers algorithmes de boosting
 - ▶ Réseau de neurones artificiels
 - ▶ Séparateur à vaste marge (SVM)
 - ▶ Estimateur statistique (ex : basé EQM)
 - ▶ Arbres de décision et forêts aléatoires
- ▶ Pourquoi ? Quelles applications ?

Calcul Matriciel



- ▶ Fonctionnalités
 - ▶ Transposée
 - ▶ Calcul du déterminant
 - ▶ Inversion
 - ▶ Multiplication (par une matrice ou un scalaire)
 - ▶ Calcul des valeurs propres
- ▶ Pourquoi ? Quelles applications ?

Les modules OpenCV - main

- ▶ core. [Core functionality](#)
- ▶ imgproc. [Image processing](#)
- ▶ imgcodecs. [Image file reading and writing](#)
- ▶ videoio. [Media I/O](#)
- ▶ highgui. [High-level GUI](#)
- ▶ video. [Video Analysis](#)
- ▶ calib3d. [Camera Calibration and 3D Reconstruction](#)
- ▶ features2d. [2D Features Framework](#)
- ▶ objdetect. [Object Detection](#)
- ▶ 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](#)
- ▶ cudaimproc. [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](#)



Les modules OpenCV - extras

- ▶ 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](#)
- ▶ cvv. [GUI for Interactive Visual Debugging of Computer Vision Programs](#)
- ▶ datasets. [Framework for working with different datasets](#)
- ▶ dnn. [Deep Neural Network module](#)
- ▶ dpm. [Deformable Part-based Models](#)
- ▶ face. [Face Recognition](#)
- ▶ fuzzy. [Image processing based on fuzzy mathematics](#)
- ▶ hdf. [Hierarchical Data Format I/O routines](#)
- ▶ line_descriptor. [Binary descriptors for lines extracted from an image](#)
- ▶ matlab. [MATLAB Bridge](#)
- ▶ optflow. [Optical Flow Algorithms](#)
- ▶ 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](#)



Les difficultés ...



- ▶ Pas simple de s'y retrouver dans l'API ! (ni dans la documentation)
- ▶ API C/C++, Java, Python différentes
- ▶ Nécessité de passer par les structures de données OpenCV
 - ▶ Wimage, WImageBuffer ...
 - ▶ Un exemple : il y a même des marqueurs pour kmeans !
 - ▶ Il faut transtyper pour travailler avec Qt, ITK, ... (classique)
- ▶ Attention aux mauvaises utilisations (coté « boîte noire »)

Nos problèmes



- ▶ Directs (partie logicielle)
 - ▶ Les algorithmes ne sont pas tous dans les mêmes modules ... (ils sont distribués dans la bibliothèques) -> difficile de les trouver dans l'API
 - ▶ Il faut faire communiquer Qt et OpenCV (transystème)
 - ▶ On va les utiliser en « boîte noire » -> limites ?
- ▶ Indirects (dûs à l'implémentation sur robot)
 - ▶ Robustesse ? (luminosité, ...)
 - ▶ Temps de calcul
 - ▶ Taille de la bibliothèque

Et maintenant quelque chose de complètement différent ...

- ▶ Partie pratique – gestion de projet :
 - ▶ Notion de tâche
 - ▶ Notion de communication : comment faites-vous ?
 - ▶ Notion de Kanban

