|  |  |
| --- | --- |
| MODULE 1Installation de la bibliothèque de vision par ordinateur OpenCV  | Objectifs de ce module :* Installation de la bibliothèque de programmation en vision OpenCV.
* Configuration de Visual Studio 2012.
* Configuration de CodeBlocks avec Kubuntu 12.04.
 |

***Table des matières***

Sujets Page

MODULE 1 1

Installation de la bibliothèque de vision par ordinateur OpenCV 1

Introduction 3

Présentation sommaire d’OpenCV 3

Installation de OpenCV sous Windows 7 3

*Projet avec Visual Studio 2012 sous Windows 7* 6

Test de la configuration sous Visual Studio 2012 8

*Installation de OpenCV sous Linux (Kubuntu 12.04 )* 10

*Projet avec CodeBlocks sous Kubuntu 12.04* 11

Introduction

La vision par ordinateur est un domaine de l’informatique qui est se situe à la croisée des chemins entre les mathématiques, le traitement de signal et l’intelligence artificielle. En fait, c’est pouvoir extraire l’information d’une scène, d’une image ou d’un flux vidéo, de manière à prendre des décisions en fonction des caractéristiques d’un ou de plusieurs objets qui composent l’image. Donc, être capable de discerner et reconnaître les formes, les couleurs, les textures, les contours, etc.

Nous allons donc utiliser une librairie de vision par ordinateur qui se nomme « OpenCV ». OpenCV (**Open** Source **C**omputer **V**ision) est une bibliothèque proposant un ensemble de plus de 2500 algorithmes de vision par ordinateur, accessibles au travers d'API pour les langages C, C++, et Python. Elle est distribuée sous une [licence BSD](http://creativecommons.org/licenses/BSD/) (libre) pour les plate-formes Windows, GNU/Linux, Android et MacOS.

Initialement écrite en C il y a 10 ans par des chercheurs de la société Intel, OpenCV est aujourd'hui développée, maintenue, documentée et utilisée par une communauté de plus de 40 000 membres actifs. C'est **la** bibliothèque de référence pour la vision par ordinateur, aussi bien dans le monde de la recherche que celui de l'industrie.

Afin de mieux vous présenter son étendue et ce qu'elle permet de faire, jetons un œil aux principaux modules accessibles au travers de son API C.

## Présentation sommaire d’OpenCV

* est une librairie open source de traitement et analyse d'images et vidéos avec des interfaces pour les principaux langages de programmation C, C++, Java, C#, Python ...
* est optimisée pour les applications temps réel
* fournit une API bas et haut niveau
* est utilisé aussi bien dans les laboratoires de recherche que dans l'industrie

Fonctions :

* manipulation d'images (chargement, sauvegarde, copie, conversion...)
* manipulation et acquisition de vidéos
* manipulations de matrices et algèbre linéaire
* structure de données utilitaires variées (listes, files, ensembles, graphes...)
* traitement d'images (filtrage, détections de discontinuités, morphologie mathématique...)
* analyse d'images (composantes connexes, ajustement de primitives, transformée de distance...)
* vision (calibration de caméra, stéréovision, recherche d'association...)
* reconnaissance de forme (modèles de Markov, ACP...)
* interface graphique (affichage d'images, de vidéos, gestion des évènements...)

Modules :

* *cv* – fonctions principals d’OpenCV.
* *cvaux* – Fonctions auxiliaires (souvent expérimentale).
* *cxcore* – Structures de donnés et module d’algèbre linéaire.
* *highgui* – fonctions pour l’affichage graphique à l’écran.

Installation de OpenCV sous Windows 7

Tout d’abord, allez chercher la librairie à partir du site Web officiel situé au <http://www.opencv.org>.

* Cliquez sur « Latest Downloads » (Tel que montré par la flèche rouge) 
* Cliquez sur la version 2.4.4 pour Windows.



* Double-cliquez sur le fichier que vous venez de télécharger pour débuter l’installation.
* Choisissez un emplacement pour l’extraction des fichiers.

Vous vous retrouvez avec une hiérarchie de répertoire ressemblant à ceci :



Comme vous pouvez le constater, les répertoires contiennent les éléments de programmation nécessaire à plusieurs plates-formes dont Android, ios et les langages de programmation C/C++.

Le répertoire « build / include » contient deux répertoires. Le répertoire « opencv » contient les fichiers d’entête « .h » pour la compatibilité avec les fonctions d’opencv en langage C alors que le répertoire « opencv2 » contient les fichiers d’entête c++ (les fichiers se terminant par .hpp).

Depuis la version 2.2, la bibliothèque est divisée en plusieurs modules. Ces modules sont disponibles dans le répertoire nommé « lib ». On retrouve, entre autres, les modules suivants :

* opencv\_core contient les fonctionnalités de base et les structures de données et les fonctions arithmétiques,
* opencv\_imgproc contient les principales fonctions de traitement d’image,
* opencv\_highgui contient les fonctions essentielles pour la lecture et l’écriture de fichier d’image et de vidéo.
* opencv\_features2d contient les fonctionnalités pour trouver les caractéristiques reliées à la détection de points et de points d’appariement.
* Opencv\_calib3d permet d’effectuer les calibrations de caméras, les estimations géométriques et les fonctions stéréo.
* opencv\_video contient les fonctionnalités pour effectuer les estimations de mouvement et les classes et fonctions d’extraction
* opencv\_objdetect contient les fonctionnalités de détection d’objet.

Un programme faisant usage de la bibliothèque de fonction C++ commencera par inclure les fichiers d’entête correspondant. Par exemple :

#include <opencv2/core/core.hpp>

#include <opencv2/imgproc/imgproc.hpp>

#include <opencv2/highgui/highgui.hpp>

Donc si le code commence par “#include <cv.h>” ceci est alors un indice que ce programme utilise l’ancienne version de la bibliothèque (< 2.2).

# *Projet avec Visual Studio 2012 sous Windows 7*

Voici les étapes nécessaires à la réalisation d’un projet utilisant la bibliothèque OpenCV et l’environnement de développement Visual Studio version 2012. Le code ne réalisera pas beaucoup de chose au niveau visuel mais nous permettra de tester si notre configuration de Visual Studio est correcte pour un projet OpenCV.

Étape 1 : Démarrez Visual Studio, créez un projet « Application console Win32 » et choisissez un projet vide sans « En-tête précompilé ».

Étape 2 : Dans les propriétés du projet, ajouter les répertoires où se trouve les fichiers d’en-tête « include » et les fichiers de bibliothèque « .lib » pour la compilation et l’édition des liens. Voici un exemple :

**1**

**2**



1. Pour les « Répertoires Include », ajoutez  le répertoire « build\include » qui se situe dans le répertoire où vous avez décompressé les fichiers d’opencv auparavant. Par exemple, dans mon cas, j’ai décompressé les fichiers sur « C:\OpenCV ». J’ai donc ajouté : « C:\OpenCV\opencv\build\include ».
2. Pour les répertoires des bibliothèques, ajoutez le répertoire « build\x86\vc10\lib » situé dans le répertoire où vous avez décompressé les fichiers d’opencv auparavant. Par exemple, dans mon cas, j’ai décompressé les fichiers sur « C:\OpenCV ». J’ai donc ajouté : « C:\OpenCV\opencv\build\x86\vc10\lib ».

Dans la boîte nommée « Dépendances supplémentaires », comme montré ci-dessous, ajoutez les fichiers suivants : **opencv\_core243.lib, opencv\_highgui243.lib et opencv\_imgproc243.lib**



## Test de la configuration sous Visual Studio 2012

#include <iostream>

//Ce fichier d'en-tête contient les éléments et structures de base

#include <opencv2\core\core.hpp>

// Ce fichier d'en-tête contient les éléments nécessaires pour l'affichage à l'écran

#include <opencv2\highgui\highgui.hpp>

using namespace cv; // L'espace de nom pour accéder aux classes

int main()

{

// La classe Mat est la classe principale qui représente une image

 Mat ImgSource;

 return 0;

}

Ajoutez un fichier .cpp vide à votre projet et ajoutez le code suivant :

Compilez et testez le programme.

Vous recevrez probablement une boîte d’erreur ressemblant à ceci :



Cette erreur vient du fait que les fichiers « dll » ne sont pas présent dans le répertoire qui contient l’exécutable du projet.

Copiez les fichiers « opencv\_core243.dll », « opencv\_highgui243.dll » et « opencv\_imgproc243.dll » comme montré ci-dessous :

Rééxecutez le programme.

# *Installation de OpenCV sous Linux (Kubuntu 12.04 )*

Les premières étapes sont très similaires à celles que l’on réalise sous un système de type Windows.

* Tout d’abord, allez chercher la librairie à partir du site Web officiel situé au <http://www.opencv.org>.
* Cliquez sur « Latest Downloads » (Tel que montré par la flèche rouge) 
* Cliquez sur la version 2.4.3 « opencv for Linux/Mac ».

Vous obtenez un fichier se terminant par tar.bz2. C’est donc un fichier compressé contenant tous les fichiers nécessaires. Sous Linux, vous devrez cependant procédé à la compilation de opencv pour votre plate-forme et votre distribution.

* Décompressez le fichier.

tar –jxvf opencv\_version.tar.bz2

Vous obtenez un repertoire “OpenCV-Version” qui contient tous les fichiers nécessaires à la compilation.

Compilation

* Assurez-vous d’avoir téléchargé les paquets suivants avant de procéder :
	+ - cmake et build-essential -> sudo apt-get install build-essential cmake
* Créez un repertoire temporaire pour la compilation des fichiers:
	+ - mkdir OpenCV-Temp
* Déplacez-vous dans ce répertoire et tapez :
	+ - Cmake -D CMAKE\_BUILD\_TYPE=RELEASE -D CMAKE\_INSTALL\_PREFIX=/usr/local .. (Attention les .. sont importants)
* make
* sudo make install

Les fichiers d’entête et les fichiers de bibliothèque sont compilés pour votre plate-forme et se trouvent dans le répertoire que vous avez mentionné pour l’option « CMAKE\_INSTALL\_PREFIX ». Dans notre exemple, le répertoire se trouve dans « /usr/local ».

# *Projet avec CodeBlocks sous Kubuntu 12.04*

On suppose que vous avez déjà compilé OpenCV pour votre plate-forme comme montré dans la section précédente.

* Ouvrir « CodeBlocks » (Si codeblocks n’est pas installé -> sudo apt-get install codeblocks)

Ouvrez un projet de type « Console Application ».

On doit indiquer à codeblocks où se situent les répertoires contenant les fichiers d’en-têtes et de bibliothèque pour pouvoir effectuer la compilation de notre projet OpenCV.

* Indiquez le répertoire des fichiers d’en-tête
* Cliquez Project -> Build Options
* Sous l’onglet « Search Directories », cliquez l’onglet « Compiler » et ajoutez le répertoire « /usr/local/include »
* Sous l’onglet « Linker », ajoutez le répertoire « /usr/local/lib ».
* Sous l’onglet « Linker Settings », ajoutez les fichiers suivants : libopencv\_core, libopencv\_highgui.
* Cliquez le bouton « Valider » pour continuer.

Puisque le code réalisé sous la plate-forme Windows est portable, vous pouvez utiliser le même code dans l’exemple sous Linux.

* Procédez ensuite à la compilation et l’édition des liens sous codeblocks. Tapez la touche « F9 » pour lancer la compilation et exécuter le programme.
* Si vous obtenez « Test réussi ! », tout est alors fonctionnel.