

TP N°=1 Initiation aux outils et environnements de développement



C/C++

Youkana Imane

imane.youkana@univ-biskra.dz

Table des matières



Objectifs	3
Introduction	4
I - Présentation des outils et environnements de développement	5
1. MATLAB	5
1.1. Quelques généralités sur Matlab	5
1.2. Exercice	7
1.3. Exercice	7
2. La bibliothèque OpenCV	7
2.1. L'environnement de développement intégré (EDI) Code::BLOCKS	8
2.2. Configurer OpenCV sous Code::Blocks	10
2.3. Exercice	13
2.4. Exercice	13
2.5. Exercice : Trier les étapes nécessaires pour la configurations de l'Open CV sous Code : :Blocks	14
II - Tester votre compréhension	15
Solutions des exercices	17
Glossaire	19
Abréviations	20
Webographie	21

Objectifs

Ce premier TP est consacré à une initiation aux outils et environnements de développement de traitement d'images,

A la fin de ce premier TP, l'étudiant sera capable de :

- Identifier l'utilité de l'environnement de développement Matlab.
- Comprendre comment utiliser la bibliothèque Open CV.
- Savoir configurer la bibliothèque OpenCV sous Codes : :Blocks.

Pré-requis

Pour pouvoir suivre ce TP avec succès il faut au préalable savoir:

- Des notions de base sur l'utilisation de l'outil informatique et Internet.
- Des notions de base en programmation en langage C/C++.

Introduction



- Le traitement d'images est un domaine très vaste qui a connu depuis les années 60, et qui connaît encore, un développement important depuis quelques dizaines d'années.
- Il permet de modifier une image numérique afin d'améliorer ou d'en extraire des informations.
- Plusieurs outils et environnements sont développés pour le traitement d'image.
- Dans ce premier TP, nous allons présenter quelques bibliothèques graphiques, ainsi que ses mise en œuvre.

Présentation des outils et environnements de développement



MATLAB	5
La bibliothèque OpenCV	7

1. MATLAB

Quelques généralités sur Matlab	5
Exercice	7
Exercice	7

MATLAB^{p.20} ^{AA} est l'abréviation de MATrix LABoratory. C'est un environnement de développement IDE^{p.19} ⁼ dans le domaine du calcul matriciel numérique. Ce logiciel est développé par la société The *MathWorks*

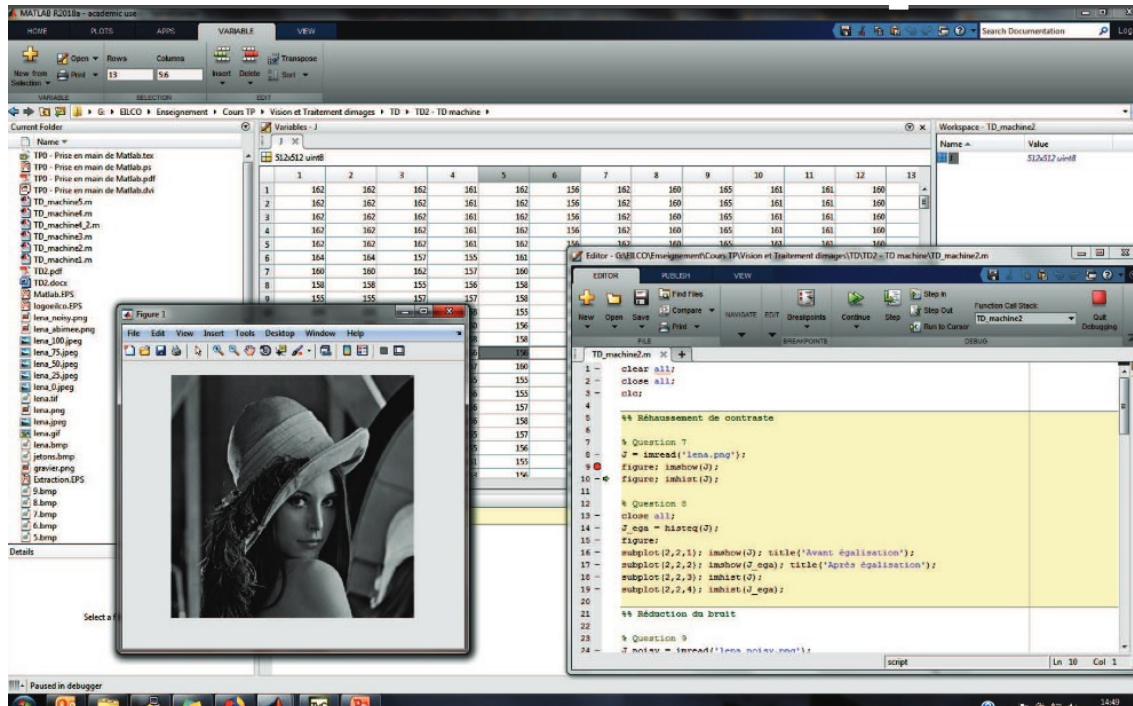
- Il permet de réaliser du calcul numérique et tracer des graphiques pour visualiser et analyser les données.
- Il dispose d'un langage et d'un environnement de programmation interactifs ainsi que d'outils pour concevoir des interfaces utilisateur graphiques.
- *Matlab* est associé à des boîtes à outils appelées toolbox permettant d'accéder à des fonctions spécifiques à un domaine d'application comme le traitement d'images par exemple.

1.1. Quelques généralités sur Matlab

La décomposition de la fenêtre *Matlab* en plusieurs onglets a été montré dans la figure suivante :

- un onglet d'édition des commandes (*Command Window*),
- un onglet de visualisation de l'espace des variables (*Workspace*),
- un onglet de visualisation des fichiers du répertoire de travail (*Current Folder*),
- un onglet de visualisation de l'historique des commandes (*Command History*),

- un onglet permettant de visualiser le contenu de fichiers (*Editor*)
- et un onglet permettant de visualiser le contenu des variables (*Variables*)



- Les fonctions sont éditées dans la fenêtre de commandes et exécutées en appuyant sur la touche: *Entrée*
- Le point virgule à la fin d'une fonction permet d'éviter d'afficher les données résultats de la fonction exécutée ou de séparer plusieurs fonctions sur une même ligne de commande.
- Plusieurs fonctions et commandes peuvent être saisies dans un fichier qui sera enregistré avec l'extension *.m*.
- En éditant le nom de ce fichier dans la fenêtre de commande, l'ensemble des fonctions déclarées dans ce fichier seront exécutées. L'édition de ce fichier peut s'effectuer en sélectionnant dans le menu *File : New → M-File*
- Il est également possible d'y créer des fonctions en utilisant la commande *function*
- La fenêtre *figure* permet de générer une fenêtre graphique permettant de visualiser les données (courbes, images, ...).
- La fonction *clear* efface les variables mises en mémoire durant une session Matlab tandis que la fonction *clc* efface le contenu de la fenêtre de commande.
- Chaque variable déclarée dans *Matlab* est stockée dans l'espace des variables à partir duquel il est possible de consulter la taille et le type de la variable ainsi que d'éditer son contenu par un double-clic sur le nom de la variable.
 - Les variables *Matlab* sont des objets de type structure ou des tableaux à n dimensions.
 - Ainsi, un scalaire est un tableau de taille 1×1 ,
 - un vecteur est un tableau à 1 dimension de taille $1 \times n$,
 - une matrice est un tableau à 2 dimensions de taille $m \times n$, . . .

- Il est ensuite possible d'accéder facilement au tableau, à un élément du tableau ou à une ou plusieurs dimensions particulières du tableau.

1.2. Exercice

[solution n°1 p.17]

Les variables *Matlab* sont :

"Plusieurs réponses sont possibles, choisir et cliquer sur le bouton correction"

- des tableaux à n dimensions.
- un scalaire est un tableau de taille 1×1
- des pointeurs
- un vecteur est un tableau à 1 dimension de taille 1×n

1.3. Exercice

[solution n°2 p.17]

Le code source utilisé dans le matlab est enregistré avec l'extension *.m*

« choisir "vrai" ou "faux", puis cliquer sur le bouton correction »

- Vrai
- Faux

2. La bibliothèque OpenCV

L'environnement de développement intégré (EDI) Code::BLOCKS	8
Configurer OpenCV sous Code::Blocks	10
Exercice	13
Exercice	13
Exercice : Trier les étapes nécessaires pour la configurations de l'Open CV sous Code : :Blocks	14

OpenCV est la librairie de référence pour le traitement d'image. Avec cette librairie de licence open source, vous avez la possibilité d'effectuer des opérations simples sur les images (contraste, rotations, etc.) et les vidéos (estimation de mouvement) mais aussi des opérations complexes comme la détection de formes géométriques, d'objets et de visages, la reconstruction d'une scène en 3 dimensions, et de nombreuses autres fonctions [p.21](#) .

Elle écrite en langage C/C++. La bibliothèque *OpenCV* est proposée par Intel pour les systèmes d'exploitation Windows et Linux.

Cette bibliothèque comprend un grand nombre d'opérateurs, parmi lesquels ^{p.21} ☺ :

- Création/libération d'images, macros d'accès rapide aux pixels
- Opérateurs standards:
 - morphologie,
 - filtres dérivatifs , filtres de contours,
 - suppression de fond
 - recherche de coins
 -
- Recherche, manipulation, traitement de contours
- Dessins de primitives géométriques (lignes,rectangles,ellipses,polygones. . . et même du texte)
- Création et utilisation d'histogrammes
- Changements d'espaces de couleurs (RGB, HSV, L*a*b*et YCrCb)
- Interface Utilisateur : Lecture/ écriture d'images (JPEG, PPM,...)
- Suivi d'objets 3D avec plusieurs caméras
-
- Il existe encore de nombreuses fonctions, structures de données pensez à consulter la *documentation d'OpenCV*

2.1. L'environnement de développement intégré (EDI) Code::BLOCKS

Pour plus d'accessibilité et pour faciliter la tâche de programmation, il est préférable d'utiliser un environnement de développement intégré (EDI^{p.19} ☺). Ils en existent plusieurs tels que NetBeans, Eclipse, Visuel studio, etc.

Dans ce travail nous utiliseront *Code::Blocks*. Dans ce qui suit, nous dériveront l'EDI utilisé, ainsi que les différentes étapes d'installation et de configuration d'*OpenCV* sous *Code::Blocks*.

Comme nous l'avons décrit précédemment, *Code::Blocks* est un environnement de développement libre (Open source), utilisable sur plusieurs plateformes, y compris sous Windows. Il est développé avec le langage de programmation C++, et peut être utilisé avec plusieurs langages tels que Fortron, C#, etc.

Notez que d'autres EDI peuvent être utilisés. Nous avons choisi d'utiliser *Code::Blocks* puisque il est simple, intuitive, et complet.

Afin d'installer *Code::Blocks* il faut juste se rendre sur ce lien : <http://www.codeblocks.org/downloads/binaries>

Dans la page qui s'affiche, on choisi la version qui correspond à notre configuration. Dans notre

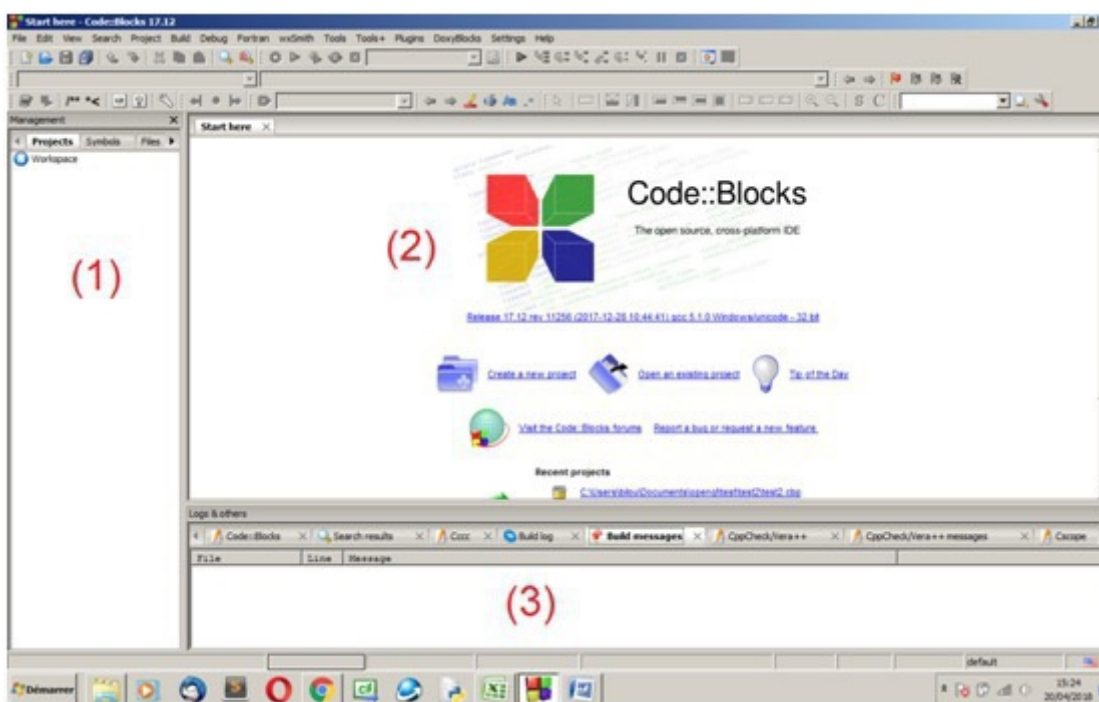
cas la version 32 bits de Windows :

File	Date	Download from
codeblocks-17.12-setup.exe	30 Dec 2017	Sourceforge.net
codeblocks-17.12-setup-nonadmin.exe	30 Dec 2017	Sourceforge.net
codeblocks-17.12-nosetup.zip	30 Dec 2017	Sourceforge.net
codeblocks-17.12mingw-setup.exe	30 Dec 2017	Sourceforge.net
codeblocks-17.12mingw-nosetup.zip	30 Dec 2017	Sourceforge.net
codeblocks-17.12mingw_fortran-setup.exe	30 Dec 2017	Sourceforge.net

Page de téléchargement de Code::Blocks

Notez qu'on doit choisir la version avec le compilateur *MinGW* (La ligne entourée en rouge dans la figure ci-dessus).

Après avoir téléchargé *Code::Blocks*, et l'installé, son lancement affiche la fenêtre principale suivante :



Fenêtre principale de Code ::Blocks

Cette fenêtre contient trois zones principales :

- La zone à gauche (1): elle contient la liste des fichiers sources dans un projet.
- La zone principale (2) : c'est la zone dans laquelle on ouvre les fichiers sources contenant nos programmes pour leur édition (éditeur de texte).
- La zone (3) : les différents messages de notifications, notamment ceux de la compilation, s'affichent dans cette zone.

Code::Blocks comme la plupart des EDI offre les trois composants principaux pour développer un programme informatique :

- Un éditeur de texte : c'est pour écrire le code source du programme. C'est la zone (2) dans Code::Blocks.
- Un compilateur : c'est l'outil qui permet de transformer un code source en code binaire (suite d'instructions compréhensible par l'ordinateur). Il y en a plusieurs dans Code::Blocks: GCC, MingW, GNU Fortran, etc.
- Un débogueur : Il sert à localiser les erreurs de programmation dans un code source, et de les résoudre

L'étape suivante consiste à configurer OpenGL pour qu'il soit utilisable avec Code::Blocks.

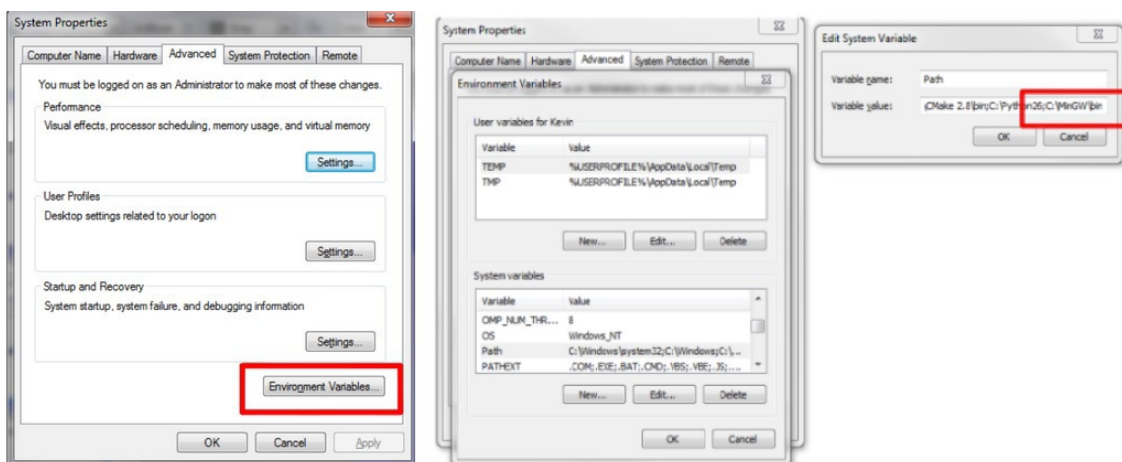
2.2. Configurer OpenCV sous Code::Blocks

La bibliothèque OpenCV peut être utilisée avec l'environnement de développement libre CodeBlocks *p.21*

Donc, pour qu'on puisse utiliser la bibliothèque *OpenCV*, nous allons devoir ajouter les fichiers nécessaires pour l'exécution de notre programme.

A cet effet, les différentes étapes nécessaires pour la configuration d'*OpenCV* sous *Code::Blocks* sont les suivantes :

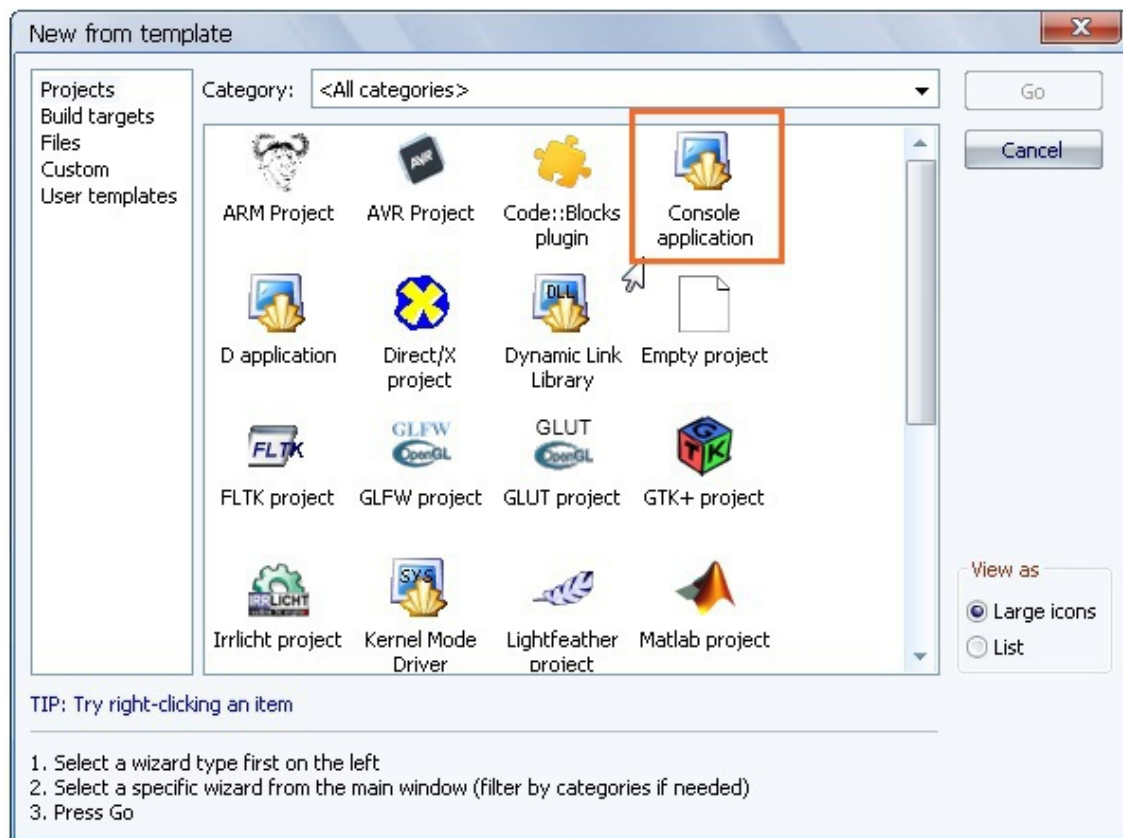
- Télécharger et Installer : minGW^{p.19} depuis : <http://sourceforge.net/projects/mingw/files/>
- Mettre minGW^{p.19} vers le chemin du système :



Configuration du minGW

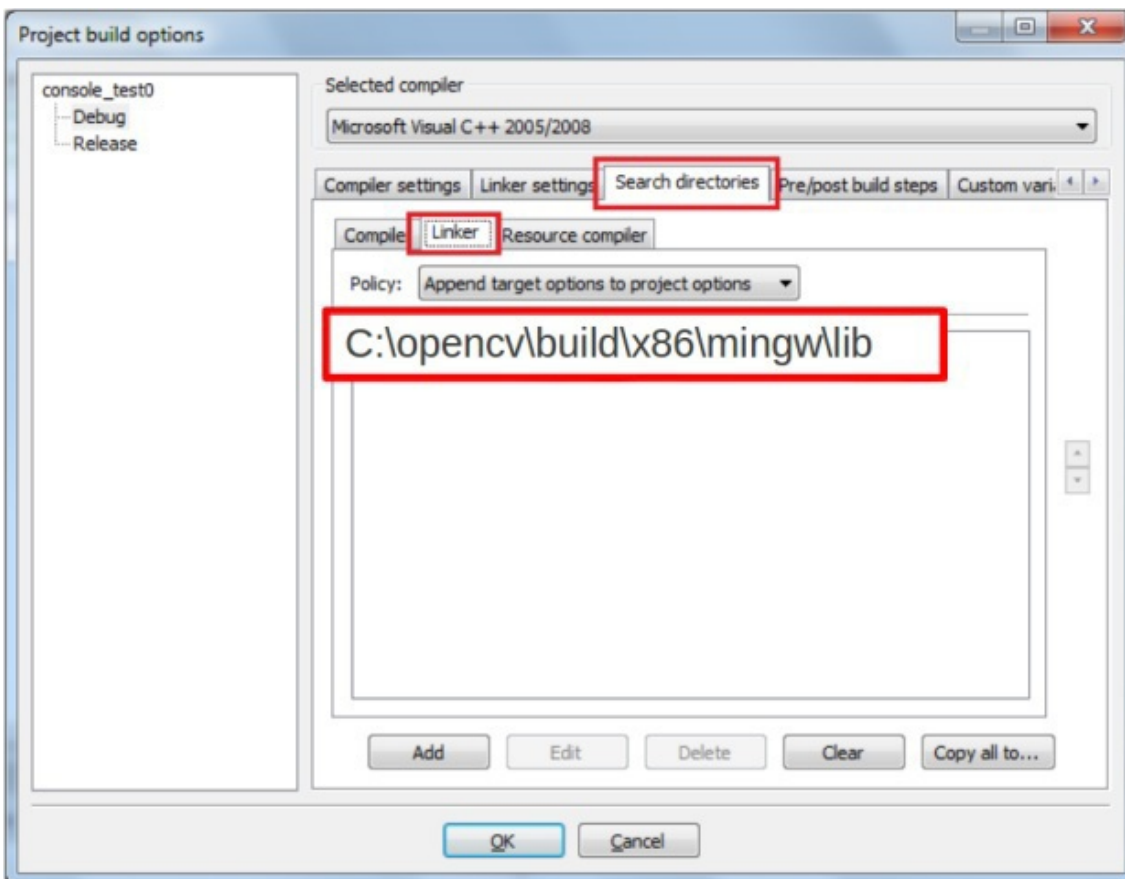
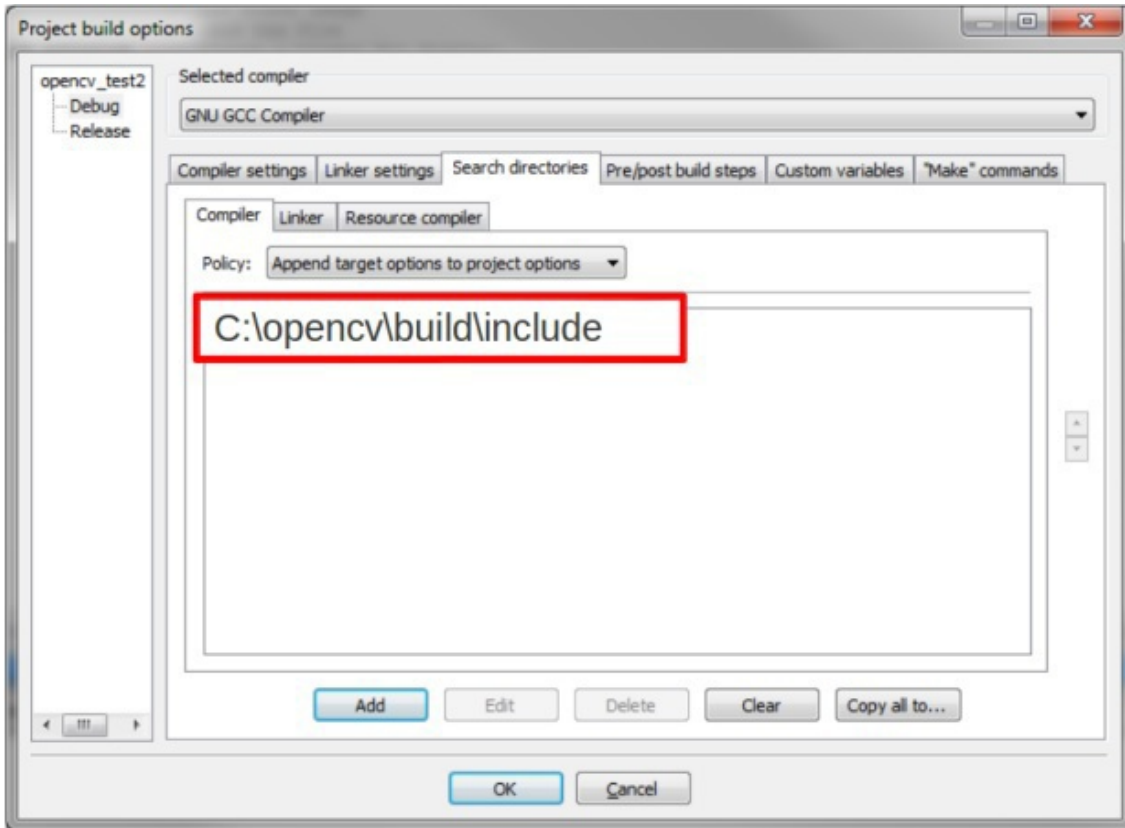
- Installer le *Code::Blocks* depuis le lien : <http://www.codeblocks.org/downloads/binaries>
 - Ouvrir le *Code::Blocks* et aller vers : *Settings* → *Compiler and Debugger*
 - sous la fenêtre *the Toolchain Executables* choisit *GNU GCC Compiler* depuis la fenêtre on bas et clique sur *AutoDetect*
 - Vérifier que *Code::Blocks* a détecté *MinGW*
- Télécharger *OpenCV* sur : <http://sourceforge.net/projects/opencvlibrary/>

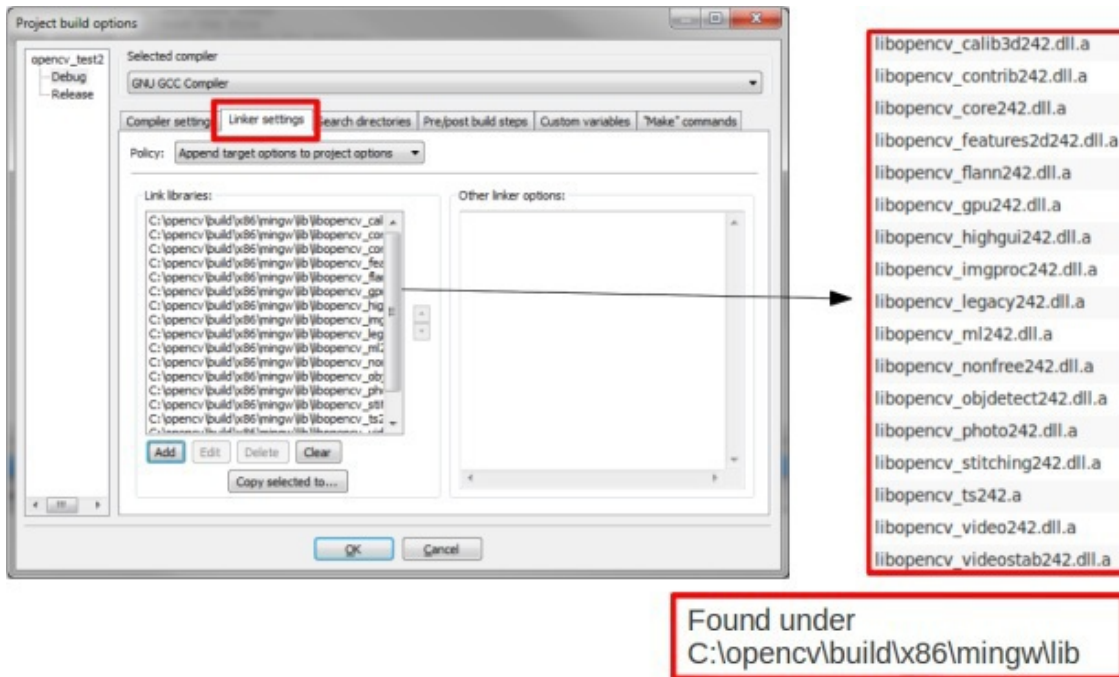
- Ouvrir un nouveau projet dans le *Code::Blocks*



Code::Blocks

- Faire un clic droit sur votre projet et choisissez build options et appliquer les changements montrés dans les figures suivantes:





Maintenant vous pouvez tester un simple programme OpenCV "Hello World" et ça marchera.
 Vous pouvez aussi regarder le vidéo suivant pour la configuration *OpenCV* → *Code::Blocks*
 Cf. "Instalacion OpenCV : CMake + Mingw + Codeblocks"

2.3. Exercice

[solution n°3 p.17]

La bibliothèque *OpenCV* est proposée par Intel pour les systèmes d'exploitation
 "Plusieurs réponses sont possibles, choisir et cliquer sur le bouton correction"

- Windows
- Linux.
- MacOS

2.4. Exercice

[solution n°4 p.17]

L'environnement de développement *Code::Blocks* contient :
 "plusieurs réponses sont possibles, choisir et cliquer sur le bouton correction"

- un compilateur
- un éditeur de texte
- Aucun de deux

2.5. Exercice : Trier les étapes nécessaires pour la configurations de l'Open CV sous Code : :Blocks

[solution n°5 p.18]

Installer le Code::Blocks

Mettre minGW vers le chemin du système

Faire un clic droit sur votre projet et choisissez build options et appliquer les changements nécessaires

Télécharger et Installer : minGW

Ouvrir un nouveau projet dans le Code::Blocks

Installer l'OpenCV

Tester votre compréhension



Exercice

L'environnement de développement `Code::Blocks` contient :

"plusieurs réponses sont possibles, choisir et cliquer sur le bouton correction"

- Un éditeur de texte
- Un compilateur
- Aucun de deux

Exercice

`Matlab` est un :

"plusieurs réponses sont possibles, choisir et cliquer sur le bouton correction"

- Langage de programmation
- Environnement de développement intégré IDE
- Aucun de deux

Exercice : Lisez le texte et remplissez les trous avec les mots corrects

La fenêtre permet de générer une fenêtre graphique permettant de visualiser les données (courbes, images, ...).

La fonction efface les variables mises en mémoire durant une session Matlab

la fonction efface le contenu de la fenêtre de commande.

Exercice

La bibliothèque OpenCV est utilisée pour analyser les images numériques:

« "choisir "vrai" ou "faux", puis cliquer sur le bouton correction" »

- vrai
- faux



Exercice

Dans ce cours la bibliothèque *OpenCV* est utilisée avec le langage de programmation :

Solutions des exercices



> Solution n° 1

Exercice p. 7

Les variables *Matlab* sont :

"Plusieurs réponses sont possibles, choisir et cliquer sur le bouton correction"

- des tableaux à n dimensions.
- un scalaire est un tableau de taille 1×1
- des pointeurs
- un vecteur est un tableau à 1 dimension de taille 1×n

> Solution n° 2

Exercice p. 7

Le code source utilisé dans le matlab est enregistré avec l'extension *.m*

« choisir "vrai" ou "faux", puis cliquer sur le bouton correction »

- Vrai
- Faux

> Solution n° 3

Exercice p. 13

La bibliothèque *OpenCV* est proposée par Intel pour les systèmes d'exploitation

"Plusieurs réponses sont possibles, choisir et cliquer sur le bouton correction"

- Windows
- Linux.
- MacOS

> **Solution n°4**

Exercice p. 13

L'environnement de développement *Code::Blocks* contient :

"plusieurs réponses sont possibles, choisir et cliquer sur le bouton correction"

- un compilateur
- un éditeur de texte
- Aucun de deux

> **Solution n°5**

Exercice p. 14

Télécharger et Installer : minGW

Mettre minGW vers le chemin du système

Installer le Code::Blocks

Installer l'OpenCV

Ouvrir un nouveau projet dans le Code::Blocks

Faire un clic droit sur votre projet et choisissez build options et appliquer les changements nécessaires

Glossaire



IDE (EDI)

EDI est un logiciel regroupant un ensemble d'outils nécessaires au développement des applications et des programmes dans un langage de programmation. Exemples d'outils: un éditeur de texte, un compilateur, un débogueur, ...

Exemples de EDI : Matlab, visuel studio, Code::Blocks, Borland c++...

MinGW

MinGW est un compilateur c/c++ pour windows



Webographie



<https://www.captronic.fr/Traitement-d-image-avec-OpenCV,1912.html>

<http://www.intel.com/research/mrl/research/opencv/overview.htm>

<http://turrier.fr/articles/programmer-avec-codeblocks-et-opencv/programmer-avec-codeblocks-et-opencv.php>

<https://fr.mathworks.com/products/matlab.html>

