Computer Vision

Catalin Stoean
catalin.stoean@inf.ucv.ro
http://inf.ucv.ro/~cstoean

Informatii generale

- Pagina web a cursului (in curs de completare)
  - http://inf.ucv.ro/~cstoean/courses/cv/
- Nota
  - Se obtine in urma prezentarii unui proiect insotit de un referat.
  - Teme de proiect vor fi enuntate pe parcursul cursului si cuprind:
    - in principal, procesari de imagini
    - dar si extrageri de informatii din imagini (masuratori, calcule etc) folosind sau nu invatare automata

Bibliografie 1/2

- Robert Laganière, OpenCV 2 Computer Vision Application Programming Cookbook, Packt Publishing, Birmingham, UK, 2011.
- Gary R. Bradski, Vadim Pisarevsky, Jean-Yves Bouguet, Open Source Computer Vision Library, Springer, 1st ed. 2006.
- Gady Agam, Introduction to programming with OpenCV, Illinois Institute of Technology, 2006, http://www.cs.iit.edu/~agam/cs512/lect-notes/opencv-intro/opencv-intro.html

Bibliografie 2/2

- D.A. Forsyth, Jean Ponce, Computer Vision - A Modern Approach (2nd Edition), Jean Ponce, 2011.
- Gary Bradski and Adrian Kaehler, Learning OpenCV: Computer Vision with the OpenCV Library, O'Reilly Media, 2008. http://www.cse.iitk.ac.in/users/vision/dipakmj/papers/OReilly%20Learning%20OpenCV.pdf .
- Peter Corke, Robotics, Vision & Control, Springer 2011.

Continutul cursului

- Computer Vision cu OpenCV
- Încărcarea, afisarea si salvarea imaginilor
  - Crearea unei aplicatii GUI folosind QT pentru procesare de imagini
- Accesarea valorilor pentru pixeli din cadrul unei imagini
- Definirea de regiuni de interes in imagini
- Procesare de imagini cu clase
- Histograma unei imagini
- Detectarea continutului unei imagini folosind histograma
- Transformarea imaginilor cu operatii morfologice
- Extragerea de linii, contururi si componente
- Detectarea de puncte de interes
- Procesarea de secvente video

Ce este Computer Vision?

- Transformarea datelor de la o camera foto sau video intr-o reprezentare noua sau chiar in decizii.
  - Camera poate fi montata pe o masina
  - Un laser poate indica faptul ca te apropii mai aproape de 1 metru de un obiect
- Aceste observatii se fac in general simplu, intruitiv de catre om.

Ce este Computer Vision?

- Este banal pentru un om sa identifice o masina intr-o poza
  - S-a focusat in imagine doar pe acea regiune
  - A vazut deja suficiente masini anterior ca sa aiba o reprezentare clara asupra lor

Dar o masina vede:

194	210	201	212	199	213	215	195	178	158	182	209
180	189	190	221	209	205	191	167	147	115	129	163
114	126	140	188	176	165	152	140	170	106	78	88
87	103	115	154	143	142	149	153	173	101	57	57
102	112	106	131	122	138	152	147	128	84	58	66
94	95	79	104	105	124	129	113	107	87	69	67
68	71	69	98	89	92	98	95	89	88	76	67
41	56	68	99	63	45	60	82	58	76	74	65
20	41	69	75	56	41	51	73	55	70	63	44
50	50	57	69	75	75	73	74	53	68	59	37
72	59	53	66	84	92	84	74	57	72	63	42
67	61	58	65	75	78	76	73	59	75	69	50

De ce Computer Vision?

- Imagini (si filme) sunt pretutindeni
- Aplicatii utile care sa extraga informatii din imagini:
  - Identificarea automata a numarului de la masina
  - Identificarea feței
  - Identificarea unor regiuni de interes intr-o imagine
  - Procesarea filmelor
- Exista deja multe soft-uri care face astfel de procesari.
  - Dar cand avem nevoie sa procesam zeci, sute sau chiar mii de imagini/filme, este esential sa ne cream propria aplicatie care sa realizeze aceste lucruri.

OpenCV

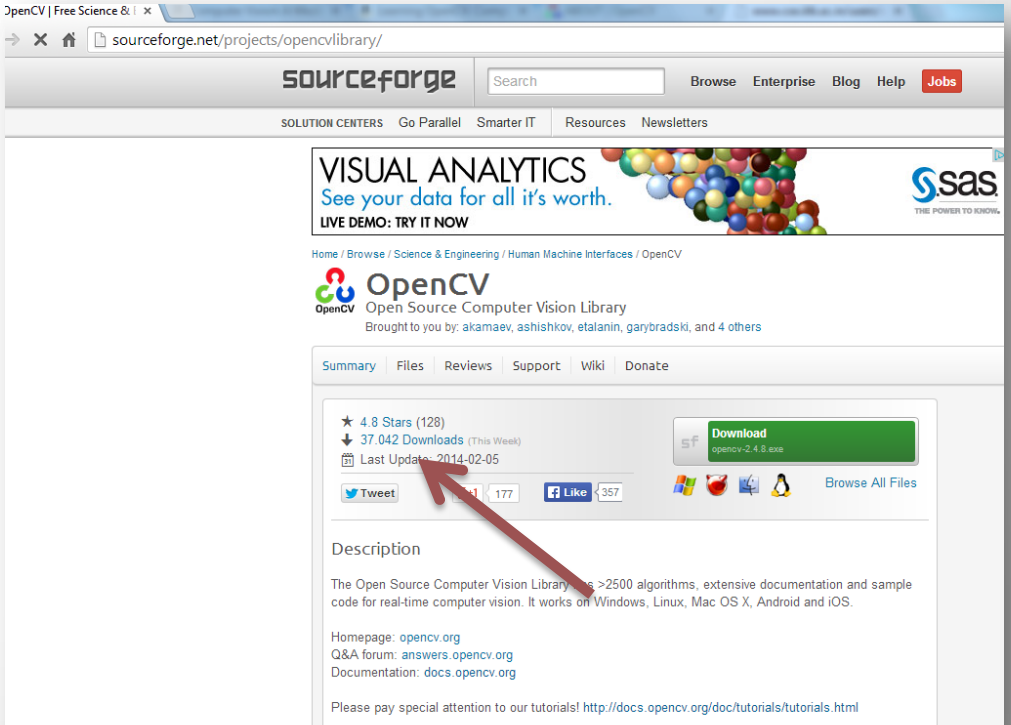
- Reprezinta o librarie gratuita (Open) pentru dezvoltare si cercetarea in Computer Vision
  - http://sourceforge.net/projects/opencvlibrary/
- Contine peste 2500 de algoritmi
- Functioneaza sub Windows, Linux, Android, Mac OS.
- Sunt dezvoltate interfete pentru alte limbaje precum: C++, C, Java, Python, Matlab.
- Ofera infrastructura pentru Computer Vision pentru a construi rapid aplicatii sofisticate
- Cursul ne va ajuta sa alegem ce algoritm sa utilizam pentru scopul avut si in ce moment

OpenCV

- Are peste 7 mil de descarcari
- Printre utilizatori se numara si companii mari precum Google, Yahoo, Microsoft, Intel, IBM, Sony etc.
- Printre algoritmii continuti sunt unii dedicati pentru:
  - Recunoasterea feței
  - Identificarea de obiecte
  - Urmarirea obiectelor in miscare
  - Gasirea de imagini similare intr-o baza de date cu imagini
  - Eliminarea ochilor rosii din poze
  - Urmarirea ochilor in miscare
- Functioneaza sub diferite medii de dezvoltare integrate (IDE) pentru C++.

Descarcarea OpenCV

- **Presupunem instalat deja Visual Studio 2010**
  - Gratuit prin contul personal de la DreamSpark http://e5.onthehub.com/d.ashx?s=bc81baqimt
- Cea mai recenta versiune de OpenCV se descarca de la http://sourceforge.net/projects/opencvlibrary/
  - Arhiva se extrage intr-un folder, de exemplu in D:\OpenCV2

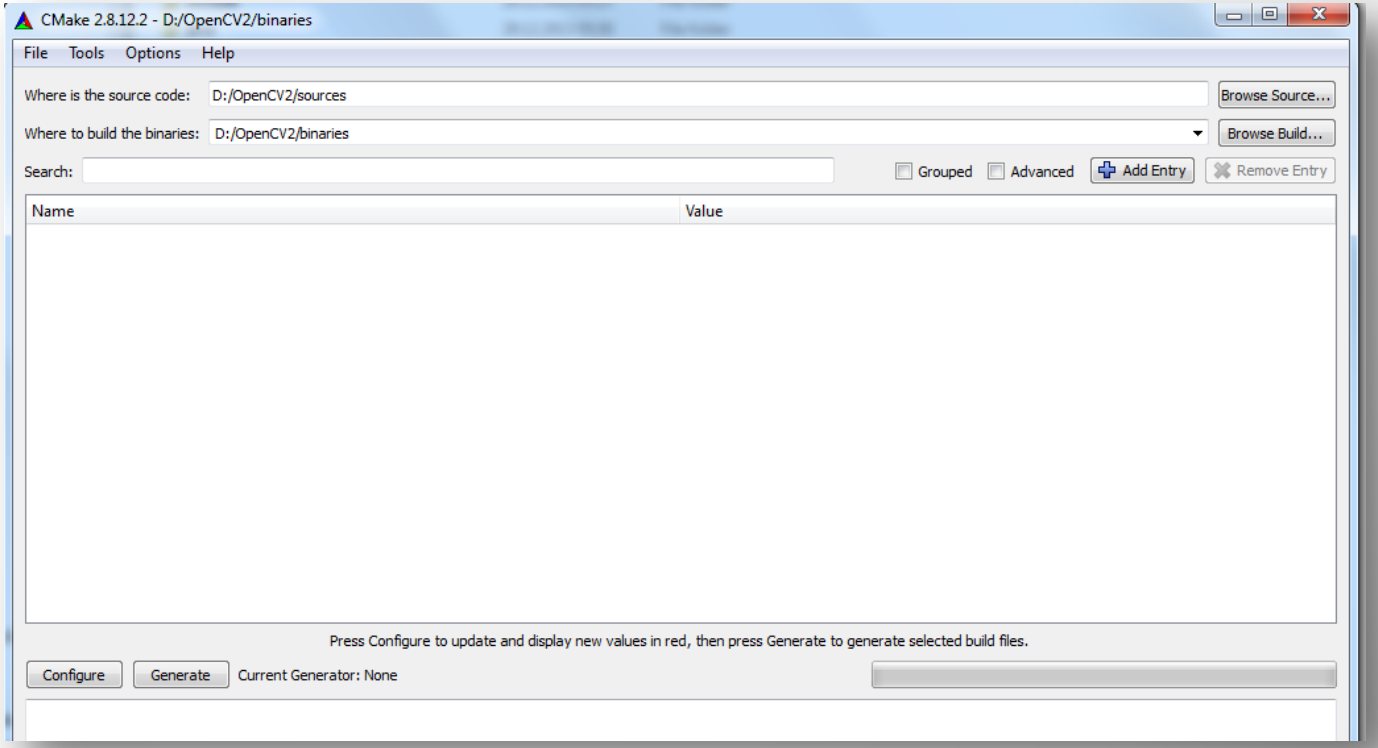


Instalarea OpenCV

- Dupa descarcare, libraria se instaleaza
- Pentru aceasta, descarcati CMake (open-source)

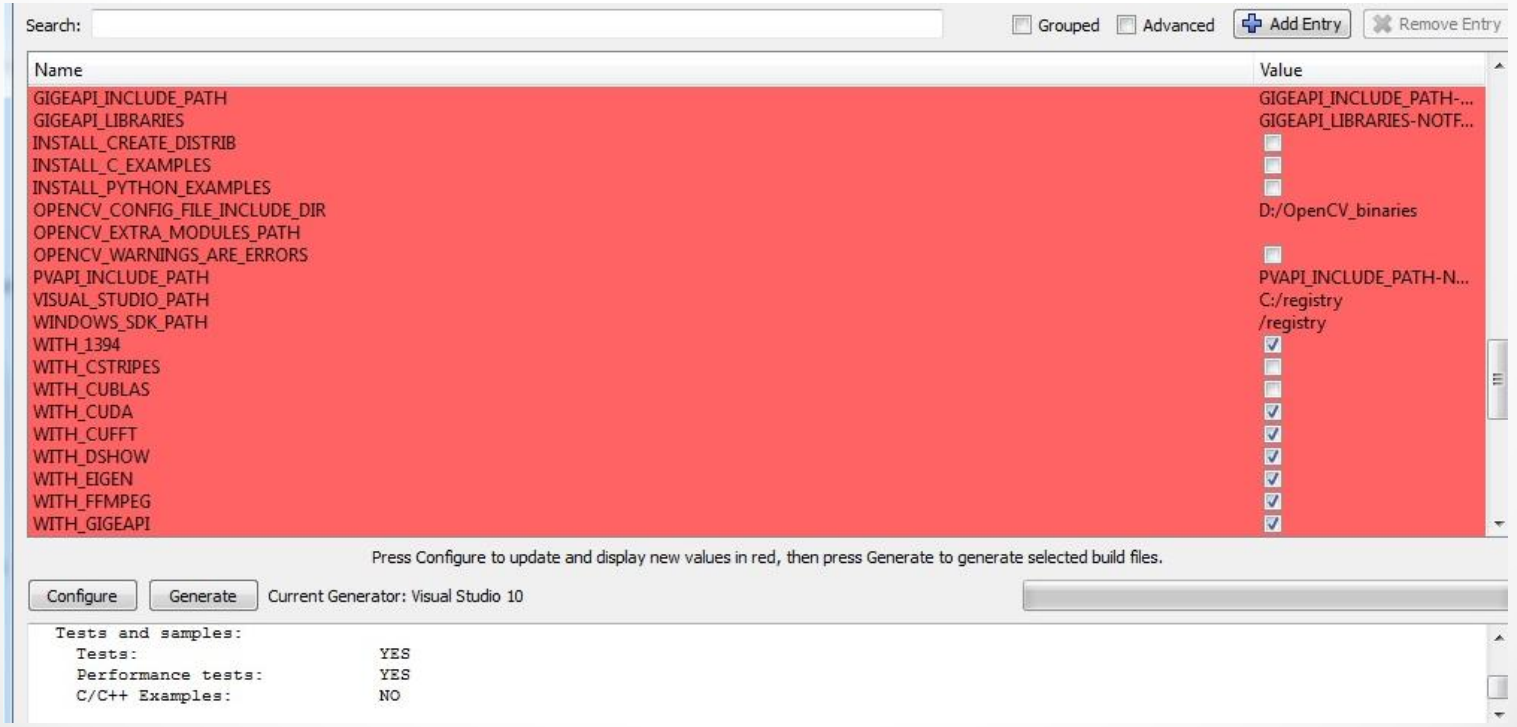
Instalarea OpenCV

- Dupa descarcare, libraria se instaleaza
- Pentru aceasta, descarcati si instalati CMake (open-source)
- Se pun:
  - Calea catre sursa
  - Calea catre libraria compilata
- Configure
  - Se alege Visual Studio 10
- Generate



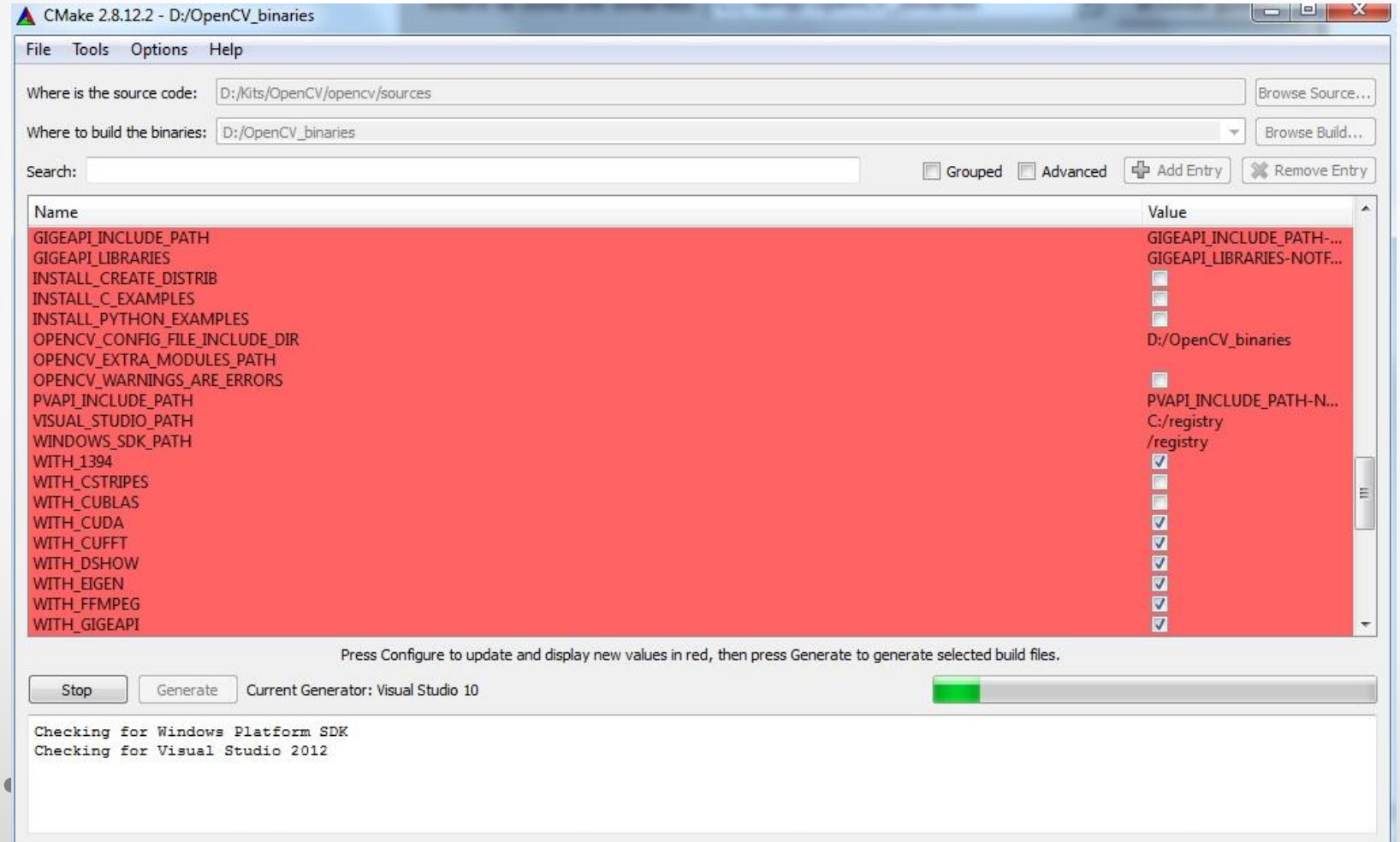
Instalarea OpenCV

- Se obtine o fereastra ca mai jos.
- Apasam din nou **Configure.**



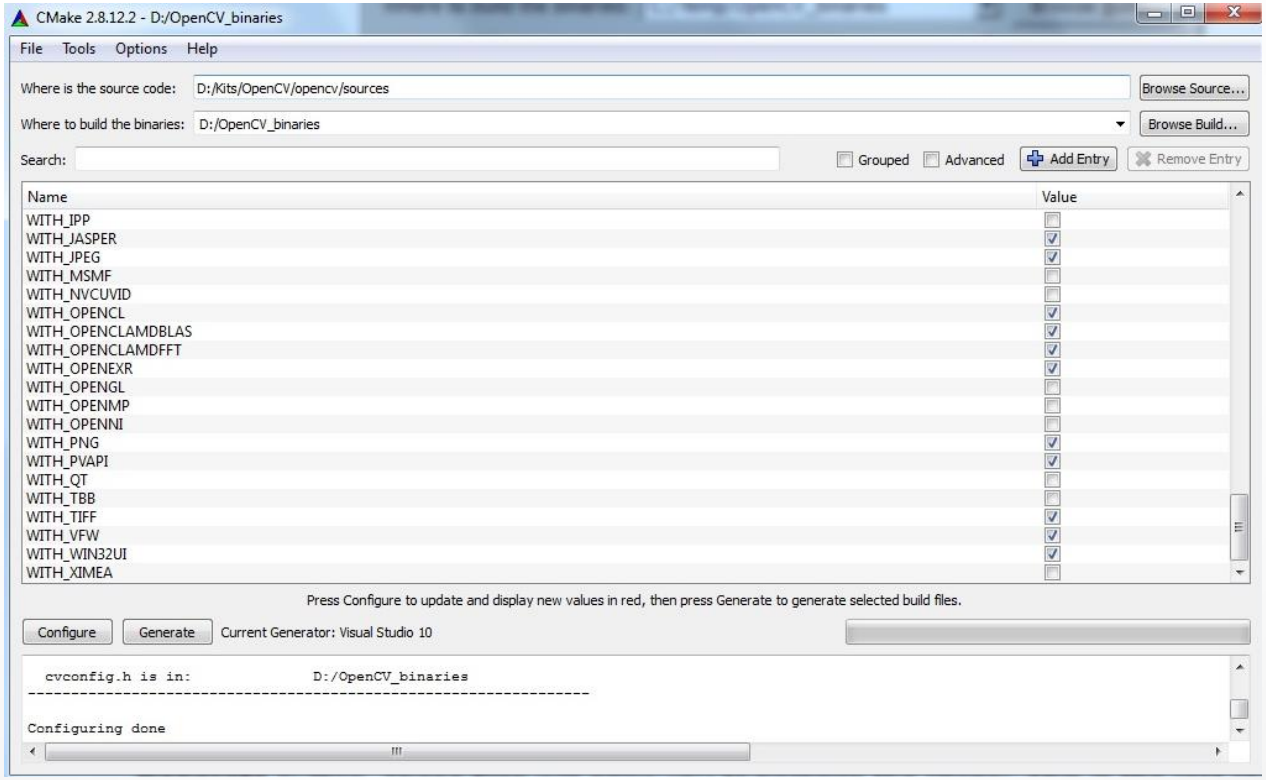
Instalarea OpenCV

- Dupa ce a fost apasat din nou **Configure**



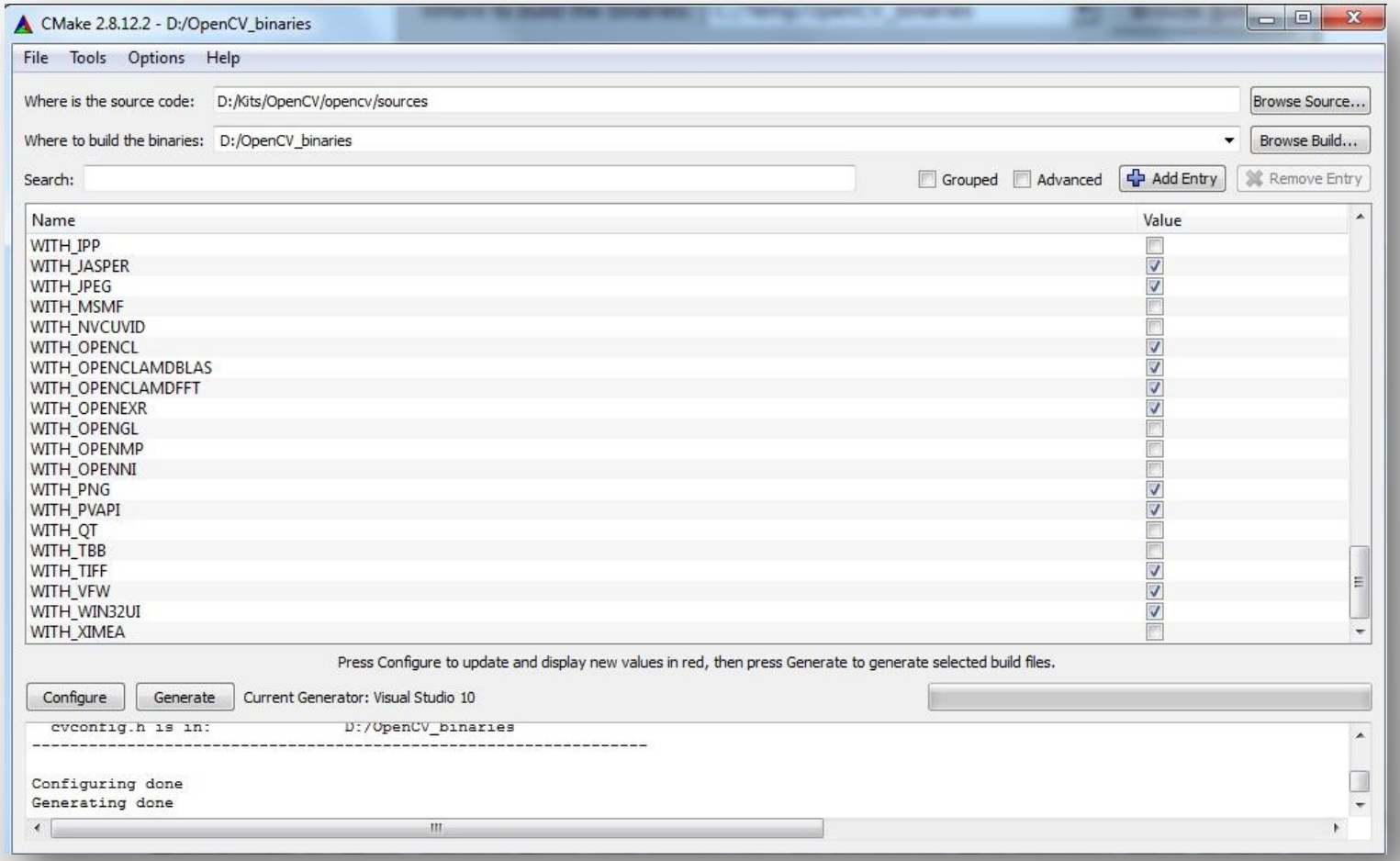
Instalarea OpenCV

- Configurarea este gata. Acum apasam din nou **Generate**



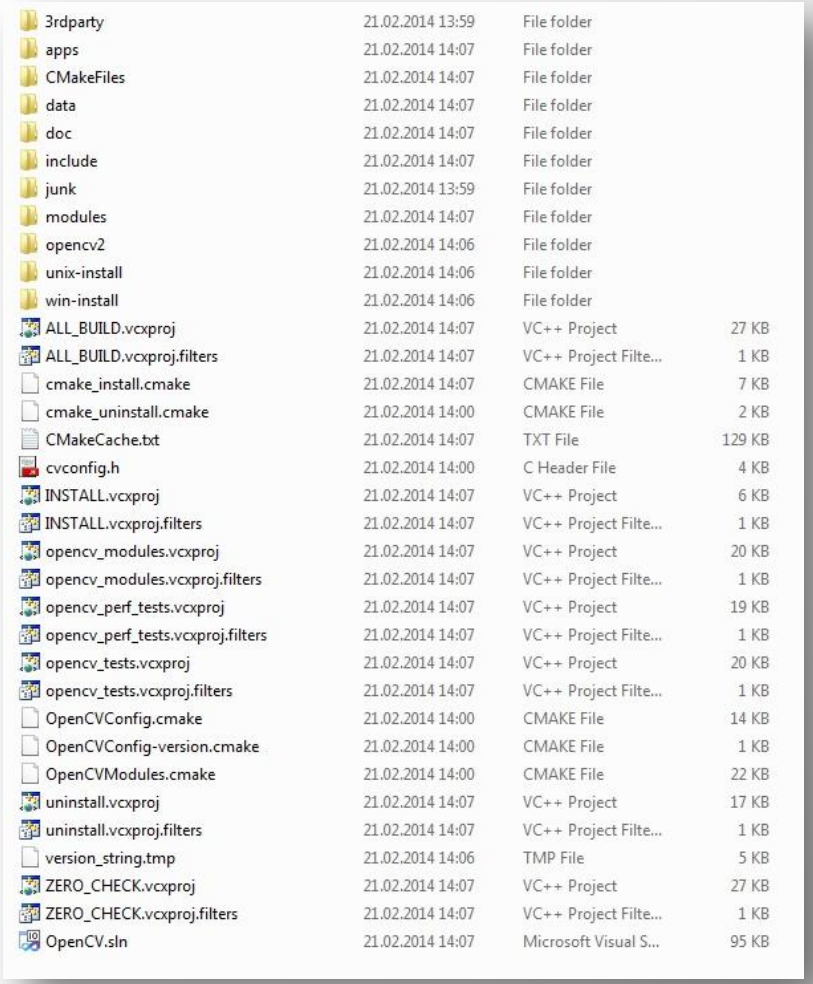
Instalarea OpenCV

- Gata si generarea



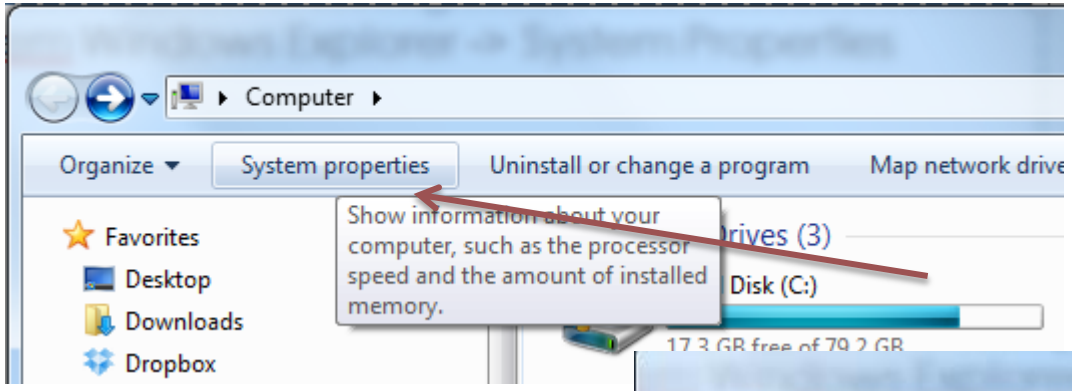
Instalarea OpenCV

- Verificam folderul in care am trimis libraria
  - Dam dublu-click pe OpenCV.sln (solutia creata) pentru a o deschide in Microsoft Studio 2010
  - Dam Build Solution
    - Dureaza peste 5 minute

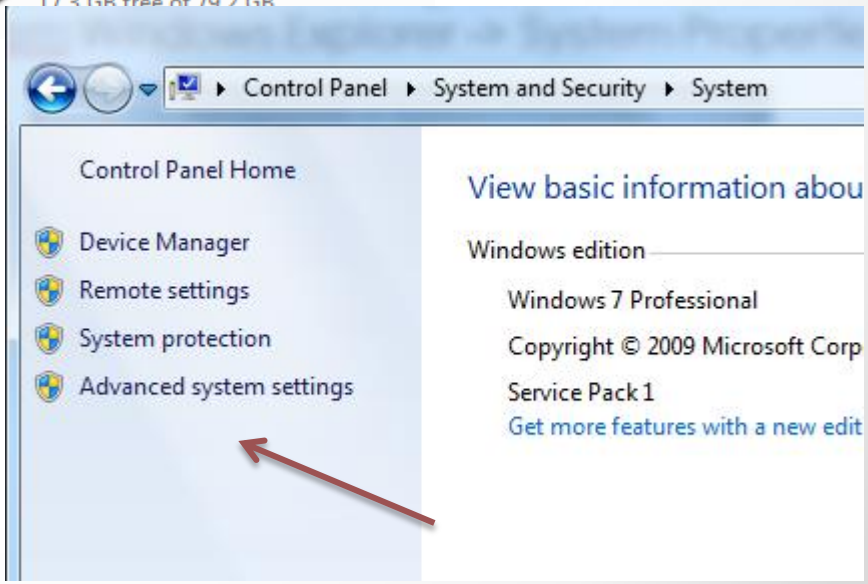


Instalarea OpenCV
Setare variabile de mediu

- Alegem Windows Explorer -> System Properties

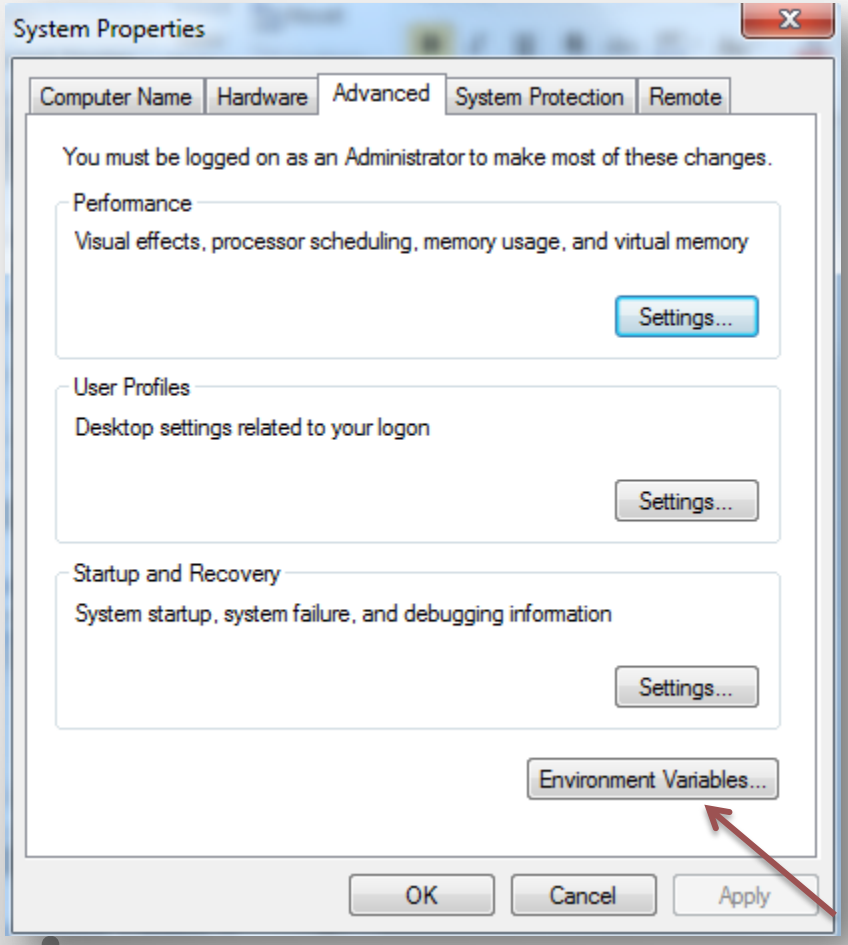


- Apoi **Advanced system settings**

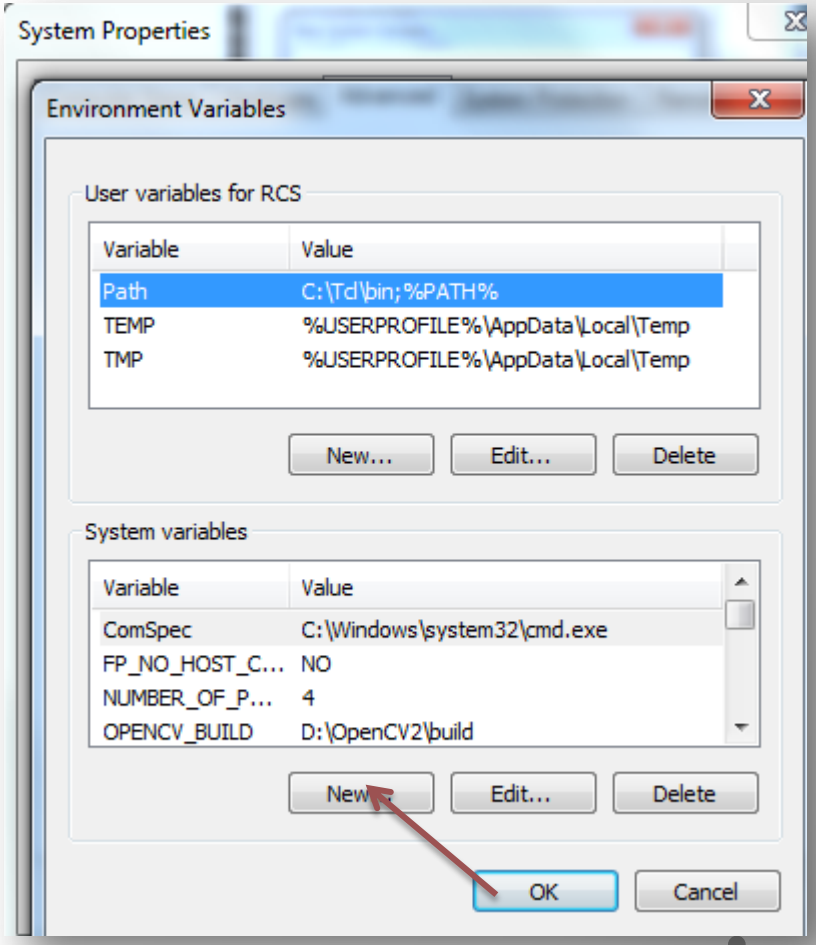


Instalarea OpenCV
Setare variabile de mediu

- Environment Variables



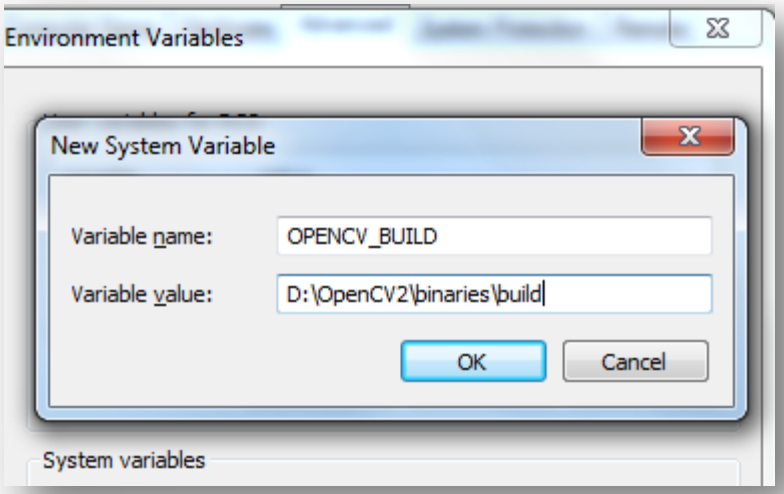
- Cream o variabila de sistem



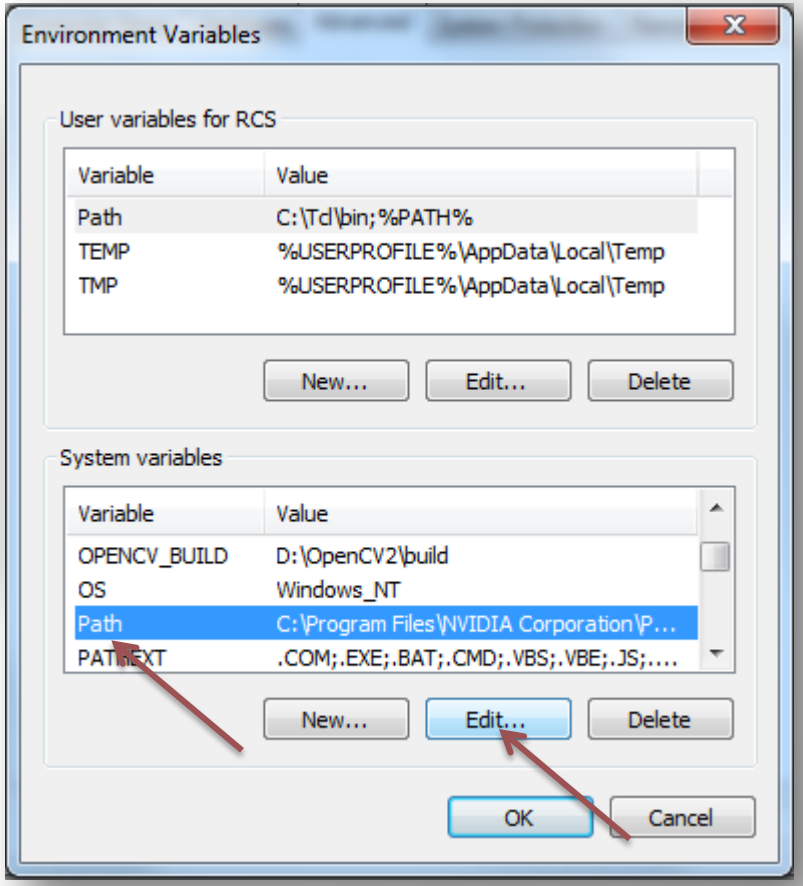
Instalarea OpenCV
Setare variabile de mediu

- Environment Variables

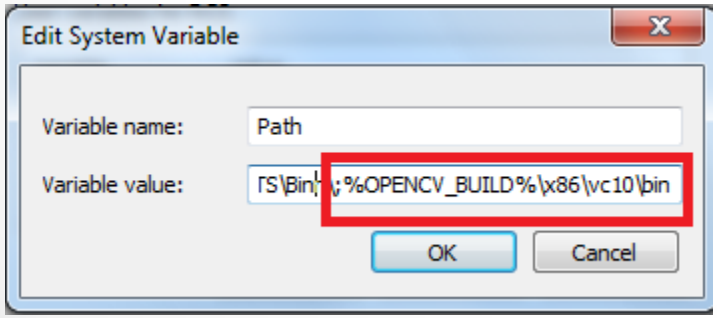
1



2



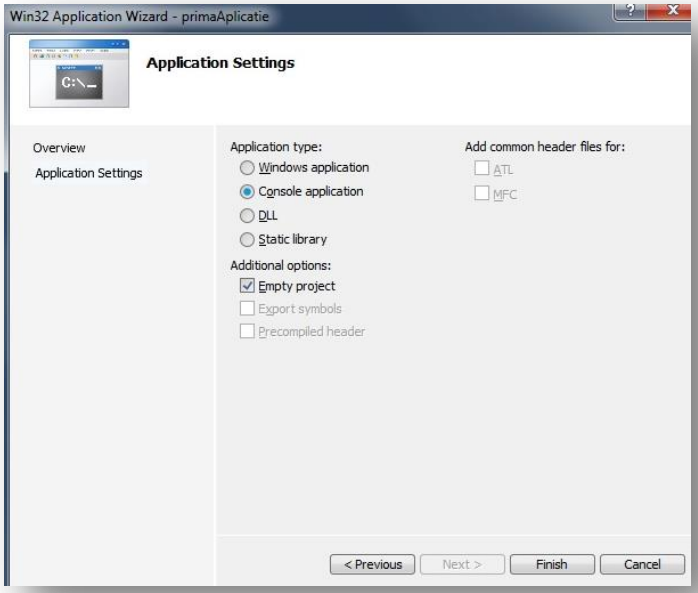
3



- Daca sistemul este pe 64 de biti, se alege folderul x64 in loc de x86.

Proiect OpenCV folosind Visual Studio 10

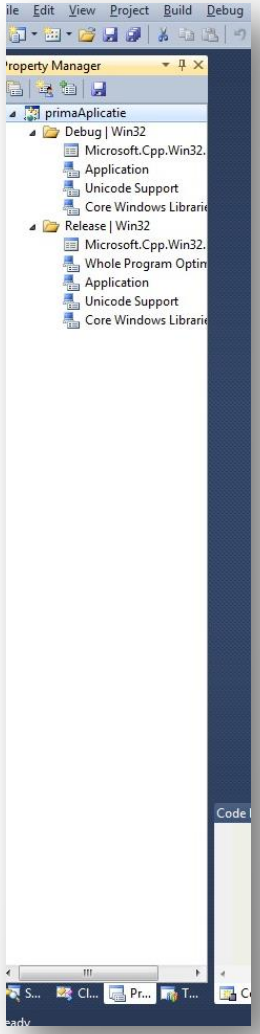
- Cream un proiect de tip **Console Application**, **Empty project**, fara precompiled header.



- In continuare, trebuie sa specificam unde se gasesc librariile OpenCV.

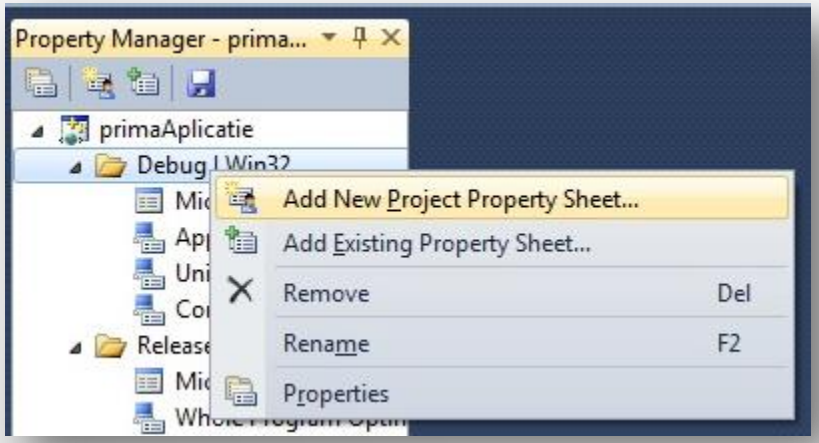
Proiect OpenCV folosind Visual Studio 10

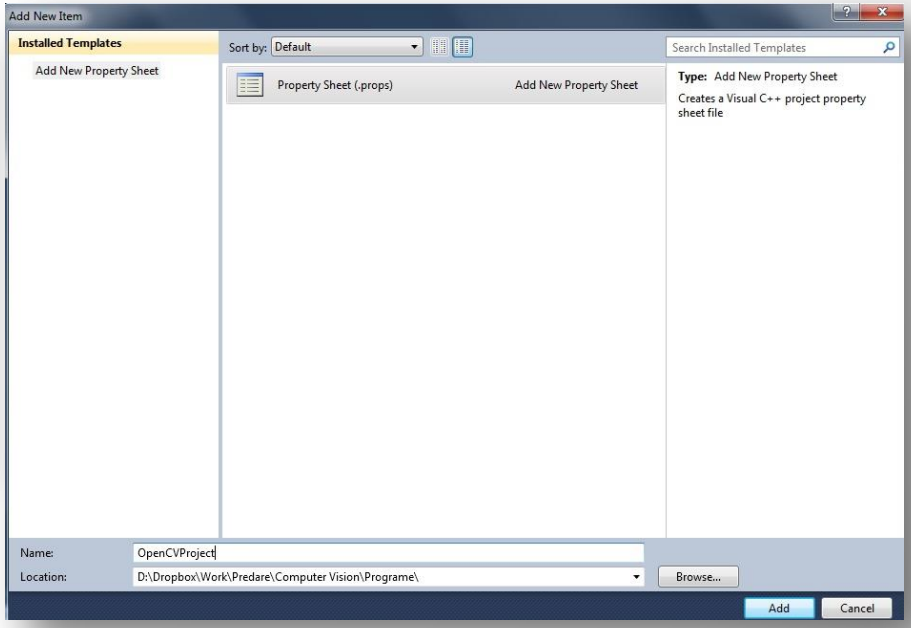
- Cea mai buna optiune pentru a specifica unde se gasesc librariile OpenCV este sa cream un **Property Sheet** pe care sa il putem utiliza si in alte proiecte.
- Pentru aceasta, mergem la Property Manager.
  - Avem de adaugat cate unul pentru **Debug** si **Release**.



Property Sheet

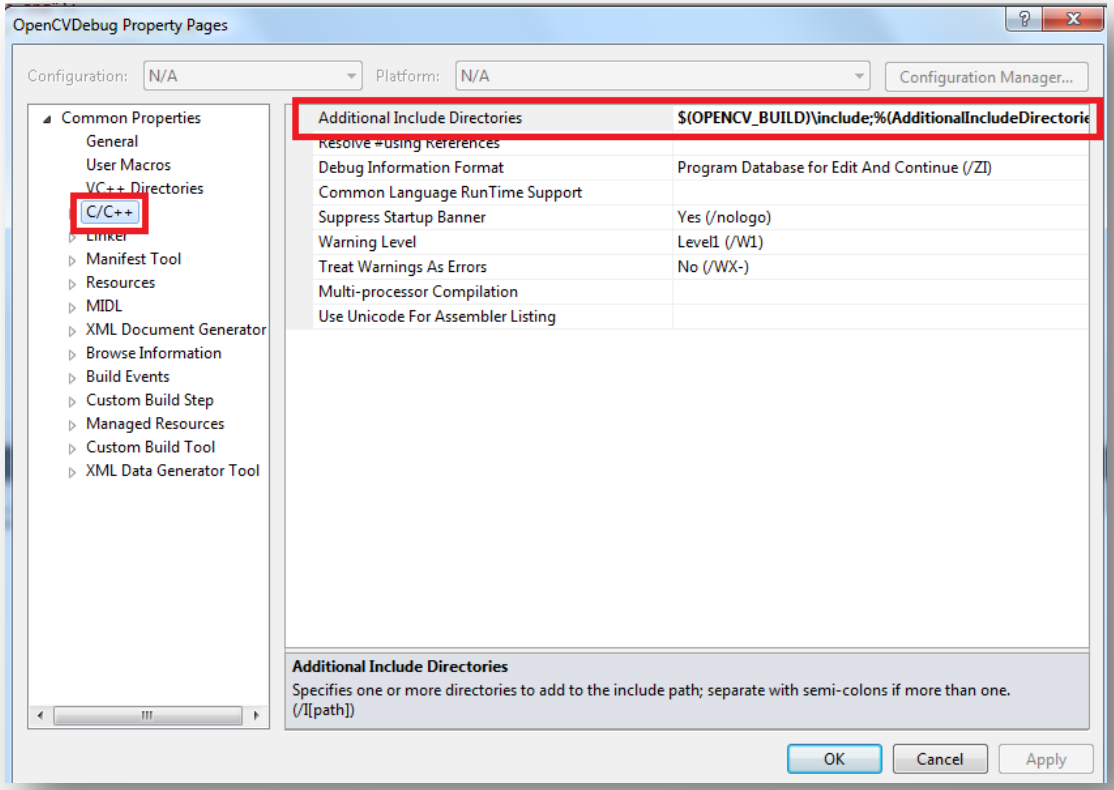
- Click-dreapta pe Debug si selectam **Add New Property Sheet…**
- In fereastra care se deschide ii punem un nume, de exemplu **OpenCVDebug**, apoi **Add**.





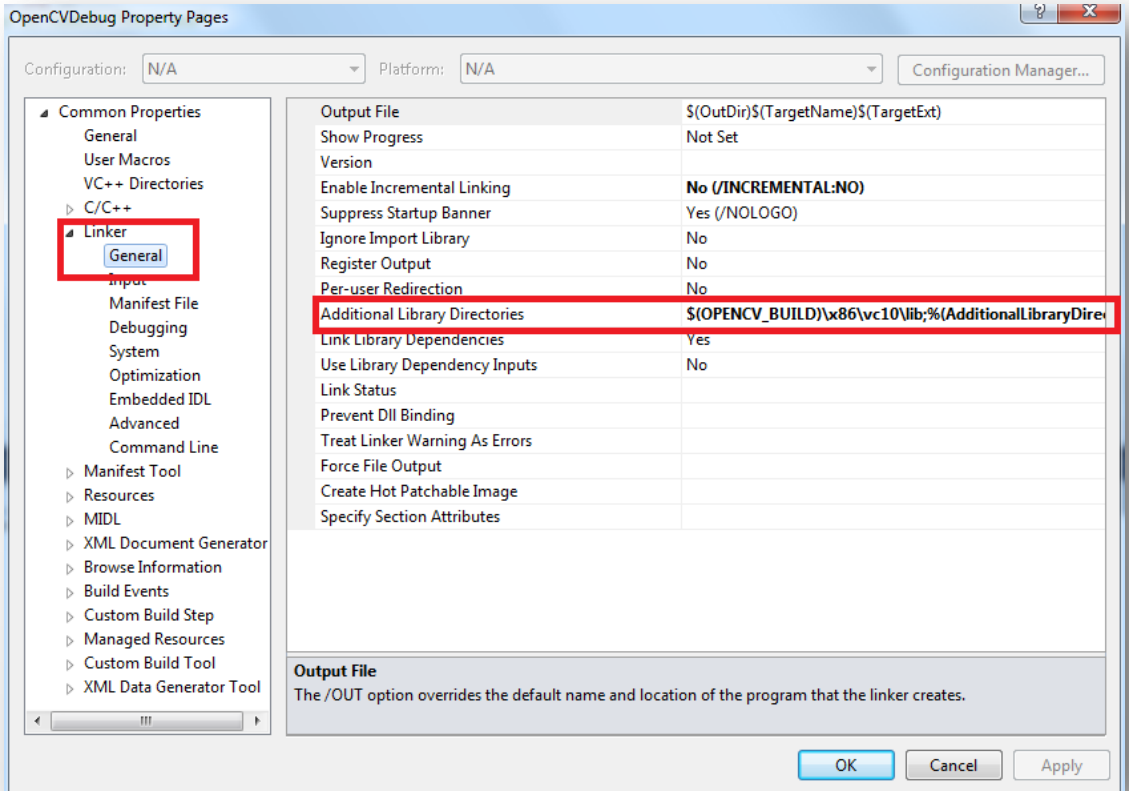
Property Sheet

- Dublu-click pe noul property sheet creat, **OpenCVDebug**.
- La **C/C++**, alegem **Additional Include Directories**, apasam **Edit** si in fereastra deschisa adaugam **$(OPENCV_BUILD)\include**
  - Astfel furnizam calea catre bibliotecile din OpenCV pe care le vom utiliza



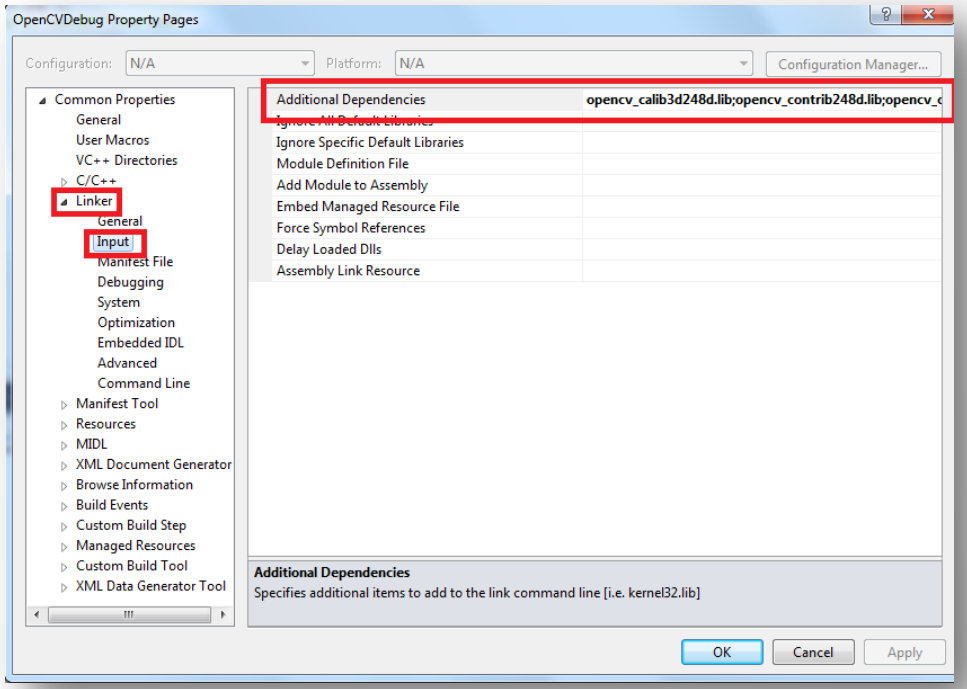
Property Sheet

- Mergem apoi la **Linker** -> **General** -> **Additional Library Directories**, apasam **Edit…** si introducem **$(OPENCV_BUILD)\x86\vc10\lib**.
  - Din nou, daca avem SO pe 64 de biti alegem folderul **x64**.



Property Sheet

- Tot la **Linker**, **Input** si **Additional Dependencies**, **Edit**.
- Adaugam in fereastra care se deschide:
  - opencv_calib3d248d.lib
  - opencv_contrib248d.lib
  - opencv_core248d.lib
  - opencv_features2d248d.lib
  - opencv_flann248d.lib
  - opencv_gpu248d.lib
  - opencv_highgui248d.lib
  - opencv_imgproc248d.lib
  - opencv_legacy248d.lib
  - opencv_ml248d.lib
  - opencv_nonfree248d.lib
  - opencv_objdetect248d.lib
  - opencv_photo248d.lib
  - opencv_stitching248d.lib
  - opencv_ts248d.lib
  - opencv_video248d.lib
  - opencv_videostab248d.lib

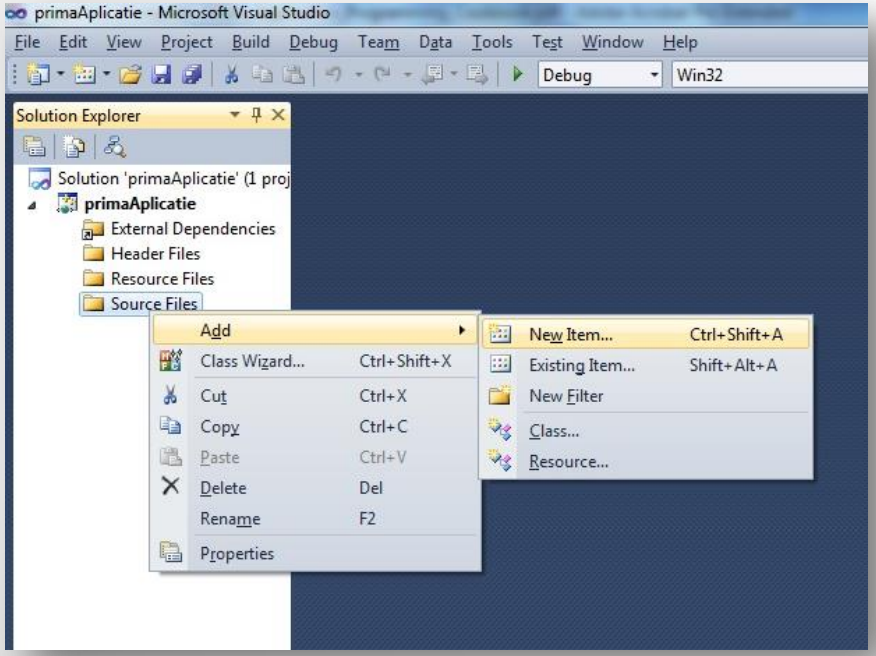


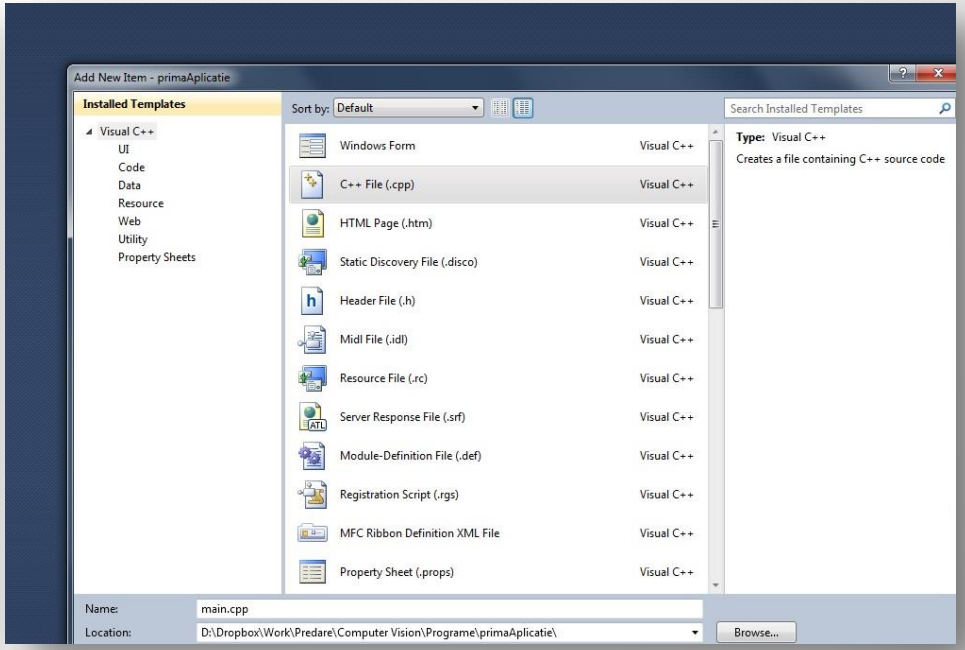
- 248 vine de la versiunea OpenCV. Daca aveti o versiune diferita, schimbati valorile.

Property Sheet

- Caracterul d de la finaul librariilor de pe slide-ul anterior vine de la **debug**.
- Cum ama facut pentru Debug va trebui sa facem si pentru varianta de Release.
  - **Add New Property Sheet** cu numele **OpenCVRelease**
  - **C/C++** -> **Additional Include Directories**
  - **Linker** -> **General** -> **Additional Library Directories**
  - **Linker** -> **Input** -> **Additional Dependencies**
    - opencv_calib3d248.lib
    - opencv_contrib248.lib
    - opencv_core248.lib
    - …
    - Toate fara d-ul de la final

Proiect OpenCV folosind Visual Studio 10





Build si Run

- Daca nu merge si nu identificati o eroare de sintaxa, trebuie reluati pasii anteriori. ☺

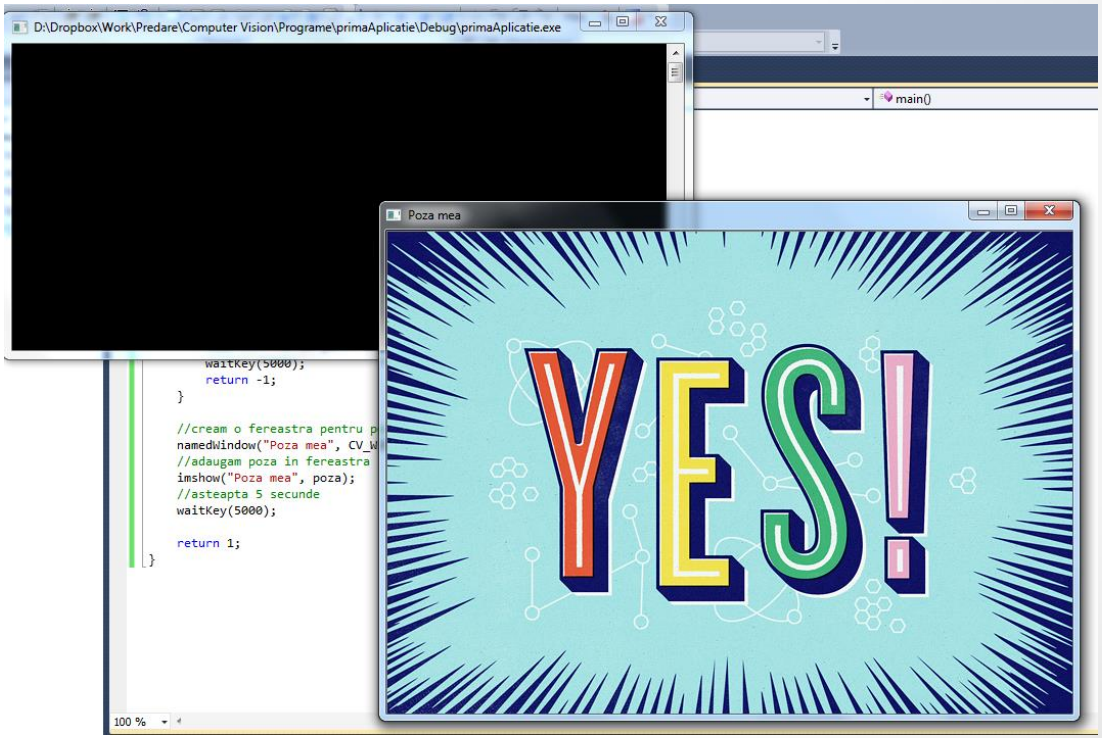
main.cpp

(Global Scope)

```
#include <iostream>
#include <opencv2/core/core.hpp>
#include <opencv2/highgui/highgui.hpp>

using namespace std;
using namespace cv;

int main()
{
    //citim o imagine
    Mat poza = imread("D:/poza.jpg");

    if(! poza.data )                              // Verific daca citirea este valida
    {
        cout <<  "Nu am gasit poza" << endl ;
        waitKey(5000);
        return -1;
    }

    //cream o fereastra pentru poza cu numele "Poza mea"
    namedWindow("Poza mea", CV_WINDOW_AUTOSIZE);
    //adaugam poza in fereastra
    imshow("Poza mea", poza);
    //asteapta 5 secunde
    waitKey(5000);

    return 1;
}
```

Ce face programul

- Afiseaza o poza



Proiect OpenCV folosind QT

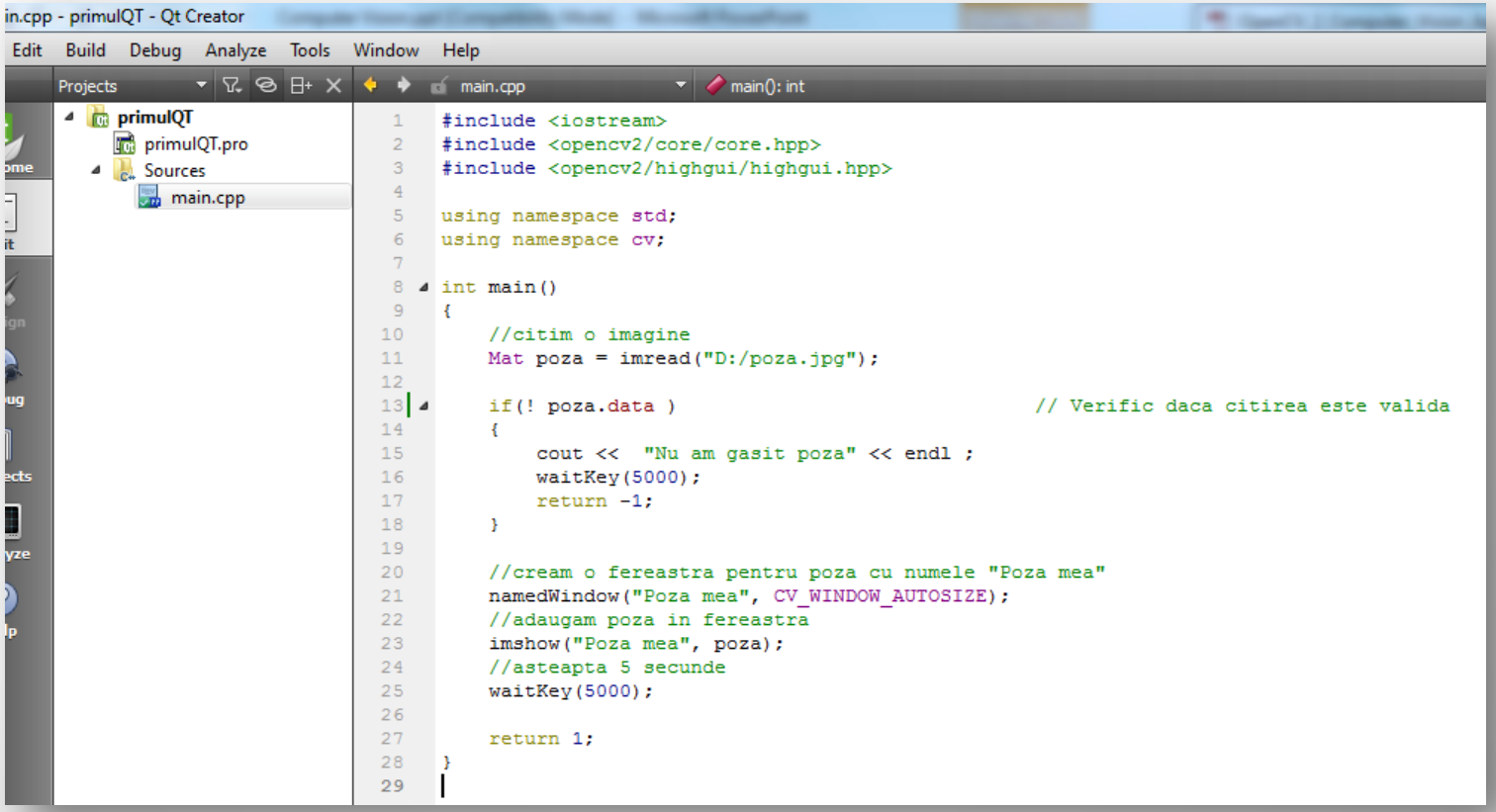
- QT se descarca de la

  http://qt-project.org/downloads

- Dezvoltat de compania norvegiana Trolltech si cumparat de Nokia in 2008.
- Open source
- Cross-platform

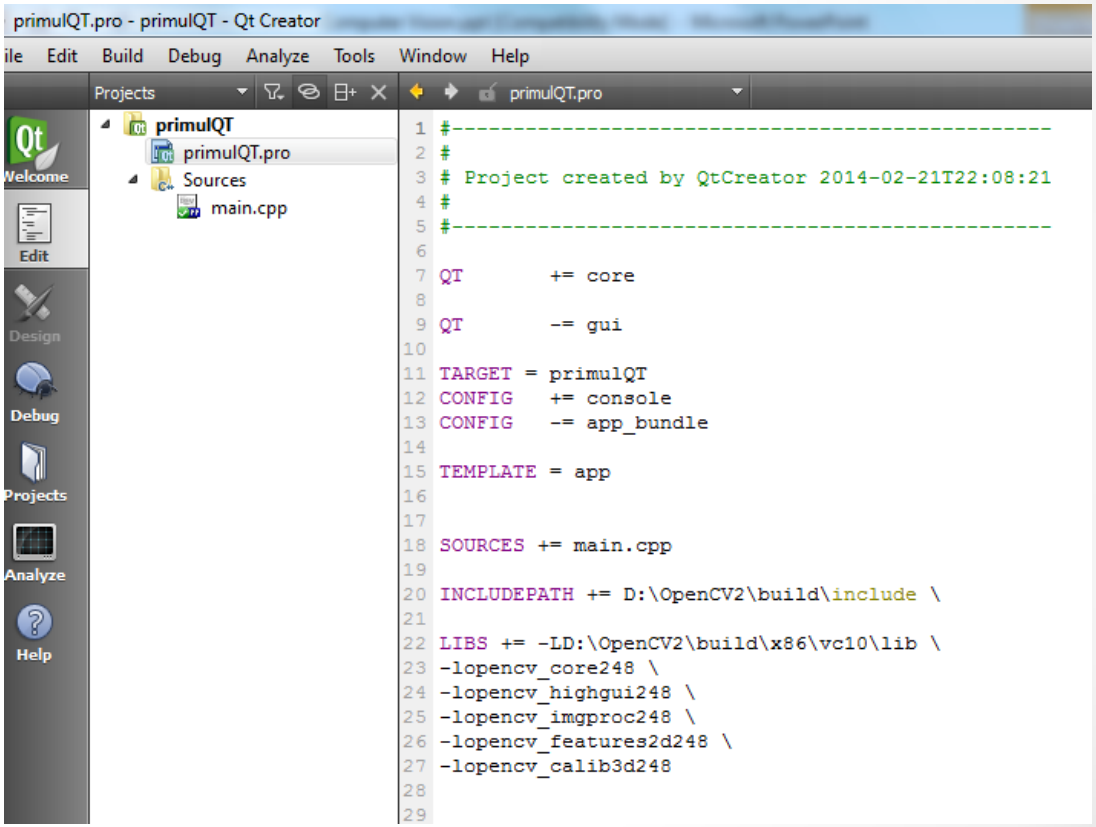
Proiect OpenCV folosind QT

- Cream un proiect de tip **Console Application**.



Proiect OpenCV folosind QT

- In fisierul .pro se specifica calea catre folderul **include** si catre librarii.
- Nu este nevoie de setari aditionale.



Proiect OpenCV folosind QT