بسمه تعالی

پروژه بینایی ماشین

نام ونام خانوادگی:رحیم برومندی

استاد راهنما:جناب دکتر حامد مسندی شیرازی

هدف پروژه:تشخیص ماشین از کنار در یک عکس(Side Car Detection)

الگوریتم مورد استفاده :الگوریتم ویولاجونز

محیط های مورد استفاده:متلب و اپن سی وی

تابستان 94 دانشگاه شیراز

مقدمه:

این پروژه کارشناسی مدل کردن یک ماشین از کنار(Side Car)،از طریق الگوریتم ویولا جونز می باشد.در این پروژه از محیط های اپن سی وی و متلب استفاده کرده،و در نهایت از روی دیتاست های جمع اوری شده،مدل ماشین که به صورت یک فایل xml می باشد،ساخته شده واز ان در تشخیص به کار گرفته می شود.تمامی سورس های در فولدر پروژه اورده شده است.تمامی عکس های استفاده شده در پروژه،از اینترنت و اسناد اپن سی وی گرفته شده است.در نهایت یک اپ ساده(به علت کمبود وقت به این اپ ساده بسنده کرده ام)برای اندروید،که با استفاده از مدل ماشین ساخته شده در این پروژه ماشین را تشخیص می دهد،اگر چه خطا هم دارد،کسی منکر خطا داشتن ان نیست.باید وقت گذاشت،ودیتابیس های بهتری جمع اوری کرد.این نوشته خالی از خطا نیست،و اگر اساتید خطای را دیدند،به بزرگی خودشان ببخشند.در نهایت از استاد گرانقدر جناب دکتر حامد مسندی،که بنده را در این پروژه یاری کردند،سپاسگذارم.

rahim.borumandi71@gmail.com

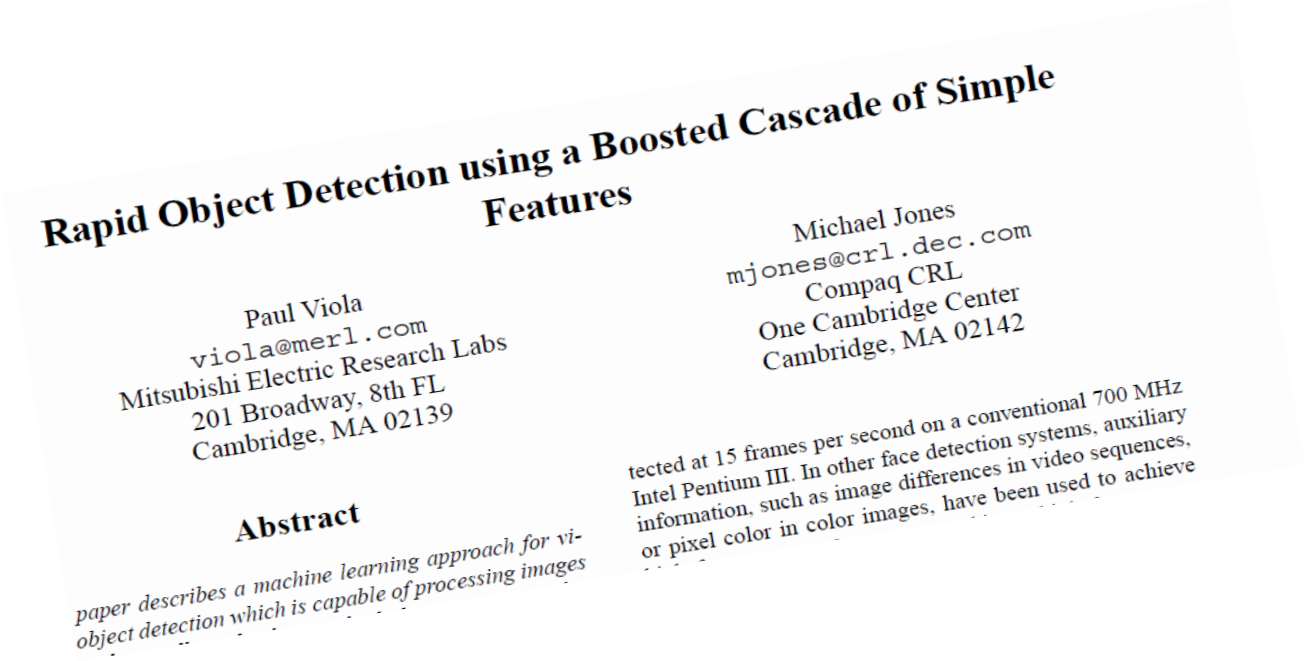
الگوریتم ویولا جونز:

کاربرد:به طور وسیع در تشخیص اشیا به صورت بلادرنگ به کار می رود.

مزایا:یکی از مزایای این الگوریتم این می باشد،که تشخیص خیلی سریع می باشد.(نسبت به الگوریتم های دیگر اصطلاحا بلادرنگ است." Real Time Detection ")

معایب:یکی از معایب ان ،اموزش ان زمان زیادی می برد.

روش Haar feature-based cascade classifiers یکی از موثرترین زوش ها در بینایی ماشین(تشخیص اشیا) می باشد،که به وسیله پل ویولا و مایکل جونز(Paul Viola وMichael Jones) در مقاله ای تحت عنوان " Rapid Object Detection using a Boosted Cascade of Simple Features" در سال 2001 منتشر شد.مقاله در فولدر پروژه اورده شده است.



Rapid Object Detection using a Boosted Cascade of Simple Features

Paul Viola
viola@merl.com
Mitsubishi Electric Research Labs
201 Broadway, 8th FL
Cambridge, MA 02139

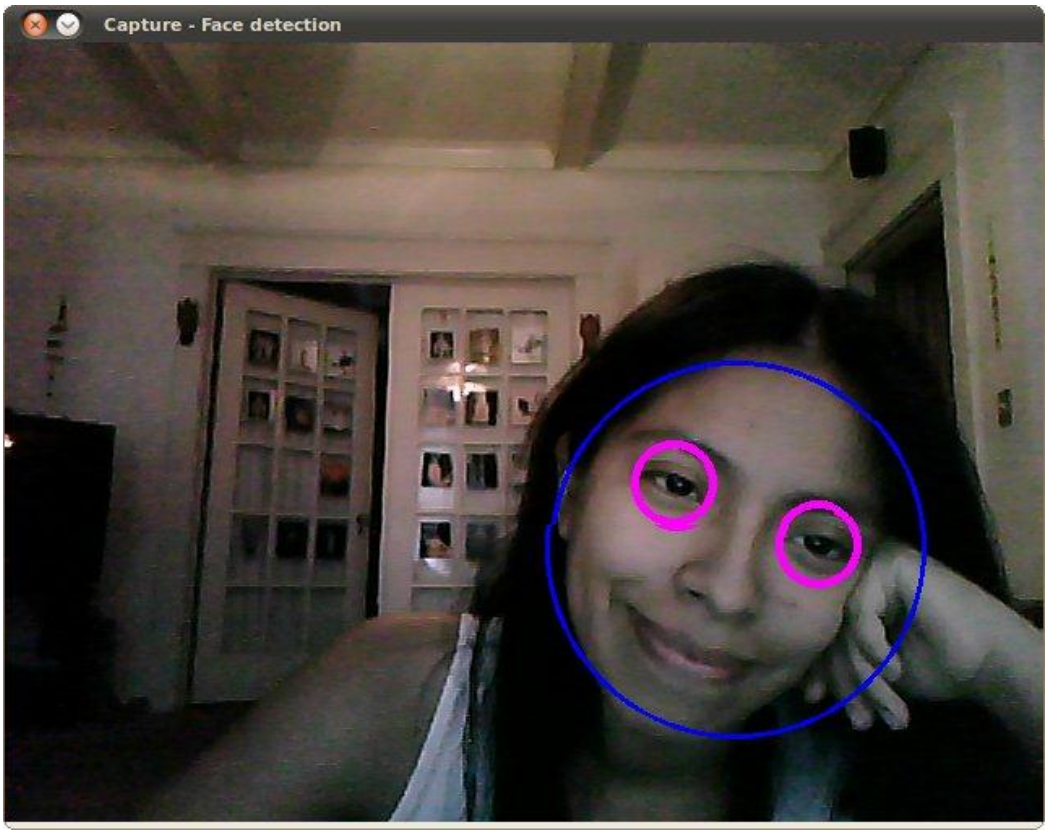
Michael Jones
mjones@crl.dec.com
Compaq CRL
One Cambridge Center
Cambridge, MA 02142

Abstract

paper describes a machine learning approach for vi-
object detection which is capable of processing images

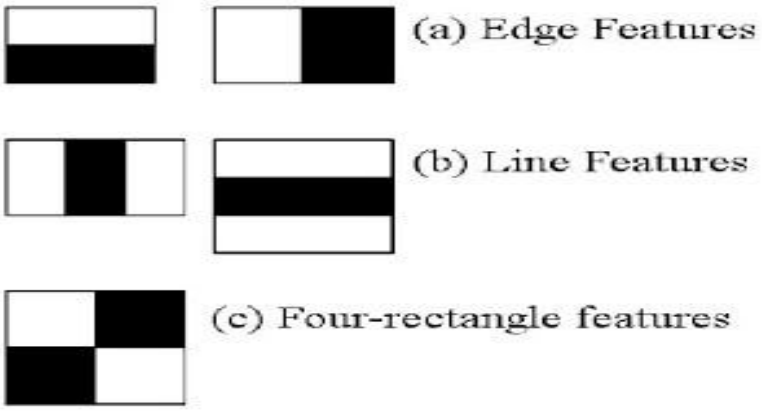
tected at 15 frames per second on a conventional 700 MHz
Intel Pentium III. In other face detection systems, auxiliary
information, such as image differences in video sequences,
or pixel color in color images, have been used to achieve

این الگوریتم ترکیب الگوریتم های AdaBoostو Cascadeمی باشد،که قادر بود در یک تصویر288x384 چهره را در مدت زمان 067.ثانیه تشخیص دهد.این الگوریتم 15 بار سریع تر ازاشکارساز های state-of-the-art و بادقتی بالاتر می باشد.این اگوریتم یکی از پیش رفته ترین الگوریتم های بینایی ماشین در دهه ی گذشته تا



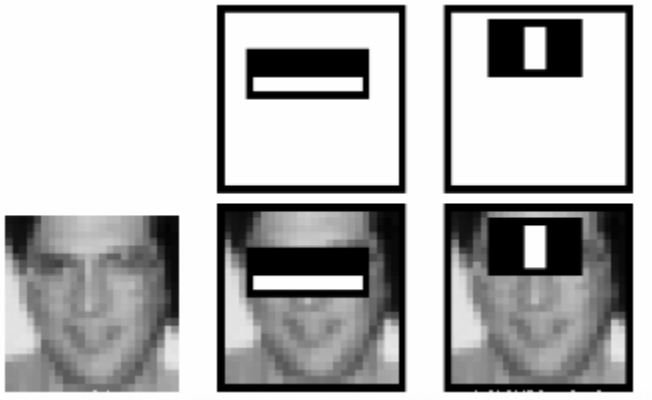
به حال بوده است.اما به طور خلاصه نقش هرکدام از الگوریتم های AdaBoostو Cascadeدر این الگوریتم شرح می دهیم.

ابتدا تصویر ورودی به تصاویر 24x24تقسیم می شود،هر زیر تصویر بیان گر یک بردار ویژگی است.برای اینکه محاسبات موثر وکار امد باشد،از یک سری ویژگی های خیلی ساده استفاده می کنیم.تمام مستطیل های ممکن،درون زیر تصویر بررسی می شوند.در هر مستطیل چهار ویژگی به کمک ماسک هایی که در شکل زیر امده استخراج می شود.(چهار ماسک وِیژگی که برای هر مستطیل استفاده می شود.معمولا ماسک Cکمتر مورد استفاده قرار می گیرد.)



با هرکدام از این ماسک ها،مجموع پیکسل های سطح خاکستری در نواحی سفید از مجموع پیکسل های نواحی سیاه کم می شود.پس می توانیم این طور بگوییم درون یک تصویر 24x24 یک میلیون ویژگی می توانیم داشته باشیم.(البته این ویژگی ها خیلی سریع محاسبه می شوند و می توانند کمتر از یک میلیون هم باشند مثلا 64000،البته لازم به ذکر است هرچه تعداد ویژگی کمتر کنیم اموزش سریع تر ،اما دقت کاهش می یابد.) هرویژگی به عنوان یک یادگیرنده ی ضعیف در نظر گرفته می شود.الگوریتم یادگیری پایه تلاش می کند ،که بهترین کلاسیفایر ضعیف را کوچکترین خطلا را در کلاس بندی دارد،پیدا کند.حال در بین تمامی زیر تصاویر ممکن،و تمامی مکان های کرنل،را باید چک کند،که ویژگی های بسیار زیادی بدست می اید.حال اگر بخواهیم این ویژگی ها را کمتر کنیم باید یک معیار داشته باشیم.حال تصور کنید،حال برای یک تصویر24x24 ما 16000 ویژگی در نظر بگیریم.برای محاسبه ی هر ویژگی ما باید،جمع پیکسل های سیاه و سفید را بیابیم.برای حل سریع تر ان ها انتگرال تصاویر را معرفی کردند،این روش جمع پیکسل ها را ساده تر می کند.حال در میان این همه ویژگی اکثر انها نامربوط می باشند،و برای کار تشخیص ما مفید نمی باشند.برای مثال به تصویر زیر نگاه

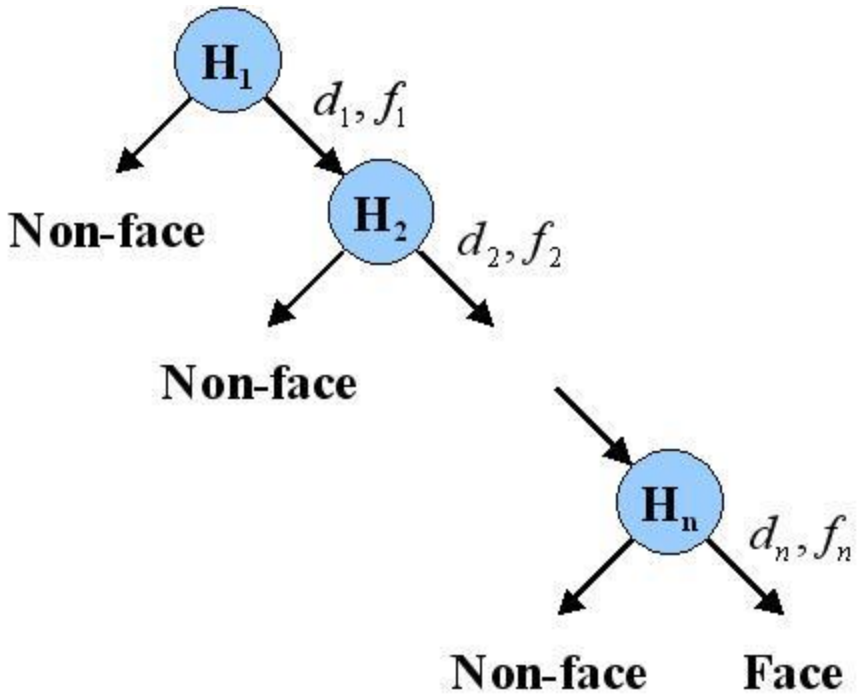
کنیدفاین دو ویژگی ردیفی وستونی زیر را در نظر بگیرید.



ویژگی ردیفی چون شامل چشم می باشد،که معمولا نسبت به بخش های دیگر صورت کمی تار تر است ویژگی خوبی به نظر می رشد،ویژگی ستونی هم ویژگی خوبی است،چون،بر روی بینی تمرکز کرده است،اما ویژگی های در بعضی جاها نامربوط می باشد و کمکی در تشخیص به ما نمی کند.معیاری که با ان با بهترین ویژگی ها را انتخاب کنیم AdaBoostمی باشد.برای این کار ما برای objectو no object بودن یک حد استانه(که میزان خطا را مشخص می کند)تعریف می کنیم،و برای ان ویژگی هایی که از حد استانه کمتر باشند،ان ها را حذف می کنیم.

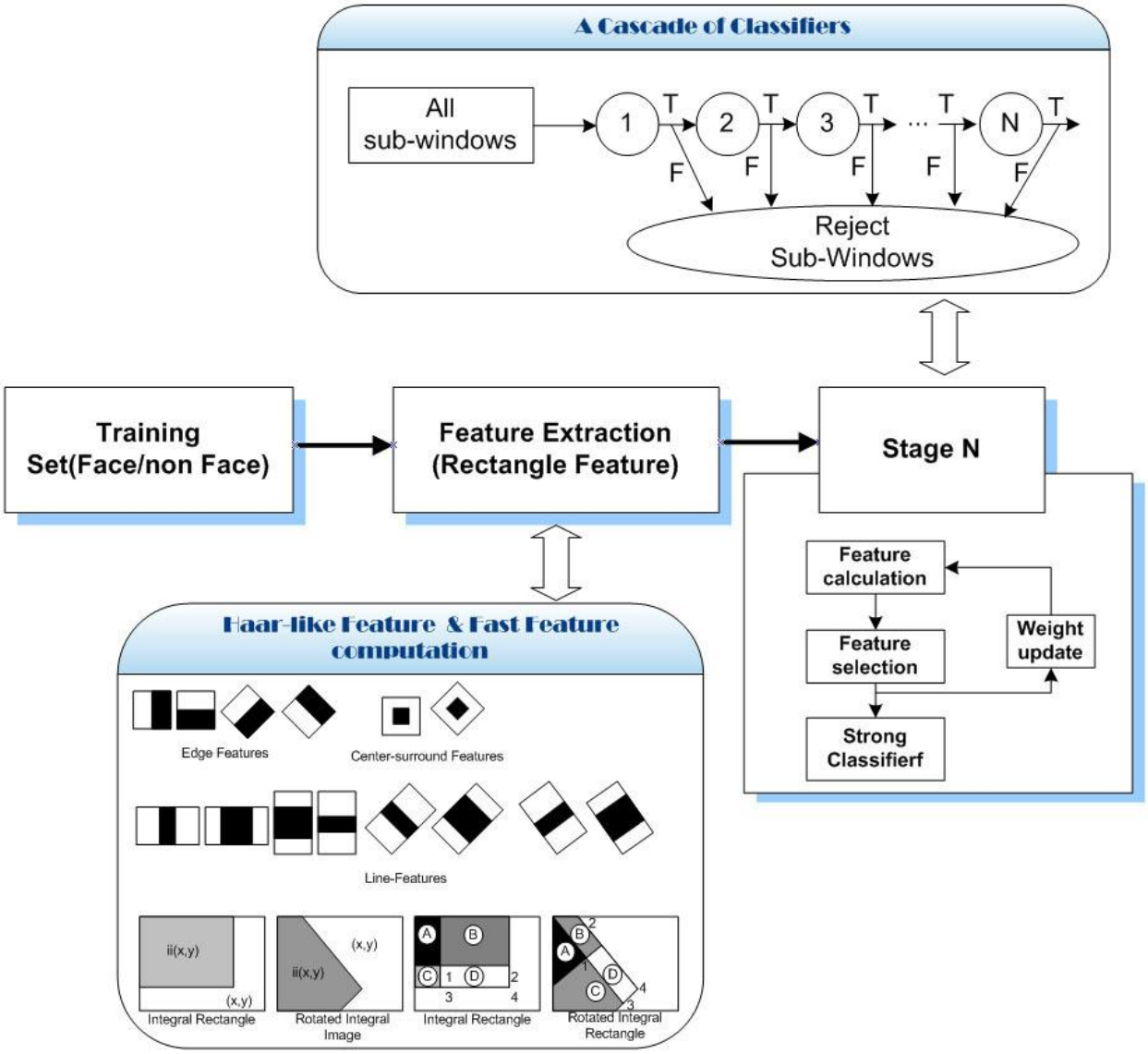
الگوریتم کسکید(Cascade Algorithm):

یک روش عددی در ریاضیات می باشد،که متناوبا پشت سرهم تکرار می شود،که بعد از مثلا N مرحله خطای ان از یک حد استانه کمتر می شود.از قضا این الگوریتم در پردازش تصویر نیز استفاده کرده اند.



الگوریتم ادا بوست(AdaBoost Algorithm):ادابوست که کوتاه شده ی" Adaptive boosting" می باشد،یکی از الگوریتم های بینایی ماشین می باشد،که به وسیله " Yoav Freund and Robert Schapire" فرمول بندی شد،که در سال 2003 جایزه "Gödel Prize" را بردند.این الگوریتم می تواند با سایر الگوریتم های بینایی ماشین ترکیب شود،که در نهایت باعث افزایش کارایی الگوریتم می شود.خروجی دیگر الگوریتم های بینایی ماشین(یادگیرنده ضعیف)با جمع وزن دار ترکیب می شود،که خروجی نهایی از کلاسیفایر بوست را بیان می کند.یکی از معضلات بینایی ماشین "curse of dimensionality" می باشد،همان طور که در قسمت بالا بحث شد،هر مثال ویژگی های زیادی دارد،که اگر بخواهیم همه ان ها را محاسبه کنیم مطمینا زمان زیادی را یاد برای ان صرف کرد.برای مثال وقتی ما از الگوریتم Viola–Jones object detection استفاده می کنیم تعداد Haar features ها به در یک تصویر 24x24 پیکسل 162,336 می رسد،که ارزیابی هر ویژگی می

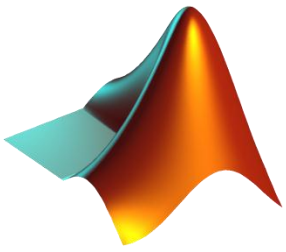
تواند نه تنها زمان اموزش را بالا ببرد،بلکه reduce predictive power, per the Hughes Effect را کاهش می دهد که این اصلا برای کارایی و اعتبار ان خوب نمی باشد.بر خلاف neural networks و SVM ها ،پروسه اموزش ادابوست ان ویژگی هایی را انتخاب می کنند،که predictive power را بهبود و dimensionality را کاهش بدهند،زمان اموزش را کاهش،ویژگی های نامربوط را حذف کنند.



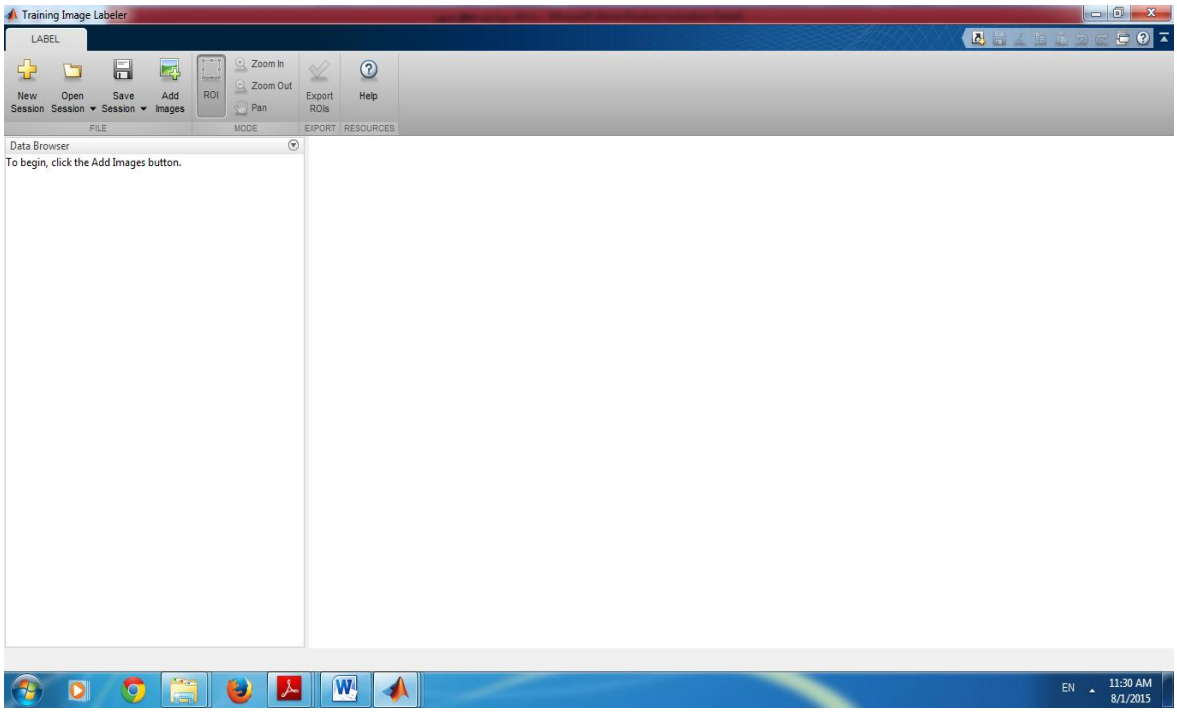
کلاسیفایر از برچسب گذاری داده ها،اموزش داده می شود.در اموزش با روش الگوریتم ما با دو گروه عکس (به عنوان دیتا)سرو کار داریم. object ,no object که درو روش ویولا جونز از 5000تا faceو از 300میلیون no faceتشکیل شده

است.عکس های faceها همگی نرمالیزه شده اند.برای مطالعه عمیق تر الگوریتم های پردازش تصویر می توانید به کتاب های پرداش تصویر مراجعه کنید.

روش انجام پروژه در محیط های متلب و اپن سی وی:

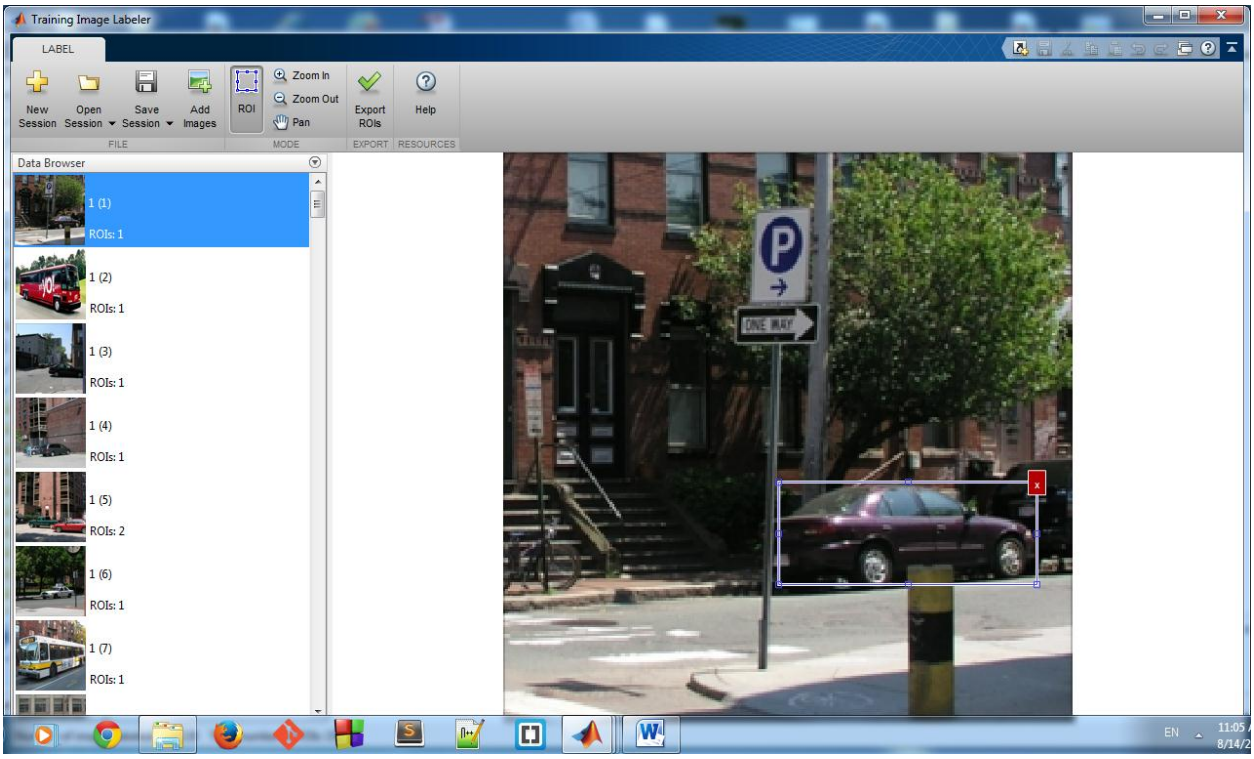


از انجا که پروژه بنده تشخیص ماشین با استفاده از الگوریتم viola-jonesمی باشد ،داده های عکس من شامل carو no carمی باشد،که این دو فولدر عکس از روی اینترنت جمع اوری شده است.بعد از اینکه این دوفولدر عکس اماده شد،(قاعدتا هرچه تعداد داده ها بیش تر باشد،زمان اموزش افزایش می یابد ،اما دقت ان افزایش می یابد.)نکته ای که باید بیان کنم تمامی عکس های استفاده شده در پروژه،ار اینترنت و اپن سی وی و متلب گرفته شده است.بعد از ان برای اموزش ان در متلب ،باید با یک ابزار به نام trainingImageLabeler استفاده می کنیم که کافی نام انرا در کامند ویندو متلب تایپ کنیم.این ابزار جهت استفاده ساده تر از الگوریتم viola-jones در اموزش اشیایی می باشد که مدل ان در متلب موجو د نیست.(مدل های صورت،بینی،چشم،دست،علامت راهنمای ایست اموزش داده شده اند.)این ابزار در متلب R2015a اضافه شده است.(برای اینکه در انجام مراحل جهت ساخت مدل مورد نظر خود،به مشکل برخود کنید ورژن های نرم افزار ها را دقت کنید،که از همین ورژن(شدیدا توصیه می کنم) یا ورژن های جدید تر استفاده کنید.)

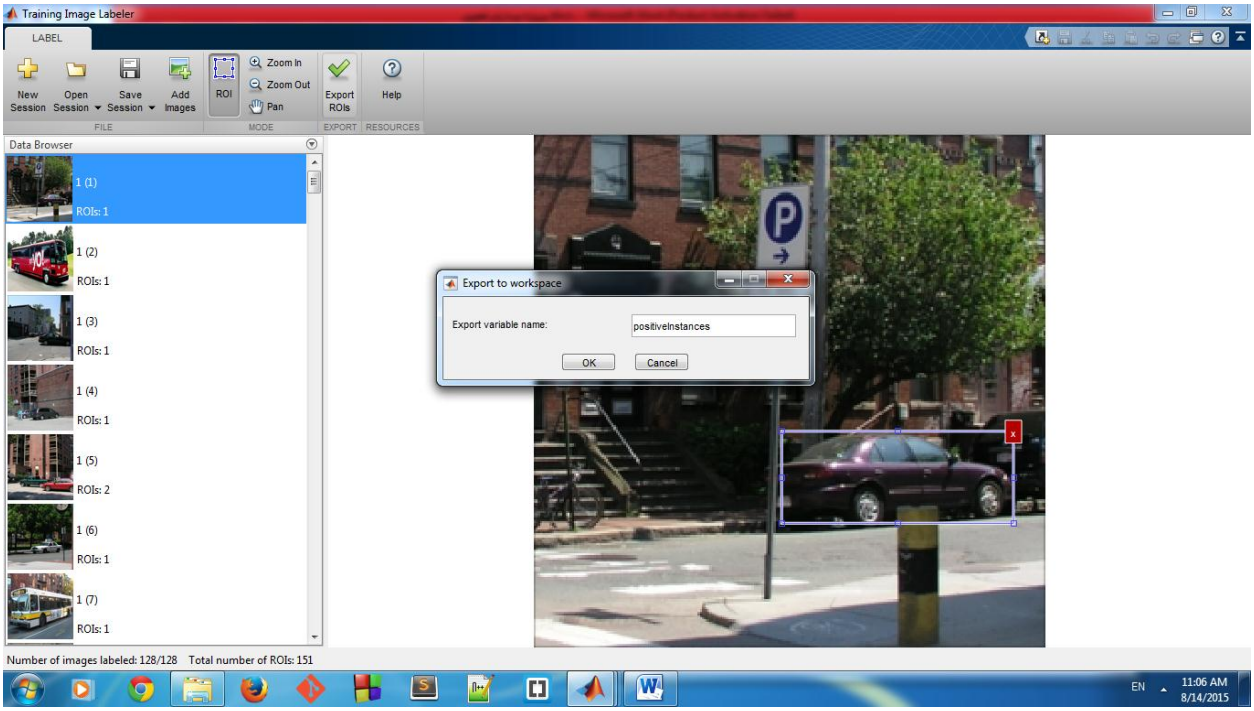


حال می بینم که این ابزاراز طریق Add image می توانیم تصویر های carخود را به ان داده و ROI(Region of interest) هر تصویر را مشخص کنیم.همان طور که از نامش پیدا است یعنی باید با این منو ناحیه از تصویر که شی ما دران واقع شده

است مشخص کنیم،باید برای تک تک تصاویر این کار را کنیم،و در نهایت متلب از روی ان ها یک فایل با فرمت mat می سازد،که این فایل در اموزش مدل خیلی مهم است،که با زدن منوی ذخیره(save) فایا با فرمت mat را در محل دلخواهمان ذخیره می کنیم.(اگر کنجکاو شدید که این فایل چیست،باید بگم مشیر عکس های مثبت و ROI در خود ذخیره می کند،برای اطلاعات بیش تر بهworkspace متلب رجوع کنید.)



حال برای اموزش با استفاده از داده های carو no car وفایل matساخته شده از تکه کد زیر در متلب استفاده می کنیم.قبل اجری این کد،برای ابزار برچسب گذار تصویر منو exportرا زده،و نام ورودی را نام پیش فرض بگذارید.



exportکردن در اینجا در واقع متغیر های داخل فایل matرا به workspaceمتلب انتقال می دهد.

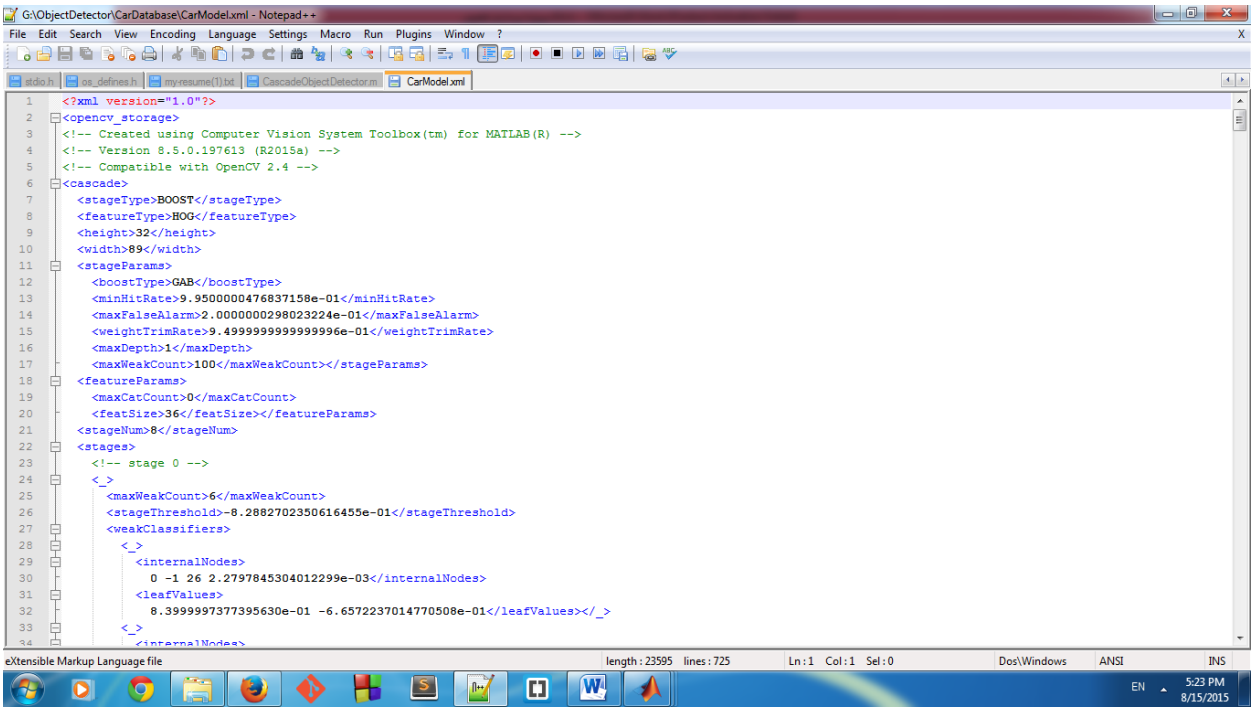
حال تکه کد زیر را اجرا کنید،

```
load('labelingSessioncardetect.mat');
imDir = fullfile('car');
addpath(imDir);
negativeFolder = fullfile('nocar');
trainCascadeObjectDetector('CarModel.xml',
positiveInstances,negativeFolder,'FalseAlarmRate',0.2,'NumCascadeStages',5);
```

در تکه کد بالا اولین چیزی که باید بارگذاری شود،فایل matمی باشد،(لازم به ذکر است نام ان را اگر غیر این گذاشته اید،نام خود بگذارید،وباید این فایل در current path متلب باشد،در غیر این صورت خطا خواهد داد،بعد از ان ما تمام فایل های فولدر carوبعد از ان فولدر nocar زا اضافه می کنیم.

که در کد بالا 2.مقدار،خطای قابل قبول و 5 تعداد لایه های اموزش می باشد. باید توجه داشته باشیم هرچه تعداد لایه های اموزش ما بیشتر و ضریب خطای ما کمتر باشد،برای عکس های negative که در اینجا nocar می باشد،باید خیلی زیاد باشد،مثلا در صفحات بعد به ان خواهیم رسید که بنده برای خطای 2. و 8 مرحله اموزش حدود سه هزار عکس استفاده کرده ام.

بعد از اموزش خروجی در متلب یک فایل xmlخواهدبود که ما از روی ان در تشخیص اشیا به کار می گیریم.(می دانید که xml یک زبان است که برای دیتا به کار می رود)نمونه یک فایل xmlساخته شده در زیر می بینید.(اطلاعاتی در مورد ویژگی ها،نوع الگوریتم،وی یک سری پارامتر های دیگر که اگر دوست داشتید داخل فولدر پروژه CarDatabase می توانید ببینید.



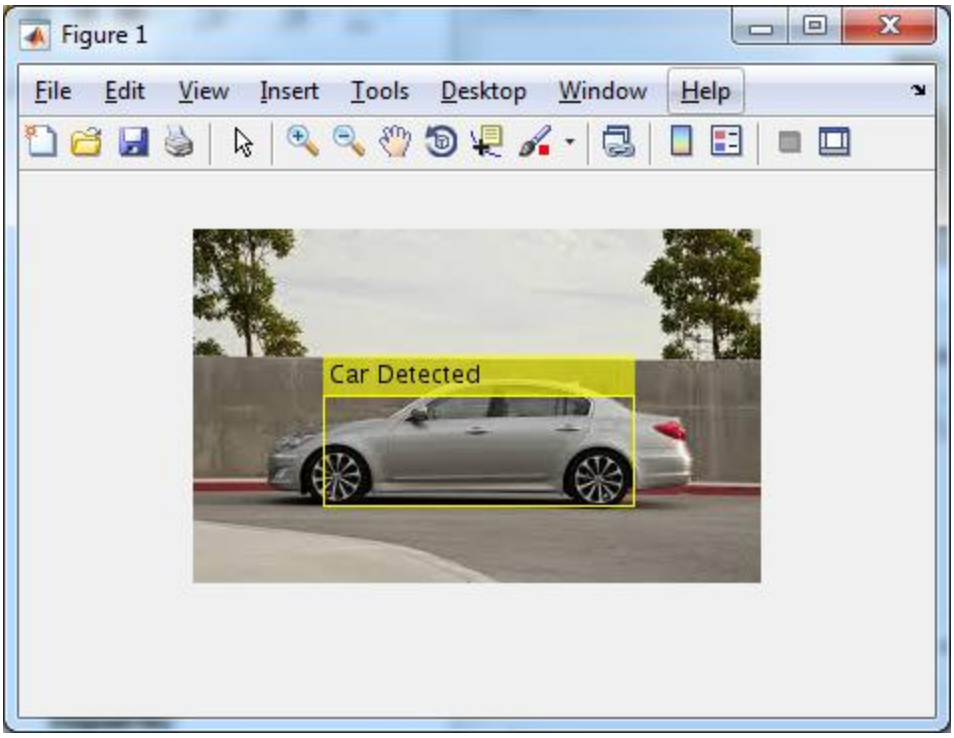
بعد از ان با تکه کد زیر ما برای تشخیص ماشین در تصویر استفاده می کنیم،که اول مدل خود را به اشکار ساز معرفی می کنیم.بعد که باید نام تصویر را در قسمت خواندن تصویر جایگزین کنیم.خط بعد در عکس در صورت وجود مدل را تشخیص می دهد،و خط بعدی،دور مدل قرار است مستطیل بکشد،خط بعد هم نشان دادن عکس خروجی.

```
detector = vision.CascadeObjectDetector('CarModel.xml');
img = imread('image.jpg');
bbox = step(detector,img);
detectedImg = insertObjectAnnotation(img,'rectangle',bbox,'cardetection');
figure;
imshow(detectedImg);
```

اکنون ما باید داده های تصویر خود را زیاد کنیم تا ،دقت تشخیص ماشین در عکس را افزایش دهیم.برای این کار ما از دیتابیس تصویر جمع اوری شده به وسیله خودم از روی اینترنت،اموزُش داده و نتایج را با هم مدل ساخته شده به وسیله ی UIUC dataset مقایسه می کنیم.لازم به ذکر است فرمت عکس های که برای اموزش استفاده می کنیم، jpg است.ما برای اموزش از 128 عکس مثبت و 3567 عکس منفی استفاده کرده ودر هشت لایه اموزش می دهیم.

دیتابیس UIUC یک دیتابیس استاندارد،برای ماشین می باشد،که فرمت عکس های ان pgmمی باشد،با دستور imreadمتلب به راحتی می توانید،ان هارا بخوانید،trainingImageLabeler هم این فرمت را پشتیبانی می کند،که به وسیله این افراد Shivani Agarwal, Aatif Awanو Dan Roth ساخته شده ،که در فایل پروژه در فولدر database std فایل CarData.tar.gz می باشد،که با WinRaR به راحتی می توانید اکسترکت کنید.

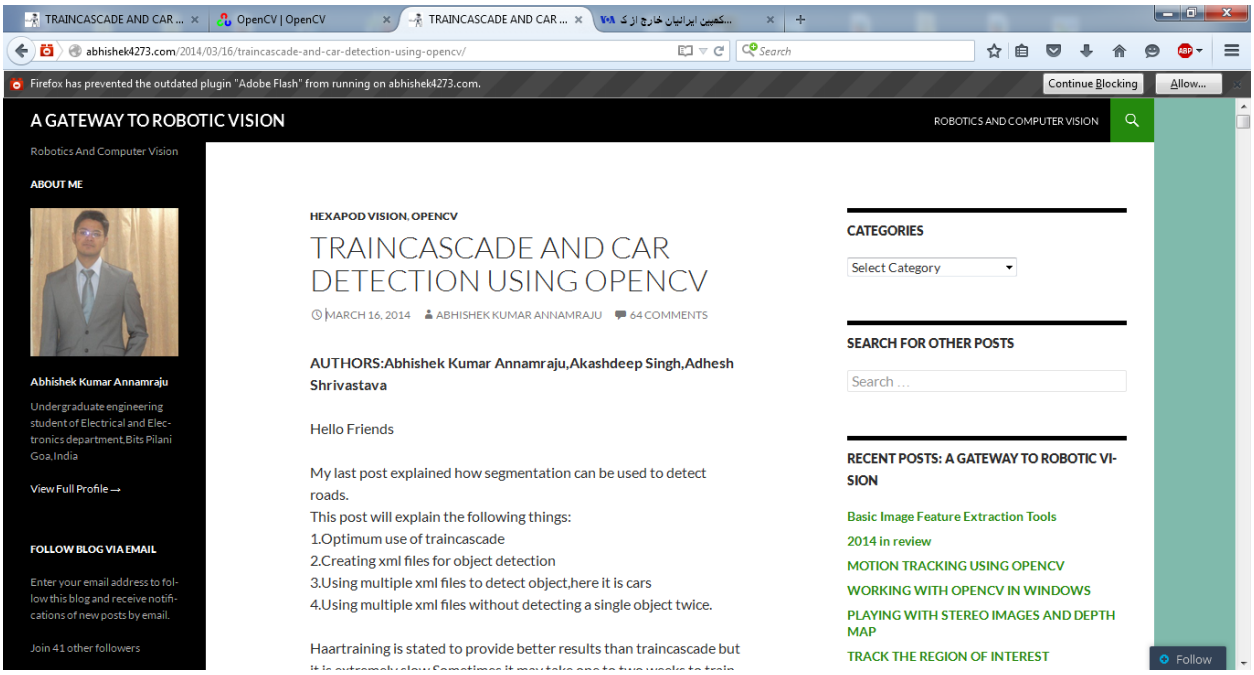
البته لینک این دیتابیس هم https://cogcomp.cs.illinois.edu/Data/Car می باشد.



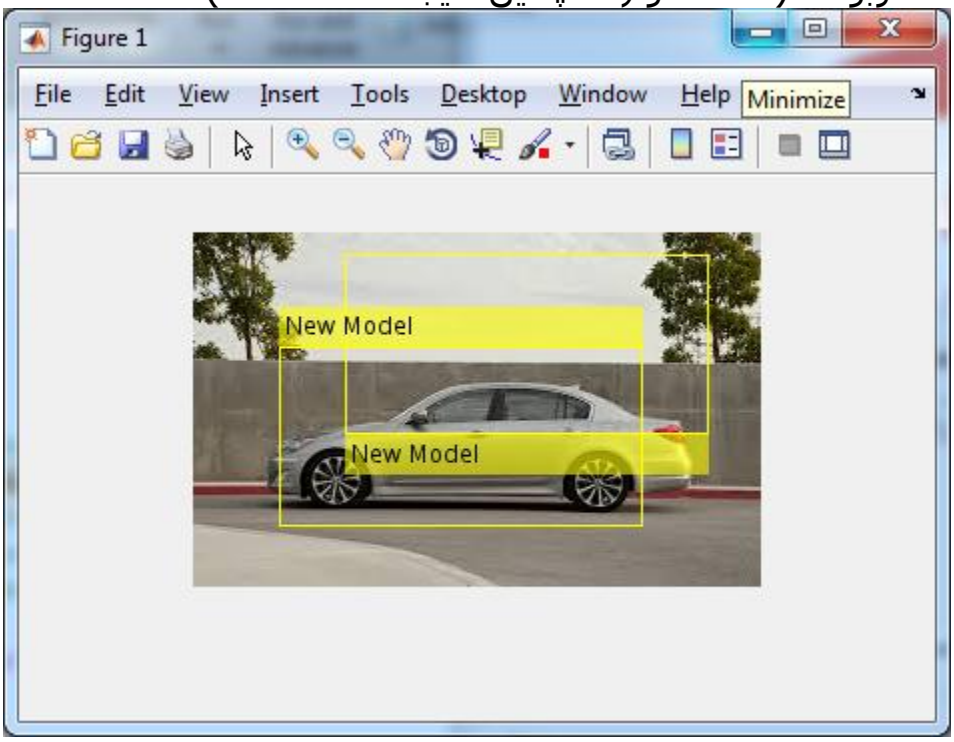
یک فرد به نام abhi-kumar امده با این دیتابیس استاندارد،از طریق اپن سی وی ،مدل ماشین را اموزش داده ،که یک سری فایل xml به اشتراک گذاشته،به عنوان مدل ماشین ،که قرار شد (استاد و بنده) مدل خودمون را با این مدل مقایسه کنیم.

اگر از مدل دیگری که به سیله دیتاست UIUC توسط abhi-kumar اموزش داده شده است ،که فایل های ان در پوشه database std قرار دارد،مقایسه کنیم،می بینیم، فایل مدل ما دقت بهتری دارد.لینک سایت و پروژه ان را از لینک زیر می توانید مشاهده کنید.

http://abhishek4273.com/2014/03/16/traincascade-and-car-detection-using-opencv/

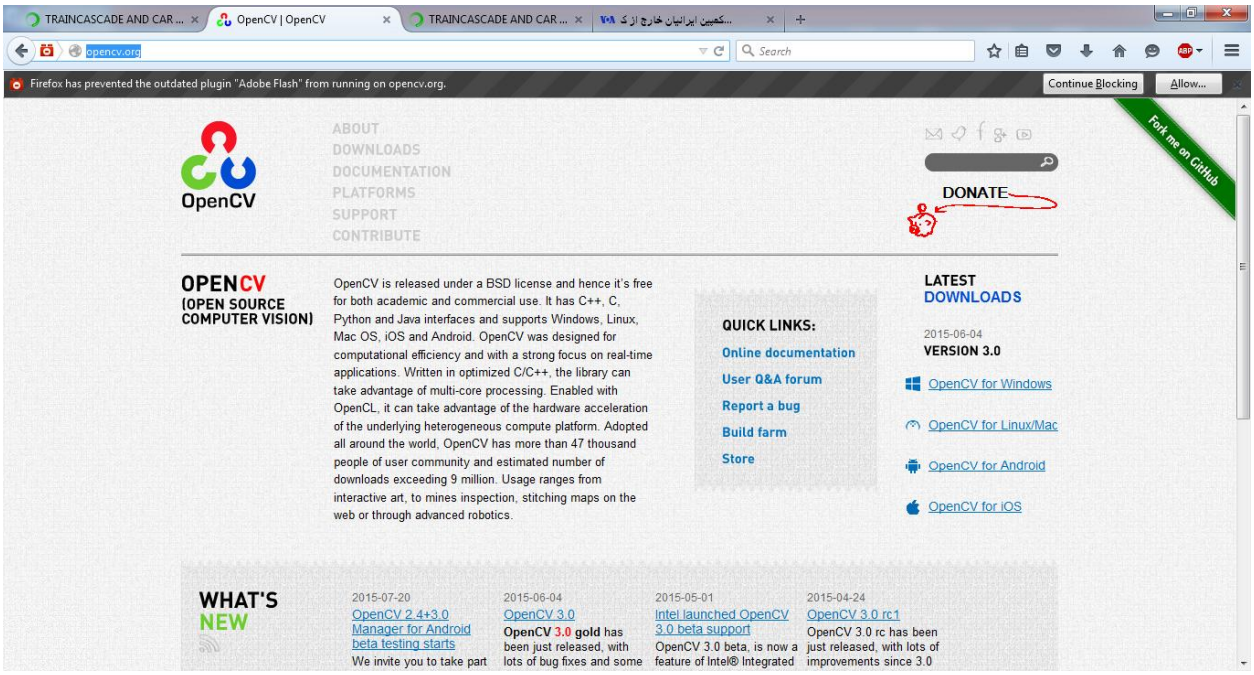


عکس خروجی با مدل مربوطه:(مدل سوم ان چنین نتیجه ای داشت.)

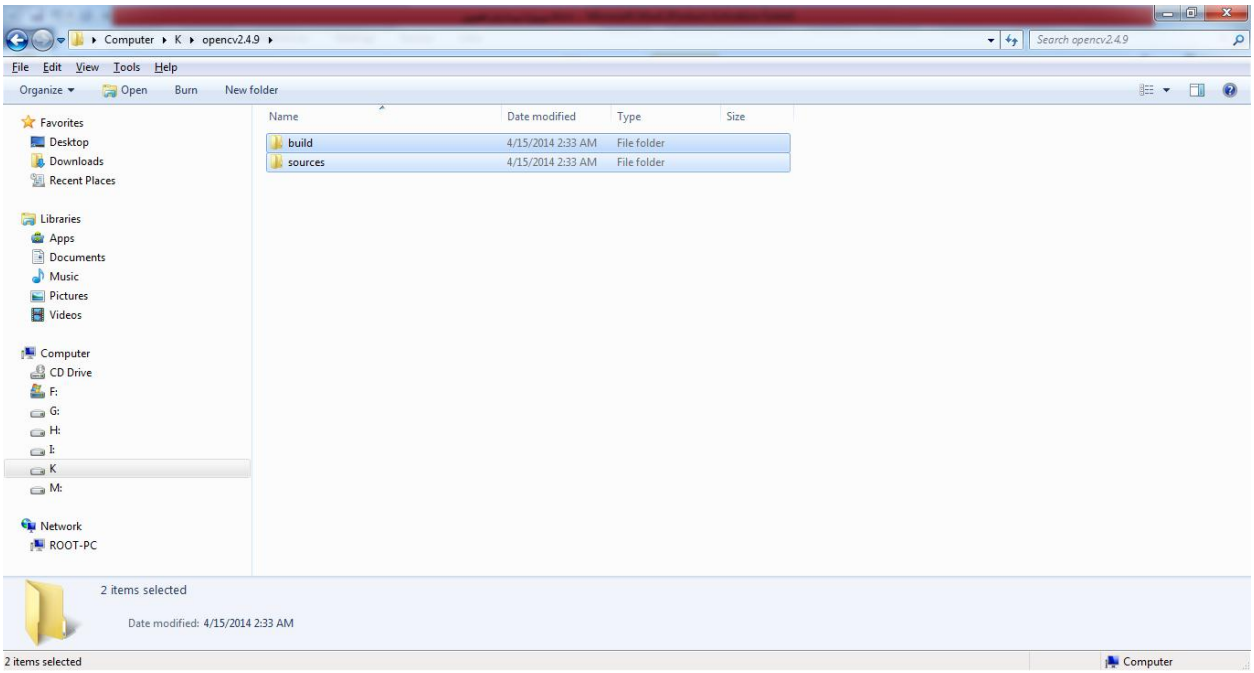


برای اموزش در اپن سی وی از دستورات خط فرمان استفاده کرده و اموزش را می دهیم در نهایت یک فایل xmlساخته می شود.(مانند متلب این فایل با فایل که در متلب ساخته می شود،فرقی ندارد،در متلب هم می توان از ان استفاده کرد.)

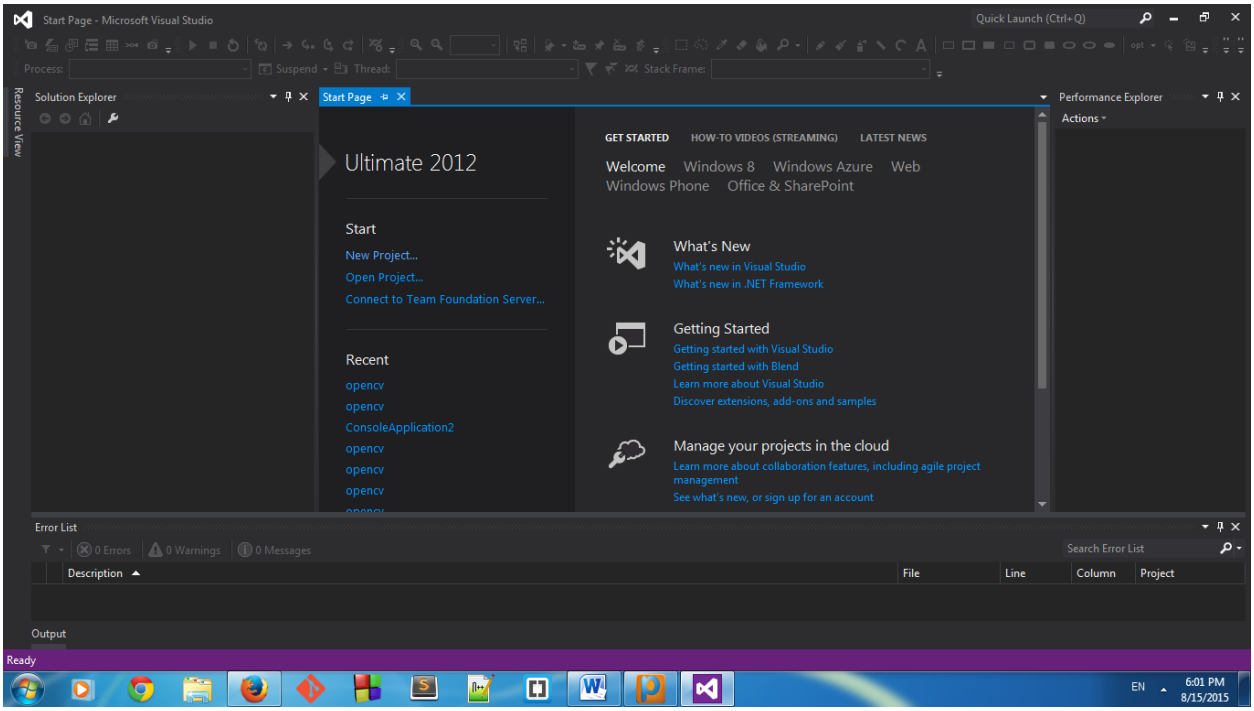
برای استفاده از اپن سی وی در پردازش تصویر باید اولین مرحله به سایت http://opencv.org رفته و نسخه opencv2.4.9 را برای ویندوز دانلود کنید.



دومین مرحله بعد از اکسترکت کردن ان،دو پوشه دارد، build و source که ما از نسخه بیلد شده استفاده کرده،در Visual Studio 2012با ید ان را پیکره بندی کنیم.نسخه سورس برا وقتی هست که ما می خواهیم ازطریق یک کامپایلر خودمان نسخه بیلد را بسازیمس،مثلا برای استفاده از اپن سی وی در کیوت حتما باید خودتون از روی سورس ان را بیلد کنید.

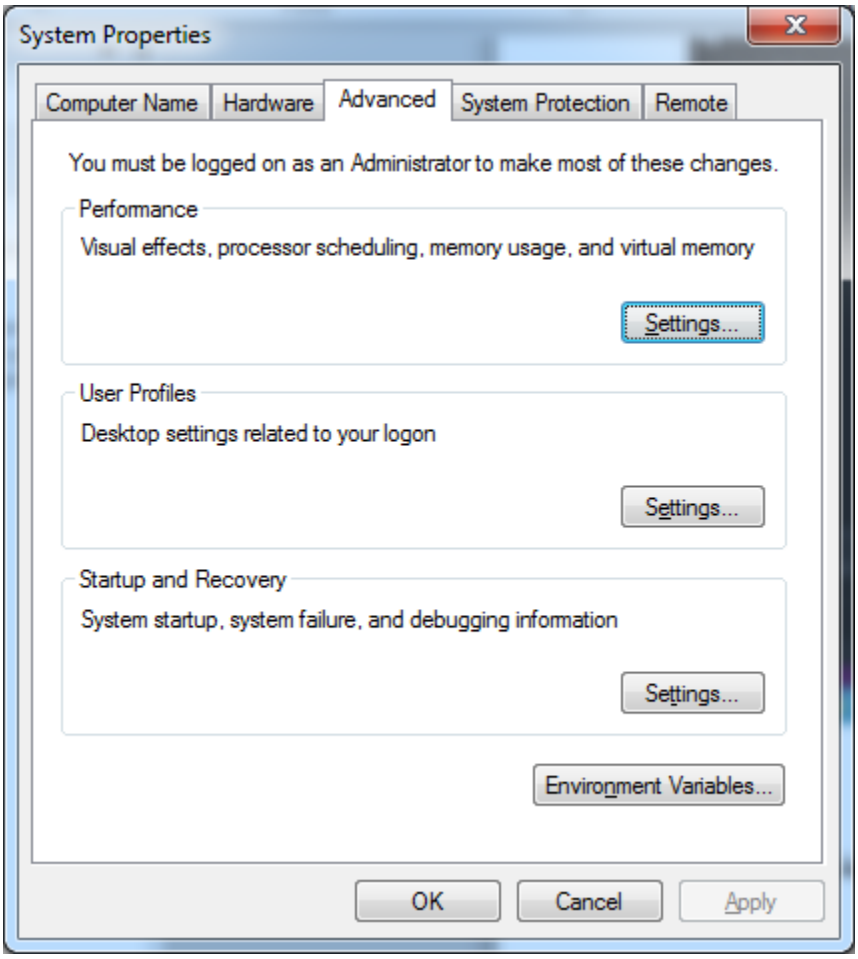


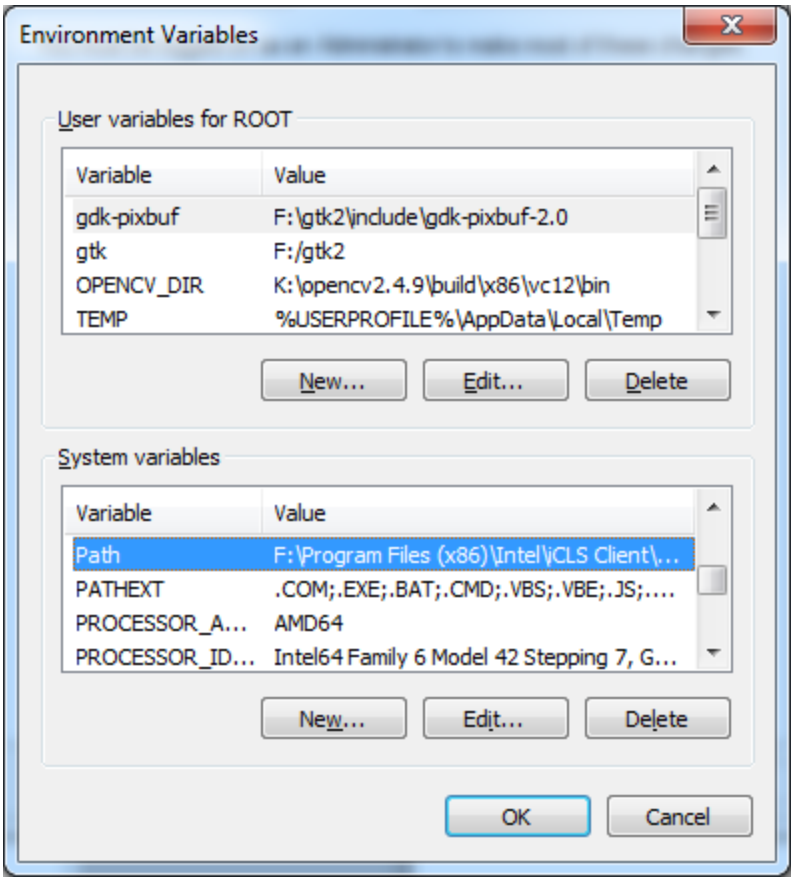
(پس فرض می کنم ویژوال استدیو 2012 را نصب دارید.)

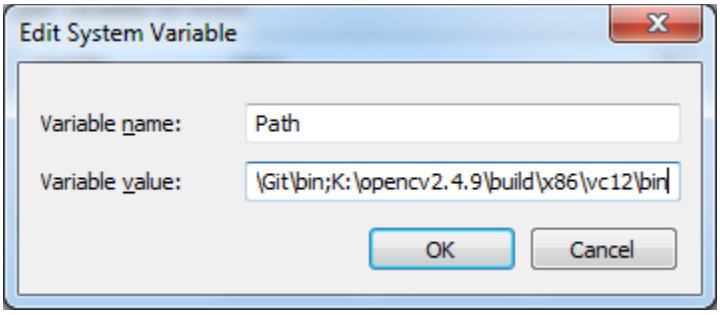


اولین مرحله باید مسیر opencvرا به متغیر سیستمی ویندوز اضافه کنید،در منوی سرچ کافی است تایپ کنید Edit system environment variables .

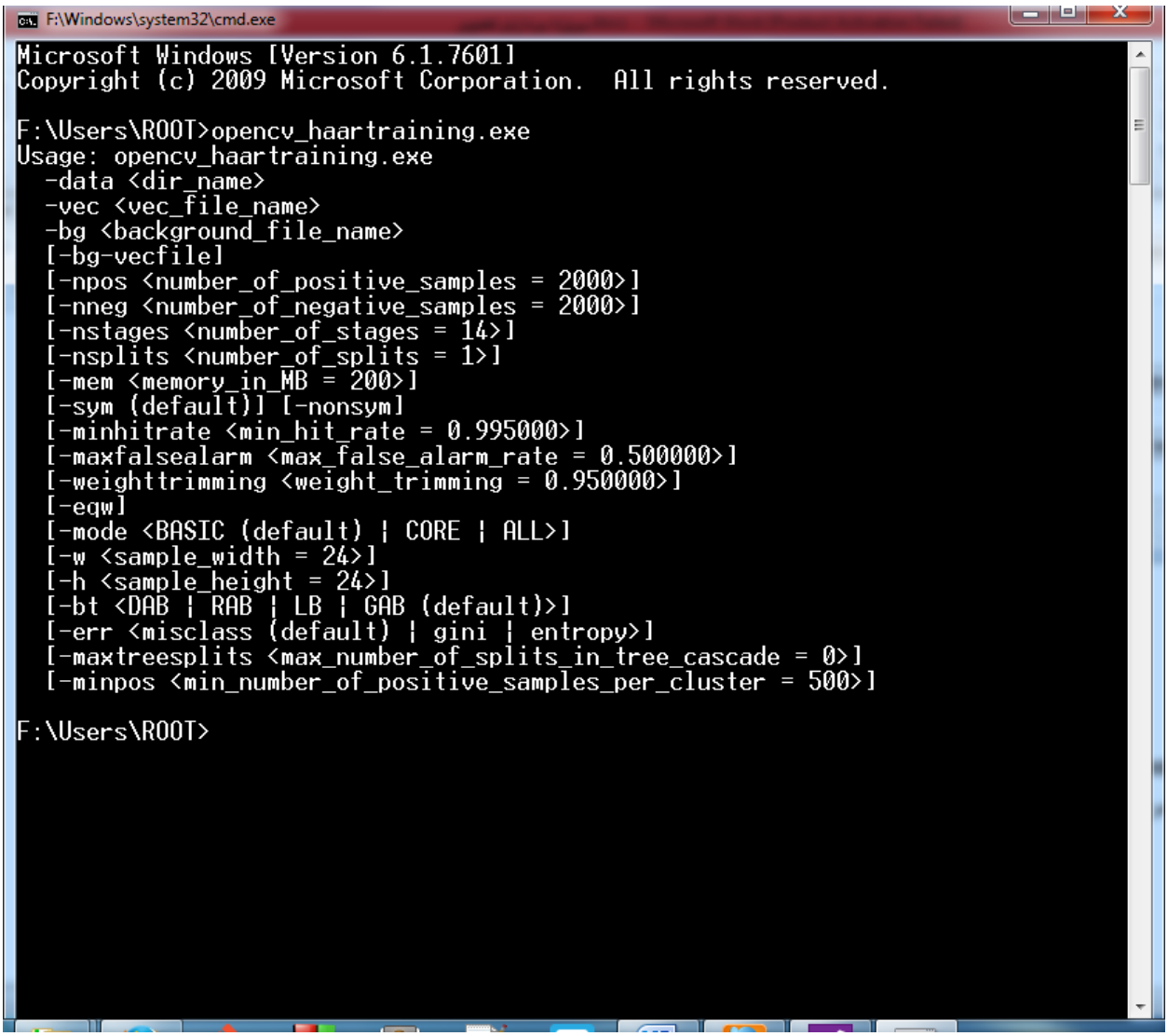
بعد روی Environment Variablesکلیک کنید.در قسمت system variables ، path را باز کرده،و مسیر K:\opencv2.4.9\build\x86\vc12\bin را به انتهای PATH اضافه کرده و در انتهای ان سمی کولن بگذارید و ذخیره کنید.(اگر درایوی که اپن سی وی داخل ان است C است به جای K ، C بگذارید.)



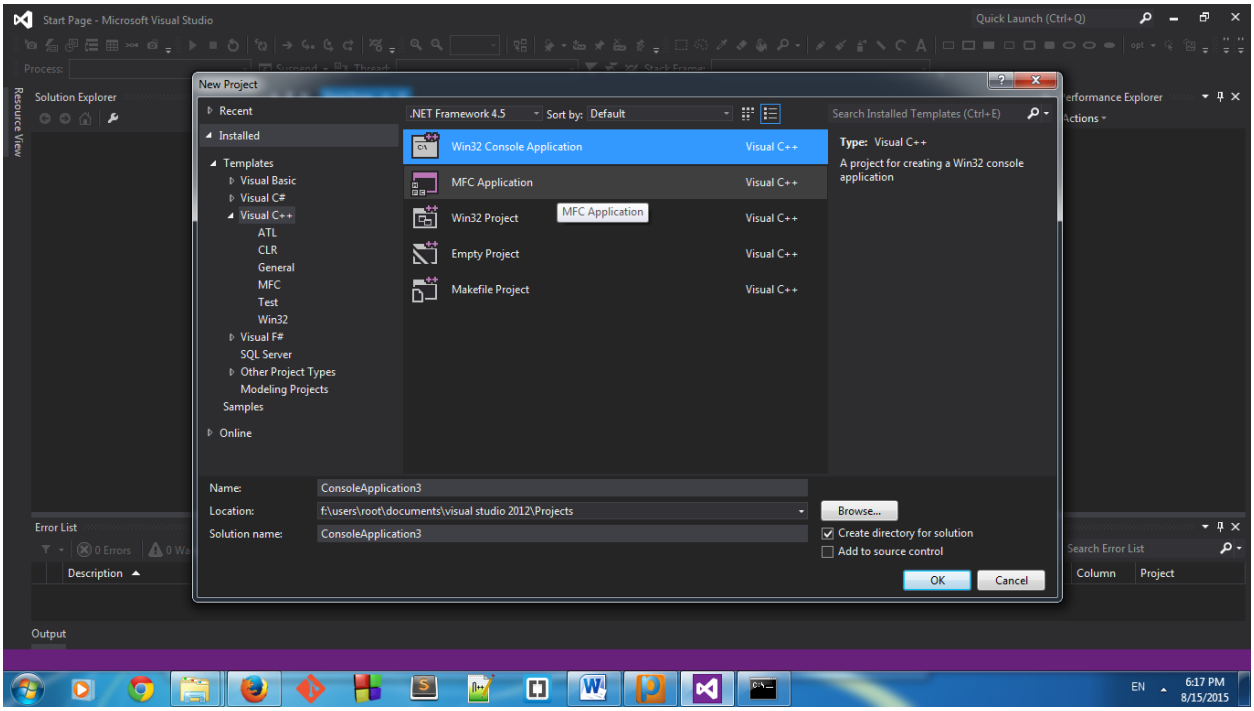


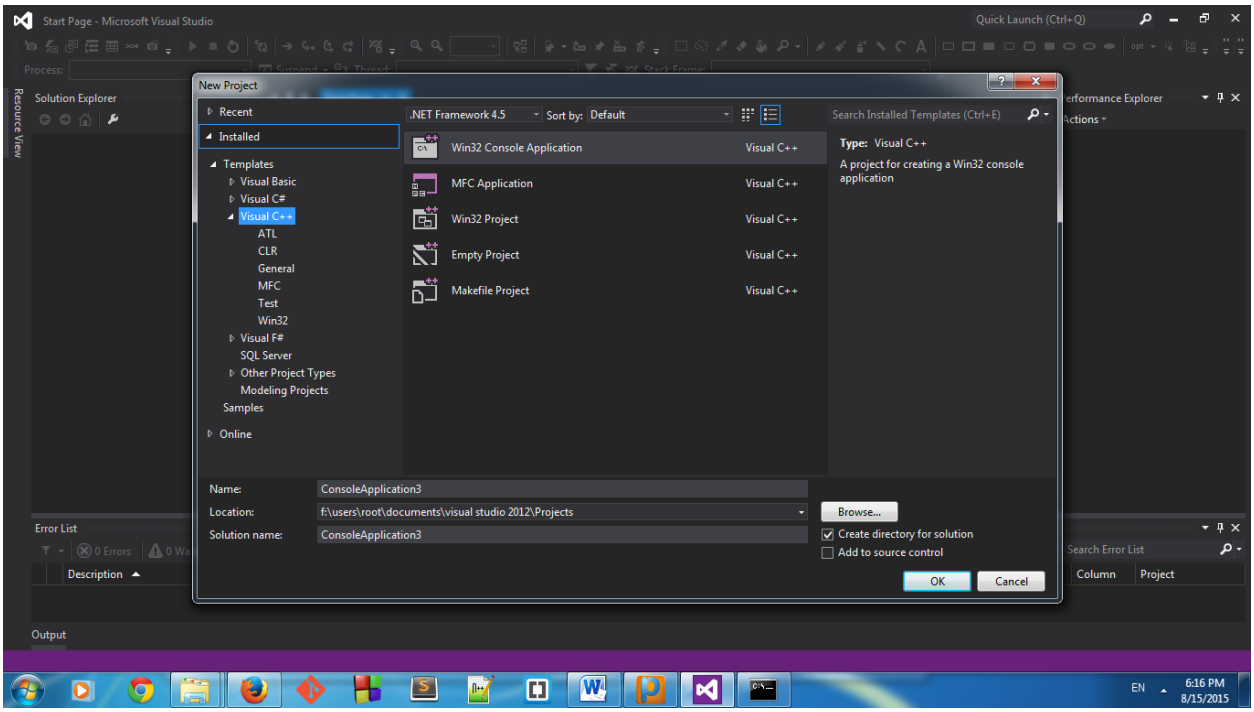


بعد از ان opencv_haartraining.exe داخل CMDتایپ کنید،اگر نتایج مانند عکس بود یعنی مرحله اول را درست رفته اید.

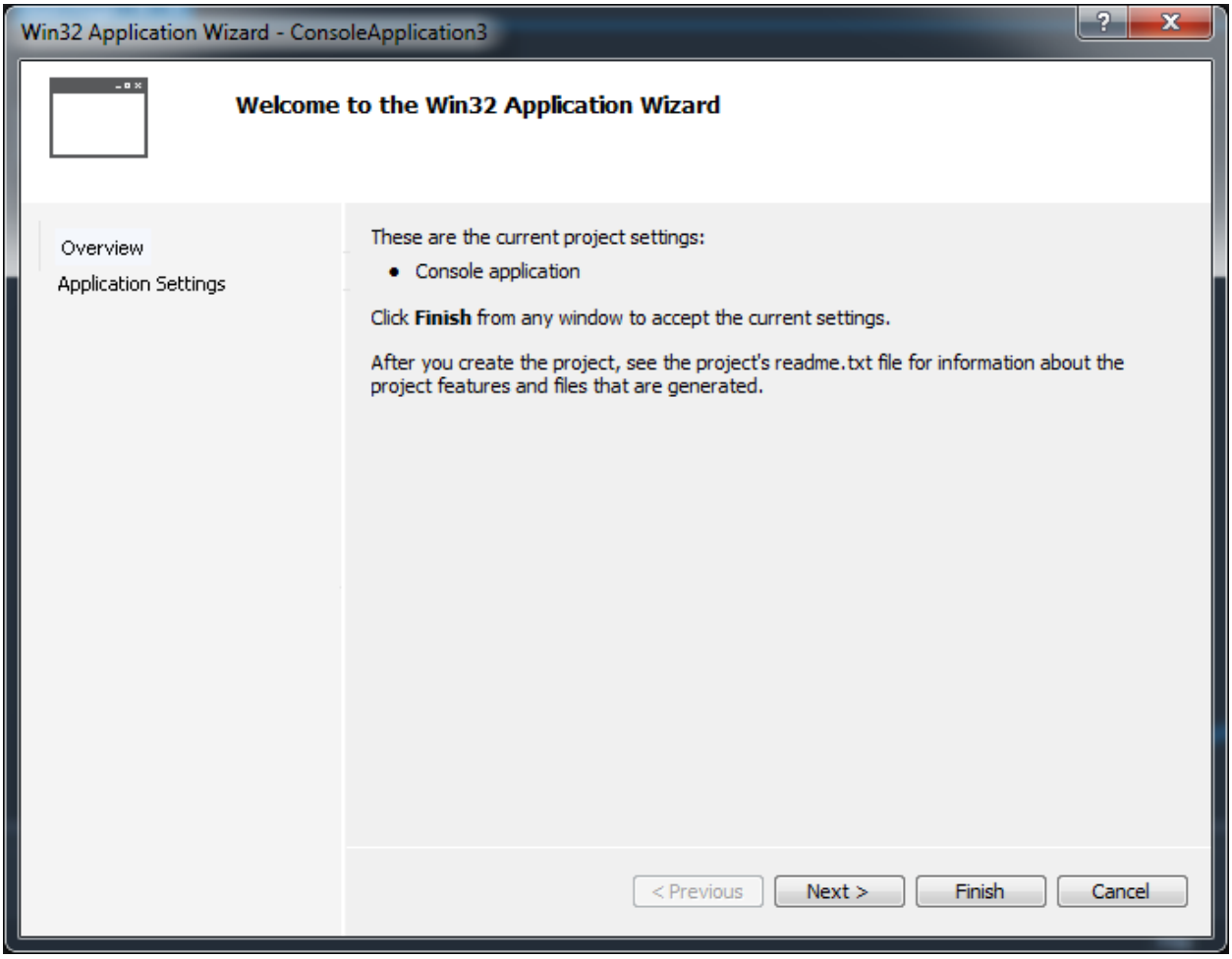


بعد از ویژوال استدیو 2012 را باز کنید و New project>Visual C++>Win 32 Console Aplication را انتخاب کنید.

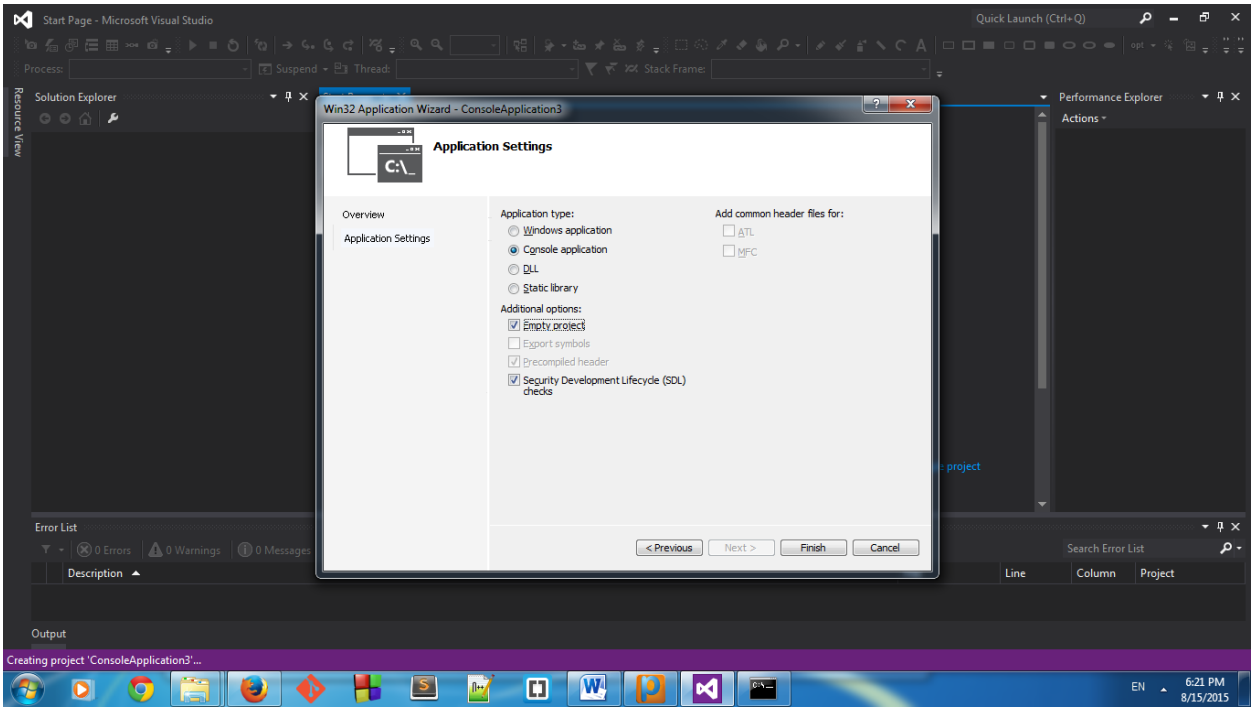

Start Page - Microsoft Visual Studio
Quick Launch (Ctrl+Q)
New Project
Recent
Installed
Templates
Visual Basic
Visual C#
Visual C++
ATL
CLR
General
MFC
Test
Win32
Visual F#
SQL Server
Other Project Types
Modeling Projects
Samples
Online
.NET Framework 4.5
Sort by: Default
Win32 Console Application
MFC Application
Win32 Project
Empty Project
Makefile Project
Visual C++
Type: Visual C++
A project for creating a Win32 console application
Name: ConsoleApplication3
Location: f:\users\root\documents\visual studio 2012\Projects
Solution name: ConsoleApplication3
Browse...
Create directory for solution
Add to source control
OK
Cancel
Solution Explorer
Error List
Output
EN
6:17 PM
8/15/2015

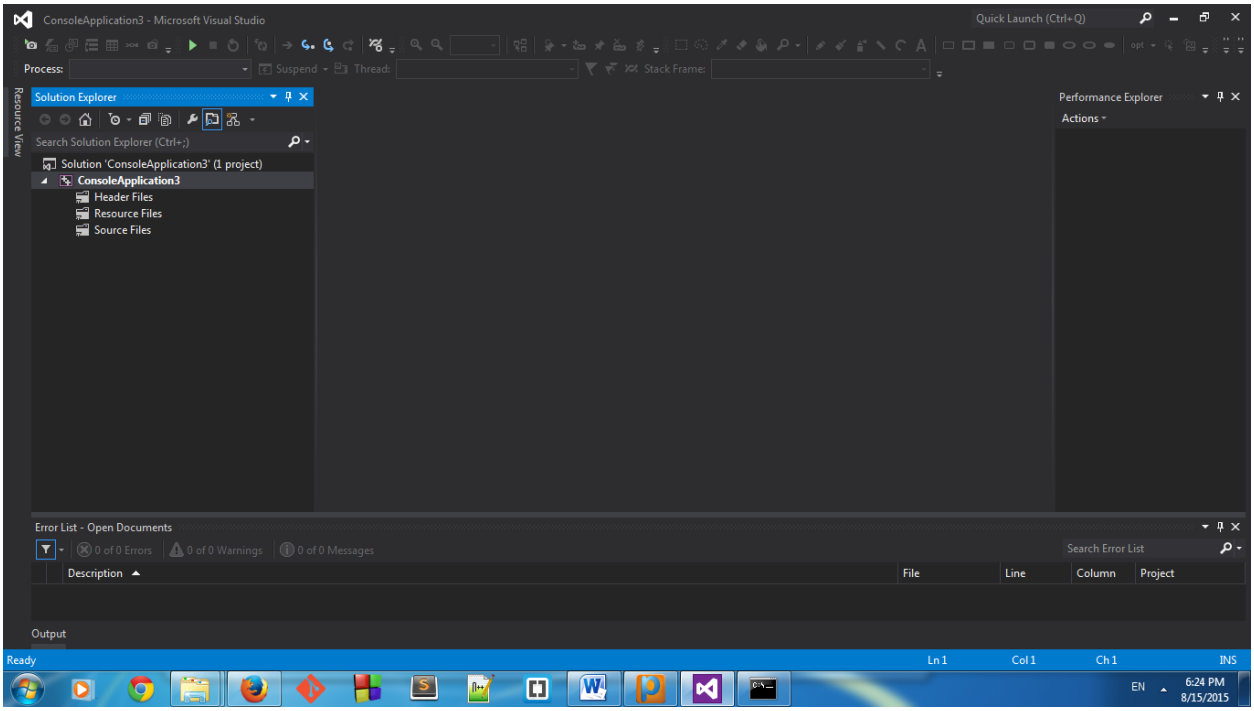

Start Page - Microsoft Visual Studio
Quick Launch (Ctrl+Q)
New Project
Recent
Installed
Templates
Visual Basic
Visual C#
Visual C++
ATL
CLR
General
MFC
Test
Win32
Visual F#
SQL Server
Other Project Types
Modeling Projects
Samples
Online
.NET Framework 4.5
Sort by: Default
Win32 Console Application
MFC Application
Win32 Project
Empty Project
Makefile Project
Visual C++
Type: Visual C++
A project for creating a Win32 console application
Name: ConsoleApplication3
Location: f:\users\root\documents\visual studio 2012\Projects
Solution name: ConsoleApplication3
Browse...
Create directory for solution
Add to source control
OK
Cancel
Solution Explorer
Error List
Output
EN
6:16 PM
8/15/2015

در مرحله بعد ،پنجره زیر باز می شود، Next را بزنید.

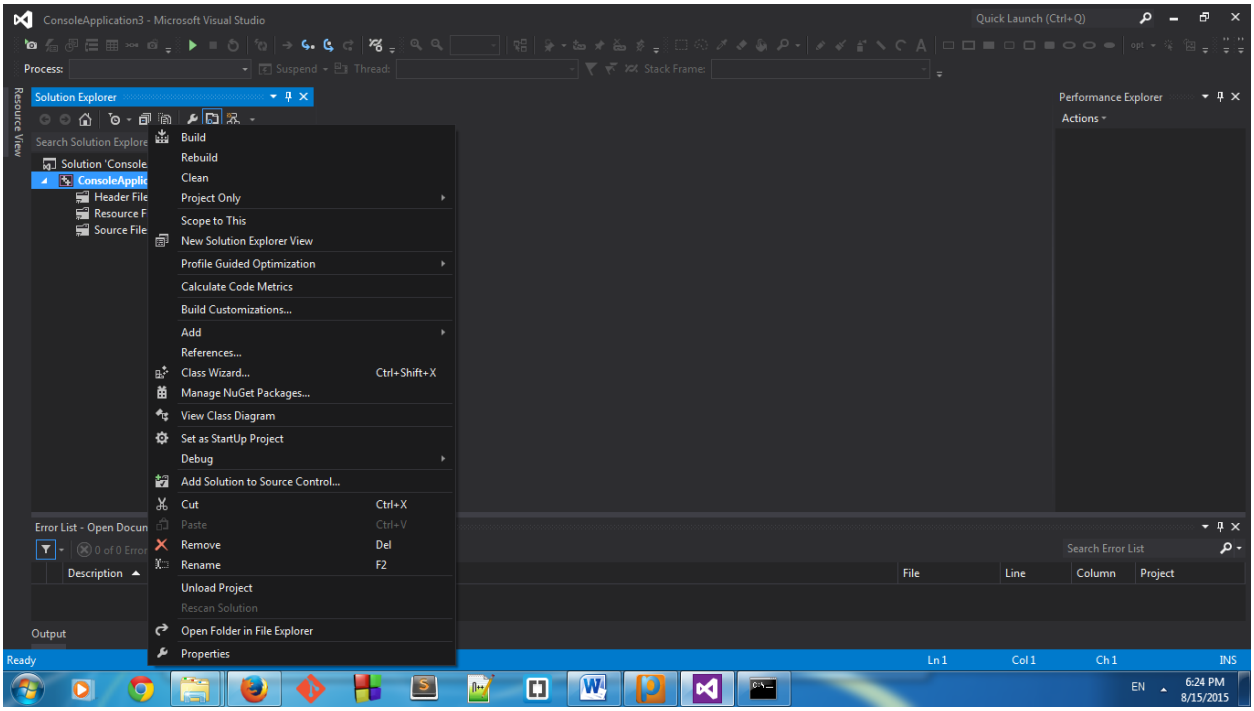


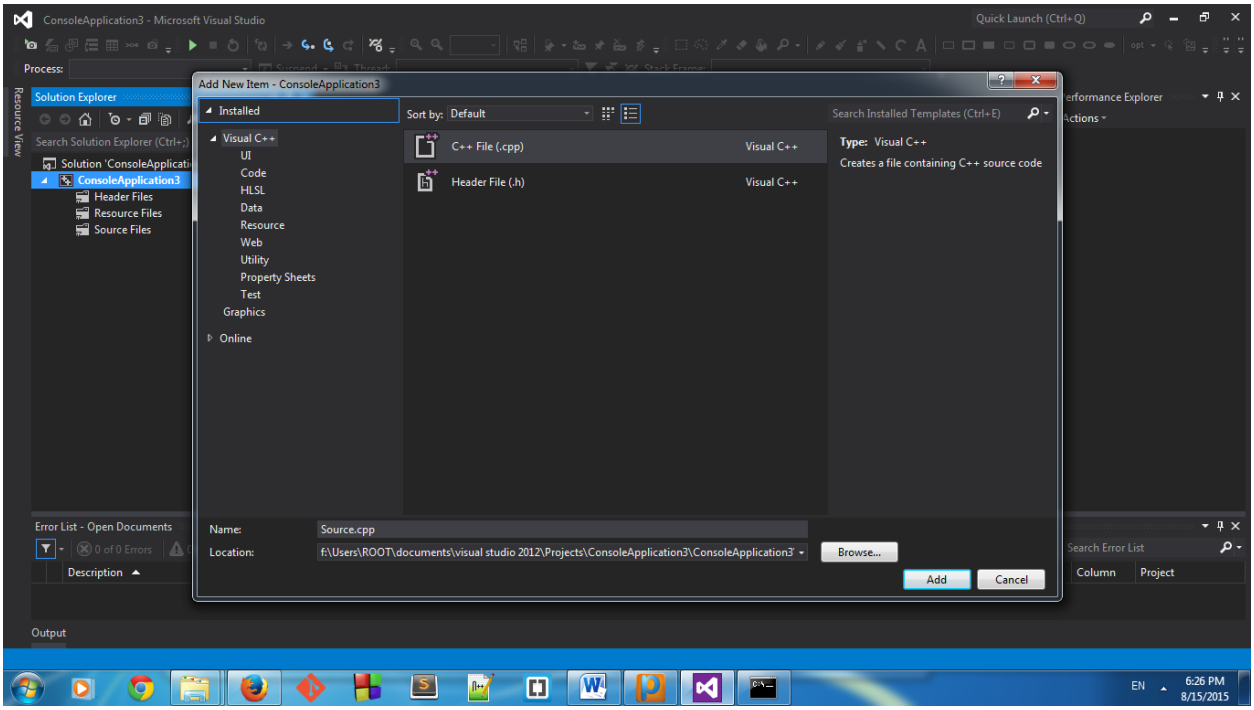
در مرحله بعد empty projectرا تیک و بعد Finish را بزنید.


Start Page - Microsoft Visual Studio
Quick Launch (Ctrl+Q)
Solution Explorer
Performance Explorer
Actions
Win32 Application Wizard - ConsoleApplication3
Application Settings
Overview
Application Settings
Application type:
Windows application
Console application
DLL
Static library
Additional options:
Empty project
Export symbols
Precompiled header
Security Development Lifecycle (SDL) checks
Add common header files for:
ATL
MFC
< Previous
Next >
Finish
Cancel
Error List
0 Errors
0 Warnings
0 Messages
Search Error List
Description
Line
Column
Project
Output
Creating project 'ConsoleApplication3'...
EN
6:21 PM
8/15/2015

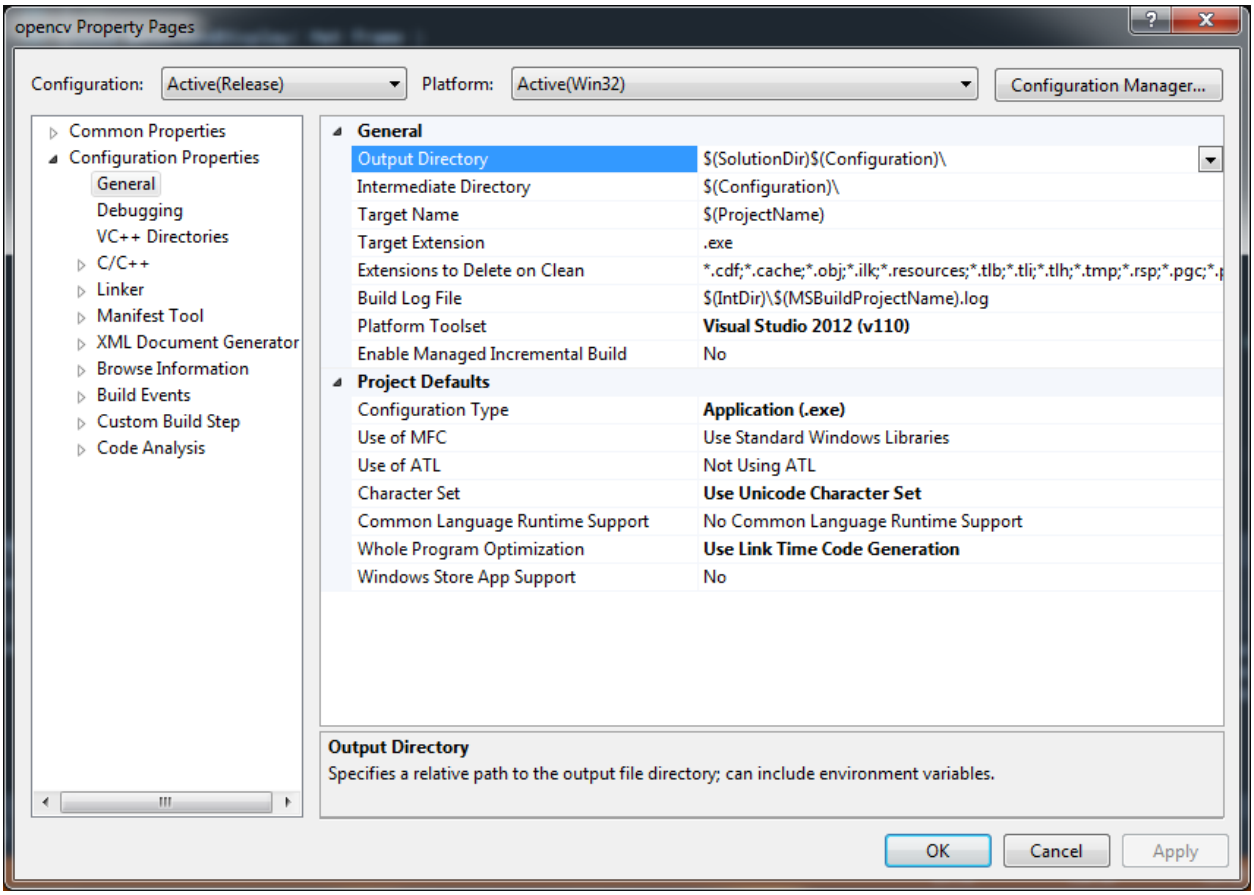

ConsoleApplication3 - Microsoft Visual Studio
Quick Launch (Ctrl+Q)
Solution Explorer
Search Solution Explorer (Ctrl+;)
Solution 'ConsoleApplication3' (1 project)
ConsoleApplication3
Header Files
Resource Files
Source Files
Performance Explorer
Actions
Error List - Open Documents
0 of 0 Errors
0 of 0 Warnings
0 of 0 Messages
Search Error List
Description
File
Line
Column
Project
Output
Ready
Ln 1
Col 1
Ch 1
INS
EN
6:24 PM
8/15/2015

بعد بر روی پروژه کلیک راست کرده، Add>New Item>Visual C++> C++ File را بزنید.





بعد از ان،روی پروژه خود راست کلیک کرده، Propertiesرا انتخاب کنید،در پنجره باز شده، C++>general را بزنید.

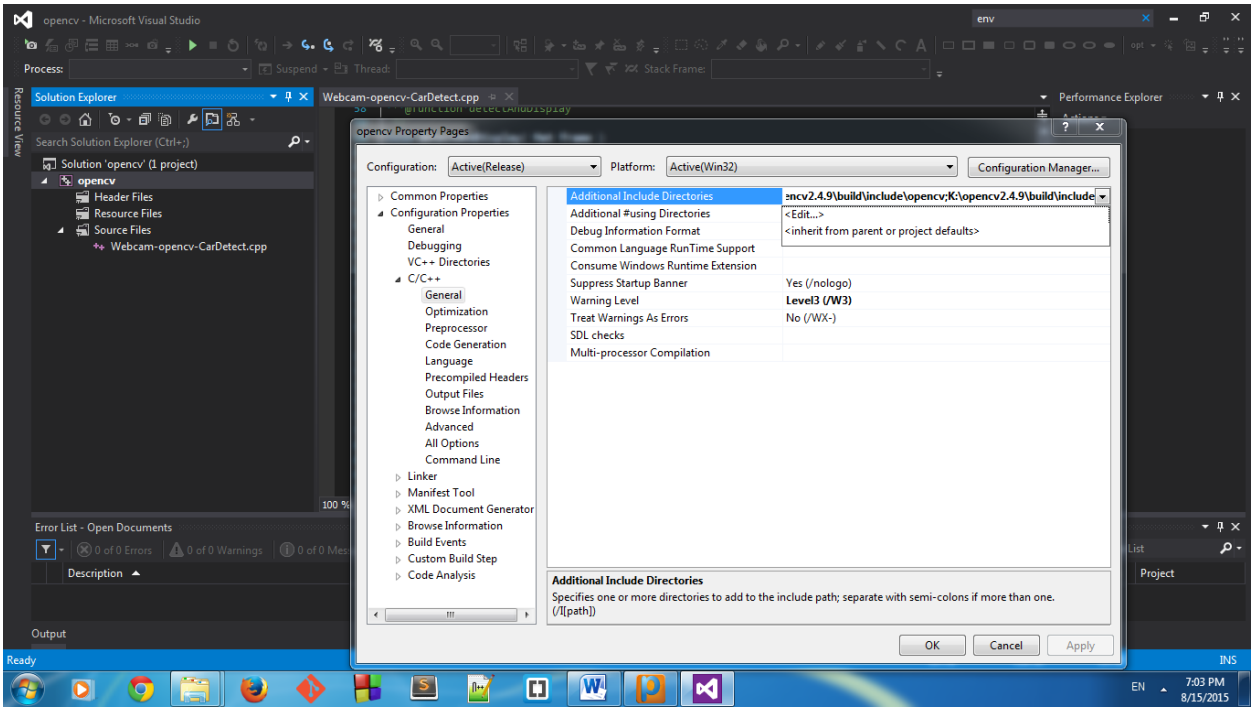


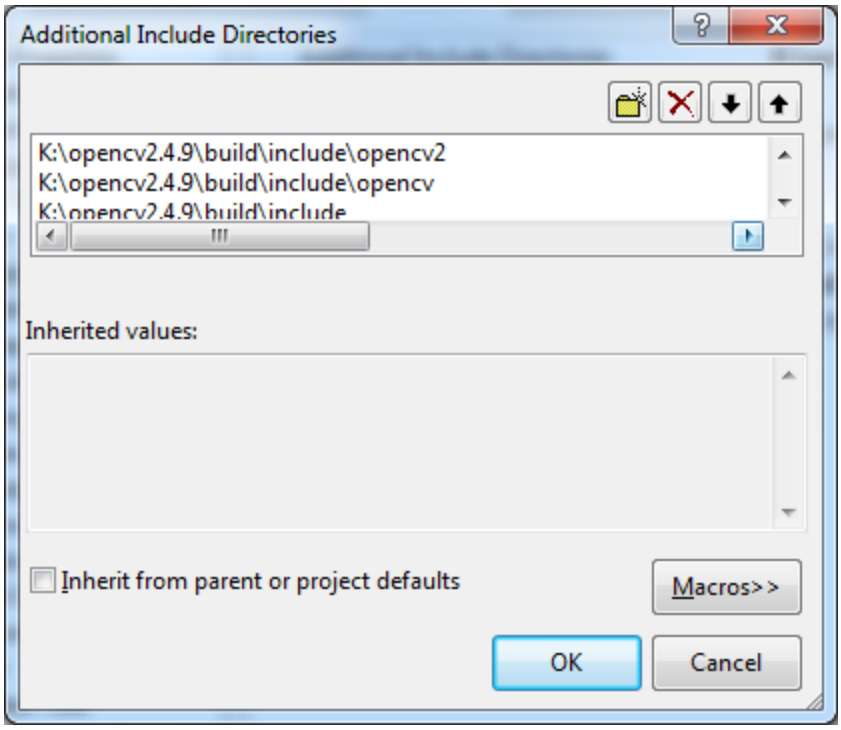
درقسمت Additional include directory این را کپی کنید.(اگر درایو یک چیزدیگه بود،اسم اون را بگذارید.)

K:\opencv2.4.9\build\include\opencv2

K:\opencv2.4.9\build\include

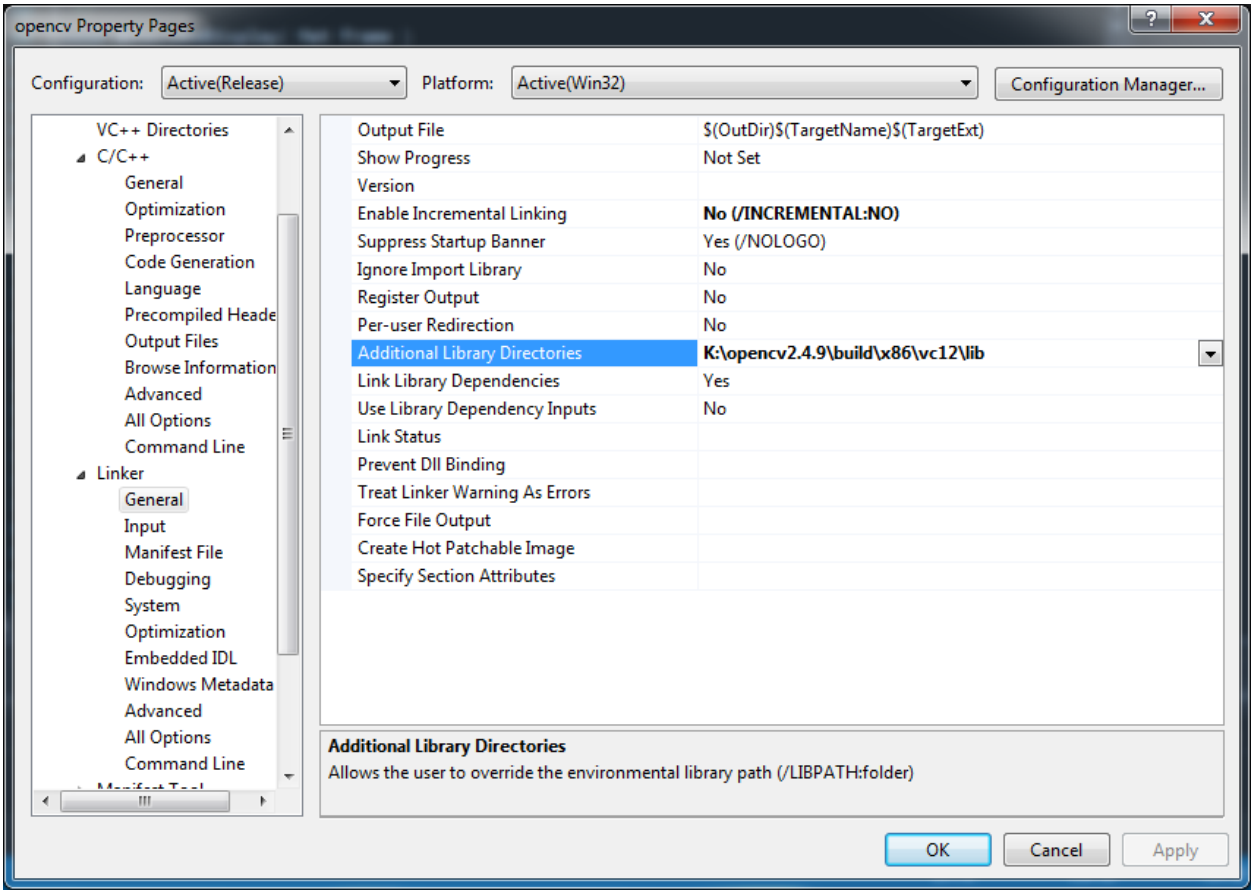
K:\opencv2.4.9\build\include\opencv





به قسمت لینکر بروید،در قسمت Additional library directory ،این را کپی کنید.

K:\opencv2.4.9\build\x86\vc12\lib



بعد از در قسمت linker>input رفته، در قسمت Additional dependencies ،عبارت زیر را کپی کنید.

opencv_calib3d249.lib

opencv_calib3d249d.lib

opencv_contrib249.lib

opencv_contrib249d.lib

opencv_core249.lib

opencv_core249d.lib

opencv_features2d249.lib

opencv_features2d249d.lib

opencv_flann249.lib

opencv_flann249d.lib

opencv_gpu249.lib

opencv_gpu249d.lib

opencv_highgui249.lib

opencv_highgui249d.lib

opencv_imgproc249.lib

opencv_imgproc249d.lib

opencv_legacy249.lib

opencv_legacy249d.lib

opencv_ml249.lib

opencv_ml249d.lib

opencv_nonfree249.lib

opencv_nonfree249d.lib

opencv_objdetect249.lib

opencv_objdetect249d.lib

opencv_ocl249.lib

opencv_ocl249d.lib

opencv_photo249.lib

opencv_photo249d.lib

opencv_stitching249.lib

opencv_stitching249d.lib

opencv_superres249.lib

opencv_superres249d.lib

opencv_ts249.lib

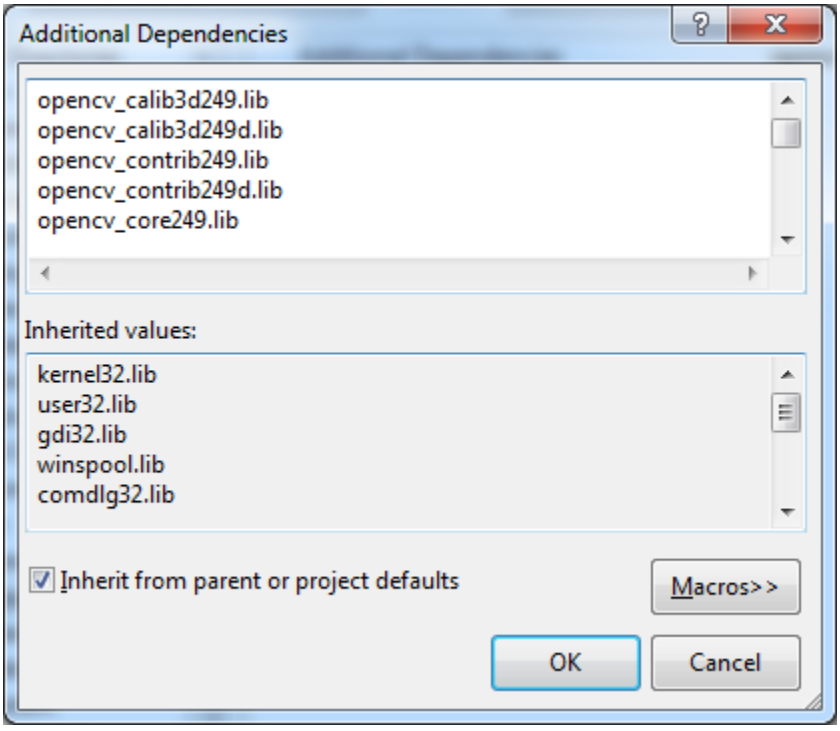
opencv_ts249d.lib

opencv_video249.lib

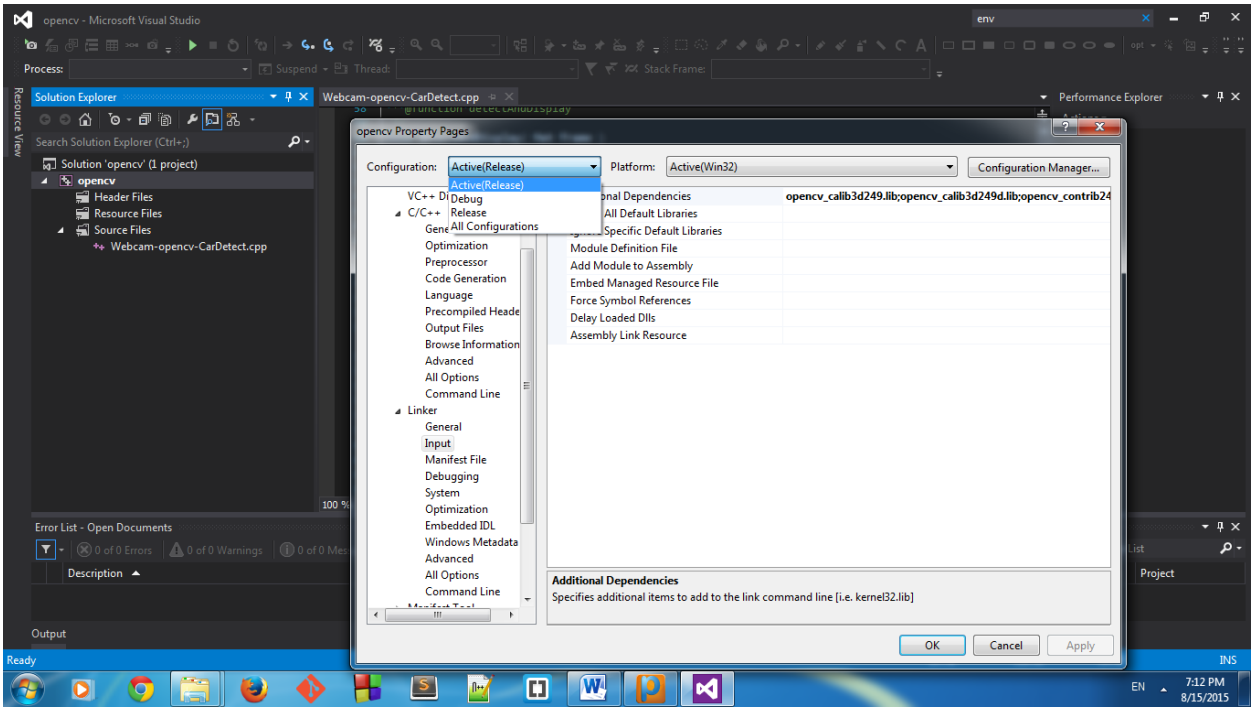
opencv_video249d.lib

opencv_videostab249.lib

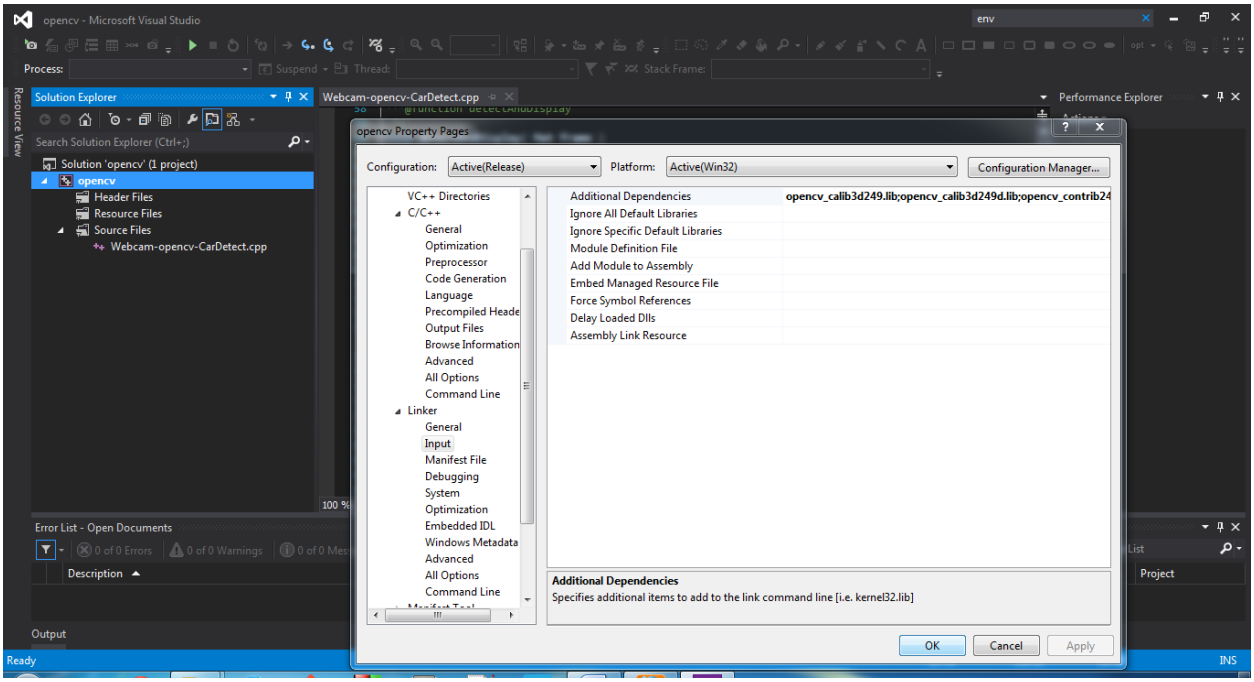
opencv_videostab249d.lib



همه این کار ها،در مد Releaseبود ،مد را به حالت Debugبرده و همه ی این مراحل را دوباره رفته،



بعد یادتون باشه، Platform هم از همان اول کار روی Win32 باشد ،بعد هم ok می زنیم.

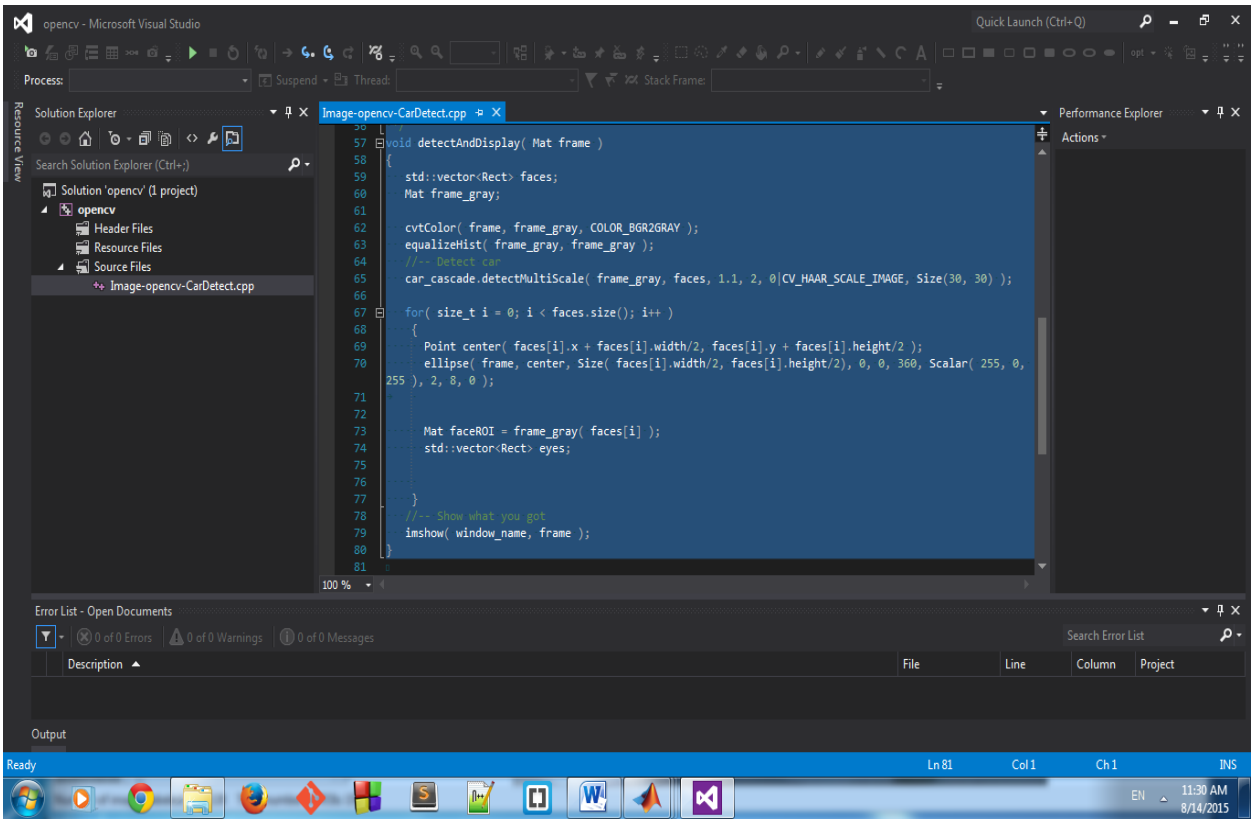


حالا در سورس اماده نوشتن کد خود هستیم،حال برای چک کردن اینکه کار می کند،سورس زیر را در ان کپی کنید،بعد یک عکس با نام image.jpgدر پوشه پروژه قسمت سورس ان کپی کنید،باید یک عکس خروجی را به ما نشان دهد و اگر ماشین در ان است باید برای ما تشخیص دهد.

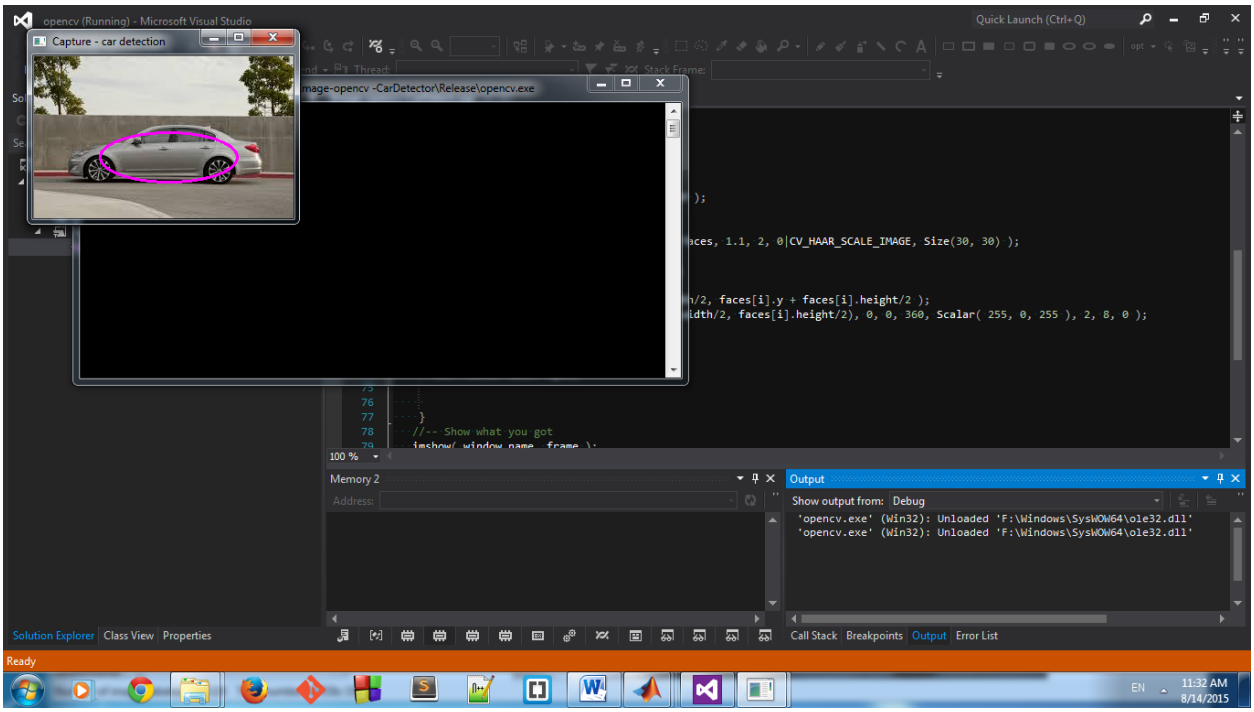
برای تشخیص ماشین در عکس در محیط اپن سی وی از سورس کد زیر استفاده می کنیم،لازم به ذکر است opencv 2.4.9در ای دی ای ویژوال استدیو 2012 پیکره بندی شده است.باید فایل CarModel.xml و image.jpg را در پوشه سورس خود کپی کنیم.

```
/**
 Image CarDetect By R.Borumandi  Shiraz University
 */
#include "opencv2/objdetect/objdetect.hpp"
#include "opencv2/highgui/highgui.hpp"
face_cascade_name
#include "opencv2/imgproc/imgproc.hpp"

#include <iostream>
#include <stdio.h>

using namespace std;
using namespace cv;

/** Function Headers */
void detectAndDisplay( Mat frame );

/** Global variables */
//-- Note, either copy these two files from opencv/data/haarscascades to your current
folder, or change these locations
String car_cascade_name = "CarModel.xml";


CascadeClassifier car_cascade;

string window_name = "Capture - car detection";
RNG rng(12345);

/**
 * @function main
 */
int main( void )
{
```

```
  Mat frame;
  string imageName("image.jpg");

  frame= imread(imageName.c_str(), IMREAD_COLOR);

  //-- 1. Load the cascades
  if( !car_cascade.load( car_cascade_name ) ){ printf("--(!)Error loading\n"); return -1;
};


      //-- 3. Apply the classifier to the frame

       detectAndDisplay( frame );



   waitKey(0);

  return 0;
}

/**
 * @function detectAndDisplay
 */
void detectAndDisplay( Mat frame )
{
   std::vector<Rect> faces;
   Mat frame_gray;

   cvtColor( frame, frame_gray, COLOR_BGR2GRAY );
   equalizeHist( frame_gray, frame_gray );
   //-- Detect car
   car_cascade.detectMultiScale( frame_gray, faces, 1.1, 2, 0|CV_HAAR_SCALE_IMAGE,
Size(30, 30) );

   for( size_t i = 0; i < faces.size(); i++ )
    {
      Point center( faces[i].x + faces[i].width/2, faces[i].y + faces[i].height/2 );
      ellipse( frame, center, Size( faces[i].width/2, faces[i].height/2), 0, 0, 360,
Scalar( 255, 0, 255 ), 2, 8, 0 );


      Mat faceROI = frame_gray( faces[i] );
      std::vector<Rect> eyes;


    }
   //-- Show what you got
   imshow( window_name, frame );
}
```



خروجی ان به صورت زیر است:



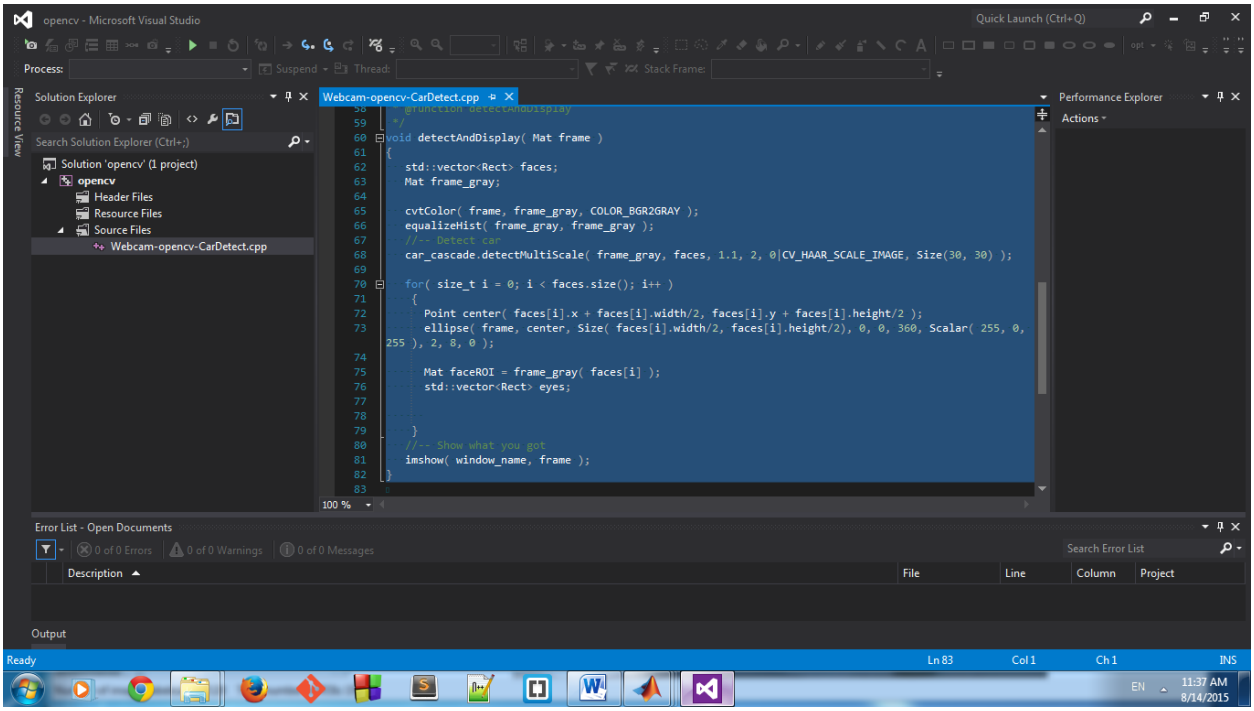
پس از این به بعد برای ساخت پروژه اپن سی وی مراحل قبل را رفته،ودر قسمت سورس ان را کپی کرده البته باید توجه داشته باشیم،اگر حوصله ندارید هردفعه همه مراحل را بروید از یکی که درست کردید،کپی کنید،و برای مراحل بعد فقط سورس ان را عوض کنید،چون این تنظیمات در فایل های مایکروسافت همراه پروژه ذخیره شده است.(لازم به ذکر دو پروژه اپن سی وی در فولدر پروژه در قسمت opencv CarDetection قرار داده شده است.

برای تشخیص عکس ماشین در محیط اپن سی وی از طریق وب کم از کد زیر استفاده می کنیم.(یادتون نره مدل CarModel را داخل پوشه سورس بگذارید.)

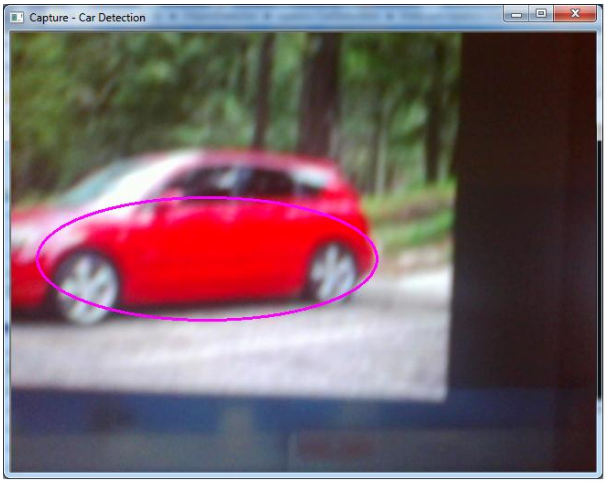
```
/**
 Car Detection  Project By R.Borumandi Shiraz University
 */
#include "opencv2/objdetect/objdetect.hpp"
#include "opencv2/highgui/highgui.hpp"
#include "opencv2/imgproc/imgproc.hpp"

#include <iostream>
#include <stdio.h>

using namespace std;
using namespace cv;

/** Function Headers */
void detectAndDisplay( Mat frame );

/** Global variables */
//-- Note, either copy these two files from opencv/data/haarscascades to your current
folder, or change these locations
String car_cascade_name = "CarModel.xml";
CascadeClassifier car_cascade;
string window_name = "Capture - Car Detection";
RNG rng(12345);
```

```
/**
 * @function main
 */
int main( void )
{
  VideoCapture capture;
  Mat frame;

  //-- 1. Load the cascades
  if( !car_cascade.load( car_cascade_name ) ){ printf("--(!)Error loading\n"); return -1;
};

  //-- 2. Read the video stream
  capture.open( 0 );
  if( capture.isOpened() )
  {
    for(;;)
    {
      capture >> frame;

      //-- 3. Apply the classifier to the frame
      if( !frame.empty() )
       { detectAndDisplay( frame ); }
      else
       { printf(" --(!) No captured frame -- Break!"); break; }

      int c = waitKey(10);
      if( (char)c == 'c' ) { break; }

    }
  }
  return 0;
}

/**
 * @function detectAndDisplay
 */
void detectAndDisplay( Mat frame )
{
   std::vector<Rect> faces;
   Mat frame_gray;

   cvtColor( frame, frame_gray, COLOR_BGR2GRAY );
   equalizeHist( frame_gray, frame_gray );
   //-- Detect car
   car_cascade.detectMultiScale( frame_gray, faces, 1.1, 2, 0|CV_HAAR_SCALE_IMAGE,
Size(30, 30) );

   for( size_t i = 0; i < faces.size(); i++ )
    {
      Point center( faces[i].x + faces[i].width/2, faces[i].y + faces[i].height/2 );
      ellipse( frame, center, Size( faces[i].width/2, faces[i].height/2), 0, 0, 360,
Scalar( 255, 0, 255 ), 2, 8, 0 );

      Mat faceROI = frame_gray( faces[i] );
      std::vector<Rect> eyes;
```

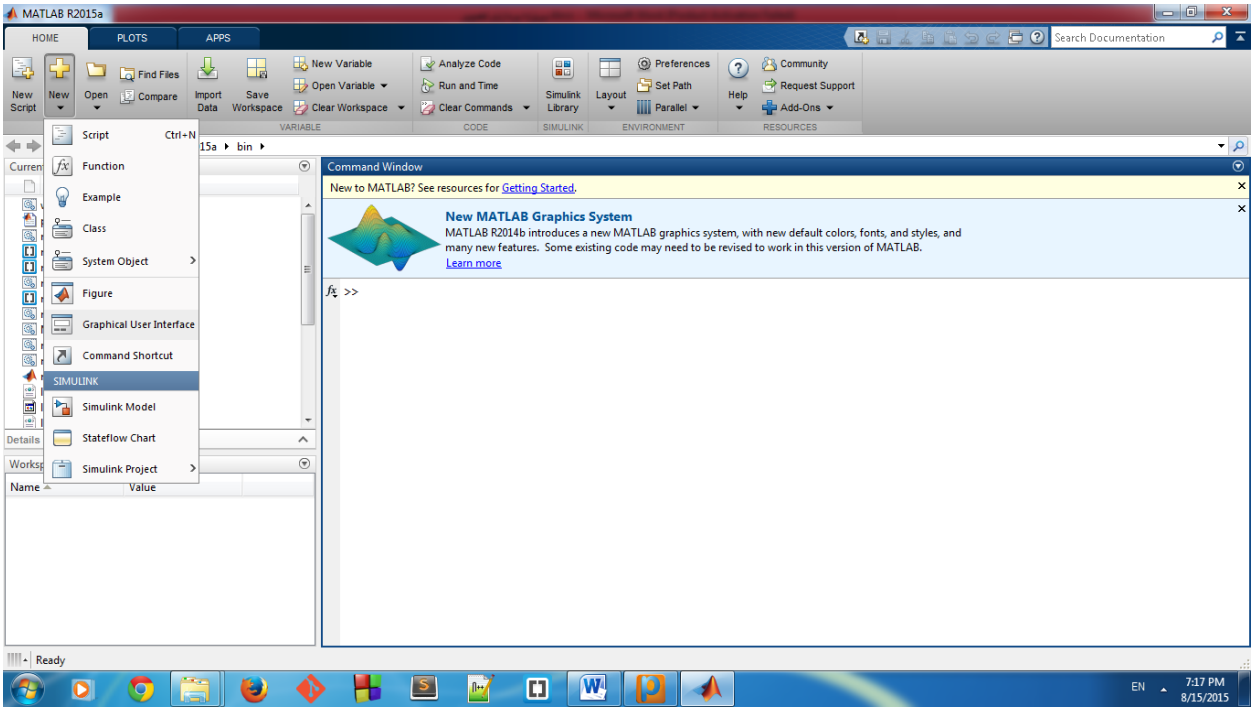
```
    }
   //-- Show what you got
   imshow( window_name, frame );
}
```

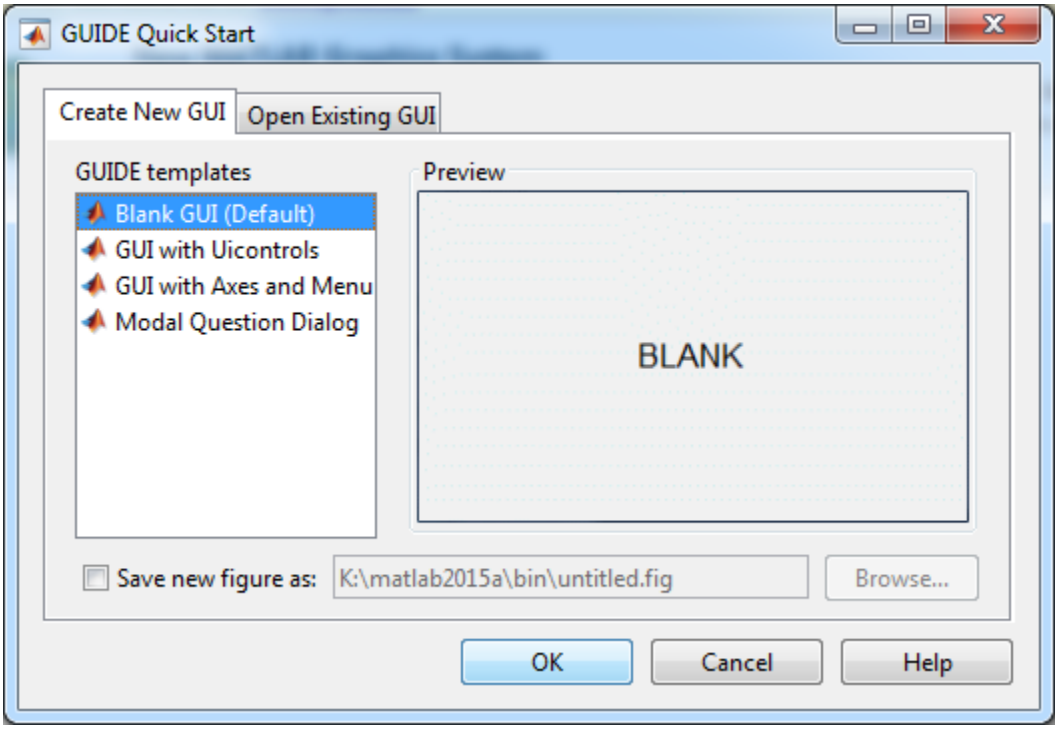


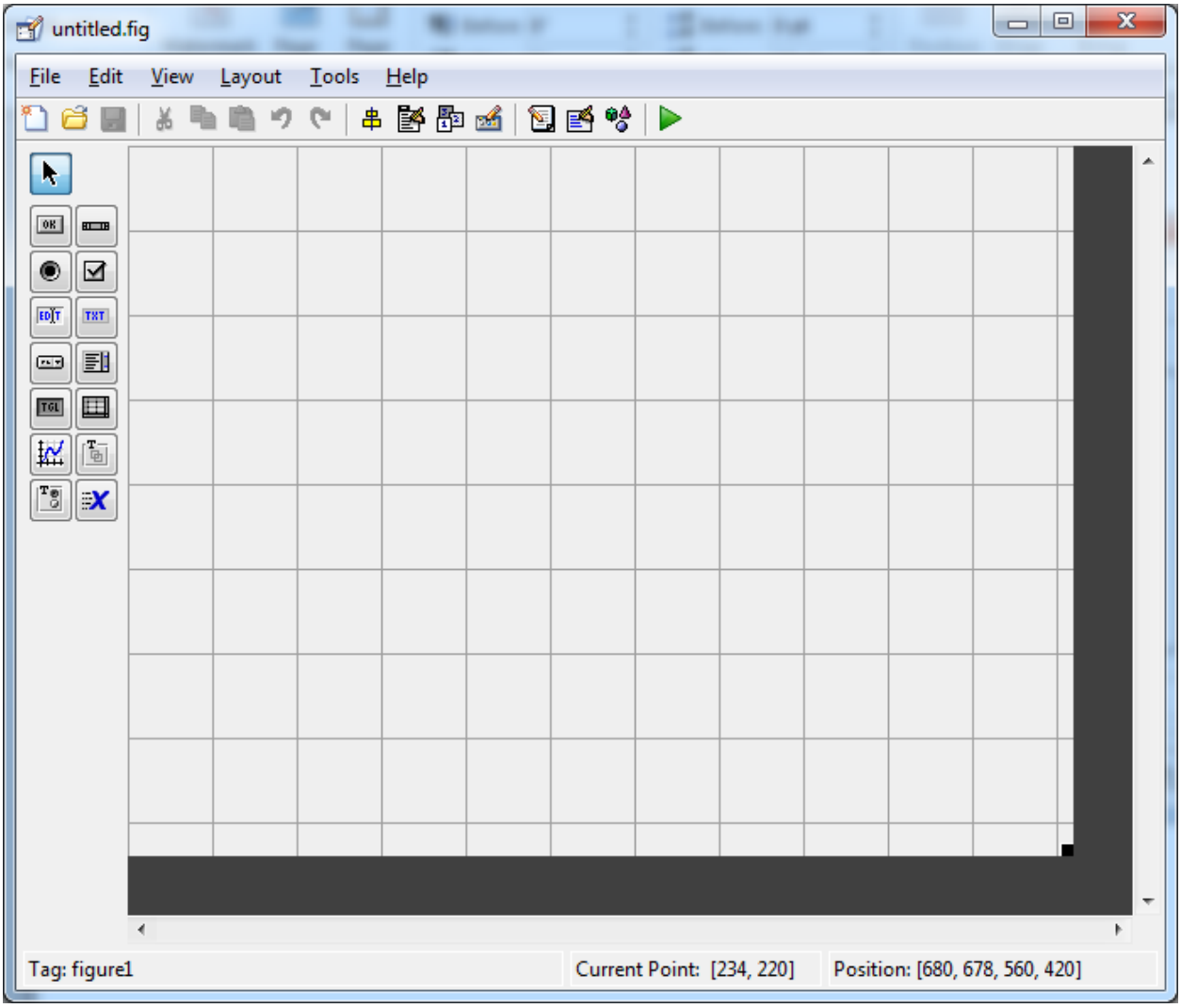
خروجی ان چنین است:



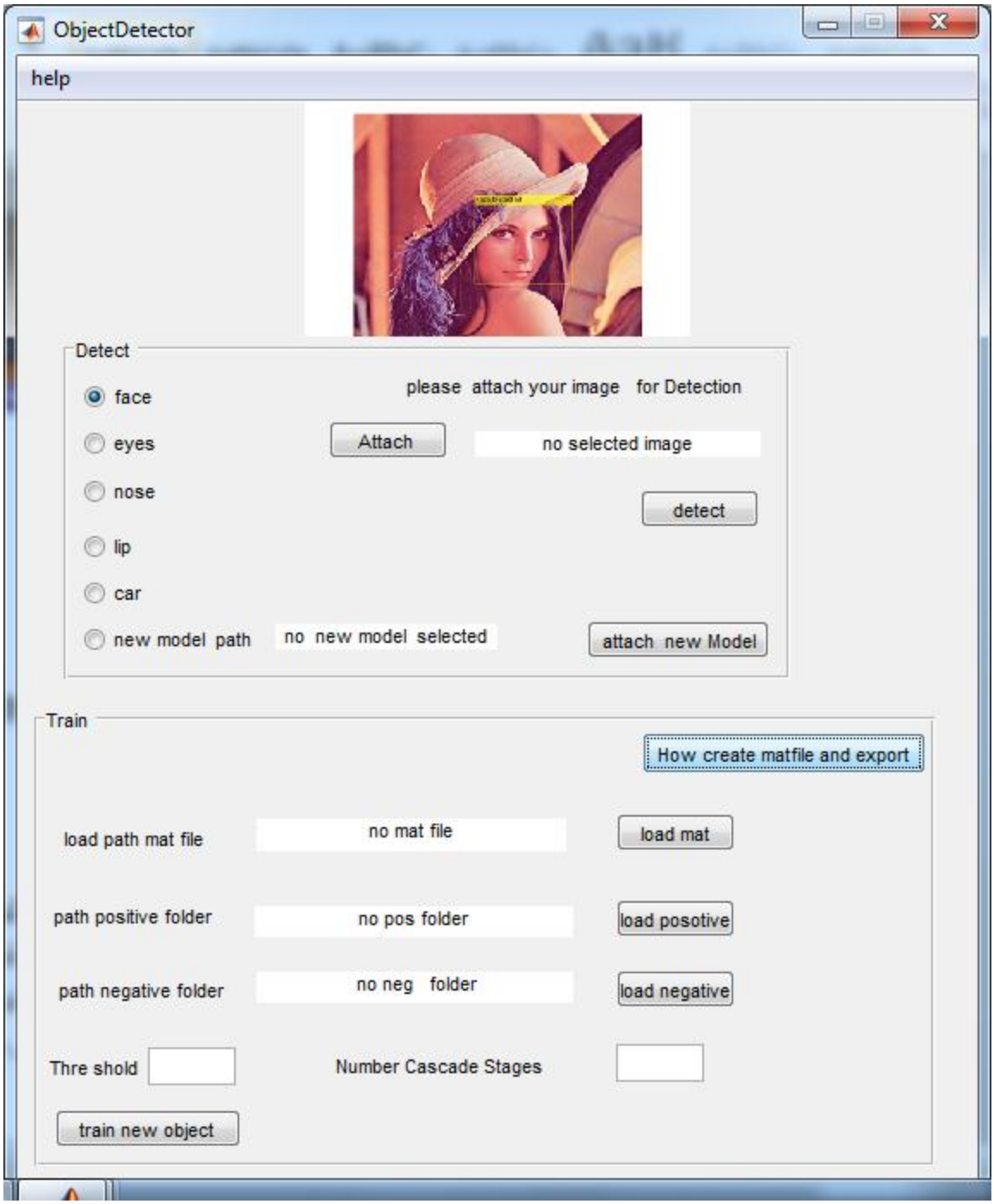
در نهایت ما از GUIمتلب استفاده کرده،یک مینی تولباکس نوشتیم که کار تشخیص اشیا و اموزش مدل جدید را برای ما بسیار ساده می کند،شما با این تولباکس به راحتی می توانید هر مدل جدیدی با استفاده از الگوریتم ویولا جونز اموزش دهید.عکس های Object Detectorرا در زیر می بینید.برای ساختن GUI می توانید،از منوی New>Graphic User Interface یک فایل figبسازید.


MATLAB R2015a
HOME
PLOTS
APPS
Search Documentation
New Script
New
Open
Find Files
Compare
Import Data
Save Workspace
New Variable
Open Variable
Clear Workspace
Analyze Code
Run and Time
Clear Commands
Simulink Library
Layout
Preferences
Set Path
Parallel
Help
Community
Request Support
Add-Ons
VARIABLE
CODE
SIMULINK
ENVIRONMENT
RESOURCES
Script Ctrl+N
Function
Example
Class
System Object
Figure
Graphical User Interface
Command Shortcut
SIMULINK
Simulink Model
Stateflow Chart
Simulink Project
Command Window
New to MATLAB? See resources for Getting Started.
New MATLAB Graphics System
MATLAB R2014b introduces a new MATLAB graphics system, with new default colors, fonts, and styles, and many new features. Some existing code may need to be revised to work in this version of MATLAB.
Learn more
>>
Name
Value
Ready
EN
7:17 PM
8/15/2015

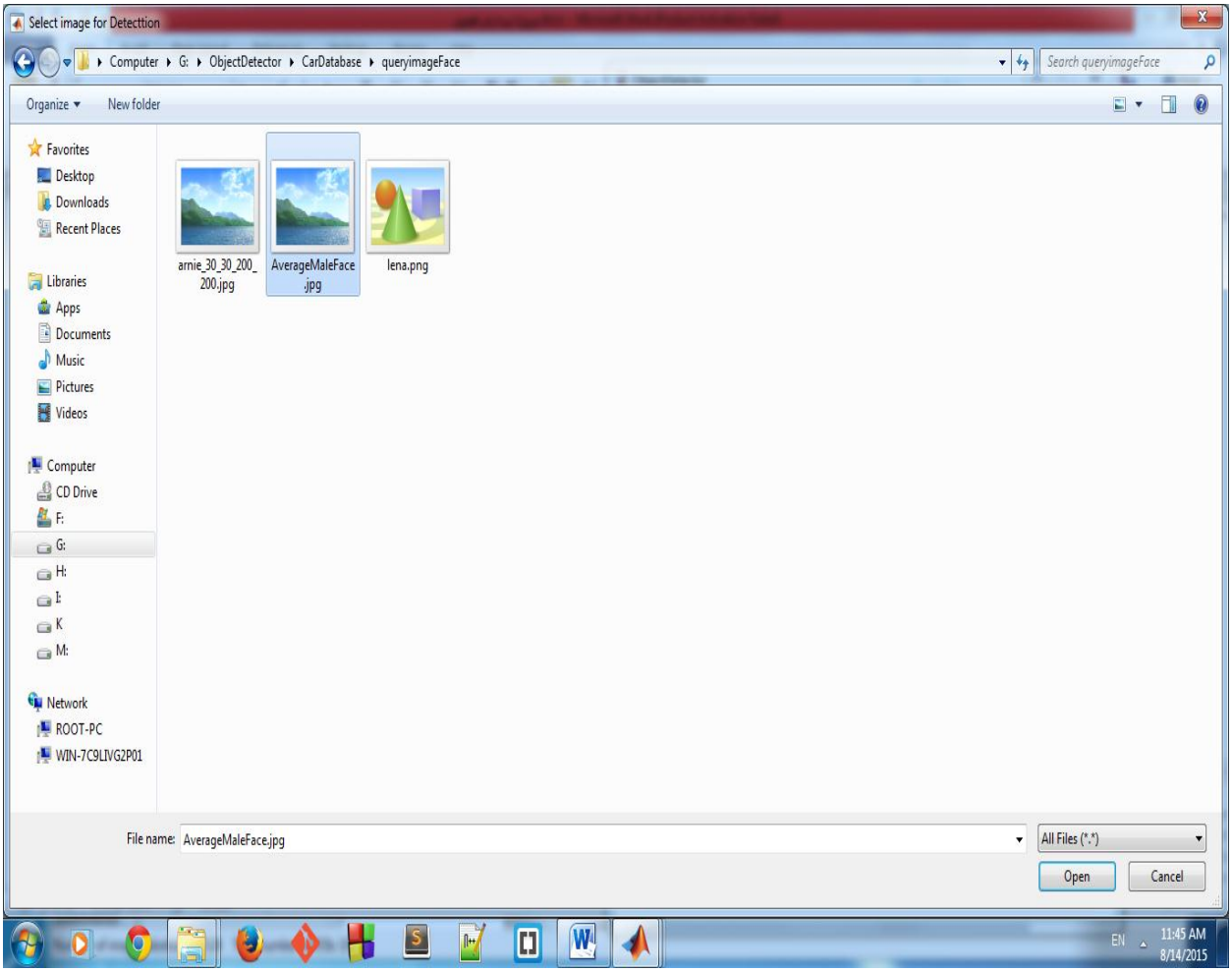

GUIDE Quick Start
Create New GUI
Open Existing GUI
GUIDE templates
Blank GUI (Default)
GUI with Uicontrols
GUI with Axes and Menu
Modal Question Dialog
Preview
BLANK
Save new figure as:
K:\matlab2015a\bin\untitled.fig
Browse...
OK
Cancel
Help

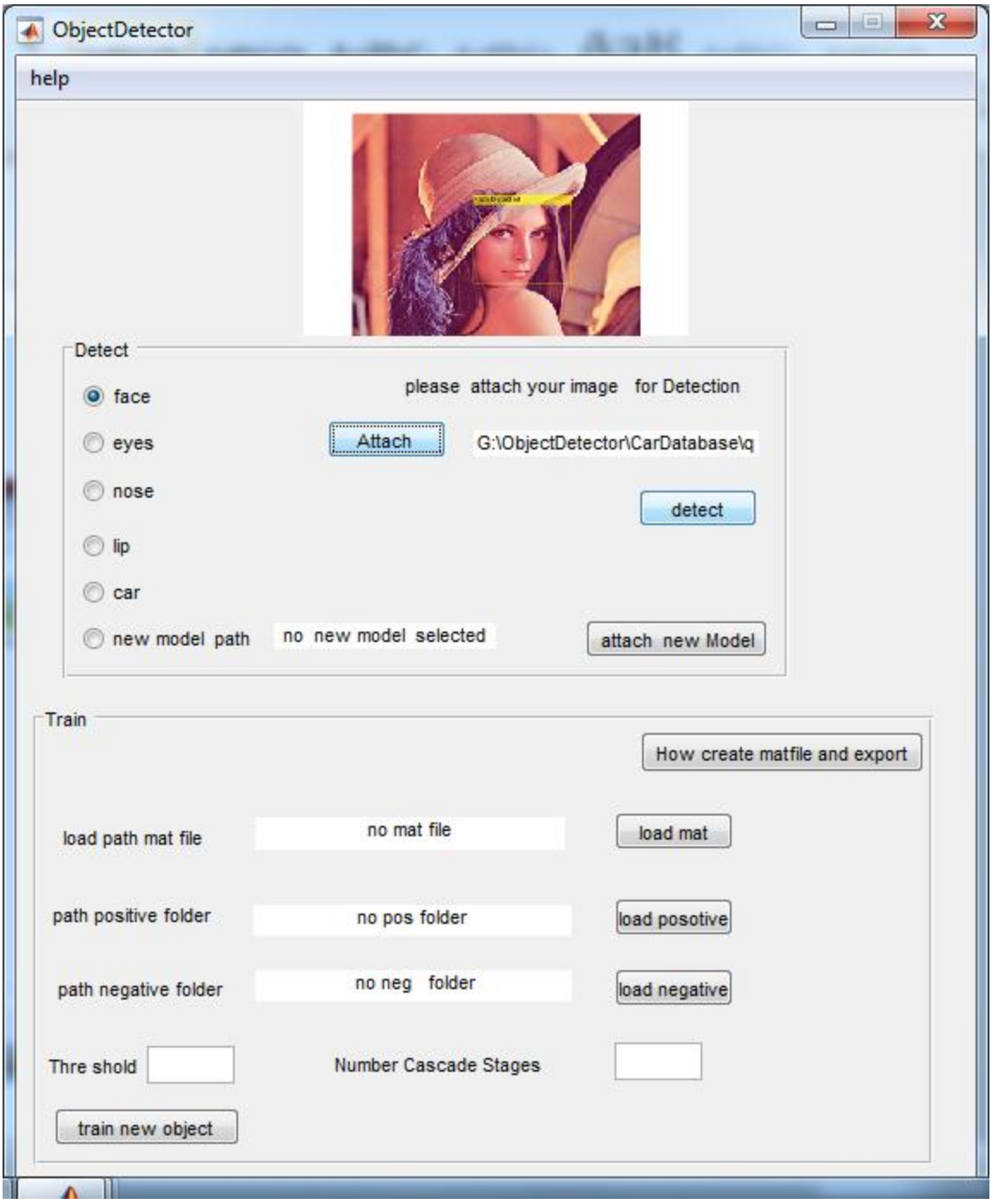


که برای اجزای گرافیکی باید کد نویسی کرد،برای اموزش نحوه کد نویسی به سایت www.mathworks.com *مراجعه کنید.نحوی استفاده از تولباکس را شرح می دهیم.فایل* ObjectDetector.mرا که در پوشه پروژه می باشد،باز کرده،وبا متلب اجرا می کنیم.



در قسمت Detectکافی است عکس را لود کنید،و بگید چه چیزی قرار است ،تشخیص بده،در قسمت اموزش هم با گرفتن پارامتر ی لازم برای شما مدل مد نظرتون را اموزش می دهد.


Select image for Detecttion
Computer ▸ G: ▸ ObjectDetector ▸ CarDatabase ▸ queryimageFace
Search queryimageFace
Organize
New folder
Favorites
Desktop
Downloads
Recent Places
Libraries
Apps
Documents
Music
Pictures
Videos
Computer
CD Drive
F:
G:
H:
I:
K
M:
Network
ROOT-PC
WIN-7C9LIVG2P01
arnie_30_30_200_200.jpg
AverageMaleFace.jpg
lena.png
File name: AverageMaleFace.jpg
All Files (*.*)
Open
Cancel
EN
11:45 AM
8/14/2015


ObjectDetector
help
Detect
face
eyes
nose
lip
car
new model path
no new model selected
please attach your image for Detection
Attach
G:\ObjectDetector\CarDatabase\q
detect
attach new Model
Train
How create matfile and export
load path mat file
no mat file
load mat
path positive folder
no pos folder
load posotive
path negative folder
no neg folder
load negative
Thre shold
Number Cascade Stages
train new object

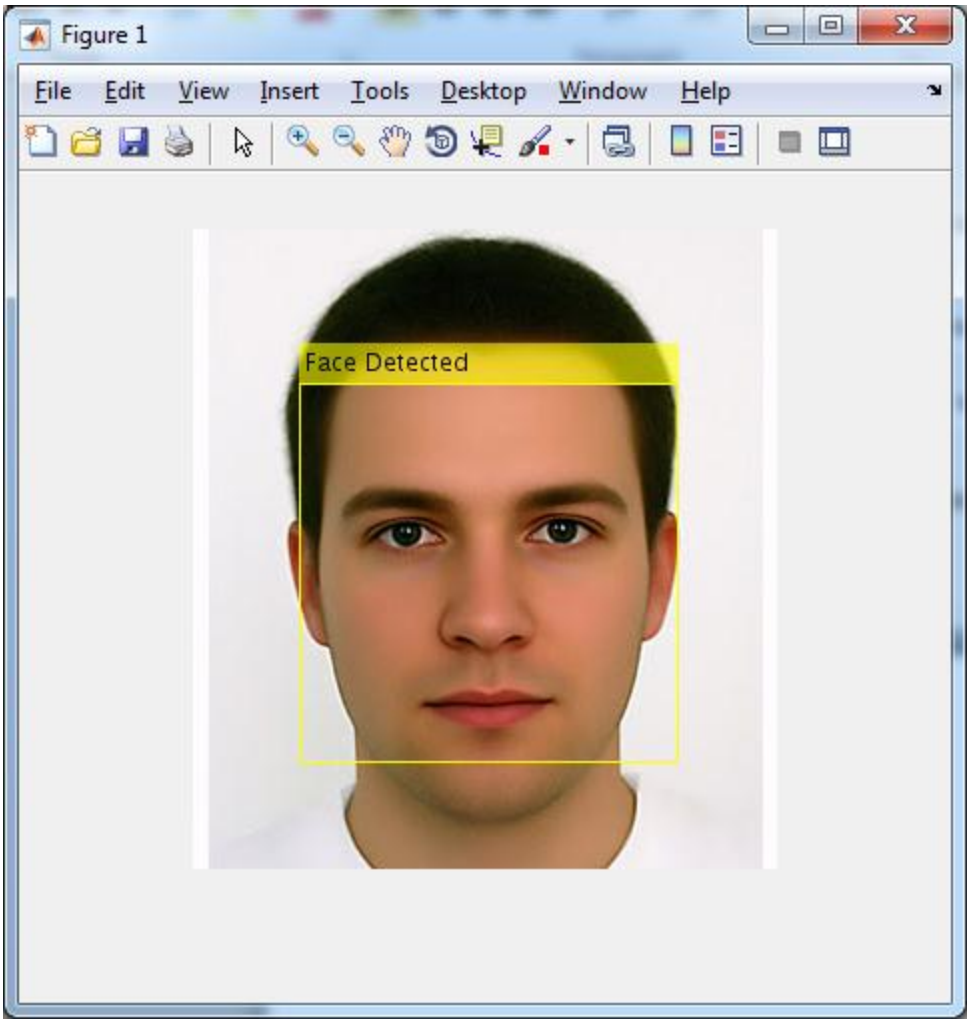
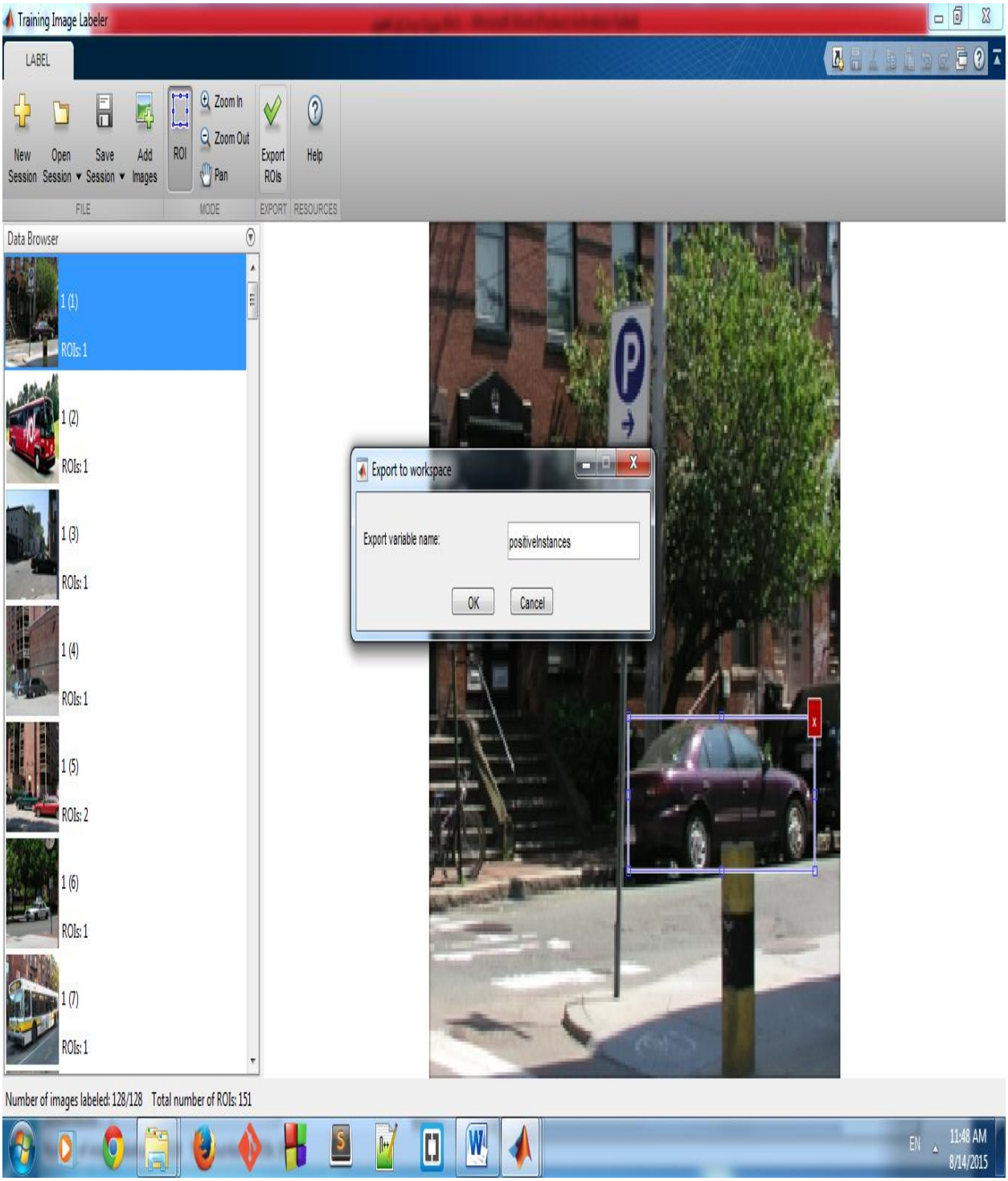
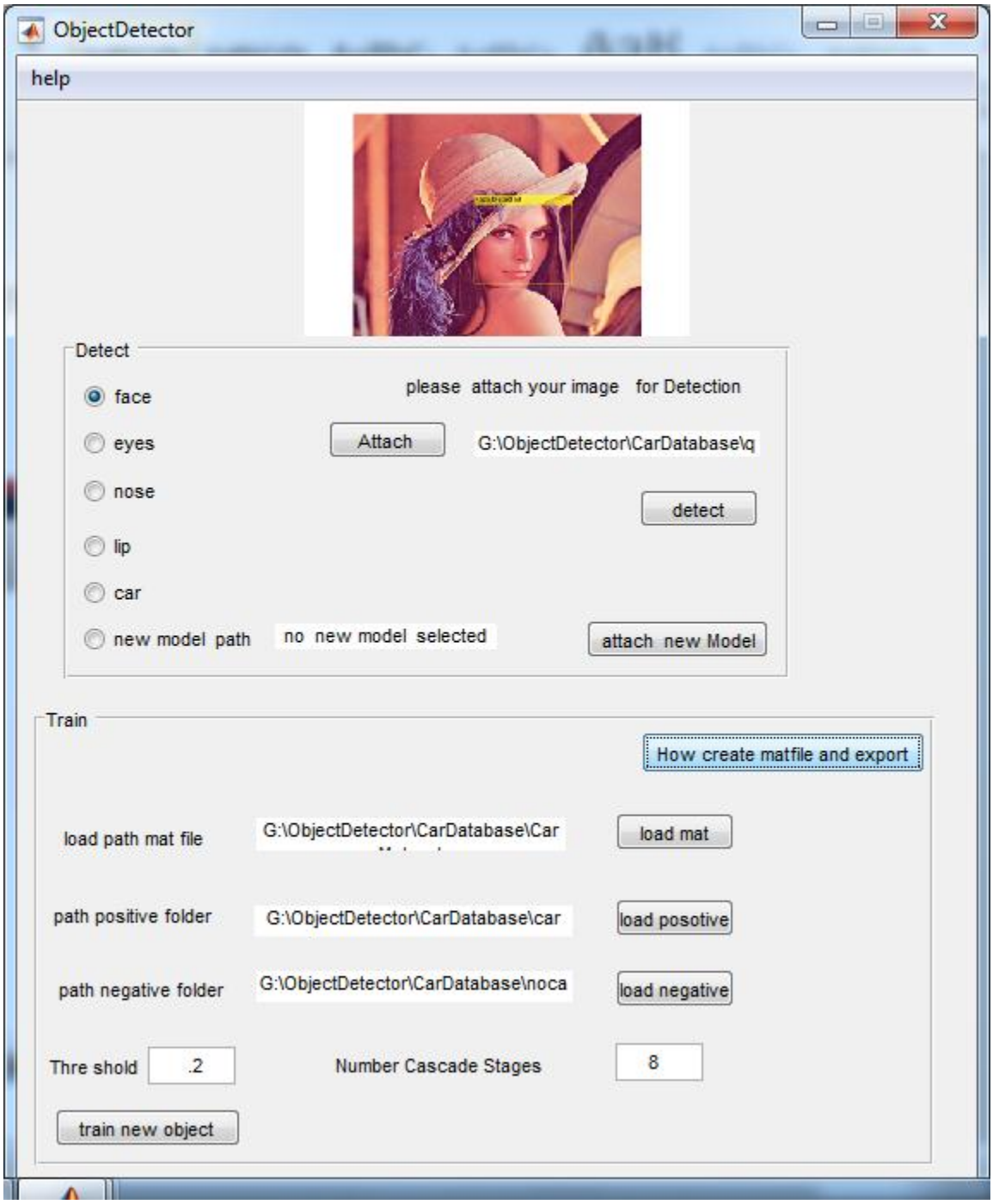
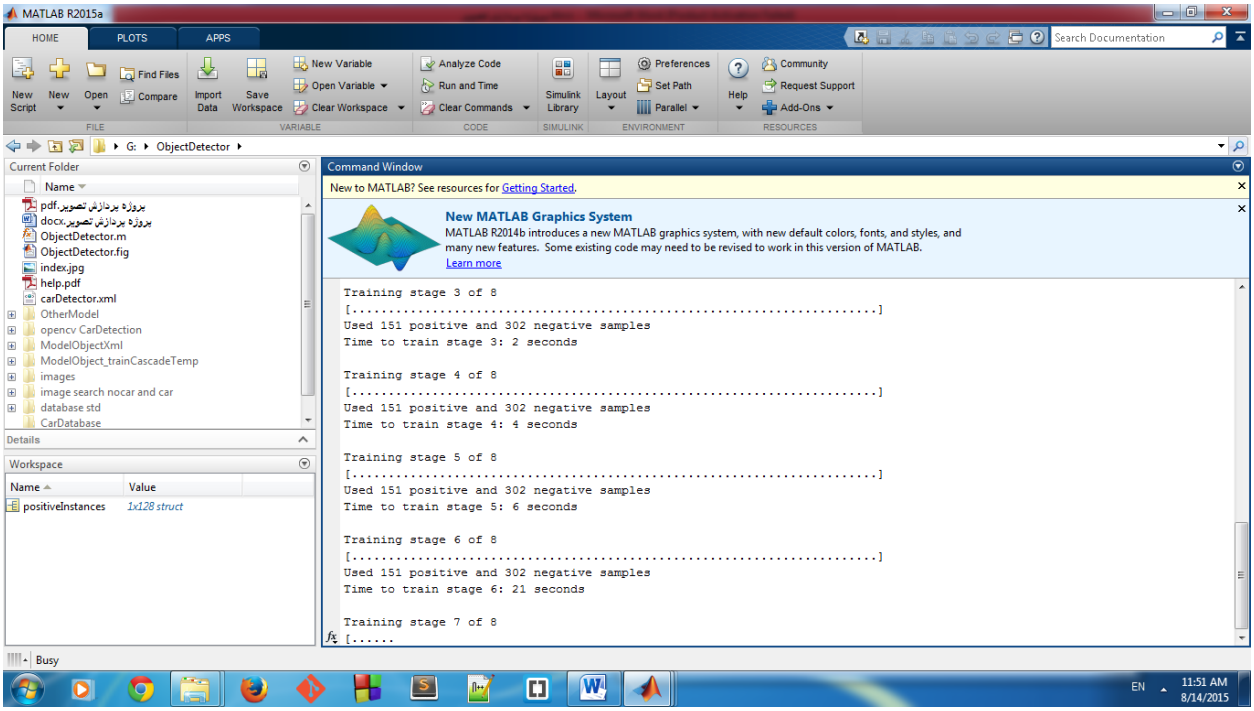


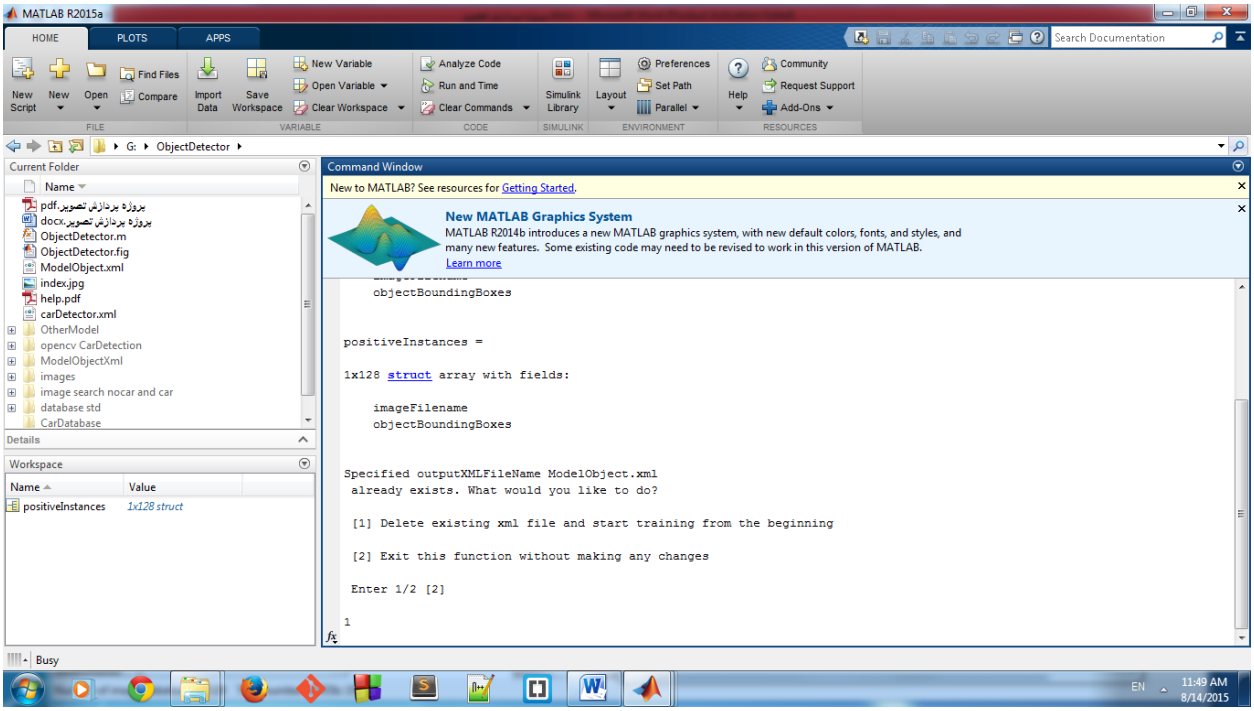
Figure 1
File Edit View Insert Tools Desktop Window Help
Face Detected

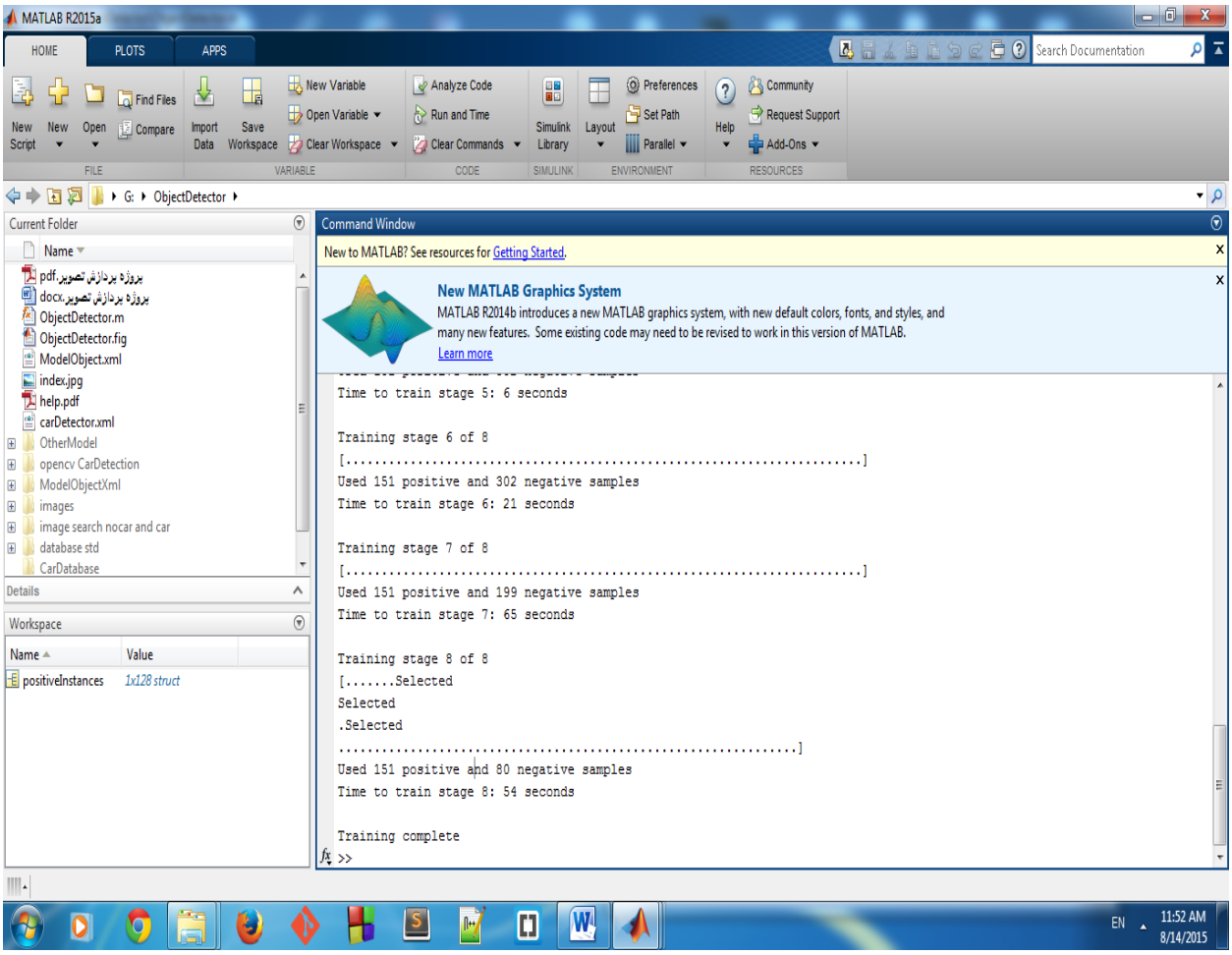
بعد از دادن پارامتر های مورد نیاز اموزش و اکسپورت کردن داده های مت فایل اگر دکمه اموزش را بزنیم،خواهیم دید

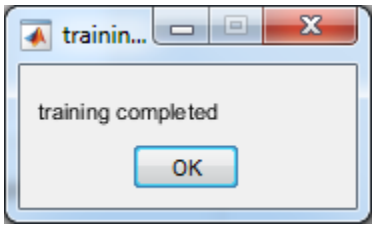


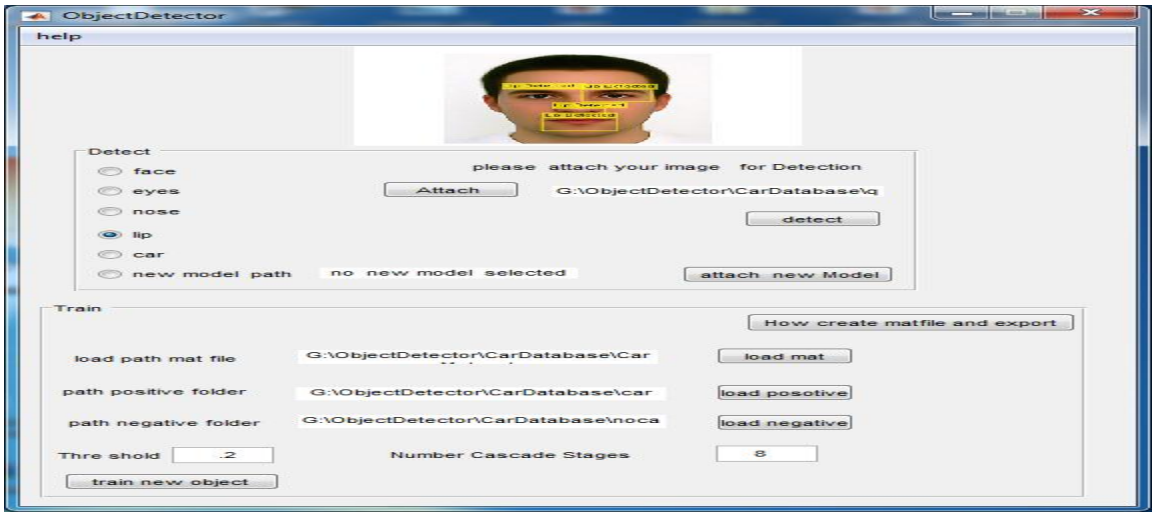

ObjectDetector
help
Detect
face
eyes
nose
lip
car
new model path
please attach your image for Detection
Attach
G:\ObjectDetector\CarDatabase\q
detect
no new model selected
attach new Model
Train
How create matfile and export
load path mat file
G:\ObjectDetector\CarDatabase\Car
load mat
path positive folder
G:\ObjectDetector\CarDatabase\car
load posotive
path negative folder
G:\ObjectDetector\CarDatabase\noca
load negative
Thre shold
.2
Number Cascade Stages
8
train new object

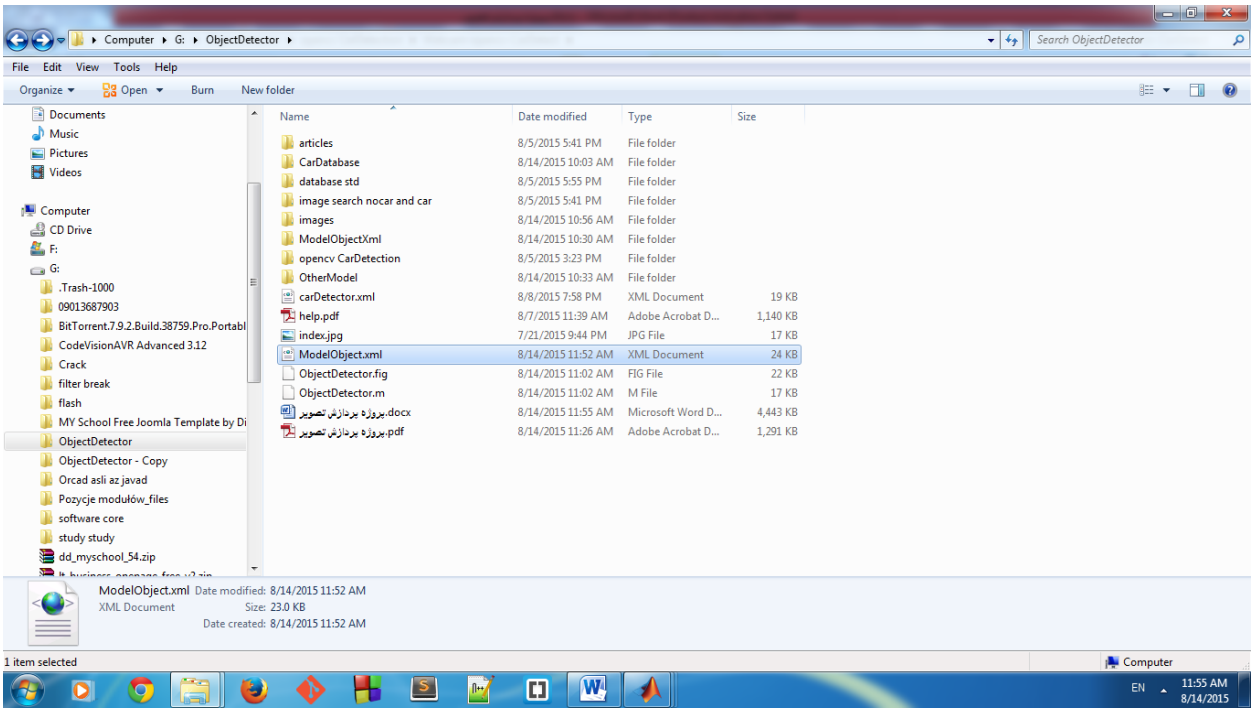

MATLAB R2015a
HOME
PLOTS
APPS
Search Documentation
New Script
New
Open
Find Files
Compare
Import Data
Save Workspace
New Variable
Open Variable
Clear Workspace
Analyze Code
Run and Time
Clear Commands
Simulink Library
Layout
Preferences
Set Path
Parallel
Help
Community
Request Support
Add-Ons
FILE
VARIABLE
CODE
SIMULINK
ENVIRONMENT
RESOURCES
G: ▸ ObjectDetector ▸
Current Folder
Name
پروژه پردازش تصویر.pdf
پروژه پردازش تصویر.docx
ObjectDetector.m
ObjectDetector.fig
index.jpg
help.pdf
carDetector.xml
OtherModel
opencv CarDetection
ModelObjectXml
ModelObject_trainCascadeTemp
images
image search nocar and car
database std
CarDatabase
Details
Workspace
Name
Value
positiveInstances
1x128 struct
Command Window
New to MATLAB? See resources for Getting Started.
New MATLAB Graphics System
MATLAB R2014b introduces a new MATLAB graphics system, with new default colors, fonts, and styles, and many new features. Some existing code may need to be revised to work in this version of MATLAB.
Learn more
Training stage 3 of 8
[.....]
Used 151 positive and 302 negative samples
Time to train stage 3: 2 seconds
Training stage 4 of 8
[.....]
Used 151 positive and 302 negative samples
Time to train stage 4: 4 seconds
Training stage 5 of 8
[.....]
Used 151 positive and 302 negative samples
Time to train stage 5: 6 seconds
Training stage 6 of 8
[.....]
Used 151 positive and 302 negative samples
Time to train stage 6: 21 seconds
Training stage 7 of 8
[.....
Busy
EN
11:51 AM
8/14/2015


MATLAB R2015a
HOME
PLOTS
APPS
Search Documentation
New Script
New
Open
Find Files
Compare
Import Data
Save Workspace
New Variable
Open Variable
Clear Workspace
Analyze Code
Run and Time
Clear Commands
Simulink Library
Layout
Preferences
Set Path
Parallel
Help
Community
Request Support
Add-Ons
FILE
VARIABLE
CODE
SIMULINK
ENVIRONMENT
RESOURCES
G: ▸ ObjectDetector ▸
Current Folder
Name
پروژه پردازش تصویر.pdf
پروژه پردازش تصویر.docx
ObjectDetector.m
ObjectDetector.fig
ModelObject.xml
index.jpg
help.pdf
carDetector.xml
OtherModel
opencv CarDetection
ModelObjectXml
images
image search nocar and car
database std
CarDatabase
Details
Workspace
Name
Value
positiveInstances
1x128 struct
Command Window
New to MATLAB? See resources for Getting Started.
New MATLAB Graphics System
MATLAB R2014b introduces a new MATLAB graphics system, with new default colors, fonts, and styles, and many new features. Some existing code may need to be revised to work in this version of MATLAB.
Learn more
objectBoundingBoxes
positiveInstances =
1x128 struct array with fields:
imageFilename
objectBoundingBoxes
Specified outputXMLFileName ModelObject.xml
already exists. What would you like to do?
[1] Delete existing xml file and start training from the beginning
[2] Exit this function without making any changes
Enter 1/2 [2]
1
Busy
EN
11:49 AM
8/14/2015


MATLAB R2015a
HOME
PLOTS
APPS
Search Documentation
New Script
New
Open
Find Files
Compare
Import Data
Save Workspace
New Variable
Open Variable
Clear Workspace
Analyze Code
Run and Time
Clear Commands
Simulink Library
Layout
Preferences
Set Path
Parallel
Help
Community
Request Support
Add-Ons
FILE
VARIABLE
CODE
SIMULINK
ENVIRONMENT
RESOURCES
G: ▸ ObjectDetector ▸
Current Folder
Name
پروژه پردازش تصویر.pdf
پروژه پردازش تصویر.docx
ObjectDetector.m
ObjectDetector.fig
ModelObject.xml
index.jpg
help.pdf
carDetector.xml
OtherModel
opencv CarDetection
ModelObjectXml
images
image search nocar and car
database std
CarDatabase
Details
Workspace
Name
Value
positiveInstances
1x128 struct
Command Window
New to MATLAB? See resources for Getting Started.
New MATLAB Graphics System
MATLAB R2014b introduces a new MATLAB graphics system, with new default colors, fonts, and styles, and many new features. Some existing code may need to be revised to work in this version of MATLAB.
Learn more
Time to train stage 5: 6 seconds
Training stage 6 of 8
[.....]
Used 151 positive and 302 negative samples
Time to train stage 6: 21 seconds
Training stage 7 of 8
[.....]
Used 151 positive and 199 negative samples
Time to train stage 7: 65 seconds
Training stage 8 of 8
[.....Selected
Selected
.Selected
.....]
Used 151 positive and 80 negative samples
Time to train stage 8: 54 seconds
Training complete
>>
EN
11:52 AM
8/14/2015

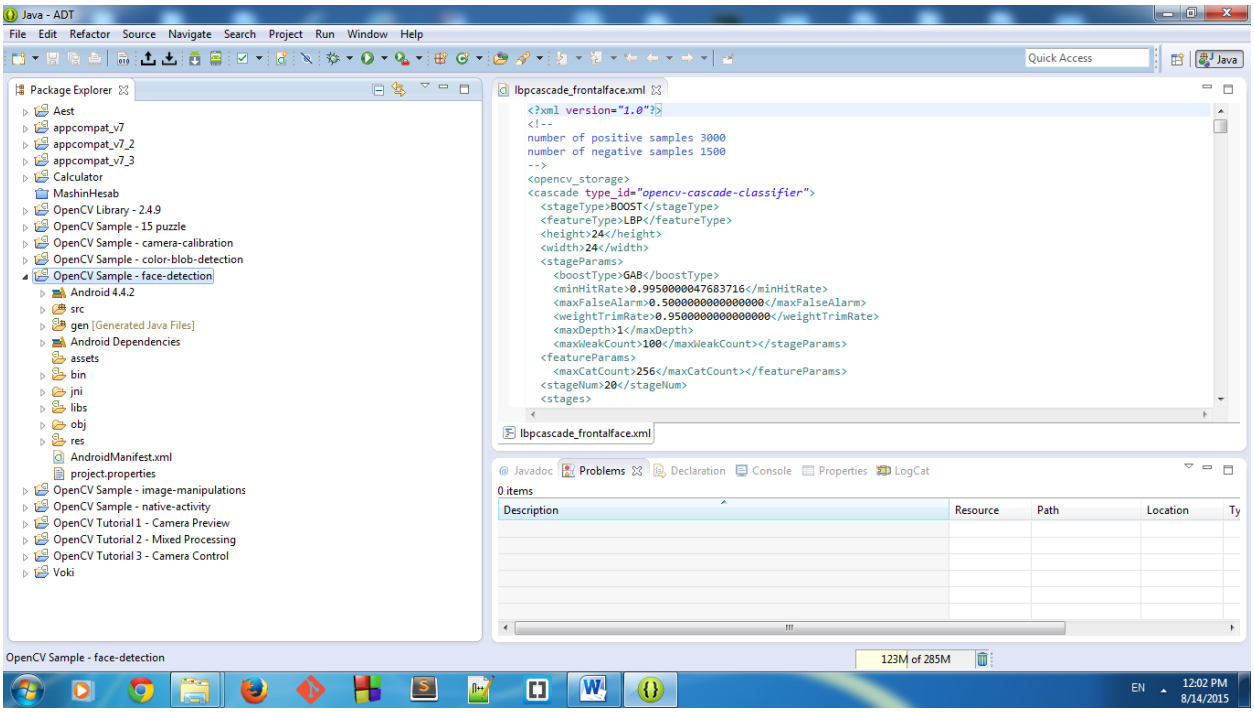

trainin...
training completed
OK



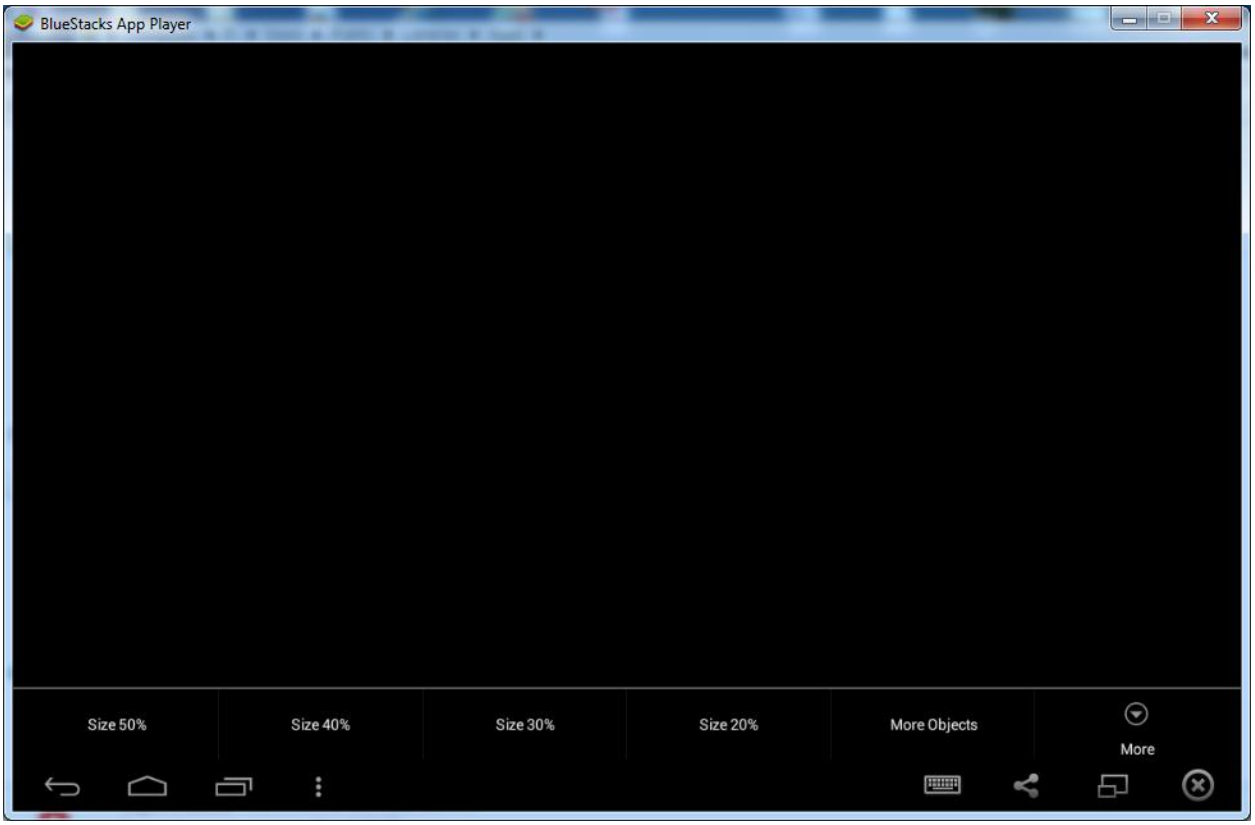


در نهایت از ADTاندروید استفاده کرده، OPENCV4Android در ان پیکره بندی می کنیم،از مثال تشخیص چهره ان را کمی عوض کرده ،مدل ماشین را در ان

وارد می کنیم،با این اپ اندرویدی قادر خواهیم بود ماشین را از طرف بغل تشخیص دهیم.(برای اظلاع از برنامه نویسی اندروید به سایت
http://developer.android.com/index.html رجوع کنید.



در نهایت فایل apk،تشخیص ماشین را ساخته که در پوشه پروژه اورده شده است،که برای نصب ان،به پوشه apkرفته،به پوشه opencvmanager یکی از ان ها را براساس سخت افزار گوشی که هسته ی ان چه نوع ارمی است،که معمولا OpenCV_2.4.9_Manager_2.18_armv7a-neon.apk انتخاب کرده ونصب کنید.این نرم افزار برای اجرای صحیح ObjectDetectorالزامی است،بعد به پوشه Object Detector رفته ونرم افزار را نصب کنید،که در تصویر نتیجه کار را می بینید.


BlueStacks App Player
Size 50%
Size 40%
Size 30%
Size 20%
More Objects
More

دوست داشتم داخل محیط کیوت Qtهم opencvرا بیارم،که به علت کمبود وقت موفق نشدم.شما را به خدای بزرگ می سپارم.

موفق باشید