

تبدیل کد متلب به فایل dll یا Jar

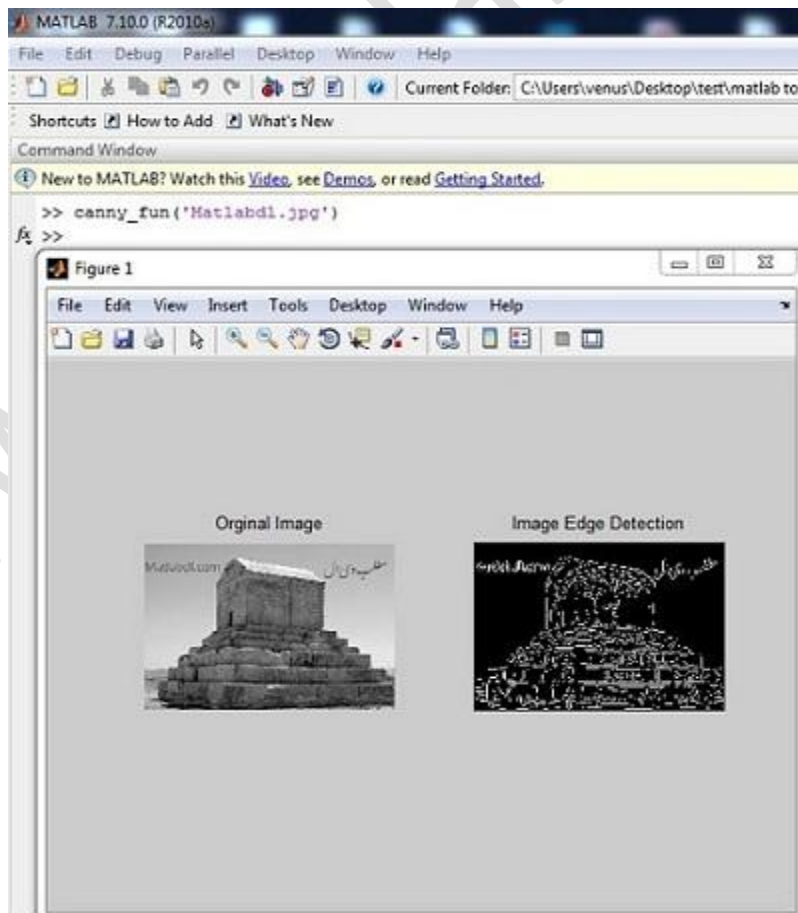
گاهی اوقات نوشتن یک برنامه ساده با سی شارپ باعث می شود چندین خط کدنویسی کنیم در صورتی که همان برنامه در متلب با یک یا دو خط نوشته می شود. در آموزش می خواهیم به شما آموزش بدهیم که بتوانید این نوشتن اجباری را دور بزنید. برای این کار کافیهست که از کدهایی که درون متلب می نویسید dll بسازید. البته این تبدیل فقط مربوط به dll نمی شود، بلکه می توانید از کدهایتان exe, com و jar هم ایجاد کنید.

برای این کار ما با یک آموزش ساده شروع می کنیم، لبه یابی تصویر با canny.

یک new function ایجاد می کنیم که یک ورودی از نوع تصویر می گیرد و آن را خاکستری می کند و سپس لبه یابی canny را پیاده سازی می کند.

```
function canny_fun( image )  
I=imread(image);  
x=rgb2gray(I);  
subplot(1,2,1),imshow(x),title('Original Image');  
BW=edge(x,'canny');  
subplot(1,2,2),imshow(BW),title('Image Edge Detection');  
end
```

بعد این تابع را با نام canny_fun ذخیره می کنیم. سپس آنرا اجرا می کنیم تا مطمئن شویم تابع درست کار می کند. پس در محیط command تابع را صدا می کنیم و تصویر Matlabdl.jpg را به عنوان ورودی به آن می دهیم.

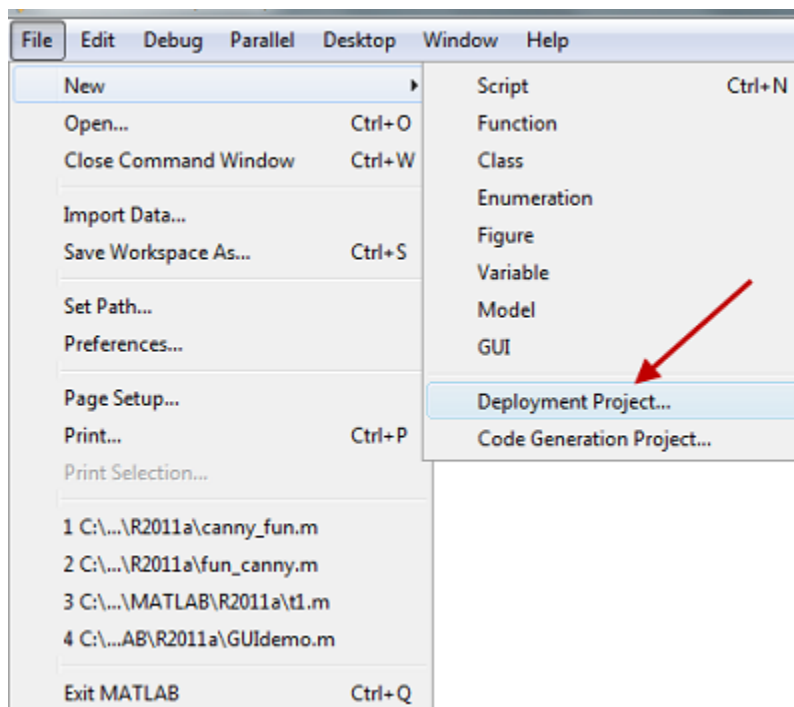


بعد از این کار به سراغ ساخت dll از این برنامه می رویم.

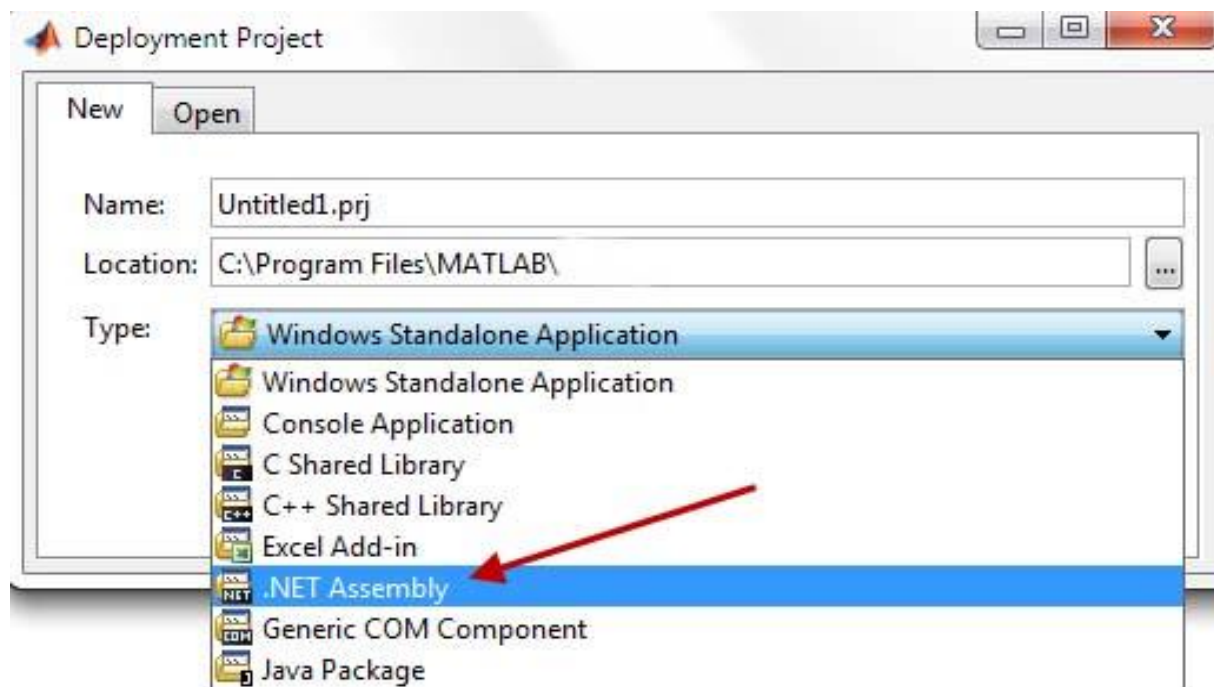
برای ساخت dll,exe,com و پسوند های دیگر سه روش داریم:

1. استفاده از Deployment Project از منوی file
2. تایپ دستور deploytool در محیط command
3. استفاده از کامپایلر mcc (کافیست آن را درون command تایپ کنید)

که ما در این آموزش از روش اول استفاده می کنیم.

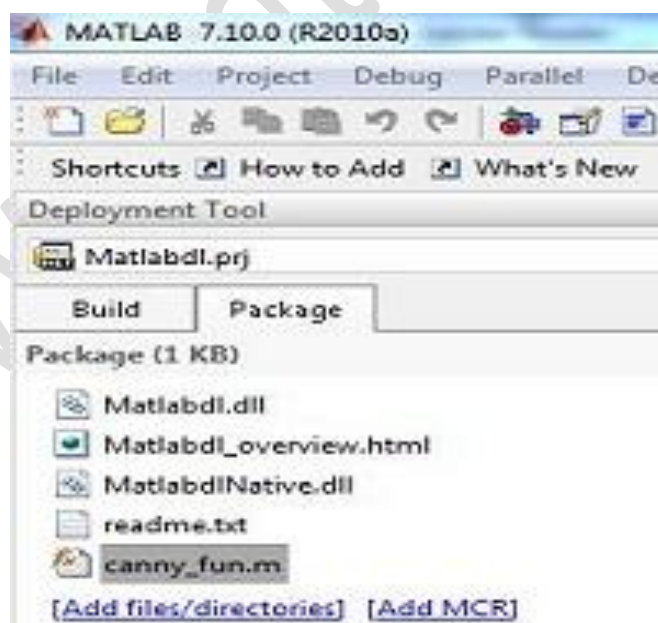


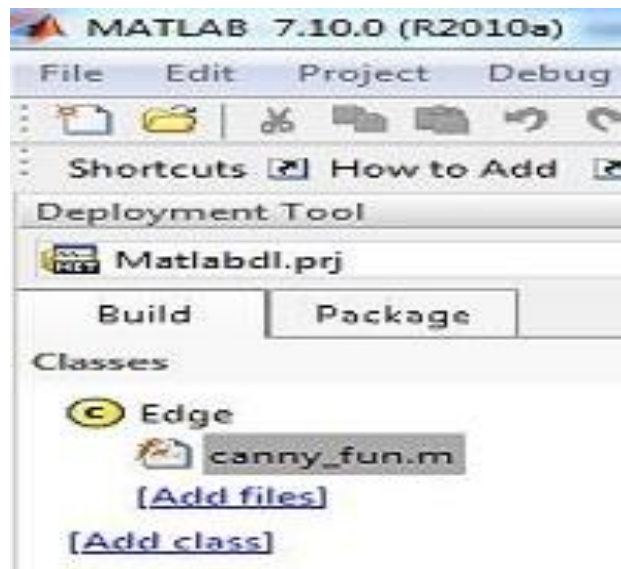
با انتخاب گزینه ی Deployment Project یک پنجره به شکل زیر باز می شود. ابتدا یک اسم برای پروژه انتخاب می کنیم (مثلا Matlabdl)



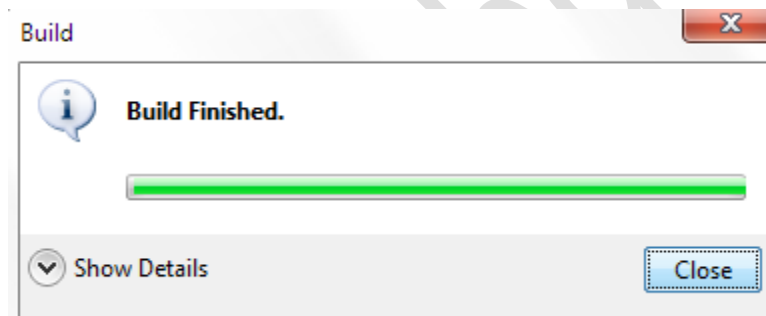
چون می خواهیم این پروژه را در محیط ویژوال استفاده کنیم .NET Assembly انتخاب می کنیم.

بعد از انتخاب این گزینه پنجره ای دیگر با عنوان .NET Assembly باز می شود که دارای دوتب می باشد یکی build که شامل کلاس و اضافه کردن متد به کلاس می باشد و یکی Package که فایل مربوطه را برای تبدیل به dll رابه آن اضافه می کنیم.

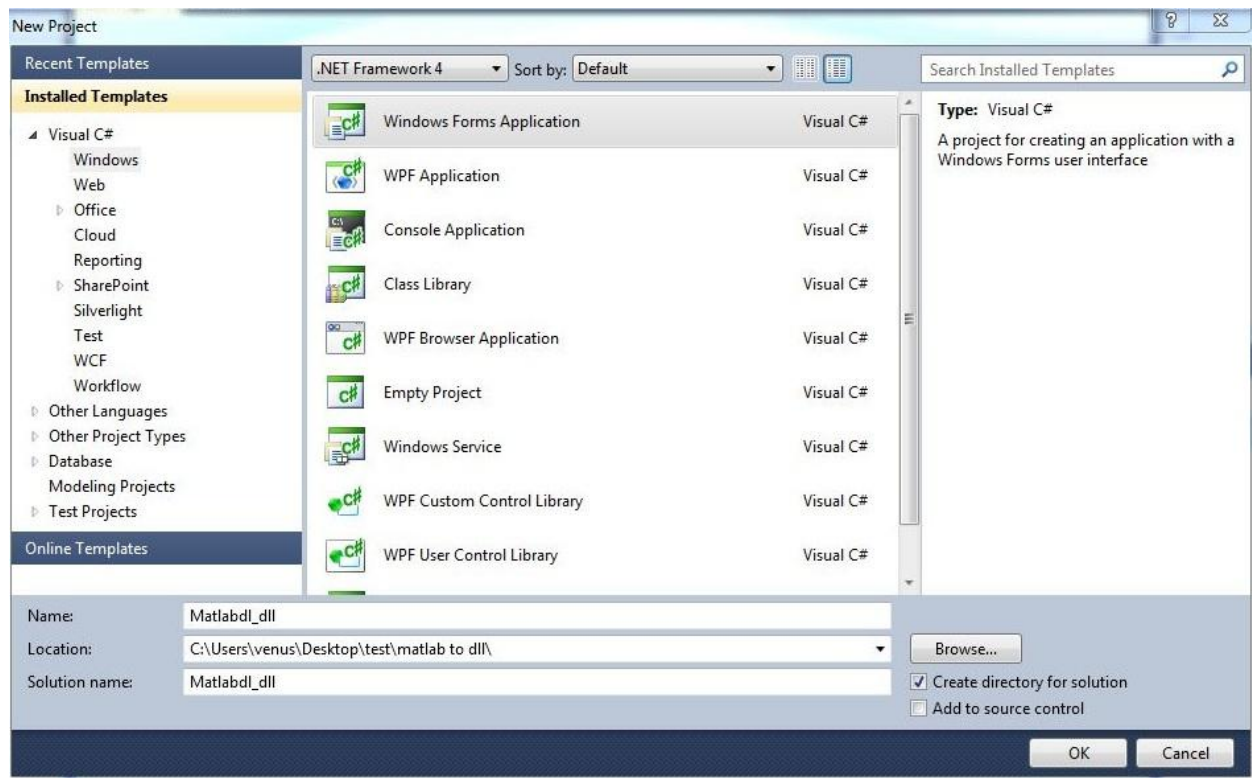




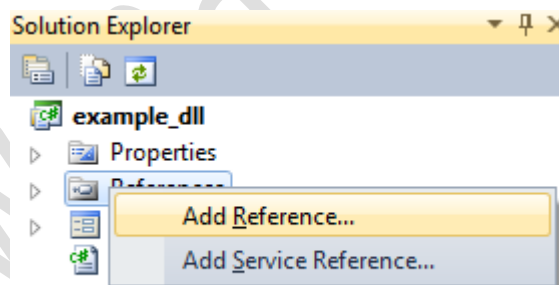
یک اسم مانند Edge را برای کلاس انتخاب می کنیم و سپس متد Canny_fun را به آن اضافه می کنیم. بعد از این کار دکمه Build را می زنیم تا شروع به ساخت dll کند.

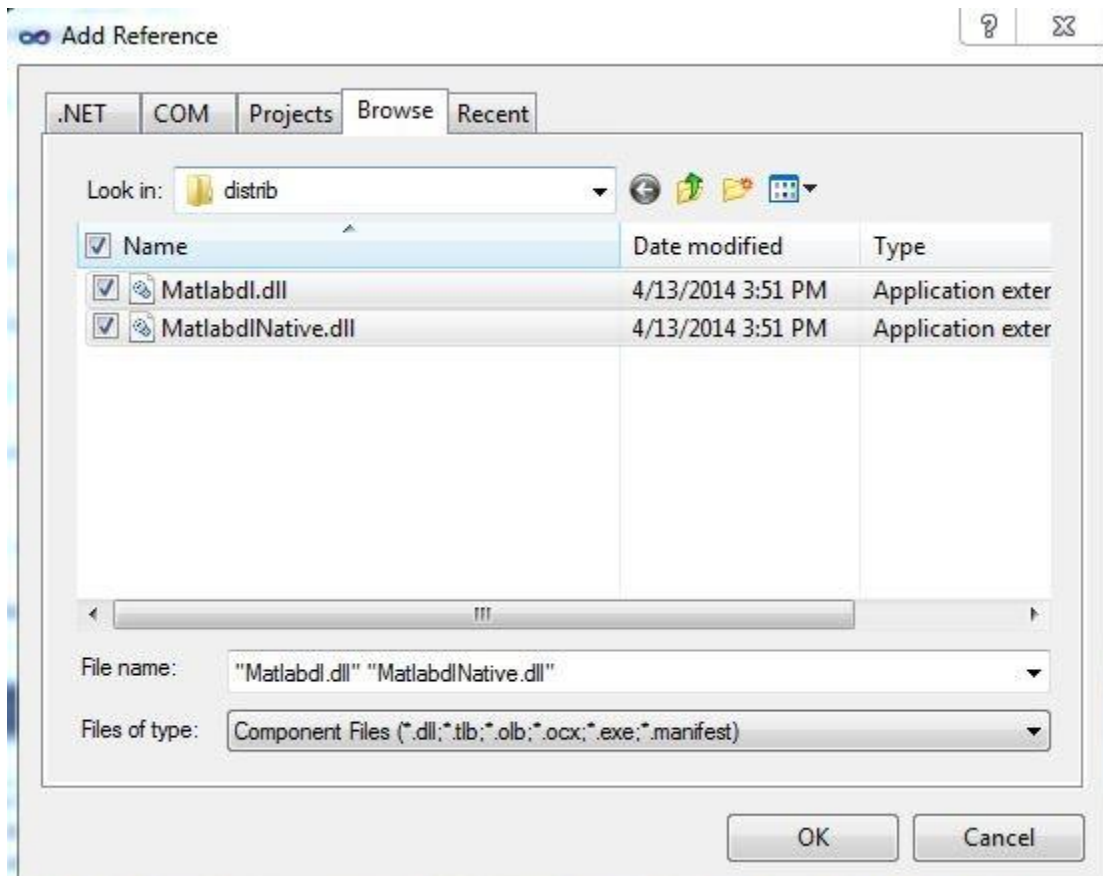


حالا سراغ ویژوال 2010 می رویم و یک پروژه از نوع Windows Forms Application به نام Matlabdl_dll ایجاد می کنیم.

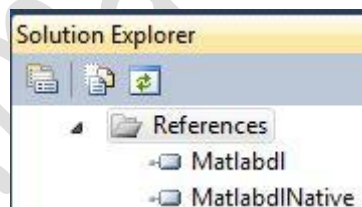


حال می خواهیم dll ای را که ساخته بودیم را به سی شارپ اضافه کنیم برای این کار از Reference Explorer solution Explorer راست کلیک می کنیم و با انتخاب Add Reference از Browse با رفتن به مسیری که پروژه ی Matlabdl ذخیره کرده بودیم هر دو dll ساخته شده را از پوشه ی distrib را انتخاب می کنیم.





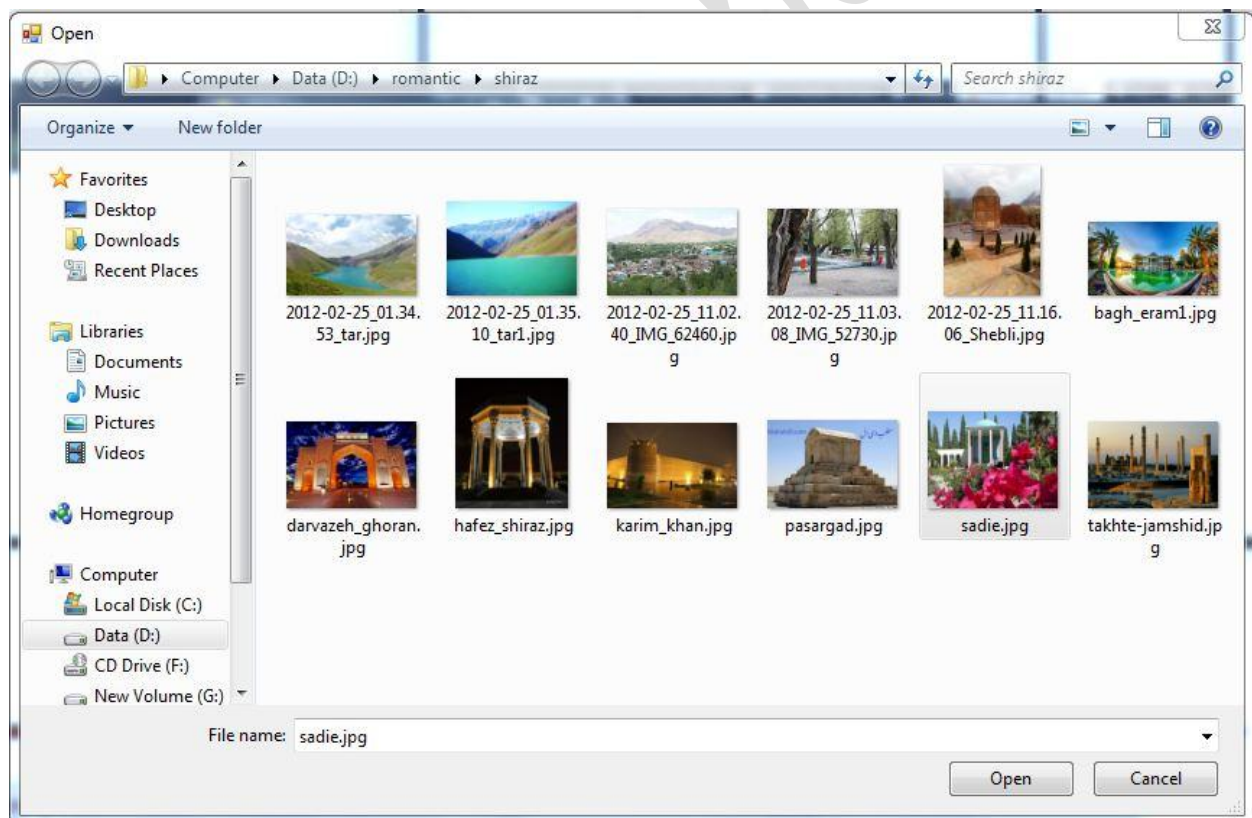
مشاهده می کنید که هر دو dll به برنامه اضافه شد

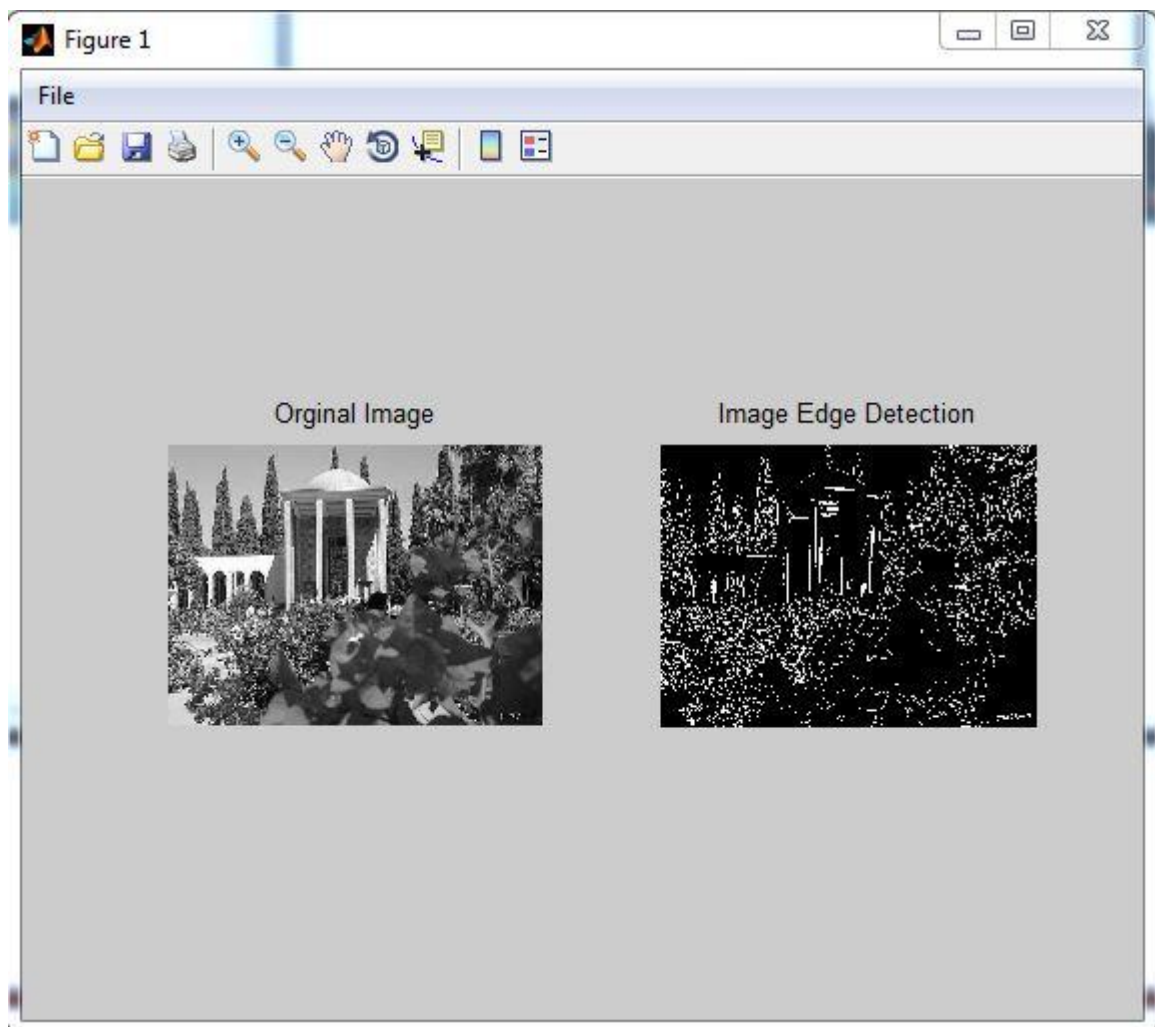


با کمی خلاقیت می خواهیم فرم پروژه را شفاف کنیم، برای اینکار از properties فرم خصوصیت Transparency key را به Gray تغییر می دهیم و یک panel را بر روی فرم قرار می دهیم و خاصیت back color آن را به Gray تغییر می دهیم. یک دکمه بر روی فرم قرار می دهیم و خاصیت text آن را نمایش می گذاریم. و یک open file dialog برای اینکه کاربر بتواند عکسش را انتخاب کند بر روی فرم قرار می دهیم. و در آخر هم در رویداد دکمه کلیک زیر را می نویسیم.

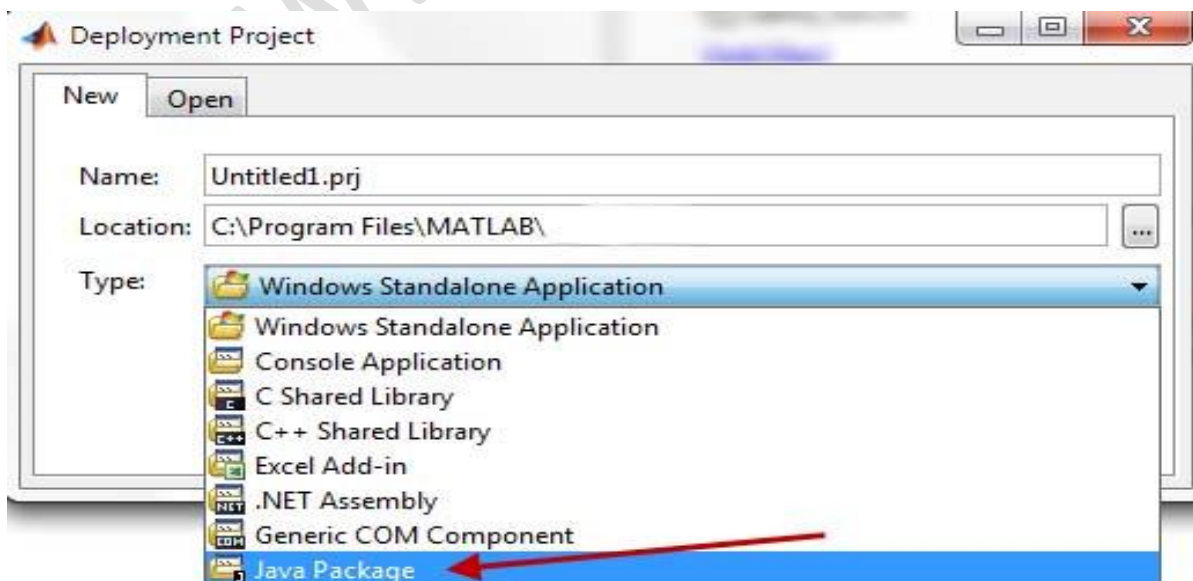
```
MatlabdlNative.Edge Ma = new MatlabdlNative.Edge();
DialogResult dr;
dr = openFileDialog1.ShowDialog();
if (dr == DialogResult.OK)
{
    Ma.canny_fun(openFileDialog1.FileName);
}
```

در زیر من فرم شفاف را درون سایت [خودم](http://www.matlabdl.com) اجرا کردم که اگر دکمه نمایش را بزنیم پنجره ی open باز می شود و می توانیم تصویرمان را انتخاب کنیم. و بعد از چند ثانیه به شکل اجرا یک figure نمایش داده می شود.





حال اگر بخواهیم همین برنامه را در جاوا پیاده سازی کنیم، همانند مطالب گفته شده در بالا از منوی file گزینه ی Deployment Project را انتخاب می کنیم و این بار نوع پروژه java Package قرار می دهیم.



یک پروژه ی جدید از نوع جاوا اپلیکیشن ایجاد می کنیم و داخل JFrame یک دکمه به نام نمایش قرار می دهیم در ضمن در اینجا باید ابتدای کدنویسی دو import مهم داشته باشیم.

```
Import project name.class name;
```

```
Import com.mathworks.toolbox.javabuilder.MWException;
```

