

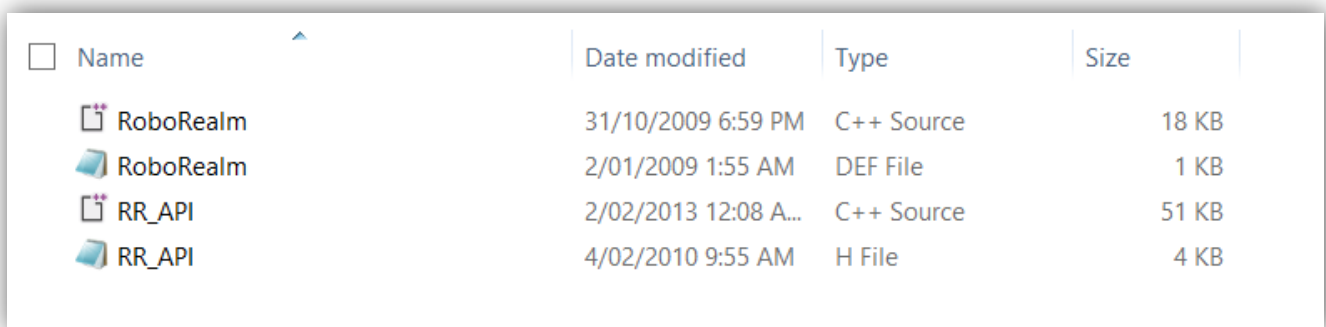
Roborealm API Tutorial

This project will require you to connect to the Roborealm servers in order to view the video streams which you may use to track your race vehicles.

In order to connect to the Roborealm servers through Matlab, the Roborealm API for Matlab must be downloaded and compiled.

API download can be found on the courses webpage (or you can use your google-fu).

You should have these files as shown in the figure below



| Name | Date modified | Type | Size |
|-----------|----------------------|------------|-------|
| RoboRealm | 31/10/2009 6:59 PM | C++ Source | 18 KB |
| RoboRealm | 2/01/2009 1:55 AM | DEF File | 1 KB |
| RR_API | 2/02/2013 12:08 A... | C++ Source | 51 KB |
| RR_API | 4/02/2010 9:55 AM | H File | 4 KB |

These files need to be compiled into mexw64 files. To do this, in Matlab, navigate to the directory where the source files are located, then call:

```
>>mex RoboRealm.cpp
```

This should build the file "Roborealm.mexw64".

Note!

If you do not have a valid C/C++ compiler linked to Matlab, ensure you have a supported compiler installed.

A list of supported compilers can be found here: [Mathworks Compiler Table](#)

Call:

```
>>mex -setup
```

and follow the prompts.

Command Window

```
>> mex -setup
```

```
Welcome to mex -setup. This utility will help you set up  
a default compiler. For a list of supported compilers, see  
http://www.mathworks.com/support/compilers/R2013a/win64.html
```

```
Please choose your compiler for building MEX-files:
```

```
Would you like mex to locate installed compilers [y]/n? y
```

```
Select a compiler:
```

```
[1] Microsoft Visual C++ 2008 SP1 in C:\Program Files (x86)\Microsoft Visual Studio 9.0
```

```
[0] None
```

```
Compiler: 1
```

```
Please verify your choices:
```

```
Compiler: Microsoft Visual C++ 2008 SP1
```

```
Location: C:\Program Files (x86)\Microsoft Visual Studio 9.0
```

```
Are these correct [y]/n? y
```

```
*****  
Warning: MEX-files generated using Microsoft Visual C++ 2008 require  
that Microsoft Visual Studio 2008 run-time libraries be  
available on the computer they are run on.  
If you plan to redistribute your MEX-files to other MATLAB  
users, be sure that they have the run-time libraries.  
*****
```

```
Trying to update options file: C:\Users\John\AppData\Roaming\MathWorks\MATLAB\R2013a\mexopts.bat  
From template: C:\PROGRA~1\MATLAB\R2013a\bin\win64\mexopts\msvc90opts.bat
```

```
Done . . .
```

Now that we have successfully compiled the RoboRealm mex file, lets start grabbing images from the server!

Let's start with a new Matlab script then include the lines:

```
RoboRealm('connect', '127.0.0.1', int32(6060));  
[width, height] = RoboRealm('get_dimension');  
img = RoboRealm('get_image');  
fig1 = figure(1);  
imshow(img);  
RoboRealm('disconnect');
```

This connects to the RoboRealm Server located at address '127.0.0.1' on port '6060'. Now for those who don't have RoboRealm running on their computers, this should result in an error. We will endeavour to finalise the IP address for the server ASAP. (You can download a feature complete 30 free trial of RoboRealm should you wish to do so)

The code is self-explanatory, and you can place the code inside a loop to do real time video processing.

To grab another image from a different port, we need to call the above code with the desired new port number in place of '6060'.

Please note that you **MUST** disconnect for every connection to RoboRealm made. Each connection will timeout after 10ms, and calling any RoboRealm functions (such as Roborealml('get_dimension')) after a timeout will cause the API to attempt to connect over local host (unless default behaviour is changed in the source files). Calling any RoboRealm function whilst a connection is active will keep the connection alive.

RoboRealm has a **100 maximum concurrent connection limit**, and if that limit is reached, all other connections will time out.