CMTK instructions for Windows

Modified from Takaaki Miyazaki’s website by Gabriella Sterne: researchmap.jp/t\_m/

|  |  |  |
| --- | --- | --- |
| **2014/11/05** | [**Registration using Computational Morphometry Toolkit (CMTK) on Windows**](http://researchmap.jp/jo57feb6e-2041067/#_2041067) | [Tweet This](javascript:void%20window.open('http://twitter.com/intent/tweet?text='+encodeURIComponent('Registration%20using%20Computational%20Morphometry%20Toolkit%20(CMTK)%20on%20Windows%20http://researchmap.jp/jo57feb6e-2041067/#_2041067'), '_blank');)[Send to Facebook](javascript:void%20window.open('http://www.facebook.com/share.php?u='+encodeURIComponent('http://researchmap.jp/jo57feb6e-2041067/')+'&t='+encodeURIComponent('Registration%20using%20Computational%20Morphometry%20Toolkit%20(CMTK)%20on%20Windows'),%20'_blank');)| by [t\_m](http://researchmap.jp/t_m/Research-blog/) |

**Overview**

* 1. Prepare your PC environment.
  + 1.1. Install Perl.
  + 1.2. Install CMTK.
  + 1.3. Install libfftw3-3.dll.
* 2. Prepare confocal stack data.
  + 2.1. Make a folder for work.
  + 2.2. Format confocal stack data.
  + 2.3. Format template brain data.
* 3. Do registration.
  + 3.1. Make a command script file for registration.
  + 3.2. Execute the registration.
  + 3.3. Make a command script file for reformatting other channels.
  + 3.4. Execute the reformatting of other channels.

**1. Prepare your PC environment.**

**1.1. Install Perl.**

Active Perl is used to run Perl scripts. You can download Perl for Windows from the following site: <http://www.activestate.com/activeperl/downloads>. Then, install Perl into your PC. You need an administrator privilege for this. Either 32 bit version (x86) or 64 bit version (x64) will work. You might use Strawberry Perl instead of Active Perl.

**1.2. Install CMTK.**

You will find only Mac OS X version or Cygwin version of CMTK in the "Download Now" pull-down menu in the [CMTK website(https://www.nitrc.org/projects/cmtk)](https://www.nitrc.org/projects/cmtk), but when you follow "See All Files" link (<https://www.nitrc.org/frs/?group_id=212>) you can find a link to download the Windows version. Download [CMTK-3.2.2-Windows-x86.zip](https://www.nitrc.org/frs/download.php/6800/CMTK-3.2.2-Windows-x86.exe) (\* If you choose .exe version, you will need an administrator privilege for the install, but .zip version doesn't require it.) (You should use this specific file, not newer versions). When you extract the zip file, you will find all of the files are in the folder "CMTK-3.2.2-Windows-x86\Program Files\CMTK". Copy the "CMTK" folder (and all of the files and subfolders below it) to somewhere you have a privilege to write (e.g. C:\Users\[user name]\). (\* To avoid following many levels of folder hierarchy, just copy "CMTK" folder.) Here, let's call the target folder (C:\Users\[user name]\CMTK\, in the example) "[folder1]."

Avoid placing “[folder1]” in the “Documents” folder of a specific user. This may lead to file path conflicts. If possible place directly in C:\ (i.e. C:\CMTKregistration\CMTK).

**1.3. Install libfftw3-3.dll.**

Download fftw-3.3.4-dll32.zip from <http://www.fftw.org/install/windows.html>. The website provides both32bit and 64bit versions, but you must use the 32bit version for CMTK. Extract the downloaded file. You will find libfftw3-3.dll in the resulting files. Copy this file into [folder1]\bin.

**2. Prepare confocal stack data.**

**2.1. Make a folder for work.**

Make a folder for work. For example, C:\CMTKregistration\CMTKreg\. Let's call this folder "[folder2]." Make two folders whose names are "images" and "template" in [folder2].

**2.2. Format confocal stack data.**

Open confocal stack data with Fiji. Check to make sure the brain is not rotated. If rotated, choose “Image” – “Transform” – “Rotate” to rotate the image as close to the orientation of the template as possible. When performing this step, only rotate the image one time and do not check the box for interpolation to avoid reductions in image quality. The input data should also have the same total depth of the brain as the template. To achieve this, figure out the total depth of the template from where the nc82 starts to where it stops. Then do the same for the data to be registered. In ImageJ, calculate what the step size of the data to be registered would need to be to perfectly match the depth of the template and change the z step size accordingly (If using the JFRC2010 template your final voxel size should be 0.62x0.62xcalculated for correct depth). If the data contain multiple channels, split them by selecting "Images" - "Color" - "Split Channels"from the menu. Select "File" - "Save As" - "Biorad" for each channel to save it in [folder2]\images\\*. The filenames will be set with ".pic" extension. The file name must match the following rules.

* Ending of the filename "01.pic" needs to be assigned to a channel containing data subject to registration.
* Other channels need to have filenames ending with "02.PIC," "03.PIC" ...\*\*
* All channels containing data of the same sample need to have the same text string of the filename except the endings.

\*16-bit files are preferable to 8-bit files, and .nrrd format will work as well as Biorad files. However, the bit depth and format of the data to be registered must exactly match that of the template.

\*\*Hideo Otsuna at Janelia has programmed a Fiji macro that does all of the steps in this section automatically. You can contact him at otsunah@janelia.hhmi.org

**2.3. Format template brain data.**

Open a file of template brain data. If it contains multiple channels, split them by selecting "Images" - "Color" - "Split Channels" from the menu. Select "File" - "Save As" - "Biorad" for the channel subject to registration to save it in [folder2]\template\. The must be "template.PIC." [No, it doesn’t have to have that exact name if you’ve specified the name in registration.cmd and reformat.cmd.) The template must be the same file type and bit depth as the images you are registering, so change accordingly if you are using 16-bit .nrrd files. If using the JFRC2010 template, the voxel size should be 0.62x0.62x1.

**3. Do registration.**

**3.1. Make a command script file for registration.**

Start Notepad, input as following, and save the contents with".cmd" extension in "[folder2]\command\" subfolder. Here, let's set "registration.cmd" as a filename.

cd "[folder2]"

perl "[folder1]\bin\munger" -b "[folder1]\bin" -awr 01 -s "template/template.PIC" "images"

As an example, if you used the filepaths above, registration.cmd should read:

cd "C:\CMTKregistration\CMTKreg"

perl "C:\CMTKregistration\CMTK\bin\munger" -b "C:\CMTKregistration\CMTK\bin" -awr 01 -s "template/template.PIC" "images"

There are several options you can add to change the way the program runs. One useful addition to your registration command script allows you to limit the number of threads the program will use. By default, the program uses all threads, and will often use 100% of your CPU. Placing –T 8 between –awr 01 and –s, for example, will limit the number of threads the program uses to 8. So if you have 12 threads available, that will leave 4 threads for other tasks you are doing on the same computer.

Another option is to modify the accuracy with which the registration is performed. If you are using the program to do registrations that don’t require the utmost perfection, you can place -A "--accuracy 0.6" -W "--accuracy 0.6" between –awr 01 and –s (before –T if you are using it). This will decrease the accuracy and thus the time it takes to computer to perform the registration.

If you are having issues with running CMTK, adding the –v “verbose” function to the command file will supply additional information while the program is running.

As an example, this is the cmd file that Hideo Otsuna uses to register thousands of images:

cd "C:\CMTKregistration"

perl "C:\CMTK\bin\munger" -b "C:\CMTK\bin" -a -w -r 010203 -A "--accuracy 0.8" -W "--accuracy 0.8" -T 8 -s "template/JFRC2010\_CLAHE\_16bit.nrrd" "images"

**3.2. Execute the registration.**

Double click "registration.cmd." A Command Prompt window will open and calculation will be carried out (\* This may take a long time).

If the command fails, you should try to replace 'perl' in the 2nd line of "registration.cmd" with "[Perl installation folder]\bin\perl\".  
When the calculation is completed successfully, two folders whose name are "reformatted" and "Registration" will be generated in [folder2]. You will find a ".nrrd" file in the "reformatted" folder. This is a processed result for the channel subject to registration (nc82). You can open this file with Fiji.

**3.3. Make a command script file for reformatting other channels.**

Start Notepad, input as following, and save the contents with".cmd" extension in "[folder2]\command\" subfolder. Here, let's set "reformat.cmd" as a filename.

cd "[folder2]"

perl "[folder1]\bin\munger" -b "[folder1]\bin" -r 0203 -f 01 -s "template/template.PIC" "images"

\*This example shows a case where the data consists of two channels in addition to nc82. The string '0203' needs to be fit depending on the number of the channels other than nc82, e.g. '02' for a single channel,'020304' for three channels.

**3.4. Execute the reformatting of other channels.**

Doubleclick "reformat.cmd." A Command Prompt window will open and calculation will be carried out (\* This may take a long time.). If the command fails, you should try to replace 'perl' in the 2nd line of"registration.cmd" with "[Perl installation folder]\bin\perl\".  
When the calculation is completed successfully, you will find additional".nrrd" files in the "reformatted" folder. These are processed results. You can open these files with Fiji.