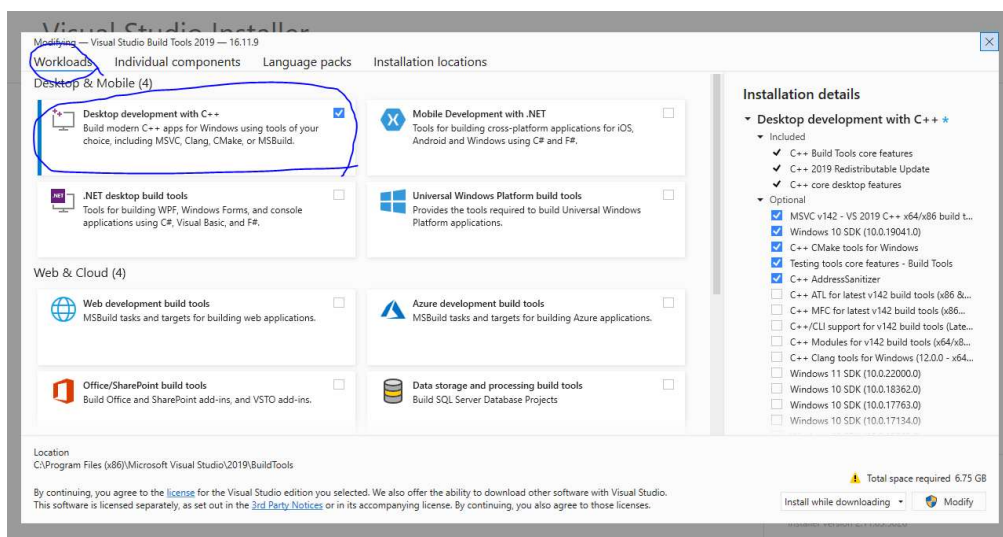


Auto3dgm_Python is an end-to-end implementation of Auto3dgm in Python. Compared to previous releases, Auto3dgm_Python contains 1) a GUI for ease of initiating analyses, 2) interactive visualization for quick check of alignment success, 3) multi-core processing architecture for more rapid analysis

Installation instructions for **Auto3dgm_Python**

I. Requirements

- 1) Anaconda3 <https://www.anaconda.com/products/individual>
- 2) Windows users must have a C++ compiler. We recommend Visual Studio, found at <https://visualstudio.microsoft.com/visual-cpp-build-tools/>
 - a) Most likely this will result in “Visual Studio Installer” being launched on your system.
 - b) Within Visual Studio Installer, go to the “workloads” tab and select “Desktop Development with C++” and Install/modify (see screenshot). It will be a few GB worth of programs/files and take a little while to complete. You may be asked to restart your system after it finishes.



- 3) If you are using Windows, we recommend downloading and installing Git Bash: [Git for Windows](#).

II. Installing Auto3dgm_Python

- 1) Download the main branch of this repository. You can **either**
 - a) clone the repository by typing in your bash terminal (the usual terminal on Unix operating systems, and the one you get with Git Bash or WSL on Windows): **git clone** https://github.com/ToothAndClaw/Auto3dgm_Python.git wherever you wish to save the repository
 - b) or download the ZIP file (click the green Code button) and unzipping the contents of Auto3dgm_Python-main.zip where you wish

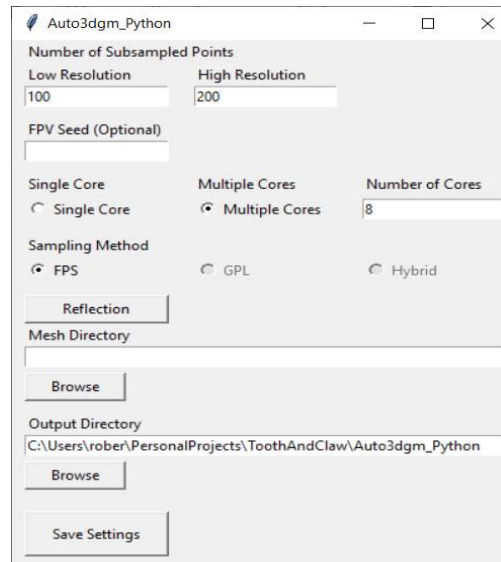
- 2) Step for Mac & Unix users only: In your system's file explorer, navigate within the downloaded code to the folder entitled *envs* (e.g., **Users\Documents\Auto3dgm_Python-main-main\envs**) and open the .yml file corresponding to your operating system (the .yml file can be opened with a text editor like wordpad or notepad). In the last line of the .yml file, you will see "prefix: C:\ProgramData\Anaconda3\envs\sams". **Change the text after "prefix:" to the directory where your conda environments are located.** MacOS and Linux users must change this field (at minimum to reflect their username).
- 3) In your Anaconda prompt (Windows) or bash terminal (Unix), navigate to the directory containing the unzipped Auto3dgm_Python folder (e.g., "**cd C:\Users\Documents\Auto3dgm_Python-main**")
 - a) Note that if your Auto3dgm_Python folder is on a different drive than Anaconda (e.g., on your "E" drive instead of the 'C' drive) you will need to write the command like this: **cd / d E:\Users\Documents\Auto3dgm_Python-main.**
- 4) Type or copy/paste the following commands depending on your operating system

conda env create --name auto3dgm --file ./envs/env_YOURYMLFILE.yml

The above .yml file should be the same that you used previously.

III. Initiate the GUI and run an analysis

- 1) Whenever you want to run an analysis at any time after sections I-II above have been completed, begin by opening an Anaconda or bash terminal in your computer.
- 2) Navigate to where you placed the code (e.g., if using the installation from II, you would begin by typing "**cd C:\Users\Documents\Auto3dgm_Python-main**")
- 3) Type or copy/paste "**conda activate auto3dgm**" and hit enter to activate the Auto3dgm environment.
 - a) NOTE: A common issue with anaconda is you might get an error trying to create the environment, with a missing module "six". To fix this, in the anaconda terminal type or copy/paste "pip install six" and then try again the beginning of step 4 again.
 - b) If you need to deactivate the environment, type or copy/paste "conda deactivate"
- 4) Type or copy/paste "**python Auto3dgm_Python.py**" and hit enter. This will open a GUI for running the software. Screenshot of the GUI below.



- 5) Once the GUI is open, hopefully the rest will be mostly straightforward if you have used auto3dgm before. If not see <https://toothandclaw.github.io/how-to-use/> and go to “step 3. Set up Parameters”. Once you hit “save settings” on the GUI, your screen should look like the below image (on Windows 10)lin

