

11/16/2022

How to Manually Install WebODM on Windows 10 and 11 pc

Open Geospatial Information
Infrastructure for Research (oGIIR)

geo
portti

Finnish Geospatial
Research and
Education Hub

Consortium:



Compiled and edited by: [Augustine-Moses Gbagir \(MSc.\)](#)
[Alfred Colpaert \(Prof\)](#)

DEPARTMENT OF GEOGRAPHICAL AND HISTORICAL STUDIES,
UNIVERSITY OF EASTERN FINLAND.

Contents

Step 1: Install Git	2
Step 2: Install and set-up Docker Desktop (includes Docker-compose)	3
Step 3: Update Windows subsystem for linux (WSL)	4
Step 4: Install Python and Pip.....	5
Step 5: Install WebODM	6
Step:6 Navigate to WebODM folder and start the service.....	6
Step 7: Launch WebODM from any browser.....	7
Step 8: WebODM Dashboard and Processing your first drone images.....	7
Run ODM From Command Line on Windows PC	8
Check Docker Memory on Your PC From WebODM Dashboard and Command Line Interface (CLI)	8
Change Allocated Docker Memory on Your PC From WebODM Dashboard and Command Line Interface (CLI).....	9
WebODM Memory Test on Windows 11 PC	9

List of Figures

Figure 1 Installation prompt of Git	2
Figure 2 Installing and configuring Docker	3
Figure 3 Download of docker	3
Figure 4 Download successfully installed	4
Figure 5 Prompt to update WSL kernel Download successfully installed	4
Figure 6 Updating Windows 11 WSL kernel	5
Figure 7 Starting and running WebODM on Windows 11.....	6
Figure 8 WebODM welcome page.....	7
Figure 9 WebODM dashboard.....	7
Figure 10 WebODM dashboard shows allocated windows memory to docker.....	8
Figure 11 Windows Linux bash Command Line Interface (CLI) shows allocated windows memory to docker	9
Figure 12 WebODM Memory test on Windows 11.....	10

These instructions are modified from the original guides on the WebODM website located [here](#).

Pre-requisites

Install these before installing WebODM

Git

Docker

Docker-compose

Python

Pip

Note: Codes to run are highlighted in **green color**

Step 1: Install Git

Download and Install suitable Git version for your operating system

<https://git-scm.com/downloads>

After you download Git, click on the executable file and a window prompt will open (e.g. figure 1 below). Click next and follow the instructions to install.



Figure 1 Installation prompt of Git

Step 2: Install and set-up Docker Desktop (includes Docker-compose)

Note: If you do not install docker as an admin, then your username may not belong to the docker group and you or an admin will need to add this manually docker user-group.

<https://www.docker.com/>

Note: If you install Docker Desktop, Docker Compose plugin will be installed at the same as well as Docker Engine and Docker Command Line Interface (CLI).

After downloading click on executable and begin the installation process (see figure 2). Click ok to continue and the installation process begins (figure 3). The installation is quite fast (about 20 - 60 secs ?). May depend on the speed of your pc. When installation is done you will see the window with message shown in figure 4.

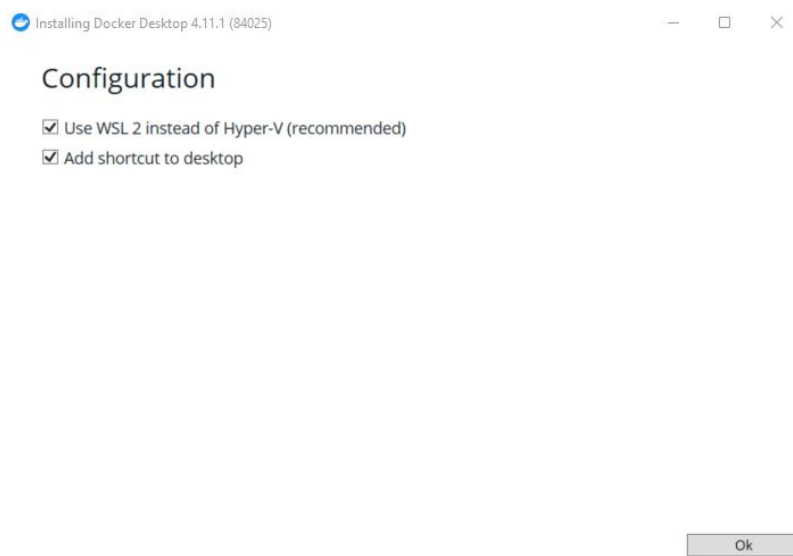


Figure 2 Installing and configuring Docker

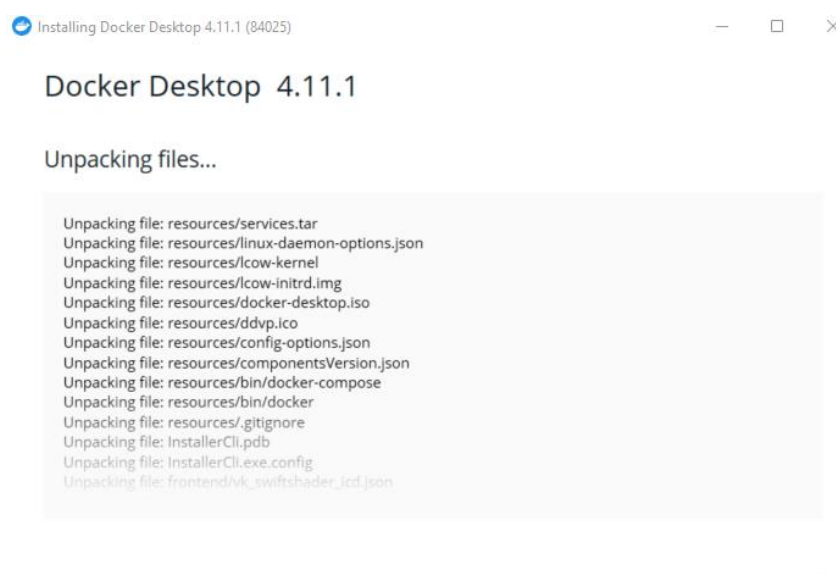


Figure 3 Download of docker

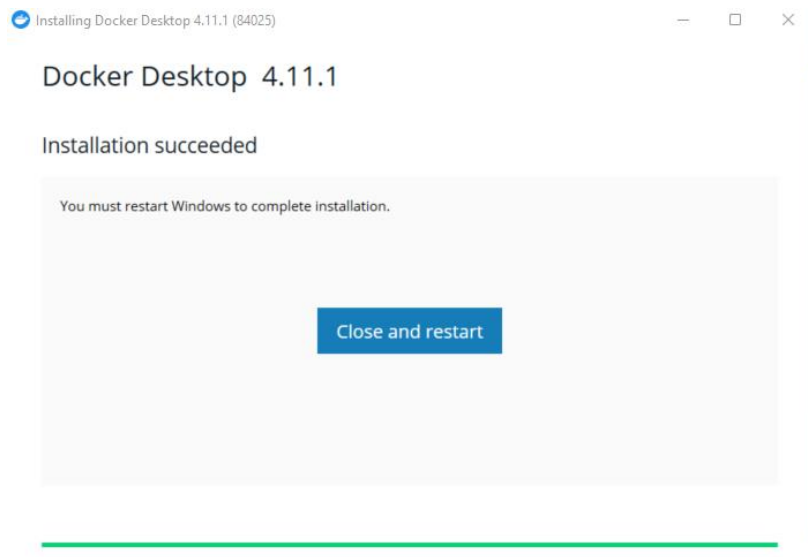


Figure 4 Download successfully installed

Step 3: Update Windows subsystem for linux (WSL)

A successful re-start and clicking the Docker icon will give a prompt to update the WSL kernel (figure 5).

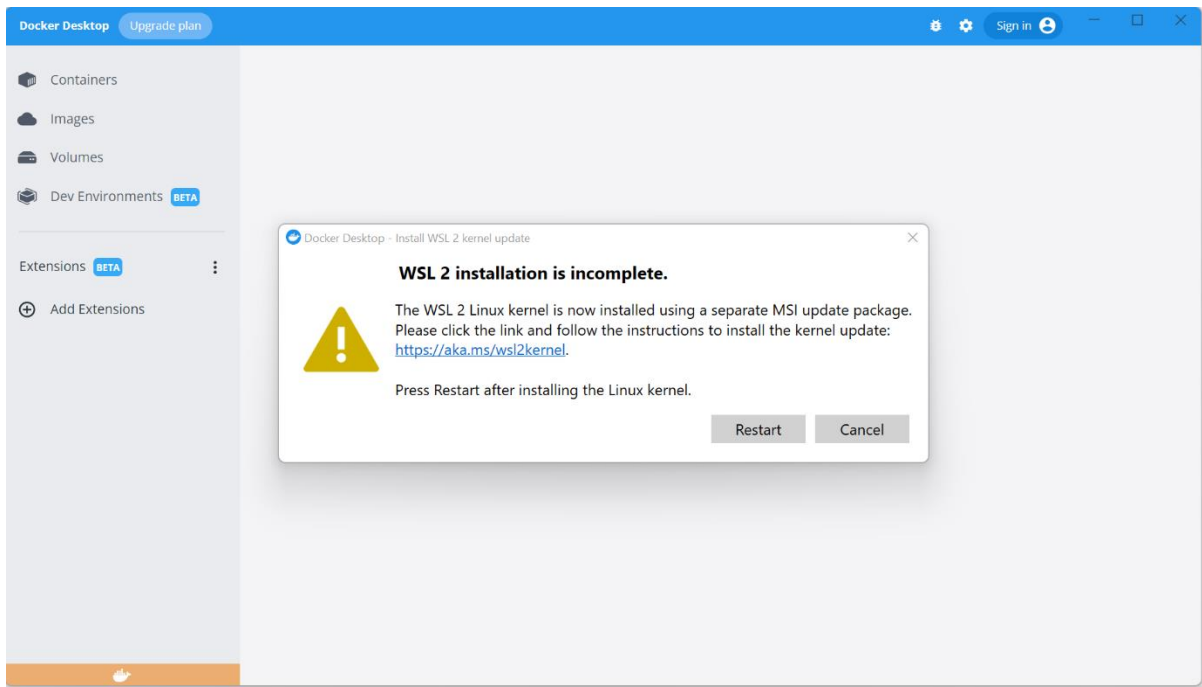


Figure 5 Prompt to update WSL kernel Download successfully installed

Download and update WSL kernel from this link below. Follow the steps as indicated on the website.

<https://docs.microsoft.com/en-gb/windows/wsl/install-manual#step-4---download-the-linux-kernel-update-package>

Note: The steps will be like figures 6 a-g

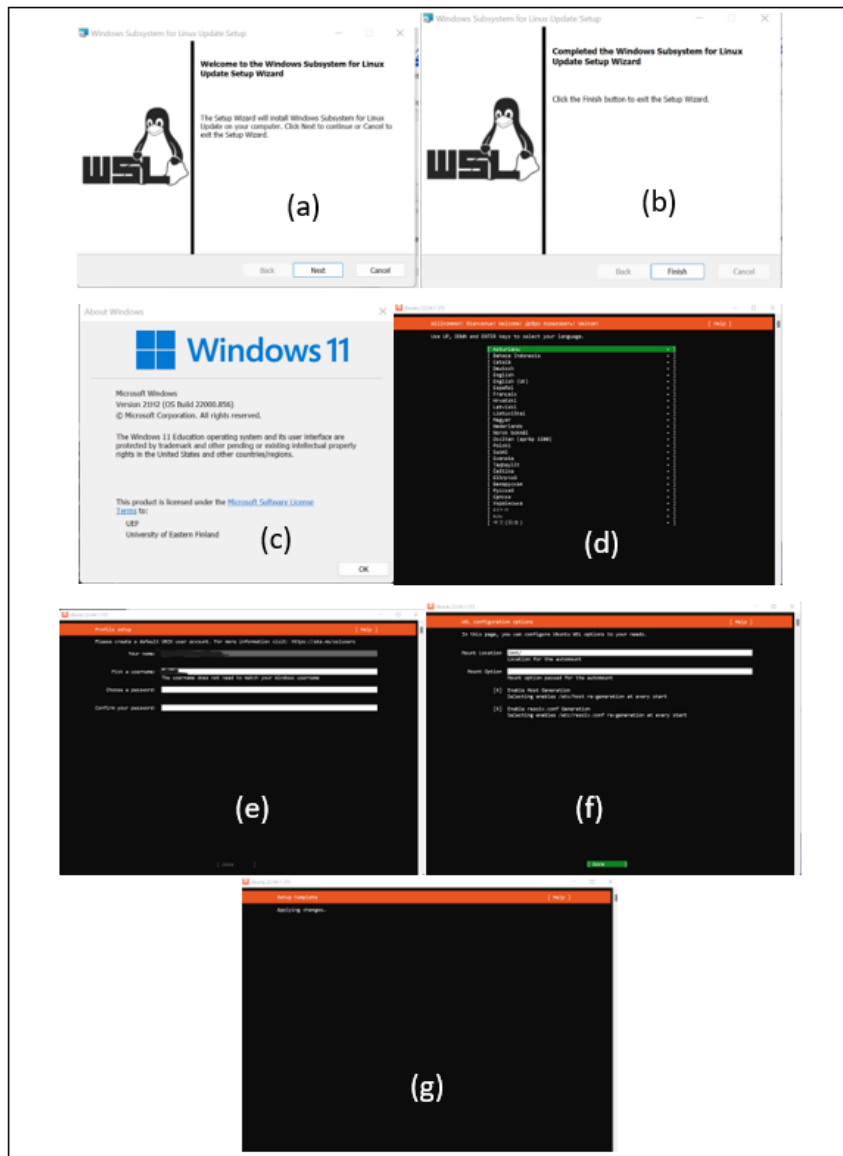


Figure 6 Updating Windows 11 WSL kernel

Step 4: Install Python and Pip

Python + Pip (Pip is installed when you install Python.) Download and click the installer to install python and pip.

<https://www.python.org/downloads/>

Step 5: Install WebODM

Now that you have installed all the pre-requisites dependencies, you can now install WebODM by running the following command in the Windows Power Shell.

```
git clone https://github.com/OpenDroneMap/WebODM --config core.autocrlf=input --depth 1
```

NOTE: The WebODM installation will most likely be installed under your username (i.e. C:\Users\your user name\WebODM).

Step:6 Navigate to WebODM folder and start the service

```
cd WebODM
```

```
./webodm.sh start
```

When you start the service, a similar output such as the one below will be displayed (figures 7 a-h below).



Figure 7 Starting and running WebODM on Windows 11

Note: Using command line to start is only necessary for the first-time after installation. Subsequently, when you start docker and the containers will start running and you can launch the WebODM browser successfully.

Step 7: Launch WebODM from any browser

Now copy and paste this “<http://localhost:8000/>” in any web browser to launch WebODM

If all goes well you should be greeted with a window similar to the one below

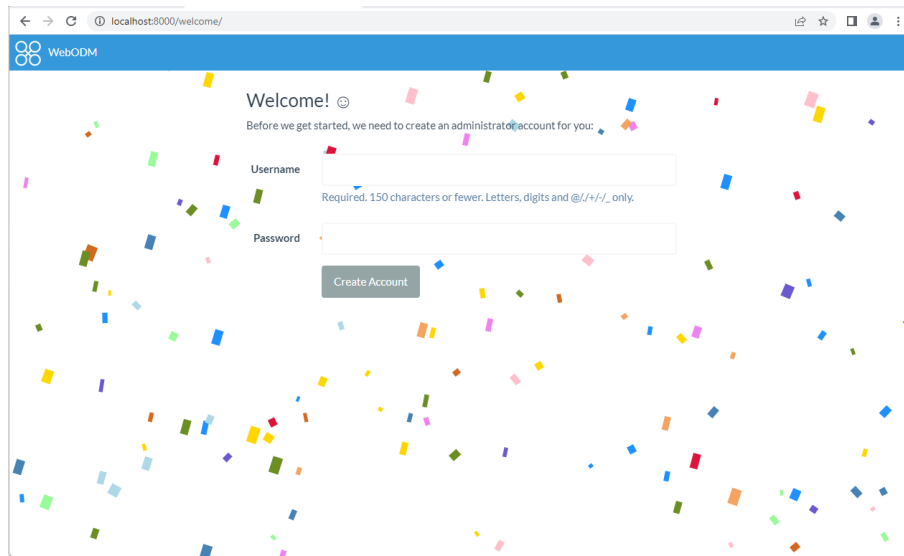


Figure 8 WebODM welcome page

The only thing you need to do now is create a username and password of your choice and you are ready to start processing your drone images.

Step 8: WebODM Dashboard and Processing your first drone images

After you create your WebODM user account you will be greeted with a dash like the one in figure this below.

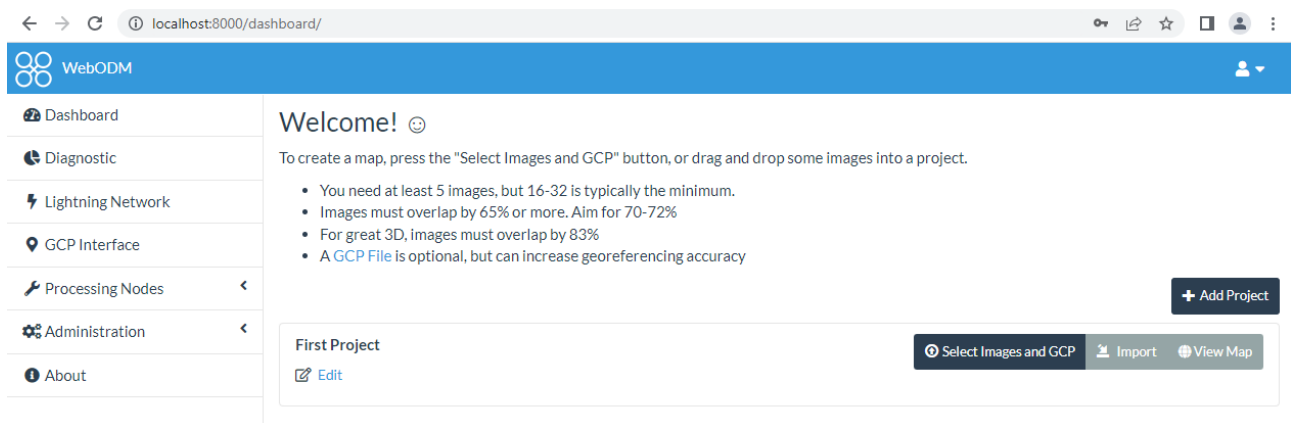


Figure 9 WebODM dashboard

From here you can now create projects load images and start processing.

Run ODM From Command Line on Windows PC

To run ODM from the command line,

first, place your images in a path directory, for example “D:/my/project/images” or anywhere else you prefer.

Now, start the linux power shell in Windows and run either of the two commands below depending on what you need. Command two has option for gpu (highlighted in yellow) version

1. `docker run -ti --rm -v D:/my/project:/datasets/code opendronemap/odm --project-path /datasets`
2. `docker run -ti --rm -v D:/my/project:/datasets/code opendronemap/odm:gpu --project-path /datasets`

Note:

- Next time you want to run WebODM, just launch docker to start the WebODM container image
- Disable the default settings that allows docker to automatically start whenever you power on your pc. This is because docker uses a lot of your pc memory when running

Check Docker Memory on Your PC From WebODM Dashboard and Command Line Interface (CLI)

By default, 50 % of your pc memory is allocated to docker (figure 10 below). You can check this from the WebODM dashboard by clicking “Diagnostic” on left panel. In the next section, we discuss how you can modify memory allocated to docker on your pc.

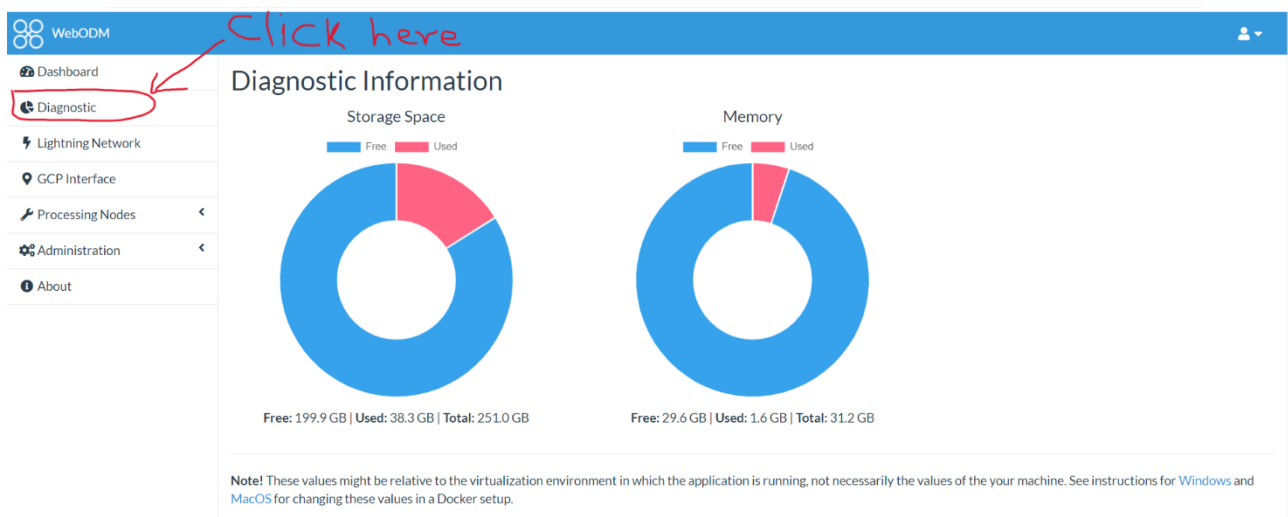


Figure 10 WebODM dashboard shows allocated windows memory to docker.

From the linux window power shell, you can also check the allocated memory (figure 11 below) to docker by running the command below.

`free -mh`

```
root@kali:~# free -mh
              total        used          free        shared  buff/cache   available
Mem:           31Gi        3.2Gi         26Gi          22Mi         1.5Gi         27Gi
Swap:          8.0Gi          0B          8.0Gi
root@kali:~#
```

Figure 11 Windows Linux bash Command Line Interface (CLI) shows allocated windows memory to docker

Change Allocated Docker Memory on Your PC From WebODM Dashboard and Command Line Interface (CLI)

To change the memory allocated to docker, you must create a “.wslconfig” file and place it c:\users\here.

Here is an example on how you can modify the global settings for docker running on WSL 2 by creating a “.wslconfig” file.

First, close all running instances of docker or WSL 2.

Open a text editor and enter the text in red below. You can add other settings as you deem fit and meets you needs and the resources of your pc.

[wsl2]

memory=25 GB # Limits VM memory in WSL 2

processors=6 # Makes the WSL 2 VM use 4 virtual processors

localhostForwarding=true # Boolean specifying if ports bound to wildcard or localhost in the WSL 2 VM should be connectable from the host via localhost:port.

Now save this file as a .wslconfig format. No adding a .txt or other file type at the end. Remember to place this file in the location c:\users\

WebODM Memory Test on Windows 11 PC

Based on the memory configure (figure 10 above) we ran a quick memory test by processing 100 – 3000 images. Here we show only the result of 2500 and 3000 images (figure 12 below). In this test, we used the default settings of WebODM. We processed the 2500 images successfully, but the 3000 images failed due to insufficient memory. Note that adding more customization options will increase memory demand.

Ebee-Pallas-2500-images

Select Images and GCP Import View Map

1 Tasks Edit

Muonio - 6/5/2019
2500
04:14:15
✓ Completed
⋮

Created on: 10/27/2022, 10:46:18 AM

Processing Node: node-odm-1 (auto)

Options: auto-boundary: true, dsm: true

Average GSD: 13.49 cm

Area: 4,569,253.62 m²

Reconstructed Points: 67,376,259

Task Output: On Off

Download Assets
View Map
View 3D Model
Restart
Delete
Edit

Ebee-Pallas-3000-images

Select Images and GCP Import View Map

1 Tasks Edit

Muonio - 6/5/2019
3000
05:45:08
✗ Not enough memory
⋮

Created on: 10/25/2022, 9:57:23 AM

Processing Node: node-odm-1 (auto)

Options: auto-boundary: true, dsm: true

Task Output: On Off

Restart
Delete
Edit

Figure 12 WebODM Memory test on Windows 11.