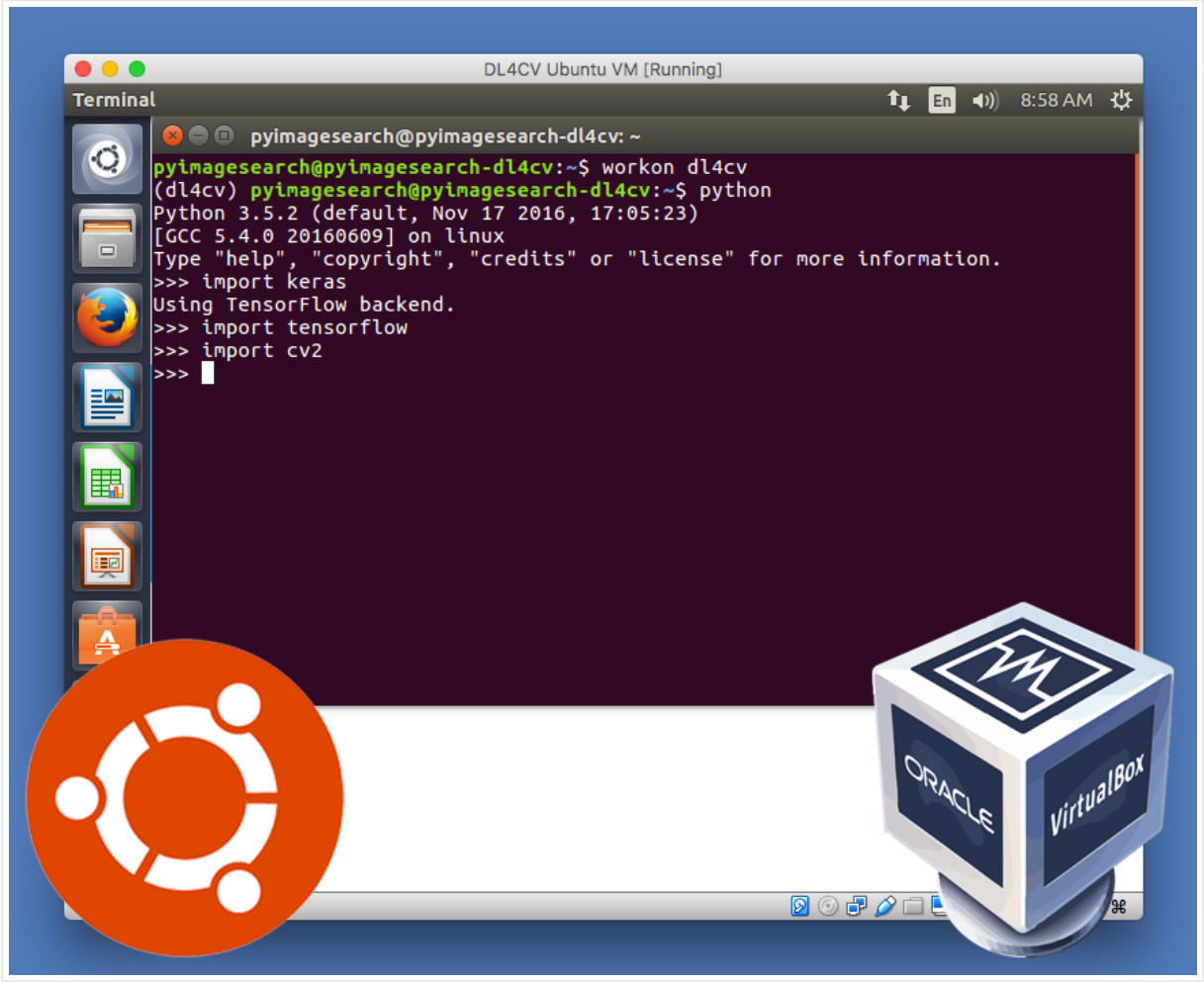
Your deep learning + Python Ubuntu virtual machine

by **Adrian Rosebrock** on September 22, 2017 in **Deep Learning**, **DL4CV**



When it comes to working with deep learning + Python I *highly recommend* that you use a Linux environment.

Deep learning tools can be more easily configured and installed on Linux, allowing you to develop and run neural networks quickly.

Of course, configuring your own deep learning + Python + Linux development environment can be quite the tedious task, *especially* if you are new to Linux, a beginner at working the command line/terminal, or a novice when compiling and installing packages by hand.

In order to help you jump start your deep learning + Python education, I have created an ***Ubuntu virtual machine*** with all necessary deep learning libraries you need to successful (including *Keras*, *TensorFlow*, *scikit-learn*, *scikit-image*, *OpenCV*, and others) ***pre-configured*** and ***pre-installed***.

This virtual machine is part of all three bundles of my book, *Deep Learning for Computer Vision with Python*. After you purchase your copy you'll be able to download the virtual machine and get started with deep learning immediately.

In the remainder of this tutorial I'll show you:

- How to download and install *VirtualBox* for managing, creating, and importing virtual machines.
- How to import the pre-configured *Ubuntu* virtual machine for deep learning.
- How to access the *pre-installed deep learning libraries* on the virtual machine.

Let's go ahead and get started.

Your deep learning + Python virtual machine

Your purchase of *Deep Learning for Computer Vision with Python*includes a pre-configured Ubuntu virtual machine for deep learning. In the following sections I'll show you how easy it is to import your Ubuntu deep learning virtual machine.

This tutorial is broken down into three parts to make it easy to digest and understand:

1. Download and install VirtualBox.
2. Download and import your pre-configured Ubuntu deep learning virtual machine.
3. Access the Python development environment inside the deep learning virtual machine.

Step #1: Download and install VirtualBox

The first step is to download VirtualBox, a free open source platform for managing virtual machines.

VirtualBox will run on ***macOS***, ***Linux***, and ***Windows***.

We call the physical hardware VirtualBox is running on your ***host machine***. The virtual machine that will be imported into VirtualBox is the ***guest machine***.

To install VirtualBox, first visit the downloads page and then select the appropriate binaries for your operating system:

Figure 1: VirtualBox downloads.

From there install the software on your system following the provided instructions — I'll be using macOS in this example, but again, these instructions will also work on Linux and Windows as well:

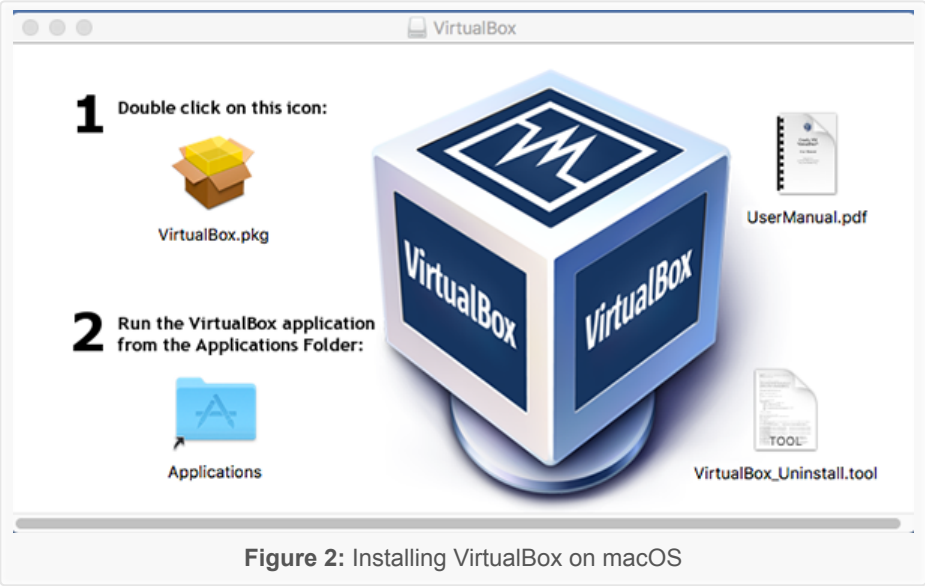


Figure 2: Installing VirtualBox on macOS

Step #2: Download your deep learning virtual machine

Now that VirtualBox is installed you need to download the pre-configured Ubuntu virtual machine associated with your purchase of *Deep Learning for Computer Vision with Python:*

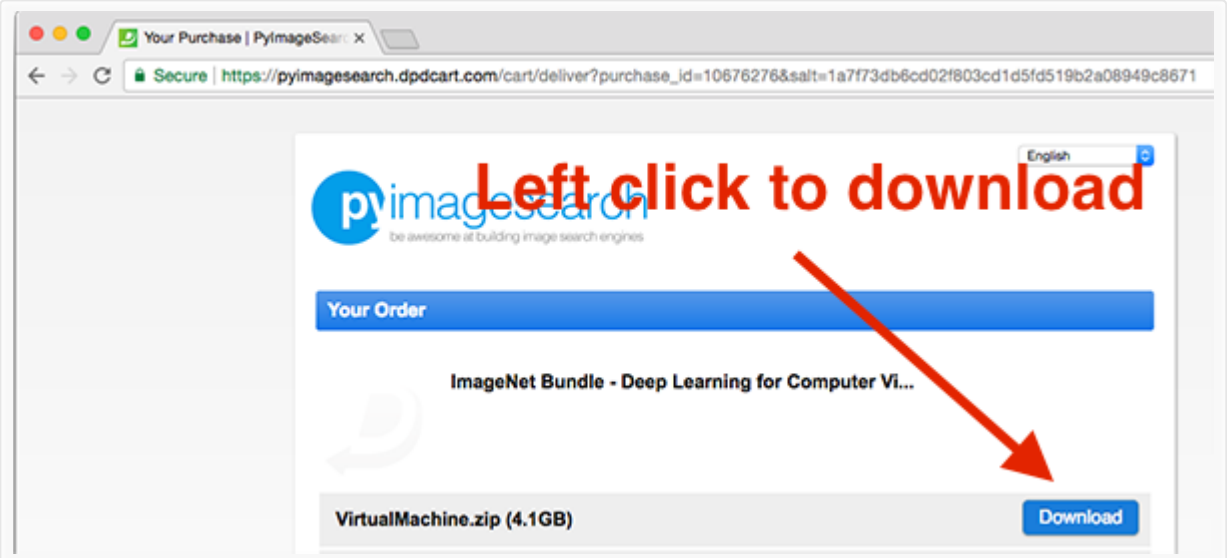


Figure 3: Downloading the pre-configured Ubuntu deep learning virtual machine.

The file is approximately 4GB so depending on your internet connection this download make take some time to complete.

Once you have downloaded the `VirtualMachine.zip` file, unarchive it and you'll find a file named `DL4CV Ubuntu VM.ova` . I have placed this file on my Desktop:

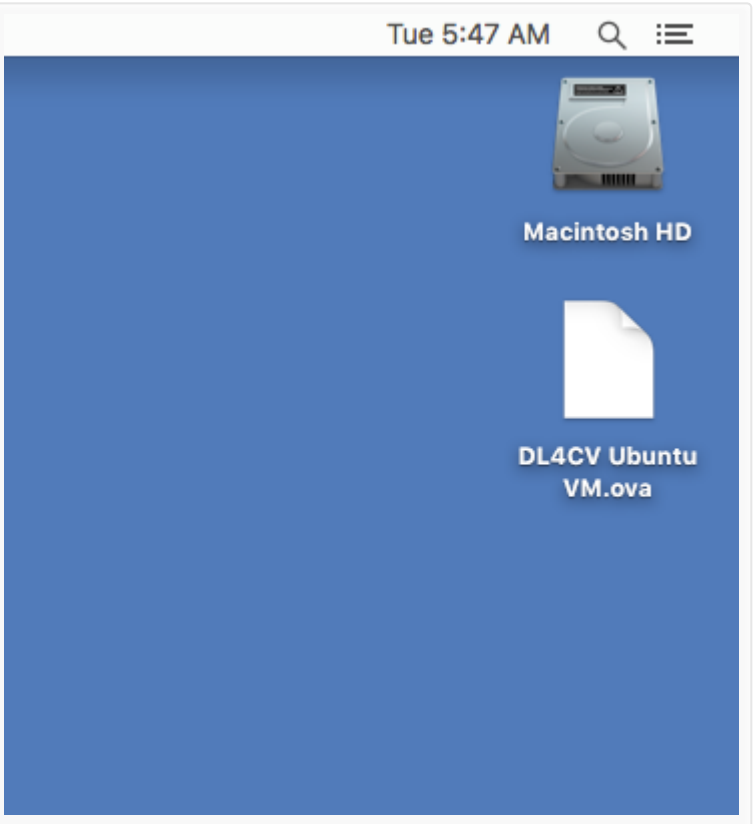


Figure 4: The DL4CV Ubuntu VM.ova file.

This is the actual file that you will be importing into the VirtualBox manager.

Step #3: Import the deep learning virtual machine into VirtualBox

Go ahead and open up the VirtualBox manager.

From there select `File => Import Appliance...` :

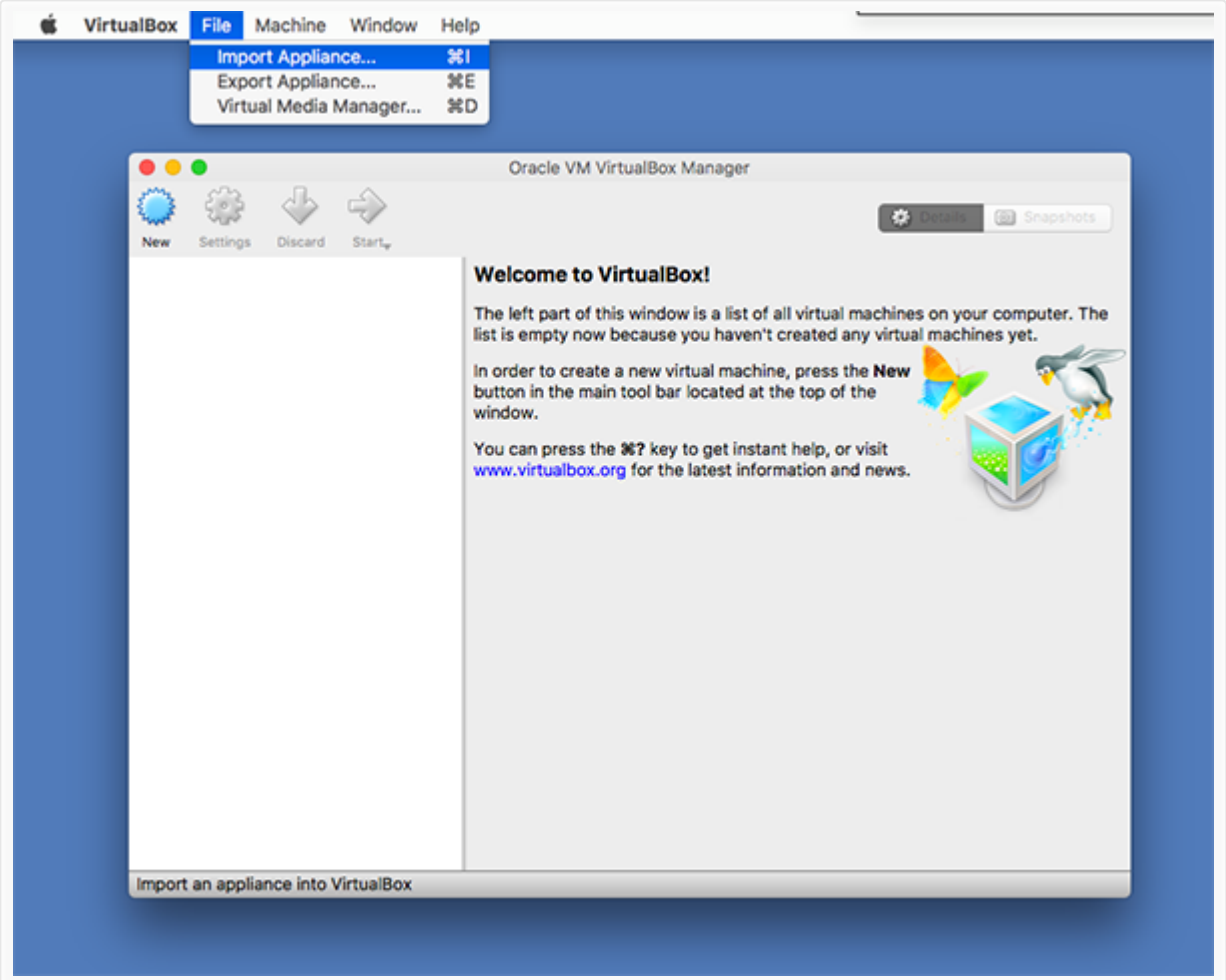


Figure 5: Importing the pre-configured Ubuntu deep learning virtual machine.

Once the dialog opens you'll want to navigate to where the `DL4CV UbuntuVM.ova` file resides on disk:

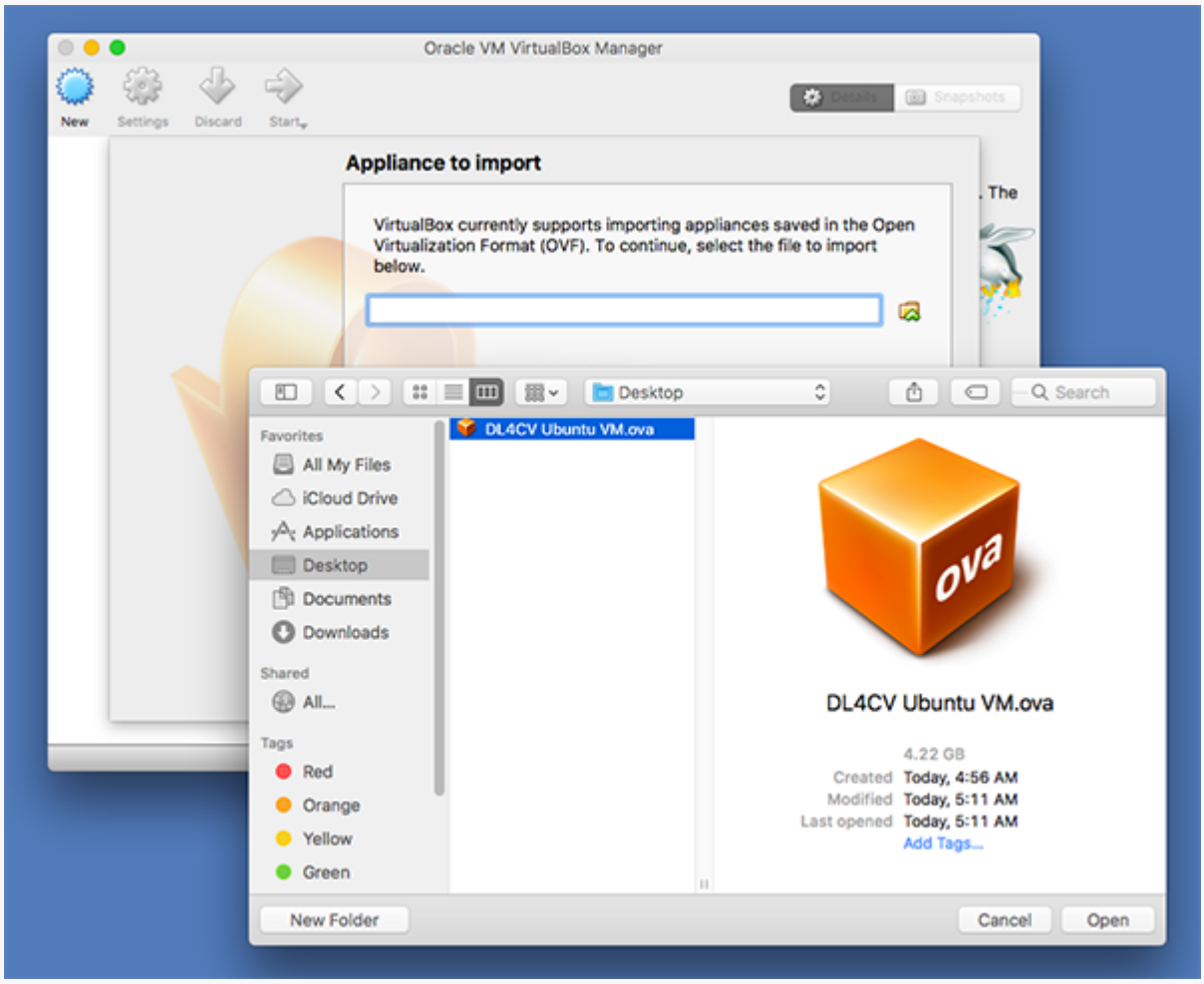


Figure 6: Selecting the pre-configured Ubuntu deep learning virtual machine.

Finally, you can click *"Import"* and allow the virtual machine to import:

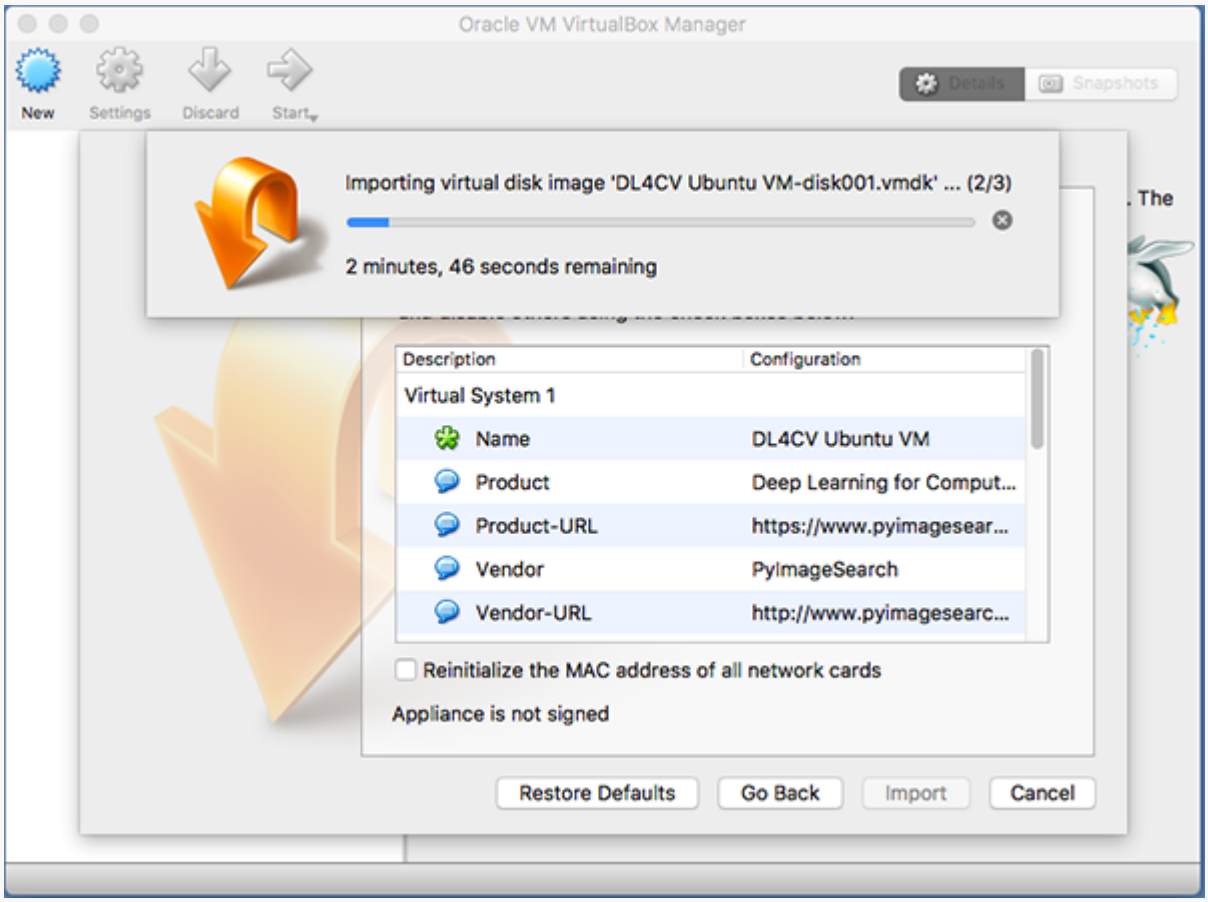


Figure 7: Importing the Ubuntu deep learning virtual machine may take 3-4 minutes depending on your system.

The entire import process should take only a few minutes.

Step #4: Boot the deep learning virtual machine

Now that the deep learning virtual machine has been imported we need to boot it.

From the VirtualBox manager select the *"DL4CV Ubuntu VM"* on the left pane of the window and then click *"Start"*:

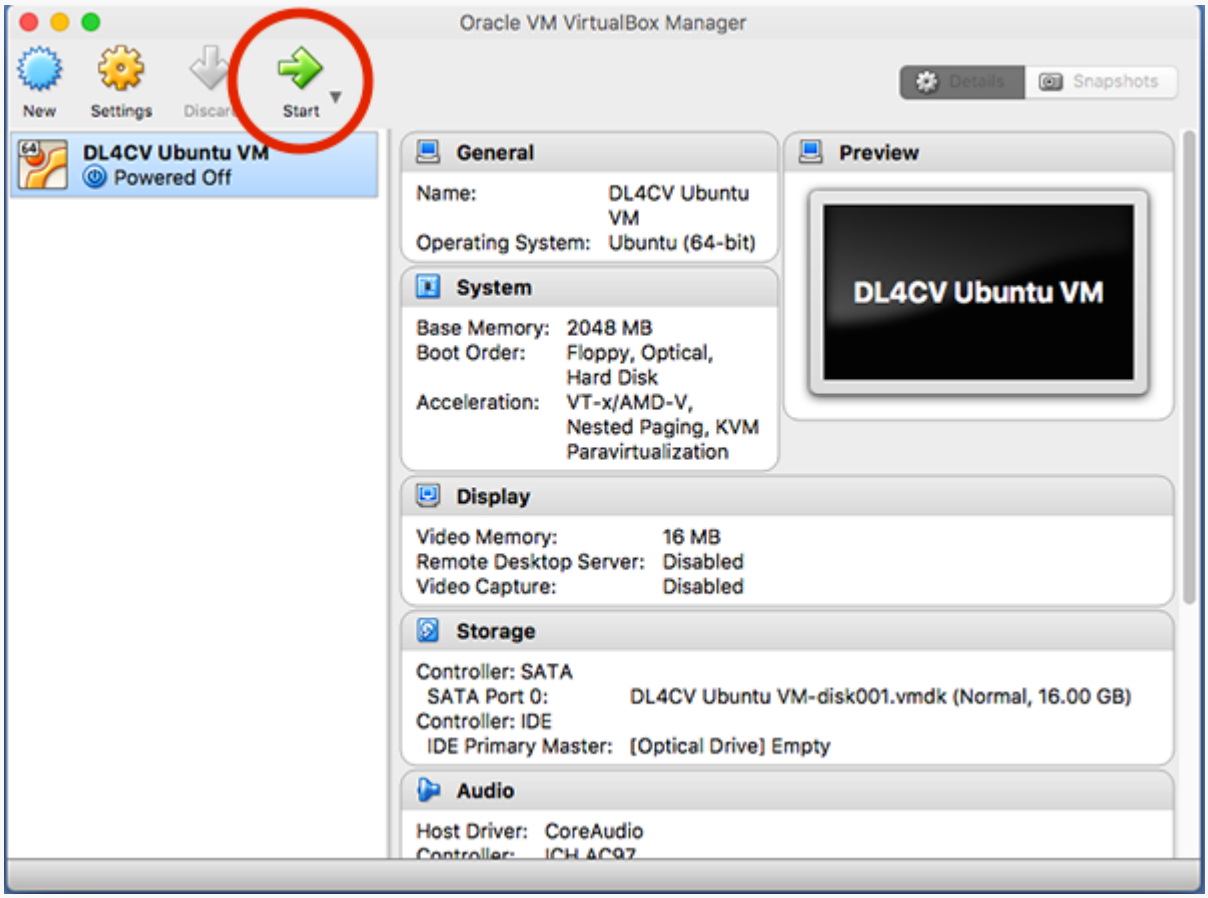


Figure 8: Booting the pre-configured Ubuntu deep learning virtual machine.

Once the virtual machine has booted you can login using the following credentials:

- **Username:** `pyimagesearch`
- **Password:** `deeplearning`

Figure 9: Logging into the deep learning virtual machine.

Step #5: Access the deep learning Python virtual environment

The next step after logging into the VM is to launch a terminal:

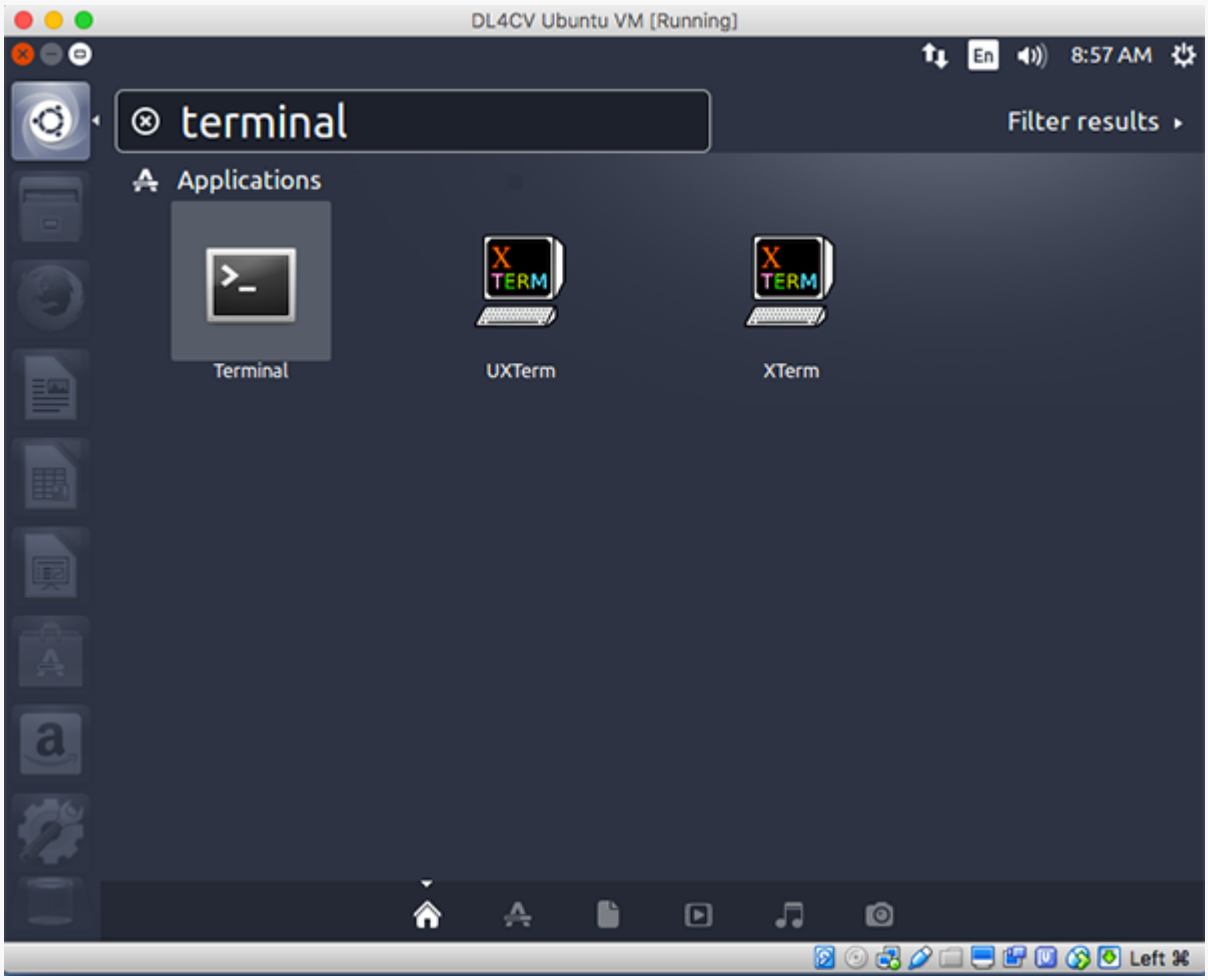


Figure 10: Launching a terminal window.

From there, execute `workon dl4cv` to access the Python + deep learning development environment:

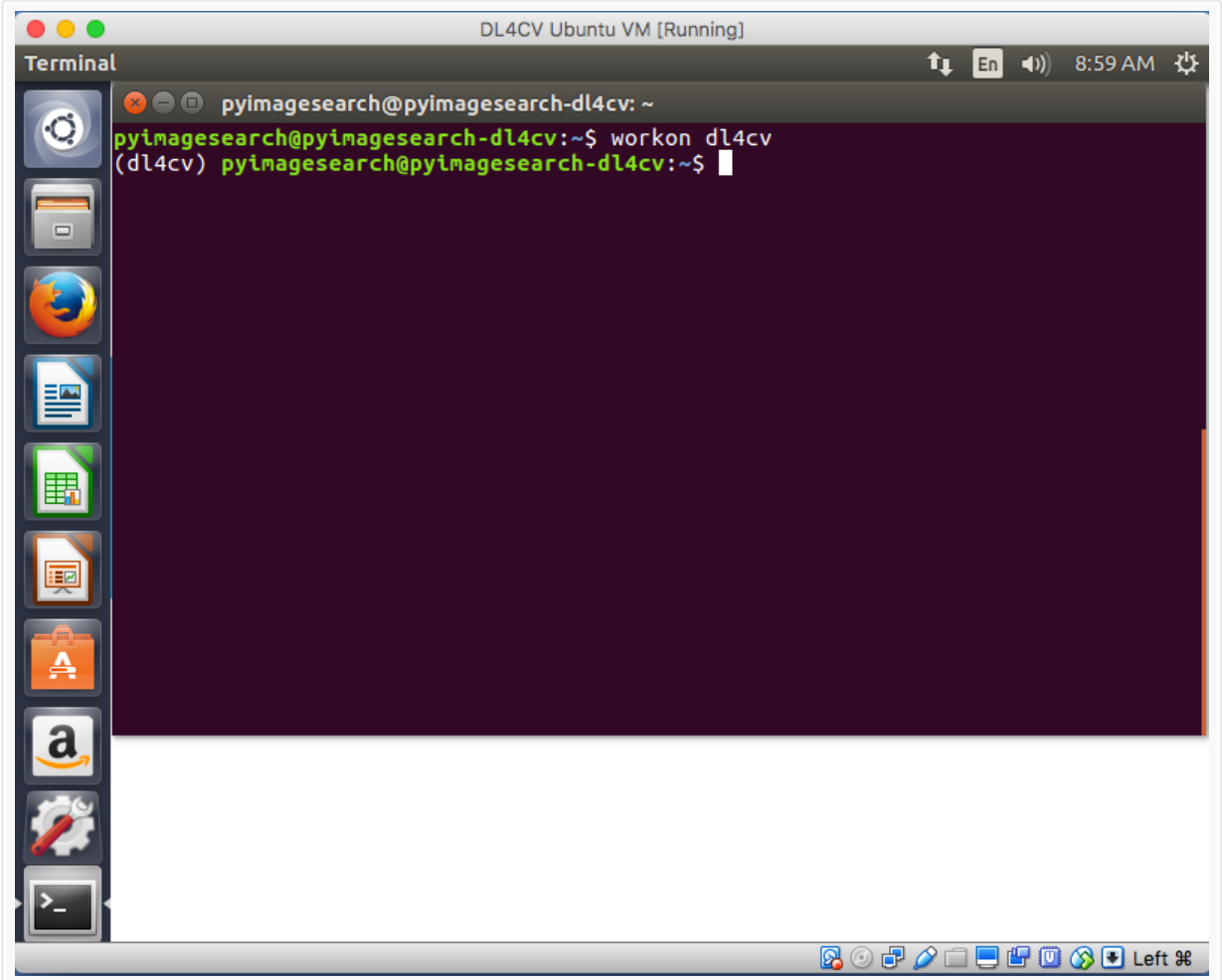


Figure 11: Accessing the *dl4cv* deep learning + Python development environment.

Notice that my prompt now has the text `(dl4cv)` preceding it, implying that I am inside the `dl4cv` Python virtual environment.

You can run `pip freeze` to see all the Python libraries installed.

I have included a screenshot below demonstrating how to import Keras, TensorFlow, and OpenCV from a Python shell: