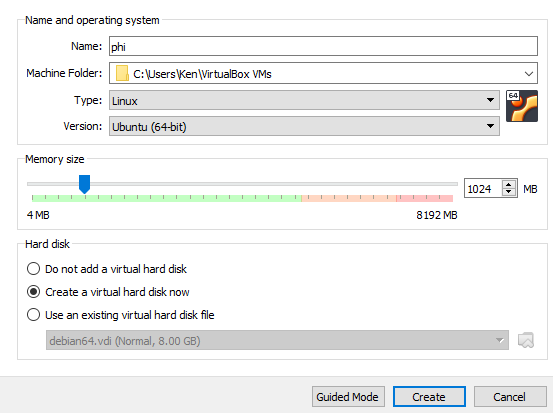
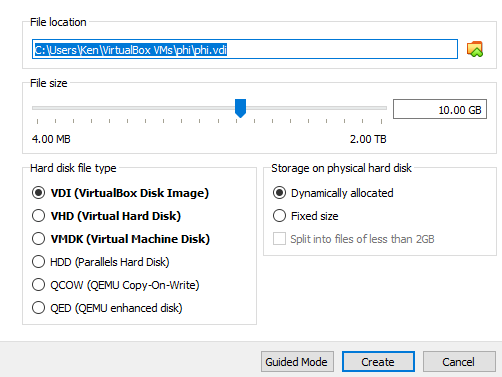
Step 1 - Setup Ubuntu

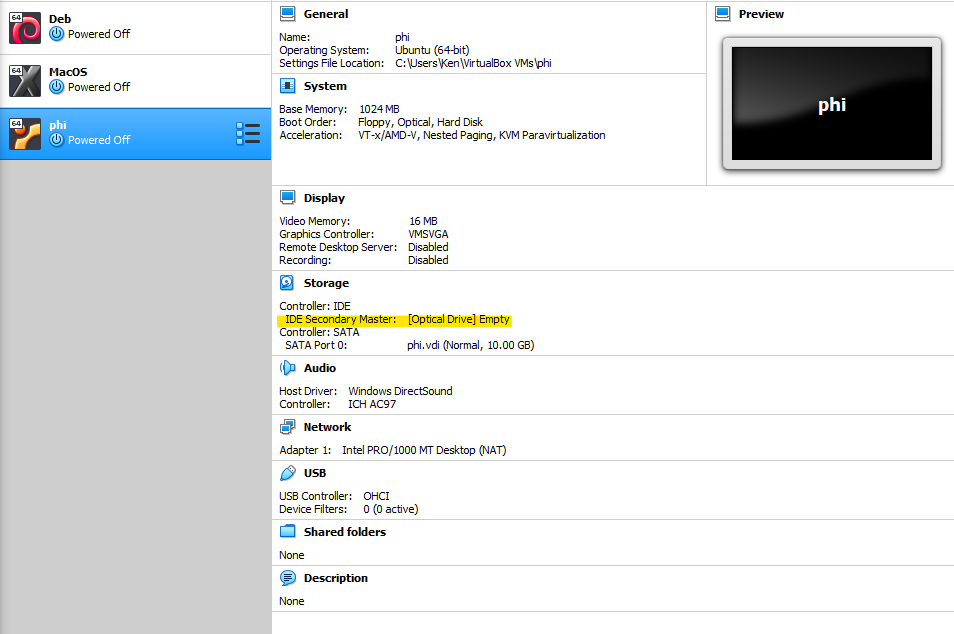
* Open VirtualBox
* Click “New”, give it a name (phi, for instance). Follow the screenshot:



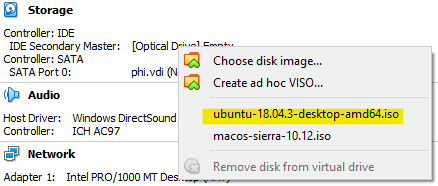
* Click “Create” for both screenshots above and below:



* Click the highlighted portion:



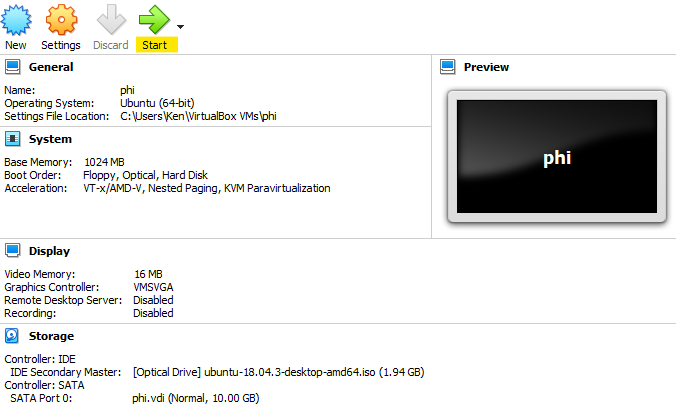
* Select the iso file that you just downloaded from [here](https://drive.google.com/open?id=1XVFw3Q7ANxTVmL-MHsb1dl5VWM0HhWz3):



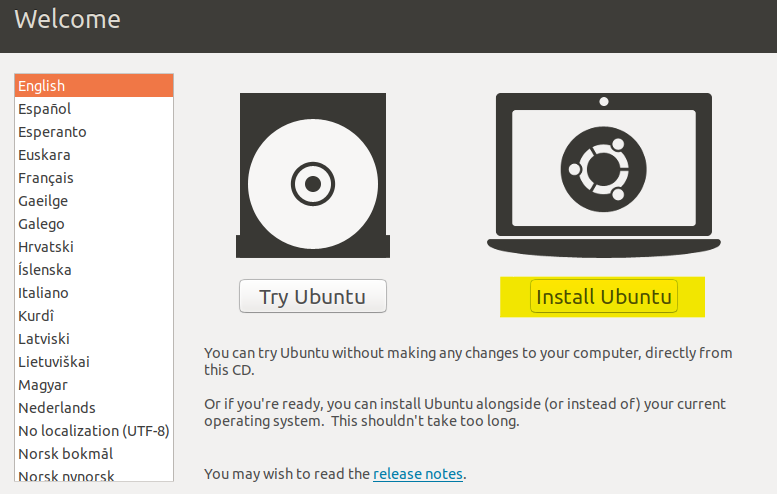
* Then, click “Settings”



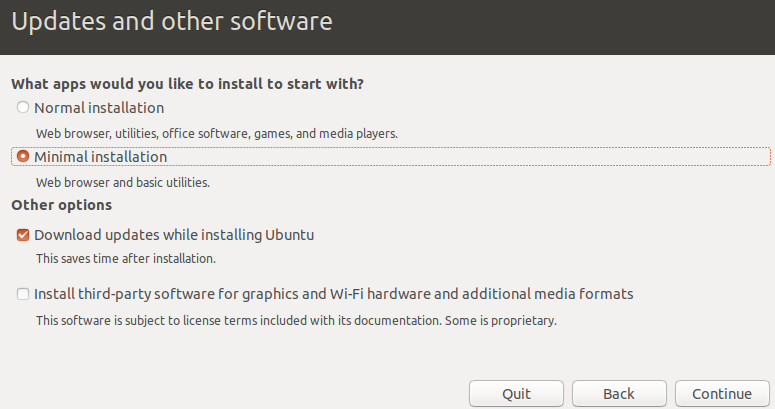
* In the “System” section, go to the “Motherboard” tab and set Base Memory to 2048MB.
* Then, the “Display” section. In the “Screen” tab, set Video Memory to 128MB or the highest you could do
* Click “Ok”
* Then, click “Start”



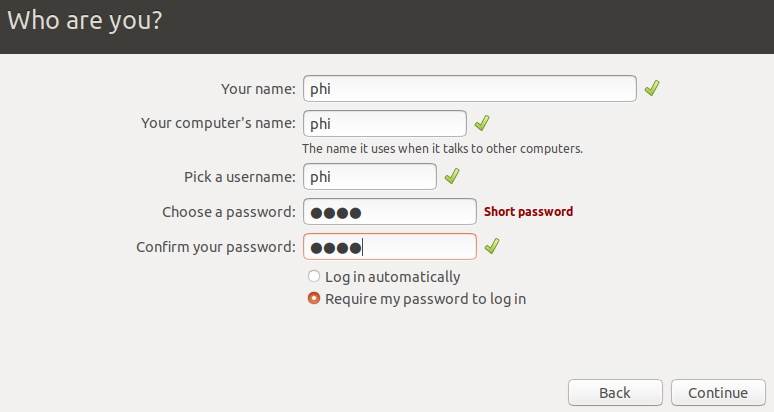
* Follow through the instructions of the Ubuntu installation:



* If there is any unresponsiveness, wait for ~2 mins. If the problem persists, power off the machine and try again.
* Then, select your language of choice and “Continue”.
* Make sure you have the following choice in the next section:



* Click “Continue” and in the next section, select “Erase disk and install Ubuntu”. Follow through.
* Customize your timezone and your user account accordingly. Try to keep it simple.



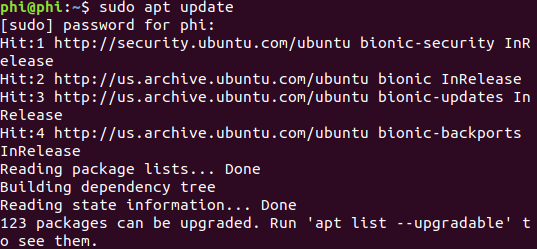
* Let it install.
* Restart VM by clicking “Restart Now”, then “Power Off” the VM.
* Click “Start” in the VB Dashboard to reboot Ubuntu.
* Your Ubuntu is ready for the next step!

Step 2 - [Setup](https://www.digitalocean.com/community/tutorials/how-to-install-python-3-and-set-up-a-programming-environment-on-an-ubuntu-18-04-server) Python 3, virtual environment, and Git

* Open terminal and type the following: (agree on everything asked)

$ sudo apt update

* You should get something similar to this:



* Proceed by typing:

$ sudo apt -y upgrade

* Let the terminal do its work.
* Then, type:

$ python3 -V

* You should have an output similar to this

“Python 3.X.X”, where X represents a number. I had 3.6.8.

* Then type:

$ sudo apt install -y python3-pip

* Followed by:

$ sudo apt install build-essential libssl-dev libffi-dev python3-dev

* Now, set up our virtual environment. Type the following in terminal:

$ sudo apt install -y python3-venv

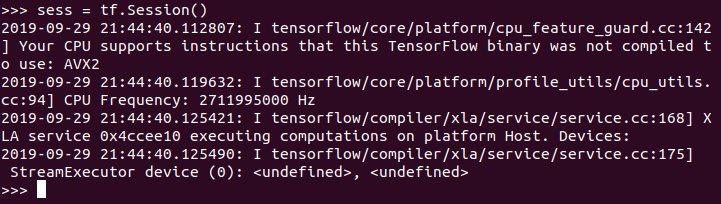
* Install Git by typing:

$ sudo apt-get update

$ sudo apt-get install git

Step 3 - Installing TensorFlow (TF)

* Follow the instructions mentioned in **Step 2** of this [guide](https://www.digitalocean.com/community/tutorials/how-to-install-and-use-tensorflow-on-ubuntu-18-04).
* While following the instructions, you may encounter something as such:



* This isn’t something to stress about. You can choose to ignore it or [not](https://stackoverflow.com/questions/47068709/your-cpu-supports-instructions-that-this-tensorflow-binary-was-not-compiled-to-u).
* Now, proceed to **Step 3** of the link mentioned at the beginning of our third step.

Step 4.1 - Trying things out!

* Suppose you’ve finished all the steps in the guide above, you should have tried running your classify\_image.py for at least once.
* In fact, this is as far as we could get during our meeting because your president is retarded.
* Anyways, TO TRY DIFFERENT IMAGES, you must be in the

~/tf-demo/models/tutorials/image/imagenet directory.

* Then, do

$ python classify\_image.py --image\_file /home/user\_name/Downloads/name\_of\_image

* user\_name refers to the name you gave to your computer. In my case, it was phi.
* name\_of\_image refers to the name of the image you just downloaded/saved. Be sure to include its extension (.jpg/ .jpeg/ .png) as well.
* Your path may be different than mine, but when the article mentioned “Absolute Path”, it means the directory of your image.
* You can identify the directory of your image by navigating to the directory you saved the image with the help of the following commands:

$ cd name\_of\_directory <- Takes you “into” the specified directory.

$ cd.. <- Takes you “out” by one directory.

$ cd <- Takes you “out” all the way.

$ pwd <- Prints the path of your directory so that you can

type it down in the

~/tf-demo/models/tutorials/image/imagenet directory.

Shoutout to Brian Yu and Matthew Jiang for pointing out that we should try including image\_file and -- as part of our argument. In fact, we did, but we were so close to getting the syntax right. At least it's solved now :)

Step 4.2 - Try different images!

* Here we have a picture of a dog



* If I let the program to classify it, it would result in something like this:

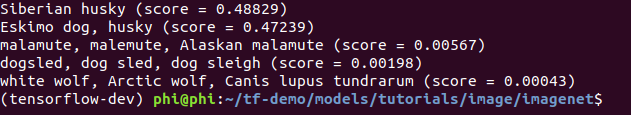


|

|

|

V



Pretty accurate isn’t it? Assuming that “Siberian husky” and “Eskimo dog” are referring to the same species.

* What if I try something like this:



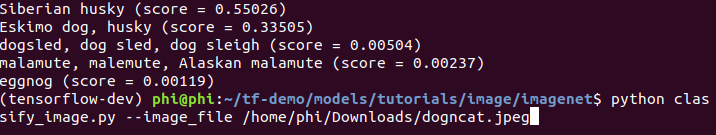


|

|

|

V



* Not as much anymore, isn't it? I mean, where’s the cat?
* This is just one of many examples that demonstrated the limitations of computer vision, but that is not to say that it will be like this forever.
* We can train our program to be specialized in certain objects by supplying data sets that solely has the images of the targeted object. For instance, if I were to train my image classifier solely for the purpose of identifying cats, I am certain that it can tell that there is a cat, but not a dog. Thus, to have it to identify both, I will have to supply images of both animals to increase its accuracy, which sounds easy but it's not.

Here’s another example that shows the level of specialization of our image classifier:



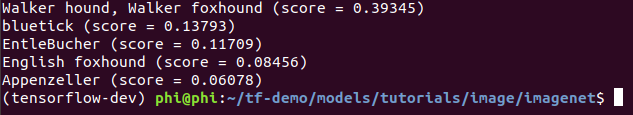


|

|

|

V



* It is evident that our classifier was not trained to identify ice cream other than dogs.
* Do try it out with other images and see what you get!