**User Manual for setting up a stimulus computer for running FlyFly at high temporal resolution**

This document explains how to set up a stimulus computer with a **Nvidia Geforce GTX 970 Graphics card**. The computer may already have Windows installed. It will explain how to install **Ubuntu 20.04** and **Matlab 2016a** with the **Psychtoolbox 3.0.16 - Flavor: beta**. The manual can be used for set-ups with one stimulus screen (most ephys), or 3 stimulus screens (tethered flight arena)

Note that the latest version of Matlab cannot run FlyFly. We use **Matlab 2016a**, and **Psychtoolbox** 3.0.14 for intra, **3.0.16** for intra/extracellular and behavior. Also note that FlyFly will not run with newer **graphics cards**. It seems to work well with **2080Ti and earlier** versions, but not later ones, such as 3080.

If you are using a computer with Mac OS X, NOTE THAT the makers of PsychToolbox strongly discourage using PsychToolbox on a Mac!

Our recommended environment for an **experimental** setup is some form of Linux (for other versions of Linux, you will have to use these instructions as broad guidelines only).

PLEASE FOLLOW EACH STEP IN THIS DOCUMENT AND DO NOT SKIP ANY OF THEM.

Before installing Ubuntu, make sure you back up your Windows 10 system first. The recommended way to back up your system is using “File History”.

* Open the **Control Panel** and go to **File History**.
* At the bottom of the left panel, find **System Image Backup**, click the link.
* Choose a location to save your system image backup (Windows 10 will automatically create a partition for your system image backup).
* Click **Next**, confirm your settings and click **Start backup**.
* Address bar will show ‘Control panel> System and Security> Backup and Restore (windows 7)’. This does not mean you are backing up windows 7.
* To save externally, insert a new DVD/CD RW into a drive, and click **Create a system repair disc**.

NOTE:Please only backup a clean system to save your hard drive’s space.For graphic instructions of Win 10 system backup, please search “Backup Windows 10” online.

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**Step 1: Install Ubuntu 16.04**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

Now you are ready to install Ubuntu 16.04. You can either install it alongside windows or on its own. Please make sure you read **section 1.10** as it contains important updates!

**1.1 Create a bootable Ubuntu USB drive**

NOTE: we have a bootable Ubuntu 16.04 and 20.04 USB in the lab already.

To create a bootable Ubuntu USB drive, you need to:

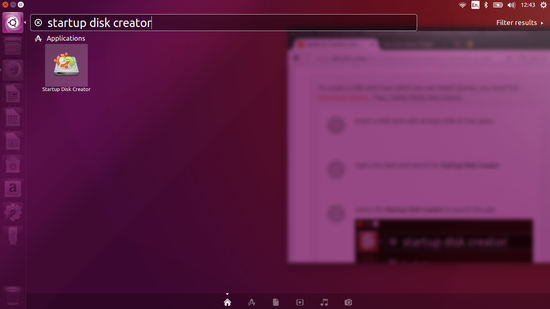
1. Have a USB stick with at least **2GB** of free space
2. Have a PC running Ubuntu or Windows
3. Download Ubuntu from <http://www.ubuntu.com/download> and copy it to the PC.

\*\*\*If you have a computer running Windows system **(RECOMMENDED METHOD)**

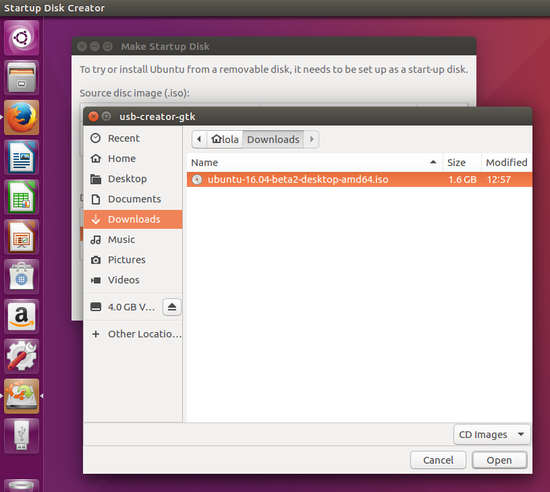
* Download Rufus USB installer from [https://rufus.akeo.ie](https://rufus.akeo.ie/)
* Search **Create a bootable Ubuntu USB drive** online for instructions. Essentially the steps are:
* Download the appropriate Ubuntu image from the official Ubuntu website as an .iso file
* Insert a new USB stick – this will be formatted so should have no valuable data on it.
* Start up Rufus.
* Use Rufus to copy the .iso file onto the USB and make it bootable (takes about 5 minutes)

\*\*\*If you already have another computer with a running Ubuntu system

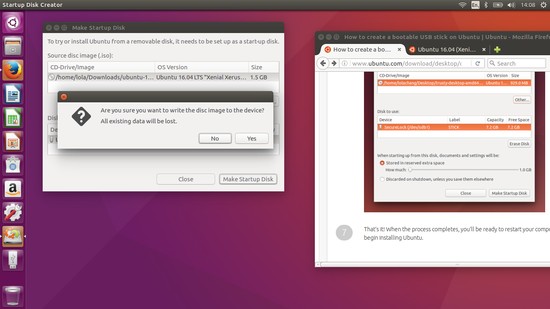
* The easiest way to create a bootable Ubuntu USB drive is using **Startup Disk Creator**, which is a built-in software in the Ubuntu system. **EDIT: Note this does not always work! Some computers in our lab have a corrupted version of Ubuntu, resulting in the next setup you install propagating the error. This method is NOT recommended.**
* Open the **dash** and search for **Startup Disk Creator,** andelect the **Startup Disk Creator** to launch the app:



* If the ISO file is not found automatically, click **Other** to choose the downloaded ISO file and click **Open**



* Select the USB stick in the bottom box and click **Make Startup Disk,** then **Yes.**

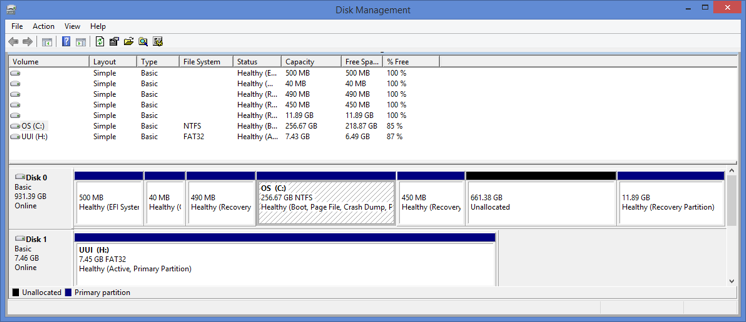


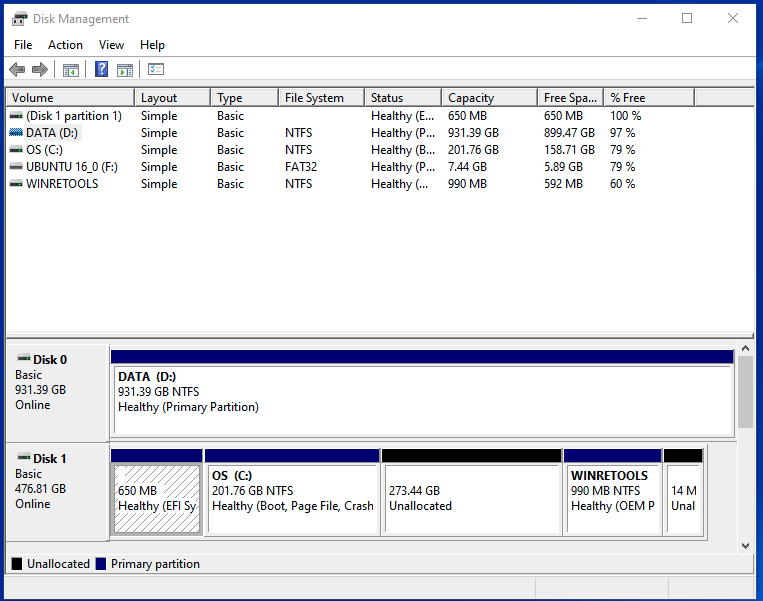
* When the process completes, you’re ready to install Ubuntu.

NOTE: During the process, the admin password may be asked, so please stay next to your computer and wait for the process to complete.

**1.2A Shrink your Windows partition**

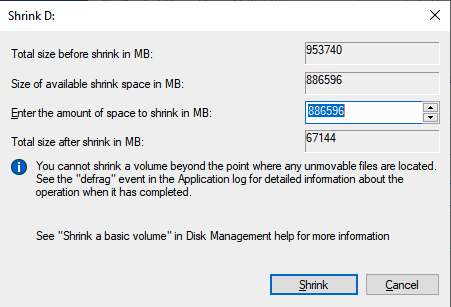
* IMPORTANT: the instructions in this section, and later under 1.6 ‘Install Ubuntu’, are for the simplest setup, in which all of your Ubuntu installation will reside together on a single hard disk drive (HDD). This will work, but often a computer will feature more flexible storage arrangements, e.g. solid state drives (SSD) as well as traditional HDDs, in which case you may want to utilise the disks in a better way. **If so, go to 1.2B below**
* In order to install Ubuntu, you need to make space for it. Press the **Window Key** on the keyboard and click the magnifying glass in the top right corner. In the search box start typing **Partitions**. Click **Create and format partitions**, which brings you to **Disk Management** screen:





* Right click OS (C:) volume and select **Shrink volume**

NOTE: You can choose how much you want to shrink the drive by. Please make sure that you leave enough for each system to perform normally. Please check online for the minimum amount you need to ensure your system performs normally.



* The number in the ‘Enter the amount of space to shrink in MB’ box is effectively the amount of memory you will take from Windows and give to Ubuntu for installation.
* Click **Shrink**. When the process is finished, you can see the **unpartitioned space**. This is where you will install Ubuntu.

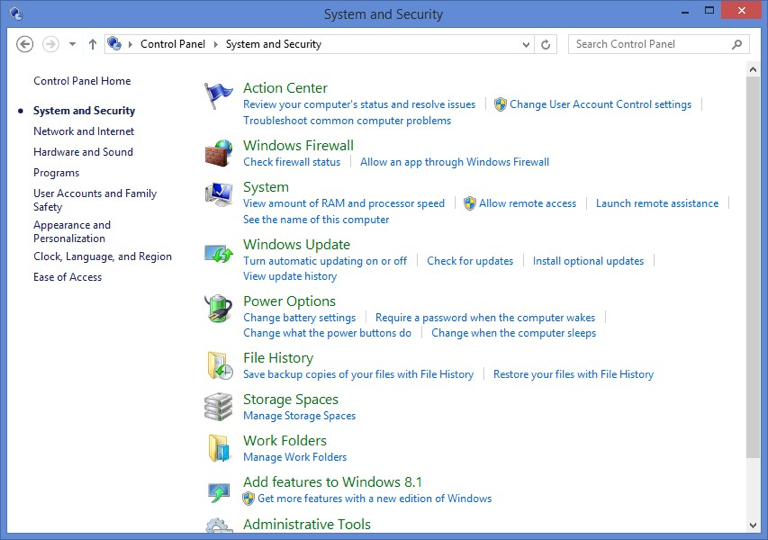
**1.2B Alternative partitioning**

* The instructions in Section 1.2A above are for installing Ubuntu on a single contiguous partition. This not always optimal; for example, some computers feature solid-state drives (SSDs) which give faster access to data than traditional hard drives (HDDs), in combination with one or more HDDs. The idea is that the operating system is installed on the SSD for fast access, while the HDD is used for data and other programs. If you wish to make use of a setup like that, it is necessary to set up the partitions differently from the example given above.
* The following discusses an example of a setup featuring multiple partitions with one SSD and one HDD. This is how we have configured one machine in our lab. See also <https://itsfoss.com/install-ubuntu-dual-boot-mode-windows/> for additional details.
* To install Ubuntu in SSD (OS C:), we gave 300 GB for Ubuntu and 200GB for Windows. To save experimental data in HDD (DATA D:), we set 800 GB for Ubuntu and 200 for windows.
* For this installation, we separate the partition for Ubuntu itself, which goes onto the SSD, and another partition for all other programs, files and data, which can be stored on the slower HDD.
* During installation, you can specify what each of your multiple partitions are used for. The partition for the operating system itself is called ‘/’ and is pronounced ‘root’ - this goes on the SSD. For all other files, you use a ‘/home’ partition, which is also the folder where you start when opening a new terminal, or a new file explorer window; this partition is on the HDD. We will also make use of a ‘/swap’ partition on the SSD, although this is optional.
* The Windows Disk Management program will show you the SSD (which typically will contain Windows, as well as the Recovery Drive), and the HDD (which will typically just be the whole disk if you haven’t added partitions yet). You now have to decide how much space you will allocate to Windows, both for the OS and for whatever programs, data etc you will ever actually store on Windows. If you are installing a dual-boot machine which will in reality mostly be used for stimulus delivery and data collection in Ubuntu, then you probably don’t need to be too generous to Windows.
* In the example setup, we had a 4TB HDD and a 500GB SSD. We shrank the Windows partition on the SSD down to 200GB, and the remaining 300GB of space was left unallocated for installing the Ubuntu OS, (the ‘/’ partition). We also set aside 100G on the HDD for files in Windows, and the remaining 3.9TB was made unallocated, to be used for Ubuntu’s ‘/home’ partition.

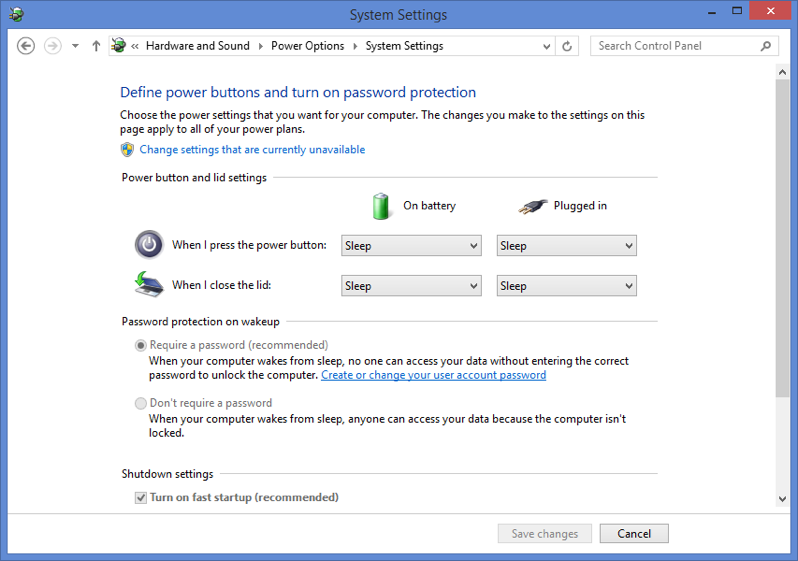
**1.3 Turn off fast boot**

To be able to boot to your Ubuntu USB drive, you need to turn off the fast boot option.

* Press the **Windows Key** and then click the magnifying glass in the top right corner. Click **Control Panel,** then **System and Security**



* Click **Power Options,** then click **Choose what the power buttons do.** Then you will have a screen with the heading **Define power buttons and turn on password protection:**



* If options are greyed out and you can’t change them, click **Change settings that are currently unavailable**
* Down the bottom you will find an option called **Turn on fast startup (recommended)**. Uncheck this option, then click **Save changes**.

**1.4A Turn off secure boot**

Turn off secure boot is recommended, however, there are slight differences among different models of computers.

Generally speaking

* Click **Windows Key** then click on the power icon.
* Hold down the **shift key** and select restart computer. Keep the shift key held down until your computer reboots.
* You will now have an option to go into your **UEFI boot settings**. Within the UEFI settings look for the option to turn off secure boot.
* **Save** and exit the UEFI settings

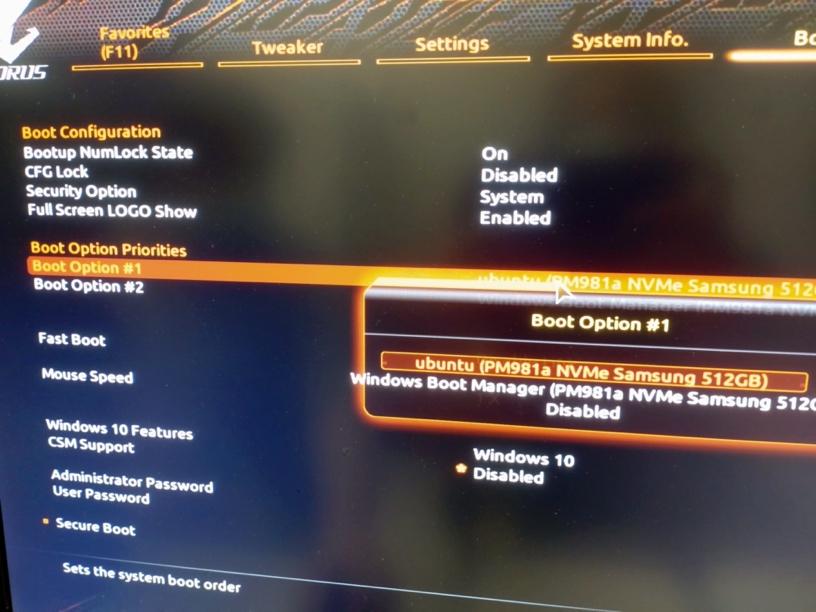
Alternatively, you can access UEFI settings as follows:

* Reboot, then press the appropriate key repeatedly when the computer starts until you get to Setup (the key is often F2, sometimes F12 or F6)

**1.4B Turn off secure boot using BIOS, and allocate RAM**

Turn off secure boot is recommended, however, there are slight differences among different models of computers.

Generally speaking

* Click **Windows Key** then click on the power icon.
* Spam the **delete key** until computer restarts
* In BIOS, go to Advanced Mode
* Then to Boot and select restart options
* While in BIOS, go to Advanced Mode
* Then to Tweaker
* Chose Extreme memory profile
* Chose Profile 1
* Save and Exit

Timeline

Description automatically generated

**1.5 Disable RAID (if installing to SSD)**

If you are installing Ubuntu partially or entirely onto a **solid-state drive (SSD)**, then you need to follow the steps below.

We did not do this step in the Jan 2023 install, and it worked fine.

If you have a Windows system that has the “SATA Operation” mode in UEFI settings set to “RAID”, then when you try to install Ubuntu it won’t be able to see the SSD. You need to change the setting in UEFI from RAID to “AHCI” in order to make the SSD visible to the Ubuntu installation process.

Unfortunately, you can’t just go and change it in UEFI – if you do, you will lose access to Windows! The issue is that the Windows boot loader doesn’t load the driver needed for AHCI if the operation was previously set to RAID.

See the discussions and solutions here <https://davidvielmetter.com/tricks/installing-ubuntu-dual-boot-on-a-dell-precision-which-already-runs-windows-10/>

And here <http://triplescomputers.com/blog/uncategorized/solution-switch-windows-10-from-raidide-to-ahci-operation/>.

Summarising these articles, the solution is to boot into Windows in safe mode with AHCI on. This will automatically load the AHCI driver. After that, you can set Windows back to normal mode. Here is what you need to do:

1. Search for “cmd” in Windows. When it brings up a list which includes “Command Prompt”, right-click and select “Run as administrator”.
2. Enter the following:  
     
   bcdedit /set {current} safeboot minimal  
     
   This command will start Windows in safe mode next time you restart.
3. Restart the computer, and get into UEFI settings (i.e. either by holding shift down while clicking on Restart and then navigating to the UEFI settings, or by pressing the appropriate key (e.g. F2) when the computer restarts.
4. Select System Configuration > SATA Operation.
5. If ‘Raid ON’ is selected, click on ‘AHCI’ instead.
6. Exit and continue with startup. The computer will now boot into safe mode, and load the AHCI driver behind the scenes.
7. Now you can switch off safe mode. Start up a command prompt as administrator, as above, and enter:  
     
   bcdedit /deletevalue {current} safeboot
8. Restart the computer.

**1.6 Install Ubuntu**

* First, you need to boot from a USB:
* Connect your bootable Ubuntu USB drive to your PC
* Boot to **Advanced startup** options within Windows 10
* Alternatively, spam F12 while computer is booting
* After the computer restarts, click **Use a device,** then click **USB drive.** The computer will now restart and boot from USB.
* However, that’s in an ideal world. Probably, that didn’t work, and you need to go back into your UEFI settings, and explicitly set up booting from USB. Do that as follows:
  + Go into UEFI settings as before.
  + Go to Change Boot Sequence.
  + If you already have an option listed on the right to boot from USB, move that one up to the top of the boot sequence (see below). Then save and exit UEFI settings to restart.
  + Otherwise, create a boot option by clicking on Add Boot Option.
  + Under ‘Boot option name’, give it a name like ‘Ubuntu USB’ so you can find it again
  + Under ‘File System List’, select an option that mentions the word ‘USB’ somewhere.
  + Under ‘Filename’, select the right file to start from when booting, which for 16.04 is EFI/BOOT/grubx64.efi.
  + Save the option by clicking on ‘OK’.
  + Now move the option to the top in the boot sequence displayed at the top right. Click on the USB option to select it, then click on Move Up until it moves to the top of the list.
  + Save and exit UEFI settings, and reboot. If all goes well, you will now boot from USB!
* Then you have the option of Try Ubuntu and Install Ubuntu, select **Install Ubuntu**.
* Chose **Normal installation**
* Now you need to decide what language to use before installing (please choose English!). NOTE: please make sure you have a wired internet connection before you start to install Ubuntu. Do not install the system without internet connection.
* Third-party software is optional, uncheck it, then click **Continue**.
* Now you are at a screen called **Installation Type**. Choose **Something else** then click **Next.** NOTE: This is very important, DO NOT choose erase disk and install Ubuntu, as you will then also erase you Windows system!
* If you are installing Ubuntu without Windows, chose the **Erase disk and install Ubuntu option**
* **Graphical user interface, text, application

  Description automatically generated**
* **Be careful with the next few steps!!! Check section 1.10 before proceeding.**
* Now you can see how your hard drives are split into partitions. The largest free space will be the one for you to install Ubuntu.

Text

Description automatically generated with medium confidence

* The crucial (non-obvious) step in creating partitions is to click on the ‘+’ icon under the list of partitions and free space, to create a new partition!
* Choose it, click **Install Now**. Note: Please only choose **free space** to install your Ubuntu.
* If you are using several hard drives, ensure you choose the correct one. During Ubuntu installation, we created three separate partitions. There will be a fairly large number of partitions and unallocated areas shown. As a rule of thumb, HDD partitions are usually called ‘/dev/sdaN’ by Ubuntu, where N is the number of the partition, while SSD partitions often get called something like ‘/dev/nvme0n1’ or similar.
  + First, we created the (optional) swap partition. A recommended size is twice the size of your RAM. We ended up allocating 85GB of the 300GB on the SSD to /swap. Select the free space, then use the ‘+’ icon to create a new partition. Set the size of the new partition by typing it into the box at the top. Under ‘Use as’, select ‘swap area’. Click on ‘OK’.
  + Then use the remaining space on the SSD for root (i.e. Ubuntu itself). Create a new partition, use the rest of the space, and under ‘Use as’, select ‘ext4’ and for mount point select ‘/’. This partition will be used for the operating system.
  + Finally, select the partition for the /home partition as all the unallocated HDD space (should see it under /dev/sda\*). Under ‘Use as’, select ‘ext4’, and mount point is ‘/home’.Text, whiteboard

    Description automatically generated
* The installation will begin, during the process, you will be asked a few configuration type questions. First, choose your location, then keyboard layout and create a default user. For example, we configured our system as follows:

Location: Adelaide, Australia

Keyboard layout: English UK (please check your own keyboard)

Default user: hoverfly

It’s now time to sit tight and wait for it to be finished, but don’t be too happy now as this is probably not the end of installing your Ubuntu alongside Windows.

**1.7 Boot repair**

After the installation is finished, choose restart, when the restart begins, unplug your USB, then your computer probably boots straight into Windows without giving you an option for Ubuntu.

*If everything works, skip this section.*

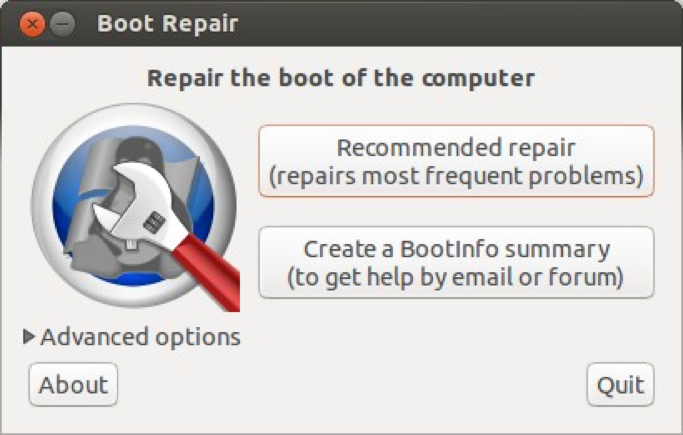
* Refer to step 1.5, now you need to connect your bootable USB and boot your computer from USB again.
* This time, choose **Try Ubuntu** instead of Install Ubuntu.
* After you are in the live session of Ubuntu, open **dash** (refer to figure 1) and type in **terminal**
* Within the terminal window, enter the following commands one by one:

sudo add-apt-repository ppa:yannubuntu/boot-repair

sudo apt-get update

sudo apt-get install –y boot-repair && boot-repair

Then the boot repair screen will appear.



* Click **Recommended repair.** Follow the information that appears on the screen and finish the boot repair.

**1.8 Boot loader**

After finishing the boot repair, restart your computer, when it restarts, unplug your USB drive.

If today is not your day, you will still boot straight into Windows.

*If everything works, skip this section.*

* Log into Windows, right click on the bottom left corner. You will see an option for **administrator’s command prompt**. Click this option.
* Within the administrator’s command prompt, type the following command:

bcdedit / set “{bootmqr}” path \EFI\ubuntu\grubx64.efi

* Exit the command prompt

**1.9 Reboot**

Reboot your computer, now you will have 4 options

1. Ubuntu
2. Ubuntu (advanced)
3. Windows
4. Setup

Choose 1, or just wait, your computer will boot into Ubuntu 16.04.

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**Step 2: Install Nvidia Geforce GTX 970 driver**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

NOTE: From this step on, we are starting to use the Ubuntu system. It is very important to keep in mind that when you log in to Ubuntu, you’re only a user who IS NOT allowed to change system files. However, when installing software, the system files need be changed. Therefore, operating as a so-called **superuser** enables you to change system files, to do so, we need to use **sudo/gksudo** in the terminal window.

Usually, the Ubuntu system does not automatically install the Nvidia Geforce graphic card driver. There are three methods below that you can use.

\*\*\* METHOD 1: Ask Ubuntu to find the driver

* Check your system settings, let your Ubuntu find an appropriate driver for you. Open **System setting**, click **Software & Updates**



* Click **Additional Drivers**, it will take a few seconds to find the drivers.

Graphical user interface, text

Description automatically generated

* Choose a Nvidia Driver if there is one.

\*\*\* If Ubuntu cannot find an appropriate driver for you, you can install the driver manually.   
METHOD 2

* Open the **dash** and search for **terminal** (refer to the first figure), type the following commands one by one in Terminal window:

sudo add-apt-repository ppa:graphics-drivers/ppa

sudo apt-get update

sudo apt-get install nvidia-525 (see how to find version below!!)

sudo reboot

If locked:

Sudo dpkg--configure-a

The install command above gives a specific version(430) of the Nvidia driver. Note that by the time you read this, it’s almost certain that there will be a newer version of the driver! Instead of “430” in the above command, install the newest stable version available from the repository.   
  
To find the latest stable version:  
1. Consult the repository web page – it usually says the name of the latest stable / long-term release near the top of the page.  
2. When you add the repository according to the first command above, you will see a long section of text rushing past. If you scroll up to the top of the text, it is (at the time of writing) the same as the text on the web page, so you can find the latest stable version here.

NOTE:

Previous versions of these notes had the repository ppa:xorg-edgers/ppa – that one may still work as well, but the above repository is apparently “official”.

Previous notes also suggested finding your driver on <http://www.geforce.com/drivers>.

\*\*\* The final option is to download the driver directly from Nvidia and install manually.  
METHOD 3:

* Download the appropriate driver for your graphics card from <http://www.geforce.com/drivers>.
* Save the driver in an appropriate folder that you can easily access using the Terminal. For instance you can leave it in Downloads if your browser downloaded it to there.
* Open the Terminal. Go to the folder where the downloaded files exist, e.g. by entering  
    
   cd Downloads  
    
  To see the name of the driver, list all files in the folder by entering   
    
   ls  
    
  Then enter the following lines  
    
   sudo chmod +x NVIDIA-xxxxx.run  
    
  where the “xxxxx” part is the name of the driver.   
  (Tip: you can just type the first part of the name, then click on the Tab key to auto-complete it, instead of typing the whole thing. You do need to type in enough of the name so that there is no other file with the same name.)  
  The ‘xxxxx’ part will vary for the specific driver you downloaded. Now enter  
    
   sudo /Downloads/NVIDIA-xxxx.run --no-x-check   
    
  This will install the driver! (you need to change the path in front of the name of the driver if you saved it somewhere other than “Downloads”).  
  See also <https://medium.com/better-programming/how-to-install-nvidia-drivers-and-cuda-10-0-for-rtx-2080-ti-gpu-on-ubuntu-16-04-18-04-ce32e4edf1c0> for more detail.

After rebooting your computer, go to **System setting** then **Software & Updates** and **Additional Drivers,** make sure you’re using the Nvidia Graphic card driver.

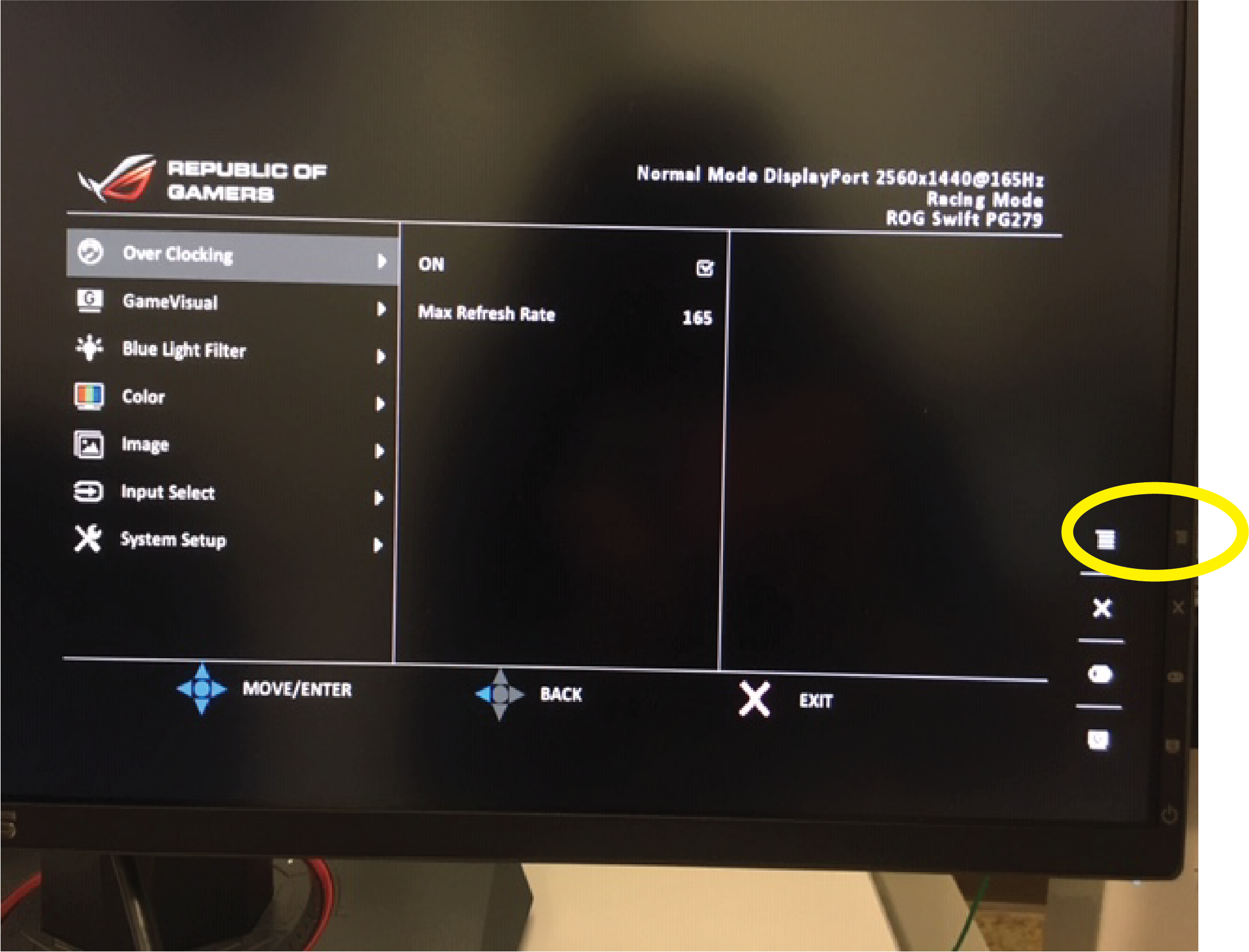
**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**Step 3: Set dual display and change the refresh rate**

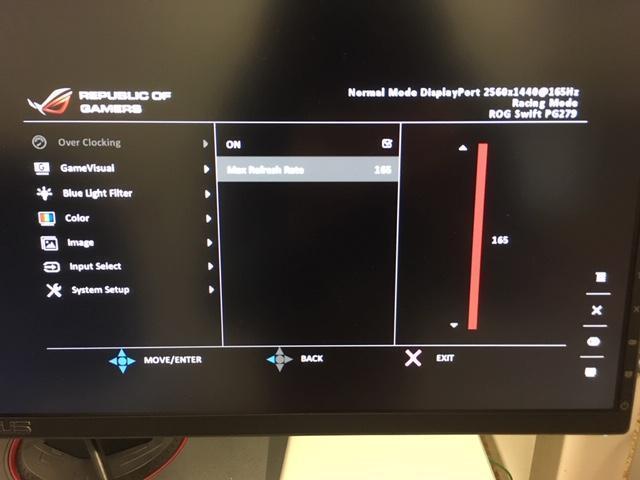
**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**3.1 Set monitor resolution to 165 Hz**

Connect monitors to Nvidia graphics card (lower ports) via display cables. You need to manually set the refresh rate to 165 Hz on each monitor displaying stimuli. There are buttons hidden behind on the right-hand bottom corner of the screen (yellow circle). The main menu is called up by pressing the top joystick. Select **Over Clocking** by pressing the joystick.



Use the same joystick to set the refresh rate to 165 Hz. By pressing the centre of joystick, you can save the change.

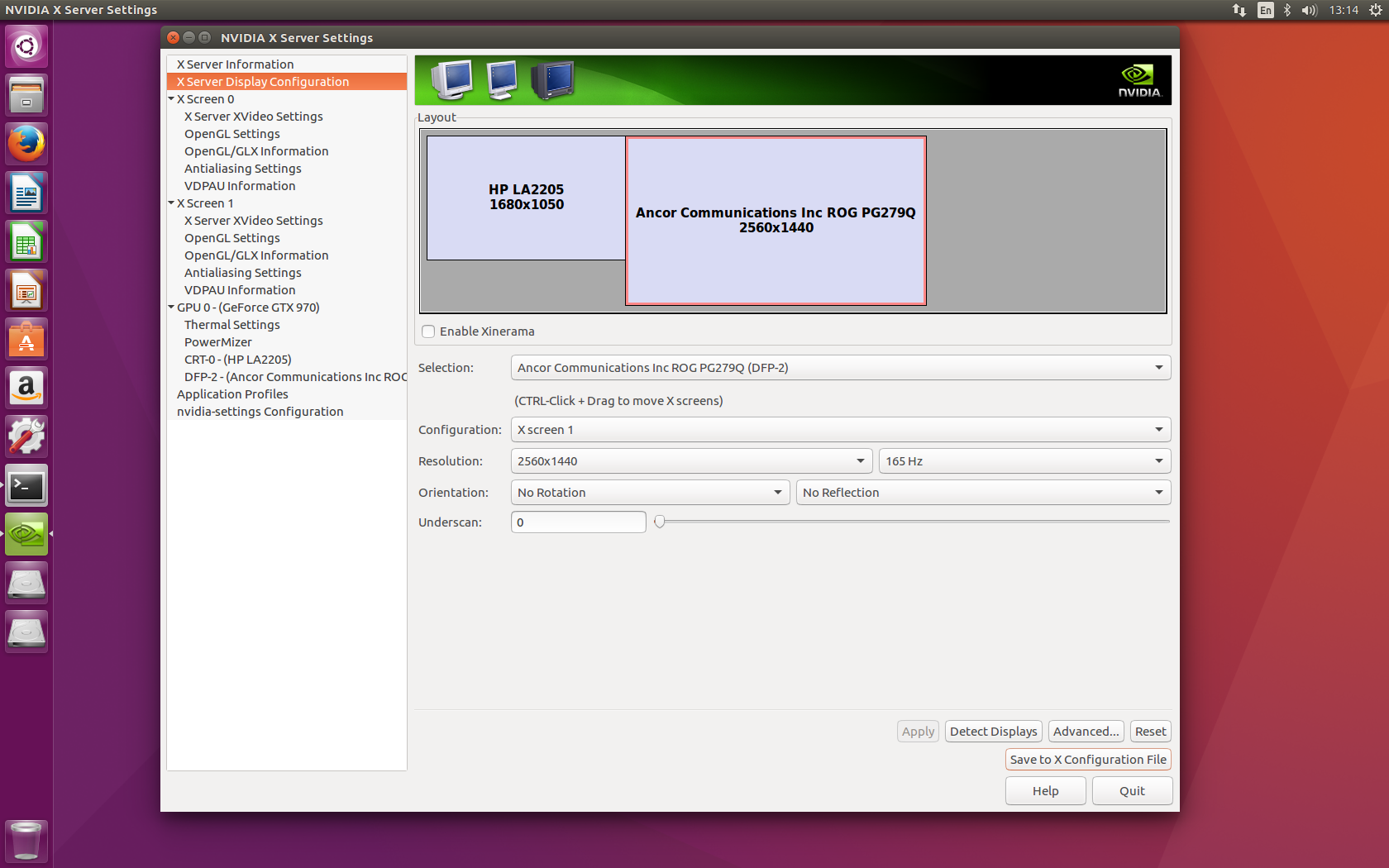


**3.2 NVIDIA settings for one stimulus monitor**

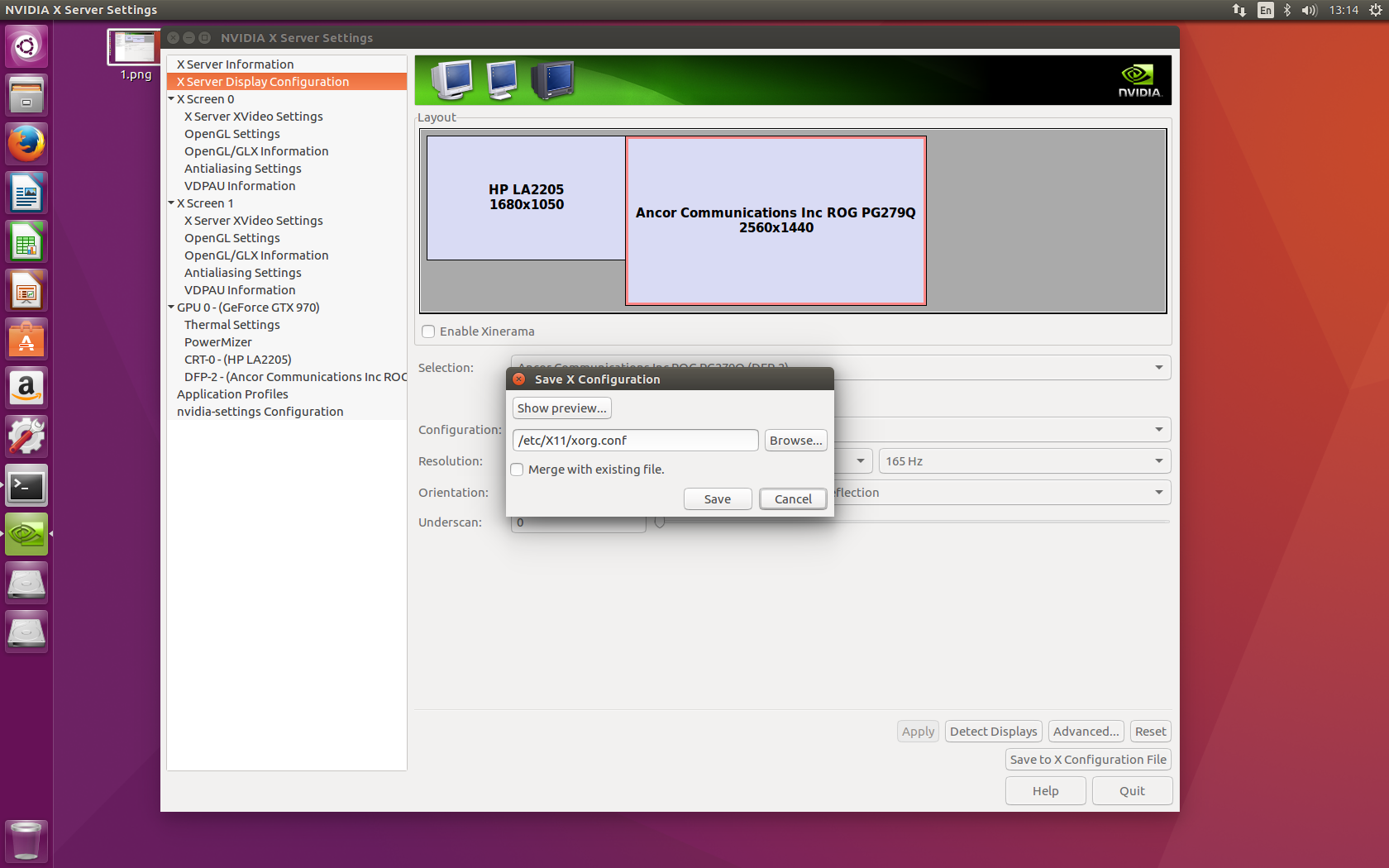
* Start the NVIDI app or use the **Terminal** window, to type:

sudo nvidia-settings

* In the Nvidia Settings window that will launch, choose **X Server Display Configuration**
* Select your monitor for displaying stimuli (we call it the second screen) and
* choose **Separate X screen (requires X restart)** for **Configuration.**
* choose your monitor size and your desired refresh rate for **Resolution**.
* For example, for Asus PG279, we choose 25601440, 165 Hz.



* Then click **Save to X Configuration File**
* In the pop-out window, uncheck **Merge with existing file** and click **Save**



* Then click **Apply** in the Nvidia Setting window, then **Quit**.

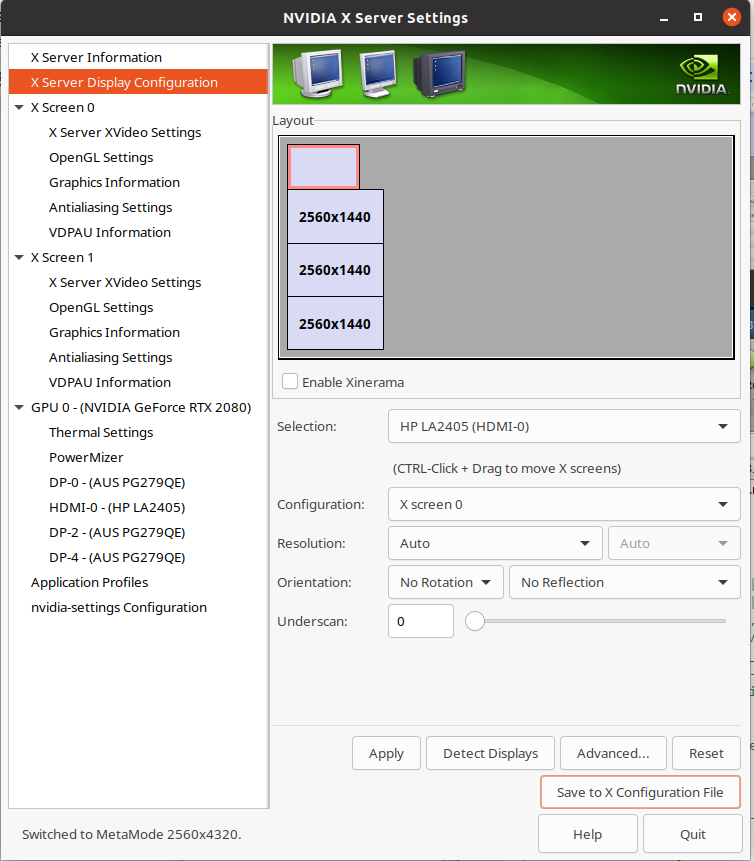
NOTE:

* If the second screen goes black, don’t panic, it might still work! Following the next steps, we will test this screen using flyfly.
* Restart your computer after finishing the settings.
* If you run into issues, try this site:[https://linuxconfig.org/how-to-install-the-nvidia-drivers-on-ubuntu-21-04](https://protect-au.mimecast.com/s/0_mYC2xMGPikR494cnR1BB?domain=linuxconfig.org)

**3.2 NVIDIA settings for three stimulus monitors**

Follow the instructions above, but note the order from top to bottom: DP-2, DP-4, DP-0.  
Resolution: 2560 x 1440 @ 165hz

Please note that all 3 stimulus screens need to be X Screen 1.



**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**Step 4: Install Matlab**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**4.1 Download Matlab installer and preparation**

* Matlab 2016a has been tested and used extensively and is compatible with FlyFly. Note that this version will not necessarily work with new graphics cards.
* Note that 2018b and later versions are NOT compatible with FlyFly at the time of writing (Jan 2023), but 2016a is. We have not tested any versions in between these yet.
* Download the matlab version you need from (matlab for Linux) <http://au.mathworks.com/downloads/>
* Unzip the Matlab installer (right click on the folder, choose **Extract here**)

**4.2 Install Matlab**

* In the Terminal, go to the downloaded directory
* Run the installer as a super user

sudo ./install

* Follow the instructions of the installer and install in /usr/local/MATLAB/R2015b (default path)
* Unclick ALL toolboxes
* To add an entry to the launcher and some other features, install the matlab-support package:

sudo apt-get install matlab-support

* During the installation you have to confirm the installation path of matlab and you have the option to restrict the use of matlab to certain users. In addition, you could rename the GCC libraries, do not rename them.

**4.3 Install Update**

After matlab is installed, you will likely need to download an update.

* Download the update from Matlab’s homepage.
* Go to the Downloads folder in Terminal

sudo sh R2016a\_Update\_xxxxx.sh

**4.3 Download and Install Runtime**

After matlab is installed, you will likely need to download Runtime.

* Download the Runtime and its update from Matlab’s homepage (64 bit version, Linux)

Text

Description automatically generated

* Go to the Downloads folder and chose **Extract here**
* Go to the extracted folder in Terminal

sudo ./install

* Go to the Downloads folder in Terminal

sudo sh filenamexxxxx.sh

* Go back to Terminal

sudo apt-get install libncurses5

**4.3 Launch Matlab**

You can now launch Matlab by typing the following commands in the terminal window:

cd /usr/local/MATLAB/R2016a/bin

sudo ./matlab

or

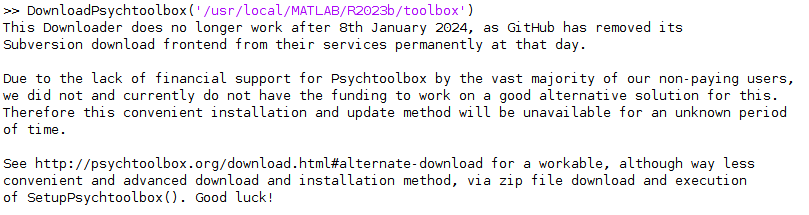
sudo matlab

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**Step 5: Install the Psychophysics toolbox and flyfly**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**NOTE: THE OLD METHOD OF INSTALLING PSYCHTOOLBOX NO LONGER WORKS AFTER JAN 8TH 2024, AS SEEN IN THE SCREENSHOT BELOW. WHILST THIS METHOD IS NO LONGER AVAILABLE FOR INSTALLING THE TOOLBOX, THE OLD INSTRUCTIONS HAVE BEEN KEPT AS THEY WERE FOR THE SAKE OF DOCUMENTATION. THESE OLD INSTRUCTIONS CAN BE FOUND UNDER APPENDIX A AT THE BOTTOM OF THE DOCUMENT.**



**5.1 Downloading Psychtoolbox-3**

To install PsychToolbox, you first need to first connect to the NeuroDebian repository, which houses the most up-to-date and stable version of PsychToolbox. To do this, navigate to <https://neuro.debian.net/> and select both your operating system (If you are unsure what this is, open up a terminal and enter ‘**lsb\_release -a**’ to find your Ubuntu release) and a server to download from (Recommended to choose either Chinese server). Click on ‘all software’ when the option appears and you should be provided with two terminal prompts to copy & paste into your terminal. Do this, and you will be connected to the NeuroDebian repository.

Once this is done, run the commands ‘**sudo apt-get update**’ and then ‘**sudo apt-get install matlab-psychtoolbox-3**’ to install the toolbox.

**5.2 Installing Psychtoolbox-3**

To install Psychtoolbox-3 to your MATLAB installation, first run the command ‘**ptb3-matlab**’ in a Linux terminal. This will open up a MATLAB instance with Psychtoolbox-3 active. To permanently add the toolbox to your MATLAB installation, first write ‘**savepath**’ into the MATLAB command window. This will save the toolbox and ensure Psychtoolbox is active whenever you open MATLAB next. Then, run ‘**PsychLinuxConfiguration**’ in the command window to begin the configuration process to set up all the files needed to run the toolbox with Flyfly. Answer yes to any user questions that pop up until the configuration script is complete, at which point Psychtoolbox-3 is setup for use with Flyfly. To verify that you have done this correctly, restart MATLAB and run ‘**PsychtoolboxVersion**’ in the command window. The installation was done correctly if a version greater than 3.08 is displayed in response.

**NOTE: When using Flyfly with Psychtoolbox, you MUST run MATLAB as a regular user, *not as sudo*. Running MATLAB as sudo causes crashes when displaying stimui.**

@(hObject,eventdata)tableGui('uitable1\_KeyPressFcn',hObject,eventdata,guidata(hObject))

**5.3 Install and launch flyfly**

* Download the latest flyfly from <http://hoverflyvision.weebly.com/software.html>
* It is also available here: GoogleDrive/Motion Vision Lab/Software\_and\_mauals/FlyFly/new version
* Chose FlyFly3.2.rotated.screen.video for the 3-screen version
* Unzip it by right click and choose **Extract here**
* Launch Matlab (Following Step 4.3), and go to the directory where you unzipped flyfly
* Type flyfly in the command window in Matlab

Now, you can test if flyfly works! If it works, ignore the following step. For instructions on how to use Flyfly, download the manual here <http://hoverflyvision.weebly.com/software.html>. If the screen went black under step 3, you should test it now by pressing ‘Init Screen’ in Flyfly. If you’re lucky, the screen will initialize and your stimuli are ready to be run.

If it’s not your day again, we have ways to fix it.

**5.5 Install missing files**

Probably, you get an Invalid MEX file error regarding libdc1934.so.22 when you launch flyfly. What you need to do is to install this library file on your system.

* Type the following commands one by one in the terminal window:

sudo apt-get update

sudo apt-get install libdc1394-utils

You may also need to run the following commands

sudo apt-get install libdc1394-dev

sudo apt-get install libdc1394-22-dev

Now you can start flyfly, however, you will probably get another MEX file error when you try to initialize screen.

* Type the following commands one by one in the terminal window:

sudo apt-get update

sudo apt-get install build-essential

sudo apt-get install freeglut3-dev

If you get an error about libstdc++ and GLIBCXX\_3.4.20 run the following commands

sudo add-apt-repository ppa:ubuntu-toolchain-r/test

sudo apt-get update

sudo apt-get install gcc-4.9 (may not execute, if so ignore)

sudo apt-get upgrade libstdc++6

once the commands have been run you will also likely need to find the libstdc++.so.6 file, located in use/local/MATLAB/R2016a/sys/os/glnxa64/

open a terminal in this folder and rename the file using the following command

sudo mv libstdc++.so.6 libstdc++.so.6\_old

Matlab should now be able to run FlyFly

However, if you encounter more invalid MEX file error, install the missing files or find detailed trouble shooting instructions from <http://psychtoolbox.org/ubuntu/>.

Now flyfly should work! For instructions on how to use Flyfly, download the manual here <http://hoverflyvision.weebly.com/software.html>

If you are using the tethered flight version, you will need to set your **save path**:

SavedData/YYYYMMDD/parameters

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**Step 6: Install cameras (for tethered flight set-up)**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

Make sure the camera is connected. Instructions here:

* To install the software, in Terminal:

sudo apt-get install guvcview

* With FlyFly open, Press Init screen.
* This should start guvcview
* go through the camera settings:

Video codec: Raw camera input

Untick increment filename

Audio controls: No sound

YUYV-YUYV 4:2:2

320 X 240

100 fps

* You will need to set your **save path**:

SavedData/YYYYMMDD/XXX

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

**Appendix A: Old Psychtoolbox Setup instructions**

**\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\*\***

Note that PsychToolbox’s own instructions for installing on Linux now deviate from these. Their preferred method of installing is using the NeuroDebian project’s distribution of the PsychToolbox packages.   
It appears that the following will still work, though, so this is still recommended. But if you encounter any difficulty, it may be worth consulting the following pages for the new method:

<http://psychtoolbox.org/download>

And the “additional tips” on

<http://psychtoolbox.org/linux>

First, you should make sure you don’t already have Psychtoolbox-3. Type the following in the Matlab command window:

>> PsychtoolboxVersion

**5.1 Install subversion**

Before you’re able to install the Psychophysics toolbox, you need to install Subversion on your computer. To do so, open terminal window, and type:

sudo apt-get install subversion

**5.2 Download DownloadPsychtoolbox.m script**

Download the DownloadPsychtoolbox.m script from <http://psychtoolbox.org/download/>, and save it in the folder where Matlab is installed.

**5.3 Install Psychtoolbox**

* Launch Matlab, and type the following line in the command window:

DownloadPsychtoolbox('/usr/local/MATLAB/R2016a/toolbox')

NOTE: this should be the folder where all your Matlab toolboxes are installed

* During the installation, update whatever it asks you to update. NOTE that it doesn’t terminate installation even when something very bad happens, so READ ALL the messages! In particular, you may need to install GStreamer, and glut (See “Install missing files” below.)
* Restart Matlab after the installation finished
* Check the version by typing

>> PsychtoolboxVersion

* If this is not correct, go to Psychtoolbox webpage and download desired source code
* Open Terminal in the Matlab folder and type

sudo nautilus

* This will give you some type of superuser control, so you can navigate to the Download folder and merge the folder with the source code into the Psychtoolbox folder already installed in the Matlab folder
* Restart matlab and check the version by typing

>> PsychtoolboxVersion