EEL 4930/5934: <u>Autonomous Robots</u> Configuring Jetson Nano 2GB <u>Credit</u>: prepared by Ailani Morales (<u>RoboPI Lab</u>)

Stuck on Boot Screen with Nvidia Logo Fix

- 1. Follow the steps in the video linked below approximately until 6:00 https://www.youtube.com/watch?v=XTjh8Km7coM&t=334s
- 2. A window like the one below may appear because SDK Manager does not support Ubuntu 20.04, but it does support 18.04.



To gain access to the Target Operating System options:

- Close SDK Manager
- Open the terminal
- Enter the command "sudo nano /usr/lib/os-release"
- Copy the text in this file and save it elsewhere. For example, I saved a text file (os_release.txt) in the Downloads folder to have a copy of the original os information.
- Enter the command "sudo nano /usr/lib/os-release"
- Change the current text in this file with the text in the window below provided by this link: <u>https://blog.thewatertower.org/2020/01/07/stash-of-etc-os-release-files/</u>

• Ubuntu 18.04 (symlink to /usr/lib/os-release)

NAME="Ubuntu" VERSION="18.04.3 LTS (Bionic Beaver)" ID=ubuntu ID_LIKE=debian PRETTY_NAME="Ubuntu 18.04.3 LTS" VERSION_ID="18.04" HOME_URL="https://www.ubuntu.com/" SUPPORT_URL="https://help.ubuntu.com/" BUG_REPORT_URL="https://help.ubuntu.com/" BUG_REPORT_URL="https://bugs.launchpad.net/ubuntu/" PRIVACY_POLICY_URL="https://www.ubuntu.com/legal/terms-and-policies/privacy-policy VERSION_CODENAME=bionic UBUNTU_CODENAME=bionic

- Save the file with its original name (os-release) and exit
- Enter the command "sdkmanager" and the Target Operating System options should be available now

For the options in the window below:

- I left the Host Machine unchecked
- I chose the Jetson Nano [2GB Developer Kit Version] for the Target Hardware
- I chose JetPack 4.6.2 for the Target Operating System
- I left DeepStream selected

STEP 01 DEVELOPMENT ENVIRONMENT	PRODUCT CATEGORY	Jetson	e
STEP 02	HARDWARE CONFIGURATION	Host Machine	Target Hardware Jetson Nano modules ✓ Jetson Nano ● (refresh)
	TARGET OPERATING SYSTEM	Linux JetPack 4.6.2 What's New	Ø.
	ADDITIONAL SDKS	DeepStream DeepStream 6.0.1	
Repair / Uninstall			CONTINUE >

I left the Jetson OS and Jetson SDK Components selected as well as the default directories in the figure below.

SDK Manage	er 1.8.3.10426			
	STEP 01	JETPACK 4.6.2 LINUX FOR JETSON NANO MODULES		Expand all
	DEVELOPMENT	 ✓ TARGET COMPONENTS ✓ ✓ Jetson 0S 	DOWNLOAD SIZE	STATUS
	STED 02	Jetson OS image Flash Jetson OS	1,714 MB 0 MB	
	DETAILS AND LICENSE	 ✓ Jetson SDK Components > CUDA 	1,027 MB	
		CUDA-X AI Computer Vision	1,126 MB 165.3 MB	
		NVIDIA Container Runtime Multimedia	1.0 MB 71.7 MB	
		> Developer Tools	64.2 MB	
		L	isk space during setup.	
		Download folder: /home/rocket/Downloads/nvidia/sdkm_downl Target HW image folder: /home/rocket/nvidia/nvidia_sdk	loads <u>change</u> (5GB required) <u>change</u> (15GB required)	
		I accept the terms and conditions of the <u>license agreement</u>	ts. Download now. Install later.	< BACK TO STEP 01

Before clicking Continue, make sure the SD card is in the Jetson Nano board.

- 1. Continue following along to the steps in the video starting at 8:10 https://www.youtube.com/watch?v=XTjh8Km7coM&t=334s
- 2. Once you reach 11:15 in the video, if the result from the command in the figure below shows a size similar to the SD card (for example, in the 50s GB range for a 64 GB SD card), then the Jetson Nano is done being set up.

😑 🗇 🕤 rocket@i	ubuntu:	~				
ocket@ubuntu:-	Sdf -	h				
rilesystem /dev/mmcblk0p1 hone tmpfs tmpfs tmpfs tmpfs tmpfs tmpfs tmpfs tmpfs	Size 14G 1.74 2.0G 2.0G 5.0M 2.0G 396M 396M \$	Used 5.8G 9 40K 37M 4.0K 9 12K 128K	Avail 8.1G 1.7G 2.0G 1.9G 5.0M 2.0G 396M	Use% 39% 0% 1% 2% 1% 0% 1%	Mounted on / /dev /dev/shm /run /run/lock /sys/fs/cgroup /run/user/120 /run/user/1000	

- 3. Return the os-release file to its original contents:
 - Copy the text from the text file you saved with the original /usr/lib/os-release contents (os_release.txt in the Downloads folder, for example)
 - Open the terminal
 - Enter the command "sudo nano /usr/lib/os-release"
 - Delete the current text in os-release and paste the text with the Ubuntu 20.04 os information
 - Save the file with its original name (os-release) and exit