**INSTALLATION OF INTEL®DISTRUBTION OF OPEN VINO TM TOOLKIT FOR LINUX.**

Make sure that you have an internet for the steps to be followed in the directory given below. Beginning with the introduction to Intel® distribution of Open VINO tm toolkit makes the applications easier to be used which are of often surpassed by human vision. As per CNN the toolkit enhances the workloads of computer vision across hardware Intel® that further maximizes the efficiency.

The toolkit enables convolutional neural networks based learning to be enhanced. It enhances the speed of time-to-market by an easy-to-use computer library vision functions and kernels that are pre-optimized.

INCLUSIONS WITH THE INSTALLATIONS:

1. Model optimizer: improvement, conversion and optimization of models trained in various popular frameworks for getting a format that can be used bv intel tools.
2. Inference engine: Includes the engine running the deep model learning. It has an inclusion of libraries that enable east inference integration to the applications of interest.
3. Drivers and runtime OpenCL TM version2.1: to enable open CL on GPU/CPU
4. Open C V: compilation for intel hardware.
5. Interference engine code samples: console applications enabling the user demonstration of Open VINO capabilities.
6. Demo applications: application template for guiding implementation of deep learning scenario.
7. Documentation for models pre-trained: pre-trained model documentation.

SYSTEM REQUIREMENTS

1.HARDWARE: 6th to 10th gen Intel ® core tm processors and intel ® xeon® processors; cooper lake; skylake and cascade lake; sandy bridge, Broadwell; Intel® neural compute stick 2.

2.OPERATING SYSTEM: Ubunto 18.04\* LTS(64-bit); Cent OS 7.4(64-bit); Yocto project v3.0 (64-bit)

3. PROCESSOR NOTES: a chipset supporting graphics

INSTALLATION OF INTEL® open VINO tm TOOLKIT COMPONENTS

* Download Intel ® distribution toolkit package (<https://software.intel.com/en-us/openvino-toolkit/choose-download> ) .
* Open the dropdown menu and select the toolkit.
* Command prompt terminal window is to be opened. Directory is to be changed from where downloaded. If downloaded to current user’s download directory, then: cd ~/Downloads/
* Unpacking of .tgz file is to be done now: tar -xvzf l\_ openvino\_toolkit\_ p \_ <version>.tgz
* Open the l\_openvino\_toolkit\_p\_<version> directory. If having any previous versions either rename or delete them.

INSTALLATION

* Run the scripts relative to installation option as root.
* For GUI installation: sudo ./install\_GUI.sh; for CUI installation : sudo ./install.sh
* Go as per the instructions and in the end a prompt with all the core being installed would appear.

INSTALL THE EXTERNAL DEPENDENCIES OF SOFTWARE

* Required for optimized build of CV library.
* Deep learning interference engine and a deeper sense of model optimization tools.
* The change is to be made through install\_dependencies directory.
* Script should be run for downloading and further installation of external software dependencies.

SETTING THE ENVIRONMENTAL VARIABLE

* Environmental variable is to be compiled before running the openVINO tm application.
* For temporary set of environmental variables, run: source/opt/intel/openvino/bin/setupvars.sh

CONFIGURATION OF MODEL OPTIMIZER

A key component of Intel VINO toolkit. The description of the entire model (intermediate representation) is given by:

* .xml : that tells about topology of the network.
* .bin : bias binary data

TO RUN THE VERIFICATION SCRIPTS FOR VERIFYING INSTALLATIONS

* The inference engine demo directly is to be used: cd /opt/intel/openvino/deployment\_tools/demo
* Image classification-verification script to be run: ./demo\_squeezenet\_download\_convert\_run.sh. It convers the model to .xml and .bin. verification system builds “image classification sample”whoch runs on the located directory.
* Running the inference pipeline verification script: the script would download the pre-trained model IR , that builds the “xyz” application and the runs it with downloaded model and the image from the located directory is to show an inference pipeline. Verification script initializes the recognition with specific attributes.
* Now close the image and verification script is complete.