2.Network configuration and Jtop

Jtop installation

(1) Installing JTOP to check CPU usage

```
sudo apt-get update
sudo apt-get full-upgrade
sudo apt install curl
sudo apt install nano
curl https://bootstrap.pypa.io/get-pip.py -o get-pip.py
sudo python3 get-pip.py
sudo pip3 install jetson-stats
jtop
```

Check the installed system components

(1) The OS image of Jetson nano already comes with JetPack, cuda, cudnn, opencv, and other installed examples. The installation path for these examples is as follows

```
TensorRT /usr/src/tensorrt/samples/
CUDA /usr/local/cuda-10.2/samples/
cuDNN /usr/src/cudnn_samples_v8/
VisionWorks /usr/share/visionworks/sources/samples/
/usr/share/visionworks-tracking/sources/samples/
/usr/share/visionworks-sfm/sources/samples/

OpenCV /usr/share/opencv4/samples/
```

(2) Check that CUDA version 10.2 is already installed in CUDA Jetson nano, but if you run nvcc - V at this time, it will not succeed. You need to write the path of CUDA to the environment variable. The Vim tool comes with the OS, so run the following command to edit the environment variables.

Check if there is nvcc in the bin directory of cuda:

```
ls /usr/local/cuda/bin
```

If present,

```
sudo vim ~/.bashrc  Enter the configuration file; Add the following two lines on
the last side:
```

Note: In vim, use Esc to return to command mode, and switch to the input module through I to enter text in input mode

```
export PATH=/usr/local/cuda/bin:$PATH

export LD_LIBRARY_PATH=/usr/local/cuda/lib64:$LD_LIBRARY_PATH
```

```
    alias ls='ls --color=auto'
    #alias dir='dir --color=auto'
    #alias vdir='vdir --color=auto'

    alias grep='grep --color=auto'
    alias fgrep='fgrep --color=auto'
    alias egrep='egrep --color=auto'
fi

# colored GCC warnings and errors
#export GCC_COLORS='error=01;31:warning=01;35:note=01;36:caret=01;32:locus=0

# some more ls aliases
alias ll='ls -alF'
alias la='ls -A'
alias l='ls -CF'

# Add an "alert" alias for long running commands.  Use like so:
#   sleep 10; alert
alias alert='notify-send --urgency=low -i "$([ $? = 0 ] && echo terminal ||

# Alias definitions.
# You may want to put all your additions into a separate file like
# ~/.bash_aliases, instead of adding them here directly.
# See /usr/share/doc/bash-doc/examples in the bash-doc package.

if [ -f ~/.bash_aliases ]; then
    . ~/.bash_aliases
fi

# enable programmable completion features (you don't need to enable
# this, if it's already enabled in /etc/bash.bashrc and /etc/profile
# sources /etc/bash.bashrc).
if ! shopt -oq posix; then
  if [ -f /usr/share/bash-completion/bash_completion ]; then
    . /usr/share/bash-completion/bash_completion
  elif [ -f /etc/bash_completion ]; then
    . /etc/bash_completion
  fi
fi
export PATH=/usr/local/cuda/bin:$PATH
export LD_LIBRARY_PATH=/usr/local/cuda/lib64:$LD_LIBRARY_PATH
```

Note: After exiting the command mode through Esc, press: to start inputting commands, wq to save and exit, q to exit, q! For forced exit
Save to exit.

```
source ~/.bashrc
```

```
beckhans@Jetson:~$ nvcc -V
```

```
nano@nano-desktop:~$ nvcc -V
nvcc: NVIDIA (R) Cuda compiler driver
Copyright (c) 2005-2019 NVIDIA Corporation
Built on Wed_Oct_23_21:14:42_PDT_2019
Cuda compilation tools, release 10.2, V10.2.89
```

(3) Check OpenCV
OpenCV4.1.1 version is already installed in Jetson nano. You can use the command to check if OpenCV is installed properly

pkg-config opencv4 --modversion

If OpenCv is installed properly, the version number will be displayed, and my version is 4.4.1

```
nano@nano-desktop:~$ pkg-config opencv4 --modversion
4.1.1
```

(4) Check cuDNN

CuDNN has been installed in Jetson nano and there are examples available for operation. Let's run the examples to verify the CUDA above.

Enter jtop at the terminal, press the right arrow key on the keyboard to select * * 7info * *, and you can see the version of cuDNN, as shown in the following figure.