

deepstream 6.4<onboarding

- dGPU 설정 요구사항

참고: [Quickstart Guide — DeepStream documentation 6.4 documentation](#)

위 메뉴얼에서 요구하는 시스템 사양

Ubuntu 22.04

NVIDIA driver 535.104.12

GStreamer 1.20.3

CUDA 12.2 TensorRT 8.6.1.6

- NVIDIA 드라이버 설치

참고: NVIDIA 드라이버는 535.104.12 이사이면 상관 없을 듯 함.

```
1 sudo apt install -y nvidia-driver-535
```

또는

```
1 research/onboarding/0.install-nvidia-deiver.sh
```

* 참고: <https://github.com/AlcheraInc/core-deepstream> Github 계정 연결

```
● lwl@lwl-cuda:~/work/core-deepstream/research/onboarding$ cat ./0.install-nvidia-deiver.sh
#!/bin/bash
SCRIPT=$(realpath -s $0)
SCRIPTPATH=$(dirname ${SCRIPT})
SCRIPTNAME=$(basename ${SCRIPT})
cd ${SCRIPTPATH}
set -euo pipefail

sudo apt install -y nvidia-driver-535

sudo reboot now
○ lwl@lwl-cuda:~/work/core-deepstream/research/onboarding$
```



```
● lwl@lwl-cuda:~/work/WF-image-provider/v1.5$ nvidia-smi
Tue Jan 30 11:27:20 2024
+-----+
| NVIDIA-SMI 535.154.05      Driver Version: 535.154.05    CUDA Version: 12.2 |
+-----+
| GPU  Name        Persistence-M | Bus-Id     Disp.A  | Volatile Uncorr. ECC | | |
| Fan  Temp     Perf Pwr:Usage/Cap | Memory-Usage | GPU-Util  Compute M. |
|          |          |             |              | MIG M.               |
+-----+
| 0  NVIDIA GeForce RTX 3090     Off  00000000:01:00.0 Off   N/A |
| 0%   34C    P8     21W / 350W |   10MiB / 24576MiB |   0%      Default N/A |
+-----+
+-----+
| Processes:                               GPU Memory |
| GPU  GI CI PID  Type  Process name        Usage  |
| ID  ID   |                               |          |
+-----+
| 0    N/A N/A 2282   G   /usr/lib/xorg/Xorg  4MiB |
+-----+
```

- 도커 설치

```
1 research/onboarding/1.install-docker.sh
```

```
● lwl@lwl-cuda:~/work/core-deepstream/research/onboarding$ cat ./1.install-docker.sh
#!/bin/bash
SCRIPT=$(realpath -s $0)
SCRIPTPATH=$(dirname ${SCRIPT})
SCRIPTNAME=$(basename ${SCRIPT})
cd ${SCRIPTPATH}
set -euo pipefail

sudo apt-get update
sudo apt-get install -y \
    ca-certificates \
    curl \
    gnupg \
    lsb-release

curl -fsSL https://download.docker.com/linux/ubuntu/gpg \
    | sudo gpg --dearmor -o /usr/share/keyrings/docker-archive-keyring.gpg
echo \
    "deb [arch=$(dpkg --print-architecture) signed-by=/usr/share/keyrings/docker-archive-keyring.gpg] https://download.docker.com/linux/ubuntu \
    $(lsb_release -cs) stable" | sudo tee /etc/apt/sources.list.d/docker.list > /dev/null

sudo apt-get update
sudo apt-get install -y \
    docker-ce \
    docker-ce-cli \
    containerd.io

sudo docker version
sudo systemctl enable docker

sudo usermod -aG docker $USER

sudo reboot now
● lwl@lwl-cuda:~/work/core-deepstream/research/onboarding$ docker --version
Docker version 25.0.1, build 29cf629
○ lwl@lwl-cuda:~/work/core-deepstream/research/onboarding$
```

- NVIDIA 컨테이너 룰킷 설치

1 research/onboarding/2.install-nvidia-container-toolkit.sh

```
● lvl@lvl-cuda:~/work/core-deepstream/research/onboarding$ cat ./2.install-nvidia-container-toolkit.sh
#!/bin/bash
SCRIPT=`realpath -s $0`
SCRIPTPATH=`dirname ${SCRIPT}`
SCRIPTNAME=`basename ${SCRIPT}`
cd ${SCRIPTPATH}
set -euo pipefail

curl -fsSL https://nvidia.github.io/libnvidia-container/gpgkey | sudo gpg --dearmor -o /usr/share/keyrings/nvidia-container-keyring.gpg \
&& curl -s -L https://nvidia.github.io/libnvidia-container/stable/deb/nvidia-container-toolkit.list | \ 
    sed '$!d; s/deb https://deb.sury.org https://#g' | \ 
    sudo tee /etc/apt/sources.list.d/nvidia-container-toolkit.list

sudo sed -i -e '/experimental/ s/^#/' /etc/apt/sources.list.d/nvidia-container-toolkit.list

sudo apt-get update
sudo apt-get install -y nvidia-container-toolkit

sudo apt install -y nvidia-container-runtime
sudo apt install -y nvidia-docker2

sudo reboot now
● lvl@lvl-cuda:~/work/core-deepstream/research/onboarding$ 
```



```
● lvl@lvl-cuda:~/work/core-deepstream/research/onboarding$ docker run --rm --gpus all ubuntu:18.04 nvidia-smi
Unable to find image 'ubuntu:18.04' locally
18.04: Pulling from library/ubuntu
7c457f213c76: Pull complete
Digest: sha256:152dc042452c496607f07ca0127571cb9c29697f42acbfad72324b2bb2e43c98
Status: Downloaded newer image for ubuntu:18.04
Tue Jan 30 03:11:31 2024
+-----+
| NVIDIA-SMI 535.154.05      Driver Version: 535.154.05 CUDA Version: 12.2 |
| Persistence-M  Bus-Id     Disp.A  Volatile Uncorr. ECC |
| GPU Name      Perf      Pwr:Usage/Cap | Memory-Usage | GPU-Util Compute M. |
| Fan Temp     |          |          |          |          |          | MIG M. |
+-----+
| 0  NVIDIA GeForce RTX 3090      OFF | 00000000:01:00.0 OFF |   0%   | N/A | | |
| 0% 35C   P8      22W / 350W | 10MiB / 24576MiB |   0%   | Default |
|          |          |          |          |          | N/A |
+-----+
+-----+
| Processes:                               GPU Memory |
| GPU  GI CI PID Type Process name        Usage |
| ID   ID   |          |          |          |          |
+-----+
● lvl@lvl-cuda:~/work/core-deepstream/research/onboarding$ 
```

- deepstream 도커 이미지 생성

1 research/onboarding/3-1.build-deepstream.sh

```
● lvl@lvl-cuda:~/work/core-deepstream/research/onboarding$ cat ./3-1.build-deepstream.sh
#!/bin/bash
SCRIPT=`realpath -s $0`
SCRIPTPATH=`dirname ${SCRIPT}`
SCRIPTNAME=`basename ${SCRIPT}`
cd ${SCRIPTPATH}
set -euo pipefail

sudo docker build \
-f deepstream.dockerfile \
-t deepstream:6.4 \
.

exit $? 
```



```
● lvl@lvl-cuda:~/work/core-deepstream/research/onboarding$ cat ./deepstream.dockerfile
FROM mcr.microsoft.com/deepstream:6.4-gc-triton-devel

#####
# Install necessary packages for building GLib (According to release notes : https://github.com/GNOME/glib/blob/2.76.6/INSTALL.md)
RUN apt update \
RUN apt install -y pkg-config python3-dev libffi-dev libmount-dev meson ninja-build wget libxml2-utils

# Install GLIB 2.76.6
WORKDIR /opt/nvidia/deepstream/deepstream/lib
RUN git clone -b 2.76.6 https://gitlab.freedesktop.org/GNOME/glib.git ..glibc-2.76.6

WORKDIR /opt/nvidia/deepstream/deepstream/lib/glibc-2.76.6
RUN git submodule update --init --recursive

RUN meson_build
RUN ninja -C _build
RUN ninja -C _build install

ENV LD_LIBRARY_PATH=/usr/local/lib/x86_64-linux-gnu:$LD_LIBRARY_PATH

#####
# Install Gst-nvtracker
RUN wget 'https://api.ngc.nvidia.com/v2/models/nvidia/tao/reididentificationnet/versions/deployable_v1.0/files/reidnet50_market1501.ptl' -P /opt/nvidia/deepstream/deepstream/samples/models/tracker/
RUN wget 'https://vision.in.tum.de/datasets/gtae_reid.onnx' -P /opt/nvidia/deepstream/deepstream/samples/models/tracker/
ENV CUDA_VER=12
WORKDIR /opt/nvidia/deepstream/deepstream/sources/apps/sample_apps/deepstream 
```

참고: [Frequently Asked Questions — DeepStream documentation 6.4 documentation](#)

우분투 22.04에서 디풀트로 제공하는 GLIB 22.72 버전에 문제가 있어, 2.76으로 업그레이드 해야 함, 그 내용을 dockerfile에 적용.

- 모델 다운로드

1 research/onboarding/deepstream-6.4/samples/models/download-models.sh

참고:

[Gst-nvtracker — DeepStream documentation 6.4 documentation](#)

[Gst-nvtracker — DeepStream documentation 6.4 documentation](#)

- 테스트코드 실행을 위한 도커 실행

```
1 research/onboarding/4-1.run-deepstream.sh
```

```
● lwl@lwl-cuda:~/work/core-deepstream/research/onboarding$ cat ./4-0.run-deepstream.sh
#!/bin/bash
SCRIPT=$(realpath -s $0)
SCRIPTPATH=$(dirname ${SCRIPT})
SCRIPTNAME=$(basename ${SCRIPT})
cd ${SCRIPTPATH}
set -euo pipefail

export DISPLAY=:1
xhost +
#docker rm -f deepstream64-devel
sudo rm -rf /tmp/.X11-unix

docker run -it \
--rm \
--gpus all \
--net=host \
--name=deepstream64 \
-v /tmp/.X11-unix:/tmp/.X11-unix \
-v ./deepstream-6.4/samples/models/Tracker:/opt/nvidia/deepstream/deepstream-6.4/samples/models/Tracker \
-v ./deepstream-6.4/samples/models/peopleNet:/opt/nvidia/deepstream/deepstream-6.4/samples/models/peopleNet \
-e DISPLAY=$DISPLAY \
-e CUDA_CACHE_DISABLE=0 \
--device /dev/snd \
deepstream:6.4

exit $?

# cd /opt/nvidia/deepstream/deepstream-6.4/sources/apps/sample_apps/deepstream-test1
# make
# ./deepstream-test1-app dtest1_config.yml
lwl@lwl-cuda:~/work/core-deepstream/research/onboarding$
```

※ 위에서 다운로드 한 모델만 불름으로 잡혀 있다

- deepstream-test1

```
1 # cd /opt/nvidia/deepstream/deepstream-6.4/sources/apps/sample_apps/deepstream-test1
2 # make
3 # ./deepstream-test1-app dtest1_config.yml
```



deepstream-test1

- deepstream-test2

```
1 # cd /opt/nvidia/deepstream/deepstream-6.4/sources/apps/sample_apps/deepstream-test2
2 # make
3 # ./deepstream-test2-app dtest2_config.yml
```

deepstream-test2

- deepstream-test3

```
1 # cd /opt/nvidia/deepstream/deepstream-6.4/sources/apps/sample_apps/deepstream-test3
2 # make
3 # ./deepstream-test3-app dtest3_config.yml
```

deepstream-test3

- deepstream-test4

```
1 research/onboarding/3-2.build-deepstream-kafka.sh #카프카가 포함된 도커 빌드
2 research/onboarding/4-2.run-deepstream-kafka.sh #도커 실행
```

```
● lwl@lwl-cuda:~/work/core-deepstream/research/onboarding$ cat ./3-2.build-deepstream-kafka.sh
#!/bin/bash
SCRIPT=$(realpath -s $0)
SCRIPTPATH=$(dirname ${SCRIPT})
SCRIPTNAME=$(basename ${SCRIPT})
cd ${SCRIPTPATH}
set -euo pipefail

sudo docker build \
-f deepstream-kafka.dockerfile \
-t deepstream-kafka:6.4 \
.

exit $?
```

```
22  # Install Apache Kafka
23 RUN apt install -y openjdk-8-jdk
24 #export JAVA_HOME=$(readlink -f /usr/bin/java | sed "s:/bin/java::")
25 ENV JAVA_HOME=/usr/lib/jvm/java-8-openjdk-amd64/jre/
26
27 RUN mkdir /opt/apache/
28 RUN wget 'https://downloads.apache.org/kafka/3.6.1/kafka_2.13-3.6.1.tgz' -P /opt/apache/
29
30 WORKDIR /opt/apache/
31 RUN tar -zxf kafka_2.13-3.6.1.tgz
32 RUN ln -s ./kafka_2.13-3.6.1 ./kafka
33 #WORKDIR /opt/apache/kafka
34
35 ENV PATH=/opt/apache/kafka/bin:$PATH
36
37 # Install supervisord
38 RUN apt-get update && apt-get install -y supervisor
39 COPY supervisord-kafka.conf /etc/supervisor/conf.d/supervisord.conf
40 CMD [""/usr/bin/supervisord", "-c", "/etc/supervisor/conf.d/supervisord.conf"]
```

도커파일 내, 카프카 관련 인스를 추가(deepstream-kafka.dockerfile)

```
● lwl@lwl-cuda:~/work/core-deepstream/research/onboarding$ cat ./supervisord-kafka.conf
[supervisord]
nodaemon=true

[program:zookeeper]
command=/opt/apache/kafka/bin/zookeeper-server-start.sh /opt/apache/kafka/config/zookeeper.properties

[program:kafka]
command=/opt/apache/kafka/bin/kafka-server-start.sh /opt/apache/kafka/config/server.properties
```

토픽 추가(카프카)

```
1 # kafka-topics.sh --create --topic quickstart-events --bootstrap-server localhost:9092
```

```
1 # cd /opt/nvidia/deepstream/deepstream-6.4/sources/apps/sample_apps/deepstream-test4
2 # make
3 # apt install -y vim
4 # vim ./dstest4_config.yml
```

conn-str 및 topic 변경

```
msgbroker:
  proto-lib: /opt/nvidia/deepstream/deepstream/lib/libnvds_kafka_proto.so
  conn-str: <host>:<port>
  topic: <topic>
  sync: 0
```

||||||||||||||||||||||||||||||

```
msgbroker:
  proto-lib: /opt/nvidia/deepstream/deepstream/lib/libnvds_kafka_proto.so
  conn-str: localhost:9092
  topic: quickstart-events
  sync: 0
```

```
1 # ./deepstream-test4-app dstest4_config.yml
```

다른 터미널에서

```
1 docker exec -it deepstream-kafka /bin/bash
```

수신되는 메시지 확인(카프카)

```
1 kafka-console-consumer.sh --topic quickstart-events --from-beginning --bootstrap-server localhost:9092
```

```
lwl@lwl-cuda:~/work/core-deepstream/research/onboarding$ docker exec -it deepstream64-kafka /bin/bash
root@lwl-cuda:/opt/nvidia/deepstream/deepstream-6.4/sources/apps/sample_apps/deepstream-app# kafka-console-consumer.sh --topic quickstart-events --from-beginning --bootstrap-server localhost:9092
[2024-01-30 07:22:51,646] WARN [Consumer clientId=console-consumer, groupId=console-consumer-38974] Error while fetching metadata with correlationId 2 : {quickstart-events=LEADER_NOT_AVAILABLE} (org.apache.kafka.clients.NetworkClient)
```

```

Frame Number = 784 Vehicle Count = 20 Person Count = 4
Frame Number = 785 Vehicle Count = 20 Person Count = 4
Frame Number = 786 Vehicle Count = 20 Person Count = 4
Frame Number = 787 Vehicle Count = 20 Person Count = 4
Frame Number = 788 Vehicle Count = 20 Person Count = 4
Frame Number = 789 Vehicle Count = 20 Person Count = 5
Frame Number = 790 Vehicle Count = 20 Person Count = 4
Frame Number = 791 Vehicle Count = 20 Person Count = 6
Frame Number = 792 Vehicle Count = 20 Person Count = 5
Frame Number = 793 Vehicle Count = 20 Person Count = 4
Frame Number = 794 Vehicle Count = 20 Person Count = 5
Frame Number = 795 Vehicle Count = 20 Person Count = 5
Frame Number = 796 Vehicle Count = 20 Person Count = 4
Frame Number = 797 Vehicle Count = 20 Person Count = 5
Frame Number = 798 Vehicle Count = 20 Person Count = 6
Frame Number = 799 Vehicle Count = 20 Person Count = 5
Frame Number = 800 Vehicle Count = 19 Person Count = 5
Frame Number = 801 Vehicle Count = 20 Person Count = 5
Frame Number = 802 Vehicle Count = 20 Person Count = 4
Frame Number = 803 Vehicle Count = 20 Person Count = 4
Frame Number = 804 Vehicle Count = 20 Person Count = 5
Frame Number = 805 Vehicle Count = 20 Person Count = 5
Frame Number = 806 Vehicle Count = 20 Person Count = 5
Frame Number = 807 Vehicle Count = 20 Person Count = 7
Frame Number = 808 Vehicle Count = 20 Person Count = 5
Frame Number = 809 Vehicle Count = 20 Person Count = 6
Frame Number = 810 Vehicle Count = 20 Person Count = 5
Frame Number = 811 Vehicle Count = 20 Person Count = 6
Frame Number = 812 Vehicle Count = 20 Person Count = 5
[]

    "hair" : "black",
    "cap" : "none",
    "apparel" : "formal",
    "confidence" : 0.40882647037506104
},
"bbox" : {
    "topleftx" : 462,
    "toplefty" : 478,
    "bottomrightx" : 485,
    "bottomrighty" : 516
},
"location" : {
    "lat" : 0.0,
    "lon" : 0.0,
    "alt" : 0.0
},
"coordinate" : {
    "x" : 0.0,
    "y" : 0.0,
    "z" : 0.0
},
"pose" : { },
"event" : {
    "id" : "5fb40f3a-186f-4a71-bb0f-0faaad423893",
    "type" : "entry"
},
"videoPath" : ""
}

```

- deepstream-test5

```
1 research/onboarding/4-2.run-deepstream-kafka.sh #도커 실행
```

토픽 추가(카프카)

```
1 # kafka-topics.sh --create --topic quickstart-events --bootstrap-server localhost:9092
```

설정파일 변경

```
1 # cd /opt/nvidia/deepstream/deepstream-6.4/sources/apps/sample_apps/deepstream-test5
2 # make
3 # apt install -y vim
4 # vim configs/test5_config_file_src_infer.yml
```

```
sink1:
enable: 1
#type - 1=FakeSink 2=EglSink 3=File 4=UDPSink 5=nvdrmvideosink 6=MsgConvBroker
type: 6
msg-conv-config: dstest5_msgconv_sample_config.yml
#(0): PAYLOAD_DEEPSTREAM - Deepstream schema payload
#(1): PAYLOAD_DEEPSTREAM_MINIMAL - Deepstream schema payload minimal
#(256): PAYLOAD_RESERVED - Reserved type
#(257): PAYLOAD_CUSTOM - Custom schema payload
msg-conv-payload-type: 0
msg-broker-proto-lib: /opt/nvidia/deepstream/deepstream/lib/libnvds_kafka_proto.so
#Provide your msg-broker-conn-str here
msg-broker-conn-str: <host>;<port>;<topic>
topic: <topic>
#Optional:
#msg-broker-config: ../../deepstream-test4/cfg_kafka.txt
```

!!!!!!

```
sink1:
enable: 1
#type - 1=FakeSink 2=EglSink 3=File 4=UDPSink 5=nvdrmvideosink 6=MsgConvBroker
type: 6
msg-conv-config: dstest5_msgconv_sample_config.yml
#(0): PAYLOAD_DEEPSTREAM - Deepstream schema payload
#(1): PAYLOAD_DEEPSTREAM_MINIMAL - Deepstream schema payload minimal
#(256): PAYLOAD_RESERVED - Reserved type
#(257): PAYLOAD_CUSTOM - Custom schema payload
msg-conv-payload-type: 0
msg-broker-proto-lib: /opt/nvidia/deepstream/deepstream/lib/libnvds_kafka_proto.so
#Provide your msg-broker-conn-str here
msg-broker-conn-str: localhost:9092;quickstart-events
topic: quickstart-events
#Optional:
#msg-broker-config: ../../deepstream-test4/cfg_kafka.txt
```

```
1 ./deepstream-test5-app -c configs/test5_config_file_src_infer.yml -p 1
```

- deepstream-app

```
1 research/onboarding/4-1.run-deepstream.sh #도커 실행
```

설정파일 확인 및 변경

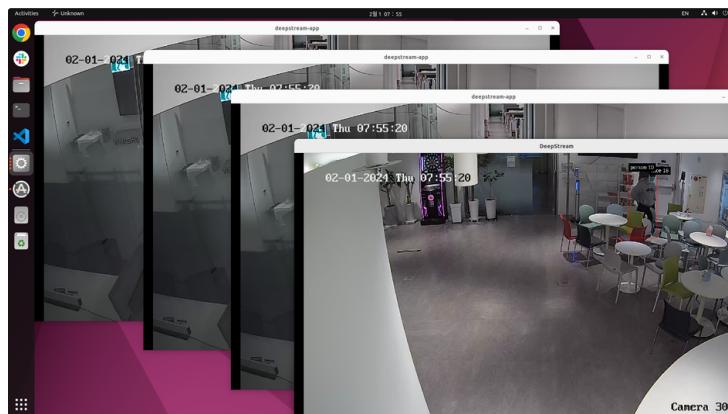
```

1 # cd /opt/nvidia/deepstream/deepstream-6.4/sources/apps/sample_apps/deepstream-app
2 # make
3 # apt install -y vim
4 # vim _configs/_config.yml

```

※ 코멘트를 확인하면서, 다양한 방법으로 설정 바꾸어 테스트 해 볼 수 있다

영상출력 분리



영상 저장



RE-ID Feature 출력

[Gst-nvtracker — DeepStream documentation 6.4 documentation](#)

```

root@lwl-cuda:/opt/nvidia/deepstream/deepstream-6.4/sources/apps/sample_apps/deepstream-app# cat ./reid-track-output/00_003.001547.txt
9 -0.011836 -0.032167 -0.018981 -0.026267 0.006473 0.004598 -0.018856 -0.032807 -0.033731 0.021238 -0.010798 0.027387 0.067880 0.067711
0.007526 0.011383 -0.033145 0.018625 -0.061704 -0.063162 -0.052036 0.005109 0.008930 -0.043861 0.009792 0.004794 0.046847 -0.035295 -0.0
38505 -0.012660 -0.034762 -0.04772 -0.010441 0.023726 -0.002444 0.059110 0.046705 -0.003510 0.018981 0.042049 -0.008437 0.015586 -0.00
77345 0.025260 0.005680 -0.025601 0.023441 -0.000152 -0.054523 0.143053 0.023642 -0.051543 0.043826 0.050383 0.015526 0.049465 0.02
0519 -0.045532 -0.039323 -0.009540 0.040769 0.111465 0.037783 0.052339 0.058890 0.097518 0.035117 0.012423 0.018447 0.073505
8.014395 -0.048304 -0.021940 0.048340 0.026196 -0.056596 -0.010876 -0.010823 -0.002859 -0.059359 -0.020108 0.069240 -0.023317 0.027840
0.006522 -0.033056 0.055697 -0.037392 0.009405 0.011552 0.028435 -0.041835 -0.064192 0.006780 0.03196 -0.013800 0.013347 0.010102 -0.118
575 0.019300 0.054085 0.012760 -0.037463 0.130447 -0.053636 0.089660 0.074994 0.005523 0.085092 -0.065252 -0.017345 0.035568 0.065714 0.
009668 -0.038424 -0.073789 0.002923 -0.054808 0.040208 -0.007398 0.018678 0.040271 0.028506 -0.027333 -0.027227 -0.052925 0.023388 -0.05
8577 -0.009783 0.076377 0.010893 0.025343 -0.079761 -0.002654 -0.008792 -0.048480 0.079121 -0.013898 -0.008282 0.005154 -0.001
744 0.087652 0.011674 -0.163878 0.016786 -0.005771 0.044688 -0.077628 -0.002615 -0.092859 0.051180 0.058612 -0.017825 -0.029839 0.063446
-0.024579 -0.000157 -0.059674 0.019531 0.010183 -0.027671 0.016484 -0.009321 -0.051361 0.062700 0.051530 -0.011961 -0.036130 0.018127 0.
026551 0.010192 -0.052285 0.008255 -0.039774 -0.040876 0.074358 0.027440 -0.074003 0.006251 0.029573 -0.025023 0.065342 -0.001112 0.059
216 0.006740 -0.004750 0.003213 -0.006149 0.019780 0.045567 0.054441 0.029016 -0.009060 -0.014991 0.065558 -0.070109 0.013569 -0.01391
5 0.004770 0.010136 -0.043316 -0.043316 0.043316 0.043316 0.043316 0.043316 0.043316 0.043316 0.043316 0.043316 0.043316 0.043316
.004073 0.0113528 -0.0533165 -0.021855 -0.020473 -0.043841 0.043719 -0.017123 0.037723 0.041378 -0.000002 -0.000741 0.058541 -0.026378
.056088 0.051077 0.083690 0.082630 0.021288 0.074216 0.001408 -0.004683 -0.028542 -0.032210 0.016510 -0.008538 0.047416 0.051668 0.0
43328 -0.002572 0.081669 0.040769 -0.022570 -0.036962 0.016084 -0.007726 -0.043364 -0.015284 0.126252 -0.013969 0.059963 0.028453 0.0241

```

Primary-GIE0|| PeopleNet 적용

[Gst-nvtracker — DeepStream documentation 6.4 documentation](#)



- deepstream-3d-action-recognition

```
1 # cd /opt/nvidia/deepstream/deepstream-6.4/sources/apps/sample_apps/deepstream-3d-action-recognition  
2 # make
```

설정파일 설정(config_infer_primary_3d_action.txt)

```
1 tlt-encoded-model=../../../../samples/models/actionrecognitionnet/resnet18_3d_rgb_hmdb5_32.tlt  
2 tlt-model-key=nvidia_tao  
3 model-engine-file=../../../../samples/models/actionrecognitionnet/resnet18_3d_rgb_hmdb5_32.tlt_b4_gpu0_fp16.engine
```

설정파일 설정(config_infer_primary_2d_action.txt)

```
1 tlt-encoded-model=../../../../samples/models/actionrecognitionnet/resnet18_2d_rgb_hmdb5_32.tlt  
2 tlt-model-key=nvidia_tao  
3 model-engine-file=../../../../samples/models/actionrecognitionnet/resnet18_2d_rgb_hmdb5_32.tlt_b4_gpu0_fp16.engine
```

실행

```
1 # ./deepstream-3d-action-recognition -c deepstream_action_recognition_config.txt
```

