Attitude detection

introduce

This project includes a 3D body posture application built using the Deepstream SDK.

Prerequisite

DeepStream SDK 6.2 has been installed and can be obtained from the following website http://developer.nvidia.com/deepstream-sdk Please follow the instructions in /opt/nvidia/deepstream/deepstream/sources/apps/sample_apps/deepstream-app/README Prerequisites for installing and building Deepstream SDK applications.

build

1. Clone Application to /opt/nvidia/deepstream/deepstream/sources/apps/sample_apps/And define the project homepage as export BODYPOSE3D_HOME=/deepstream-bodypose-3d.

```
echo "export BODYPOSE3D_HOME=/opt/nvidia/deepstream/deepstream/sources/apps/sample_apps/deepstream_reference_apps/deepstream-bodypose-3d" << ~/.bashrc
source ~/.bashrc
```

2. Install the NGC CLI downloader (there is no need to install the yahboom version, if you have built your own image, please refer to the following connection) https://ngc.nvidia.com/setup/installers/cli
   If the network is not good, you can find and transfer the file to orin nano from the attachment. The following models can also be used in this way

```
sudo mkdir -p $BODYPOSE3D_HOME/models
cd $BODYPOSE3D_HOME/models
# Download PeopleNet
ngc registry model download-version "nvidia/tao/peoplenet:deployable_quantized_v2.5"
# Download BodyPose3DNet
ngc registry model download-version "nvidia/tao/bodypose3dnet:deployable_accuracy_v1.0"
sudo apt-get install tree #The image of yahboom does not require this step
```

If the model cannot be downloaded, it can be transferred to the home directory through the computer, and then moved to the corresponding folder through the mv command
So far, the directory tree should look like this

```
tree $BODYPOSE3D_HOME -d
$BODYPOSE3D_HOME
├── configs
├── models
|   ├── bodypose3dnet_vdeployable_accuracy_v1.0
|   └── peoplenet_vdeployable_quantized_v2.5
├── sources
|   ├── deepstream-sdk
|   └── nvdsinfer_custom_impl_BodyPose3DNet
└── streams
```

3. Download and extract feature values 3.4.0 under the project。

```
cd $BODYPOSE3D_HOME
sudo wget https://gitlab.com/libeigen/eigen/-/archive/3.4.0/eigen-3.4.0.tar.gz
sudo tar xvzf eigen-3.4.0.tar.gz
sudo ln eigen-3.4.0 eigen -s
```

If the download is not possible, you can transfer it to the home directory through your computer and then move it to the corresponding folder using the mv command

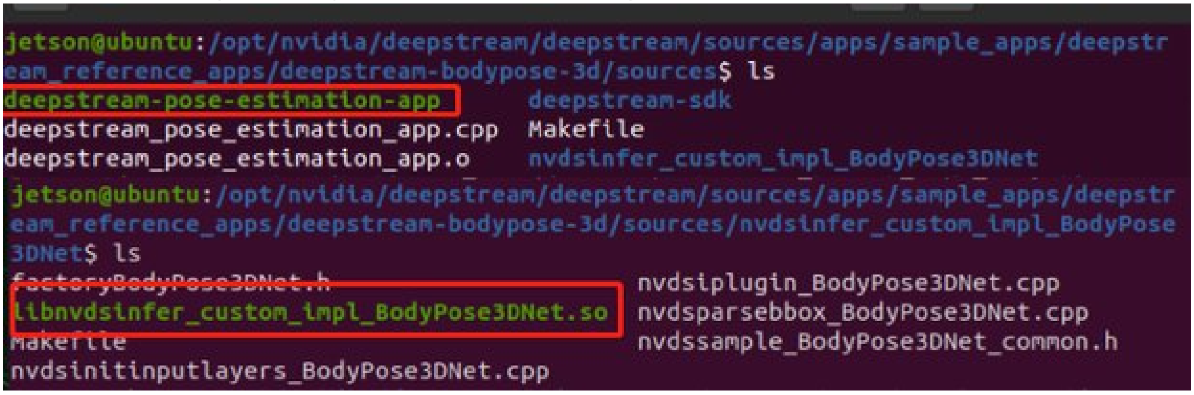
4. This step is not required for Deepstream SDK version 6.2 or later. The image of yahboom can be operated without any need

```
# Copy deepstream sources
cp $BODYPOSE3D_HOME/sources/deepstream-sdk/eventmsg_payload.cpp
/opt/nvidia/deepstream/deepstream/sources/libs/nvmsgconv/deepstream_schema
# Build new nvmsgconv library for custom Product metadata
cd /opt/nvidia/deepstream/deepstream/sources/libs/nvmsgconv
sudo CUDA_VER=11.4 make && make install
```

5.compile

```
# Build custom nvinfer parser of BodyPose3DNet
cd $BODYPOSE3D_HOME/sources/nvdsinfer_custom_impl_BodyPose3DNet
sudo CUDA_VER=11.4 make
# Build deepstream-pose-estimation-app
cd $BODYPOSE3D_HOME/sources
sudo CUDA_VER=11.4 make
```

After successful operation, you can see the following files in these two directories



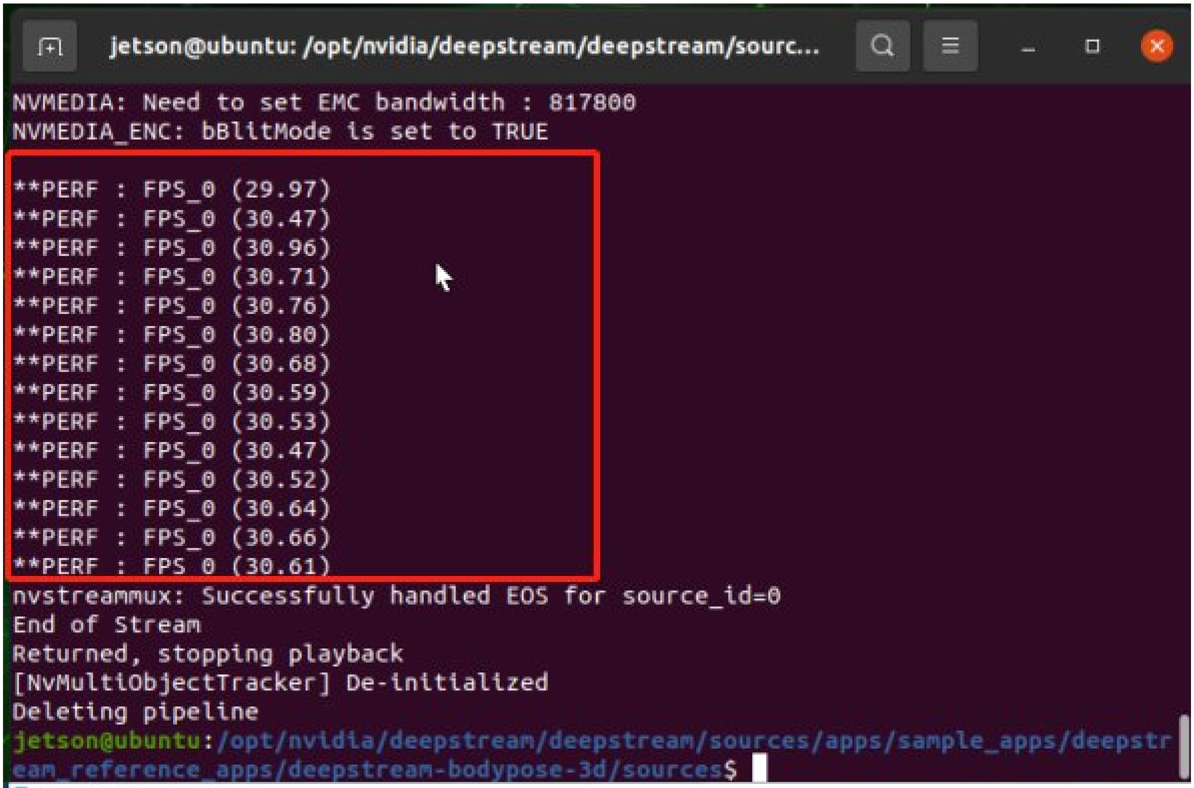
```
jetson@ubuntu:/opt/nvidia/deepstream/deepstream/sources/apps/sample_apps/deepstr
eam_reference_apps/deepstream-bodypose-3d/sources$ ls
deepstream-pose-estimation-app        deepstream-sdk
deepstream_pose_estimation_app.cpp    Makefile
deepstream_pose_estimation_app.o      nvdsinfer_custom_impl_BodyPose3DNet
jetson@ubuntu:/opt/nvidia/deepstream/deepstream/sources/apps/sample_apps/deepstr
eam_reference_apps/deepstream-bodypose-3d/sources/nvdsinfer_custom_impl_BodyPose
3DNet$ ls
factoryBodyPose3DNet.h                     nvdsiplugin_BodyPose3DNet.cpp
libnvdsinfer_custom_impl_BodyPose3DNet.so  nvdsparsebbox_BodyPose3DNet.cpp
Makefile                                   nvdssample_BodyPose3DNet_common.h
nvdsinitinputlayers_BodyPose3DNet.cpp
```

run a program

1. The following command processes the input video in URI format and presents superimposed pose estimation in the window.

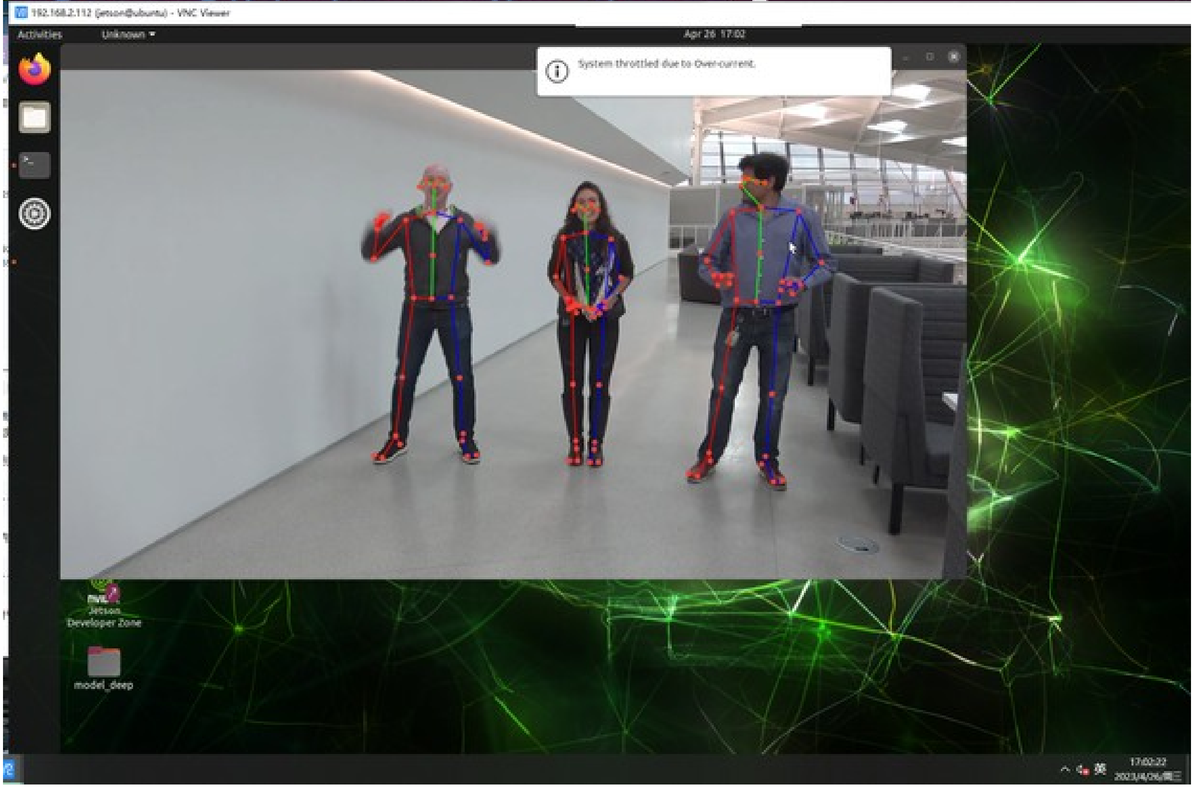
```
./deepstream-pose-estimation-app --input
file://$BODYPOSE3D_HOME/streams/bodypose.mp4
```

Note: Please provide the absolute path of the source video file.
You can see the results of video playback, with a frame rate of up to 30 frames



```
jetson@ubuntu: /opt/nvidia/deepstream/deepstream/sourc...
NVMEDIA: Need to set EMC bandwidth : 817800
NVMEDIA_ENC: bBlitMode is set to TRUE

**PERF : FPS_0 (29.97)
**PERF : FPS_0 (30.47)
**PERF : FPS_0 (30.96)
**PERF : FPS_0 (30.71)
**PERF : FPS_0 (30.76)
**PERF : FPS_0 (30.80)
**PERF : FPS_0 (30.68)
**PERF : FPS_0 (30.59)
**PERF : FPS_0 (30.53)
**PERF : FPS_0 (30.47)
**PERF : FPS_0 (30.52)
**PERF : FPS_0 (30.64)
**PERF : FPS_0 (30.66)
**PERF : FPS_0 (30.61)
nvstreammux: Successfully handled EOS for source_id=0
End of Stream
Returned, stopping playback
[NvMultiObjectTracker] De-initialized
Deleting pipeline
jetson@ubuntu:/opt/nvidia/deepstream/deepstream/sources/apps/sample_apps/deepstream_reference_apps/deepstream-bodypose-3d/sources$
```


192.168.2.112 (jetson@ubuntu) - VNC Viewer
Activities
Unknown
Apr 26 17:02
System throttled due to Over-current.
Jetson Developer Zone
model_deep
17:02:22

2.When the data source is a video file, the following command saves the output video with skeleton overlay to $BODYPOSE3D_HOME/streams/bodypose_3dbp.mp4 And save the key points of the skeleton to $BODYPOSE3D_HOME/streams/bodypose_3dbp.json

```
./deepstream-pose-estimation-app --input
file://$BODYPOSE3D_HOME/streams/bodypose.mp4 --output
$BODYPOSE3D_HOME/streams/bodypose_3dbp.mp4 --focal 800.0 --width 1280 --height
720 --fps --save-pose $BODYPOSE3D_HOME/streams/bodypose_3dbp.json
```

json analysis:

```
[{
  "num_frames_in_batch": 1,
  "batches": [{
    "batch_id": 0,
    "frame_num": 3,
    "ntp_timestamp": 1639431716322229000,
    "num_obj_meta": 6,
    "objects": [{
      "object_id": 3,
      "pose25d": [707.645203, 338.592499, -0.000448, 0.867188, ...],
      "pose3d": [297.649933, -94.196518, 3520.129883, 0.867188, ...]
    },{
    ...
    }]
  }]
}, {
```

- Pose25d includes a 34x4 float. A set of four represents the values of [x, y, zRel, conf] at a critical point. X and y are the positions of key points in the image coordinates; ZRel is the relative depth value of the root key point (i.e. pelvis) of the bone. x. The values of y, zRel are in millimeters. Conf is the confidence value of the prediction.
- Pose3d also includes a 34x4 float. A set of four values represents a key point of [x, y, z, conf]. x. Y, z are the 3D positions of the key points in world coordinates with the camera as the origin. x. The values of y and z are in millimeters. Conf is the confidence value of the prediction.

3. When the data source is an RTSP stream and the results are published to the RTSP stream://localhost:8554/ds-test

```
./deepstream-pose-estimation-app --input rtsp://<ipa_address>:<port>/<topic> --
output rtsp://
```

4. To publish the pose3D and pose25D metadata to the message broker, please do the following

```
./deepstream-pose-estimation-app --input
file://$BODYPOSE3D_HOME/streams/bodypose.mp4 --conn-str "localhost;9092;test"
```

"localhost;9092;test":It is the connection string of the message broker, localhost, port number 9092, and topic name test.
Please put double quotation marks around the connection string because of reserved characters in the shell.