داکیومنت مربوط به اجرای شبکه deepsort به وسیله tensorRT

مقدمه آشنایی با tensorRT:

هدف از کتابخانه tensorRT این است که فرایند استنتاج (inference) شبکه‌های عصبی را سرعت ببخشد. به این صورت که، شبکه‌ای که برای استنتاج آماده است را تا حد امکان ساده می‌کند به گونه‌ای سرعت شبکه افزایش و دقت آن کمترین مقدار ممکن کاهش یابد. برای این کار نیاز است که لایه‌های مختلف شبکه در tensorrt مشخص شود و سپس وزن‌های آموزش داده شده در tensorRT لود شود. که در انتها tensorrt با استفاده از ساختار شبکه و وزن‌های داده شده یک انجین می‌سازد که سرعت فرآیند استنتاج در آن تا حد ممکن افزایش یافته است.

نحوه اجرای کدهای مربوطه:

تمامی کدهای این بخش در پوشه به آدرس/home/synopsis/Yolov5_DeepSort_Pytorch/ موجود است که در ادامه با نام $home_dir به آن اشاره خواهد شد.

برای تغییر مشخصات مدل ساخته در tensorrt باید فایل $home_dir/yolov5_trt/yolov5.cpp را تغییر داد. برای ساده سازی تعدادی define در بالا قرار داده شده که تغییراتی که معمولا پر تکرار است را میتوان به سادگی تغییر داد. برای تغییرات عمیق‌تر میتوانید که خطهای ۱۰۱ تا ۱۱۰ را تغییر دهید یا حتی flag دلخواه دیگر اضافه کنید. tensorRT با توجه به این flag ها انجین نهایی را می‌سازد.

برای ساخت انجین میتوانید از دستورات مقابل استفاده کنید.

```
$ cd build
$ cmake ..
$ make
$ ./yolov5 -s yolo5l.wts yolov5l.engine l
```

در خط انتهایی نیز ورودی اول s- به معنی ساخت انجین است، ورودی دوم آدرس فایل wts. وزن‌ها است و ورودی سوم نام فایلی که انجین با آن ذخیره شود و ورودی چهارم هم نوع شبکه است که در این مثال نوع شبکه تبدیل شده l است. تنها نکته باقی مانده نحوه ساخت فایل wts. است که برای این کار نیز در آدرس $home_dir/yolov5_trt/ دستور مقابل را اجرا میکنید.

```
python gen_wts.py -w yolov5s.pt -o yolov5s.wts
```

که w- آدرس وزن‌های pt. ورودی است و o- نیز آدرس فایل خروجی است.

برای اجرای شبکه deepsort با مدل تبدیل شده به yolo کافی است که ابتدا در خط ۱۰۲ و ۱۰۴ آدرس انجین و پلاگین های ساخته شده توسط tensorRT را وارد کنید و سپس در $home_dir دستور مقابل را وارد کنید.

```
CUDA_VISIBLE_DEVICES=0 python3 track.py --source Input4.mp4 --img-size 1280 --save-vid
```

دستور بالا هم به این صورت است که ابتدا اطمینان حال کنید که CUDA_VISIBALE_DEVICES با آنچه در ابتدا برای ساخت انجین در دیفاین‌های فایل yolov5.cpp تعیین کردید یکسان باشد. چرا که انجین بهینه شده برای یک gpu خاص را نمی توان بر روی gpu دیگر اجرا کرد. بقیه ورودی‌های شبکه به مانند شبکه deep_sort عادی است.

یکی از نکات دیگری که باید مورد توجه قرار داد این است که درون کد برای اجرای انجین ساخته شده از کلاس Yolov5TRT موجود در فایل $home_dir/yolo5_trt/yolo5_trt.py استفاده شده است و مراحل پیش پردازش و پس پردازش در این فایل انجام میشود. و برای اعمال تغیرات مورد نیاز بر روی هر کدام از این مراحل به فایل گفته شده مراجعه کنید.