Use dockerfile to build your docker image

FROM → The FROM instruction initializes a new build stage and sets the base image for subsequent instructions

WORKDIR → The WORKDIR instruction sets the working directory for any RUN,

CMD, ENTRYPOINT, COPY and ADD instructions that follow it

COPY <src> <dest> → The COPY instruction copies new files or directories from <src> and adds them to the filesystem of the container at the path <dest>

RUN → The RUN instruction will execute any commands in a new layer on top of the current image and commit the results

CMD → The main purpose of a CMD is to provide defaults for an executing container

Note that COPY and RUN are executed when you build your docker image using 'docker build', and CMD is executed when you run your docker container using 'docker run'.

Your program in the docker container should do the following:
--read the files (.nrrd) in the folder **/input**
--apply your algorithm to them.
--save the results to **/predict**. Make sure the predicted results are consistent with the submitted results in the validation phase on Grand challenge.

Requirements
https://docs.docker.com/get-docker/

Details to build a docker image for your algorithm

Please name the docker image with your team name (i.e., **teamname.tar.gz**), and make sure the tag is latest (**Noted: the docker image name requires lowercase letters**). You can use the following command to generate the docker tar file.
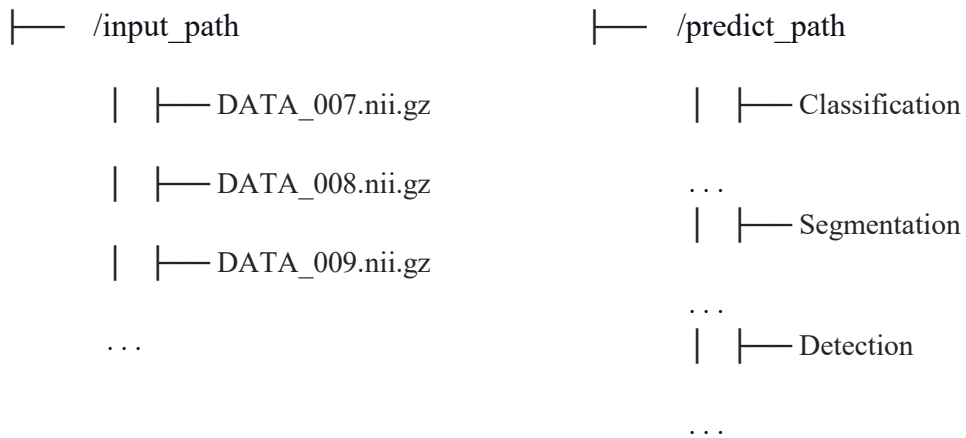
docker save teamname:latest -o teamname.tar.gz

Don't forget the **-o** option, otherwise, the docker image will not be loaded properly for saving and loading on different operation systems (Windows or Unix).

After we receive the docker tar file, we will run your program with commands as follows:

docker load < teamname.tar.gz
docker run --gpus "device=0" -name teamname -v /input_path:/input -v /predict_path/:/predict --shm-size 8g teamname:latest

Folder **/input_path** (the folder contains all files for testing) will be synchronized with **/input** in your docker container, and folder **/predict_path** (an empty folder used to save your predicted results) will be synchronized with **/predict** in your docker container.

```
├── /input_path                          ├── /predict_path

    │   ├── DATA_007.nii.gz                   │   ├── Classification

    │   ├── DATA_008.nii.gz                   . . .
                                              │   ├── Segmentation
    │   ├── DATA_009.nii.gz

    . . .                                     . . .
                                              │   ├── Detection

                                              . . .
```

The format of the submission result of each task should be the same as that of the validation phase (note: the results of segmentation tasks do not need to be compressed).

We highly recommend that you test the docker with the training data before sending it to us since downloading and uploading the zip file is time consuming for both the participants and the organizers.