# Final Project

ORB\_SLAM2-docker images

ORB\_SLAM (Localization and Simultaneous Mapping, SLAM) is an open source[[1]](#footnote-1) algorithm for automatic vehicle in an unknown environment. I am researching in SLAM algorithm for my double degree master in China. There is the complexity configuration the library environment for ORB\_SLAM especially in wins and also ubuntu system because of compatible problems of platform for ORB-SLAM. So, this is my objective to develop an image for simplifying the implement of ORB-SLAM.

My ORB-SLAM docker images are pushed in my GitHub and Docker hub[[2]](#footnote-2) where you can find Dockerfiles and images. In my hub, you can find two versions of ORB-SLAM image.

* V1.0[[3]](#footnote-3) is based on Ubuntu 16.04 and VNC for desktop connection

## Get image

### Pull the image from Docker hub:

docker pull qihaoliu/orbslam2-docker:v1.0

### Or use Dockerfile to build locally

git clone --branch v1.0 https://github.com/buaalqh/orbslam-docker.git

cd ./orbslam-docker

docker build -t qihaoliu/orbslam2-docker:v1.0 .

## Run the docker container

### Download Dataset

This ORB\_SLAM2 routine provides a demonstration program for the [KITTI odometry](http://www.cvlibs.net/datasets/kitti/eval_odometry.php) Dataset. Here, using sequence 04 in [data\_odometry\_gray](http://www.cvlibs.net/datasets/kitti/eval_odometry.php).

### Create container

Execute the following command to instantiate the container:

docker run -it --rm -v YOUR\_PATH\_TO\_KITTI/sequences/04:/root/Dataset/04 -p 5900:5900 qihaoliu/orbslam2-docker:v1.0

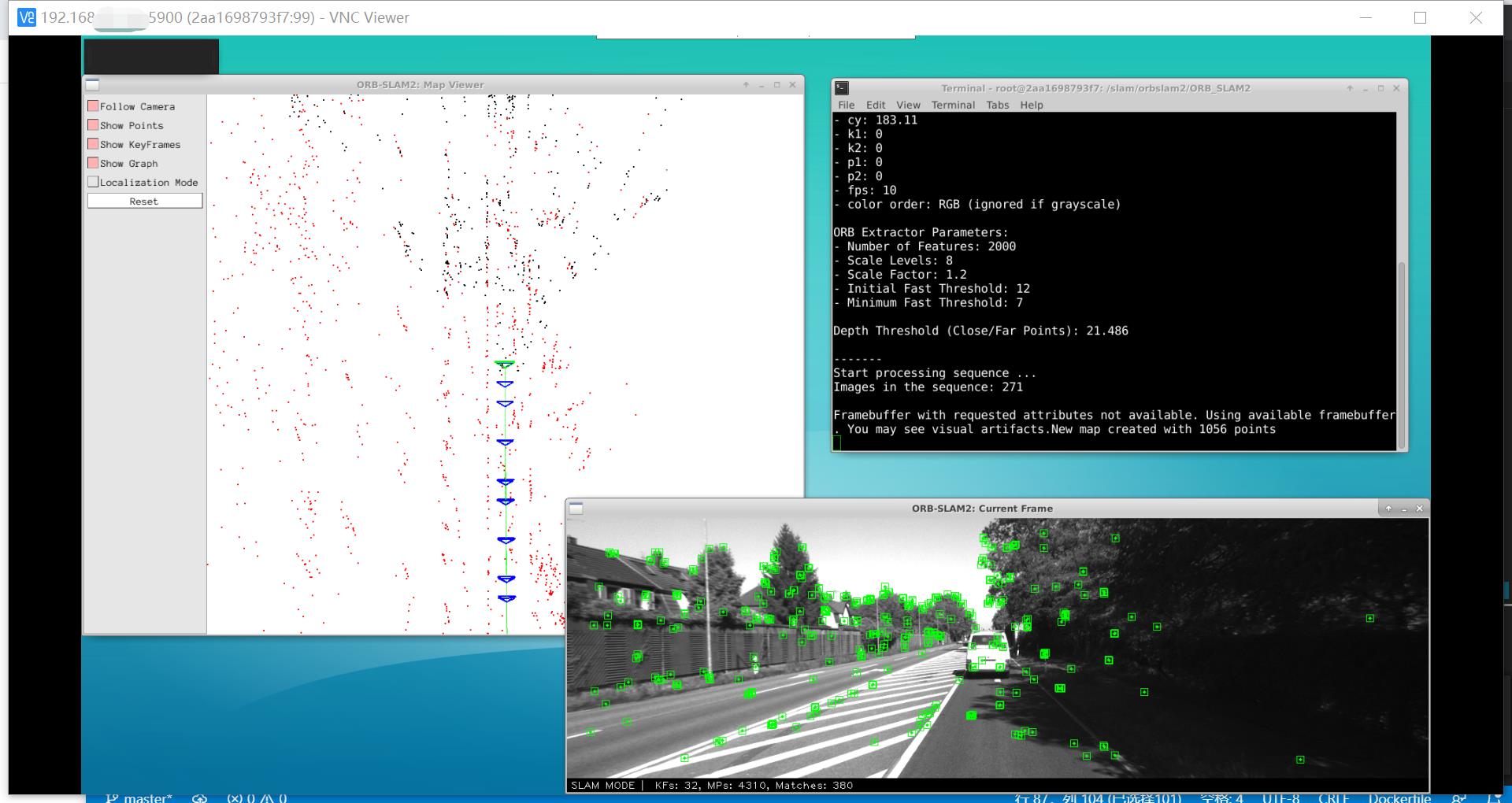
### Connect VNC Desktop port 5900 to host:

Then open VNC Viewer Desktop and use adress localhost:5900 to connect. And in this image, the access is free (no password) as default. (Note: docker toolbox in wins, the adress of localhost is a special adress assigned to your docker VM, ex: http://192.168.xx.xx:5900/)

### Pangolin routines

cd /slam/orbslam2/ORB\_SLAM2

./Examples/Stereo/stereo\_kitti Vocabulary/ORBvoc.txt Examples/Stereo/KITTI04-12.yaml /root/Dataset/04

[](https://github.com/buaalqh/Git-Operation-commands/blob/master/Images/orbslam2-docker1.0-3.jpg)

* V2.0[[4]](#footnote-4) is based on Ubuntu 20.04

## Get image

### Pull the image from DockerHub:

docker pull qihaoliu/orbslam2-docker:v2.0

### Or use Dockerfile to build locally:

git clone --branch v2.0 https://github.com/buaalqh/orbslam-docker.git

cd ./orbslam-docker

docker build -t qihaoliu/orbslam2-docker:v2.0 .

## Run the docker container

### Download Dataset

This ORB\_SLAM2 routine provides a demonstration program for the [KITTI odometry](http://www.cvlibs.net/datasets/kitti/eval_odometry.php) Dataset. Here, using sequence 04 in [data\_odometry\_gray](http://www.cvlibs.net/datasets/kitti/eval_odometry.php).

### Create container

Execute the following command to instantiate the container:

docker run -it --rm -v YOUR\_PATH\_TO\_KITTI/sequences/04:/root/Dataset/04 -p 6080:80 -p 5900:5900 qihaoliu/orbslam2-docker:v2.0

### Connect VNC Desktop port 5900 to host:

Then open VNC Viewer Desktop and use adress localhost:5900 to connect. And in this image, the access is free (no password) as default. (Note: docker toolbox in wins, the adress of localhost is a special adress assigned to your docker VM, ex: http://192.168.xx.xx:5900/)

### Connect VNC web port 6080 in browser[[5]](#footnote-5):

Note: If you would like to protect vnc service by password, set environment variable VNC\_PASSWORD, for example

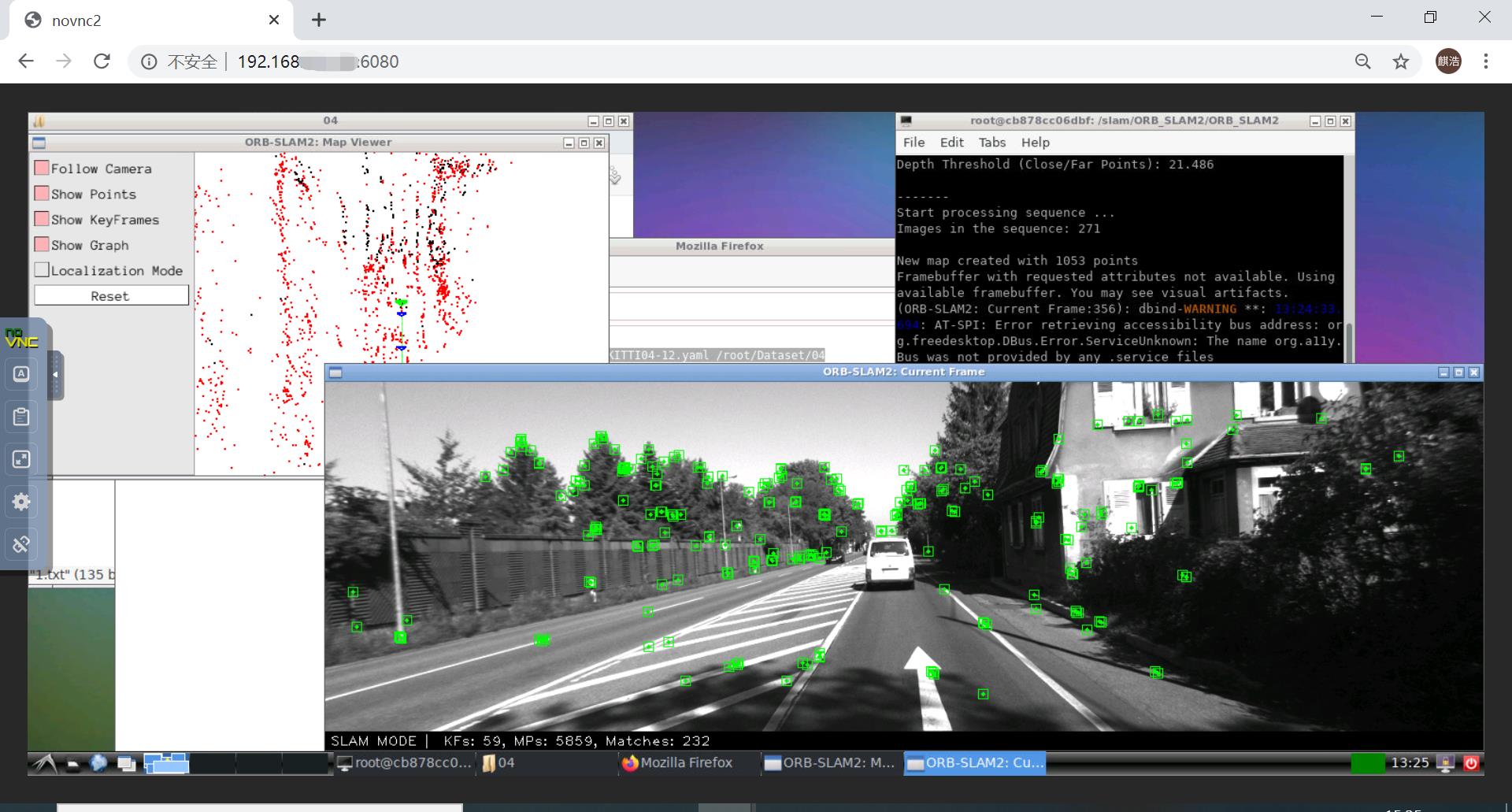
docker run -it --rm -e VNC\_PASSWORD=mypassword -v YOUR\_PATH\_TO\_KITTI/sequences/04:/root/Dataset/04 -p 6080:80 -p 5900:5900 qihaoliu/orbslam2-docker:v2.0

A prompt will ask password either in the browser or vnc viewer.

## Pangolin routines

cd /slam/ORB\_SLAM2/ORB\_SLAM2

./Examples/Stereo/stereo\_kitti Vocabulary/ORBvoc.txt Examples/Stereo/KITTI04-12.yaml /root/Dataset/04

[](https://github.com/buaalqh/Git-Operation-commands/blob/master/Images/orbslam2-docker2.0-4.jpg)

## End

* **Dockerfile[[6]](#footnote-6) of V1.0 (details see files in my Docker hub and Git hub):**

FROM ubuntu:16.04

MAINTAINER Qihao LIU <qihao.liu@student-cs.fr>

# Get dependencies for Pangolin (logical for visualization and user interface)

RUN apt-get update && apt-get install -y \

        libgtk2.0-dev \

……

ENV OPENCV 3.4.1 #Set environment

# Install OpenCV

RUN cd /root && \

    wget https://github.com/opencv/opencv/archive/3.4.1.zip && \

    unzip 3.4.1.zip && \

    cd opencv-3.4.1 && mkdir build && cd build && \

    cmake -D CMAKE\_BUILD\_TYPE=RELEASE -D WITH\_CUDA=OFF -D WITH\_OPENGL=OFF .. && \

    make && make install

# Install Pangolin

# Install Eigen3

# Get orb-slam2 from github and compile it

RUN mkdir -p /slam

WORKDIR /slam

RUN cd /slam && \

    git clone https://github.com/buaalqh/orbslam2.git && \

    cd orbslam2/ORB\_SLAM2 && chmod +x build.sh && sh build.sh

# Enable remote access to the Docker container desktop by VNC

ENV DEBIAN\_FRONTEND noninteractive

RUN apt-get update -y && apt-get install -y \

    openssh-server xfce4 xfce4-goodies x11vnc sudo bash xvfb && \

    useradd -ms /bin/bash ubuntu && echo 'ubuntu:ubuntu' | chpasswd && \

    echo "ubuntu ALL=(ALL) NOPASSWD:ALL" >> /etc/sudoers && apt-get clean && \

    rm -rf /var/lib/apt/lists/\*

COPY    x11vnc /etc/init.d/ # configuration remote vnc desktop

COPY    xvfb /etc/init.d/ # congifuration Virtual display server

COPY    entry.sh / # start vnc server

RUN     sudo chmod +x /entry.sh /etc/init.d/\*

# port for connecting VNC

EXPOSE  5900

# Set the command to run when the container starts

ENTRYPOINT [ "/entry.sh" ]

1. <https://github.com/raulmur/ORB_SLAM2> [↑](#footnote-ref-1)
2. <https://hub.docker.com/r/qihaoliu/orbslam2-docker> [↑](#footnote-ref-2)
3. <https://github.com/buaalqh/orbslam-docker/tree/v1.0> [↑](#footnote-ref-3)
4. <https://github.com/buaalqh/orbslam-docker/tree/v2.0> [↑](#footnote-ref-4)
5. <http://192.168.xx.xx:6080/> [↑](#footnote-ref-5)
6. <https://github.com/buaalqh/orbslam-docker/tree/v1.0> [↑](#footnote-ref-6)