# WASS stereo video procees using in cerhouse $shot\ tutorial$

Pedro V. Guimarães

February 23, 2017

# Contents

1	WASS instalation	2
	1.1 WASS in a Docker container	2
	1.1.1 Create a image	3
	1.1.2 Create a container	4
<b>2</b>	Calibration	4
3	Run a sample	5
	3.1 Get a sample	5
	3.2 wass autocalibrate	5
	3.3 Create Grid.mat file	5
	3.4 Create StrNum_List.txt and StrNum_List_multi.txt	6
4	Directory structures and dependencies	6
<b>5</b>	Run WASS in cerhouse1	7

# 1 WASS instalation

To install WASS in personal computes is better check the official webpage: http://www.dais.unive. it/wass/documentation/install.html

To install in ubuntu 14.04 several modifications are needed, for that check: https://github.com/fbergama/wass/issues/1#issuecomment-252091485

Maybe interesting take a look in  $\mathit{TOOLSWASSinstall\_wass.sh}$ 

### 1.1 WASS in a Docker container

As *cerhouse* uses a old ubuntu version an efficient solution to run wass in the server is using a Docker container.

For more informations about Docker see: https://www.docker.com/ There is an image already created in:

/home/cercache/project/stereo/WASS/WASSdocker/docker\_image\_wass\_v1.0.tar

and a container should be available in all nodes named as:

container\_wass

If it is not contact cloud suport for it!

#### 1.1.1 Create a image

Anyway, if for some reason it is not working anymore. Goes to a node with a docker installed, as br156-061.

```
>> ssh br156-061
>> <mark>cd</mark> /home/cercache/project/stereo/WASS/WASSdocker/
>> vi Dockerfile
```

Dockerfile

```
FROM ubuntu:16.04
RUN groupadd -g 10622 lpogtech
RUN groupadd -g 10731 lpo
RUN groupadd -g 11115 gswarp
RUN groupadd -g 10042 droos
RUN useradd -d /home/pverasgu -u 25016 pverasgu -M -g lpo -g lpogtech -G droos
RUN apt-get -y update && apt-get install -y git
RUN apt-get -y update && apt-get install -y build-essential
RUN apt-get -y update && apt-get install -y cmake
RUN apt-get -y update && apt-get install -y curl
RUN apt-get -y update && apt-get install -y liblapack-dev
RUN apt-get -y update && apt-get install -y libblas-dev
RUN apt-get -y update && apt-get install -y libboost-all-dev
RUN apt-get -y update && apt-get install -y ffmpeg
RUN apt-get -y update && apt-get install -y libavcodec-dev
RUN apt-get -y update && apt-get install -y libavformat-dev
WORKDIR /WASSdocker/
RUN git clone https://github.com/opencv/opencv.git --depth 1
WORKDIR "opency"
RUN mkdir build && cd build && \
   cmake ../ -DCMAKE_INSTALL_PREFIX="../dist/" -DCMAKE_BUILD_TYPE="Release" && \
   make && make install
WORKDIR /WASSdocker/
RUN git clone https://github.com/fbergama/wass
WORKDIR "wass"
RUN git submodule update --init
RUN mkdir build && cd build && \
   cmake ../src/ -DOpenCV_DIR="../../opencv/build" && \
   make && make install
RUN cd WASSjs && cd ext && \
   tar xvfz redis-2.8.19.tar.gz && cd redis-2.8.19 && make
RUN echo "WASS files are in ${PWD}/dist/bin" && ls dist/bin && echo "DONE"
```

To built and save the image:

>> docker built -t wass . >> docker save -o docker\_image\_wass\_v<u>1.0.tar wass</u>\_

For configure the user ID information for groupadd, useradd ..., tipe in the terminal:

>> id

After fill in the Dockerfile with your user informations, like:

```
RUN groupadd -g 10158 cersat
RUN groupadd -g 10152 droos
RUN useradd -d /home/agrouaze -u 54605 agrouaze -M -g cersat -G droos
```

#### 1.1.2 Create a container

One container should be create in each node with docker installed. A container called 'container\_wass' already exist. But it is also possible add this command lines in the .pbs script or directly run in the terminal: Command to load the image

```
>> docker -H :4243 load < /home/cercache/project/stereo/WASS/WASSdocker/
docker_image_wass_v1.0.tar
```

Command to create the container

```
>> docker run -i -t -d --name='container_wass' -v /home/cercache:/home/cercache -v /
home/cerdata:/home/cerdata -v /home/cersat5:/home/cersat5 -v /mnt:/mnt -v /home3/
homedir7:/home3/homedir7 wass:latest /bin/bash
```

Command to list containers

>> docker **ps** 

Command to close the container

>> docker rm container\_name

# 2 Calibration

The Calibration should be done as usual, see for that:

```
TOOLS/WASS/DOC/Calibration_tutorial.txt
```

After the Calibration, it is need to set the *config* folder and set the *.xml* files considering this calibration. The *config* must have this files:

```
distortion_00.xmlintrinsics_00000000.xmlmatcher_config.txtdistortion_01.xmlintrinsics_00000001.xmlstereo_config.txtext_R.xmlintrinsics_00.xmlext_T.xmlintrinsics_01.xml
```

For more informations about this files check official wass webpage http://www.dais.unive.it/wass/index.html

In matlab: Go to your *Calib* folder and use the function *calib2xml* to create the *.xml* files:

```
cd ../Calib
calib2xml('Calib_Results_stereo.mat');
!cp -r ~/SOFT/WASS/wass/test/WASS_TEST/W07/config .
!cp ../Calib/*.xml config/.
```

# 3 Run a sample

Before run it in a cloud could be good to run a small sample in personal computer to set correctly the directory and files configurations.

#### 3.1 Get a sample

Get a sample of images, this script could be useful:

```
TOOLS/WASS/PROCESS/get_images.sh
```

There are tow easy scripts to run a stereo sample like:

```
TOOLS/WASS/PROCESS/Run_WASS_serial.sh
or
TOOLS/WASS/MATLAB/run_WASS.m
```

#### 3.2 wass autocalibrate

If the idea is just use the  $wass\_autocalibrate$  tool to update the Calib/New\_Stereo\_Calib.mat file. The wass\\_autocalibrate tool will create a scaled  $ext\_R.xml$  and  $ext\_T.xml$  files result. However the T vector should be rescaled considering the cameras distance.

In matlab, load the stereo calib file and update the T and R values

```
cd ../ Calib
load('Calib_Results_stereo.mat');
scale=T(1);
T = [-1.9950]
   -0.0722
   -0.4328]*scale
R = [0.9998]
               0.0112
                         -0.0141
   -0.0112
               0.9999
                         -0.0022
               0.0024
                          0.9999];
    0.0141
save('Calib_Results_stereo_autocalib.mat')
```

After that is just rerun the *Create\_Calib\_Index\_Files.m*, in matlab:

Create\_Calib\_Index\_Files >> 1 >> 2 >> Calib\_Results\_left.mat >> Calib\_Results\_right.mat >> Calib\_Results\_stereo\_autocalib.mat

#### 3.3 Create Grid.mat file

Before go to the cloud, without matlab interface, it is important define the grid zone. For that just run the matlab scritp: *xyzC2nc.m*, in your computer, it will call the *Define\_limits\_for\_grid.m* from old stereo TOOLS.

```
\| xyzC2nc(000001)
```

As output it will create a config/Grid.mat required to create 3D/Surface\_NNNNN.nc

#### 3.4 Create StrNum\_List.txt and StrNum\_List\_multi.txt

The  $StrNum\_List.txt$  is use to run a qsub or a single job,  $StrNum\_List\_multi.txt$  is use to run a goglist job. To create it use the function  $Create\_StrNum\_List.m$  as it is done in the old toolbox. In matlab:

```
cd run_2016-10-17_15h06m30.189sZ
d=dir('Images_Raw/*_01.tif');
disp([' Number of files: ',d(1).name(1:6),' to ',d(end).name(1:6)])
Create_StrNum_List(str2num(d(1).name(1:6)),str2num(d(end).name(1:6)),6)
```

Also see *TOOLS/WASS/MATLAB/create\_list.m*, could be helpful.

## 4 Directory structures and dependencies

Maybe some directly path can or have change, they are usually set in the begin of  $Run_WASS.sh$ , or  $run_WASS.m$ , or ....

The most important directory in *run\_yyyy-mm-dd\_HHhMMmSSsZ*/ to car are:

```
>> <mark>cd</mark> run_2016-10-17_15h06m30.189sZ
>> ls
3D Images_Raw config output StrNum_List_multi.txt StrNum_List.txt
```

One level below it must have ../Calib Verify in Run\_WASS.sh if the 'CALIB\_DIR' is correctly defined, like:

CALIB\_DIR='/home/cercache/project/stereo/DYNATREZ\_2016/raw/Calib'

For DIAMCAM acquisition, the images do not need to be renamed anymore. If it is the case the images raw will be in cam\_0 and cam\_1:

```
>> cd run_2016-10-17_15h06m30.189sZ
>> ls
3D cam_0 cam_1 config output StrNum_List_multi.txt StrNum_List.txt
```

In this case it is necessary just edit the Run\_WASS.sh to consider this other structure.

```
INPUT_C0_DIR = ${WORK_ROOT}'/cam_0/'
INPUT_C1_DIR = ${WORK_ROOT}'/cam_1/'
cam0sufix='' # if is need modify it for other images format just here
cam1sufix='' # if is need modify it for other images format here
cam0idx='_01.tif' # if is need modify it for other images format here
cam1idx='_02.tif' # if is need modify it for other images format here
```

# 5 Run WASS in cerhouse1

Considering the all run\_... are correctly configured and the *container\_wass* already exist in already exist: Goes to project directory and copy Run\_WASS scripts as example:

```
>> ed /home/cercache/project/stereo/DYNATREZ_2016/raw/
>> cp /home/cercache/project/stereo/TOOLS/WASS/PROCESS/Run\_WASS.sh .
>> cp /home/cercache/project/stereo/TOOLS/WASS/PROCESS/Run\_WASS.pbs .
>> cp /home/cercache/project/stereo/TOOLS/WASS/PROCESS/Run\_WASS.gog .
```

Configure the 'CALIB\_DIR' and other directory definitions if it is needed:

```
CALIB_DIR='/home/cercache/project/stereo/DYNATREZ_2016/raw/Calib'
INPUT_C0_DIR = ${WORK_ROOT}'/cam_0/'
INPUT_C1_DIR = ${WORK_ROOT}'/cam_1/'
...
```

In Run\_WASS.pbs and/or Run\_WASS.gog verify the EXE path:

EXE="/home/cercache/project/stereo/TOOLS/WASS/PROCESS/Run\_WASS.sh"

#### Goglist

> ed run\_2016-10-17\_15h06m30.189sZ/

 $>> \dots / \operatorname{Run}_WASS.gog$ 

or specify the list file:

>> ../Run\_WASS.gog "list\_file.txt

>> qstat -u pverasgu

qsub

>> cd run\_2016-10-17\_15h06m30.189sZ/

>> qsub .../Run\_WASS.pbs

or specify the list file:

>> qsub .../Run\_WASS.pbs -F "list\_file.txt

Verify the output:

>> qstat -u pverasgu >> tail -f WASS\_pbs.out.\$\$