GROUP PROJECT

Robotics Data Preprocessing for Training of Machine Learning Model

UNIVERSITY OF GENOVA

ACADEMIC YEAR - 2021 / 2022

ROBOTICS ENGINEERING

Supervisors :

Alessandro Carfì

Simone Macciò

Valerio Belcamino

Done By:

Manoj Kunapalli

S4848891

INTRODUCTION

An experiment is performed with the Baxter robot by multiple people and the various actions of the robot are recorded and a huge datasets are generated performing various actions by the robot. The videos contained in the dataset have already been labelled, that is each video has been analysed and has inserted manual labels (in the image metadata) indicating the beginning and end of actions performed by the participants.

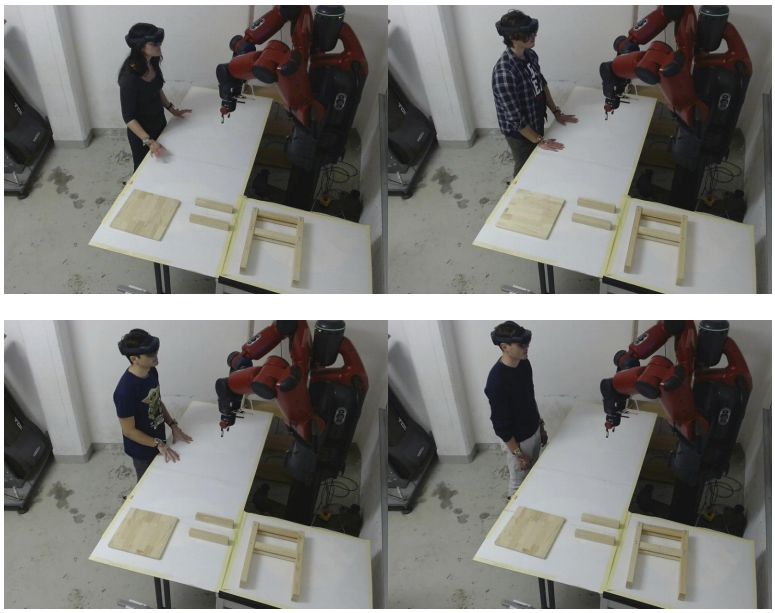

fig1 : Multiple people performing experiments with the Baxter Robot.

AIM of the Project

Preprocessing of data from the given huge dataset in such a way that it can be used for training purpose by removing all the unnecessary data and synchronizing the sensor data from all the required files.

Tasks to be done

- Writing a script which extracts the timestamps corresponding to each image, according to the action label.
- To synchronize the action timestamps with the accelerometer data's timestamps, obtain a precise correspondence between each action and its resulting acceleration pattern.
- To Find all the actions in the metadata of Images, extract the information, find the start and end labels of its data, and synchronize it with the required sensory output.

Information about the Dataset

Regarding the files, you will see that each folder is structured as follows:

- a /frames sub-folder, containing the sequence of images which make up the video of the experiment
- a timestamp file (in txt format), specifying each frame's timestamp in the aforementioned subfolder.
- 4 text files containing the data of wearable accelerometers worn by the person during the experiment. In particular, each person was wearing one accelerometer on the back of the hand and another one on the wrist, both for the left and right arm. The data in the txt file also contain the timestamps, specifying the instant in which each sample was acquired.

METADATA

Photo metadata is a set of data describing and providing information about the rights and administration of an image.

It allows information to be transported with an image file, in a way that can be understood by other software and human users.

The pixels of image files are created by automated capture from cameras or scanners.

Extraction of Metadata

Now the first step is to extract the Metadata of all the images. For this, we can use an ExifTool extractor.

Download exiftool.exe from https://exiftool.org/

Just download and un-zip the archive then double-click on "exiftool(-k).exe" to read the application documentation, drag-and-drop files and folders to view meta information, or rename to "exiftool.exe" for command-line use. Runs on all versions of Windows.

Save the output of the metadata as **out.csv**

```
======== G:/dataset/s01/frames/frame_1.jpeg
ExifTool Version Number         : 12.43
File Name                       : frame_1.jpeg
Directory                       : G:/dataset/s01/frames
File Size                       : 50 kB
File Modification Date/Time     : 2022:02:16 11:51:58+01:00
File Access Date/Time           : 2022:07:15 00:07:01+02:00
File Creation Date/Time         : 2022:02:16 11:51:58+01:00
File Permissions                : -rw-rw-rw-
File Type                       : JPEG
File Type Extension             : jpg
MIME Type                       : image/jpeg
JFIF Version                    : 1.01
Resolution Unit                 : None
X Resolution                    : 1
```

Y Resolution	: 1	
Image Width	: 800	
Image Height	: 600	
Encoding Process	: Baseline DCT	Huffman coding
Bits Per Sample	: 8	
Color Components	: 3	
Y Cb Cr Sub Sampling	: YCbCr4:2:0 (2 2)	
Image Size	: 800x600	
Megapixels	: 0.480	

To extract the timestamps corresponding to each image, according to the label and synchronize the action timestamps with the timestamps of the accelerometer data

Download python 3

Clone the repository : https://github.com/Manojkun1996/Group-Project.git

Navigate to Group-Project/datasets/s01/

Launch the python file Meta_data.py

Enter the file location in the script example: FILE_LOCATION = r"G:\datasets\s01\out.csv"

RUN the program.

We get the synchronized output as shown below files.

my_data_left_backPose.csv, my_data_left_wristPose.csv, my_data_right_backPose.csv, my_data_right_wristPose.csv

timestamp	a_left_backPose	b_left_backPose	c_left_backPose	File Name
1.64E+18	8288	1423	2907	frame_1000.jpeg
1.64E+18	8498	1159	2732	frame_1001.jpeg
1.64E+18	8498	1159	2732	frame_1002.jpeg
1.64E+18	8143	1147	2806	frame_1003.jpeg
timestamp	a_right_wristPose	b_right_wristPose	c_right_wristPose	File Name
1.64E+18	2208	4841	7232	frame_1000.jpeg
1.64E+18	2293	4659	6766	frame_1001.jpeg
1.64E+18	2015	3922	6289	frame_1002.jpeg
1.64E+18	2138	4607	6953	frame_1003.jpeg
timestamp	a_right_backPose	b_right_backPose	c_right_backPose	File Name
1.64E+18	-3578	-5396	-6055	frame_1000.jpeg
1.64E+18	-3283	-5021	-5485	frame_1001.jpeg
1.64E+18	-2948	-4141	-5348	frame_1002.jpeg
1.64E+18	-3482	-4641	-5721	frame_1003.jpeg
timestamp	a_right_wristPose	b_right_wristPose	c_right_wristPose	File Name

1.64E+18	2208	4841	7232	frame_1000.jpeg
1.64E+18	2293	4659	6766	frame_1001.jpeg
1.64E+18	2015	3922	6289	frame_1002.jpeg
1.64E+18	2138	4607	6953	frame_1003.jpeg

Types of actions:

ASSEMBLY1_BIMANUAL, ASSEMBLY2_BIMANUAL, ASSEMBLY2_RIGHT

BOLT_LEFT, BOLT_RIGHT

DELIVERY_BIMANUAL, DELIVERY_RIGHT

HANDOVER_LEFT, HANDOVER_RIGHT, IDLE

PICKUP_LEFT, PICKUP_RIGHT, SCREW_RIGHT

To get the information regarding each particular action

Here we have to get the data b/w all the action labels of start and end w.r.t to the timestamps and the sensory data.

Navigate to Group-Project/datasets/s01/

Launch the python file Meta_data_actions.py

Enter the file location in the script example: FILE_LOCATION = r"G:\datasets\s01\out.csv"

RUN the program.

You find all the outputs in the files

last-action-df1, last-action-df2, last-action-df3, last-action-df4

The output shown below is coded as 0 if the action is not present and as 1 if action is present at that particular time stamp.

Example output:

timestamp	File Name	ASSEMBLY1_BIMANUAL	ASSEMBLY2_BIMANUAL	ASSEMBLY2_RIGHT	BOLT_LEFT	BOLT_RIGHT	DELIVERY_BIMANUAL	DELIVERY_RIGHT
1.64E+18	frame_416.jpeg	0	0	0	0	0	0	1
1.64E+18	frame_417.jpeg	0	0	0	0	0	0	1
1.64E+18	frame_418.jpeg	0	0	0	0	0	0	1
1.64E+18	frame_419.jpeg	0	0	0	0	0	0	1

1.64E+18	frame_808.jpeg	1	0	0	0	0	0	0
1.64E+18	frame_809.jpeg	1	0	0	0	0	0	0
1.64E+18	frame_810.jpeg	1	0	0	0	0	0	0
1.64E+18	frame_811.jpeg	1	0	0	0	0	0	0
1.64E+18	frame_812.jpeg	1	0	0	0	0	0	0
1.64E+18	frame_813.jpeg	1	0	0	0	0	0	0

We get the required output of all the respected actions individually, which u can see inside the folders: data_left_backPose, data_left_wristPose, data_right_backPose, data_right_wristPose

A, b, c are all the sensory data.

The Example output is shown below

timestamp	File Name	a_left_backPose	b_left_backPose	c_left_backPose	ASSEMBLY1_BIMANUAL
1.64E+18	frame_704.jpeg	4353	2557	6583	1
1.64E+18	frame_705.jpeg	4353	2557	6583	1
1.64E+18	frame_706.jpeg	4416	2406	6384	1
1.64E+18	frame_707.jpeg	4416	2406	6384	1

timestamp	a_left_wristPose	b_left_wristPose	c_left_wristPose	BOLT_LEFT
1.64E+18	-2920	-8458	-705	1
1.64E+18	-3151	-8768	487	1
1.64E+18	-2847	-7496	845	1
1.64E+18	-3160	-7592	1201	1

timestamp	a_right_backPose	b_right_backPose	c_right_backPose	DELIVERY_RIGHT
1.64E+18	-11606	-3197	-6343	1
1.64E+18	-10951	-3554	-5550	1
1.64E+18	-7735	-3313	-3589	1
1.64E+18	-3800	-3691	-1555	1

timestamp	a_right_wristPose	b_right_wristPose	c_right_wristPose	SCREW_RIGHT
1.64E+18	-455	9214	1441	1
1.64E+18	-575	7044	1685	1
1.64E+18	-637	6839	1417	1
1.64E+18	-723	8298	2546	1

CONCLUSION

The required data of all the actions between start and end labels are extracted and are synchronized with respect to the sensory data and timestamps. Preprocessing of the data is completed for training of the machine learning model.