AHMEDABAD UNIVERSITY
School of Engineering and Applied Science

Human Activity Recognition

End Semester Presentation

Prepared by: Group 07

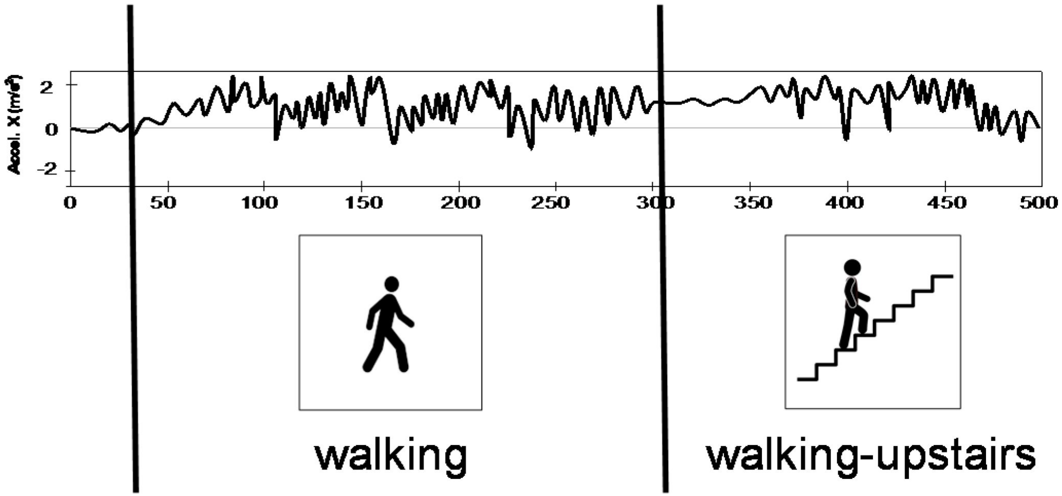
Ashna Jain (201501008)
Dhruti Chandarana(201501015)
Charmi Chokshi (201501021)
Hena Ghonia (2015010132)
Janvi Patel (201501072)
Nishi Patel (201501076)

Subject: Computer Vision
Guided by: Dr. Mehul Raval
Date: September 19, 2018

Problem Statement

Deep Learning based Human Activity Recognition

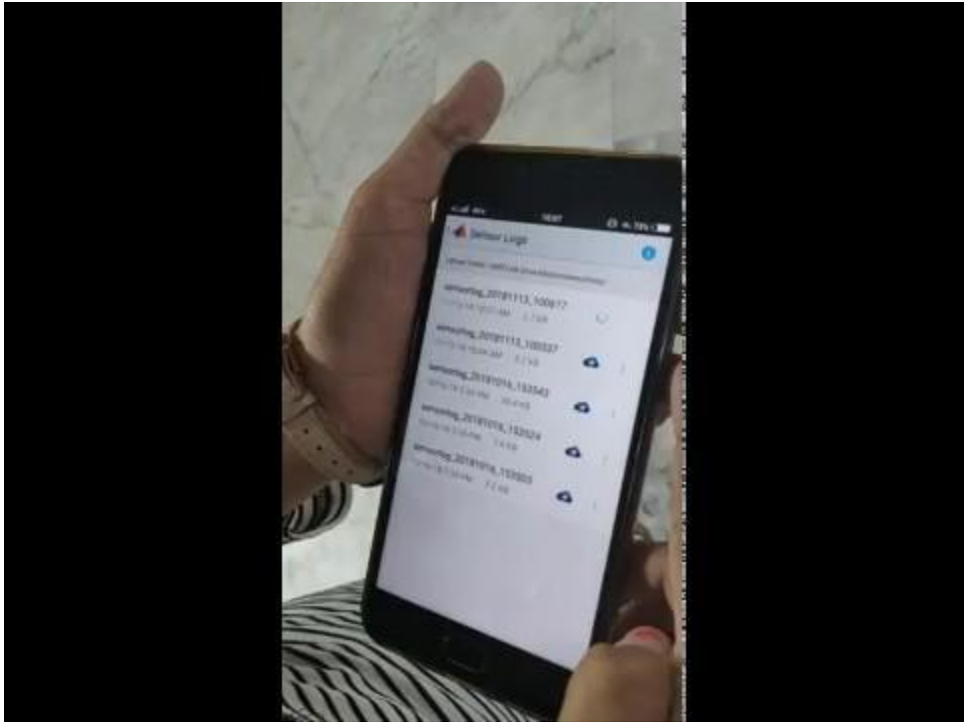
- The objective is to classify activities into one of the activities performed.



Data Set

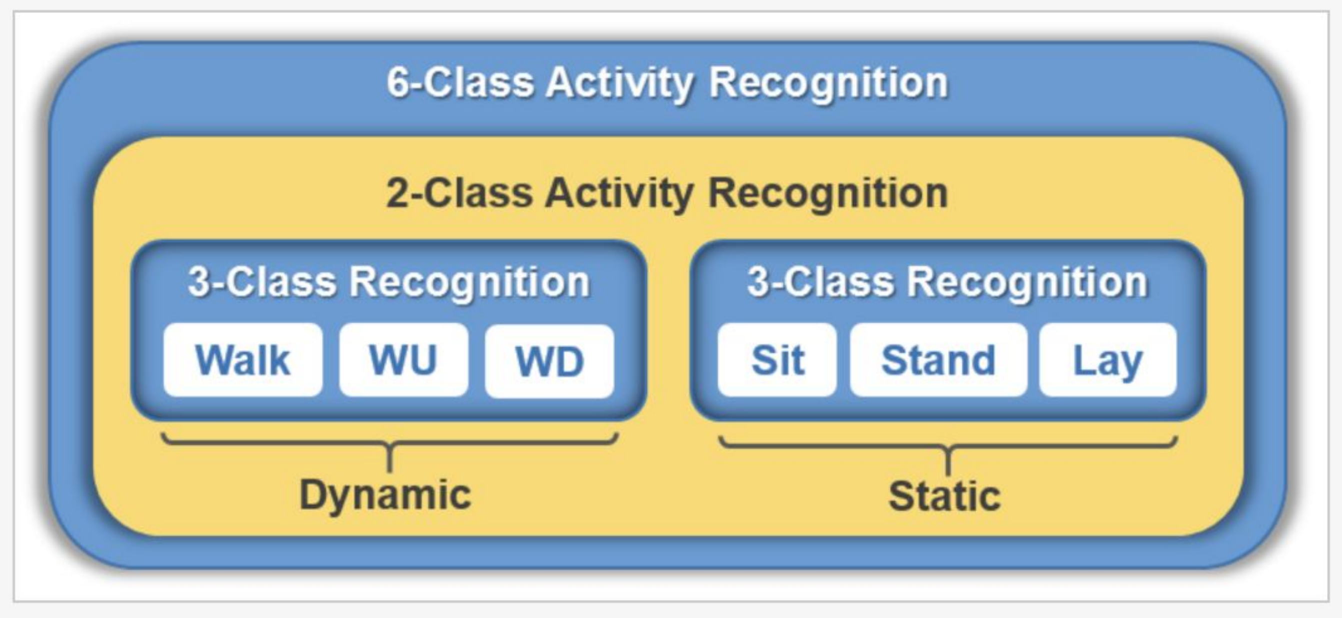
- **UCI HAR dataset**
- The dataset was collected from the **in-built accelerometer and gyroscope of a smartphone** worn around the waist of participants.
- Number of subjects: 30
- Number of activities: 06
    - **Walking**
    - **Walking upstairs**
    - **Walking downstairs**
    - **Sitting**
    - **Standing**
    - **Laying**

Data Gathering

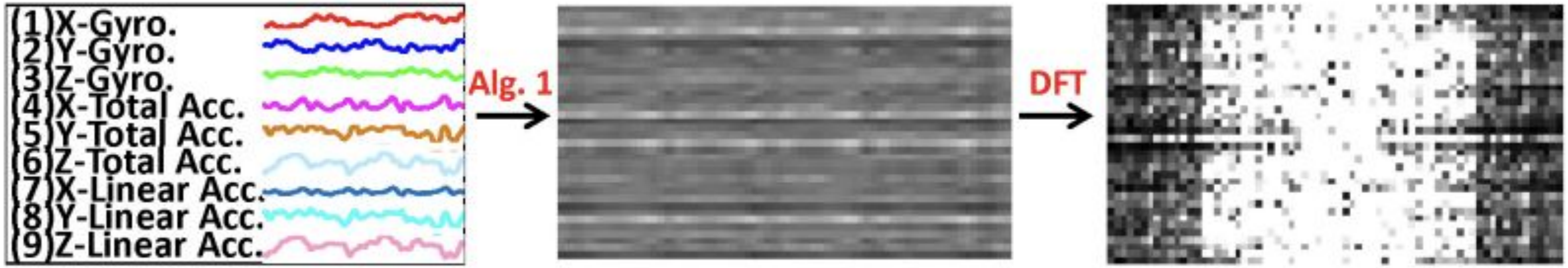


Data Gathering

Separation of Activities



Processing of Raw Sensor Data into an Image



(a).Raw Signals (RS) (b).Signal Image (SI) (c).Activity Image (AI)

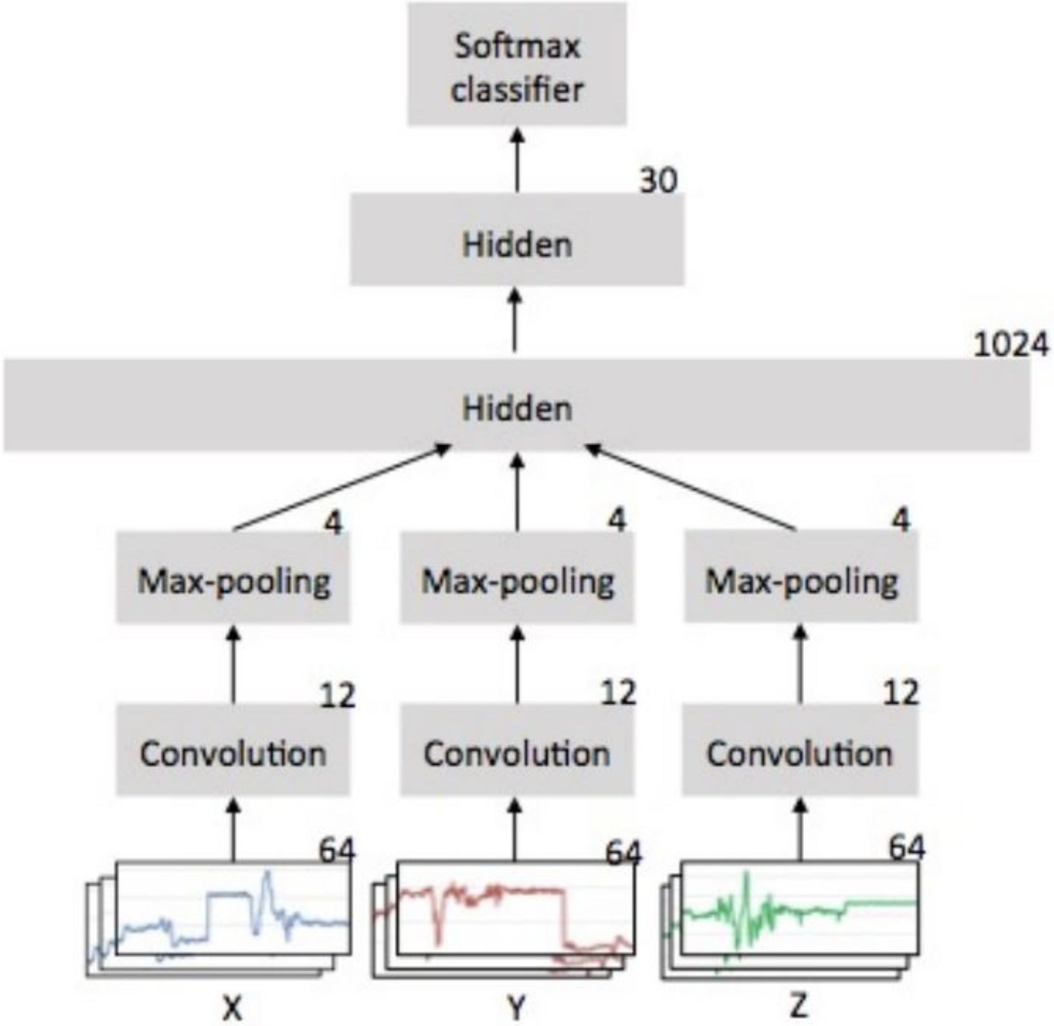
Comparative study of Literature

Article name	Algorithm	Performance metrics	Dataset used
Two Stream Convolutional Networks for Action Recognition Videos [6]	Spatial ConvNet and temporal stream ConvNet -> fusion of two ConvNet by averaging and multiclass SVM.	Maximum accuracy achieved for UCF-101:88% and for HMDB-66.8%	UCF101 and HMDB-51
Deep convolutional and LSTM Recurrent Neural Networks for Multimodal Wearable Activity Recognition[2]	DeepConvLSTM for recurrent cells and Baseline CNN for non recurrent cells and fully connected network. MLP but input and output are independent; RNN used for fed back to itself and to provide memory of past activities. LSTM with RNN is used to learn temporal dynamic of sequential data.	Maximum accuracy for OPPORTUNITY - 88% Skoda - 71.3%	OPPORTUNITY (D01) Skoda (D02) mHealth (D13)
Human Activity Recognition using Wearable Sensors by Deep Convolutional Neural Networks [3]	Sensed signals from Accelerometer, gyroscope - From Raw signals stacked image - Discrete Fourier Transform - Activity image - DCNN - Low level to High level features learning - Activity Recognition	Maximum accuracy for UCI - 95.18 % USC - 97.01 % SHO - 99.93 %	UCI, USC, SHO

Comparison study of Literature review

Article name	Algorithm	Performance metrics	Dataset used
LSTM Networks for Mobile Human Activity Recognition [4]	LSTM+Multiclass classification	Accuracy : 92.1%	WISDM
Deep, Convolutional, and Recurrent Models for Human Activity Recognition Using Wearables[5]	Finding impact of hyperparameters using fANOVA framework. Three different approaches: CNN, DNN, RNN(bi-directional and forward in time) Dropout and Max-in-Norm regularization	Maximum accuracy for D04 dataset- CNN Approach- 93.7% D01 dataset- RNN(bi-directional) Approach- 74.5% D14 dataset- RNN(forward in time) Approach- 76%	D01, D04, D14

CNN based Model architecture



Experimental Setup

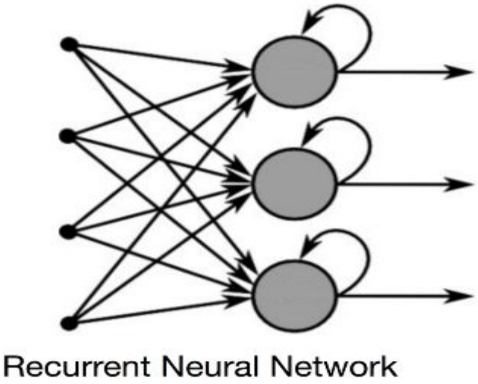
Parameter	Value
The size of input vector	128
The number of input channels	6
The number of feature maps	10–200
Filter size	1×3–1×15
Pooling size	1×3
Activation function	ReLU (rectified linear unit)
Learning rate	0.01
Weight decay	0.00005
Momentum	0.5–0.99
The probability of dropout	0.8
The size of minibatches	128
Maximum epochs	5000

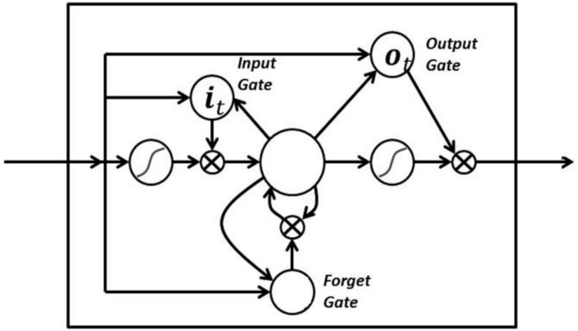
Result of CNN based architecture

dropout	learning_rate	training_epoch	training_accuracy	testing_accuracy
1	0.001	100	0.9592	0.8782
0.9	0.001	100	0.9334	0.8724
0.85	0.001	100	0.945	0.868
0.8	0.001	50	0.936	0.866
0.8	0.001	100	0.925	0.865
1	0.001	30	0.9237	0.8629
1	0.001	100	0.9539	0.8622
0.75	0.001	50	0.935	0.861
0.9	0.001	100	0.9162	0.8588
1	0.005	30	0.9377	0.8558

LSTM vs RNN

- A Recurrent Neural Network is able to remember its past, because of it’s internal memory. It produces output, copies that output and loops it back into the network.
- There are two issues of standard RNN: Exploding and Vanishing Gradients.
- Long Short-Term Memory (LSTM) networks are an extension for recurrent neural networks, which basically extends their memory and solves the problem of vanishing and exploding gradients.

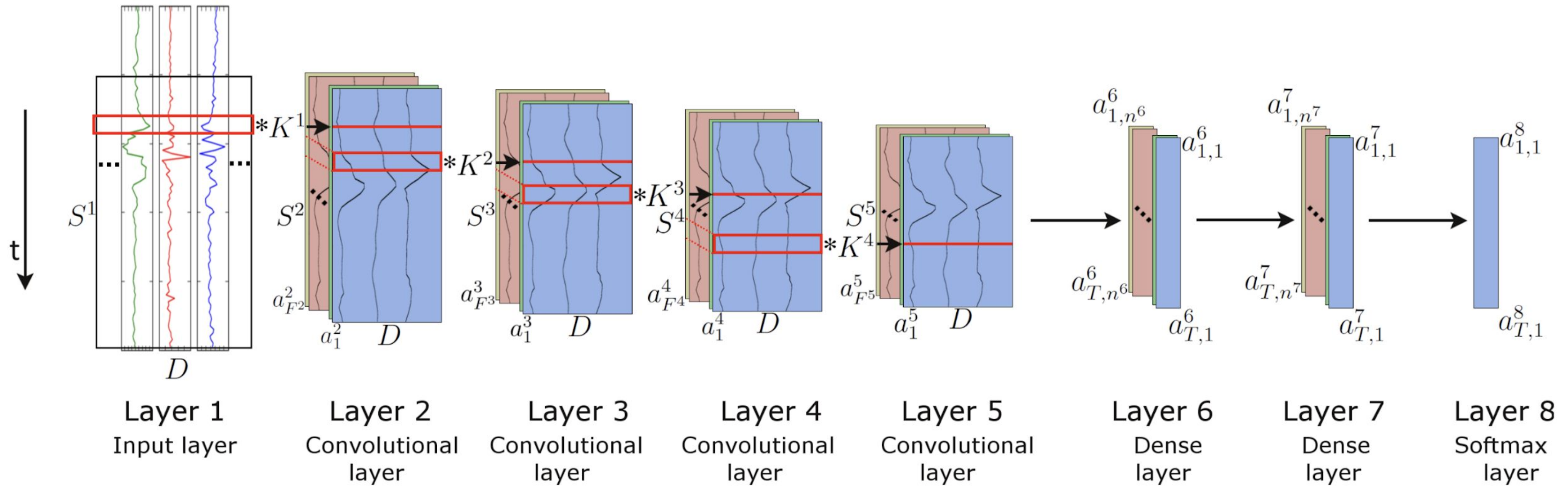




ConvLSTM approach

- CNN ->read subsequences of the main sequences in block->extract feature from each block
- LSTM->interpret the features extracted from each block.
- Input:
    - **Samples**: n, for the number of windows in the dataset.
    - **Time**: 4, for the four subsequences that we split a window of 128 time steps into.
    - **Rows**: 1, for the one-dimensional shape of each subsequence.
    - **Columns**: 32, for the 32 time steps in an input subsequence.
    - **Channels**: 9, for the nine input variables.

CNN-LSTM based Architecture



Model summary

```
_________________________________________________________________
Layer (type)                 Output Shape              Param #
=================================================================
time_distributed_56 (TimeDis (None, None, 30, 64)      1792
_________________________________________________________________
time_distributed_57 (TimeDis (None, None, 28, 64)      12352
_________________________________________________________________
time_distributed_58 (TimeDis (None, None, 28, 64)      0
_________________________________________________________________
time_distributed_59 (TimeDis (None, None, 14, 64)      0
_________________________________________________________________
time_distributed_60 (TimeDis (None, None, 896)         0
_________________________________________________________________
lstm_12 (LSTM)               (None, 100)               398800
_________________________________________________________________
dropout_24 (Dropout)         (None, 100)               0
_________________________________________________________________
dense_23 (Dense)             (None, 100)               10100
_________________________________________________________________
dense_24 (Dense)             (None, 6)                 606
=================================================================
Total params: 423,650
Trainable params: 423,650
Non-trainable params: 0
_________________________________________________________________
None
```

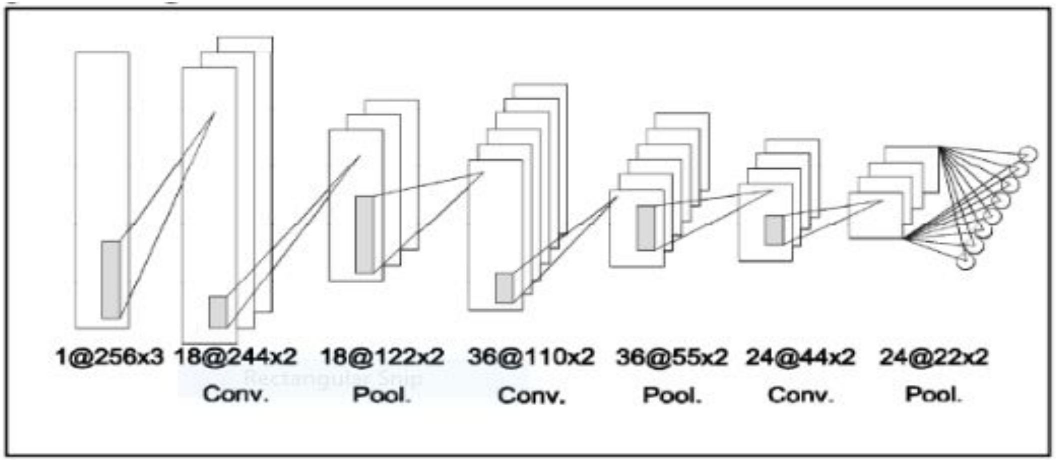
Result of ConvLSTM

```
Using TensorFlow backend.
(7352, 128, 9) (7352, 1)
(2947, 128, 9) (2947, 1)
(7352, 128, 9) (7352, 6) (2947, 128, 9) (2947, 6)
2947/2947 [==============================] - 4s 2ms/step
>#1: 88.768
2947/2947 [==============================] - 8s 3ms/step
>#2: 91.381
2947/2947 [==============================] - 8s 3ms/step
>#3: 90.567
2947/2947 [==============================] - 8s 3ms/step
>#4: 88.768
2947/2947 [==============================] - 8s 3ms/step
>#5: 90.872
2947/2947 [==============================] - 9s 3ms/step
>#6: 90.261
2947/2947 [==============================] - 8s 3ms/step
>#7: 89.413
2947/2947 [==============================] - 9s 3ms/step
>#8: 89.617
2947/2947 [==============================] - 9s 3ms/step
>#9: 91.279
2947/2947 [==============================] - 9s 3ms/step
>#10: 89.786
[88.76823888700373, 91.38106549032915, 90.56667797760434, 88.76823888700373, 90.87207329487615, 90.2612826603254, 89.41296233
457754, 89.61655921275874, 91.27926705123855, 89.78622327790974]
Accuracy: 90.071% (+/-0.906)
```

Assignment 2

A Deep Learning Approach to Human Activity Recognition Based on Single Accelerometer

Dataset: The recorded raw 3D acceleration signal stream is cropped into the same size with an overlap of 50%.
Every sample is a matrix with the size of 256x3.



Result

```
Epoch:98,batch:3216,loss:0.00000000,accuracy:1.00000000
Epoch:98,batch:4816,loss:0.00000000,accuracy:1.00000000
Epoch:98,batch:6416,loss:0.00000000,accuracy:1.00000000
Epoch:99,batch:016,loss:0.00002368,accuracy:1.00000000
Epoch:99,batch:1616,loss:0.00000000,accuracy:1.00000000
Epoch:99,batch:3216,loss:0.00000000,accuracy:1.00000000
Epoch:99,batch:4816,loss:0.00000000,accuracy:1.00000000
Epoch:99,batch:6416,loss:0.00000000,accuracy:1.00000000
Epoch:100,batch:016,loss:2.61394072,accuracy:0.93750000
Epoch:100,batch:1616,loss:0.00000000,accuracy:1.00000000
Epoch:100,batch:3216,loss:0.00000000,accuracy:1.00000000
Epoch:100,batch:4816,loss:0.00000000,accuracy:1.00000000
Epoch:100,batch:6416,loss:0.00000000,accuracy:1.00000000
Optimization finished!
Accuracy of testing:0.87987787
```

Result


Training and Test Accuracy
1.0
0.9
0.8
0.7
0.6
0.5
Loss
0
20
40
60
80
100
Epochs
Training Accuracy
Test Accuracy


Training and Test loss
200000
150000
100000
50000
0
Loss
0
20
40
60
80
100
Epochs
Training loss
Test loss

References

[1] Wang, Jindong, et al. "Deep learning for sensor-based activity recognition: A survey." Pattern Recognition Letters (2018).

[2] Ordóñez, Francisco Javier, and Daniel Roggen. "Deep convolutional and lstm recurrent neural networks for multimodal wearable activity recognition." Sensors 16.1 (2016): 115.

[3] Jiang W, Yin Z Proceedings of the 23rd ACM international conference on Multimedia. Human Activity Recognition using Wearable Sensors by Deep Convolutional Neural Networks[C] ACM, 2015:1307-1310.

[4] Yuwen Chen, Kunhua Zhong, Ju Zhang, Qilong Sun and Xueliang Zhao, “LSTM Networks for Mobile Human Activity Recognition” in International Conference on Artificial Intelligence: Technologies and Applications (ICAITA 2016).

[5] Hammerla, Nils Y., Shane Halloran, and Thomas Ploetz. "Deep, convolutional, and recurrent models for human activity recognition using wearables." arXiv preprint arXiv:1604.08880(2016).

[6] Simonyan, Karen, and Andrew Zisserman. "Two-stream convolutional networks for action recognition in videos." *Advances in neural information processing systems*. 2014.

Thank you!