Skeleton-based Activity Recognition using Kinect v2

Anonymous

ABSTRACT

Current Activity Recognition systems mostly use Convolutional Neural Networks (CNN) for training purposes. Using a CNN is the gold standard for performing such tasks. But the downside to it is that even a simple CNN architecture like VGG16 would take up memory space anywhere between 800MB to 1GB. In our approach we make use of the Kinect v2 to provide us the joint points and based on these joint points we trained a Multi Layer Perceptron (MLP) to detect the activity performed. Moreover the trained model is only 15MB in size and is very much feasible to be deployed on systems with memory constraints, for instance pi zero.

1. INTRODUCTION

Activity recognition is an important and challenging problem, with applications in surveillance, where multiple cameras and sensors can be placed to track human activities and detect any abnormalities based on a normal model given to the system. A summary of recent trends in the research in abnormal activity detection[4] shows how this are has been a lot under focus since last 10-15 years and a lot of tools and concepts were applied for the modeling. Other applications include anti-crimes, logging of activities, understanding online videos for captioning, etc. Smart environments are being developed to help people everyday. Such an environment should be able to locate and track a person, their daily activities, emergencies, the limbs and other objects that the person interacts with. In computer vision based activity recognition, several approaches have been considered in the research like Optical flow, Kalman filtering, Hidden Markov Models, etc and several modalities like RGB, skeleton, Depth, infrared, etc.

In this paper, we aim to extract skeleton from the Kinect version 2 camera and recognize the activity performed with the help of a model trained on the KARD dataset. Some research about human activity recognition is discussed in Section2. The section3 explains the methodology we used. Sections4 and 5 are to mention the major learnings from this project and the future work.

2. RELATED WORK

A lot of research work has been done since more than a decade now. A comprehensive survey about the recent techniques[1] gives a system design that is common in most of them. The basic blocks of any activity recognition system can be seen as fig1. The first block is that a system will take inputs from sensors or cameras and it can be a still image or a sequence of images. The segmentation is difficult in moving objects and also has to be adjusted according to the moving camera in addition to the object moving itself. Feature extraction is getting desired properties from the images eg. colour. There could be several types of tracking including feature-based tracking, model-based tracking, and optical flow-based tracking. Multiple datasets are mentioned in the survey which are being used since 2001 till date. With time, the datasets have begun to be robust to occlusion and increase in number of types of actions. Tanakon et al. [6] present a comparison of models for classification of 10 basic activities. The different models compared are : Neural Network, SVM, Naive Bayes and Decision Trees. All of the models require a labelled data for training and testing. Datasets from 5 different sources are used including KARD, MSR3D, etc. out of which the KARD dataset performed the best with all the models and Neural network worked the best with all the datasets. Although this was a common result, the model did not perform very well with similar gesture activities, i.e. 'thai greeting' and 'bow'.

Vision based Human Activity recognition is taking spatial and temporal information from videos that is useful for interpreting the scene and understanding the activity[5]. Motion and shape features are extracted and several machine learning models are trained including SVM, K-nearest neighbours, linear discriminant classifier, etc. The classifiers are tested with and without normalization. The Naive Bayes performed best with un-normalized data. When the complexity was increased by changing the camera position or light, the KNN worked the best. Normalization increased accuracies of all the models. This comparison shows how the models react to different classes(activites) and complexities involved in the environment.

3. APPROACH

3.1 Dataset

The dataset used is the KARD[3] dataset i.e. Kinect Activity Recognition Dataset. This dataset consists of 18 activ-

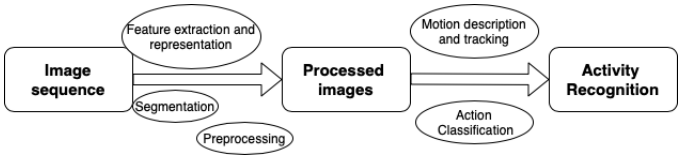


Figure 1: Human Activity Recognition basic system

Figure 2: Kinect v2

ities, performed 3 times by 10 different people. Each of these has 4 different types of files - depth map, RGB video, skeleton real world co-ordinates and skeleton screen co-ordinates.

3.1.1 Activities:

List of activites : [*Horizontal arm wave, High arm wave, Two hand wave, Catch Cap, High throw, Draw X, Draw Tick, Toss Paper, Forward Kick, Side Kick, Take Umbrella, Bend, Hand Clap, Walk, Phone Call, Drink, Sit down, Stand up]*

3.1.2 Data pre-processing:

There are total 2160 files, 4 for each instance of an activity. All the instances by multiple subjects are merged together to have same activity in a file. All these points were normalized for better loss convergence.

3.2 Data Acquisition

Camera used : Kinect v2[2]2. Kinect is a line of motion sensing input devices used as hands-free interface with the machine. Originally developed as a Xbox add-on, kinect later found role in academia, healthcare, etc. The microsoft kinect v2 camera tracks a skeleton which has upto 25 joints, numbered 0 to 24. Each of these joints has 11 properties : color (x, y), depth (x, y), camera (x, y, z), orientation(x, y, z, w).

3.3 Methodology

3.3.1 Kinect Data Processing:

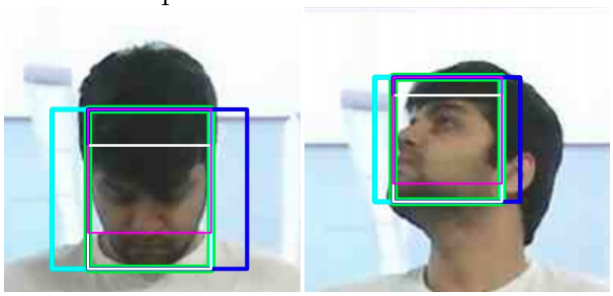
Figure 3: Skeleton from Kinect

The skeleton data provided by the kinect had curvature on both the axis. The raw disparity image provided by the kinect was accurate. (Checked through various experiments on flat surfaces). The raw disparity image is provided by the Kinect in pixel coordinates while the skeleton data provided by the Kinect are in world or object coordinate system. To map the depth from image coordinate to world / object coordinate a planar homography was computed. After this mapping was successfully done, the skeleton data had no more curvature. The skeleton obtained can be seen in the fig.3

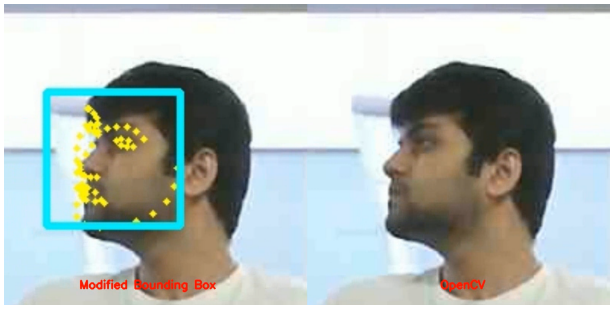
3.3.2 Head Pose Estimation:

The skeleton data provides only one joint for the head and this results in loss of information for different activities for instance shaking your head. To get the head pose we acquired the facial landmarks using the Dlib face predictor. OpenCV also has a functionality but is not as accurate as dlib. Both OpenCV and Dlib provide 68 facial landmarks for the face.

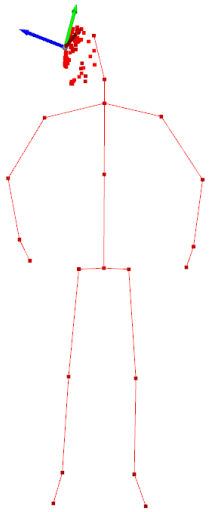
To obtain accurate facial landmarks, face detections needs to be performed first. Most of the face detection algorithms that are readily available fail to detect poses at extreme angles (Angles approx greater than 70 degrees). All these face detectors work only on the RGB stream but since we did have access to depth image as well, we modified the face detection algorithm more precisely we modified the Single Shot Multi-Box Detector to detect faces even at extreme poses of head.



Four possible bounding boxes were predicted. To choose the optimal bounding box certain conditions were imposed on the corresponding depth image.

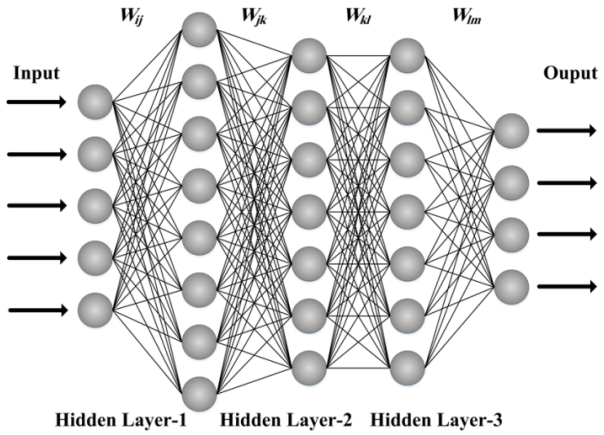


Now that we had 3D points (facial landmarks) along with the 2D point correspondences we were able to obtain a rotation vector and translation vector using the PnP algorithm, precisely the P6P algorithm. The rotation vector and translation vector were then converted to Euler angles to get the roll, pitch and yaw.

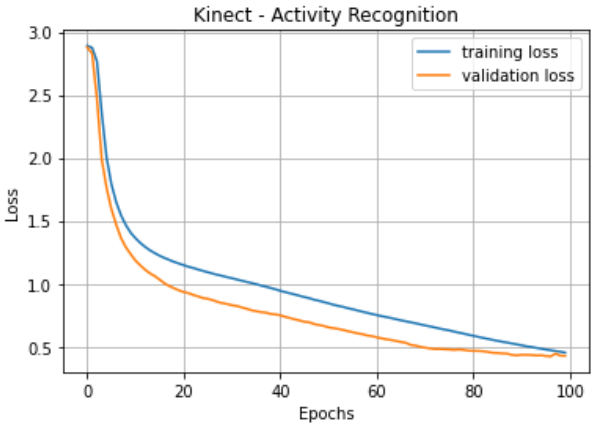


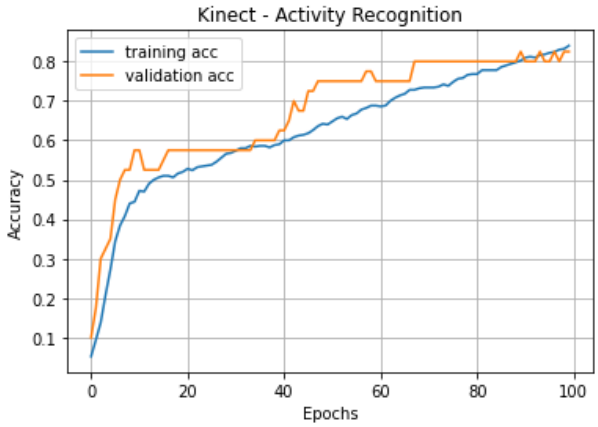
3.3.3 Training:

Now that we had all the data we needed, we built a MLP with with 6 hidden layers to classify 18 activities.



The input was the joint position in the camera space for 50 frames, i.e.: 15 joints per frame for 50 frames. Each joint provided 3 coordinates. Therefore the first layer was a 1-Dimensional long vector of size 15x3x50 which is equal to 2250 input neurons.
The model was trained for 100 epochs with a learning rate of 0.0001. Cross Entropy loss was used as the cost function and the optimizer chosen was Adam. To compensate for overfitting a L2 penalty of 0.001 was imposed and dropouts with probability of 0.5 and 0.3 were added at hidden layers 5 and 6 respectively.
The MSE error after 100 epochs was 0.2 and 0.3 for training and testing respectively. Testing accuracy of 85% was achieved.





3.3.4 Results:

A video file containing some of the activities has been attached.

4. LESSONS LEARNED

This project helped us learn about multiple state of the art activity recognition techniques and approaches from the literature survey. Microsoft kinect v2 is used for taking the input. Learned about v2 and how it works to give a skeleton or a depth map. Choosing a dataset among several available datasets online was a tough task. Several requests had to be made for accessing the datasets. Some datasets have labels while the datasets like MSR3D dataset have only the videos. Found the appropriate dataset and prepared it by merging the same acitivity files together to create the label for the particular activity. Neural network is then trained using the pyTorch framework to recognize the activity in real time.

5. CURRENT STATUS & FUTURE WORK

Kinect v2 gives a skeleton which is used to recognize the activity perfomed infront of the

camera. eg. Arm wave, side kick, sit down, etc. Neural network does the recognition with 85% accuracy. Several other activities can be included in the dataset to increase the scope. Addition of depth and RGB features might improve the recognition.

6. REFERENCES

[1] G. S. Eisa Jafari Amirbandi. Exploring methods and systems for vision based human activity recognition. *1st Conference on Swarm Intelligence and Evolutionary Computation (CSIEC2016), Higher Education Complex of Bam, Iran,*, 2016.

[2] L. Jamhoury. Understanding kinect v2 joints and coordinate system. July 2018.

[3] G. G. S. Morana, Marco; Lo Re. Kinect activity recognition dataset", mendeley data, v1. May 2017.

[4] K. W. Oluwatoyin P. Popoola. Video-based abnormal human behavior recognition—a review. *IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS—PART C: APPLICATIONS AND REVIEWS*, 42, November 2012.

[5] M. M. S. N. R. J. I. Sumaira Ghazal, Umar S. Khan. Human activity recognition using 2d skeleton data and supervised machine learning. *IET Image Processing*, 13, November 2019.

[6] P. S. Tanakon Sawanglok, Tananya Thampairoj. Activity recognition using kinect and comparison of supervised learning models for activity classification. 2018.