Automated Assessment System for Embodied Cognition in Children *

Ashish Jaiswal
ashish.jaiswal@mavs.uta.edu

Lalit
lalitjayavantt.goski@mavs.uta.edu

March 27, 2020

Abstract

The main idea of this project is to automate computer vision system in understanding cognitive abilities in children. We are working on building an automated assessment system for Activate Test for Embodied Cognition (ATEC)[1], a test which measures cognitive skills through different physical tasks. More specifically, we focus on the Ball-Drop-to-the-Beat task and implement vision and deep learning techniques to identify if a child is correctly performing the task or not. The ground truth to test our models are manual annotations for video segments. The main goal is to classify which of the given tasks is performed by a child. Current progress includes pre-processing of data output from Kinect sensor and making it ready for our deep learning model. The model contains a one-layer LSTM network on top of a 3-layered CNN network. Our current best accuracy for the model is around 82% for the detection of the task being performed in a video segment.

1 Introduction

There are different ways to assess cognitive abilities in children and one of them is to make use of executive functions for analysing if a child has any kind of Attention-Deficit/Hyperactivity Disorder (ADHD). Executive functions [2] are higher-order cognitive processes involved in multitasking, time management, attention, planning, inhibition, self-regulation, and memory. Children with ADHD exhibit weaknesses in executive functions, specifically response inhibition, planning, vigilance, and working memory.

We already have the visual data collected from different children using a Kinect Sensor. The main idea is to implement computer vision and deep learning methodologies on the collected data to build an automated system that classifies the correctness of a task performed by a child. We are using extracted RGB frames from each video segment or clip to identify the type of task being performed in the video. To achieve activity recognition in a video, we make use of Recurrent Neural Networks (RNN) [5], specifically Long Short-Term Memory (LSTM) [3] network. The data is collected by placing a camera in front of a child performing certain tasks. The task we are interested in is called the Ball-Drop-to-the-Beat[1] task. Here, the child has to perform one of the three different movements: Pass the ball, Raise the hand with the ball, and Do nothing. We plan to identify these movements by training a classifier model that classifies a video segment to one of these classes.

*Progress Report for Computer Vision project - 6367

Our initial focus has been in training CNN models on top of an LSTM layer and also applying different image processing techniques to enhance the inputs to our model. The other techniques that can be implemented for enhancing results would be pose estimation [7] of children and using body key points to track their movements. By doing this, we can trace the ball or wrist joints of children to see if they performed the task properly or not.

2 Methodology

Our workflow is divided into two parts. In the first part, images are extracted from video segments and sent to a pose estimation network to extract body keypoints of the people present in a video. We are using AlphaPose, a library for the network Poseflow [6] which efficiently tracks and extracts body keypoints from a video in real time. For now, our system is not real time as we have static images as output from Kinect sensor. After the body keypoints are extracted for all the people present in the video, we select the person who is closest to the camera sensor by using a flag output by the poseflow network. The output keypoints include coordinates of 17 body points in a person but not all of them are important for our analysis. So, we select only 9 of those keypoints which highly contribute in tracking the motion of the child in the video. They 9 keypoints are arranged in the dimension (nx9x2) where 'n' refers to the number of frames we consider to utilize from a video segment. Each keypoint contains value (x, y) which represent the pixel coordinate for that point in the image. The keypoints are normalized (mean=0 & std=1) based on their mean and standard deviation as it makes it easier for the model to learn any kind of relationship between the keypoints in corresponding frames. These keypoints are stored in JSON (JavaScript Object Notation) format which are later used as inputs to our deep learning model.

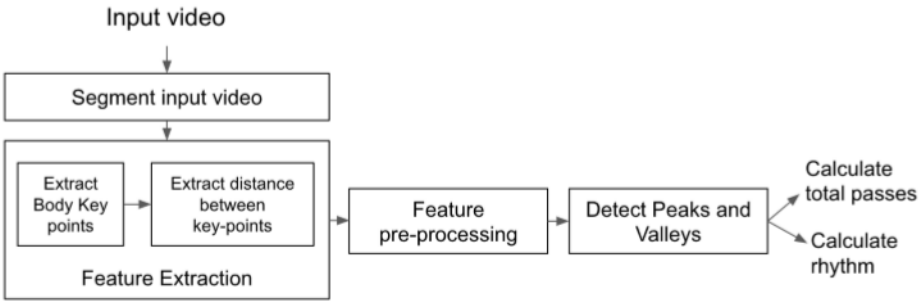

Figure 1. Pipeline of the model.

On the other hand, we have built a deep neural network model consisting of a 3-layered CNN and a 1-layer LSTM. The CNN network expects an input of (n, 1, 9, 2) where 'n' is the number of frames in a segment, '1' is the number of channels, (9, 2) is the size of the key-points extracted. The purpose of CNN is to learn important features from the keypoints provided and from those extracted important features, the LSTM layer tries to learn the relationship between the frames of the segment. Once the network learns the necessary

```
In [57]:
from sklearn.metrics import confusion_matrix

confusion_matrix(actual, preds, labels=list(range(3)))

Out[57]: array([[199,   3,   4],
                [ 24,  90,   5],
                [  4,  11,  54]])

In [58]:
from sklearn.metrics import confusion_matrix

confusion_matrix(r_actual, r_preds, labels=list(range(2)))

Out[58]: array([[ 34,  56],
                [ 46, 258]])
```

Figure 2. Confusion Matrix

relationship, it is able to classify the type of task being performed in the video. While CNNs are very good in reducing the input vectors into meaningful features, LSTM networks identify patterns in sequence by remembering past inputs. This way we train the model to on the obtained body keypoints.

3 Results

The experiment: For the ball drop task, there are three main events involved: ball pass, no ball pass and hand raise. The subject in the videos, is asked to perform one of the above mentioned actions. A ball pass event is considered when the subject moves the hand holding the ball towards their other hand, makes the transfer and moves back to the actual position. In such a scenario,the distance between the wrist points decreases until the transfer happens and distance increases again. Likewise, a hand raise event is accounted when the subject moves the hand holding the ball towards the shoulder of the same hand and withdraws back to the actual position where the wrist and shoulder joint distance initially increases and starts to decrease while retreating. (Figure 1) [1] A peak is formed every time such an event occurs when the hands are closest together. After processing the segmented features obtained from the CNN, the aim is to detect peaks and valleys in the segment. Mathematically, peaks and valleys represent local maxima and minima.

```
preds, actual, acc, r_preds, r_actual = test(net)

Accuracy: 82.05583756345177
Rythm Accuracy: 74.11167512690355
```

Figure 3. Accuracy of the model.

We have trained a Convolutional Neural Network(CNN) to perform the pose estimation and an LSTM to identify the type of events. The classification accuracy of the events obtained is about 82% (Figure 3). Plus, we also estimate the rhythm for every identified event using the LSTM. An event is in rhythm if the peak is within an annotated time frame, determined by task. We have obtained the rhythm accuracy of about 74%. The Figure 2 represents the confusion matrices of the classification model and rhythm validation set respectively.

4 Discussion and Conclusions

There is still some work to be done to improve the performance of the current model. One way would be to deal with the biasedness in the data that we have. One class contains greater number of data samples than the other classes. For countering this issue, we can assign weighted loss functions. Also, including more data can crank up the accuracy of the model. Further work can be done in learning better features from the actual images using self-supervised methods like RGB-Flow correspondence. Similar work can be found in this paper [4]. Also, there are annotations for the rhythm scores that defines the correctness of the task performed by the child. In future work, we can include another loss function to determine the rhythm score too.

5 Contributions

Ashish: Worked in building the Convolution-LSTM model. Also, extracted keypoints by setting up the AlphaPose library. Researched in incorporating the original RGB images to a separate model along with the current body keypoints approach to enhance the performance of the network.

Lalit: Extracted RGB frames from the available videos. Applied transformations and augmentation along with necessary image processing techniques to the data in order to make it better for training the model. Researched in understanding important body keypoints features (out of the 17) for faster convergence of the model.

References

[1] Alex Dillhoff, Konstantinos Tsiakas, Ashwin ramesh babu, Mohammad Zakizadehghariehali, Benjamin Buchanan, Morris Bell, Vassilis Athitsos, and Fillia Makedon. An automated assessment system for emxbodied cognition in children: from motion data to executive functioning. pages 1–6, 09 2019.

[2] Joel T Nigg Stephen V Faraone Erik G Willcutt, Alysa E Doyle and Bruce F Pennington. Validity of the executive function theory of attention-deficit/hyperactivity disorder: a meta-analytic review. pages 1336–1346, 11 2005.

[3] Sepp Hochreiter and Jürgen Schmidhuber. Long short-term memory. *Neural Computation*, 9(8):1735–1780, 1997.

[4] Z. Lai and W. Xie. Self-supervised learning for video correspondence flow. In *BMVC*, 2019.

[5] Bastiaan Quast. rnn: a recurrent neural network in r. *Working Papers*, 2016.

[6] Yuliang Xiu, Jiefeng Li, Haoyu Wang, Yinghong Fang, and Cewu Lu. Pose Flow: Efficient online pose tracking. In *BMVC*, 2018.

[7] Tomas Simon Shih-En Wei Zhe Cao, Gines Hidalgo and Yaser Sheikh. Openpose: realtime multi-person 2d pose estimation using part affinity fields. 2018.