RESEARCH PROPOSAL

Workout Recognition Using 2D and 3D CNNs

**Author**

James Peralta

BSc Computer Science

Department of Computer Science

**Supervisor**

Dr. Jeffrey Boyd

Video and Motion Analysis Laboratory

Department of Computer Science

**Date**

September 27, 2019

# Abstract

fsdfd

# 1 Introduction

In 2017, nearly 50% of deaths in Canada were caused by cancer and heart disease[Statistics2017]. In fact, 44% of adults over the age of 20 have at least 1 of 10 common chronic diseases[Government2019]. To help reduce these chronic diseases, we must invest in preventive measures such as empowering our society into a healthy lifestyle. Physical activity has proven to be effective in the primary and secondary prevention of a variety of chronic diseases including cancer and cardiovascular disease[Warburton2006]. However, creating the habits to routinely engage in physical activity is a long-term process that requires a lot of personal motivation. Only 20% of people succeed in long-term weight loss maintenance [Wing2005].

A promising area of technology - persuasive technology, is designed to influence and change human behaviors. Emerging persuasive technology in the fitness sector has already shown to improve the health and fitness practices of its users [Ahtinen2009]. Within the space of persuasive technology, activity tracking is one of the most prevalent strategies. Users of activity trackers that monitor their workout progress commented on how this technology had motivated them and helped them make durable changes in their lifestyle [Fritz2014]. Activity Trackers can be used for more simple data, such as counting steps but fail to capture advanced data, such as repetitions of workouts. In this paper, we focus on a specific type of activity tracking, workout logging.

Although workout loggers offer plenty of value to their users, they have shown very poor retention rates. [Ledger2016] found that within the first two weeks, 62% of users that downloaded a workout logging app stopped using it. This was because manual entry of data can be too repetitive and tedious, eventually leading to these technologies to go unused [Ahtinen2009]. There have been attempts at removing the step of manually entering data through wearables [Morris2014, Muehlbauer2011, Ordonez2016], computer vision[Khurana2018], and adding sensors onto equipment [Ding2015].

Computer vision is a promising solution because it can capture a richer set of spatial features. Wearables lose accuracy depending on where the sensor is placed. For example, a sensor placed on the wrist will have trouble detecting certain leg exercises such as a leg press because the hand is stationary during this exercise[Find this]. Furthermore, adding sensors to equipment will not be able to track bodyweight exercises such as pushups where the user is not using any equipment. Not only can computer vision avoid these pitfalls, it is also the only solution that can be implemented as a single-point solution, which will avoid instrumenting many users or many machines.

# 2 Related Work

## 2.1 Using optical flow

The current state of the art for this problem is Gymcam[Khurana2018]. GymCam used optical flow to generate a feature vector of 27 features. This feature vector was passed along a pipeline of three separate multilayer perceptrons. With this procedure they were able to obtain an accuracy of 84.6% for exercise segmentation, 93.6% exercise recognition and count the number of repetitions within +/-1.7 on average. Other related contributions were for similar video analysis tasks; video classification and human gesture recognition.

## 2.2 Feature based versus neural networks

[Patsadu2012] evaluated feature-based and neural network classifiers for human gesture recognition. In their study, they used a Kinect camera to extract a body-joint position vector containing the locations of 20 joints including the head, hands, and feet. Using this vector as input into a classifier, they classified between three actions; standing, sitting down, and laying down. They observed the performance of multiple classifiers such as Back Propagation Neural Networks, SVMs, Decision Trees, and Naive Bayes. Out of every classifier, neural networks performed the best.

## 2.3 Using 2D Convolutional Neural Networks

[Karpathy2019] surveyed the performance of two dimensional convolutional neural networks (2D ConvNets) for video classification. 2D ConvNets avoid the feature extraction phase required by Gymcam and feeds raw frames as input into the network. This simplified the pipeline by removing the need to generate custom features and was shown to have significant performance gains over the feature-based approaches. Since 2D ConvNets take in a single frame as input, it can only encode spatial features. This can be a problem due to the temporal nature of videos. As part of their contribution, they proposed three ways of encoding this temporal information using early fusion, late fusion and slow fusion architectures (see figure 1).

## 2.4 Using 3D Convolutional Neural Networks

Facebook introduced a three dimensional Convolutional Neural Network they named, C3D [Tran2014]. The 3D ConvNet takes videos as input and encodes both spatial and temporal information without the need to perform workarounds such as early fusion. The difference between a 2D ConvNet and a 3D ConvNet is that the convolution operations use two dimensional and three dimensional kernels respectively (see figure 2). They went on to show how 3D ConvNets were better than 2D ConvNets at preserving temporal information and outperformed 2D Convnets on various video analysis tasks. More interestingly, 3D Convnets we’re shown to run 2x faster than optical flow solutions.

# 3 Proposed Method

Our study will involve three phases; data collection, data labelling and building a classifier. We will use ConvNets as our classifier due to the improvements in speed and performance over feature-based approaches. Our goal is to establish a baseline for neural networks for this task, analyze the benefits of using neural networks, and report the pitfalls of using neural networks.

## 3.1 Data collection

For data collection, we have arranged to book an athlete gym at the University of Calgary. We will strategically place cameras around the gym to capture each machine. Participants will be invited for a 30-90 minute workout session where they will perform 6 different exercises of their choice. Each camera will only store footage when it detects a participant in the field of view. This will create clips that start when a participant enters and ends when there is no longer any participants in the view. 60 participants, performing 3 sets per exercise will leave us with 1080 exercises captured on video.

## 3.2 Data labelling

After building up the dataset of exercises, we will have to label them before training our Neural Network. We will use GymCam’s annotation tool which is optimized for labelling workouts and their repetitions. For each clip, we will draw a bounding box around a user when they begin performing a workout and remove it when they finish the set. Along with this bounding box we will label the exercise they performed along with the number of repetitions. This will create a dataset that can be used for exercise segmentation, exercise localization, exercise recognition and repetition counting.

## 3.3 Building a classifier

With this dataset we can now design a Neural Network that will locate the user, recognize the workout, and count repetitions. To perform these tasks we will create a pipeline of ConvNets as follows. The raw frame is passed into an OpenPose network to detect humans in the field of view. Once a human is located we will crop a rectangle around them to focus in on the user. This frame will be fed into a Neural Network that will predict if the person in the frame is performing an exercise. If the person is performing an exercise, we will begin creating a clip using all subsequent frames until the user stops performing the exercise. Lastly, we will input this clip into a classifier that will predict the workout being performed and the number of repetitions. Each classifier mentioned above will either be a 2D or 3D ConvNet.

## 3.4 Evaluating the classifier

Similar to [Khurana2018], we will separately evaluate the performance of exercise recognition and repetition counting. The performance of a predictor can be measured in using metrics of accuracy, precision, and recall. There is often a trade-off between optimizing recall versus precision and in our case, we will optimize the recall metric. To generate these metrics with the least amount of variance and bias[Kohavi1995], we will perform stratified ten-fold cross validation on our collected dataset.

# 4 Current Progress

We have developed the first stage of the data processing pipeline by creating the software that streams frames from a wired camera (Logitech C270) or a network camera (Ezviz Mini 0) into a server. To understand how we will use CNNs for this problem, we have experimented with 2D CNNs, OpenPose, and YoloV3 using a dataset of handpicked YouTube workout videos. We have also submitted an ethics review to the University of Calgary Conjoint Faculties Research Ethics Board (CFREB). Once our study is approved by the CFREB, we can begin phase 1 of our expected deliverables.

# 5 Expected Deliverables

## Phase 1: 5 weeks

* Set up the infrastructure in the gym: 1 week
* Invite participants into the gym to collect data: 4 weeks

## Phase 2: 2 weeks

* Label the data collected: 1.5 weeks
* Aggregate visual and descriptive statistics of the dataset: 0.5 weeks

## Phase 3: 11 weeks

* Experiment with OpenPose and YoloV3 for localizing users: 2 weeks
* Develop, evaluate, and reiterate on a 2D CNN: 6 weeks
* Experiment transfer learning using a pre-trained 2D CNN: 3 weeks

## Phase 4: 11 weeks

* Develop, train, evaluate, and reiterate on a 3D CNN: 8 weeks
* Experiment using transfer learning on a pre-trained 3D CNN: 3 weeks

## Total: 29 weeks