DeepSquat: Analyzing Weightlifting Form with Deep Learning

Weixiong Zheng (zhengwx), Peicun Jiang (pcjiang)

GitHub url: https://github.com/maxwxzheng/deep-squat

Abstract

Personal Training is an industry that has seen a significant demand boost in the past decade.[1] This growth is fueled by increase obesity trends, yet relative poverty—which correlates with obesity[2]—has precluded many from personal training. This project uses deep learning to conduct a personal trainer's job of analyzing a person's squat form. Using self-collected video data which we preprocessed and augmented, we built a model comprising 4 Conv-Conv-MaxPool-Dropout layers and trained it with 500 epochs, ultimately achieving a training accuracy of 98.36%, an evaluation accuracy of 88.18%, and a test accuracy of 91.59%. With a greater quantity and quality of data, this project can be expanded to more exercises so that personal training can be made more accessible to all.

Introduction

Personal training is a $10 billion industry. According to research by IBISWorld, over the five years to 2018, revenue for the Personal Trainers industry is estimated to have increased at an annualized rate of 1.9% to $9.1 billion, including a 2.7% increase in 2018 alone.[3] That said, growth of the industry is largely fueled by an underlying worsening obesity trend. According to the Centers for Disease Control and Prevention, more than one third of Americans are obese. Consequently, there is rising need for weight-loss services and greater interest in customized workout regimes, increasing demand for the Personal Trainers industry. However, due to the high prices of personal trainer, populations such as students, and mid- to low-income earners are seldom able to afford personal training, which partly explains the correlation between poverty and obesity.[4] Faced with a phenomenon as such, we believe the task can be done by a deep learning model so that health and fitness training can be made more accessible to less financially well-off individuals.

A major part of a personal trainer's work is to help the client keep good form during a workout. Thus, in this project, we will apply deep learning to analyze images taken during a workout in order to tell if the subject's form is good or not. As for the exercise, we picked squats as the target of this analysis for two reasons. Firstly, conditioning specialists universally agree the squat is among the top three prescribed exercises for sports training, rehabilitation and pre-habilitation. Secondly, considering the scope and timeline of the class, it is the most reasonable exercise for the architecture we chose.

---

[1] Fernandez, 2018
[2] Daniels et al., 2007
[3] *Ibid.*
[4] *Ibid.*

We tested different CNN architectures with different hyperparameters to build a model to identify if a squat is of good form or bad form. The input to our model is a pose-only image extracted from an image of a person doing squat. The output of our model is 1 or 0 representing if the squat's form is good or not.

Dataset and Features

Our dataset contains 9424 pose images (resolution: 108×192) of subject doing squats. Our training set contains 8544 (90%) images, while validation and test sets each has 440 (5%).

I. Data Collection

We recorded 34 videos with resolution 1080×1920 of subjects doing 219 squats in total. The subjects are 19-24 years of age and are all male.

18-19	20-21	22-24
23%	32%	45%

Table 1 Age Breakdown

Each video includes 5 to 10 squats, some with good form and some with common mistakes in form. The mistakes in form include subject losing balance to one side, bending their back, over-extending their knees etc. To reduce noise in the data, we kept the background consistent and relatively neutral.

From the videos we manually located and labeled the frame of subject at full-squat. A subject is at full-squat when the subject reaches the deepest position in the squat. About half of the images are labeled positive (good form) and half of them are labeled negative (bad form). Below are two sample images we extracted.

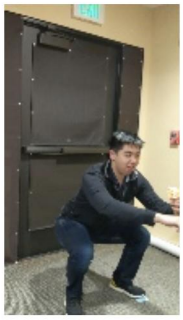

Figure 1. Positive Image

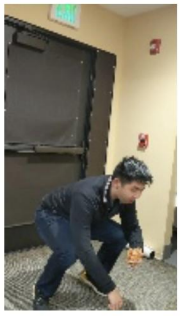

Figure 2. Negative Image

From each squat video, we extracted up to 11 images in total, including up to 5 frames prior to the full-squat frame, the full-squat frame, and up to 5 frames post the full-squat frame. In total, we extracted 2356 squat images from all videos.

II. Data Preprocessing

We performed the following steps for preprocessing the 2356 squat images we extracted from the videos.

1. Resize the images to 108×192.
2. Use tf_pose library to extract pose to a new frame.
3. Apply horizontal flip data augmentation.
4. Apply zoom out data augmentation

Figure 3. Original Image

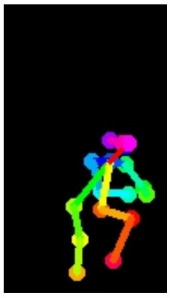

Figure 4. Original Pose

Figure 5. Zoomed out

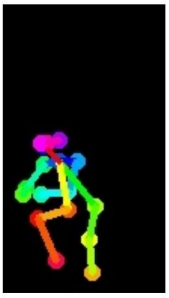

Figure 6. Flipped

Figure 7. Flipped & Zoomed Out

III. Data Shuffling

Since each squat generated roughly 11×4 = 44 images, if we shuffled the images randomly, there is an extremely high chance that the training set (90%) will include images from all squats. Thus we decided to shuffle the data by squats and labels. The validation set and the test set each included all images generated from 10 squats. The training set included all images generated from the rest 199 squats.

Methods

Our best model uses the following architecture. We used binary cross-entropy as the loss function. We used loss on the evaluation set as the main evaluation metric. The model has 4 Conv-Conv-MaxPool-DropOut layers. In the end there are 3 fully connected layers.

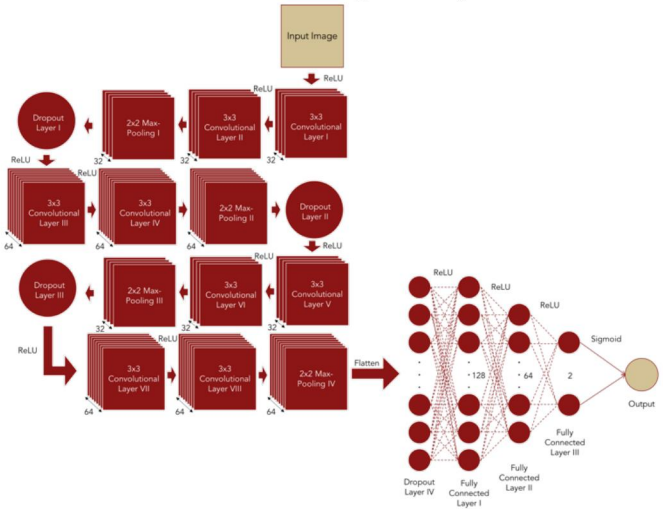

Figure 8. Model Architecture

Experiments

We first tested different architectures. The main difference between different architectures is the number of Conv-Conv-MaxPool-Dropout layers in the model. All the architectures we tested used the same fully-connected layers at the end. For all the tests, we used learning rate 0.00001 and ran the training for 50 epochs. We also tested different parameters for the L2 regularization. Note that all of our models are trained on an AWS EC2 p3.2xlarge instance with one GPU.

Number of Conv-Conv-MaxPool-Dropout layers	L2 regularization parameter					
	0.01		0.001		0.0001	
	Train Loss	Eval Loss	Train Loss	Eval Loss	Train Loss	Eval Loss
1	2.21	3.54	0.31	2.01	0.04	1.36
2	3.62	4.75	0.46	1.38	0.07	1.12
3	5.25	5.38	0.85	0.94	0.26	0.4
4	5.10	5.25	0.82	0.95	0.35	0.49

Table 2. Experiments with Models

According to the experiment, when the model gets larger, it needs more iterations to achieve the same training loss as the smaller models. However, the evaluation loss is closer to the training loss for bigger models. Also 0.0001 as the L2 regularization parameter generated the best results for all models.

We also ran an experiment to test how different kernel size in the Convolutional layer impacts the model performance. For the experiment we used 3 Conv-Conv-MaxPool-Dropout layers. Learning rate is 0.00001. We ran each model for 500 epochs.

Kernel Size in Convolutional Layer	L2 regularization parameter					
	0.0001		0.00001		0.000001	
	Train Loss	Eval Loss	Train Loss	Eval Loss	Train Loss	Eval Loss
2 * 2	0.05	1.91	0.01	1.82	0.002	1.84
3 * 3	0.05	2.00	0.007	1.94	0.002	1.97
4 * 4	0.05	1.83	0.008	1.83	0.003	1.97

Table 3. Experiments with Kernel Sizes

According to the experiment, kernel size in the convolutional layer doesn't have significant impact on model performance.

Results

For the final model, we decided to use 4 Conv-Conv-MaxPool-Dropout layers; 0.000001 as the L2 regularization parameter; kernel size 3 * 3; learning rate 0.00001. We trained the model with 500 epochs.

Training Loss	Training Accuracy	Eval Loss	Eval Accuracy	Test Loss	Test Accuracy
0.05	98.36%	0.48	88.18%	0.32	91.59%

Table 4. Resulting Accuracies

	Predict 1	Predict 0
Label 1	400	40
Label 0	428	12

Table 5. Analysis on Dev. Set

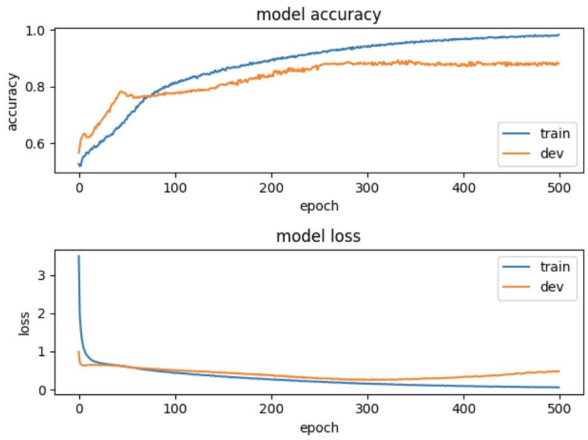

Figure 9. Accuracy and Loss

Discussion

From Table 5, we can see that the model is very good at recognizing bad forms and is stricter on recognizing good forms. One limitation of our project is the quality and quantity of data. In terms of quality, some of our negative data veer towards the extremes of bad form, whereas in reality, people's squat forms are more nuanced. Moreover, we did not standardize the subjects' arm positions; in reality, most people who do squats in the gym do so with a bar on their shoulders—the position of which also affects the correctness of their form. In terms of quantity, our project could be improved with more data, encompassing subjects of different genders, physique, and perhaps weight used for the squats.

Conclusion

In conclusion, the CNN we built with 4 Conv-Conv-MaxPool-Dropout layers performed better than other models with fewer layers. The model is complicated enough to understand the features we use and also generalize well on the dev set. Future research may consider using real gym data to capture the nuances of lifters' forms, as well as the relations between weight used and types of mistakes in form. In the future, this project can also be generalized to other popular weightlifting exercises to become a more complete AI personal trainer.

Contributions

Weixiong (Max) Zheng: Data preprocessing. Experimentation with architectures. Training models. Building final architecture. Writing repository.

Peicun Jiang: Background research. Data collection. Data labeling. Poster design and construction.

References

Baker D. "Comparison of upper–body strength and power between professional and college–aged rugby league players. *J Strength Cond Res*. 2000;15(1):30–35

Daniels, Dianne Yow, Queen, J. A llen, and Donald Schumacher. "Obesity and Poverty: A Growing Challenge" *Principal*. Feb. 2007

Fernandez, Cecilia. "Building muscle: Demand will continue to grow as public health concerns mount." *IBISWorld*. Dec. 2018

Del Vecchio, Daewoud, & Green. "The Health and Performance Benefits of the Squat, Deadlift, and Bench Press." *MOJ Yoga & Physical Therapy*, vol. 3, no. 2, 2018, medcraveonline.com/MOJYPT/MOJYPT-03-00042.pdf.

Toshev, Alexander, and Christian Szegedy. "DeepPose: Human Pose Estimation via Deep Neural Networks." *2014 IEEE Conference on Computer Vision and Pattern Recognition*, 20 Aug. 2014