Computer Vision Solutions for Judging Squat Depth in Powerlifting Competitions

Hunter M. Von Tungeln
hvt7@hawaii.edu

Tyler Mak
tylermak@hawaii.edu

Benjamin Banilower
bbanilow@hawaii.edu

University of Hawaii at Manoa
2500 Campus Rd.
Honolulu, HI 96822

Abstract

Powerlifting is a sport consisting of three main barbell lifts, the squat, the bench press, and the deadlift. For each movement, there are three attempts to perform. All three of these movements are judged by three judges, two on the sides, and one in the front. These lifts must be performed to a certain standard in order to be called a "good lift." This paper examines the barbell squat and seeks to utilize existing computer vision technology to judge squat depth. For a squat to be considered "depth" in Powerlifting, the hip crease must be below the top of the knee. To do this, recent pose estimation models are used to generate pose estimations from videos of a human squatting, as well as key points corresponding to joints and other body parts. The pose estimation video output and key points are then fed into a trained neural network that predicts if a squat meets the depth standard. We demonstrate that our frameworks for judging squat depth need massive improvement and that more, state-of-the-art methods should be investigated for their application to this problem.

1. Introduction

Pose estimation is a fundamental problem in computer vision and artificial intelligence, with many applications. The way pose estimation is done is predicting locations of key points, like joints in a given video. Human 3D pose estimation aims to predict and map key points from a 2D video into 3D space. In the last decade, there have been multiple models developed that aim to solve this problem. This paper is an application of this solution to another problem, being judging in the sport of Powerlifting.

1.1. Powerlifting Background

Powerlifting is a sport consisting of three main barbell lifts, the squat, the bench press, and the deadlift. For each movement, there are three attempts to perform. All three of these movements are judged by three judges, two on the sides, and one in the front. These lifts must be performed to a certain standard in order to be called a "good lift." Judges will either give a white light for a good lift, or a red light for a lift that is no-good. To be counted as a good lift, a lifter must receive at least two white lights from the judges. If a lifter receives at least one white-light, they may contest the judge's decisions. The main rule that we are concerned with for the purpose of this paper is the squat depth rule, where the hip crease must reach below the top of the knee joint at the bottom of the squat movement.

1.2. Motivation

Sometimes, judging in Powerlifting can be subjective, because it is the perception of the judges that counts. An innattentive or unfocused judge may not see if the lift was performed to the set standard, and may very well red-light a good lift, and vice-versa. It is possible that a seasoned judge may also make a bad call, and red-light a good lift, or white-light a bad lift. The motivation behind this project is to make judging lifts at a powerlifting meet more objective than subjective, which can be achieved with the use of computer vision technology.

2. Related Work

Currently, the standard method for judging in powerlifting involves three judges positioned at different angles: one on the left, one on the right, and one in the center. This setup is the standard in competitions. In higher-level competitions, lifts are often recorded to address any disputes about the judges' calls. In such cases, a

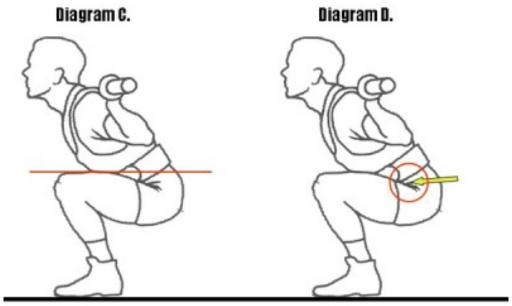

Figure 1. Diagram of a squat considered to be “depth.”[**?**]

jury reviews the recordings and makes the final decision. However, the judges’ decisions can sometimes be imprecise and subjective. Integrating a computer-based evaluation system could enhance both the accuracy and objectivity of the judging process.

One option for analyzing squat depth with high precision is the use of marker-based motion capture devices. These systems involve attaching reflective markers to specific points on the athlete’s body, which are then tracked by multiple cameras placed around the performance area. The cameras capture the motion of the markers in three dimensions, enabling precise reconstruction of the athlete’s movements.

Recent advancements in pose estimation using computer vision, such as PoseFormer[**?**], Detectron2[**?**], and PETR-Pose [**?**], offer unobtrusive, accurate, and cost-effective alternatives to marker-based motion capture. The primary techniques used in 3D pose estimation involve convolutional neural networks (CNNs). However, some models, such as PoseFormer and PETR-Pose, use transformer-based networks.

CNNs excel at extracting local features from images through convolutional layers, making them highly effective for tasks that require localized information, such as identifying keypoints in pose estimation. They are also computationally efficient, enabling faster training and inference compared to transformer models. However, CNNs often struggle to capture global context due to their limited receptive fields. This limitation can reduce accuracy in complex scenarios involving occlusions or interactions between multiple body parts. Transformer-based models address this limitation with their self-attention mechanism, which captures long-range dependencies and understands the global structure of human poses. This capability makes them more robust in handling occlusions within images, particularly in scenarios where multiple body parts may be partially or fully obscured.

For this problem, the goal is not to develop new 3D pose estimation models but to apply existing models to a specific, niche challenge. Our focus is on how the output from these models can be utilized to determine squat depth in powerlifting.

3. Data And Collection

3.1. Data Collection

The data for this project consisted of barbell squat videos with a single person in focus. A single frame was manually extracted from each video. Videos for this paper was collected through publicly available videos on Instagram and YouTube, as well as from friends of the authors. Videos were cropped to isolate the person squatting. The dataset included squat videos from multiple people at different angles in order to make our decision making model more robust. A total of 230 videos were collected for this project, with one frame per video being used for the dataset.

3.2. Pre-Processing

To preprocess, frames were renamed, indexed, and given an “Is Depth” attribute, annotated with a default boolean false value. Frames were then manually annotated by one author, Hunter, as he is an experienced powerlifter with a lot of informal experience of judging squat depth. This information was stored in a CSV file, with the index, frame path, and Is Depth as columns. These frames were then fed into our pose estimation model, provided by Google’s MediaPipe, to generate landmarks.

3.3. Landmarks and Pose Estimation

When a frame is fed into the pose estimation model provided by Google’s MediaPipe, an annotated image was generated, and two sets of landmarks for body locations were generated. The first set is a list of landmarks relative to locations within the image, while the second set is a list of landmarks of real-world positions in 3D space. Each landmark has a corresponding x-coordinate, y-coordinate, and z-coordinate, resulting in 99 numeric data points in each set of landmark, for a total of 198 numeric points for each frame. These values were stored in CSV files, with each column corresponding to the the specific landmarks and their respective x, y, and z values, along with paths to the original frame, the annotated image, and the Is Depth attribute. One CSV file contained only the image landmarks, another contained only the world landmarks, and the third contained both.

4. Methods and Experiments

In this section, we present multiple frameworks for judging Powerlifting squat depth.

4.1. Comparison Test

As a baseline for the results, we performed a simple comparison of the hip vertices to the knee vertices. This

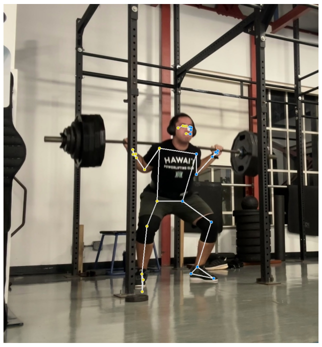

Figure 2. A "good" pose estimation.

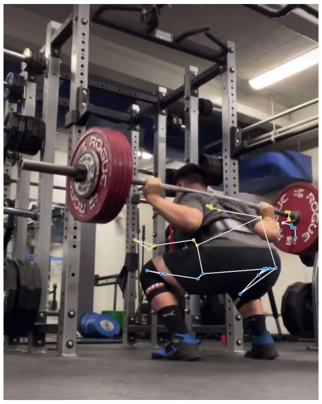

Figure 3. An example of a "bad" pose estimation.

method utilized the 3D points generated from Mediapipe's pose landmarks, focusing on a series of height comparisons to determine whether the hip vertices descended below the level of the knee vertices during the squat.

4.2. Multi-Layer Perceptron

Leveraging the three different datasets from the output of Mediapipe's pose landmark detection algorithm, we use a MLP classifier from sklearn to judge squat depth for a variety of images.The features and labels are prepared by removing unnecessary columns and selecting the target labels. A hyperparameter grid is defined to tune the MLP classifier's hidden layer sizes, learning rates, and L2 regularization. Stratified k-fold cross-validation ensures balanced class distribution. Multiple random states are tested to evaluate model robustness, with stratified splitting maintaining class balance during training, validation, and testing to ensure that the small size of the data does not heavily skew performance metrics. Standard scaling is applied to standardize features, improving convergence. GridSearchCV is used to find the best hyperparameters, and the optimal MLP model is trained and evaluated on validation and test sets. From experimentation, it seems that standardization of the data provided the largest improvement in accuracy of around 20% across all folds, likely due to balancing feature importance and helping with regularization.

4.3. Convolutional Neural Network

Using a Convolutional Neural Network on a set of squat images, annotated with a boolean if it is depth or not, we first preprocessed the data by normalizing the pixel values by dividing the pixel values into the range of $[0, 1]$ as neural networks perform better on normalized input. The data is split into a 60/20/20 split before being converted to numpy arrays. A model is then created using Keras. The architecture includes: Three convolutional layers with increasing filter sizes (32, 64, 128), each followed by max pooling. A flattening layer to convert 2D data into a 1D vector. A fully connected dense layer with 128 units and ReLU activation. A dropout layer with a 50% dropout rate to prevent overfitting. An output dense layer with a sigmoid activation for binary classification The model is then compiled using the Adam optimizer, binary cross-entropy loss, and accuracy as the metric used to evaluate performance. Given the limitation of having a small dataset, anymore than three convolutional layers in the architecture results in overfitting of the training data and a vanishing gradient, resulting in the testing accuracy being 0.5, no better than random guess.

5. Results

Overall, our frameworks performed poorly on our given data. Much of this can be attributed to the very small dataset and bad pose estimations generated by MediaPipe.

5.1. Comparison Test

For the baseline comparison of squat depth, we utilized Mediapipe's world coordinates (WorldLandmarks) and normalized coordinates (Landmarks) to evaluate performance. Since this method did not require the use of a neural network, we were able to process the entire ground truth dataset without concerns about overfitting during this part of the pipeline.

To assess the performance, we compared the ground truth labels with the outputs from this baseline method. We calculated the following metrics:

- True Positives (TP): Cases labeled as good depth where the output also indicated good depth.
- True Negatives (TN): Cases labeled as bad depth where the output correctly identified bad depth.
- False Positives (FP): Cases labeled as bad depth where the output incorrectly identified good depth.
- False Negatives (FN): Cases labeled as good depth where the output incorrectly identified bad depth.

For our 171 data points, the results were as follows:

World Coordinates Results:

- True Positives: 42
- False Positives: 82
- True Negatives: 7
- False Negatives: 40

Evaluation Metrics:

- Accuracy: 0.286
- True Positive Rate: 0.512
- False Positive Rate: 0.921
- Precision: 0.339

Normalized Coordinates Results:

- True Positives: 38
- False Positives: 79
- True Negatives: 10
- False Negatives: 44

Evaluation Metrics:

- Accuracy: 0.281
- True Positive Rate: 0.463
- False Positive Rate: 0.888
- Precision: 0.325

Both of these results indicate poor accuracy as over half of the data set were given an incorrect result. Even though the results for both coordinate types were similar, we believe that the WorldLandmarks are a better choice to use if we were to refine this process. This is due to the WorldLandmarks providing real-world measurements of the detected landmarks. This should minimize the effects of variations in camera angle, resolution, or positioning within the frame.

5.2. Multi-Layer Perceptron

In the MLP model, the validation and test accuracies show significant variability across different random states, with validation accuracies ranging from 0.46 to 0.78 and test accuracies from 0.46 to 0.77. This indicates that the model's performance is highly dependent on the specific data split, underscoring the need for a larger and more diverse dataset to achieve consistent results. The average confusion matrix for each random state suggests that while the model generally performs better than random chance, there is plenty of room for improvement in reducing false positives and false negatives.

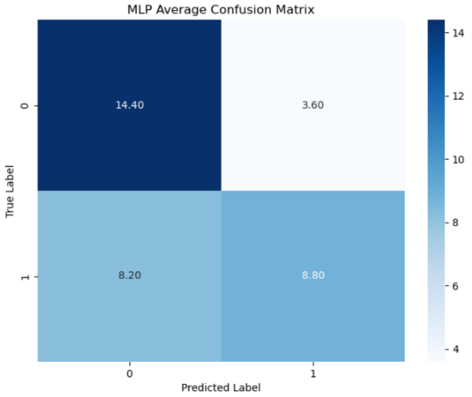

Figure 4. MLP Confusion Matrix

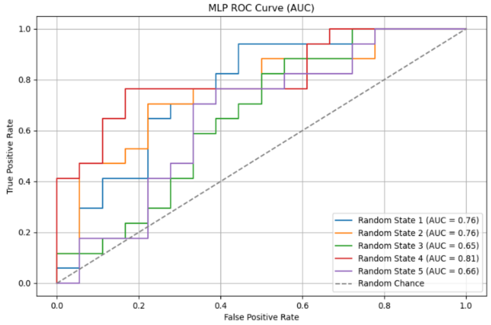

Figure 5. MLP ROC Curve

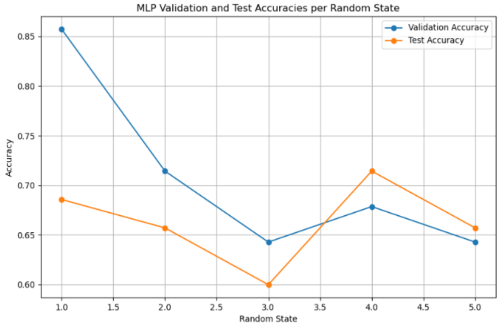

Figure 6. MLP Random State and Accuracy Correlation

5.3. Convolutional Neural Network

For the convolutional neural network, the training accuracy starts at around 0.53 and improves to approximately 0.68 by the 10th epoch. The validation accuracy, however, fluctuates significantly, beginning at 0.37 and reaching up to 0.61. This fluctuation suggests that the model is struggling to generalize well to unseen data, due to the small dataset size and the inherent variability in squat images. The test accuracy achieved is approximately 0.54, which is only slightly better than random guessing. This result reinforces the observation that the model struggles to generalize from the training data to the test set because of possible overfitting and lack of a sufficient data samples for the model to effectively learn from.

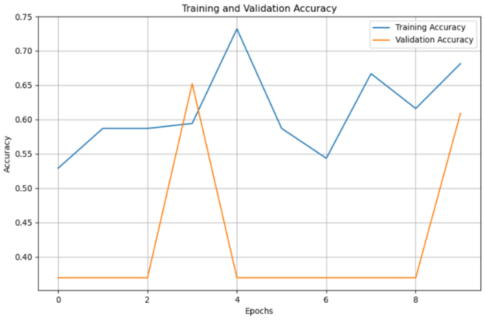

Figure 7. CNN Accuracy Given Epochs.

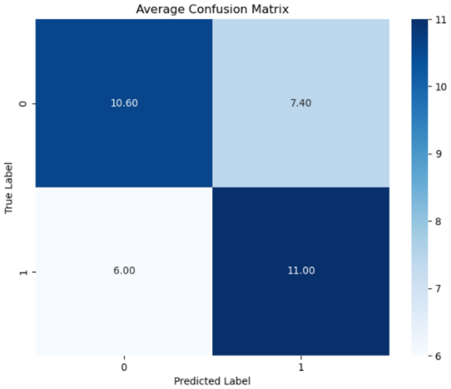

Figure 8. CNN Confusion Matrix.

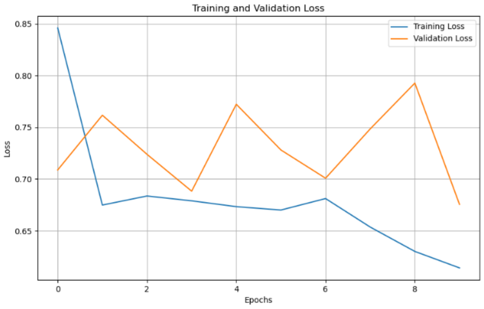

Figure 9. CNN Training and Validation Loss

6. Conclusion

Our solution is very limited given the pose estimation model chosen, and lack of a vast amount of data to train on. Even given these circumstances, these frameworks are not effective in judging squat depth. This is mainly in part due to the pose estimation model chosen. These frameworks would not be fit for judging a squat in Powerlifting under any circumstance.

6.1. Areas For Future Improvement

More data should be collected and worked with for this project. This project was heavily limited by the amount of data collected, as collection of data was very tedious and time-consuming. More data would allow for more robust training and better generalization of our decision making machine learning models. Multiple angles of the same squat would be ideal, as in a Powerlifting meet setting, there are three judges that judge the lift at three different places. Cameras would be at the same height as the lifter's kneecap, which would be most effective in capturing the depth of the squat. The quality of the data should also be improved. Most videos were obtained by screen-recording on Instagram on an iPhone 15, which, while being better than nothing, severely hurts the quality of the actual video in comparison. In addition, more pose estimation models should be used to compare performance between MediaPipe's solution and other, more recent and state-of-the-art methods. For this project, trying to use other, more recent, open source models took too long to learn and setup. MediaPipe was by far the easiest to get started with. Many of these newer, open source models, also use a dataset that was not able to be accessed, that being the Human 3.6 dataset. Our pose estimation model was also heavily affected by occlusions and obstructions. The model sometimes was not even being able to estimate a pose from a given image due to the occlusions and obstructions. Using more state-of-the-art models and methods and comparing their performance given these occlusions and obstructions would be helpful in solving this problem.

Acknowledgements

We would like to acknowledge Benjamin Chee, Jack Lai, Hannah Royer, Henry Tran, Krystian Mierzejewski, Matthew Lewis Vena, Ian Villa, Shawn Deloso, and Alijandro Zachraias for their squat videos.