### **Analyzing Barbell Back Squat Form Using Machine Learning**

*Alexander Steele*

Leesburg, Virginia

### **Table of Contents**

[Introduction](#_njietlt73w9u) 3

[Materials and Methods](#_13b4vpzcam6c) 5

[Results](#_tdo49dkbo7i2) 9

[Discussions and Conclusions](#_ipx9iwggx1c) 10

[References](#_spj1egry28hi) 11

### 

### **Introduction**

Due to the COVID-19 pandemic, more people than ever are aware of the importance of maintaining their physical and mental health. One of the most effective and popular ways to do this is by lifting weights at the gym. 39% of United States citizens have a membership at a gym, which totals nearly 130 million people (Stasha, 2021). Many of these gym members perform conventional lifts such as squats, deadlifts, and bench presses. However, it's common for new gym members to be unaware of the proper practices for these conventional lifts. These practices include proper form, weight, reps, sets, and more. Lifting without proper practices can cause serious injury and hinder results. Even sub-elite to elite lifters report that 22%-32% of their injuries are related to squat, 18%-46% to the bench press, and 12% - 31% to the deadlift (Bengtssson et al., 2018). Although direct causation cannot be determined for these injuries, they are generally attributed to overuse, using overly excessive weight, and/or lifting with improper form.

This paper will focus on the last of those potential causes, improper form. Proper form is a key aspect of weightlifting for any gym member. It not only prevents injury but can also improve results. The majority of people lifting weights do so in order to burn calories and lose weight, gain muscle, or simply stay in shape. Lifting with proper form can benefit those results, while also reducing the risk of injury.

Currently, there are limited methods that new gym members can pursue in order to improve their form. These methods typically consist of hiring a personal trainer or relying on a friend for assistance. However, these options are often expensive and inconsistent. A solution that a typical gym-goer could use by themselves at little to no cost would be extremely effective, and the market for this problem is large and continuing to grow.

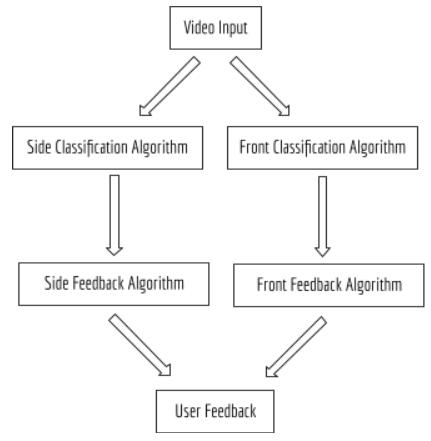
This paper will discuss research on an all-encompassing solution to this problem that uses machine learning and human pose estimation to analyze barbell back squat form. Human pose estimation is a computer vision-based technology that can detect and analyze human posture in videos or on live cameras. It identifies the key points on the human body such as the shoulders, the head, the hips, etc. These key points are translated into quantitative coordinates which can be analyzed. This solution builds off of previous research and involves four distinct algorithms. Two of the algorithms will be classification algorithms built using a k-nearest neighbors algorithm in order to classify barbell back squat form as "good" or "bad". One of these algorithms will classify squat form from a side-facing angle, and the other will classify squat form from a front-facing angle. These algorithms will be able to provide a score or rating on how good the squat form is. The third and fourth algorithms will use the coordinates from the human pose estimation to form vectors and will be able to provide specific feedback to users on what they can do to improve their form. One algorithm will analyze form from a side-view while one algorithm will analyze form from a front-facing view.

Similar algorithms have been developed in other research, however, they use different types of human pose estimation systems. For example, Eivindsen and Kristensen (2020), used multiple types of human pose estimation systems including OpenPose and AlphaPose and found AlphaPose to grant the highest accuracy. However, they reported poor accuracy from their pose estimation tools on a side view video, which caused their analysis algorithms to be more inaccurate from a side view. This research uses MediaPipe Pose, a human pose estimation system created by Google, to see if it results in a higher accuracy of analysis of barbell back squats from a side angle. Additionally, the three algorithms together will provide a comprehensive tool for people to use to improve their barbell back squat form.

### 

### **Materials and Methods**

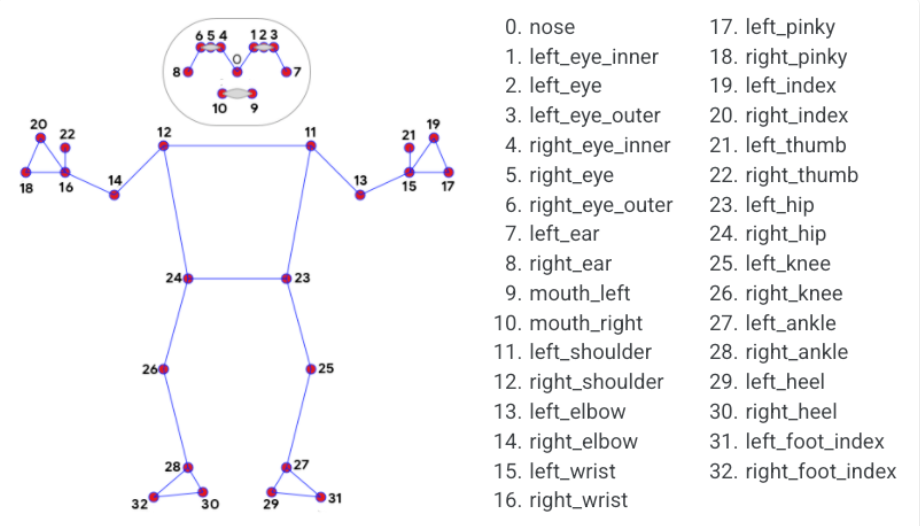
The goal of this study is to create four algorithms that provide different analyses of squat form as effective methods to improve users' barbell back squat form. The overall structure of the project can be seen below in *Figure 1*.



### *Figure 1. Flowchart of the final product*

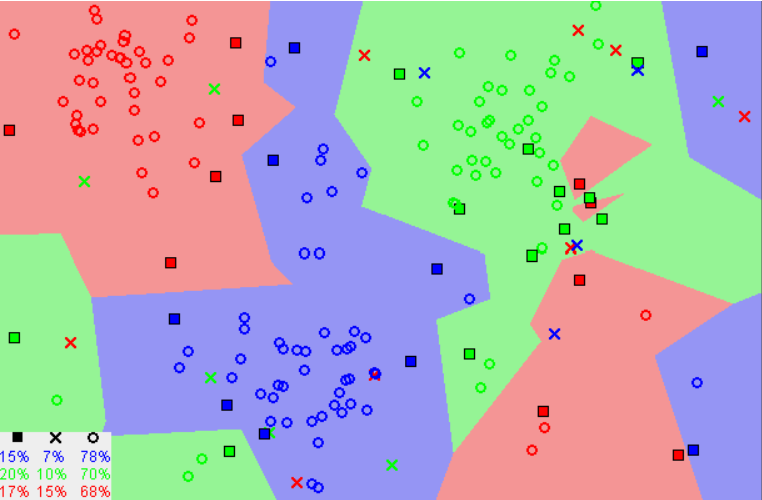
The side classification algorithm was developed first using the MediaPipe Pose for human pose estimation and a kNN algorithm for classification.

MediaPipe Pose by Google was chosen as a human pose estimation system due to its fast runtime, easy implementation, and the fact that it tracks more body coordinates than other popular human pose estimation systems (33 vs. OpenPose's 27 and AlphaPose's 26). The specific coordinates tracked by MediaPipe Pose are below in *Figure 2*.



### *Figure 2. MediaPipe Pose coordinates*

A k-nearest neighbors (kNN) classification algorithm was chosen to perform the classification, as it functions by calculating the euclidean distance between points in the dataset and grouping them together. This is ideal for the classification of barbell back squat form as the quality of the squat form largely depends on the distance between certain body parts at specific instances of the squat. *Figure 3* displays an example of how a kNN classification algorithm groups data together.



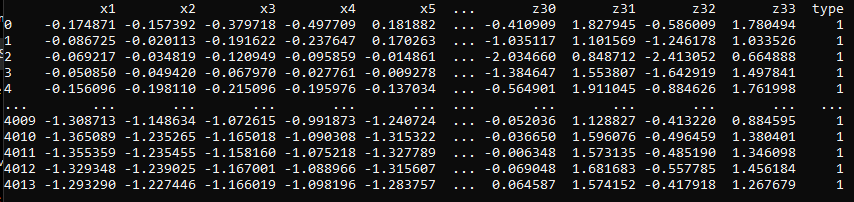
### *Figure 3. kNN Classification Grouping Example*

A major parameter of the kNN algorithm is the k-value, which indicates where the algorithm should form boundaries between the data points. For example, with a k-value of two, the algorithm would group the data into groups of two and classify incoming data based on these small groups. However, with a k-value of 30, the algorithm would group the data into groups of 30 and classify the incoming data based on these much larger groups. A small k-value typically leads to unstable decision boundaries due to their small size, making a substantially large k-value lead to better classification as it leads to smoothening the decision boundaries due to the larger groups. The optimal k-value is typically found to be the square root of the total number of samples in the dataset.

The side classification algorithm was the first one created. The training data for this algorithm was gathered via two methods: Youtube videos and recording videos of the researcher squatting. The Youtube videos were screen recorded using a Python script with a computer vision library called OpenCV. The videos were then manually cropped to just encompass the beginning and the end of the squat. The videos of the researcher were recorded using an iPhone and also cropped to encompass the beginning and the end of the squat. In total, 48 videos of people squatting from a side view were recorded with 24 coming from Youtube and 24 from the researcher. 26 of these videos were videos of squats with good form and 22 of them were videos of squats with bad form. The training data included people of different body types and the bad videos included examples of different parameters of the squat being adjusted that can be seen from a side angle such as the positioning of the arms, torso, knees, hips, head, and feet, along with examples of squats with a lack of depth.

Before manually classifying these videos as good or bad squats, extensive research was done on what determines good squat form from a scientific standpoint. The metrics used to manually classify squat form from a side angle were degree of knee flexion, knee drift, trunk angle, and head position. Escamilla (2001) found that squatting to a parallel depth or 100° knee flexion was most effective in reducing injury while activating the appropriate muscles. Schoenfield (2010) found that minimal knee drift, a straight trunk, arms in line with trunk, full surface of foot in contact with the ground, and a straight or upward head position during the squat were most effective in injury reduction while maintaining appropriate muscle activation. The form in each video was initially manually classified into good or bad categories based on these metrics. Additionally, the videos of bad form in the dataset included multiple examples of each of these parameters modified in order to account for a variety of bad types of squat form. Specifically, videos were gathered with the squatter having their heels off the ground, their arms in front of their trunk, their head facing downwards, their trunk rounded, their trunk leaning forwards excessively, not squatting to parallel depth, and knees drifting over toes.

After all of the videos were gathered, each of them was run through the MediaPipe Pose system and converted to quantitative coordinates. 33 coordinates in x, y, and z respectively were gathered for each frame of each video. These coordinates were then placed in respective JSON files, one for each video. Once all 48 JSON files were created they were entered into the dataset used to train the kNN algorithm. Before entering the dataset, the coordinates in each file were standardized respectively to ensure that the location of the squatter within the video didn't affect the accuracy of the model, and only the distance between the coordinates themselves was a factor. They were standardized by ensuring that the mean and standard deviation are 0 and 1 respectively. The data was also split up into frames before entering the dataset. The final dataset includes 4013 frames and can be seen below in *Figure 4*.



### *Figure 4. Side-Facing Dataset Overview*

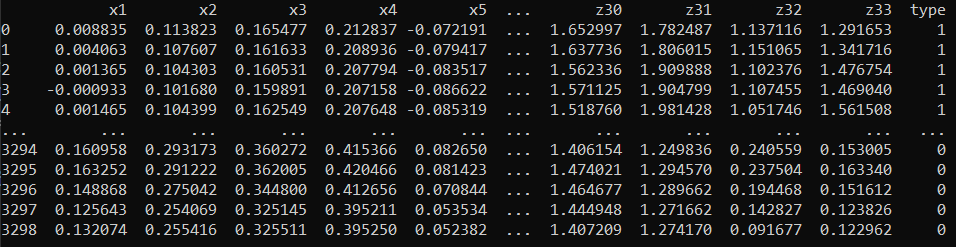
Each row in the dataset represents one frame, and each row is labeled with a "1", or "0", signifying if the form in that frame is good or bad respectively. All of the frames taken from good examples of barbell back squat form were labeled as good, and all of the frames taken from bad examples of barbell back squat form were labeled as bad.

The kNN classification algorithm was then trained on this dataset, with a target of 90% accuracy in the classification of the frames. When receiving a new video of someone squatting from a side angle, the algorithm classifies each individual frame in the video as having good or bad form by marking it with a "1", or "0", then outputs the percentage of good frames in the video. This percentage can be considered the "rating" of the squat form. While training the algorithm, the k-value was optimized to result in the highest accuracy model without creating unstable decision boundaries or overfitting the training data. The optimal k-value was found to be 63, which is approximately the square root of the total number of frames in the dataset.

The front classification algorithm was the next algorithm created. The training data for this algorithm was gathered via the same two methods that were used for the side classification algorithm: Youtube videos and recording videos of the researcher squatting. In total, 40 videos of people squatting from a front view were recorded with 15 coming from Youtube and 25 from the researcher. 20 of these videos were videos of squats with good form and 20 of them were videos of squats with bad form. The training data included people of different body types and the bad videos included examples of different parameters of the squat being adjusted that can be seen from a front angle, such as the positioning of the arm, torso, knees, hips, and head.

The aforementioned research on squat form was also used to manually classify the squatting videos from a front facing angle into good or bad categories. Specifically, videos were gathered with the squatter having their arms in front of their trunk, their head facing downwards and to the left or right rather than upright, their trunk leaning to the left, their trunk leaning to the right, their knees drifting too far inwards, their hips leaning to the left and their hips leaning to the right.

After all of the videos were gathered, each of them was run through the MediaPipe Pose system and converted to quantitative coordinates just like with the side classification algorithm. Once all 40 JSON files were created they were entered into the dataset used to train the kNN algorithm. Before entering the dataset, the coordinates in each file were also standardized respectively to ensure that the location of the squatter within the video didn't affect the accuracy of the model, and only the distance between the coordinates themselves was a factor. The final dataset includes 3298 frames and can be seen below in *Figure 5*.



### *Figure 5. Front-Facing Dataset Overview*

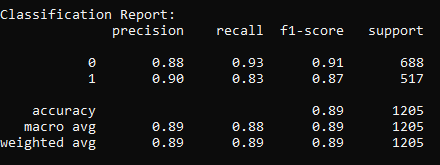
Another kNN classification algorithm was trained on this dataset using the same method as the side classification algorithm, also with a target of 90% accuracy in the classification of the frames. The percentage score for new video inputs also works the same way as the side classification algorithm. The optimal k-value for this algorithm was found to be 57, which is approximately the square root of the total number of frames in the dataset.

The two feedback algorithms are next to be developed. These algorithms will use the coordinates from MediaPipe Pose to calculate vectors between specific joints, such as the knees and hips. The magnitudes and angles of these vectors relative to each other will be used to provide specific feedback to the user regarding what particular parts of the squat need to be improved.

Once the feedback algorithms are complete, they will be integrated into an iOS application. The app will allow users to input videos of themselves squatting and receive feedback on their form. Once the user inputs the video, they will select if they're inputting a video of a side-facing squat or front-facing squat and it will run through the respective k-nearest neighbors classification algorithm and feedback algorithms. The user will then receive a percentage score on how many frames from their video were classified as having good form along with specific feedback on what particular parts of the squat can be modified to improve the form.

### **Results**

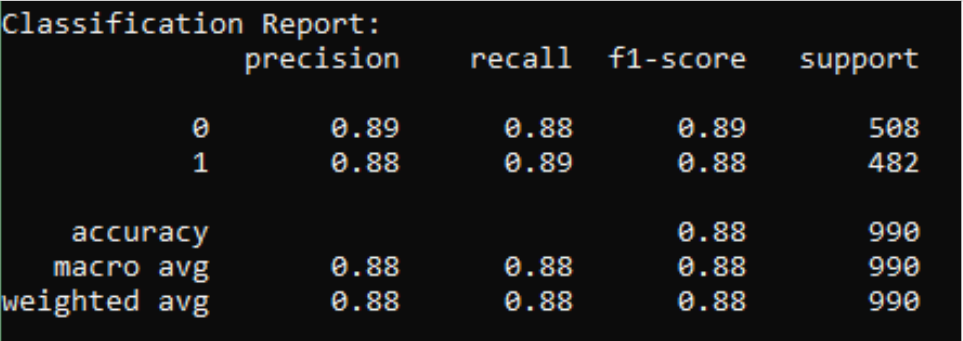
The side-facing kNN classification algorithm succeeded in classifying frames from videos of people squatting from a side angle as it obtained an accuracy of 89%, as seen below in *Figure 6*. This accuracy is slightly below the original target of 90%.



### *Figure 6. Analysis of kNN Classification Algorithm for side-facing squat frames*

In *Figure 6*, "1" represents good frames and "0" represents bad frames. The recall for good frames at 0.83 was significantly lower than the other values, meaning that the algorithm identified good frames correctly at a lower overall accuracy. The lower precision value of 0.88 for bad frames compared to the 0.90 for good frames, along with the low recall for good frames shows that the algorithm has a tendency to over classify frames as bad, as it inaccurately classifies bad frames more frequently than good frames.

The front-facing kNN classification algorithm succeeded in classifying frames from videos of people squatting from a front angle as it obtained an accuracy of 88%, as seen below in *Figure 7*. This accuracy is also slightly below the original target of 90%.



### *Figure 7. Analysis of kNN Classification Algorithm for front-facing squat frames*

This algorithm was found to be more balanced than the side-facing algorithm, as its precision and recall accuracies were very similar for both good and bad frames.

### **Discussions and Conclusions**

The results of this study show that side-facing classification for barbell back squat form can be performed accurately despite previous research finding side-facing analysis inaccurate. Furthermore, it demonstrates that a k-nearest neighbor algorithm combined with MediaPipe Pose is an effective method in classifying barbell back squat form, and may be more effective than Alpha Pose in classifying barbell back squat form from a side angle. These tools could potentially be used to classify other exercises, such as the deadlift or the bench press, by using similar methods.

The algorithms' high degree of accuracy can be attributed to the following factors relating to the training data: the amount of data, the variety of data, and the optimization of the algorithm. The side-facing algorithm's dataset includes 48 videos resulting in 4013 frames ensuring that the algorithm had enough examples to accurately classify the squats. Due to the many different parameters that can be seen from a side view that can cause barbell back squat form to be considered poor, examples of them were included within the training data which allowed the algorithm to be robust and classify many different types of squats. Additionally, by modifying the k-value to 63 as the square root of the total number of frames in the dataset, the accuracy of the algorithm was maximized. The same approach was used for the front-facing algorithm's dataset, however it includes slightly less examples (40 vs. 48) which may be the cause of the front-facing algorithm's accuracy (88%) being slightly lower than the side-facing algorithm's accuracy (89%). Examples of the many different parameters of the squat that can be seen from the front view were included within the training data for the front-facing algorithm as well, and the k-value was optimized to 57 to maximize the accuracy of the algorithm.

The over-classification of frames with bad squat form in the side-facing algorithm may be attributed to the lack of videos of bad squats in the dataset. There were only 22 videos of bad squats while there were 26 videos of good squats. The front-facing algorithm, however, did not suffer from over classification of frames with bad squat form. This is most likely because the dataset was balanced, as it included 20 videos of good squats and 20 videos of bad squats. Therefore, the side-facing algorithm's issue of over-classifying frames with bad squat form may be resolved by balancing out the amount of videos of good and bad squats in the dataset for the side-facing algorithm.

The current algorithms in tandem could be used as an effective, low-cost method for people to gauge how good their barbell back squat form is and record their improvement over time based on the percentage of good frames in the video. When combined with the feedback algorithms, this system could be an all-encompassing solution to improving barbell back squat form.

### **References**

Band, A. (2022, March 3). *How to find the optimal value of K in Knn?* Medium. Retrieved March 10, 2022, from <https://towardsdatascience.com/how-to-find-the-optimal-value-of-k-in-knn-35d936e554eb#:~:text=K%20value%20indicates%20the%20count,is%20a%20lazy%20learning%20algorithm>.

Bengtsson, V., Berglund, L., & Aasa, U. (2018). Narrative review of injuries in powerlifting with special reference to their association to the squat, bench press and deadlift. *BMJ open sport & exercise medicine*, *4*(1), e000382. <https://doi.org/10.1136/bmjsem-2018-000382>

Eivindsen, J. E., & Kristensen, B. Y. (2020). *Human Pose Estimation Assisted Fitness Technique Evaluation System* (thesis). Norwegian University of Science and Technology, Trondheim.

Escamilla, RF. (2001). Knee biomechanics of the dynamic squat exercise. *Medicine and Science in Sports and Exercise*, 127–141. <https://doi.org/10.1097/00005768-200101000-00020>

Harrison, O. (2019, July 14). *Machine learning basics with the K-nearest neighbors algorithm*. Medium. Retrieved March 3, 2022, from <https://towardsdatascience.com/machine-learning-basics-with-the-k-nearest-neighbors-algorithm-6a6e71d01761>

*Pose*. mediapipe. (2020). Retrieved March 10, 2022, from <https://google.github.io/mediapipe/solutions/pose.html>

Schoenfeld, B. J. (2010). Squatting kinematics and kinetics and their application to exercise performance. *Journal of Strength and Conditioning Research*, *24*(12), 3497–3506. <https://doi.org/10.1519/jsc.0b013e3181bac2d7>

Stasha, S. (2021, February 14). *Fitness Industry Statistics for 2021: Policy advice*. PolicyAdvice. <https://policyadvice.net/insurance/insights/fitness-industry-statistics/>.