Creating an Automated Coaching Assistant: Implementing Technology to Assess Quality of Strength Training

Michael Clougher and Ozgur Izmirli

Abstract – **While strength training is beneficial for both athletes and non-athletes alike, proper instruction in the area can prove expensive, inconvenient, and not always readily available. Considering the above, we have attempted to create a viable substitute for a human coach in the form of an automated one. This paper explores the feasibility of this with respect to the deadlift exercise.**

I INTRODUCTION

One of the most common strength building exercises performed in gyms is the deadlift. While certainly an effective exercise when performed correctly, correct technique of the deadlift is difficult to master. Compounding this issue is that repeated poor execution can quite easily lead to injuries due to the nature of the movement. In this paper, we have developed an algorithm utilizing the depth and skeletal recording capabilities of the Kinect to evaluate performance of the deadlift. The National Strength and Conditioning Association has released a video describing proper deadlift form and points out five common errors, which include: bending of the arms, feet not being shoulder width apart, raising the trunk of the body faster than the lower portion of the body, having the bar not travel up and down along a straight path, and finally, rounding of the back. Our algorithm attempts to search for and provide feedback on each of these five key points. Due to limitations of the technology, we have chosen to use two cameras with different viewpoints of the subject. In this way, one view is able to inform the other to overcome the lack of one complete three dimensional skeleton with processing done offline using MATLAB. We explore the effectiveness that this technology may serve as a substitute to a human coach when one may not be available.

II RELATED WORKS

In [1] and [2] we see that the Kinect has indeed been proven reliable in terms of its depth data accumulated and effective in the fighting the problem of foreground extraction. In [3] and [4], the authors were able to show that not only did the depth data returned by the Kinect prove reliable, but perhaps even more importantly for this paper is that the joint position estimates competed well with more expensive equipment. This is particularly important to us as not only is the joint tracking an essential part of our methods of tracking, but we want the general public to be able to simply and affordably access this technology one day. The Kinect provides a relatively cheap venue to do so, and with hardly any set up required. [5] and [6] provide results that mirror other gaming systems' attempts at branching beyond simply video games and putting systems like the Nintendo Wii to other uses.

III SET UP

The Kinect is not as accurate or robust as other motion capture systems such as PhaseSpace, however, what it lacks in accuracy and robustness, it makes up for in convenience and ease of use. Simply stand at an appropriate distance from the camera, and a suitable skeleton will automatically be fit onto the user's body. To evaluate the deadlift, we set up two Kinects each connected to a different laptop computer. Each Kinect has been placed 290 cm. from where the user is asked to stand, and each has been elevated 69 cm. and set to an elevation angle of 5°. One has been placed directly in front of the user's point of view, and another at 90° to the right side. In this manner, we are able to fit the entire skeleton of any user up to at least 6'10" comfortably in each frame. Once these machines are properly positioned, the user simply opens the Kinect Stream Saver Application on each computer, specifies a filename and folder, and hits record. The user then walks over to where he or she has been asked to stand and starts to perform the exercise with the bar which has already been placed. After the user has completed as many repetitions of the deadlift as desired, he or she simply puts the bar down and must hit the button on the application which indicates the recordings to be stopped. The user then opens MATLAB and simply inputs the filename to begin processing.

IV Procedures

A) Synchronization

The Kinect Stream Saver application saves images captured of the user at a rate of 30 frames per second. We first must synchronize the two streams of data in order to properly evaluate the user's performance. To do this, we have decided to simply find the point at each stream in which the bar leaves the floor. For the side view, we track the position of the weight. The weight on the bar is large enough that given a rough estimate of its center coordinates, we may quickly find its true center to coordinates to within a few pixels by using an algorithm similar to a Hough transform in which we grow a circle from the estimate and search for edges on the depth image. Given that the user is asked to stand 290 cm away from each Kinect, we may justify hardcoding in a rough approximation of the weight. For the frontal view, we utilize the skeleton data and search for the frame at which the estimated height of the hand joints are both sufficiently close to the bar, and are being raised upwards. We use the estimate of the center of the weight obtained from the side to find the height of the bar. To find the ending point, we again turn to the side view; and, tracking the weight as it is raised and lowered, find the frame at which the weight is both close to the ground and has not moved for a sufficiently long time, (through experimentation we found this to be 25/30 of a second). In this manner, the user is not limited to a predetermined number of repetitions, and may perform as many as desired. Since we have already found the starting points for each stream and they both record at a constant rate of 30 frames per second, it is sufficient to only find the ending side frame and to calculate the front end from there.

B) Bending of Elbows

We first look to ensure that the user has kept his or her forearms straight so as not to pull the bar with the arms, which would not target the muscles to which the deadlift

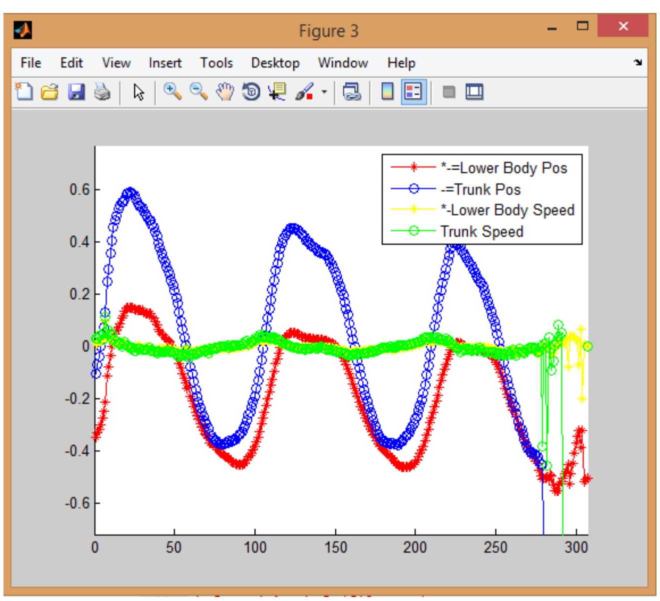

Figure 1: Example of good relative speeds between trunk and lower body.

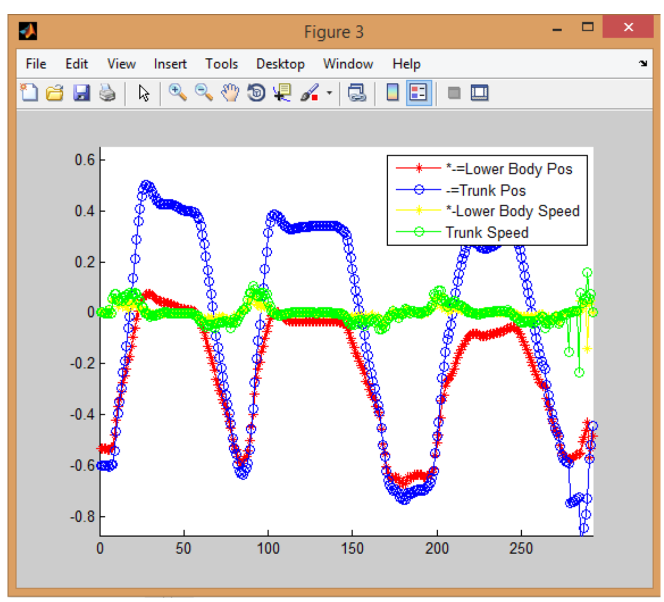

Figure 2: Example of poor relative speeds between trunk and lower body.

is directed. We do this by looking at the joint positions obtained from the front skeletal view of each elbow over all frames. After removing any outliers, we check the x positions of all these joints and if there is a large enough standard deviation, we determine that the user must have bent his or her elbows and thus not kept the arms straight. The coaching assistant then outputs a piece of advice, or positive reinforcement.

C) Position of Feet

We next use a similar method to determine that the user's feet are shoulder width apart. We analyze the skeletal joint positions for each shoulder and foot and, after removing outliers, make a judgment as to where the true positions of the feet are relative to the shoulders. Feedback is again provided whether positive or negative.

D) Speed of Trunk

To determine whether the user has raised their upper body before their lower body, we use several joint position estimates from each over every frame. We once again remove outliers, and then calculate the velocities from these positions. Since we are only concerned with the velocity when the bar is being raised, we only look at the speed of the upper and lower bodies when the velocity is positive. If the trunk is determined to rise significantly faster than the lower body, then the feedback provided to the user will reflect this. Figures 1 and 2 provide good and bad examples of this, respectively.

E) Path of Bar

We have already detailed the general algorithm for tracking the path of the bar; we use our circle growing/edge detection method to detect the weight in each frame. To start, we have predetermined x and y estimates; for each frame after this, we use the estimated center coordinates from the previous frame. We then look at the standard deviation over every estimated x coordinate and determine whether the bar appeared to be travelling straight up and down.

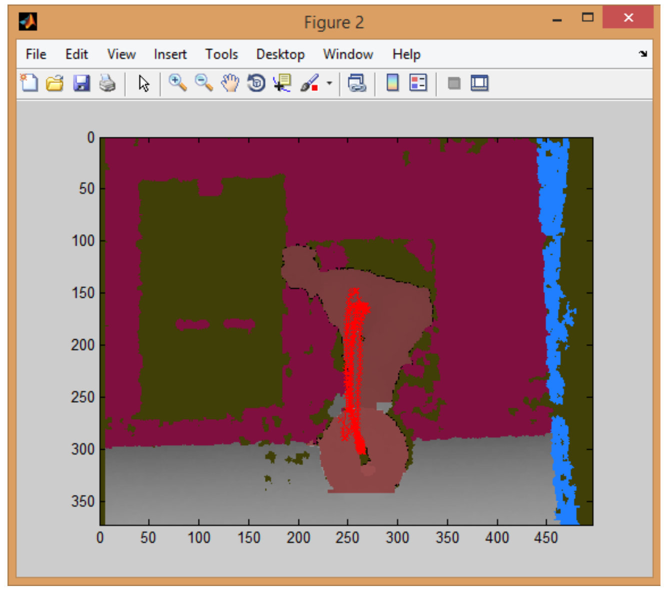

Figure 3: Example of a good path of the bar.

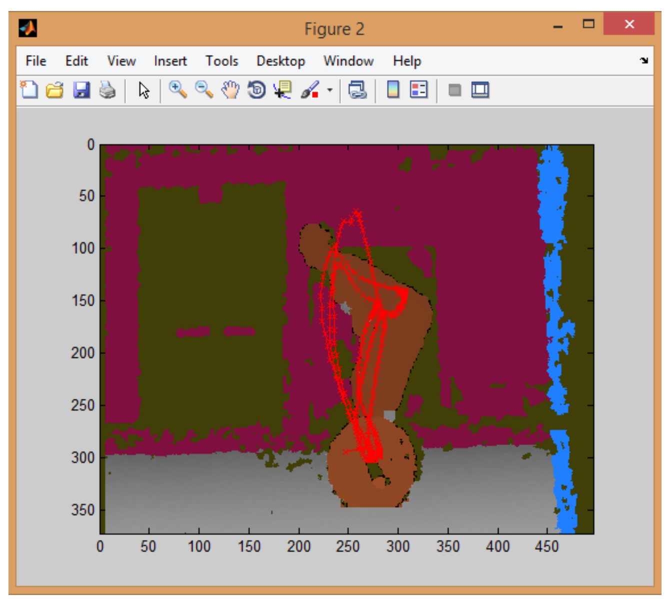

Figure 4: Example of a poor path of the bar.

F) Straightness of Back

Finally, we utilize both the front and side streams to evaluate whether the user has been rounding the back. We first use height and depth estimates for the shoulder and hip joint positions from the front to provide rough approximations for these positions in the side view. Using the slope of the theoretical line connecting these points, we then travel upwards in a perpendicular fashion from this line until we detect the edge of the back. A line of best fit of polynomial degree 2 is then plotted along the back, and the curvature of this line is recorded. Due to unreliability of the skeletal estimates, we cannot continue to rely on the joint estimates long after the first frame. To obtain the new shoulder and hip approximations from here until the end, we rely on the previously fitted line of best fit. Taking the midpoint of this line, we essentially construct a right triangle and move our estimate for the midpoint between shoulder and hip along the hypotenuse of this triangle. The length and angle of hypotenuse is dynamically calculated each frame according to the slope between endpoints on the line of best fit. Thus we are able to track the curvature of the back as the exercise is performed, and if the average curvature exceeds a certain threshold value, the feedback given will communicate this to the user.

V Evaluation of Feedback

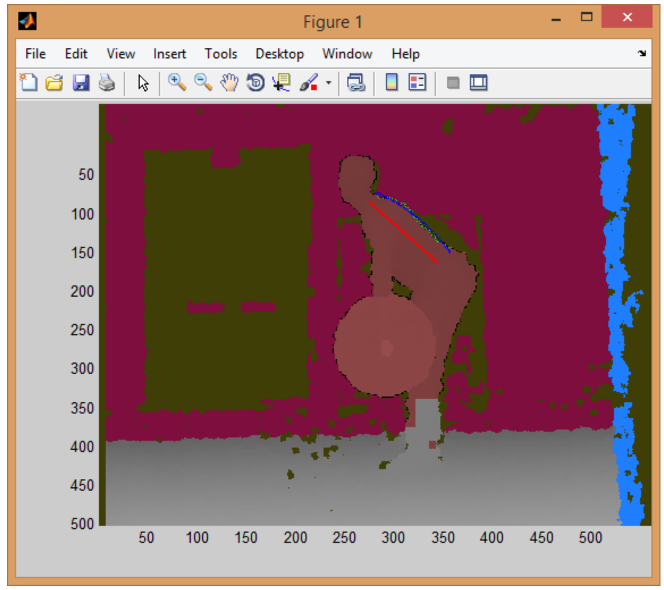

Figure 5: Example of the back being tracked. The red line represents the theoretical line between the shoulder and hip estimates, while the blue represents the line of best fit.

Table 1 summarizes the results of eleven different recordings. Various mistakes were purposefully attempted to be made by the user in some cases, and these occur when there is a "No" in the "User" column. The "Comp" columns are the feedback provided by our automated coaching assistant; "Yes" responds to positive feedback and "No" implies there is room for improvement. The cases where our agent has failed to agree with the user's own evaluation are highlighted in red. It should be noted that most of these disagreements stemmed from poor performance by the user, i.e. the weight did not travel straight up and down and thus

	Straight Arms?		Feet Shoulder Width?		Good Trunk Speed?		Bar Straight?		Straight Back?	
File Name	User	Comp	User	Comp	User	Comp	User	Comp	User	Comp
6	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
7	Yes	Yes	No	No	Yes	Yes	Yes	Yes	Yes	Yes
8	Yes	Yes	No	No	Yes	Yes	Yes	Yes	Yes	Yes
9	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes
10	No	No	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No*
11	Yes	Yes	Yes	Yes	Yes	Yes	No	No	Yes	No*
12	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	No	No
13	Yes	Yes	Yes	Yes	Yes	Yes	No	No	No	No*
14	Yes	Yes	Yes	Yes	No	No	Yes	Yes	Yes	No*
15	Yes	Yes	Yes	Yes	No	Yes	Yes	Yes	Yes	Yes
16	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes	Yes

* = weight has occluded the back too much to provide accurate estimate

Table 1: Results of eleven different recordings. "Yes" represents either the user or automated assistant assessing positive feedback in the respective aspect, as "No" represents negative feedback.

occluded the body too much to reliably estimate the shoulder/hips, making the evaluation of the back much more difficult.

VI Improvements and Conclusion

Only one subject has been evaluated thus far in our recordings; to increase the legitimacy of our automated assistant it would be critical to assess how well it performs on subjects of varying heights and proportions. More accuracy is also required in the evaluation of the trunk speed relative to the lower body speed. Perhaps adding more estimates from the trunk would increase this accuracy, though here appear to be somewhat limited by the accuracy of Kinect data. The time it takes to raise the bar is short enough to provide difficulties in estimating these two relative speeds. Finally, more work must be done to improve robustness of algorithm, particularly in the evaluation of the back. If the weight occludes the body too much, it is quite possible to lose valuable data and make estimation impossible without improvement to our method. In addition, if the user moves too quickly, it is also possible that our algorithm provides inaccurate advice. If these improvements were to be implemented, the concept of an automated coaching assistant to alleviate some of the necessity of a human coach if feasible, particularly with future improvements in the accuracy of the Kinect.

References

[1] J. Smisek, M. Jancosek, and T. Pajdla, “3d with Kinect,” in *ICCV*

Workshops, pp. 1154–1160, 2011.

[2] E. E. Stone and M. Skubic, “Evaluation of an Inexpensive Depth Camera for passive in-home fall risk assessment,” in *PervasiveHealth*, pp. 71–77, 2011.

[3] S. Obdrzalek, G. Kurillo, F. Ofli, R. Bajcsy, E. Seto, H. Jimison and M. Pavel “Accuracy and Robustness of Kinect Pose Estimation in the Context of Coaching of Elderly Population”

[4] J.-D. Huang, “Kinerehab: A kinect-based system for physical rehabilitation a pilot study for young adults with motor disabilities,” in *In Proceedings of ASSETS11*, (Dundee, Scotland, UK.), pp. 319–320, October 2426 2011.

[5] B. Lange, C.-y. Chang, E. Suma, B. Newman, A. S. Rizzo, and M. Bolas, “Development and evaluation of low cost game-based balance rehabilitation tool using the Microsoft Kinect sensor*,” Conference*

Proceedings of the International Conference of IEEE Engineering in Medicine and Biology Society, vol. 2011, pp. 1831–1834.

[6] M. J. D. Taylor, D. McCormick, T. Shawis, R. Impson, M. Griffin “Activity-promoting gaming systems in exercise and rehabilitation”