

Three-dimensional Monitoring of Weightlifting for Computer Assisted Training

Anargyros Chatzitofis

Nicholas Vretos

Dimitrios Zarpalas

Petros Daras

Centre for Research and Technology Hellas
Information Technologies Institute
6th Km Charilaou-Thermi Road

{tofis,vretos,zarpalas,daras}@iti.gr

ABSTRACT

This paper investigates the use of 3D information in the context of sports training. More specifically, a weightlifting athlete attempt is monitored, helping any coach to train athletes as she/he prefers. Our aim is to detect, collect and extract every useful data which give insight about the body technique during the weightlifting attempt and therefore make all the necessary calculations. A Kinect sensor is used for tracking the weightlifter and collecting depth data during the attempt. Afterwards, data are processed, so that useful information is extracted. In addition, after any attempt, data can be saved and loaded at any time. 2D and 3D graphs are used to illustrate the relevant information. Finally, two attempts can be loaded in parallel for the comparison between two different athletes or between a new and an old registered attempt, giving the ability for matching and correcting different techniques.

General Terms

Algorithms, Design, Experimentation, Human Factors.

Keywords

Weightlifting coach, biomechanics, Kinect sports, depth sensing, sport attempts comparison.

1. INTRODUCTION

Biomechanics is defined as the sport science field that applies the laws of mechanics and physics to humans and is used in order to gain a greater understanding of performance in athletic events through modeling [1]. In biomechanics, the structure and function of biological systems such as humans and animals is studied by means of the methods of mechanics [2]. The position and the orientation of an athlete in sport biomechanics are vital information when assessing performance, injury risk and joint loading. Although, obtaining such information quickly and accurately has been an active research area, complicated routines for the calibration of skeletons for tracking are time-consuming and can introduce fatal errors if not performed correctly, while an athlete moves her/his body to perform a complicated action.

Accurate monitoring of the training load can help coaches to improve the preparation of their athletes. The utilization of videos, to watch athlete attempts, has been proven

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

Laval Virtual VRIC '13, March 20-22, 2013, Laval, France.

Copyright 2013 ACM 978-1-4503-1243-1 ...\$10.00

a useful tool for a coach to detect and correct insignificant or important mistakes in different weightlifting techniques. Frequently, more than one cameras are used, so that the coach tries to watch the full motion from different angles [3]. Afterwards, an effort is made to identify mistakes in lifting velocity, certain angles and poses in order to explain the necessary changes for improvements. Additionally, applications as the "Weightlifting Video Overlay System" [4], use sensors on the weightlifting bar in order to acquire data for the weightlifting attempt. More specifically, a ring with two fine stainless steel cables is positioned on both ends of the barbell and each of these two cables attaches to a positioning sensor on a frame mounted above the lifting platform. The trajectory paths and velocity values of both ends of the barbell are determined by the positioning sensors from the movement of the cables during the lift. By using two positioning sensors for each ring, up-down and front-back motions of the bar are measured. On the other hand, Kinect sensor offers not only much more complete and compact information but also no setting-up equipment on the bar or on the athlete. Skeleton detection can provide information about the position of the body at any time. This information gives the ability to calculate accurately most of the physical quantities which improve remarkably the technique of an attempt [5].

In the present work, the sport which is investigated is weightlifting. Weightlifting is a sport in which participants attempt a maximum-weight single lift of a bar, which is loaded with extra weight. The two competition lifts are the 'snatch' and the 'clean and jerk' [6]. By using the Kinect sensor, the detection of the human structure is succeeded and the necessary information is collected. More specifically, by using skeleton detection of the human body, the position of the human joints is detected in real-time, in order to create the base for generating the rest advanced data. As a result, the coach is able to control and check every detail in the technique of her/his athlete.

The novelty of this approach is primarily the analysis of the athlete body in 3-dimensions. Therefore, the way the body and its motion are managed is completely changed, just by recording with suitable cameras and computing the information correctly. Moreover, joints positions from skeleton tracking are the basic seeds for the information that needs to be calculated.

The remainder of this paper is organized as follows: in Section 2, skeleton tracking and joints information are analyzed. The way the data are processed in order to obtain the advanced information necessary for the coach is presented in Section 3. In Section 4, the graphical user interface (GUI) is presented. Finally, conclusions are drawn in Section 5.

2. SKELETON DETECTION AND TRACKING

As already mentioned, the main sensor that is used for the body – skeleton tracking is the Kinect sensor. Kinect is a depth sensing device by Microsoft and PrimeSense, which works with the principal of structured light in order to get depth data [7]. Additionally, it is capable to track the body skeleton and get information about the position and the orientation of several joints.

OpenNI SDK [8] in combination with NiTE middleware [9] are used for programming the Kinect sensor. In OpenNI and NiTE, algorithms for exporting skeleton data from depth data are provided, so that, after acquisition of the depth data, the human body is detected and the skeleton can be calibrated [10].

Once the skeleton is calibrated, the whole information for the human body can be collected and ready to be used. The joints that can be identified by the sensor are twenty. Namely, the head, the neck, the shoulders, the elbows, the wrists, the hands, the spine, the center hip, the hips, the knees, the ankles and the feet [11] as shown in Figure 1. The Kinect sensor is able to detect their positions and their orientations. As a result, there is a 3D point associated to each joint, whose origin is the position of the Kinect sensor. Moreover, an orthonormal matrix represents the rotation between the joint local coordinates (where the origin is the hip joint) and the world coordinates.

It is important that the IR depth-finding camera operates at a rate of 30 Hz. This makes possible to work our data in real time and to find linear and angular velocity of every part of the body. Further, human recognition is based on the depth information and also taking into consideration the proportions of different body parts [12].

Since the position of every joint is established and the data refresh at 30 Hz, our algorithms can produce the necessary new data for the specific application. In every frame we have 20 new 3D points which are the positions of the joints. It is clear that simple vector operations can be used to create new vectors, 3D points, angles and distances. All these data, in combination with previous and next data in every depth frame, create a palette of physical quantities, useful in the context of weightlifting.

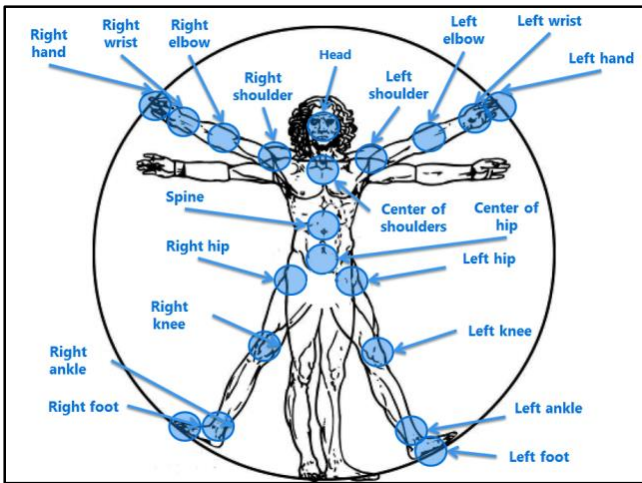


Figure 1. Joints detected from Kinect Sensor

3. WEIGHTLIFTING DATA

During the skeleton tracking the retrieved information is valuable in order to calculate our metrics. It is necessary to filter our data in order to take only the useful information. In order to filter correctly the retrieved data, it is essential to specify our needs.

The important information for a coach is: the position, the angle and the velocity of the weightlifting bar, the initial angle of the athlete's knee as well as the start time and the end time of the attempt [13]. Since, this is the information that coaches need, we have to calculate, produce and finally illustrate them correctly.

3.1 Weightlifting Bar Position

A weightlifting attempt is considered successful when the bar is on top of the athlete and she/he keeps it balanced for some time. As it is obvious, the bar is of great importance in weightlifting, so its tracking is necessary for having correct and useful information. Thus, the position, the angle and the velocity of the weightlifting bar have to be known at any time during the attempt, so as to be monitored by coach.

The weightlifting bar is connected with the athlete's hands. An athlete catches the weightlifting bar and lifts it symmetrically, so that one hand is exactly as far as the other from the center of the weightlifting bar. As mentioned before, the right and left hand are detected from the Kinect sensor, thus we already have information about their positions. In order to calculate the position of the weightlifting bar, we add two vectors: the first one a , which has initial point at the origin of the world coordinates (i.e., the Kinect position) and end point at the right hand joint position, and the second vector b , with initial point at the origin of the world coordinates and end point at the left hand joint position. Thereafter, as it is shown in Figure 2, the vector which is the midway half vector of a and b is a vector with end point at the center of the weightlifting bar.

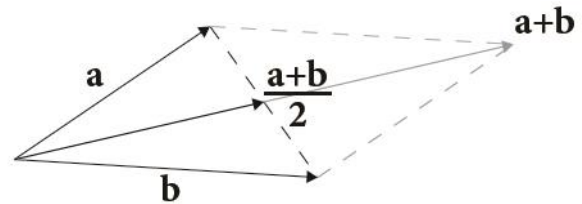


Figure 2. Weightlifting bar position calculation.

This way we get information about the weightlifting bar position. The new vector has initial point at the origin (0, 0) of the world coordinates and end point the point which is at weightlifting bar center.

3.2 Weightlifting Bar Angle

During a weightlifting attempt, the weightlifter tries to lift the weights and the weightlifting bar is turning depending on the position of her/his hands. When this angle is wide, it is too difficult for the athlete to balance the weightlifting bar and the attempt is possible to fail [14]. Monitoring the values of this angle is of primary importance.

Let this angle be θ , as it is shown in Figure 3. In order to calculate the value of θ , we have to calculate the angle between the vector c , connecting the right with the left hand, and the axis X. More specifically, we calculate the vector v (v_x, v_y, v_z), as the vector emanating from the right hand towards the left one, and then we calculate the tangent of the angle θ as:

$$\tan\theta = \frac{v_y}{v_x} \Rightarrow \theta = \arctan\frac{v_y}{v_x} \quad (1)$$

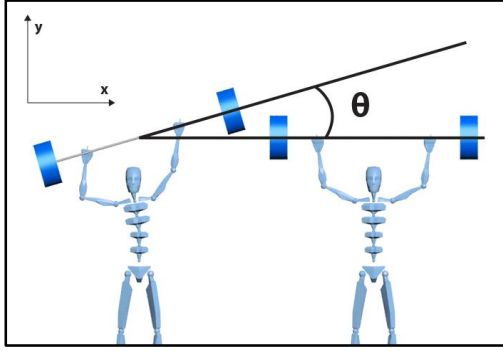


Figure 3. Angle θ at XY plane

θ is monitored in real-time during an attempt of the weightlifter and helps the coach to derive the appropriate conclusions and corrections to the athlete's technique.

3.3 Weightlifting Bar Velocity

Another important factor for succeeding a weightlifting attempt is the velocity the weightlifting bar is being lifted. The athlete produces work to lift the bar during an attempt. This work in time is the power P of the bar, $P = \frac{W}{t}$ where W is the work and t the time, which is also equal with the product of the force F and velocity u of the bar. All units in the calculations are in International System units (SI). Thus, $P = F \cdot u = \frac{W}{t}$ [15]. Athlete's technique must be based on giving as much power as needed to make a successful attempt. Neither more nor less than the necessary.

In order to compute the velocity u , we need to find the change in the weightlifting bar position in time. As mentioned before, the Kinect sensor has a refresh rate of 30 fps, therefore the capture of one frame to another lasts $t_0 = 33$ milliseconds. Then, we take the value of velocity for moving from a point A to a point B, as shown in Figure 4.

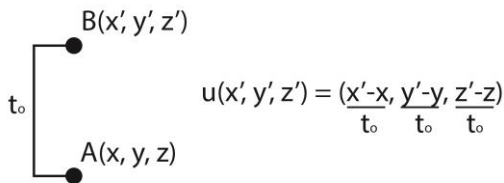


Figure 4. Transfer from point A to point B – velocity calculation

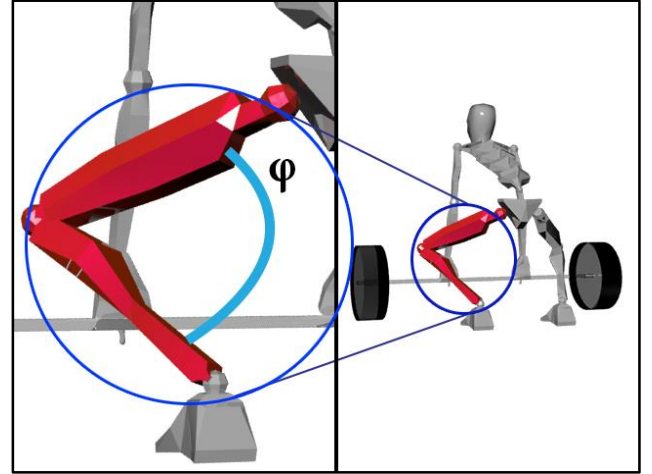
Taking the velocity at every new point is giving better the velocity at any random moment t . More specifically, weightlifting bar's velocity is computed as follows:

$$u(x', y', z') = (\text{Current Position} - \text{Previous Position}) / \text{Refresh Duration}$$

3.4 Knee Angle Calculation

In weightlifting, the muzzle angle ϕ of the knee just before the athlete starts lifting the weightlifting bar, is a quantity of high importance [16]. The angle ϕ is illustrated in Figure 5. Because of that angle, the attempt could be successful or wasted. We need to measure that angle through the skeleton joints' positions. To calculate this angle we use three 3D points of the left leg, since measuring left or right leg is practically the same since legs are positioned symmetrically. More specifically, the angle will be calculated through the scalar product between the two vectors, that connect these three 3D points. The first one is the vector that connects the point of the left knee joint position (initial point) with the point of left foot joint position (end point) and the second one connects the point of left knee joint position (initial point) with the point of the left hip joint position (end point). Let the first vector being m and the second one k , then, ϕ is calculated as:

$$\phi = \cos^{-1}\left(\frac{m_x \cdot k_x + m_y \cdot k_y + m_z \cdot k_z}{|m| \cdot |k|}\right) \quad (2)$$



where $|\cdot|$ is the Euclidian norm of a vector and m_x, m_y, m_z and k_x, k_y, k_z are the coordinates of m and k respectively.

A second part in the investigation of the initial angle is to detect the time point the weightlifter begins her/his attempt. The most appropriate moment to calculate the angle is when the athlete starts moving the weightlifting bar and the weightlifting bar speed starts increasing. To avoid mistakes during the acquisition of the data, an algorithm ensures that the angle is computed at the right moment. This algorithm checks the angle of the knee to be in a specific range between 120 – 160 degrees, the hands are on the bar and the athlete starts lifting the bar, so that she/he starts her/his attempt.

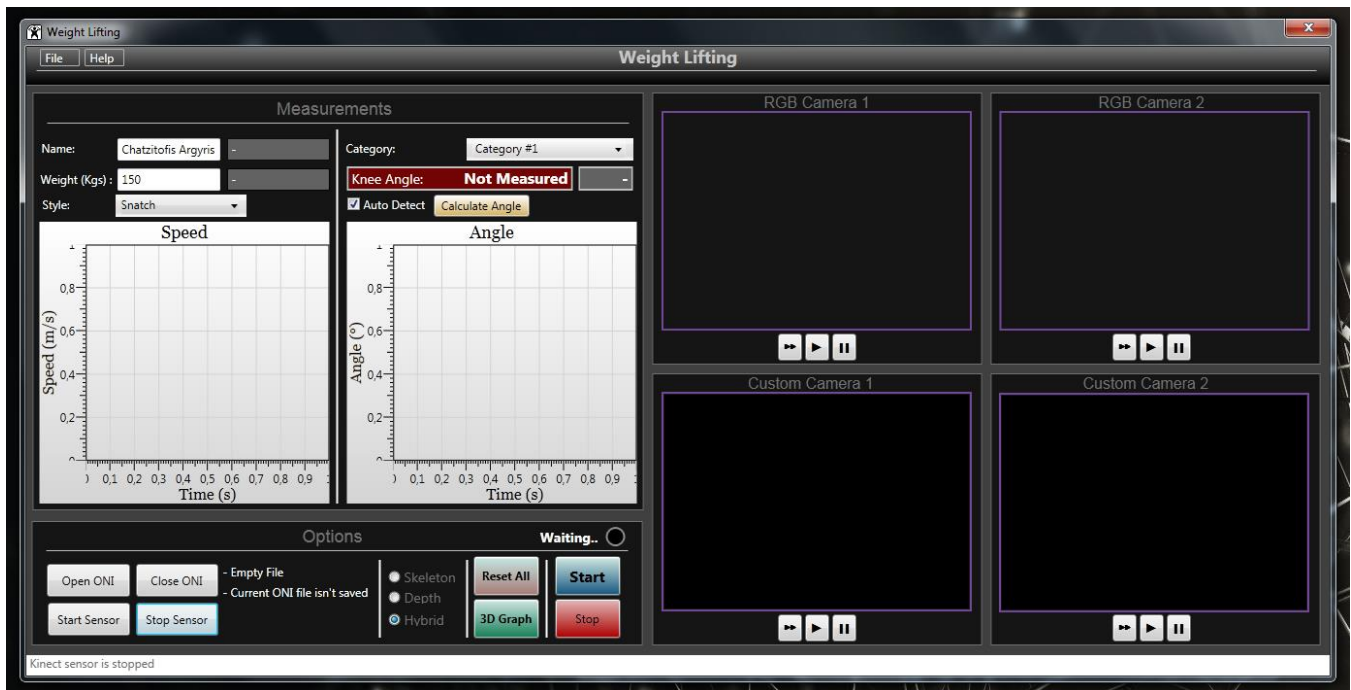


Figure 6. Application Main Environment.

4. APPLICATION ENVIRONMENT

The proposed application aims at helping weightlifting coaches to improve the way they train their athletes, as well as to assist them to organize their work. Therefore, the user interface (UI) has to be simple and comprehensive in order to provide the information that coach needs to monitor. These are the main objectives based on which the application was designed. The application can perform the following operations: a) Automatic and Manual monitoring and recording of RGB + Depth video files b) loading already recorded RGB + Depth video files c) weightlifting bar velocity-time 2D graphs d) weightlifting bar angle-time 2D graphs e) value for the initial knee angle φ f) 3D graph of the bar path, colored depending on its velocity, and g) operations and application buttons for other uses. Our application can be segmented in 5 parts:

1. The Video Screen
2. The 2D Graphs
3. The Knee Angle Frame
4. The 3D Graphs
5. Operations and Buttons

The application main environment is shown in Figure 6.

4.1 The Video Screen

After starting capturing with the Kinect sensor, the coach watches her/his athlete in real time RGB video and depth video with the skeleton appearing in two different screens. In the main environment, these screens are named RGB Camera 1 and Custom Camera 1, respectively. The coach can watch the attempts in real-time at the RGB and configure the screen to watch only the skeleton data or only the depth data or both. (Figure 7)

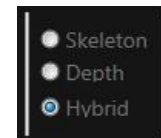


Figure 7. Screen Options

In addition, the coach is able to start and stop the sensor at any time, depending on her/his needs, to open a pre-saved attempt and watch it in the same manner as in the real-time or to load two attempts in parallel for comparison. The first loaded attempt is monitored in the right set of screens, named as RGB Camera 2 and Custom Camera 2, respectively. If the sensor is stopped, the second attempt is monitored in the left couple of screens. The screen which is designed for watching the videos in the application GUI is shown in Figure 8.



Figure 8. Loaded attempt on RGB screen

4.2 The 2D Graphs

During the attempt, two dynamic real-time 2D graphs are depicting the values of the vertical component of the velocity and the angle θ , respectively. These tools serve the coach to monitor these quantities in detail, so as to analyze or match them. Subsequently, she/he can save them or compare them with others. The comparison can be achieved easily when two attempts are loaded in parallel at the same plotter. In Figure 9, two attempts are shown.

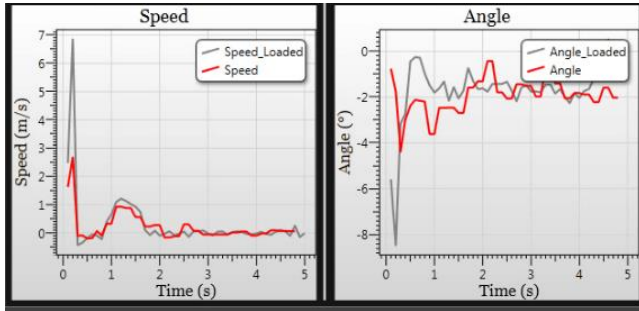


Figure 9. 2 Distinct loaded attempts on 2D graphs

In this case, the coach can focus on particular moments of the attempt to recognize what went wrong with a technique; the velocity or the angle of the weightlifting bar can be compared to a successful attempt of a specific moment. Coach can zoom in or zoom out on dynamic graphs, or drag the graph.

4.3 The Knee Angle Frame

Once the Kinect sensor is running, it is put on hold until the athlete lifts the weightlifting bar. Once the weightlifting bar starts to be lifted the application initiates the recording session. Although the athlete body is detected from the moment the weightlifter enters the field of view of the camera, recording of the attempt starts only when the athlete begins the lifting and stops when the attempt is considered complete. This occurs when the position and the motion of the athlete body signal the start. Thus the coach is not obliged to be on a computer in order to record the attempt. Naturally, there is the option for the coach to manually start and stop the recordings.

In the user interface, a border depicts the knee angle when the attempt starts. In parallel, when an older attempt is loaded, a second angle is shown referring to this attempt. A check box serves to select between automatic and manual detection of the beginning of the attempt, thus when it is checked, the program detects automatically the time the attempt

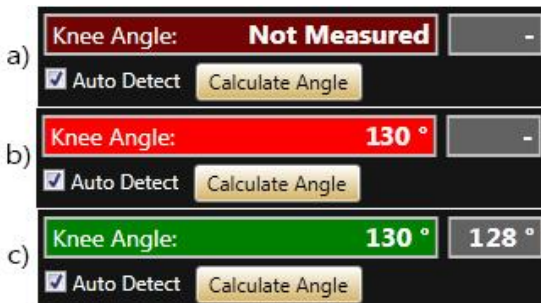


Figure 10. a) The program waits the athlete to begin. b) A new attempt is running. c) An attempt is loaded and the last attempt has finished.

starts and stops. On the other hand, when the checkbox is unchecked and the attempt is recorded manually, the 'Calculate Angle' button serves to calculate the knee angle. Figure 10 shows the different configurations of the knee angle frame.

4.4 The 3D Graphs

For a better interpretation of the weightlifting bar path, we introduce a novel visualization. The complete path of the weightlifting bar in the 3D space is constructed for comparison between paths of different attempts. In this graph, two 3D shapes are designed to show the position of the weightlifting bar in the 3D space, and therefore, make monitoring much more comprehensive. In addition, the graphs are colored with different colors to provide information about the velocity of the bar at any position. The colors belong to a gradient between green and orange color. The range of the norm of the velocity is between 0.1 to 10 meters per second. The 3D graph is designed in the center of a 3D space where the coach can navigate between different viewpoints to check position and velocity. Figure 11, illustrates these two 3D graphs.

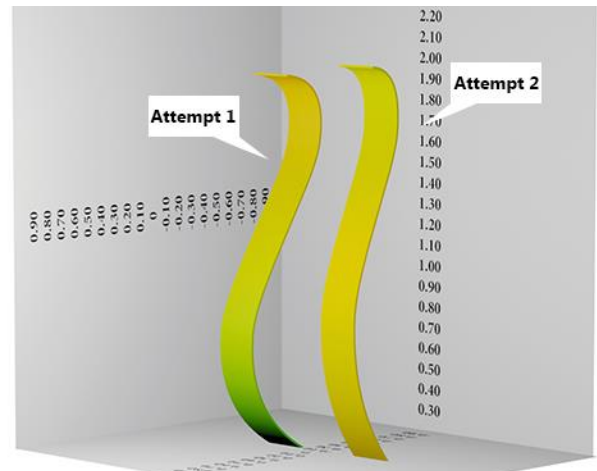


Figure 11. Two different 3D graphs, for each attempt, in 3D space

4.5 Operations and Buttons

For completing the description of the user interface, some secondary operations are also presented.

When the coach is ready to record an attempt, she/he can provide the athlete information for the specific attempt. The coach has to provide the name of the athlete, the category to which the weightlifter belongs, the weight of the weightlifting bar and the style of the lift (i.e., 'snatch' or 'clean and jerk'). The corresponding frame is shown in Figure 12.



Figure 12. Athlete and Attempt information

The main management of the attempt data takes place in 'File' menu. The 'File' menu includes important operations such as saving the data of an attempt by clicking on 'Save Attempt' and respectively load them by clicking on 'Open Attempt'. Additionally, the choice 'Save Attempt' also executes

the operation for saving the attempt in ONI file format [17] and 'Open Attempt' loads ONI files. Finally, clicking on 'Close Attempt', forces the last loaded session to close. 'Exit' terminates the application. The 'File' menu can be viewed in Figure 13.

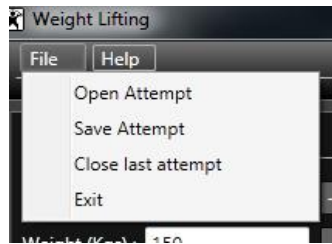


Figure 13. 'File' menu options

Another part of the user interface includes the buttons used for the management of the content. It consists of four buttons, which manage the sensor and the loaded videos and two text blocks. In Figure 14, this part is highlighted.



Figure 14. Buttons for video content management

The 'Start Sensor' button starts the Kinect sensor and initiates the RGB and the depth real-time streams on two left screens. The 'Stop Sensor' button stops the Kinect sensor and allows the coach to load a video file. The 'Open ONI' button loads RGB and depth video data on two right screens; under the condition that the sensor is not running, by clicking this button again, the coach can load a second attempt in parallel on two left screens. The 'Close ONI' button closes the ONI files one by one, starting from the last one. Concluding, two text blocks give information about the ONI files. The first one informs the coach if an attempt is loaded or not and the second if the last attempt video is saved or not.

In the last part of the user interface, a group of four buttons which execute several operations are shown. (Figure 15)



Figure 15. Manual record, 3D Graph and reset.

The 'Start' and 'Stop' buttons are used for the management of the manual recording of the attempt. More precisely, when the automatic operation is off, the coach decides when the attempt starts and stops by clicking on these buttons. '3D Graph' shows the environment where the paths of the attempts are shown. 'Reset All' discards all the loaded data.

5. CONCLUSIONS

A biomechanical software is developed by using a depth sensing input device, i.e., the Kinect sensor. Through this software, weightlifting coaches are served to improve their athlete techniques. Essentially, an "electronic" coach assistant is created, for the sport of weightlifting. As it is already mentioned, simple vector equations along with a useful and

comprehensive user interface synthesize a helpful tool for weightlifting coaches. A coach is able to monitor the attempts of her/his athlete and information such as: the velocity, the angle and the position of the weightlifting bar at any specific time, the angle of the knee in degrees. Additionally, the coach can keep history files for her/his athletes with their attempts one by one. In conclusion, new software techniques combined with depth sensors are used to improve sports coaching and more specifically weightlifting coaching.

ACKNOWLEDGEMENT

The research leading to these results has received funding from the European Union Seventh Framework Program (FP7/2007-2013) under grant agreement no 287966 (EXPERIMEDIA). The publication reflects only the authors views. The EU is not liable for any use that may be made of the information contained there.

REFERENCES

- [1] Bartlett, Roger (1997). *Introduction to sports biomechanics*. New York, NY: Routledge. p. 304. [ISBN 0-419-20840-2](#)
- [2] R. McNeill Alexander (2005) *Mechanics of animal movement*, *Current Biology* Volume 15, Issue 16, 23 August 2005, Pages R616-R619.
- [3] Daniel Casey, MS, CSCS <http://purefitclub.com/index.php/Articles/olympic-lifting> Olympic Weightlifting: Methods and Application
- [4] Olympic Coach Magazine, Fall 2004, vol. 15, no. 3 <http://www.teamusa.org/About-the-USOC/Sport-Performance/Coaching-Education/Coach-E-Magazine.aspx>
- [5] Escamilla, R.F. and John Garhammer. "Biomechanics of Powerlifting and Weightlifting Exercises." Exercise and Sports Science. Eds. Garrett and Kirkendale. Lippincott, Williams and Wilkins, 2000. 585-615.
- [6] ["IWF Technical and Competition Rules"](#) (PDF). International Weightlifting Federation. Retrieved 2009-08-10
- [7] ["Project Natal" 101](#)". Microsoft. June 1, 2009. Archived from [the original](#) on June 1, 2009. Retrieved June 2, 2009.
- [8] ["About"](#). *openni.org*. Retrieved 5 Jan 2011.
- [9] Mitchell, Richard (December 10, 2010). ["PrimeSense releases open source drivers, middleware for Kinect"](#). *Joystiq*. Retrieved December 11, 2010.
- [10] NiTE Middleware (December 19, 2012) <http://www.primesense.com/solutions/nite-middleware>
- [11] ["Play.com \(UK\) : Kinect : Xbox 360 - Free Delivery"](#). Play.com. Retrieved July 2, 2010. "This information is based on specifications supplied by manufacturers and should be used for guidance only."
- [12] M Peddi, A., Huan, L., Bai, Y., & Kim, S. (2009). Development of Human Pose Analyzing Algorithms for the Determination of Construction Productivity in Real-time. Building a Sustainable Future - Proceedings of the 2009 Construction Research Congress (pp. 11-20). Seattle, WA, United states: American Society of Civil Engineers..
- [13] Weightlifting Equipment and History <http://www.olympic.org/weightlifting-equipment-and-history>

- [14] Michael Janofsky (November 16, 1987). "[Olympic Notebook: 7 Unlikely Nations Join Winter Games](#)". *The New York Times*. Retrieved 2009-10-02.
- [15] NASA Official: [Tom Benson](#) Last Updated: Jul 11 2008
<http://www.grc.nasa.gov/WWW/k-12/airplane/powr.html>
- [16] Bai, X. and H. Wang. "Three-dimension Kinematics Simulation and Biomechanics Analysis of Snatch Technique." Proceedings of 1st Joint International Pre-Olympic Conference of Sports Science & Sports Engineering Volume I: Computer Science in Sports (2008): 291-296.
- [17] "Programmer's Guide" File Devices
<http://www.openni.org/openni-programmers-guide>