Movement Estimation Using Mediapipe BlazePose

Ainun Syarafana Binti Pauzi, Firdaus Bin Mohd Nazri, Salisu Sani,
Ahmad Mwfaq Bataineh, Muhamad Nurul Hisyam, Mohd Hafiidz Jaafar,
Mohd Nadhir Ab Wahab, and Ahmad Sufril Azlan Mohamed(✉)

School of Computer Sciences, Universiti Sains Malaysia, Minden, 11800 Penang, Malaysia
sufril@usm.my

Abstract. The paper describes a system to track the body movement of a person from a video source while augmenting the labelled skeleton joints onto the body of the person. This work has endless applications in the real world especially in the physical-demanding working environment as well as in the sports industry by implementing deep learning, the techniques can recognize the joints on a person's body. An algorithm namely Mediapipe Blazepose has been applied using PoseNet dataset to detect and estimate curated movements specifically designed for body injury during heavy workload. The propose method has been compared to IMU based motion capture and the difference accuracy is within 10% since IMU capture real data of the sensors while the deep learning method using 2D image analysis. The expected outcome from this project is a working system that is able to correctly identify and label the skeleton joints on a person's body as well as perform various calculation such as movement velocity and the angle of joints which could be crucial for determining whether certain body movements could result in injuries either in the short- or long-term period.

Keywords: Marker-less · Motion capture · Deep learning · Marker-based

1 Introduction

In the current augmented reality field, there exists numerous marker-based motion captured software which uses markers which are attached to a suit that has to be worn by the subject. These markers act as sensors which will give input to the motion capture software which will then calculate the subject's location and thus displaying the subject to the display screen [1]. This model of motion capture is costly as it requires designated tools such as the markers as well as specialized camera. Although the marker-based system cost is high, the accuracy is very satisfactory [2, 3].

There are many applications such as Microsoft Kinect which uses Time-of-Flight (ToF) and other sensors which uses Dynamic Time Warp (DTW) that measures the distance of IR sensors and the time it takes to return to the sensor due to its reflections [4, 5]. However, the data captured consists of heavy noise as the data captured everything seen from the IR sensor and camera [6]. Other technology such as Inertia Measuring Unit

H. Badioze Zaman et al. (Eds.): IVIC 2021, LNCS 13051, pp. 562–571, 2021.
https://doi.org/10.1007/978-3-030-90235-3_49

(IMU) has the advantageous of capturing only the needed movements of the sensors; gyroscopes and accelerometer, in real time however, due to its hardware dependent system, a dedicated connection is needed, and sensors can be influenced by wireless reception and earth gravity giving some noise or error in detection. IMU is very useful as the system not required visual field to tracking a motion [7] (see Table 1).

Table 1. Comparisons of marker-less motion capture.

Software	Strength	Weakness
Microsoft Kinect [15]	Users are more free to explore the functionalities and can create their own variation of product	No support from Microsoft. User needs to tweak software on their own
Kinetisense [16]	The screen can be customized from a wide selection of 75 assessment which could prove beneficial for a more comprehensive workflow	Offers assessment for performance and sport but nothing beyond it. The focus of the application is just the analysis of motion
Human trak [17]	Real-time display of important data overlayed on the user. Among the data that could be displayed are the joint range of motion and balance metrics	For the top-tier hardware and performance, users are looking at an expensive yearly subscription when compared with other options in the market
Inertia measuring unit [7]	Uses real-time sensors capturing from gyroscope and accelerometer	Need dedicated network connection and prone to earth gravity noise

Therefore, the aim of this work is to explore a marker less motion capture method that is able to perform skeleton joint detection using 2D images that are more accessible to the general public. This would reduce the cost of motion capture as tools that are readily available [8].

2 Related Works

The existing system in the market also has a hard time detecting limbs that are performing fast paced movement or even subjects that are equipped with loose clothing. Furthermore, if the subject is placed far away from the camera, this will also result in poor tracking of their joints [9]. Another problem that existing systems have is detecting torso bending which is due to the systems unable to perform depth estimation correctly from the images obtained from the front part of the body and the inability to sense the back part of the body [10, 14].

Apart from that, the current existing system in the market does not provide an indication when a risk-prone injury movement is being performed by a subject. The risk-prone movement needs to be identified by the user of the system by further analysis using the data obtained from the system. The use of deep learning able to eliminate the complication of the hardware and by using the 2D digital image, features can be extracted and accurately in recognizing the human structure while using a suitable human pose dataset [11, 12]. The sparse Inertial Measurement Unit will paired with the Deep Learning model earlier gives a better estimation for the system for a more accurate detection of the joints. However, IMU [13] sensors need to be placed on subjects' body which irritates the process and constant recalibration needed due to its prone to magnetic interference.

The aim of the study is to propose a method that leads to marker-less motion capture with deep learning implementation to correctly recognized human body pose and movements. The threefold objectives that leads to the aim of the study are mainly focusing on integrating deep learning model for correctly estimating the movement of the body joints, to measure the distance between one joint to another creating the skeleton frames with each joint record individual velocity and angle at every successive frames, and to compare the accuracy between the proposed marker-less method and the marker-based motion capture.

3 Methodology

This proposed method aims to automatically give an indication during a risk-prone movement is being perform by a subject. Apart from that, existing motion capture that is in the market has a hard time capturing the motion of limbs accurately especially when subject is equipped with loose clothing or during subject performing motions that are fast paced. Furthermore, if the subject is placed far away from the camera, this will also result in poor tracking of their joints. In addition, movement such as torso bending makes the existing system of motion capture have a hard time detecting these movements accurately.

3.1 Application Architecture

The architecture of the proposed method is shown in Fig. 1 below, where the system is divided into 2 major subsections which are the front-end and the back-end. The front-end of the system involves in getting user input as well as displaying the necessary information to the user. The back-end of the system is responsible in processing the input provided by the user to produce an understandable output to be used by the user for further analysis. The underlying modules contained within the application architecture will be explained in the same order from the user of the system launches the program until the user exits the program.

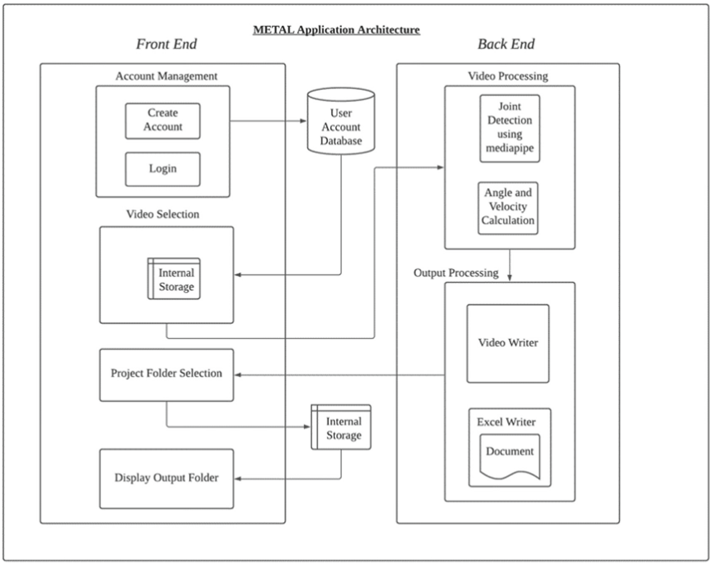


Fig. 1. System architecture diagram

3.2 Video Processing

The main focus of the method is the Video Processing module. The input video will be processed frame-by-frame, meaning that each frame of image of the video will be processed independently. The system will first take the first frame of the video and employs the holistic functionality of Mediapipe BlazePose. This package is a pretrained model package which was developed by Google. The difference between this model as compared to other models in the market is that mediapipe is able to accurately track human body pose almost in real-time. Mediapipe also offers more key points (33 key points) as compared to other body pose detection model which generally are built based on Common Objects in Context (COCO) topology (17 key points). Figure 2 below illustrates the pose detected by BlazePose model. The COCO topology is colored in green whereas the blue key points are the included key points offered by BlazePose model [12].

The difference in the number of key points is among the factors that allows Mediapipe to process the image input almost in real-time. For the sake of our system, we will only display 12 different key points which have been determined to be the most vital key points for analysis of body angle. Body parts such as feet, hand and face which are also detected by Mediapipe BlazePose are ignored by the calculation modules since they are not related to angle calculation.

Mediapipe's BlazePose algorithm works by utilizing two-step-detector-tracker Machine Learning pipeline. The pipeline will first locate the person's region-of-interest (ROI) within the frame. Once the ROI is determined, the tracker will predict the pose landmarks within the ROI. In the system's case, the detector will be invoked only in the first frame. For subsequent frames, the pipeline will derive the ROI of the new frame

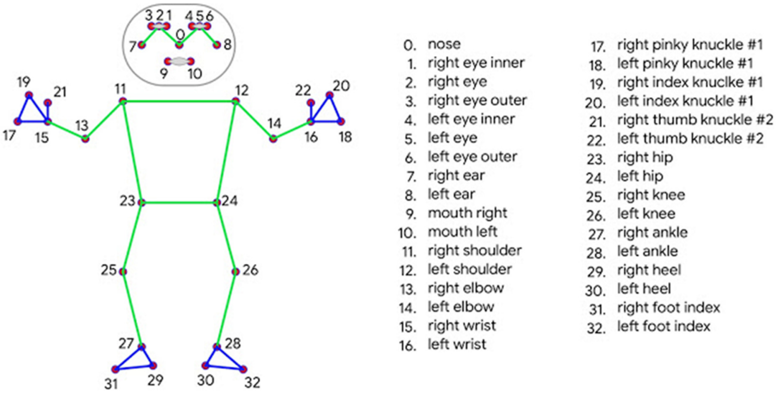


Fig. 2. COCO model in green, BlazePose model is inclusion of blue key points (Color figure online)

from the previous frame's pose landmarks. This is also one of many reasons that allows Mediapipe BlazePose to compute the landmark poses in almost real-time. The Pose Detection model of Mediapipe BlazePose is trained from an image dataset containing around 85,000 images including 30,000 of the images obtained from consented images of people using a mobile AR application captured with smartphone cameras in various "in-the-wild" conditions.

The model is used for predicting the human body center (middle of hip). Once the body center is determined, Mediapipe will use the body center to determine the pose landmarks located within the radius from the body center. Once the pose landmarks have successfully detected by Mediapipe BlazePose, the results will be stored in variables to denote the x and y coordinates of the respective pose landmark. Each pose landmark of a body part will be stored in different variable which will be used later for angle and velocity calculation.

Once the x and y coordinates have been obtained from the joint detection module, the values of the x and y coordinates will be passed to the angle and velocity calculation module. This module will perform mathematical computation based on the coordinates of the joint-of-interest and its 2 respective adjacent body parts connected to the joint-of-interest. This calculation is performed based on the mathematical principle called "The Law of Cosines" as shown in Eq. (1).

$$a^2 = b^2 + c^2 - 2bc\cos A \tag{1}$$

We can see that the formula (1) is used to calculate the length of side a given that the information of side b and c as well as angle A is known. For the case of our system, we need to rearrange the equation to a different form since we are interested in finding the angle A given that we have the values for side a, b and c. Rearranging the equation above for our system's will yield us the equation below in Eqs. (2–4).

$$2bc\cos A = b^2 + c^2 - a^2 \tag{2}$$

$$\cos A = \frac{b^2 + c^2 - a^2}{2bc} \tag{3}$$

$$A = cos^{-1}\left(\frac{b^2 + c^2 - a^2}{2bc}\right) \tag{4}$$

Now that the equation to compute the angle has been determined, we just have to pass the necessary arguments to the calculation module function.

To understand the approached used in the module, we will assume that we are interested in the angle of the right elbow labelled “13” in Fig. 2. The adjacent connected body parts of the right elbow are the right wrist labelled “15” and the right shoulder labelled “11”. We can see visually that these 3 points make up a triangle which is why this particular principle can be used to calculate the angle of joint-of-interest. Since the joint-of-interest is the right elbow, we would need to find the distance between point “15” of the right wrist and point “11” of the right shoulder and denote it as a for the rearranged equation from The Law of Cosines.

The variable **b** of the equation will be taken from the distance between point “11” of the right shoulder and point “13” of the right elbow. Consequently, the c of the equation will be taken from the distance between point “13” of the right elbow and point “15” of the right wrist. The value of b and c can be interchanged since it does not affect the value of the angle. However, the value of a in the equation needs to be the value of the distance between the two adjacent connected body parts since it is the joint-of-interest. The distance between the two points can be calculated using the mathematical concept of the Distance Formula as shown in Eq. 5.

$$d = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \tag{5}$$

Since we had acquired the x and y coordinates of each joint from the joint detection module above, we can use these values to compute the distance between them. The distance value computed will be stored in a separate variable to be used by the angle calculation module above. For instance, we would need to pass the x and y coordinates of point “11” of the right shoulder and point “15” of the right wrist to the distance formula to get the distance between them. Once all the necessary distance variables have been calculated, we then pass the values to compute the angle of the joint-of-interest. For the velocity calculation, we employ the formula as Eq. 6.

$$v = \frac{d}{t} \tag{6}$$

where, v = speed, d = distance travelled and t is time taken.

From the equation above, we can see that in order to determine the velocity of a joint, we would need to determine the distance travelled in a given second. To do this, we can apply the same equation for distance above. For instance, if we were to compute the velocity of the right wrist labelled number “15” in a particular second, we would need to know the distance that the joint has travelled in a second. Since we already know the location of the particular joint in *0th* second and 1st second, we can use the distance formula to calculate the distance travelled by that particular joint in one second. However, the distance computed is the distance in terms of pixel. Thus, we need a reference of an object that we know the real-life length. The purpose of the real-life reference is for us to obtain a ratio between the pixel distance to the real-life distance. This is done by measuring beforehand the real-life distance between the right shoulder and the right hip of a participant. Once this length has been determined, we can set a variable of this constant. The next step is to measure the pixel distance between the right shoulder and the right hip of the participant which could be done by finding the distance between these two points that we have obtained in the joint detection module earlier. These two points will then be divided to obtain a ratio which will be used to multiply with the pixel distance travelled by a joint to obtain the real-life distance value travelled by that particular joint. This process is done on 4 different joints which are the left wrist, right wrist, left ankle and the right ankle. These 4 joints are chosen for the velocity calculation due to the nature that these joints are the joints identified to move at a higher speed and are more prone to injury due to speed.

3.3 Output Processing

After the angles and velocities of the joints has been calculated, these values will then be displayed on the respective image that was processed. Simultaneously, while the system is processing the frames in a video until it reaches the end, the calculated angles and velocities will be stored in an array. Once the frame processed reaches the end, this array will then be stored using 2 arrays, the angles array and the velocity array as input.

Since the reliability of the method will be based on the “gold standard” of the Inertial Measuring Unit (IMU), the calibration of the sensors are calibrated for offsets, scale factors and alignment errors in x,y and z-axes which can be formalized as:

$$w^{'} = C_w S_w w^{'} + b_w \tag{7}$$

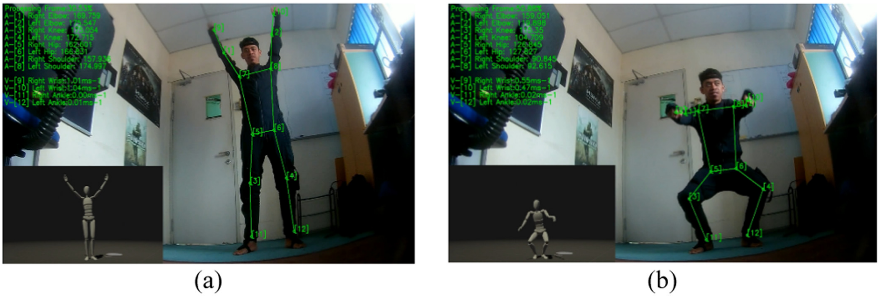
$$a^{'} = C_a S_a a^{'} + b_a \tag{8}$$

where, w’, a’ are the true angular velocity and acceleration, b_w, b_c are the biases, C_w, C_a are the rotation matrices representing the misalignment between the actual and nominal sensitivity axes of the sensors, and S_w, S_a are the diagonal matrices containing the scale factors of the three axes of each sensor [18]. The calibration needs manual adjustment until the IMU sensors are aligned to the markers preset by the system. A controlled movements are needed so offsets measurement can be done by aligning the 2D plots of the proposed method with the 3D plots of the IMU (without the Z-axis).

4 Results

Comparisons were made between the proposed method and Rokoko Smartsuit (using IMU). Since Rokoko Smartsuit comes with real-time sensor reading of gyroscope and accelerometer which is the basis of IMU components, the data generated by the Rokoko smartsuit is used as the gold standard (Fig. 3).

The capture data are then compared frame-by-frame and evaluated based on its velocity, angles and x-y coordinates. The data that will be used for comparison is determined to be compared by each second. This means that for each second that passes by from



```
Processing Frame:30.53%
A-[1] Right Elbow: 169.759
A-[2] Left Elbow: 175.547
A-[3] Right Knee: 175.054
A-[4] Left Knee: 172.715
A-[5] Right Hip: 162.601
A-[6] Left Hip: 166.831
A-[7] Right Shoulder: 157.936
A-[8] Left Shoulder: 174.997

V-[9] Right Wrist:1.01ms-1
V-[10] Left Wrist:1.04ms-1
V-[11] Right Ankle:0.00ms-1
V-[12] Left Ankle:0.01ms-1
```

```
Processing Frame:50.88%
A-[1] Right Elbow: 159.051
A-[2] Left Elbow: 119.898
A-[3] Right Knee: 105.35
A-[4] Left Knee: 104.709
A-[5] Right Hip: 126.845
A-[6] Left Hip: 127.827
A-[7] Right Shoulder: 90.845
A-[8] Left Shoulder: 92.615

V-[9] Right Wrist:0.55ms-1
V-[10] Left Wrist:0.47ms-1
V-[11] Right Ankle:0.02ms-1
V-[12] Left Ankle:0.02ms-1
```

Fig. 3. The figure shows the comparisons between the proposed method and IMU where (a) is the initial pose with both hands raised up. The lower left is showing the IMU representation. (b) is the squatting position with both hands raised straight towards the camera. The camera is capturing simultaneously with IMU to get accurate readings.

the input video, an average value of a particular joint will be computed in order to be compared from the two sets of data. This is done because the output data that Rokoko provides comes at 100 frames per second whereas our system takes the video recording from a camera that is able to record only at 60 frames per second. To get an accurate representation for the comparison, an average value within one second will be taken. Table 2 shows the mean difference (%) between the data generated from the proposed method.

Table 2. Mean differences (%) between the proposed method and IMU.

Joint	Mean difference (%)
Right elbow	8.06
Left elbow	7.72
Right knee	10.99
Left knee	6.30
Right hip	0.19
Left hip	0.52
Right shoulder	23.67
Left shoulder	24.57

The table shown above exhibits the mean difference of approximately mean difference of 10% overall with the shoulder having the worst mean differences (Right: 23.67% and Left: 24.57%). The differences at the shoulders exist when each shoulder and arm are at the same angle which confuses the algorithm to estimate the joints' location.

5 Conclusion and Future Work

Based on the findings and interpretation of the results from the analysis performed on the developed system, it is apparent that proposed method has meet the functional and non-functional requirement that has been determined beforehand. The algorithms and implementation used has been selected properly and accurately for marker-less motion capture analysis.

In the future, improvements could be made to measure at two different perspectives and videos to be processed simultaneously and could provide a better output data for a more accurate representation for the angles and velocity calculations. Another improvement that could be adopted is to auto clean wrong joint detection made due to noise of the image or occlusion of one arm to the other side.

Acknowledgement. The authors are grateful to the Ministry of Higher Education Malaysia for Fundamental Research Grant Scheme with Project Code: FRGS/1/2020/STG07/USM/02/12 for supporting this documented work.

Conflicts of Interest. The authors declare that they have no conflicts of interest to report regarding the present study.

References

1. Eldar, R., Fisher-Gewirtzman, D.: Ergonomic design visualization mapping- developing an assistive model for design activities. Int. J. Ind. Ergon. **74**, 102859 (2019)
2. Maurice, P., et al.: Human movement and ergonomics: an industry-oriented dataset for collaborative robotics. Int. J. Robot. Res. **38**(14), 1529–1537 (2019)
3. Yunus, M.N.H., Jaafar, M.H., Mohamed, A.S.A., Azraai, N.Z., Hossain, M.: Implementation of kinetic and kinematic variables in ergonomic risk assessment using motion capture simulation: a review. Int. J. Environ. Res. Public Health**18**, 8342 (2021)
4. Bortolini, M., Gamberi, M., Pilati, F., Regattieri, A.: Automatic assessment of the ergonomic risk for manual manufacturing and assembly activities through optical motion capture technology. Procedia CIRP **72**, 81–86 (2018)
5. Zhang, Z., Fang, Q., Gu, X.: Objective assessment of upper-limb mobility for poststroke rehabilitation. IEEE Trans. Biomed. Eng. **63**, 859–868 (2016)
6. Ong, Z.C., Seet, Y.C., Khoo, S.Y., Noroozi, S.: Development of an economic wireless human motion analysis device for quantitative assessment of human body joint. Measurement **115**, 306–15 (2018)
7. Fletcher, S.R., Johnson, T.L., Thrower, J.: A study to trial the use of inertial non-optical motion capture for ergonomic analysis of manufacturing work. Proc. Inst. Mech. Eng. Part B: J. Eng. Manuf. **232**(1), 90–98 (2018)
8. Brownlee, J., What is Deep Learning? Accessed 25 Oct 2020. https://machinelearningmastery.com/what-is-deep-learning/ (2020)
9. Plantard, P., Auvinet, E., Le Pierres, A.S., Multon, F.: Pose estimation with a kinect for ergonomic studies: evaluation of the accuracy using a virtual mannequin. Sensors (Switzerland) **15**(1), 1785–1803 (2015)

10. Wang, X., Hu, Y.H., Lu, M.L., Radwin, R.G.: The accuracy of a 2D video-based lifting monitor. Ergonomics **62**(8), 1043–1054 (2019)
11. Schechter, S.: What is markerless Augmented Reality? Retrieved October 26, 2020. https://www.marxentlabs.com/what-is-markerless-augmented-reality-dead-reckoning/. Accessed 20 Oct 2020
12. Bazarevsky, V.: BlazePose: On-device Real-time Body Pose tracking. Accessed 18 Jun 2021. https://arxiv.org/abs/2006.10204 (2021)
13. Alessandro, F., Norbert, S, Markus, M., Gabriele, B., Emanuele R., Didier S.: Survey of motion tracking methods based on inertial sensors: a focus on upper limb human motion. Sensors **17**, 1257 (2017)
14. Mohamed, A.S.A., Chingeng, P.S., Mat Isa, N.A., Surip, S.S.: Body matching algorithm using normalize dynamic time warping (NDTW) skeleton tracking for traditional dance movement. In: Badioze Zaman, H. et al. (eds.) Advances in Visual Informatics. IVIC 2017. Lecture Notes in Computer Science, vol. 10645, pp. 669-680 Springer, Cham (2017). https://doi.org/10.1007/978-3-319-70010-6_62
15. Warren, T.: A closer look at Microsoft's new Kinect sensor. https://www.theverge.com/2019/2/25/18239860/microsoft-kinect-azure-dk-hands-on-mwc-2019. Accessed 28 Nov 2020
16. Kinetisense. What Is a Functional Movement Screening? (2020). https://www.kinetisense.com/a-functional-movement-screen/. Accessed 28 Nov 2020
17. VALD Performance (n.d.). News and Research. https://valdperformance.com/blog/. Accessed 28 Nov 2020
18. Mourkani, S.S.: IMU-based Suit for Strength Exercises: Design, Calibration and Track (Phd Thesis), Technische Universitat Kaiserslautern, Germany (2021)