Biomechanics Digital Twin: Markerless Joint Acceleration Prediction Using Machine Learning and Computer Vision

Milton Osiel Candela Leal
Mechatronics Department, School of Engineering and Sciences
Instituto Tecnológico de Monterrey
Monterrey, Mexico
A01197730@tec.mx

Dacia Martínez Díaz
Mechatronics Department, School of Engineering and Sciences
Tecnologico de Monterrey
Monterrey, Mexico
A01733799@tec.mx

Cecilia Orozco Romo
Mechatronics Department, School of Engineering and Sciences
Tecnologico de Monterrey
Monterrey, Mexico
A00827013@tec.mx

Aime Judith Aguilar Herrera
Mechatronics Department, School of Engineering and Sciences
Tecnologico de Monterrey
Monterrey, Mexico
A00828691@tec.mx

Jesús Eduardo Martínez Herrera
Mechatronics Department, School of Engineering and Sciences
Tecnologico de Monterrey
Monterrey, Mexico
A01283785@tec.mx

Arath Emmanuel Marín Ramírez
Mechatronics Department, School of Engineering and Sciences
Tecnologico de Monterrey
Monterrey, Mexico
A01651107@tec.mx

Luis Orlando Santos Cruz
Mechatronics Department, School of Engineering and Sciences
Tecnologico de Monterrey
Monterrey, Mexico
A00827603@tec.mx

César Francisco Cruz Gómez
Mechatronics Department, School of Engineering and Sciences
Tecnologico de Monterrey
Monterrey, Mexico
A00827233@tec.mx

Santiago Xavier Carrillo Ruiz
Mechatronics Department, School of Engineering and Sciences
Tecnologico de Monterrey
Monterrey, Mexico
A00831369@tec.mx

Erick Adrián Gutiérrez Flores
Mechatronics Department, School of Engineering and Sciences
Tecnologico de Monterrey
Monterrey, Mexico
A00828091@tec.mx

Karen Lizette Rodríguez Hernández
Mechatronics Department, School of Engineering and Sciences
Tecnologico de Monterrey
Monterrey, Mexico
A01197734@tec.mx

Esther Aimeé Delgado Jiménez
Mechatronics Department, School of Engineering and Sciences
Tecnologico de Monterrey
Monterrey, Mexico
A00827948@tec.mx

Ricardo A. Ramírez Mendoza
Mechatronics Department, School of Engineering and Sciences
Tecnologico de Monterrey
Monterrey, Mexico
ricardo.ramirez@tec.mx

Gerardo Presbítero Espinosa
Mechatronics Department, School of Engineering and Sciences
Tecnologico de Monterrey
Monterrey, Mexico
presbitg@tec.mx

Jorge de Jesús Lozoya Santos
Mechatronics Department, School of Engineering and Sciences
Tecnologico de Monterrey
Monterrey, Mexico
jorge.lozoya@tec.mx

Mauricio Adolfo Ramírez Moreno
Mechatronics Department, School of Engineering and Sciences
Tecnologico de Monterrey
Monterrey, Mexico
mauricio.ramirezm@tec.mx

***Abstract*—For athletes, coaches, or rehabilitation patients, the systems currently used to perform biomechanical studies and the dependence on technical experts for interpreting analyses and results can limit organizational, logistical, and economic resources. In this project, a Recurrent Neural Network model was created to predict human joint accelerations through the automatic digitalization of human body movement using video and acceleration sensors. The project aimed to prevent injuries and fractures in athletes and the elderly population because there is a lack of tools that predicts the risk of these traumas as a preventive method. Acceleration data was collected using Matlab mobile installed in cell phones attached to the arms and legs of volunteers doing physical tasks (walking, running, jumping). Experiments were video recorded, and machine learning models were trained using acceleration and video using Python libraries. After model evaluation, we observed that the selected model could predict the best on the XY axes and the worst on the Z axis, probably due to predicting a three-dimensional feature with a two-dimensional input. A biomechanical Digital Twin was created by combining the information from wearable devices, computer vision, and machine learning algorithms. This tool was able to estimate human joint accelerations (up to an extent) during movements with more refinement; it can help to evaluate movement performance within exercises or tasks and aid in injury/fracture risk prediction.**

Keywords—biometrics, machine learning, biomechanics, educational innovation, higher education

I. Introduction

A. General context

The study of body movement has helped analyses in diverse fields, especially exercise science, sports

2023 Future of Educational Innovation-Workshop Series Data in Action: Digital Ecosystem and Emerging Tools for Education | 978-1-6654-9416-8/23/$31.00 ©2023 IEEE | DOI: 10.1109/IEEECONF56852.2023.10104757

performance, and kinesiology. In such fields, there is much physical risk in high-performance activities, resulting in a high incidence of injuries and fractures due to fatigue or physical stress in the body.

B. Literature Review

A stress fracture can be defined as a "partial or complete fracture of the bone due to its inability to resist subthreshold, repeated, rhythmically applied non-visible stress" [1], and it is known that concerning the muscles around the fatigued area, excessive forces concentrate on the areas surrounding the bone, generating microcracks.

Combining biomechanical analysis and motion capture systems can potentially be used in health care, sports, older adults, or patients with musculoskeletal disorders to prevent injuries or fractures in physical tasks [2]. This work focuses on using technology (biomechanics and video) to develop tools that help people identify risk factors in their body movements that can compromise their overall health and well-being.

A trending technology of the last decade is Digital Twins (DT) (digital models of actual processes or situations using real-time information from sensors), initially in the manufacturing field [3] but expanding into other areas, rapidly in healthcare and wellness. Modern DT applications include personalized computed tomography (CT) for height and weight estimation [4], intelligent sports training systems [5], and, in general, a complete evaluation of the patient's characteristics and medical history for more effective care or treatment simulation [6]. Examples of DT technology for biomechanical purposes include the classification of (6-class) human poses with 89% accuracy using video and polynomial fitting [7]; different types of artificial neural networks (ANN) to predict low extremity (three-joint) torque during a vertical jump, and predicting 3D spinal posture during reaching and lifting movements, both with correlation coefficients above 0.95 respectively [8], [9].

C. Delimitation of the object of study

The project's long-term goal is to aid athletes or rehabilitation patients with different illnesses or injuries, especially those whose injuries can be examined, including poorly executed exercises, which are self-performed movements in sports that can lead to unexpected injuries, including fractures, where the causes vary from poor posture, inadequate technique, generation of dangerous biomechanical forces in joints, impact, among others. As a first step toward the long-term goal of this work, we created an initial model and evaluated it using data from healthy participants.

D. Problem statememt

Concerning the current availability of technology in the sports biomechanics field, athletes, coaches, and rehabilitation experts might find access to 3D camera systems and lab configuration for biomechanical assessments to be financially, logistically, and organizationally limiting. These limitations are notable when trying to configure given studies within a required space appropriately, the duration of (long) tests becomes tedious, and the material costs for capturing and analyzing the intended movements are high. In addition, analyzing the obtained information presents complications for coaches in specific sports because technical experts and their consensus are needed to interpret the results.

E. Justification

The proposed system is based on a mobile application that applies Artificial Intelligence (AI), Computer Vision (CV), and wearable devices for the acquisition of body movement biometry to create a virtual representation and a reflection of the dynamic state of a physical entity (i.e., a DT). The idea is that after training the AI model, the DT tracks the person's movement and predicts joint force acceleration using a collected video as input. This application's primary purpose would be to estimate joint fracture risk, which can be used for injury prevention and to reduce the aggravation of pre-existing medical conditions. The proposed application follows a markerless approach, in which the DT technology is convenient, as it is intended to provide instantaneous feedback as the person performs the activity or movement. It also reduces the costs of the study since special equipment or markers are not needed, the time used for the tests is shortened [10], and it becomes more accessible to the final users (athletes and the elderly).

F. Theoretical framework

Biomechanical analysis for clinical applications and sports applications employs various technologies, such as inertial or electromagnetic sensors, although using optical systems with reflective markers is the most common technique today [11]. Nevertheless, the current trend in biomechanics promises to replace motion capture using markers with high-precision algorithms that allow the use of markers to be eliminated [12]. The application of markerless algorithms offers a low-cost, automated, video-based biomechanical evaluation solution in a real context, under different environments, for instance, competitive indoor swimming [13]. In addition, the number of platforms available to film and analyze movements has increased over the years, from a very basic to a more complex level [14], providing a wide range of options.

Some of the promising solutions are a combination of AI and biomechanics. The combined use of video-based biomechanical tools with AI can help the development of intelligent methods for fracture prevention in athletes and the elderly through the prediction of fracture risk with the help of the video evaluation of body movements [15][16].

G. Objectives

Main objective: To create a markerless biomechanical system that predicts three-dimensional human joint acceleration using video and combining AI, CV, and acceleration measurements during physical tasks.

Secondary objectives:

1) Using cell phones to create a low-cost and easy-to-implement markerless motion tracking approach.

2) Obtaining motion data and video from participants during specific physical tasks (walking, running, jumping).

3) Training an AI model using the motion data and videos obtained during experiments to predict joint acceleration.

4) Evaluating different methods and proposing the best approach for video-based, markerless estimation of joint acceleration.

H. Hypothesis

The system can predict significant biomechanical features such as joint accelerations by using only information from a video and combining AI and CV into a biomechanical DT. This can be achieved by obtaining relevant key points from the human body using CV and training an AI model with "real" joint accelerations to make predictions using only the video information.

II. PROPOSAL

A. Methodology

1. Data acquisition

1.1 Test Subjects

The study's population size was a sample of 14 subjects; we sought a balanced sample (both sexes).

The study recruited 14 volunteer subjects aged 18 to 23 at Tecnologico de Monterrey. Any subject was denied participation in the study if:

- They disclosed or presented signs of musculoskeletal injuries.
- Had a body mass index (BMI) equal to or greater than 30.
- Older adults were excluded from participation due to inherent risks to the tests (movement tasks).

The subjects were informed in advance about the experimental protocol. They completed a questionnaire asking their age, gender, height, body mass, and consent for the trial. They were informed of their right to abandon the experiment at any time if experiencing any discomfort. Additionally, we requested the participants' consent to publish data and images obtained during the experiments, and their subsequent publication was requested.

1.2 Experimental Protocol

Before the experiments, each participant was interviewed to determine whether or not they qualified for the study. The staff placed the participant with the devices to measure acceleration: two smartphones, one on the right arm and the other on the right knee (see representation in Fig. 1).

The subject was placed at 1.35 meters from the video camera, which recorded the performance of the movements while (three-axial) acceleration data in XYZ axes were simultaneously acquired using Matlab and Matlab Mobile (Fig. 1). Each recording had a duration of 7 seconds with a sample rate of 30 frames per second (Fps). The recorded movements were 1) walking, 2) running, and 3) vertical/horizontal jumping; three repetitions of each movement per subject were captured, always in that order. One-minute breaks were placed between recordings of the same participant. During the procedure, the subject was asked to act naturally, remembering that any reaction was correct. See Fig. 2 for a representation of the physical tasks.

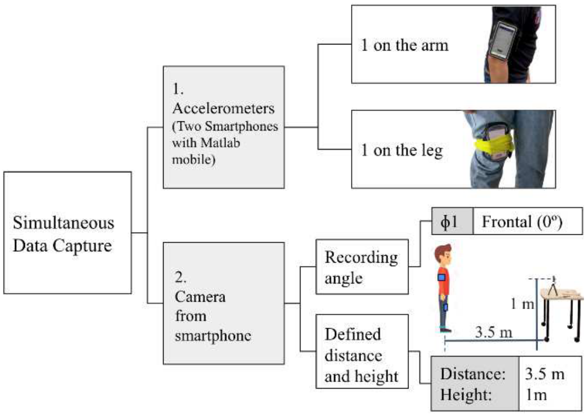

Fig. 1. Experimental design for data capture.

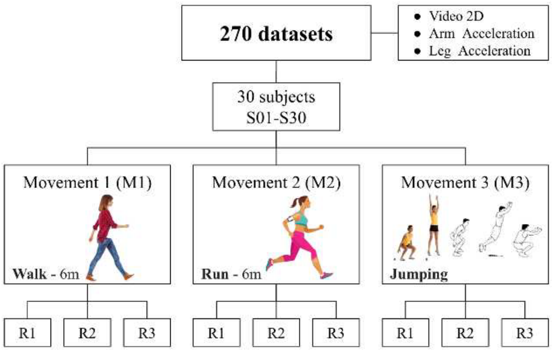

Fig. 2. Experimental database construction. Each subject performs 3 movements with 3 repetitions of each movement.

2. Motion digitalization

The API Skeletal Tracking SDK developed by Cubemos [15] was used to obtain key points of the human body (see Table I) from the recorded videos (e.g., right knee, left knee, etc.). Two-dimensional (XY) positions of the skeletal key points were extracted from the video recordings. A pipeline was designed and developed in Python to perform this step to save the key points of all the frames in each participant's video in a csv file.

The video file is read, and its features are extracted using OpenCV library functions. Frame by frame, the skeleton key points are extracted using the function "track_skeletons" included in the Skeletal Tracking software (see Fig. 3). This function provides the 2D pixel location (X and Y coordinate) of 18 key points using CV and Convolutional Neural Networks (CNN) algorithms.

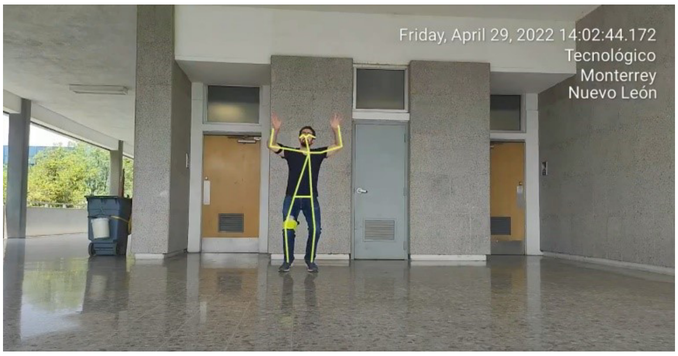

Fig. 3. Skeletal Tracking SDK predicting a human's key points in one frame from the video (ID S11M3R2, subject 11, movement 3, repetition 2)

3. *Data Pre-processing*

Acceleration data and key points were pre-processed to create a final data frame with all the data merged and synchronized using timestamps. A Python code was developed to perform pre-processing tasks, receiving three .csv files as inputs containing acceleration data from the arm, leg, and key points. The output was another .csv file per recording, synchronizing the XY values of 18 key points and XYZ acceleration from the arm and leg, with a frequency of 100 Hz. The Python libraries used in this step were Pandas, NumPy, Regex, Matplotlib, OpenCV, and Spicy.

3.1 *Pre-processing of Acceleration Data*

The acceleration data from the arm and leg were imported to a Pandas' data frame and up-sampled to 100 Hz using the Spline method for interpolating data. A low-pass, 8th-order Butterworth filter with a 15 Hz cutoff frequency was designed and applied. Figs. 4 and 5 show an example of pre-processed data from two acceleration sensors of one subject. The filter was useful to avoid noisy signals being used to train the predictive model.

3.2 *XY position data pre-processing*

Key points data pre-processing was performed as follows: Timestamps were assigned based on the start time of the recording and the frames per second of the original video (see Figs. 4-5).

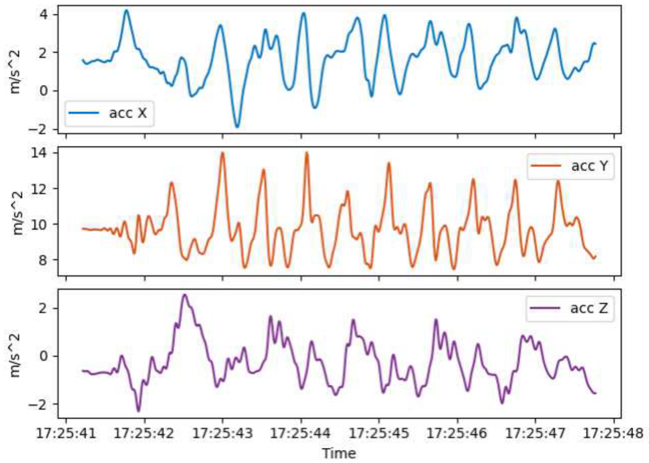

Fig. 4. Pre-Processed Arm Acceleration data corresponding to S14 M1 R1.

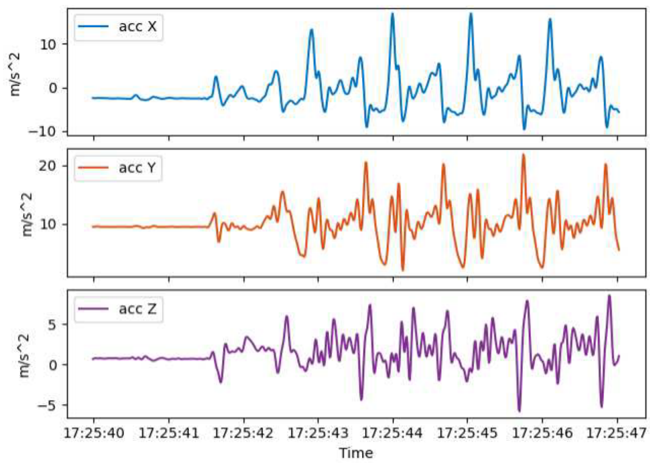

Fig. 5. Pre-Processed Leg Acceleration data corresponding to S14 M1 R1.

After that, a column for each important key point and each dimension (X and Y) was created, obtaining a total of 28 columns (see Table I).

TABLE I. IMPORTANT KEYPOINTS

No.	***Keypoint***	***No.***	***Keypoint***
1	Nose_X	15	Left_Wrist_X
2	Nose_Y	16	Left_Wrist_Y
3	Neck_X	17	Right_Hip_X
4	Neck_Y	18	Right_Hip_Y
5	Right_Shoulder_X	19	Right_Knee_X
6	Right_Shoulder_Y	20	Right_Knee_Y
7	Right_Elbow_X	21	Right_Ankle_X
8	Right_Elbow_Y	22	Right_Ankle_Y
9	Right_Wrist_X	23	Left_Hip_X
10	Right_Wrist_Y	24	Left_Hip_Y
11	Left_Shoulder_X	25	Left_Knee_X
12	Left_Shoulder_Y	26	Left_Knee_Y
13	Left_Elbow_X	27	Left_Ankle_X
14	Left_Elbow_Y	28	Left_Ankle_Y

Once the data was correctly stored in a data frame, the invalid key points (signalized with a value of –1) were replaced for a null np.NaN value.

Similar to acceleration data, position data was also up-sampled to 100 Hz using the Spline method for interpolating data. After that, an 8th-order low-pass Butterworth filter with a 5 Hz cutoff frequency was designed and applied (see Fig. 6).

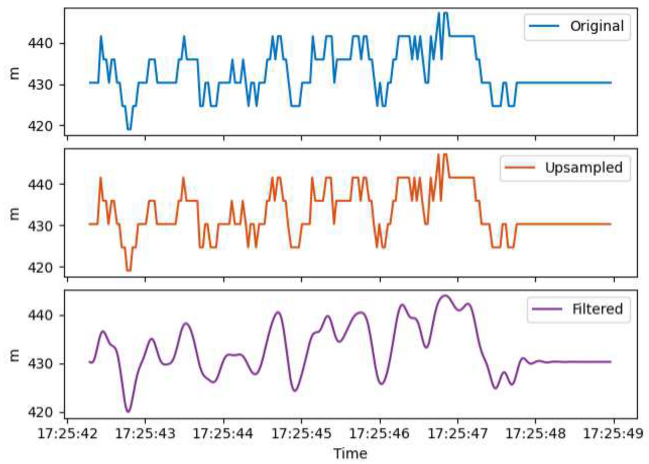

Fig. 6. Example of pre-processed key point signal of Right Knee (Y) corresponding to S14 M1 R1.

Min-max feature scaling was done using MinMaxScaler class from the scikit-learn package in Python to normalize data on each subject and feature. Data normalization helps transform each subject's data into a shared space; moreover, the scaler transforms values into a domain $1 < x < 0$ (low values), which works best when using a Deep Learning (DL) approach.

4. Deep Learning Model

For the DL model creation, the positions of the key points in XY were used as features or source variables; the acceleration with respect to time in XYZ axes were the target features (those to predict), and all variables were continuous numerical data.

For this case, it was decided to use a DL method, the Recurrent Neural Network (RNN) approach. This model is based on an ANN, but it is used for data sequences that are dynamic in time, considering the previous output as a base, together with the current source variables, to predict the output of a current time. This allows us to make predictions taking into account a certain amount of previous data to analyze the historical change of the variables. Such an approach is necessary when predicting acceleration because it is a variable that depends on past changes across time (e.g., speed and position). For this model, data from 4 subjects were used: 50% for training, 25% for testing, and 25% for validation.

The design and training of said model were carried out in Python through Tensorflow and Keras libraries. First, a search for ideal hyper-parameters was made to create a valid model, then a series of parameters were fine-tuned to obtain the DL model with the best performance for the modified parameters. For this case, parameters were changed to compare the performance of the resulting model based on performance metrics (Pearson's correlation coefficient and coefficient of determination).

Additionally, a series of hyper-parameters were adjusted so the model could be considered valid, such that the loss function was reduced in the same way in both the training and validation datasets. This can be achieved by plotting the loss function across epochs in both training and testing datasets and observing a reduction in both.

Based on the analysis of the previously proposed graph, it can be observed when the model tends to overfit (it learns too much from the training dataset and cannot generalize in the validation dataset) or underfit (it is unable to understand the training dataset). This way, these hyper-parameters can be adjusted to generalize or specialize and find the optimal parameters for the model. The modified hyper-parameters are:

1) Learning rate
2) Number of layers and neurons
3) Number of epochs
4) Batch size
5) Sequence size

Once the ideal hyper-parameters were empirically obtained, they were taken as constants. The previously presented parameters were changed as a loss function with layer type to find a combination with the best performance against performance metrics for continuous numerical data. A test was made using obtained results from a theoretical dataset [17] (of physical markers and forces with a treadmill), where this methodology was implemented. Successful performance was obtained with a one-layer RNN model, where the cell Long Short-Term Memory (LSTM) performed better than Simple and Gated Recurrent Unit (GRU); this, in conjunction with the mean squared error (MSE) loss function, as it was also the best combination obtained from the compared methods [18].

B. Infrastructure and technological tools used

Currently, there are a set of five modules working separately to create the infrastructure; these are:

1) Data Collection: Using wearables to determine acceleration (smartphones with integrated accelerometer and Matlab mobile) and video cameras (integrated into Smartphones) to record the subject's movement.

2) Motion digitalization: Using the API Skeletal Tracking SDK from Cubemos to record XY positions.

3) Data processing: The Python platform is used where all the data is collected, and then the determined pre-processing is carried out using the methods specified in the Methodology Section.

4) Prediction: of acceleration using the trained RNN model and normalized data.

5) Visualization: In real-time, the RNN's results are combined with raw video of a person making a move.

The integration of the five modules would then develop the infrastructure needed to solve the biomechanics problem in real-time, although this requires edge computing and interoperability between multiple Internet-of-Things (IoT) devices. All the modules have been tested currently in an offline manner.

C. Resources used

1) Skeletal Tracking SDK: a motion capture software created by the company Cubemos. It includes 2D and 3D skeletal tracking functionality [19].

2) Smartphones: Mobile devices with an integrated video camera that allow capturing movement videos, as well as accelerometers that obtain acceleration data in 3 coordinates.

3) Matlab Mobile: The mobile application developed by MathWorks Inc. allows users to acquire, record, and save data from different sensors integrated into a smartphone. In the current project, it was used to recover XYZ acceleration data by accessing the accelerometer integrated into the smartphone.

III. Results

As part of the model training, three metrics were plotted: MAE (Mean Absolute Error), MSE, and Mean Squared Logarithmic Error (MSLE). Figures 7-9 show the training and validation graphs of the loss function for each metric.

An ideal fit can be observed: the training and validation curves closely match each other; neither overfits nor underfits the data.

Table II shows the coefficient of determination of the three datasets: Training, validation, and testing.

TABLE II. Coefficient of Determination (R^2) on Predictions

Dataset	Leg			Arm		
	acc X	*acc Y*	*acc Z*	*acc X*	*acc Y*	*acc Z*
Training	0.055	0.603	0.681	0.168	0.573	0.553
Validation	-0.095	0.428	0.574	-0.046	0.447	0.399
Testing	-0.348	0.527	0.533	-0.248	0.521	0.369

Figures 10-11 show comparisons between true and predicted three-dimensional acceleration values during a representative trial.

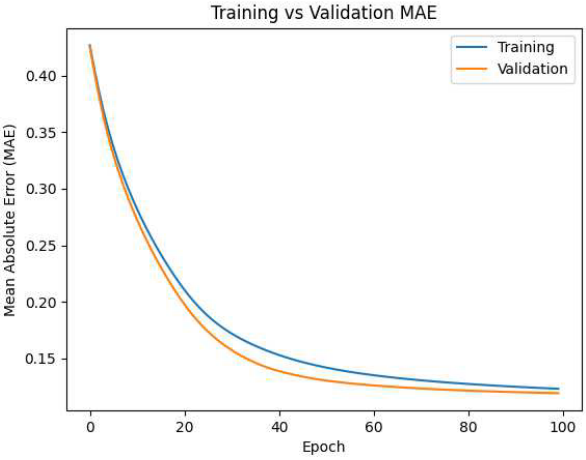

Fig. 7. Training and validation graphs of the loss function (MAE).

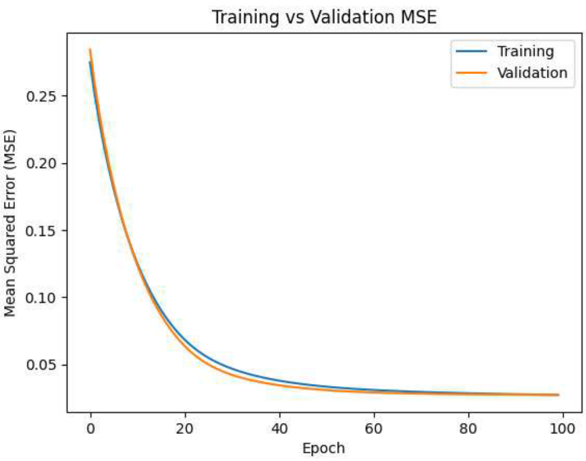

Fig. 8. Training and validation graphs of the loss function (MSE).

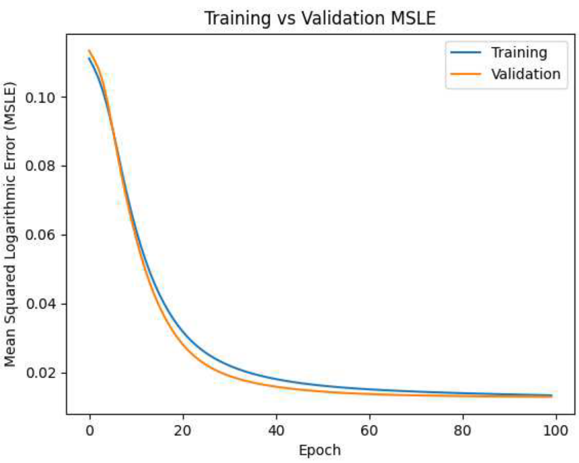

Fig. 9. Training and validation graphs of the loss function (MSLE).

IV. Discussion

The coefficient of determination (R2) was used as a performance metric to evaluate the model's performance based on the trained RNNs. For a direct interpretation, it is safe to assume that a higher R2 value indicates a better prediction. This metric was calculated over ten times, given a process of 10-fold cross-validation, to obtain an average representation of the model's performance. The process consisted of randomly selecting different training, validation, and testing sets for every iteration, then averaging all iterations' results, thus generalizing better for the given dataset. Table III shows the Ideal Hyperparameters for the best performance of the model, which consists of 2 layers: an LSTM with 16 neurons and a Dense layer with 6 neurons.

TABLE III. RNN Model Hyper-parameters

Number of Epochs	350
Batch size	128
Sequence size	10
Layers	LSTM (16), Dense (6)
Loss function	Mean Absolute Error (MAE)

In Machine Learning, testing the dataset's performance is usually considered the most valid result because it simulates a real-world scenario where the model could not access the new data. Taking this into account, Table II shows the coefficient of determination of the three datasets: Training, validation, and testing. In the testing dataset, the highest coefficient of determination corresponds to predicting leg acceleration in the X-axis with a value of 0.533. The next-best predictions, with the maximum R2, were obtained for the Y-axis in both the leg and arm.

In the coefficients of determination, the acceleration in z had a lower value because the recorded video was filmed in 2D, where the z values can't be represented. Nevertheless, the coefficients of determination obtained for the X and Y axes were closer to 0.5, which means the predictions are presented with higher accuracy than in the Z-axis. This can be observed in Figs. 10-11, which compare true and predicted three-dimensional acceleration values during a representative trial, and in both cases, the XY estimation resembles most the true signal rather than the Z axes.

It is recommended that future applications perform trials with more subjects to obtain a bigger dataset, always recording with the same device and using adequate material for the data acquisition to acquire better results. In addition, it is essential to state that the program used to obtain the key points could be more accurate, and the coefficient in z is lower compared to x and y since the recorded video is two-dimensional, while the prediction is three-dimensional.

V. Conclusions

Developing low-cost, easy-to-implement technologies for biomechanical assessment of movement is highly relevant in healthcare, sports, and rehabilitation. Combining AI and CV into a markerless, video-based biomechanical tool for joint acceleration prediction can help improve athletes' performance, estimate the risk of fractures, and prevent injuries in general.

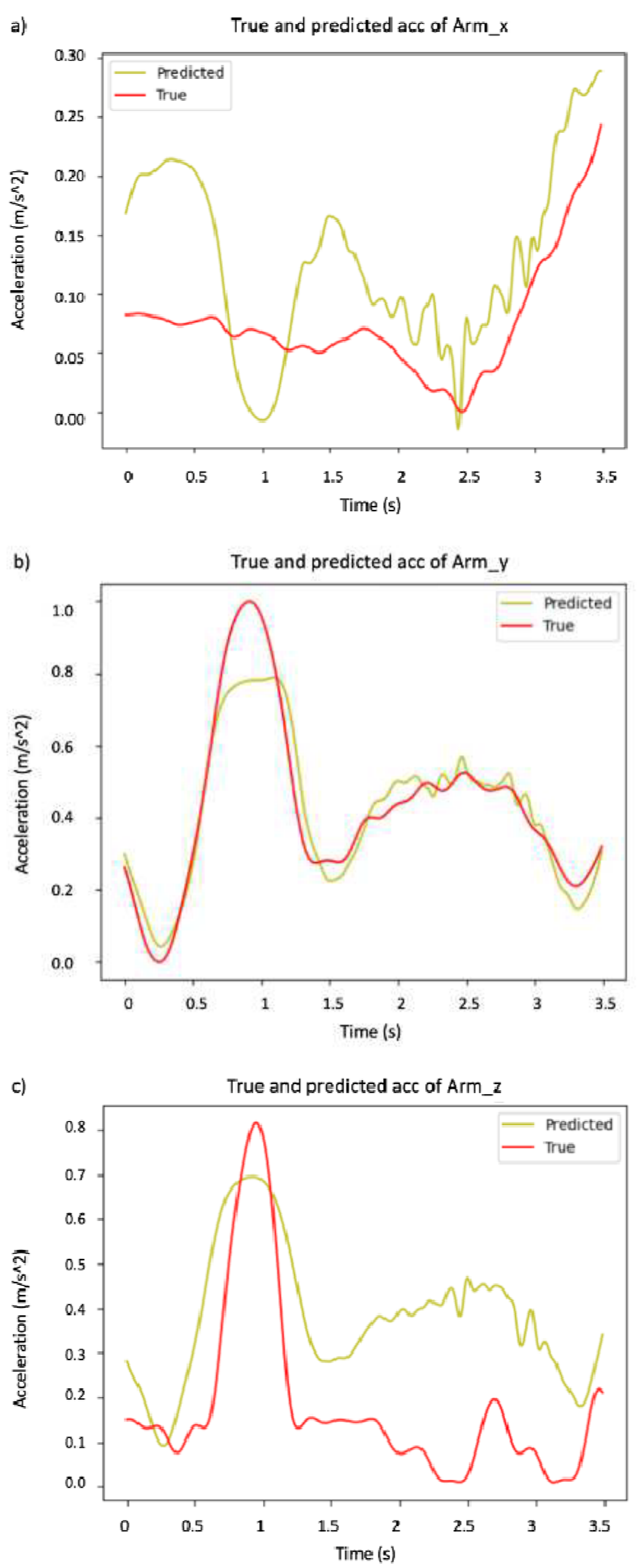

Fig. 10. True and predicted values. a) Acceleration of the arm in the X-axis. b) Acceleration of the arm in the Y-axis. c) Acceleration of the arm in the Z-axis.

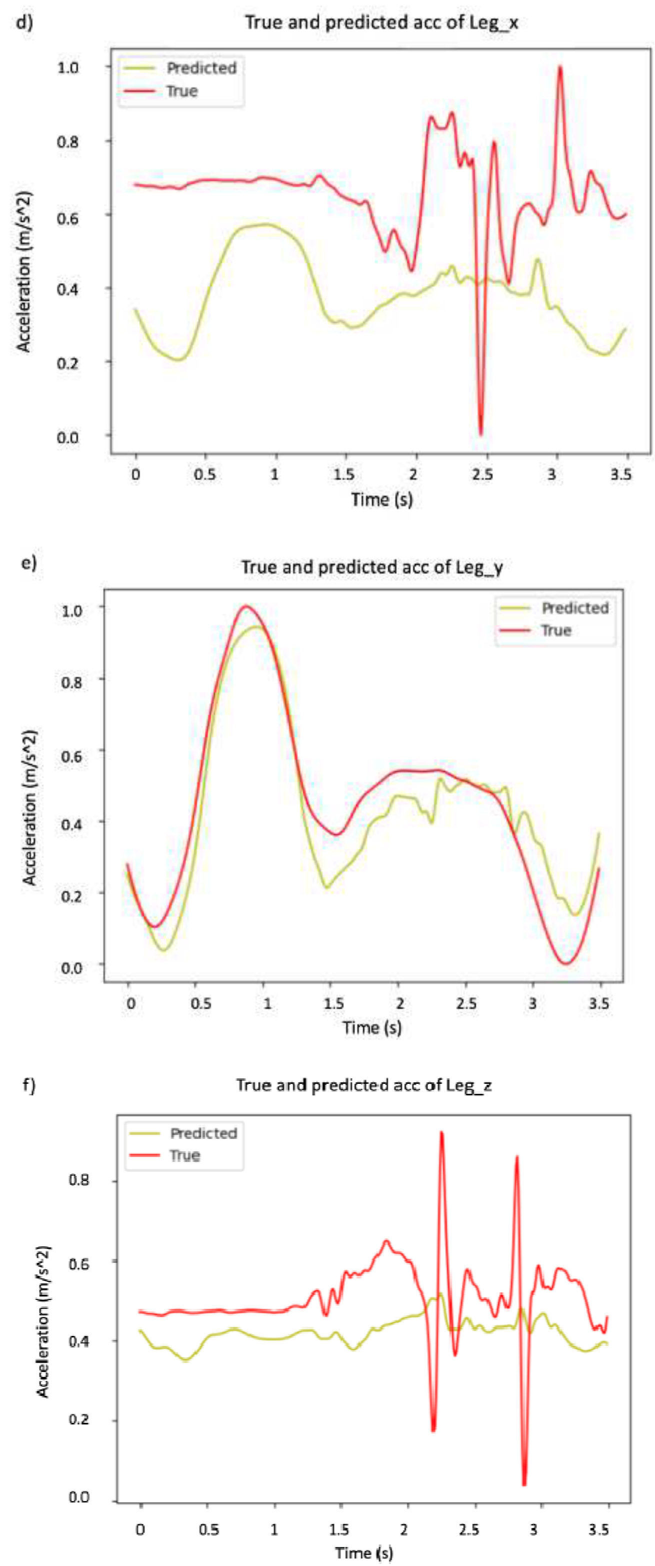

Fig. 11. True and predicted values. d) Acceleration of the leg in the X-axis. e) Acceleration of the leg in the Y-axis. f) Acceleration of the leg in the Z-axis.

Although the current work presents an initial assessment of this approach, and the model needs further improvements in its accuracy, it has the potential to develop more accessible technologies in the field of markerless biomechanical assessment.

The results showed that a relationship between key-points positions and real acceleration could be established, and thus DL approaches such as RNNs can relate these parameters with high precision. The model presented optimum prediction precision, showing a resembling behavior between the real and the predicted data, obtaining a maximum R2 value of 0.533. This indicates that, up to an extent, the hypothesis of our work was correct, as joint acceleration was predicted satisfactorily through video (leg, X-axis), although there is still room for improvement. To improve the proposed model, future work should include more sensors in the setup to increase the degrees of freedom from which the model can learn and increase the chance of improving the model's accuracy.

A perspective of this work, which is related to the project's long-term goal, is the development of an interconnected portable version of the proposed system, such that the joint acceleration estimations are performed in real-time. Also, by obtaining the estimated accelerations and converting them into estimated joint forces, the system could provide more insightful recommendations to athletes and other users about the quality of their movements and the risk of injuries.

REFERENCES

[1] G. N. Guten, Running injuries. Saunders, 1997.

[2] P. Melissa A. Furlong, PhDa, Dana Boyd Barr, PhDb, Mary S. Wolff, PhDc, and Stephanie M. Engel, "Preventive Biomechanics: A Paradigm Shift With a Translational Approach to Injury Prevention," Physiology & behavior, vol. 176, no. 1, pp. 100–106, 2016.

[3] C. J. Parris, "The Future for Industrial Services - The Digital Twin," Infosys Insights, pp. 42–49, 2016.

[4] F. Geissler, R. Heiβ, M. Kopp, M. Wiesmu ̈ller, M. Saake, W. Wuest, A. Wimmer, V. Prell, M. Uder, and M. S. May, "Personalized computed tomography - Automated estimation of height and weight of a simulated digital twin using a 3D camera and artificial intelligence," RoFo Fortschritte auf dem Gebiet der Rontgenstrahlen und der Bildgebenden Verfahren, vol. 193, no. 4, pp. 437–445, 2021.

[5] R. Gámez Díaz, Q. Yu, Y. Ding, F. Laamarti, and A. El Saddik, "Digital twin coaching for physical activities: A survey," Sensors (Switzerland), vol. 20, no. 20, pp. 1–21, 2020.

[6] I. Voigt, H. Inojosa, A. Dillenseger, R. Haase, K. Akgu ̈n, and T. Ziemssen, "Digital Twins for Multiple Sclerosis," Frontiers in Immunology, vol. 12, no. May, pp. 1–17, 2021.

[7] Chan, C.K., et al. "Human Motion Classification Using 2D Stick-Model Matching Regression Coefficients." Applied Mathematics and Computation, vol. 283, June 2016, pp. 70–89. Science Direct, 10.1016/j.amc.2016.02.032. Accessed 29 Dec. 2022.

[8] Liu, Yu, et al. "Lower Extremity Joint Torque Predicted by Using Artificial Neural Network during Vertical Jump." Journal of Biomechanics, vol. 42, no. 7, May 2009, pp. 906–911. PubMed, 10.1016/j.jbiomech.2009.01.033. Accessed 24 Aug. 2021.

[9] Gholipour, A., and N. Arjmand. "Artificial Neural Networks to Predict 3D Spinal Posture in Reaching and Lifting Activities; Applications in Biomechanical Models." Journal of Biomechanics, vol. 49, no. 13, 6 Sept. 2016, pp. 2946–2952. PubMed, pubmed.ncbi.nlm.nih.gov/27452877/, 10.1016/j.jbiomech.2016.07.008. Accessed 29 Dec. 2022.

[10] R. Robb, Three-Dimensional Biomedical Imaging. Three-dimensional Biomedical Imaging, CRC-Press, 1985.

[11] F. Adso, S. Antonio, & L. Xavier. (2012, September). Biomechanical Validation of Upper-Body and Lower-Body Joint Movements of Kinect Motion Capture Data for Rehabilitation Treatments, 2012 Fourth International Conference on Intelligent Networking and Collaborative Systems, 2012, pp. 656-661, doi: 10.1109/iNCoS.2012.66.

[12] Halilaj, E., Shin, S., Rapp, E., & Xiang, D. (2021). American society of biomechanics early career achievement award 2020: Toward portable and modular biomechanics labs: How video and IMU fusion will change gait analysis. Journal of Biomechanics, 129, 110650.

[13] Ceseracciu, E.; Sawacha, Z.; Fantozzi, S.; Cortesi, M.; Gatta, G.; Corazza, S.; Cobelli, C. Markerless analysis of front crawl swimming. J. Biomech. 2011, 44, 2236–2242.

[14] De Froda, S.F.; Thigpen, C.A.; Kriz, P.K. Two-dimensional video analysis of youth and adolescent pitching biomechanics: A tool for the common athlete. Curr. Sports Med. Rep. 2016, 15, 350–358.

[15] Lloyd, D. The future of in-field sports biomechanics: Wearables plus modeling compute real-time in vivo tissue loading to prevent and repair musculoskeletal injuries. Sports Biomech. 2021, 1–29

[16] Ortiz-Padilla, V. E., Ramírez-Moreno, M. A., Presbítero-Espinosa, G., Ramírez-Mendoza, R. A., & Lozoya-Santos, J. D. J. (2022). Survey on Video-Based Biomechanics and Biometry Tools for Fracture and Injury Assessment in Sports. Applied Sciences, 12(8), 3981.

[17] R. Fukuchi, C. Fukuchi, and M. Duarte, "A public data set of running biomechanics and the effects of running speed on lower extremity kinematics and kinetics," PeerJ, vol. 5, p. e3298, 3 2017.

[18] M. O. Candela-Leal et al., "Multi-Output Sequential Deep Learning Model for Athlete Force Prediction on a Treadmill Using 3D Markers," Applied Sciences, vol. 12, no. 11, p. 5424, May 2022, doi: 10.3390/app12115424.

[19] Cubemos SDK. Skeleton Tracking. https://www.cubemos.com/skeleton-tracking-sdk.