

Biomechanics for the Digital Twin of Performance

M.O. Candela-Leal
Tecnológico de Monterrey
Monterrey, México
milton.candela@tec.mx

V.E. Ortiz-Padilla
Tecnológico de Monterrey
Monterrey, México
A00822903@itesm.mx

K.L. Rodríguez-Hernández
Tecnológico de Monterrey
Monterrey, México
karen.rdz@tec.mx

D. Martínez-Díaz
Tecnológico de Monterrey
Monterrey, México
A01733799@itesm.mx

A.J. Aguilar-Herrera
Tecnológico de Monterrey
Monterrey, México
A00828691@itesm.mx

E.A. Gutiérrez-Flores
Tecnológico de Monterrey
Monterrey, México
A00828091@itesm.mx

C.F. Cruz-Gómez
Tecnológico de Monterrey
Monterrey, México
A00827233@itesm.mx

J.E. Martínez-Herrera
Tecnológico de Monterrey
Monterrey, México
A01283785@itesm.mx

E.A. Delgado-Jiménez
Tecnológico de Monterrey
Monterrey, México
A00827948@itesm.mx

G. Presbítero-Espinosa
Tecnológico de Monterrey
Monterrey, México
presbitg@tec.mx

M.A. Ramirez-Moreno
Tecnológico de Monterrey
Monterrey, México
mauricio.ramirez@tec.mx

J.J. Lozoya-Santos
Tecnológico de Monterrey
Monterrey, México
jorge.lozoya@tec.mx

Abstract—The current work consists in creating an easy-to-use mobile application that records a person’s movement and make predictions about the risk of fracture on their joints, by creating a biomechanical Digital Twin (DT) and applying AI-based algorithms. The methodology consists on: 3D recording of an athlete doing exercises using Intel®RealSense™camera, a Computer Vision (CV) algorithm would be applied to get the subject’s joint’s keypoints, afterwards, a Machine Learning algorithm based on deep learning using the modern Recurrent Neural Network (RNN) architecture is used to predict the force exerted by each limb, and so a risk of fracture coefficient would be calculated to determine the risk of fracture in real time. Current results are promising: a CV algorithm successfully identifies XYZ coordinates of key joints in the human body, in addition to the development of a RNN based on a dataset that relates XYZ positions to XYZ forces when using a treadmill. The results are in process of being consolidated with the objective of being portable while still providing great predictions on joints’ forces. The necessary and currently available resources are mainly devices to measure a person’s biometric data (such as Hexiwear and Empatica wristbands), a 3D and 2D camera and a research team. The system in question has applications in geriatric healthcare by including meta-features such as age, gender, osteoporosis, as well as to measure athletes’ performance in real-time, which would then provide feedback to users on how to correctly perform exercises or movement.

Index Terms—Digital Twin, Biomechanics, IoT, Real-time Simulation, Sports

I. INTRODUCTION

A. General context

Throughout the years, the study of body movement has helped us analyze various fields of study, among which physical activity and exercise stand out, a field where there’s a high risk of injuries in high-performance activities, specifically a high incidence in the formation of stress fractures has been identified. A stress fracture can be defined as “partial or complete fracture of bone due to its inability to withstand non-visible stress applied in a rhythmic, repeated, sub-threshold

manner” [1], and it is known that when muscles around the area become fatigued, they concentrate excessive forces over the zones surrounding the bone, generating microcracks.

Microcracks accumulate, grow, and can cause macro fractures, in high-intensity sports, fatigue injuries compose 20% of the total injuries [2], while in the military, they reach a percentage of 31% [3]. In the healthcare area regarding elderly patients, the risk of fracture can increase when a person gets older [4], but it is significantly reduced for patients with osteoporosis through appropriate treatment in the early stages of risk [5].

Prevention of injury or fracture in athletes has been obtained through biomechanical analysis using motion capture systems [6], and could potentially be used in healthcare or older adults. The focus of the study is the use of technology to develop tools that help people identify risk factors in their body movements that may compromise their overall health and wellbeing.

Digital Twin technology started trending as a manufacturing asset [7], and continued to expand to other areas, rapidly growing in healthcare and wellbeing, with applications such as a personalized computed tomography (CT) for estimation of height and weight [8], a smart coaching system in sports [9], and in general a more complete assessment of a patient’s characteristics and medical history for a more effective care or a simulation of treatment [10].

Commonly, motion capture systems have been used in medical science to analyze and evaluate the human body movement and function [11], and with them efforts to provide more adaptable and friendly motion capture systems by making them more affordable and portable [12]. One of the methods that have been on the rise for this same purpose has been the use of AI to assist markerless motion capture softwares such as Open Pose, AlphaPose, among others [13].

B. Delimitation of the object of study

The main applications for this study have been defined as healthcare for elderly patients, as well as technique improvement and injury prevention for athletes, with a specific focus using metadata of the person, mainly variables to consider in each individual that could affect the development of injuries and fractures, such as age, gender of the person, and the presence of previous medical history that influence the performance of such movements.

C. Problem Statement

Several studies have used biomechanical analysis to study movement, where a commonly used method is mainly through a motion capture system in a controlled environment and multiple markers placed on the subject's body, using 2D or 3D camera systems [14], also delimited with sensor systems [15] to track people's movement and use the data to the desired application of the biomechanical analysis.

For the public, 3D camera systems and laboratory setup can represent a limitation in aspects such as the setup time to accommodate the space used, long trials, and the cost it might take to acquire the materials to perform, capture and analyze these movements. Also, the general analysis represents its own complications for coaches in sports, as technical experts and the consensus formed by them to interpret the results is needed.

D. Justification

The proposed system is based on a mobile app that, through the application of Artificial Intelligence (AI), Computer Vision (CV) and wearables for biometry acquisition, can create a Digital Twin -a virtual representation and a dynamic status reflection of a physical entity- so that it tracks the person's movement without markers on the body, which subsequently gives real-time feedback to the subject with the collected data, with the main purpose of predicting risk of fracture on the joints of the person for the prevention of injury and aggravation of pre-existing medical conditions.

The app combined with the absence of markers on the body makes the tracking and creation of a Digital Twin easier and convenient, as it is more manageable and gives feedback instantly as the person is doing the activity or movement. It also lowers the costs of the study as special equipment or markers aren't needed, time used for trials is shortened [16] and it becomes more accessible for a greater portion of the desired public, as we established as athletes and older people.

E. Theoretical framework

1) *Digital Twin*: Digital Twin is defined as “an organic whole of physical asset (or physical entity) as well as its digitized representation, which mutually communicate, promote, and co-evolve with each other through bidirectional interactions”. It's a technology that has gained attention from the academia and industry, because it can represent a physical entity in the digital space in a unique way, respecting its shape, position, gestures, among other aspects, and along with

AI, machine learning and data analytics, it can be used for diagnostics, monitoring or optimization [17].

2) *Biomechanics*: Biomechanics can be defined as the application and study of mechanics to understand the motion of living beings and their response to applied loads. This field has a broad domain, as its use ranges from the flight of birds or the effect of gravity on the growth of plants, to addressing issues of health, such as diseases and injuries, as well as their treatment in humans and animals, which is why its use on biomedical technologies is rapidly growing [18].

3) *Internet of Things*: A general definition of Internet of Things (IoT) can be an open network of intelligent objects that can auto-organize, share information, data, and resources between them, constantly reacting and acting to changes in their environment. Over time, and specially during the last decade, IoT has attracted attention thanks to the prospective of a constantly communicated and connected world, and acting depending on the conditions and status of these objects in the real world, which then can facilitate simulations on the digital space of real-time events [19].

F. Objectives

The main objectives were defined as the prevention of injuries and fractures in athletes and elderly people through an easy-to-use low-cost system for movement tracking, for broader usage and accessibility. Also, among these features, it is intended that the analysis could be done without the need for markers on the subject's body, contributing to the purpose of an option that is easier to use in the general public and coaches of athletes.

G. Hypothesis

The hypothesis is that the created system would be able to predict important biomechanical features, such as force for each joint, via detected key points. Based on low-cost edge devices to foster quick biomechanical analysis in athletes and to benefit prevention of injuries in elderly people, which would be beneficial while maintaining accessibility.

II. PROPOSAL

A. Methodology

1) *Loss Functions*: In order to measure performance on continuous variable predictions, a set of evaluation metrics are used as loss functions so that the predictive, deep learning based model, could evaluate its performance and so change its parameters. The objective of these evaluation metrics is to reduce its value to a minimum, because they are measuring the difference between the reference values and their predictions, and thus the objective is to have predicted values that are very alike to reference values.

The first evaluation metric that is being used is the Mean Absolute Error (MAE), which represents the average of absolute of the differences between predicted values and observed values. MAE was calculated as shown in Eq. 1.

$$MAE = \frac{1}{n} \sum_{i=1}^n |y_i - \hat{y}_i| \quad (1)$$

Additionally, the Mean Squared Error (MSE) was also used, which represents the average of squares of the differences between predicted values and observed values. MSE was calculated as shown in Eq. 2.

$$MSE = \frac{1}{n} \sum_{i=1}^n (y_i - \hat{y}_i)^2 \quad (2)$$

Finally, Mean Squared Logarithmic Error (MSLE) measures the ratio between predicted and observed values, it is quite different as the other metrics, because it measures the percentual difference and thus is better for extreme values. MSLE was calculated as shown in Eq. 3.

$$MSLE = \frac{1}{n} \sum_{i=1}^n (\log(y_i + 1) - \log(\hat{y}_i + 1))^2 \quad (3)$$

2) *Performance Metrics*: Coefficient of determination (R^2) is a widely used regression performance metric to measure how well the predictions of a model fit into the actual data. The metric's domain is $0 \leq R^2 \leq 1$, where $R^2 = 1$ means perfect predictions, it was calculated as shown in Eq. 4.

$$R^2 = \frac{SSR}{SST} \quad (4)$$

Where SSR refers to the Sum of Squared Regression (SSR), which is also known as variation explained by the model. SSR was calculated as shown in Eq. 5.

$$SSR = \sum_{i=1}^n (\hat{y}_i - \bar{y})^2 \quad (5)$$

On the other hand, SST refers to the Sum of Squared Total (SST), which is also known as the total variation in the data. SST was calculated as shown in Eq. 6.

$$SST = \sum_{i=1}^n (y_i - \bar{y})^2 \quad (6)$$

3) *Data Pre-Processing*: Min-max feature scaling was done using *MinMaxScaler* class from *scikit-learn* package in Python, in order to normalize data depending on both subject and feature, data normalization is useful to transform each subject's data into a common space, moreover, the scaler transforms values into a domain $1 < x < 0$ (low values), which works best when using a DL approach. In order to perform the feature scaling, two equations would be needed: Firstly, a X_{std} value was calculated as shown in Eq. 7.

$$X_{std} = \frac{X - X_{min}}{X_{max} - X_{min}} \quad (7)$$

Secondly, max and min for each subject and feature was also calculated, to thus, for each X_{std} value transform it into an X_{scaled} value, which was calculated as shown in Eq. 8.

$$X_{scaled} = X_{std} \cdot (max - min) + min \quad (8)$$

Finite Impulse Response (FIR) is a digital filter was applied using *lfilter* function from *scipy* package in Python, with parameters $n = 10$, $a = 1$, an additional parameter b was

calculated as shown in Eq. 9. It is worth also noting that the parameter b must be an array of length n , and so should be repeated n times, while a remains as a constant.

$$b = [(\frac{1}{n})_1 \quad (\frac{1}{n})_2 \quad \cdots \quad (\frac{1}{n})_n] \quad (9)$$

The filter was useful to avoid noisy signals being used to train the predictive model, and so it was applied before normalization to have cleaner data on F_x and F_z .

Video could be recorded with multiple devices, which means that video dimensions and thus image dimensions could vary depending on the device that is being used. In order to normalize this parameter and reduce dimensional to reduce computational cost with the DL method that would be used, any image would be re-scale it into a normalized size of 299 by 299 pixels.

Target Resolution TR for both dimensions d would then be 299, moreover, the device would have a Current Resolution CR depending on dimensions d width and height, and so the Processed Pixel PP was calculated using the Raw Pixel RP as shown in Eq. 10, the equation would be followed for each pixel to process it, hence the i sub-index.

$$PP_{i,d} = RP_{i,d} \cdot \frac{TR_d}{CR_d} \quad (10)$$

B. Methodological proposal to use

The methodological proposal is represented as a flow diagram in Fig. 1. On the left-hand side are the real-time operations and data inflow, while the right-hand side are the steps needed to train a RNN, including data division and hyper-parameter tuning.

Describing the left-hand side: A video using Wearables is taken, and so it is processed via OpenPose or Intel®RealSense™, and thus export body keypoints to JavaScript Object Notation (JSON) files for each frame, then JSON files are read using Python, further employing *Matplotlib* to plot the results. At the same time, XYZ positions and acceleration is being extracted from the subject using *Empatica E4* and *Hexiwear*,

Conversely, a RNN model was trained using *Tensorflow*, a deep learning Python library that includes *Keras* infrastructure to create predictive models. The predictive model was trained using a public dataset of running biomechanics [20], which related XYZ extremity positions using a three-dimensional (3D) motion-capture system, and XYZ forces using an instrumented treadmill. The database consisted in 28 regular runners ($F = 1$, $M = 27$) with age ($\mu = 34.75$, $\sigma = 6.566283$), which ran at velocities: ($2.5ms^{-1}$, $3.5ms^{-1}$, $4.5ms^{-1}$) wearing standard neutral shoes.

C. Techniques and technological tools used

1) *OpenPose*: The first real-time multi-person system to jointly detect human body, hand, facial, and foot keypoints (in total 135 keypoints) on single images [21], although it can also be used in real time or to process a full video. It is based on a Convolutional Neural Network (CNN) algorithm which does not require markers to track the movement of the human body.

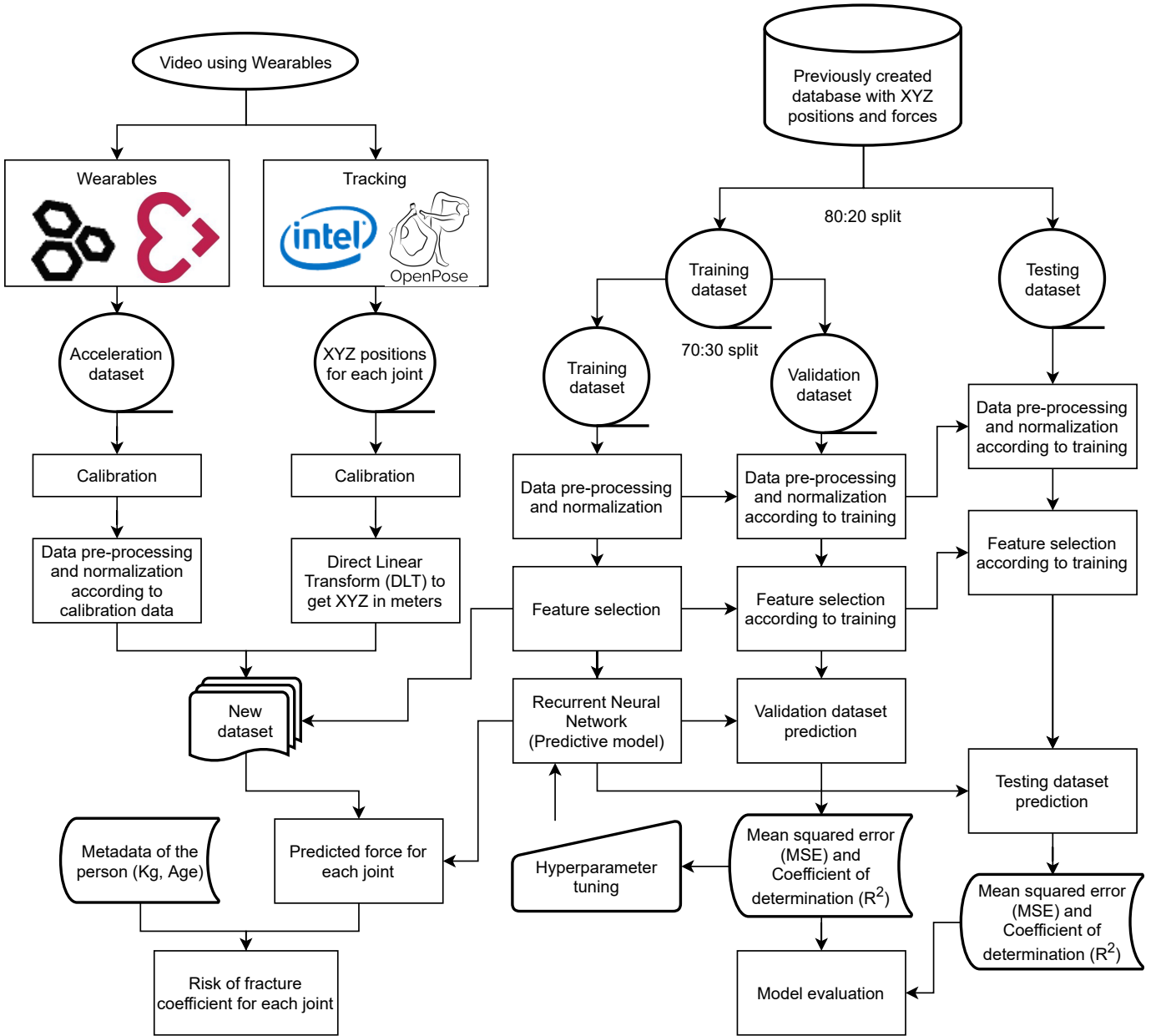


Fig. 1: Methodological proposal represented in a flow diagram that includes training on the RNN predictive model (right-hand side), and its usage on real-world scenarios with new data coming from biometric devices and cameras (left-hand side).

An example of the software is shown in Fig. 2, a video was processed using our designed pipeline that included OpenPose, although, only frame 650 would be used. The process starts with the raw video in 2a, then OpenPose detects its keypoints 2b, finally, those keypoints are read in Python and re-scaled into a (299, 299) resolution 2c.

The Computer Vision (CV) algorithm has a variety of outputs, including JSON files for each frame, an example of the pixels's values for each keypoint is shown in Table I. It is worth noting that the base body model includes 25 keypoints with coordinates in pixels, the output is then shown in the "Real" column for XY coordinates. Afterwards, the image

coordinates were normalized into a smaller dimension (299, 299) using Eq. 10, moreover, the software also outputs its level of confidence for each keypoint, which is 0 when it reaches a certain, previously defined, threshold, which also computes 0 on their XY predictions when it is not confident enough.

2) *Recurrent Neural Network*: A type of Artificial Neural Network (ANN) which uses sequential data. Unlike others ANNs, these have memory and so past inputs influence new inputs and outputs, their applications include Natural Language Processing (NLP) and time-series forecasting.

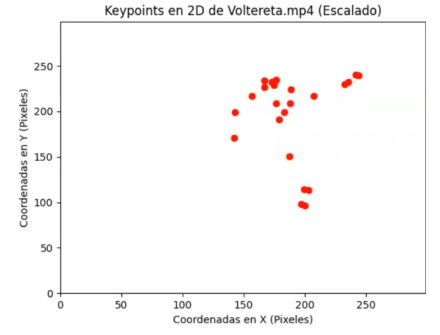
Tensorflow is a library in Python that would be used to create the RNN, using *Keras* to design the model's architecture,



(a) Raw recorded video.



(b) Pre-processed using OpenPose.



(c) JSON's read and plotted.

Fig. 2: Flow diagram of a video [22] into the multiple processing stages, a frame is displayed in (a). Afterwards, OpenPose was used to process the video, resulting in the detection of human's body keypoints in (b). OpenPose's output are JSONs files for each frame in the video, which was read and processed using *Pandas* and *Matplotlib* in Python (c).

which is mainly composed of one-layer with a *SimpleRNN* activation function that contains 128 neurons. Additionally, the following hyper-parameters can be tuned to affect the model's performance (used values are in parenthesis):

- Batch size: Total number of training examples in a single batch, as the neural network updates its internal weights when it finished with a batch (128).
- Sequence size: Number of samples that would be taking into consideration when predicting the next sequential sample, in time-series forecasting on a dataset in days, would be the number of days (5).
- Epochs: Number of times that the entire training dataset is passed forward and backward through the neural network, thus how many times would be used (25).
- Training-Validation-Testing split: Percentage of data that would be used to train the model, validate its performance during training, and finally test its performance on an unknown set of subjects (%70, %10, %20).

D. Infrastructure

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

- 1) Data collection: Wearables to determine acceleration (Hexiwear & Empatica E4), Intel @RealSense™Depth Camera *D435i* to record XYZ positions.
- 2) Data processing: Using Python to collect data and pre-process it, using methods specified in Sect. II-A3.
- 3) Prediction: Of a force and risk of fracture coefficient using the trained RNN model and normalized data.
- 4) Visualization: On a real-time basis, RNN's results combined with the raw video of a person performing a movement.

The integration of the four modules would then develop the infrastructure needed to solve the biomechanics problem, although this requires edge computing and interoperability between multiple Internet of Things (IoT) devices. The modules

TABLE I: OpenPose's predictions and level of confidence for each body keypoint (25), on frame 650 of Fig. 2b.

Body Part	Real		Re-scaled		Accuracy (%)	Body Part	Real		Re-scaled		Accuracy (%)
	X	Y	X	Y			X	Y	X	Y	
Nose	747	539	174	224	90.13	LKnee	900	539	210	224	62.48
Neck	712	525	166	218	86.17	LAnkle	1035	588	242	244	35.92
RShoulder	667	506	156	210	78.23	REye	743	545	174	226	82.81
RElbow	620	463	145	192	90.58	LEye	747	553	174	230	5.6
RWrist	622	398	145	165	87.99	REar	710	555	166	230	89.26
LShoulder	753	545	176	226	73.6	LEar	0	0	0	0	0.0
LElbow	806	506	188	210	86.95	LBigToe	1062	588	248	244	26.75
LWrist	745	475	174	197	89.43	LSmallToe	1056	592	247	246	30.81
MidHip	778	461	182	191	71.22	LHeel	1049	598	245	248	29.24
RHip	759	435	177	181	70.11	RBigToe	839	210	196	87	78.76
RKnee	784	343	183	143	79.95	RSmallToe	829	212	194	88	71.74
RAnkle	837	253	196	105	82.89	RHeel	851	251	199	104	73.53
LHip	802	480	187	200	62.85						

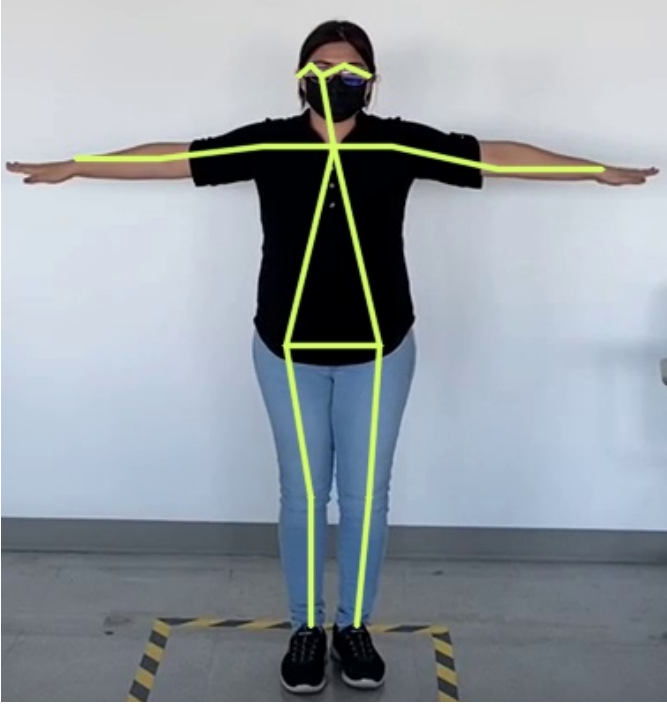


Fig. 3: Intel's SDK making predictions of human's keypoints using a XY video recorded using a cellphone.

that are being worked the most are data collection and local data processing, as the database for the RNN model is just being created.

E. Resources used

1) *Intel®RealSense™Depth Camera D435i*: One of Intel®RealSense™Computer Vision - Depth and Tracking cameras, which has an Inertial Measurement Unit (IMU) to refine its depth awareness when the camera moves. Measuring depth, means that its results would be in three dimensions: X , Y , and Z , which adds complexity to the predictive model, although, this would be useful to calibrate data and make more flexible predictions.

In order to gather data, it is recommended to use Intel RealSense Software Development Kit (SDK) 2.0, which provides time stamped IMU data, and so video could be synchronised with other biometric devices' data, such as the Empatica E4 and Hexiwear. Fortunately, the SDK is compatible with Python, although keys for using the SDK are limited, as it fast and accurate when using it with the respective camera.

Unlike OpenPose, Intel's SDK was tested on real-time video of ours, in Fig. 3 a student performed the initial calibration pose, and the SDK's CV algorithm identified main keypoints on the subject, these keypoints (18) in Table II are less in quantity when compared to OpenPose's keypoints (25) in Table I. Although, predictions are faster and capable on using them in real-time, moreover, a re-scaling method was also applied to transform an image dimension to a certain standardized dimension (299, 299).

TABLE II: Real and re-scaled positions of human's keypoints detected by Intel's SDK in Fig. 3.

Body Part	Real		Re-scaled	
	X	Y	X	Y
Nose	251	374	156	131
Neck	244	407	152	143
Right Shoulder	211	407	131	143
Right Elbow	158	414	98	145
Right Wrist	99	420	61	148
Left Shoulder	284	414	177	145
Left Elbow	344	420	214	148
Left Wrist	403	420	251	148
Right Hip	225	520	140	183
Right Knee	225	606	140	213
Right Ankle	225	679	140	239
Left Hip	271	520	169	183
Left Knee	258	606	160	213
LAnkle	251	679	156	239
Right Eye	244	367	152	129
Left Eye	258	367	160	129
Right Ear	231	374	144	131
Left Ear	271	374	169	131

2) *Empatica E4*: A smartband manufactured by the company Empatica; it provides us with a variety of signals such as an accelerometer, Blood Volume Pulse (BVP), ElectroDermal Activity (EDA) and the skin temperature. Although, in this project, it would only be used to collect subject's acceleration, as other features are more related to a human's physiological state, rather than its movement.

To acquire its data, Python was used to establish connection with the Lab Stream Layer (LSL) module. The LSL in turn, is able to interact with the "E4 streaming server" app available on Windows and MacOS platforms. The E4 streaming server acquires the data from the smartband at a frequency of approximately 1 Hz, further processed using Python.

3) *Hexiwear*: An internet of things smart device designed by the company MikroElektronika to serve as a multipurpose device. In this project, it collects subject's acceleration using its integrated *FXOS8700CQ* combo accelerometer and magnetometer sensor, although the device could also be used as a gyroscope, barometer and ambient light. Data was acquired using a Raspberry Pi model B by the sensor's Universally Unique Identifier (UUID), and further stored using Python.

The reason behind using a Raspberry was the lack of server or intermediate application by this wearable, and so it could not be directly connected to any programming language, therefore a Linux distribution (Raspbian OS) was configured in order to establish a connection with the Bluetooth dedicated socket on-board the Hexiwear device. Having the UUID's from the sensor, data was acquired using the Linux terminal, further using Python to access and store the data in real-time.

III. RESULTS & CONCLUSIONS

A. Results

The public dataset from [20] contains 48 anatomical and reflective markers (mm), resulting in a total of 98 source features sampled at a frequency of 150 Hz , these features would be used to predict the target features, which are forces (N) in the three axis (F_x , F_y , F_z), sampled at a frequency of 300 Hz . As more data would be generated for the target features with respect to source features, granularity must be solved in order to fit a ML model, and thus there are two methods to match both data on a single dataset:

- Down-sampling: The sequential elimination of samples, in this case, since the sampling frequency is doubled, then even observations were eliminated.
- Up-sampling: The sequential creation of synthetic data, via the prediction of missing samples using a linear regression model, for each feature.

Based on the trained RNN's performance, coefficient of determination (R^2) was used as a performance metric to measure a model's performance (higher is better). Results are displayed in Table III, results are divided into training, validation and testing dataset, as well as up-sampling and down-sampling technique applied to the data. On ML, testing dataset's performance is usually considered as the most valid results, because it reassembles a real-world scenario where the model would not have access to the new data.

TABLE III: Coefficient of determination (R^2) on predictions, varying RNN's loss function and sampling technique.

Dataset	Up-Sampling			Down-Sampling		
	Fx	Fy	Fz	Fx	Fy	Fz
Mean Absolute Error (MAE)						
Training	0.68	0.81	0.19	0.87	0.95	0.22
Validation	0.67	0.81	0.01	0.86	0.94	-0.01
Testing	0.71	0.81	0.28	0.88	0.95	0.18
Mean Squared Error (MSE)						
Training	0.80	0.93	0.06	0.84	0.95	0.16
Validation	0.78	0.92	-0.17	0.81	0.94	0.02
Testing	0.82	0.93	0.16	0.86	0.95	0.21
Mean Squared Logarithmic Error (MSLE)						
Training	0.77	0.73	0.16	0.87	0.88	0.15
Validation	0.74	0.72	0.10	0.85	0.87	-0.01
Testing	0.79	0.71	0.33	0.89	0.88	0.18

Best results are in down-sampling, when using MSE as the loss function (0.86, 0.95, 0.21), although, MSLE have better balanced improvement (0.89, 0.88, 0.18), which could have better performance when the sample size or training epochs increase. Additionally, the RNN model had different training times depending on the sampling technique: When using up-sampling, it lasted approximately 70 seconds per epoch, considering 25 epochs $25 \cdot 70 = 1750$ seconds, or



Fig. 4: Example of set up, with a cellphone's camera mounted on a tripod and Intel $D435i$ camera, calibration phase's pose is being done by the current subject.

29.16 minutes; on the other hand, when using down-sampling, it lasted approximately 35 seconds per epoch, considering 25 epochs $25 \cdot 35 = 875$ seconds, or 14.58 minutes. So, not only the down-sampling technique is considered better when using the performance metric, but is also when considering that training time was lower due to the decrease of training examples (from 9000 to 4500 for each subject).

On the other hand, trials are currently being performed using the biometric devices and two cameras: Intel $D435i$ and a cellphone camera, in order to record movement of a person performing a set of exercises that last for about 10 seconds. An example of the experimental set-up can be seen in Fig. 4.

B. Conclusions

As for the results described, a relationship between markers positions and treadmill forces could be established, and thus DL approaches such as RNNs are capable of relating these parameters with high precision (0.95 on R^2 for F_y). This methodology could then be applied to a dataset of our own, created using data from biometric devices and so gather the acceleration of a person, thus relating the human's keypoints to certain value of acceleration over-time.

The process of integration of multiple IoT devices is currently being done, and would greatly provide us with an astounding data to evaluate a human's biomechanics perfor-

mance without wearables, using their meta-data such as age, gender, weight; and a video of them performing movements.

Moreover, additional biomechanical features could be aggregated using other biometric devices that measure other features, such as angle, torque, and so on. Such a flexible tool could be used in multiple scenarios such as geriatric healthcare and sports, which then improve their analysis while still remaining accessible in its usage.

REFERENCES

- [1] G. N. Guten, *Running injuries*. Saunders, 1997.
- [2] M. Fredericson, F. Jennings, C. Beaulieu, and G. O. Matheson, "Stress fractures in athletes," *Topics in Magnetic Resonance Imaging*, vol. 17, no. 5, pp. 309–325, 2006.
- [3] M. Baggaley, M. Esposito, C. Xu, G. Unnikrishnan, J. Reifman, and W. B. Edwards, "Effects of load carriage on biomechanical variables associated with tibial stress fractures in running," *Gait & posture*, vol. 77, pp. 190–194, 2020.
- [4] R. Y. van der Velde, C. E. Wyers, P. P. Geusens, J. P. van den Bergh, F. de Vries, C. Cooper, T. P. van de Staa, and N. C. Harvey, "Incidence of subsequent fractures in the UK between 1990 and 2012 among individuals 50 years or older," *Osteoporosis International*, vol. 29, no. 11, pp. 2469–2475, 2018.
- [5] M. R. McClung, P. Geusens, P. D. Miller, H. Zippel, W. G. Bensen, C. Roux, S. Adami, I. Fogelman, T. Diamond, R. Eastell, P. J. Meunier, and J.-Y. Reginster, "Effect of Risedronate on the Risk of Hip Fracture in Elderly Women," *Obstetric and Gynecologic Survey*, vol. 56, no. 7, pp. 423–424, 2001.
- [6] P. Melissa A. Furlong, PhDa, Dana Boyd Barr, PhDb, Mary S. Wolff, PhDe, 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.
- [7] C. J. Parris, "The Future for Industrial Services - The Digital Twin," *Infosys Insights*, pp. 42–49, 2016.
- [8] F. Geissler, R. Heiβ, M. Kopp, M. Wiesmüller, 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.
- [9] 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.
- [10] I. Voigt, H. Inojosa, A. Dillenseger, R. Haase, K. Akgün, and T. Ziemssen, "Digital Twins for Multiple Sclerosis," *Frontiers in Immunology*, vol. 12, no. May, pp. 1–17, 2021.
- [11] A. Matić, S. Petrović Savić, B. Ristić, V. B. Stevanović, and G. Devedžić, "Infrared assessment of knee instability in ACL deficient patients," *International Orthopaedics*, vol. 40, no. 2, pp. 385–391, 2016.
- [12] N. Rizaldy, F. Ferryanto, A. Sugiharto, and A. I. Mahyuddin, "Evaluation of action sport camera optical motion capture system for 3D gait analysis," *IOP Conference Series: Materials Science and Engineering*, vol. 1109, no. 1, p. 012024, 2021.
- [13] H. S. Fang, S. Xie, Y. W. Tai, and C. Lu, "RMPE: Regional Multi-person Pose Estimation," *Proceedings of the IEEE International Conference on Computer Vision*, vol. 2017-October, pp. 2353–2362, 2017.
- [14] S. Knorz, F. Kluge, K. Gelse, S. Schulz-Drost, T. Hotfiel, M. Lochmann, B. Eskofier, and S. Krinner, "Three-dimensional biomechanical analysis of rearfoot and forefoot running," *Orthopaedic Journal of Sports Medicine*, vol. 5, no. 7, pp. 1–10, 2017.
- [15] A. J. Callaway, "Measuring kinematic variables in front crawl swimming using accelerometers: A validation study," *Sensors (Switzerland)*, vol. 15, no. 5, pp. 11363–11386, 2015.
- [16] R. Robb, *Three Dimensional Biomedical Imaging*. Three-dimensional Biomedical Imaging, CRC-Press, 1985.
- [17] Q. Qi, F. Tao, T. Hu, N. Anwer, A. Liu, Y. Wei, L. Wang, and A. Y. Nee, "Enabling technologies and tools for digital twin," *Journal of Manufacturing Systems*, vol. 58, no. PB, pp. 3–21, 2021.
- [18] J. D. Humphrey and S. L. O'Rourke, *An introduction to biomechanics: Solids and fluids, analysis and design, second edition*. Springer Science + Business Media LLC, 2015.
- [19] S. Madakam, R. Ramaswamy, and S. Tripathi, "Internet of Things (IoT): A Literature Review," *Journal of Computer and Communications*, vol. 03, no. 05, pp. 164–173, 2015.
- [20] 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.
- [21] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime multi-person 2d pose estimation using part affinity fields," in *CVPR*, 2017.
- [22] I. Yoo, "360-540-720-900 cup kick slowmotion," <https://youtu.be/gwNVprZR6fQ>, 2018. Accessed: 2021-04-14.