A Machine Learning-Based System for Human Movement Analysis

Mauro Aloisio Smaniotto[a], Fabio Kurt Schneider[b] and Heitor Silvério Lopes[c]
[a,b]CPGEI, Universidade Tecnológica Federal do Paraná, Av. 7 de Setembro-3165, Curitiba, Brazil
msmaniotto@alunos.utfpr.edu.br, fabioks@utfpr.edu.br, hslopes@utfpr.edu.br

Keywords: Computer Vision, MoveNet, Pose Analysis, Pose Estimation, Software Healthy, Movement Analysis.

Abstract: The analysis of human movement has gained prominence in sports and physical assessment applications, driven by computer vision tools capable of identifying and quantifying biomechanical patterns. Although advanced video-based systems with markers exist, their high cost and complexity limit clinical use. In this context, markerless solutions based on Deep Learning models, such as convolutional networks for detecting body points, offer low-cost alternatives, suitable for home use and requiring less preparation. This work presents software for real-time motion capture and analysis, capable of identifying joints, generating biomechanical reports containing range of motion and cadence, providing immediate feedback through RAG indicators, and allowing analysis with simultaneous cameras or pre-recorded videos. The angle validation against Kinovea software (reference method), using Bland-Altman analysis with 100 samples, demonstrated excellent agreement: mean bias of -0.036°, narrow LoA (-0.408° to 0.336°), and minimal variability, indicating an absence of systematic error and precision exceeding the limits considered excellent in the literature. Applications for this software includes rehabilitation, post-injury follow-up, exercise prescription, sports monitoring, and support for clinical decision-making, strengthening the use of decentralized and accessible motion capture.

1. INTRODUCTION

Human movement analysis is an area of growing interest in sports and physical assessment applications. Computer vision-based tools have demonstrated great potential in identifying and quantifying biomechanical patterns (Dulhazi, et al. 2024; Li, R., et al. 2024) and human body movement (Tamura and Kawasaki, 1988; Gavrila, 1999; Liu, Kortylewski and Yuille, 2023). However, there is still a need for developing real-time and decentralized solutions for application-specific movements, such as those found in Pilates, similarly in rehabilitation scenarios, where the practitioner can receive immediate feedback during the execution of the movement, the professional conducting the assessment could access a final report with quantified joint angles.

Several human motion-based systems have been used in sports, especially through video analysis technologies with reflective markers (such as Vicon, Qualisys, among others). These systems can provide continuous data on essential kinematic and kinetic parameters, including movement, speed, acceleration, and force (Hurley, 2018). On the other hand, the cost of these solutions and prior technological knowledge are two barriers that prevent some healthcare professionals from benefiting from these tools.

Currently, markerless (i.e., without the use of fiducials) motion capture systems comprise a distinct category of motion analysis devices using Machine Learning models based on Convolutional Neural Networks (CNNs) to detect key points on the body. Their advantages are reduced preparation time for measurements, the possibility of home use, and the low technical knowledge, as they only require a camera as a video source (Chen, et al. 2018).

In this context, this work aims to present a Machine Learning software for the detection of joints in the human body (e.g., knee, elbow, shoulder, heel joints). It can generate a real-time biomechanical report of the movement patterns of the individual being observed. The report contains information

[a] https://orcid.org/0009-0003-9507-2022
[b] https://orcid.org/0000-0001-6916-1361
[c] https://orcid.org/0000-0003-3984-1432

about the range of motion and movement cadence over time. Additionally, it can also be used for analyses of previously acquired videos.

Additional features were developed within the tool to aid in the evaluation and analysis of movement, such as the possibility of dual video camera acquisition at different angles (e.g., front and side views at 90 degrees) and real-time feedback on proximity to a target movement range (i.e., a desired angle interval) using the Red, Amber, and Green (RAG). This set of developed features allows applications in different health areas, allowing the analysis of human body movement beyond the optical vision of the professional.

2. MATERIALS AND METHODS

2.1 Software Overview

The tool was developed using the Python programming language, primarily based on the TensorFlow and OpenCV libraries. TensorFlow is a library used to train machine learning models based on neural networks, using predefined datasets (Talele, Patil, and Barse, 2019). More specifically, the model can load video data and detect key points of the body at each video frame. OpenCV handled frame acquisition from either a connected camera or uploaded video files, as well as pre-processing tasks such as dimension adjustment, color conversion, and the insertion of graphical elements representing the estimated pose (Mahamkali & Ayyasamy, 2015). After pre-processing, each frame was passed to the model for keypoint detection. For visualization, OpenCV functions were used to draw joints and connecting lines, enabling geometric information to be overlaid directly onto the human-body image.

The frame-processing pipeline is organized into three parallel processes connected through input and output queues to ensure efficient, real-time performance, as shown in Figure 1. Process 1 is responsible for capturing frames either from a video file or a live camera feed and placing them into the input queue. Process 2 consumes these frames and performs inference using the MoveNet model to detect body keypoints, sending both the processed frame and its corresponding keypoints to the output queue. Process 3 retrieves this information from the output buffer and handles the visualization and analytical tasks. It draws joint points and skeletal lines on the frame, calculates joint angles, and saves angle data into a .csv file. Additional features are also executed at this stage. Finally, the annotated frame is sent to the user's screen as visual feedback in real time.

To optimize performance, queues (buffers) and parallel processes were used to capture frames, perform model inference, and display results in real time. This architecture distributes computationally demanding tasks across multiple processor cores, preventing slowdowns that occur when one stage of the pipeline takes longer than the others.

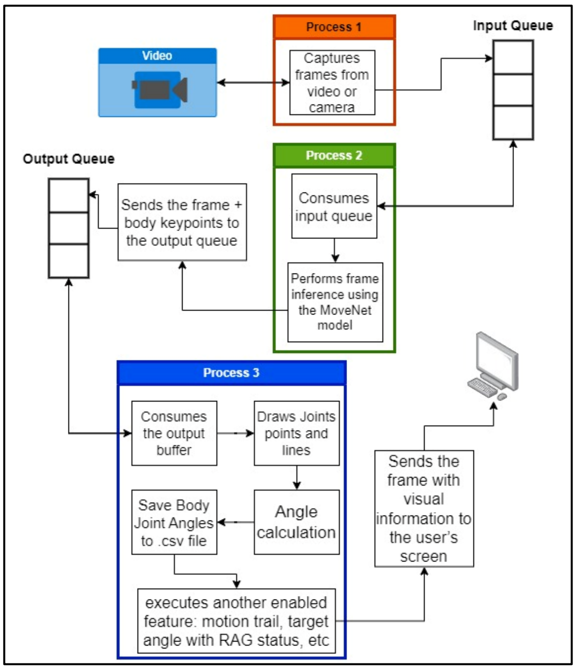

Figure 1: Software Overview

2.2 Video Capture

For both video capture methods (real-time source such as a webcam or previously recorded and uploaded video) the form factor for better inference with the machine learning model used is the 1:1 aspect ratio using a 256x256 pixels spatial input resolution. Automatic or manual cropping could be used but flattened or stretched images have the potential for degrading the system response.

The current maximum spatial resolution is 1440x1440 pixels. Resolutions higher than recommended may compromise the inference response of each frame with the model, increasing the processing time of each image and generating incorrect joint coordinates estimations. Maintaining that resolution ensures a minimum frame rate of 30 frames per second (FPS). Certain software features depend on accurate frame time stamps to operate reliably.

The recommended camera positioning relative to the individual performing the movement, depends on the angle to be studied, ensuring that the camera is at a 90° angle to the plane of movement. For example, in Figure 2, the angle under analysis is that of the right elbow joint (J1) during flexion and extension.

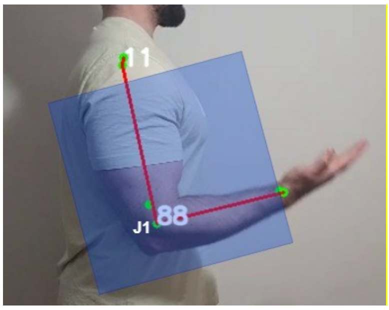

Figure 2: Elbow flexion and extension plane

The movement's plane is represented by the blue square, Figure 3 shows the camera's orientation for this case. This angle minimizes the parallax effect reducing incorrect representations of the angles of interest.

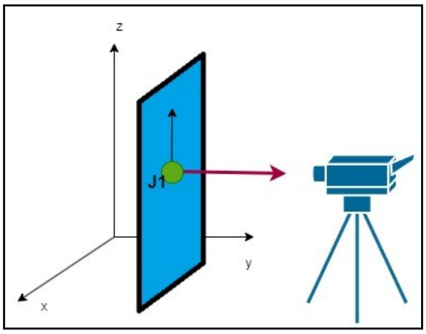

Figure 3: Camera pointing

2.3 The Machine Learning Model

The model used for detecting body joints was the MoveNet (Votel and Li, 2021), made available by Google through Kaggle. This model was designed to quickly and accurately detect relevant body points coordinates (e.g., head, shoulders, and joints of the upper and lower limbs) in images and videos. This model was developed to operate efficiently on devices with limited computing resources. It uses compact architectures derived from deep networks, such as variants of MobileNet (Votel and Li, 2021), in order to minimize the number of operations required during inference without significantly compromising accuracy.

MoveNet was trained using two different datasets, specifically COCO (Lin et al., 2014) and Active (Votel and Li, 2021). COCO is a widely used dataset for model training due to its free use by researchers, students, and companies (Li, et al. 2024; Chen, et al. 2018). However, some studies point to a demographic imbalance in the dataset's composition, and not satisfactorily covering unusual poses (Liu, Kortylewski and Yuille, 2023; LaChance, et al. 2023). Active, on the other hand, is an internal Google dataset produced with YouTube videos of people exercising (e.g., high-intensity interval training, weight-lifting, stretching, dancing) (Votel and Li, 2021). These characteristics bring a significant performance improvement when compared only to the COCO dataset (Votel and Li, 2021).

There are two variants of MoveNet, the Lightning model and the Thunder model. The Lightning variant was prioritized for computational performance and can operate above 50 FPS, particularly suitable for devices with limited resources. However, the quality of keypoint predictions is affected. The Thunder variant is a more complex architecture, providing more accurate and stable predictions when compared to Lightning in more complex poses or situations of partial body occlusion in the video.

In this work, we chose to use MoveNet SinglePose Thunder due to its favorable balance between accuracy and computational efficiency, especially when compared to architectures based on OpenPose (Jo and Kim, 2022) and other more complex pose estimation models. Although OpenPose derived models or more robust frameworks offer competitive performance, they generally have higher computational costs, higher memory consumption, and depend on more powerful GPUs to operate in real time (Jo and Kim, 2022). In contrast, MoveNet Thunder maintains high prediction quality even on resource-limited devices such as conventional laptops and some mobile devices, achieving frame rates above 30 FPS without the need for dedicated hardware.

MoveNet Thunder produces 17 post-inference body joints based on the COCO pattern as shown in Figure 4. Each resulting point has its confidence score information given, allowing the developed software to determine whether to accept or discard the estimation based on a user-defined threshold.

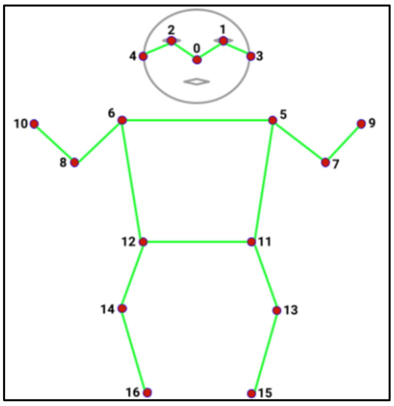

Figure 4: COCO standard Keypoints

2.4 Angle Calculation

Each vector is defined by the coordinates (x, y) of the joints obtained after inferring the MoveNet model.

For each pair of segments that share a common point P_1, we have vectors $v1$ and $v2$ defined by (1) e (2) and shown in Figure 5.

$$v1 = (x2 - x1, y2 - y1) \tag{1}$$

$$v2 = (x3 - x1, y3 - y1) \tag{2}$$

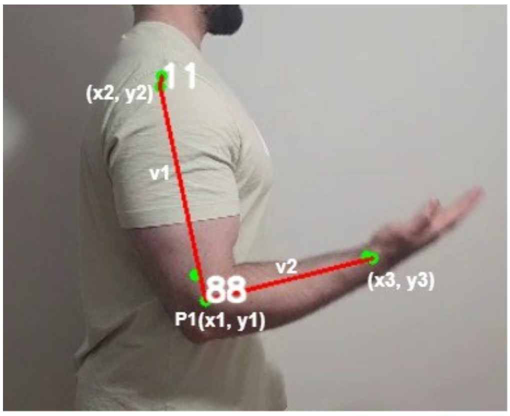

Figure 5: Vectors $v1$ and $v2$ sharing the same Joint P_1.

The absolute angle of each vector with respect to the x-axis is determined in Equations (3) and (4), they simultaneously evaluates the horizontal and vertical components, allowing each vector quadrant to be correctly identified within the interval. $(-\pi, \pi]$.

$$\alpha_1 = \arctan 2(y2 - y1, x2 - x1) \tag{3}$$

$$\alpha_2 = \arctan 2(y3 - y1, x3 - x1) \tag{4}$$

In other words, the arctan2 function returns the correct angle in each quadrant, internally delimited by Equation (5), so that it is always within the interval of $(-\pi, \pi]$, unlike the arctan function for (y/x), which does not adequately distinguish the quadrants and becomes undefined when $x = 0$.

$$\begin{cases} \tan^{-1}\frac{\Delta y}{\Delta x}, & \Delta x > 0, \\ \tan^{-1}\frac{\Delta y}{\Delta x} + \pi, & \Delta x < 0, \Delta y \geq 0 \\ \tan^{-1}\frac{\Delta y}{\Delta x} - \pi, & \Delta x < 0, \Delta y < 0 \\ +\frac{\pi}{2}, & \Delta x = 0, \Delta y > 0 \\ -\frac{\pi}{2}, & \Delta x = 0, \Delta y < 0 \\ undefined\ (null\ vector), & \Delta x = 0, \Delta y = 0 \end{cases} \tag{5}$$

Based on the angular orientations α_1 and α_2 obtained for each vector, the angle between the articular segments is determined by the smallest angular difference between them, as shown in Equation (6).

$$\theta = min\,(|\alpha_1 - \alpha_2|, 2\pi - |\alpha_1 - \alpha_2|) \tag{6}$$

Finally, the value calculated in radians of (6) is converted to degrees to facilitate biomechanical interpretation according to Equation (7).

$$\theta_{\ degrees} = \theta \cdot \frac{180}{\pi} \tag{7}$$

In this work, the combination of extracting vectors from the MoveNet model inference along with the arctan2 function to calculate joint angles, will be referred from here as the MJ-Angle.

2.5 Biomechanical Report

The joint angles are calculated at each new frame inference of the model and saved in a .csv file containing the frame number, the video or recording time, and the angles of all joints within that frame. In this way a biomechanical report is produced containing information for each body joint, such as: range of motion (maximum and minimum), symmetry information if the movement is bilateral, and graphs of angle variation (degrees) over time (seconds).

Most of the natural body movements do not require very high rates, such as 60 or 120 FPS, Therefore, the video capture rate was set to 30 FPS, since this is sufficient to clearly capture most human movements present in physical exercise, rehabilitation, ergonomics, and gestural interactions.

This setup allows detail of postural transitions without generating an excessive volume of data.

2.6 Angle Validation

To validate the method for calculating joint angles using keypoints from MJ-Angle, the Kinovea software (Dalal, et al. 2018) was used. For the angle accuracy evaluation, 100 frames were extracted from 5 different videos, angles between 20° and 170° were generated by MoveNet and annotated using Kinovea's angle drawing tool, as shown in Figure 6. To conduct the validation, the corresponding values from the two methods were subtracted, thus obtaining the differences between the angles estimated.

Next, a Bland-Altman analysis was applied to assess the agreement between the two measurement techniques. This statistical procedure is widely recommended for validation studies of measurement methods, since it allows the identification of both systematic bias between methods and the expected range of point-to-point differences (Bland and Altman, 1986; Bland and Altman, 1999).

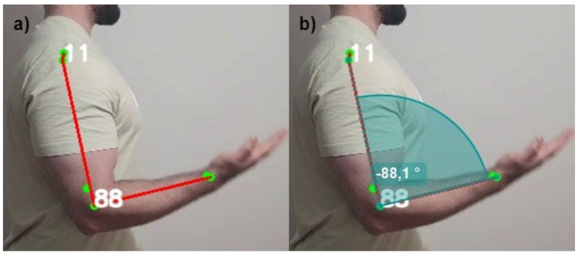

Figure 6: Similar angles of the right elbow (lateral view) calculated by the MoveNet model 88° (a) and by Kinovea 88.1° (b)

3. RESULTS AND DISCUSSION

3.1 Keypoints and Angles

Figure 7 shows the keypoints representing the body joint points as well as the angles between the segments involved. The average time for each frame inference with the model was 29.0ms for a video with a resolution of 1440x1440 in an AMD Ryzen 5 5600 CPU-based computer without dedicated graphics card (GPU) to assist in the inference.

Figure 7: Keypoints and Joint Angles - Single Capture

For the mode with two simultaneous cameras shown in Figure 8, both with a resolution of 640x480, the average time for frame inference with the model was 32.0 ms with the same configuration approach for the single camera capture mode.

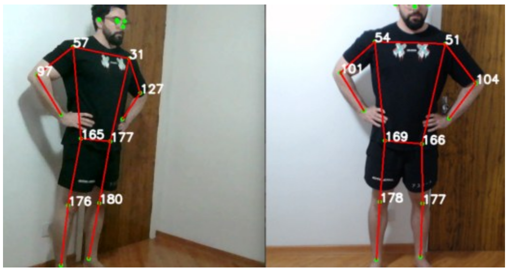

Figure 8: Keypoints and Joint Angles - Simultaneous capture

3.2 Angle Validation between MJ-Angle and reference method (Kinovea)

The results for the angle validation metrics are presented in Table 1.

Table 1: Validation metrics for MJ-Angle and Kinovea angles with 100 samples.

Metric	Value
Mean of the Differences	-0.0360°
Standard Deviation of the Differences	0.1899°
Standard Error	0.0189°
t-statistic	-1.8956
p-value	0.0609
LoA sup.	0.3360°
LoA inf.	-0.4080°

A Bland-Altman analysis performed on 100 samples revealed high agreement between the methods in joint angles. The mean bias (Mean of the Differences) was -0.036°, demonstrating that, on average, MJ-Angle recorded values only slightly lower than those obtained by Kinovea. This difference is practically negligible and has no biomechanical relevance. The standard deviation of the differences (SD = 0.18991°) indicates low point-to-point variability, reinforcing the consistency between the two methods.

The standard error of the mean (SE = 0.0189°) and the associated t-test ($t(99) = -1.8956$) resulted in a two-sided p-value of 0.06092, demonstrating that the mean bias does not differ significantly from zero at the 5% significance level. This result reinforces the absence of systematic error between the methods, indicating that there is no consistent tendency for MJ-Angle to overestimate or underestimate compared to Kinovea.

The Limits of Agreement (LoA) were calculated as −0.408° (lower limit) and 0.336° (upper limit), representing the range within which 95% of the differences between the methods are contained, as shown in Figure 9. These values are extremely narrow, with a total amplitude of less than 0.75°, which demonstrates excellent agreement and precision. The confidence intervals reinforce the stability of the estimates (Lower LoA = −0.44578 to −0.37066; Upper LoA = 0.29866 to 0.37378).

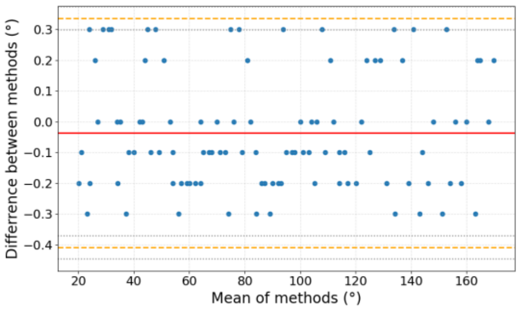

Figure 9: Bland-Altman plot comparing MJ-Angle and Kinovea methods. Scatter plot of the differences between measurements versus their mean values (blue points). The solid red line represents the mean difference, while the dashed orange lines indicate the lower and upper limits of agreement (LoA)

These results indicate that the difference between the methods is statistically irrelevant, that there is no systematic bias, and that point-to-point discrepancies remain consistently below 1°, with typical values below 0.2°. The literature demonstrates that errors between 2° and 5° are considered acceptable for human movement measurements in intra- and inter-rater contexts (Mullaney and McHugh, 2006), and that limits of agreement (LoA) below ±2° are already classified as excellent in biomechanical analyses (Millet, et al. 2002). Therefore, the observed performance with LoA close to ±0.4°, significantly narrower than the reference values, confirms that MoveNet performs equivalently to Kinovea for measuring elbow flexion within the analyzed range.

3.3 Biomechanical Report Examples

To exemplify typical applications in our research group, two videos, recorded in the same day, of the Pilates apparatus lunge exercise, partially represented in Figure 10, were uploaded to the system, one with the right leg on the moving part of the apparatus and the other with the left leg on the moving part.

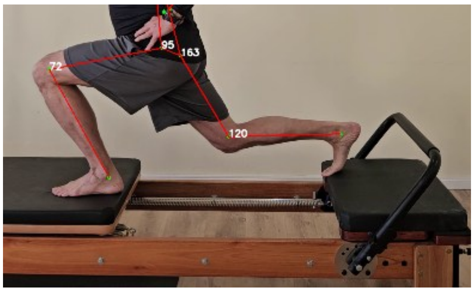

Figure 10: Lunge Pilates

As a result, two reports were generated, one for each video. The report for the left knee showed a maximum flexion of 66° and an extension of 177°, with full Range of Motion (ROM) of 111°. This information is graphically represented in Figure 11. It can also be observed the angular variation of the left knee along time, where a lack of cadence (time between extension and flexion) among repetitions and unstable maximum flexions, with the peak angle varying for each flexion.

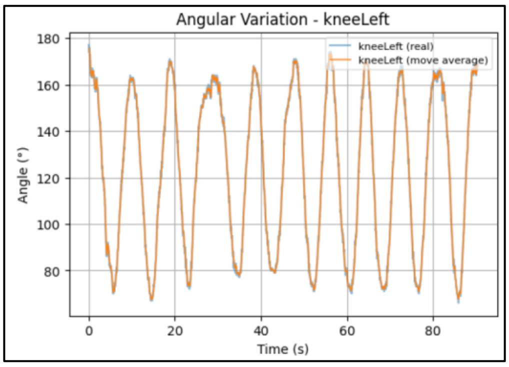

Figure 11: Angular variation - KneeLeft

In the right knee report, were observed a maximum flexion of 56° and an extension of 180°, with a ROM of 124° as shown in Figure 12. Different from the left side analysis, the peaks of right knee flexion appear more stable along movements, remaining close to 60° and with a similar movement cadence in each repetition.

Therefore, it is noted that the individual analyzed through the videos has better control, stability, and range of motion on the right knee in front of the body. Additionally, for the same time interval, a superior cadence and agility was observed, result of one extra repetition when compared to the left knee

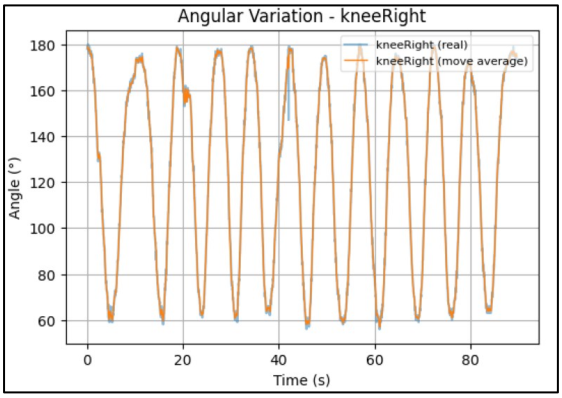

Figure 12: Angular variation - KneeRight

Angular variation reports can aid in identifying post-injury joint deficits and in monitoring orthopedic post-operative patients. Integrating this data into medical records may contribute to more accurate decisions, reducing risks and increasing the predictability of prognoses. (Damm, et al. 2023; Paul, et al. 2021).

Additionally, physical therapists can use these reports to monitor the impact of specific exercises, adjust training loads, assess the patient's response to treatment, and clearly communicate the progress achieved, including the patient's progress history in the reports. (de Raedt, et al., 2018; Mazza, et al., 2023).

In sports training, the systematic use of this tool allows for comparison between the athlete's movement pattern and reference parameters established in literature or by elite athletes. In this way, coaches can adjust techniques, correct compensations, optimize muscle action time, and develop personalized motor improvement strategies (Sinha, et al. 2023; Jones, et al. 2024).

3.4 Target Angle + RAG Status

A color-coded identification status feature was added using red, amber, and green (RAG Status) code according to the proximity of the current joint angle to the previously chosen target angle. This setting is present in both single-camera and dual-camera simultaneous capture modes. Each capture can have one preselected angle as the target.

Figure 13 shows an example of the color variations in RAG Status, starting with red when the current angle is far from the target angle, and the colors change until reaching green, which is very close to the target angle. This method is important for providing feedback on how far or close the movement practitioner is to the pre-established ideal position.

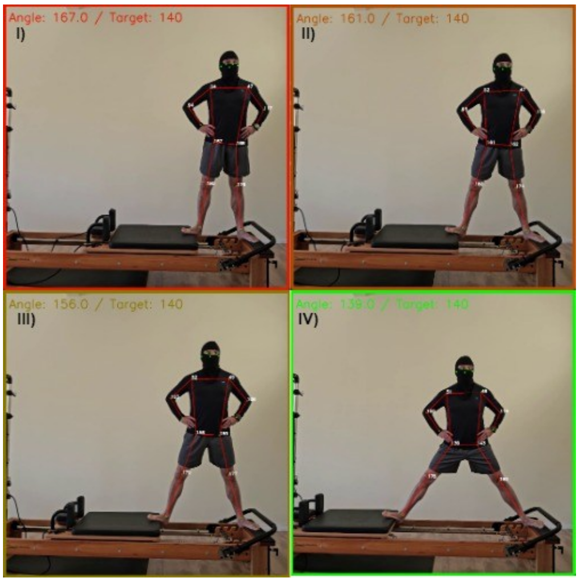

Figure 13: RAG Status on Right Hip in Target Angle Feature. Sequence starting in (I)

An example of RAG Status in simultaneous capture mode using two cameras with different pointing directions is presented in Figure 14.

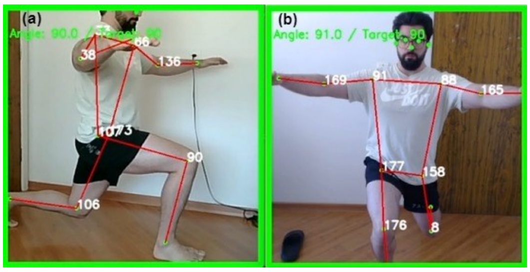

Figure 14: RAG Status - Simultaneous capture. Targe angle 90° to right knee (a) and target angle 90° to lateral raises arms (b).

3.5 Limitations

As a limitation, depending on the chosen model, the MoveNet SinglePose Thunder can only process one individual per video, requiring him/her to be closer and more centrally located in the shot if there are multiple people in the video. Additionally, the video must be taken from a distance between 90-180 centimeters. If these requirements are not met, the system may fail to correctly identify joints.

4 CONCLUSIONS

The proposed tool to be proved effective in capturing and analysing angular variation over time, providing clear and objective insights to the user. By generating continuous and interpretable reports, it supports the detection of movement deviations, the monitoring of user progress, and more informed decision-making in both clinical and sports contexts. Overall, the system offers a practical and reliable solution for enhancing biomechanical assessment and guiding individualized intervention strategies.

For the proposed angle calculation method (MJ-Angle), the results suggest that the difference between the reference method is statistically and clinically irrelevant, that there is no systematic bias, and that point-to-point discrepancies remain consistently below 1°, with typical values below 0.2°.

The next steps of this project are developing a user-friendly interface and put the system on a server so that researchers and healthcare professionals can use it through a website or mobile application, uploading videos to extract reports on angle variations over time.

REFERENCES

Atkinson, G., Nevill, A. M. (1998). *Statistical methods for assessing measurement error (reliability) in variables relevant to sports medicine.* Sports Medicine, 26(4), 217-238

Bland, J. M., Altman, D. (1986). *Statistical methods for assessing agreement between two methods of clinical measurement.* The Lancet, 1, 307–310.

Bland, J. M., Altman, D. G. (1999). Measuring agreement in method comparison studies. In *Statistical Methods in Medical Research*. SAGE Publications, 8(2), 135–160.

Chen, Y., et al. (2018). *Cascaded Pyramid Network for Multi-Person Pose Estimation.* arXiv:1711.07319.

Dalal, K. K., et al. (2018). *Effectiveness of prowling with proprioceptive training on knee hyperextension among stroke subjects using videographic observation – a randomized controlled trial.* Gait & Posture, 61:232-237

Damm, M., et al. (2023). *Postoperative activity and knee function of patients after total knee arthroplasty: A sensor-based monitoring study.* Sensors, 23(1), 123.

de Raedt, S., et al. (2018). *Accuracy of measuring knee flexion after total knee arthroplasty using wearable inertial sensors.* Sports, 6(3), 60.

Dulhazi, F., et al. (2024). *Software de análise de movimento baseado em IA para avaliação de esporte e fisioterapia.* Revista Brasileira de Medicina do Esporte, 30, e2022_0020.

Gavrila, D. M. (1999). *The visual analysis of human movement: A survey.* Computer Vision and Image Understanding, 73(1), 82–98.

Hurley, O. A. (2018). *Types of technology commonly used in sport.* In: Sport Cyberpsychology.

Jo, B., & Kim, S. (2022). Comparative analysis of OpenPose, PoseNet, and MoveNet models for pose estimation in mobile devices. Traitement du Signal, 39(1), 111–119.

Jones, R. M., et al. (2024). *Orthopedic surgeon–level joint angle assessment with artificial intelligence based on photography: A pilot study.* Journal of Orthopaedic Surgery and Research, 19(1), 45.

LaChance, J., et al. (2023). *A Case Study in Fairness Evaluation: Current Limitations and Challenges for Human Pose Estimation*. AAAI R2HCAI Workshop.

Li, R., et al. (2024). *An efficient and accurate 2D human pose estimation method using VTTransPose network.* Scientific Reports, 14, Article 7608.

Lin, T.-Y., Maire, M., Belongie, S., Hays, J., Perona, P., Ramanan, D., Dollár, P., & Zitnick, C. L. (2014). Microsoft COCO: Common objects in context. *European Conference on Computer Vision (ECCV)*, 740–755.

Liu, Q., Kortylewski, A., Yuille, A. (2023). *PoseExaminer: Automated testing of out-of-distribution.* arXiv:2303.07337.

Mahamkali, R., & Ayyasamy, A. (2015). *OpenCV for computer vision applications*. Proceedings of the International Conference on Computing and Communication Technologies (ICCCT).

Mazza, M. E., et al. (2023). *The reliability and validity of joint range-of-motion measurement using Zoom and a smartphone application.* Journal of Telemedicine and Telecare, 30(3), 445–454.

Millet, G.P., et al. (2002). *Accuracy and reliability of a miniature accelerometer-based system in the assessment of upper-limb movement.* Journal of Biomechanics, 35(1), 87–93.

Mullaney, M.J., & McHugh, M.P. (2006). *Reliability of shoulder range of motion measurements in athletes.* Journal of Shoulder and Elbow Surgery, 15(2), 166–171.

Paul, D., et al. (2021). *Diagnostic accuracy and clinical applications of wearable movement sensors for knee rehabilitation: A systematic review.* Sensors, 21(3), 1234.

Sinha, S., et al. (2023). *Strengths and weaknesses of 3D pose estimation and inertial motion capture systems for movement therapy.* arXiv preprint arXiv:2306.06117.

Talele, A., Patil, A., & Barse, B. (2019). *Detection of Real Time Objects Using TensorFlow and OpenCV.* Asian Journal For Convergence In Technolog, 5(1), 1–4

Tamura, S., Kawasaki, S. (1988). *Recognition of sign language motion images.* Pattern Recognition, 21(4), 343-353

Votel, R., Li, N. (2021, May 17). *Next-Generation Pose Detection with MoveNet and TensorFlow.js.* In TensorFlow Blog.