TENCON 2023 - 2023 IEEE Region 10 Conference (TENCON)
31 Oct - 3 Nov 2023. Chiang Mai, Thailand



# Bridging Exercise Monitoring System Using RGB Camera for Stroke Rehabilitation

Khemwutta Pornpipatsakul
*Department of Mechanical Engineering*
*Chulalongkorn University*
Bangkok, Thailand
6470007521@student.chula.ac.th

Wasutha Chuengwutigool
*International School of Engineering*
*Chulalongkorn University*
Bangkok, Thailand
6338197921@student.chula.ac.th

Ronnapee Chaichaowarat *
*International School of Engineering*
*Chulalongkorn University*
Bangkok, Thailand
ronnapee.c@chula.ac.th

Anchalee Foongchomcheay
*Department of Physical Therapy*
*Chulalongkorn University*
Bangkok, Thailand
anchalee.f@chula.ac.th

***Abstract*—Bridging exercise is a widely applied training for stroke rehabilitation to improve balancing ability on weight-bearing activities. Aiming to reduce the workload of physical therapists and enable the systematic recording of motion data, this paper presents an affordable rehabilitation monitoring system using an RGB camera. For predicting the correctness of the bridge posture, the MediaPipe framework is applied for detecting the human body segments which are used as the input data of the decision tree classifier instead of using a complex neural network. The model was trained using the data collected from five healthy participants performing the correct and Wide Knee postures when the knees are separated laterally. The experimental results show that nearly 100 percent accuracy can be achieved in confirming the correct posture and identifying the Wide Knee posture. The time performance of the decision tree classifier trained by the different number of frames is also evaluated. This system is very promising to help therapists monitor patients and provide feedback for improving the effectiveness of the rehabilitation.**

***Keywords—Rehabilitation monitoring, bridging exercise, posture recognition, machine learning***

## I. INTRODUCTION

In most regions worldwide, the population aged over 65 is growing faster than the total population. Physical deterioration is inevitable and stroke risk increases with age [1]. The advances in physical human–robot interaction e.g., impedance control [2]–[5] and robotic actuators with adjustable intrinsic properties [6]–[9] for rehabilitation offer an opportunity to increase the amount of motor practice required to relearn motor skills lost while potentially reducing therapist participation.

For improving the balance ability of stroke patients, bridging exercise is widely applied for rehabilitation [10]. The exercise has a positive impact on weight-bearing activities such as lifting the pelvis in bed and sit-to-stand motion [11], which requires large hip and knee extension moments [12]. The muscular activities vary with the posture and are strongly relied on the technique of bridging exercise [13]–[15]. Appropriate monitoring of the human posture is important for providing feedback and improving the effectiveness of the exercise.

Cameras have been used in previous studies for stroke rehabilitation therapies. The camera-based mirror visual feedback (camMVF) system can be set up using an RGB camera to improve the effectiveness of traditional mirror therapy [16]. Serious games were developed with motion-capture systems, using different types of cameras, to gamify the rehabilitation process [17]. However, most systems are developed primarily for the upper body rehabilitation. The system proposed in this paper emphasizes the lower body recovery.

Machine learning is a powerful tool that affects many industries including the medical field. The machine learning techniques were applied for assistive devices [18], [19] and for enhancing the clinical protocols on homecare rehabilitation [20]. The position and orientation of an ultrasound probe can be observed by using an RGB-depth camera to track the hand gesture [21]. Human activities can also be classified based on the 2D skeleton data by using the supervised model [22]. The neural network models and the Support Vector Machine [23] were commonly used for human activity recognition, body language recognition, and falling detection.

Aiming to reduce the workload of physical therapists on observing rehabilitation using bridging exercise and enable the systematic capturing of motion data, this paper presents an affordable rehabilitation monitoring system using an RGB camera. The MediaPipe framework is applied for detecting the human body segments which are used as the input data of the decision tree classifier (instead of a more complex neural network) for predicting the correctness of the bridge posture. Instead of classifying different postures, the error of posture (as compared to the ideal motion) is our focus. The model was trained by using the data collected from five healthy participants performing the correct posture and the "Wide Knee" posture when the knees are separated laterally. The time performance of the decision tree classifier trained by the different number of frames is also evaluated.

This paper is organized as follows: Section II provides the overview of system architecture consisting of posture detection and machine learning for posture classification. Section III describes the experimental setup and method for recording, training, and validating data for detecting and classifying the correctness of the prediction. Section IV discusses the results of experimental validation. Section V discusses the ongoing effort and Section VI summarizes the key findings.

The first and second authors contributed equally to this work.

979-8-3503-0219-6/23/$31.00 ©2023 IEEE

## II. System Architecture

### A. Posture Detection

For monitoring the bridge stand rehabilitation, detecting the position and orientation of the human body elements is the most important requirement. The Intel RealSense D435 RGB-Depth camera is used in this study. Aiming to provide an affordable solution for practical adaptations, our proposed technique only uses the RGB array of the camera. The data captured from the camera is in the camera frame, which is inconvenient to be compared in the world frame. One of the useful methods for mapping images between two different frames is the ArUco marker detection [24]. The ArUco method can return a transformation matrix that can be used to map the 3D array data in the camera frame to the desired frame. The three ArUco markers were placed at the corners of the bed, see Fig. 1, to optimize the accuracy of the transformation matrix. The output frame is parallel to the human body for convenient calculation in further steps.

The camera was located at the position where all three ArUco markers could be observed. Although there are many possible positions where the camera can be placed, there are two major concerns while using the system. First, the camera angle should not come from the foot due to the comfortable feeling of the patient while bridging. Second, the camera may not be placed from a patient's head because the patient's hip and knee will block the ankle, so the ankle position cannot be detected. Then, the camera should be in the left or right-side view instead.

For human detection, human pose estimating using MediaPipe [25] was used. The resulting position data is represented as a 3D array, relative to the camera frame. On the other hand, all three ArUco markers were used to generate transformation matrices for mapping the camera frame to the bed frame. Finally, the position data was multiplied to be respected to the bed frame for further calculation.

Even though the system is composed of the ArUco and the MediaPipe, the difference in body segment detected value can be observed because of the estimated position and function in the background, and the visibility of each ArUco and the body segment. The easy way to handle these problems is to set a visibility threshold for each function. The thresholds that were optimized by considering the prediction accuracy and speed are as follows, respectively:

- The minimum perimeter rate for detecting each of ArUco is 0.2 where the actual size of all ArUco markers is 0.181 m. This setting increases accuracy in all transformation matrices from the ArUco. In addition, the camera is also semi-fixed due to this constraint because it may not detect the marker if the camera is far away.
- While all three ArUco markers are detected, the calibration process is started. The duration of detecting the markers for recording transformation matrices was around five seconds. Once some marker is not detected, the calibration process will be restarted when all the markers are detected again. After that, the recorded matrices were averaged to get a single transformation matrix for each marker. This process was invented to ensure the correctness of the transformation matrices before use.

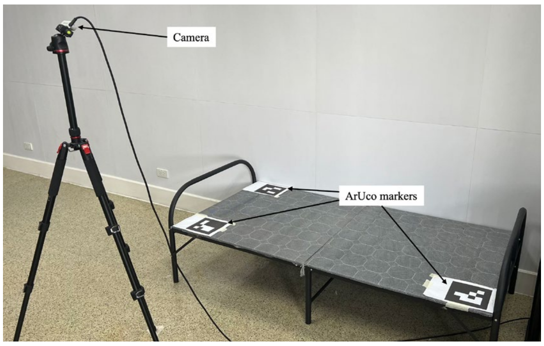


Fig. 1. The rehabilitation monitoring system uses an RGB camera with three ArUco markers.

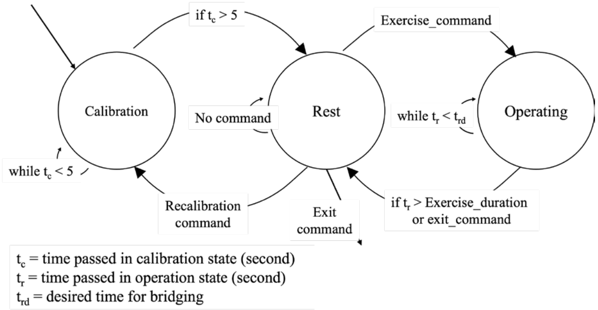


Fig. 2. The state flow of the system from start to end.

- The minimum human detection confidence and tracking confidence of the MediaPipe was set to 0.75 which the Static image mode was not used. Another threshold that convinces the accuracy of the model is the visibility of all necessary positions. Then, if the visibility of any observed position is less than the threshold, of 0.8, the system will not use the captured data for predicting the correctness of the current posture.

For a better understanding of the system, Fig. 2 shows a state chart of the system from the beginning to the end.

### B. Posture Classification

The data output from the posture detection is eight three-dimensional position data. In this paper, the focus is on detecting errors in the areas of the hips and knees during the process of performing the bridging exercise. Thus, the 3D coordinates of the shoulders, hips, knees, and ankles are extracted using the aforementioned MediaPipe in the frame of reference of ArUco makers. Taking into account both right, and left sides, and time, 25 float-type data are collected: x-coordinates, y-coordinates, and z-coordinates of both right and left joints.

This research is done to deploy a model to assist a physical therapist's diagnosis in real time. The artificial intelligence model should be able to correctly identify the correctness of the posture during the static hold phase of the bridging exercise.

The goal of this paper is to correctly classify a patient's posture into two categories: “Correct”, where the knees are properly aligned with the hips and ankles during the static phase of the exercise (Fig. 3); and “Wide Knee”, where the knees extend outward, away from the hips and ankles (Fig. 4). Furthermore, when deployed, the model should maintain roughly 10 frames per second.

The problem at hand is a classification problem based on numerical inputs. To ensure the delivery of the minimum viable product, the decision tree model is selected for its faster rate of computational speed relative to neural network models. Furthermore, there is the choice to use input from multiple frames together as the input for the decision tree model.

More advantages of the decision tree model include: easy implementation to rapidly validate the data collection methods; easy visualization of the model, easier communication with the physical therapy department to validate the research result, and may assist in training any new physical therapist.

## III. Collecting, Training, and Validating Data

### A. Data Collection

All the data used to train for the model was collected in-house. Each data collection session lasts 10 seconds in only the static position. For each session, only the time and x-y-z position data of the eight landmarks are considered. During each session, the data that fell below the confidence threshold is not collected.

The model used in this paper was trained using data collected from five participants. The five participants' ages ranged from 19 years old to 23 years old. There are one female and four males. The height of the participants ranges from 158 cm to 173 cm. Each participant performed the correct posture and the "Wide Knee" posture, both knees were forced to be far from the other side, varying five times for each. In the end, a total of 25 sessions were recorded. All the x-y-z position data were in the frame of reference of three ArUco sensors, the bed frame. The recorded x-y-z data in the transformed coordinate from a trial are shown in Fig. 3 for the correct posture and Fig. 4 for the wide knee posture.

### B. Data Processing

Despite the position data being collected, using the data as input for the decision tree model is likely to lead to overfitting issues. Thus, further data needs to be derived to serve as the input of the decision tree model. The knee joints behave similarly to a revolute joint; thus, an angle can be calculated from each knee. The hip joints behave similarly to a ball joint; thus, three angles can be calculated from each hip. A total of eight angles can be calculated for each given frame which are the knee, yaw, pitch, and roll angle of each leg. For getting the mentioned angle, all of the eight raw data points which are the transformed data respected to the bed frame were used to calculate the position data comparing to the body origin point ($Org$) such as left and right shoulder ($Shoulder_{L,R}$), hip ($Hip_{L,R}$), knee ($Knee_{L,R}$), and ankle ($Ankle_{L,R}$) position of each side via the following equations. Each point can be observed in Figs. 3 and 4, where the $Org$ positioned at the center point between the left and right.

$$Org = \frac{\left(Hip_{L\_Raw}+Hip_{R\_Raw}\right)}{2} \tag{1}$$

$$Shoulder_{L,R} = Shoulder_{L_{raw},R_{raw}} - Org \tag{2}$$

$$Hip_{L,R} = Hip_{L_{raw},R_{raw}} - Org \tag{3}$$

$$Knee_{L,R} = Knee_{L_{raw},R_{raw}} - Org \tag{4}$$

$$Ankle_{L,R} = Ankle_{L_{raw},R_{raw}} - Org \tag{5}$$

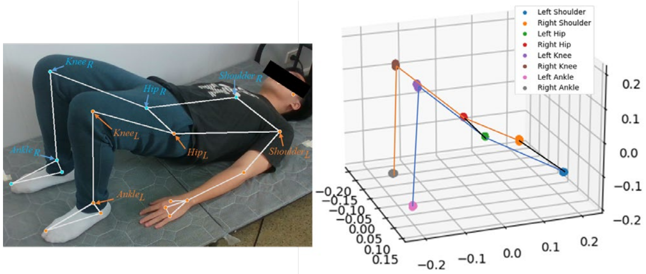


Fig. 3. The x-y-z data of the correct posture in the transformed coordinate.

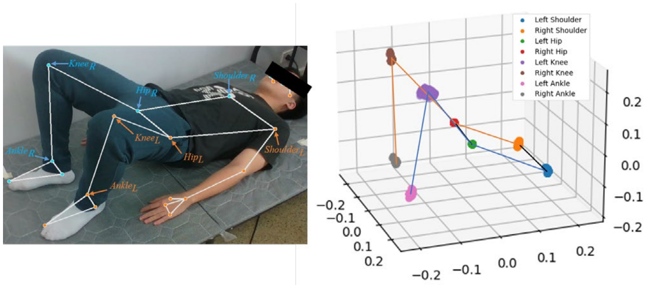


Fig. 4. The x-y-z data of the Wide Knee posture in the transformed coordinate.

Then, the necessary vectors for determining body vector ($B_V$), the vector from the left shoulder to the right ankle ($Shoulder_L2Ankle_R$) and from the right shoulder to the left ankle ($Shoulder_R2Ankle_L$) were calculated via the following equations:

$$Shoulder_L2Ankle_R = Ankle_R\text{-}Shoulder_L, \tag{6}$$

$$Shoulder_R2Ankle_L = Ankle_L\text{-}Shoulder_R. \tag{7}$$

$$B_V = Shoulder_L2Ankle_R \times Shoulder_R2Ankle_L, \tag{8}$$

where the × represents the cross product.

The $B_V$ was further used to determine the roll angle of both legs. The roll angle was defined by the angle between the body vector and knee vector ($K_V$) which can be found by using the thigh ($Thigh_{L,R}$) and shank ($Shank_{L,R}$) vectors via the following equation:

$$Thigh_{L,R}=Knee_{L,R}\text{-}Hip_{L,R} \tag{9}$$

$$Shank_{L,R}=Knee_{L,R}\text{-}Ankle_{L,R} \tag{10}$$

$$K_{V_{L,R}}=Thigh_{L,R}\times Shank_{L,R} \tag{11}$$

The other vectors that are required for finding yaw and pitch angles are the projected thigh ($Thigh_{L_{Proj},R_{Proj}}$) which are the thigh vector in the hip level in each side, vector from the left to the right hip, and the right to the left hip ($Hip_{L2R,R2L}$).

$$Thigh_{L_{Proj},R_{Proj}}=\{Thigh_{L,R}(x),\ Thigh_{L,R}(y),\ Hip_{L,R}(z)\} \tag{12}$$

$$Hip_{L2R,R2L}=Hip_{R,L}\text{-}Hip_{L,R} \tag{13}$$

All the mentioned vectors were used to determine knee angle ($KA_{L,R}$), yaw angle ($\psi_{L,R}$), pitch angle ($\theta_{L,R}$), and roll angle ($\varphi_{L,R}$) for both sides of the leg by the following equations. Finally, all the calculated angles were further used for creating a machine-learning model.

$$KA_{L,R} = Thigh_{L,R}\otimes Shank_{L,R} \tag{14}$$

$$\psi_{L,R} = Thigh_{L_{Proj},R_{Proj}}\otimes Hip_{L2R,R2L} \tag{15}$$

$$\theta_{L,R} = Thigh_{L_{Proj},R_{Proj}} \otimes Thigh_{L,R} \tag{16}$$

$$\varphi_{L,R} = B_V \otimes K_{V_{L,R}} \tag{17}$$

where the $\otimes$ represents the finding of an angle between the vectors:

$$a \otimes b = \cos^{-1}\left(\frac{a \cdot b}{\|a\|\|b\|}\right), \tag{18}$$

where $a \cdot b$ is the dot product and $\|a\|$ is the norm of a vector. All the vectors and angles are displayed in Fig. 5. Note that for reducing complexity, the figure only presents the vector and angle in the left leg. The $\varphi_L$ represents the angle between the $B_v$ vector and $K_{V_{L,R}}$ vector, but for easy understanding, the *projected* $B_v$ is displayed in Fig. 5 for visualizing the $\varphi_L$ instead of $B_v$.

### C. Model Creation

The objective of the model proposed in this paper is to classify a static posture. To avoid any data fluctuation due to the inaccuracy of posture detection, three models are created and trained on one, five, and ten frames of data respectively.

For the model that is trained on only one frame, the dataset is simply the data from each individual frame from all 25 sessions combined.

For the models that are trained on five and ten frames, separate datasets are created. For the model with five frames, every five consecutive frames are stitched together as one input in the new dataset. The dataset for the ten frames model is prepared similarly.

Then, 80% of the data from each dataset are used as a training dataset, while the rest of the data are used as a testing dataset.

The best decision tree model is selected using the area under ROC curve scoring. Other parameters such as max depth and minimum sample splits are used to avoid overfitting issues.

## IV. Results and Discussion

### A. Data Observation

Despite recording the data with the correct posture, where both sides are symmetrical across the median plane of the body, a discrepancy is observed between the angles calculated from both sides, evident in Fig. 6. This error in the observed angle is the result of posture detection with MediaPipe. MediaPipe is trained and created such that the camera facing directly in front of humans yields the best accuracy [26]. As previously discussed, to avoid patients feeling uncomfortable in front of the camera, the camera is placed with an offset to the left side, thus creating the discrepancy of the observed angle. Noting the relationship between the camera angle and MediaPipe accuracy, this paper will not investigate this subject further.

### B. Models Performance

Despite the accuracy problem with coordinates obtained from MediaPipe, the result of the machine learning model is promising. Every model is tested two times: first tested with the testing dataset split from the training dataset, and second tested during deployment. During the deployment, a new camera angle that is still capable of detecting all three ArUco markers is set up. The testing result still returns a nearly perfect result, proving the whole setup valid. Table I

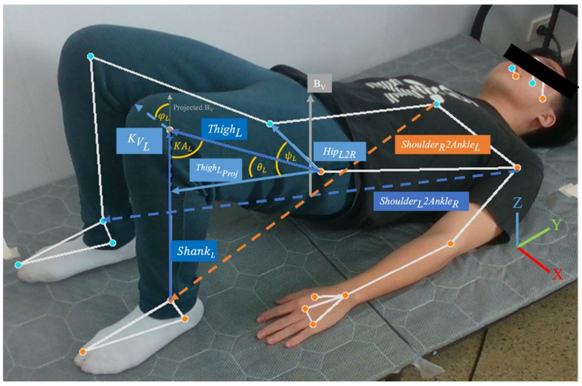


Fig. 5. All the vectors and angles in the data processing section for a machine learning model.

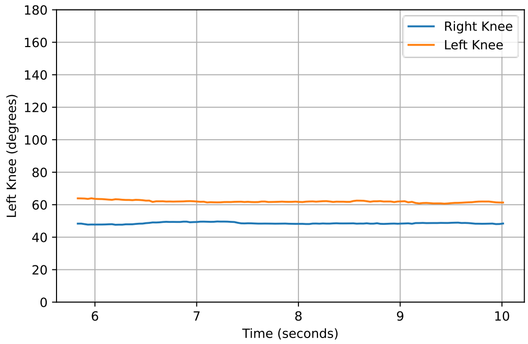


Fig. 6. Comparison between right and knee angles in the correct posture (expected the two angles to be the same).

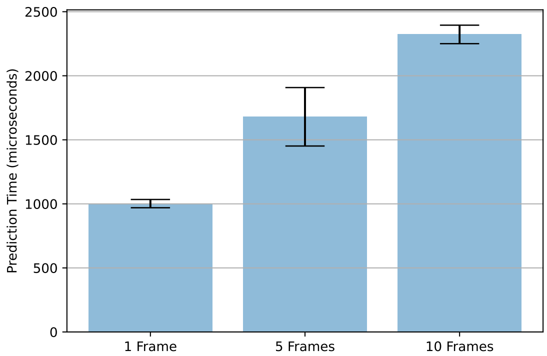


Fig. 7. Atomic prediction time of each model per instance.

shows the results of both tests. The models appear to be strong learners, capable of distinguishing the difference in posture easily. No further tweaking, such as boosting or random forest, of the models is needed.

TABLE I. Performance Evaluation of Models

| **Evaluation** | | **Models** | | |
|---|---|---|---|---|
| | | ***1 Frame*** | ***5 Frames*** | ***10 Frames*** |
| Testing Dataset | Sample size | 2561 | 2521 | 2471 |
| | Accuracy | 1.00 | 1.00 | 1.00 |
| Deployment | Sample size | 1097 | 1081 | 1061 |
| | Accuracy | 0.98 | 0.99 | 0.98 |
| Prediction Time | | 1 ms | 1.68 ms | 2.32 ms |

Since the three models have insignificant differences in terms of accuracy, the deciding factor of the model used is based on the prediction latency. Figure 7 shows the prediction time of each model. This time is collected with atomic predictions with 1,000 trials. As expected, the more frames used, the longer the prediction latency is.

## C. Model Deployment

The model trained on a single frame is chosen to be deployed due to its fast prediction time and similar prediction accuracy.

Figure 8 shows how the single frame model predicts. The most important feature in this model is the right knee angle. Figure 9 provides the mean value and standard deviation of right knee angle in both correct posture and wide knee posture collected over the 25 sessions.

Before the deployment of the model, the program could run at 20 fps with MediaPipe and ArUco. When the machine learning model is deployed, the program can predict at the rate of 10 fps.

The result of prediction is solely based on one frame, allowing the program to quickly pick up any changes in the posture. However, this one frame input also introduces some instability to the prediction, as it is possible for the program to quickly flicker between different categories. When this case does happen, a fusion layer can be added after the machine learning model. When tested with other subjects in real-time, no instability was observed.

## D. Discussion

However, data collected from 25 sessions with only one camera angle is still a very small sample size. Much larger datasets need to be acquired for the model to be generalized with patients of all sizes with cameras from all angles. This paper emphasizes highlighting the capability of the decision tree model to carry out such posture correction tasks.

The dataset used in this paper is collected in-house and is relatively small: only a total of 2,561 frames of data is collected. A previous study on real-time human activity recognition using similar techniques is trained on 16,520 frames, roughly 6.5 times larger [22]. Another study on real-time sign language recognition using similar techniques is trained 179,352 still images, roughly 70 times larger [23]. Both studies utilized different machine learning models: SVM, KNN, Decision Tree, Naïve Bayes, ANN, MLP. Despite the differences in applications, varying categories of classification, and varying dataset size, both papers achieve roughly similar accuracy: around 96 to near 99 percent accuracy for the best model. The accuracy obtained from this paper is 98 percent, which is in line with the results from these two papers. The two papers mentioned did not inform the average prediction time of the model.

While the result of this paper is promising, a potential issue may arise in the future when more categories of posture need to be detected. The current model is trained on angles manually computed from the coordinates of body landmarks. This requires the researcher to have a deep understanding of the correctness of the posture to identify the key features to be computed [27]. When scaling the model to accompany other different types of posture errors, it may be a time-intensive and complicated task to manually identify the key features to be computed.

Larger datasets and data from different camera angles are necessary to generalize the model's performance with patients of varying sizes and camera configurations. Future

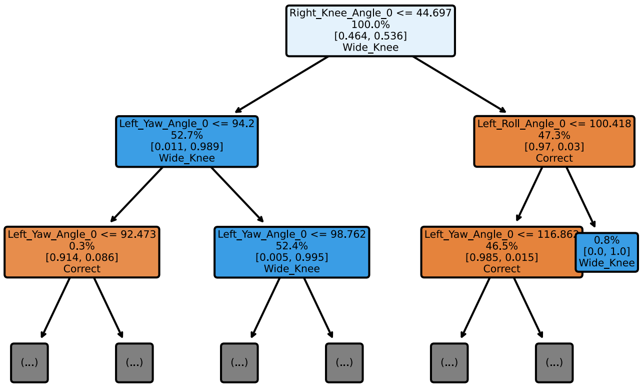


Fig. 8. Decision tree plot of the model trained with one frame.

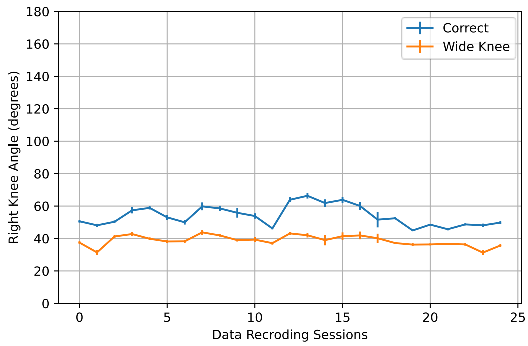


Fig. 9. Right knee angle in correct and knee posture.

work should focus on expanding the dataset, incorporating data from multiple camera angles, and comparing the performance of the decision tree model with neural network models and Support Vector Machine. Additionally, optimization techniques can be explored to improve prediction speed and further enhance the system's capabilities.

# V. Future Work

This paper serves as a review of the possibility of the use of machine learning models in the application of medial posture correction. There is a lot of work left unfinished to make the model to be more reliable and scalable.

First, the current model is trained in five subjects with similar physical statures. More data need to be collected to ensure the model can be generalized to a greater audience.

Second, the current dataset is collected from a fixed camera angle. When the camera angle changes, it is likely for the model to perform worse. Thus, the model shall also be trained on data collected from multiple camera angles.

The speed of the current prediction speed can still be improved and there are multiple approaches toward it.

*a)* The current model is built with the sci-kit learn library. The sci-kit learn library is optimized for bulk predictions. A lower prediction latency can be achieved by rewriting the prediction function.

*b)* Another method to optimize the time is optimizing the model with the targeted hardware.

Other techniques such as body parts normalization [23], humanoid models [25], and convolution neural networks can be used to ensure the better accuracy of posture estimation.

Lastly, due to the inherent insatiable nature of the camera, subject to environmental factors such as lighting and clothing, implementing data from sensors should make the models much more reliable [22], [28].

## VI. Conclusion

This paper presented a rehabilitation monitoring system that combines an RGB camera with a machine-learning model to predict the correctness of patients' bridge stand posture. The system architecture involved posture detection using the Intel RealSense D435 RGB-Depth camera and ArUco marker detection for mapping 3D array data. Human pose estimation using MediaPipe was utilized for detecting body elements, and visibility thresholds were set to ensure accurate data collection. The posture classification focused on detecting errors in the hip and knee areas during the static hold phase of the bridging exercise. Data processing involved calculating angles from the coordinates of joints, and decision tree models were trained on one, five, and ten frames of data. The models demonstrated near-perfect accuracy and were capable of distinguishing between correct and wide knee postures. The deployment of the model trained on a single frame was chosen due to its fast prediction time and similar accuracy. The program achieved a prediction rate of 10 frames per second, allowing for real-time detection of posture correctness.

Overall, this research highlights the potential of machine learning models in assisting physical therapists with real-time posture correction tasks. The decision tree model presented in this paper provides a foundation for further advancements in rehabilitation monitoring systems and the development of enhanced clinical protocols.

## Acknowledgment

This research project is partly supported by Tronormos Co., Ltd., Bangkok, Thailand.

## References

[1] M. Kelly-Hayes, "Influence of age and health behaviors on stroke risk: lessons from longitudinal studies," *J. Am. Geriatr. Soc.*, vol. 58, no. 2, pp. 325–328, 2010.

[2] R. Chaichaowarat, S. Nishimura, T. Nozaki, and H. I. Krebs, "Work in the time of Covid-19: actuators and sensors for rehabilitation robotics," *IEEJ J. Ind. Appl.*, vol. 11, no. 2, pp. 1–10, 2021.

[3] T. Tantagunninat, N. Wongkaewcharoen, K. Pornpipatsakul, R. Chuengpichanwanich, and R. Chaichaowarat, "Modulation of joint stiffness for controlling the cartesian stiffness of a 2-DOF planar robotic arm for rehabilitation," in *Proc. IEEE/ASME Int. Conf. Advanced Intelligent Mechatronics*, pp. 598–603, 2023.

[4] A. M. Abdullahi and R. Chaichaowarat, "Sensorless estimation of human joint torque for robust tracking control of lower-limb exoskeleton assistive gait rehabilitation," *J. Sens. Actuator Netw.*, vol. 12, no. 4, 53, 2023.

[5] S. Nishimura, R. Chaichaowarat, and H. I. Krebs, "Human-robot interaction: controller design and stability," in *Proc. IEEE RAS/EMBS Int. Conf. Biomedical Robotics and Biomechatronics*, pp. 1096–1101, 2020.

[6] R. Chaichaowarat, S. Nishimura, and H. I. Krebs, "Macro-mini linear actuator using electrorheological-fluid brake for impedance modulation in physical human–robot interaction," *IEEE Robot. Autom. Lett.*, vol. 7, no. 2, pp. 2945–2952, 2022.

[7] R. Chaichaowarat, S. Nishimura, and H. I. Krebs, "Design and modeling of a variable-stiffness spring mechanism for impedance modulation in physical human–robot interaction," in *Proc. IEEE Int. Conf. Robotics and Automation*, pp. 7052–7057, 2021.

[8] A. Javadi and R. Chaichaowarat, "Position and stiffness control of an antagonistic variable stiffness actuator with input delay using super-twisting sliding mode control," *Nonlinear Dyn.*, vol. 111, pp. 5359–5381, 2023.

[9] Z. Ullah, R. Chaichaowarat, and W. Wannasuphoprasit, "Variable damping actuator using an electromagnetic brake for impedance modulation in physical human–robot interaction," *Robotics*, vol. 12, no. 3, 80, 2023.

[10] G.B. Song and J.Y. Heo, "The effect of modified bridge exercise on balance ability of stroke patients," *J. Phys. Ther. Sci.*, vol. 27, pp. 3807–3810, 2015.

[11] T. Pankheaw, V. Hiengkaew, S. Bovonsunthonchai, and J. Tretriluxana, "Effect of progressive bridging exercise on weight-bearing during the extension phase of sit-to-stand, and on sit-to-stand ability in individuals with stroke: A randomised controlled trial," *Clin. Rehabil.*, vol. 36, no. 11, pp. 1463–1475, 2022.

[12] R. Chaichaowarat, S. Prakthong, and S. Thitipankul, "Transformable wheelchair–exoskeleton hybrid robot for assisting human locomotion," *Robotics*, vol. 12, no. 1, pp. 16, 2023.

[13] Y. Ryu, S. Ju, G. Park, and H.J. Jang, "Effects of bridging exercise methods on the muscular activity of the neck, trunk, and lower limbs," *J. Phys. Ther. Sci.*, vol. 23, pp. 867–869, 2011.

[14] I.M. Kit Ho, L.P. Cindy Ng, K. Leonardo Lee, and T.C. Jim Luk, "Effects of knee flexion angles in supine bridge exercise on trunk and pelvic muscle activity," *Res. Sports Med.*, vol. 28, no. 4, pp. 484–497, 2020.

[15] J. Kim and M. Park, "Changes in the activity of trunk and hip externsor muscles during bridge exercises with variations in unilateral knee joint angle," *J. Phys. Ther. Sci.*, vol. 28, pp. 2537–2540, 2016.

[16] L. Ding et al., "Effects of camera-based mirror visual feedback therapy for patients who had a stroke and the neural mechanisms involved: protocol of a multicentre randomised control study," *BMJ Open*, vol. 9, e022828, 2019.

[17] L. Evett et al., "Dual camera motion capture for serious games in stroke rehabilitation," in *Proc. IEEE Int. Conf. Serious Games and Applications for Health*, pp. 1–4, 2011.

[18] K. Pornpipatsakul and N. Ajavakom, "Estimation of knee assistive moment in a gait cycle using knee angle and knee angular velocity through machine learning and artificial stiffness control strategy (MLASCS)," *Robotics*, vol. 12, no. 2, pp. 44, 2023.

[19] B. Chen et al., "Computer vision and machine learning-based gait pattern recognition for flat fall prediction," *Sensors*, vol. 22, no. 20, pp. 7960, 2022.

[20] M. Zhu, Z. Zhang, J. P. Hirdes, and P. Stolee, "Using machine learning algorithms to guide rehabilitation planning for home care clients," *BMC Medical Inform. Decis. Mak.*, vol. 7, pp. 1–13, 2007.

[21] K. Pornpipatsakul, A. Chenviteesook, and R. Chaichaowarat, "Ultrasound probe movement analysis using depth camera with compact handle design for probe contact force measurement," in *Proc. IEEE/EMBS Annu. Int. Conf.*, 2023.

[22] S. Ghazal, U. S. Khan, M. Mubasher Saleem, N. Rashid, and J. Iqbal, "Human activity recognition using 2D skeleton data and supervised machine learning," *IET Image Process.*, vol. 13, no. 13, pp. 2572–2578, 2019.

[23] A. Halder and A. Tayade, "Real-time vernacular sign language recognition using mediapipe and machine learning," *Int. J. Res. Publ. Rev.*, vol. 2582, pp. 7421, 2021.

[24] R. Munoz-Salinas, "Aruco: a minimal library for augmented reality applications based on openCV," Universidad de Córdoba, vol. 386, 2012.

[25] J.W. Kim, J.Y. Choi, E.J. Ha, and J.H. Choi, "Human pose estimation using MediaPipe pose and optimization method based on a humanoid model," *Appl. Sci.*, vol. 13, no. 4, pp. 2700, 2023.

[26] A. Latreche, R. Kelaiaia, A. Chemori, and A. Kerboua, "Reliability and validity analysis of MediaPipe-based measurement system for some human rehabilitation motions," *Measurement*, vol. 214, pp. 112826, 2023.

[27] A. B. Sargano, P. Angelov, and Z. Habib, "A comprehensive review on handcrafted and learning-based action representation approaches for human activity recognition," *Appl. Sci.*, vol. 7, no. 1, pp. 110, 2017.

[28] Y. Tang, Q. Teng, L. Zhang, F. Min, and J. He, "Layer-wise training convolutional neural networks with smaller filters for human activity recognition using wearable sensors," *IEEE Sens. J.*, vol. 21, no. 1, pp. 581–592, 2020.