Intelligent Methods in Engineering Sciences 2(2): 052-057, 2023

INTELLIGENT METHODS
IN ENGINEERING SCIENCES

https://www.imiens.org

International
Open Access

June, 2023

e-ISSN 2979-9236

Research Article https://doi.org/10.58190/imiens.2023.13

Simple AR Method for Rehabilitation Support System Based on 3D Pose Estimation

Kazumoto Tanaka [a, *]

[a] *Department of Informatics, Faculty of Engineering, Kindai University, Higashi-hiroshima, Japan*

ARTICLE INFO

Article history:
Received 13 February 2023
Accepted 25 June 2023

Keywords:
3D pose
Augmented reality
MediaPipe
Rehabilitation
Support system

ABSTRACT

Studies have been conducted on the application of Augmented Reality to support rehabilitation of motor function recovery. The goal of these studies is to facilitate functional recovery training through patient interaction with virtual objects generated by AR. Many of them use special devices such as depth sensors to superimpose virtual objects at appropriate positions in images, but a simple method that does not require such a device is desired. In order to realize superimposition using only a personal computer (PC) with a camera, this study utilizes a deep neural network that estimates the 3-dimensional (3D) coordinates of keypoints, such as human joints, from camera images. Specifically, a coordinate transformation matrix for superimposition is calculated from the 3D coordinates of keypoints. In order to clarify the effectiveness of this method, we conducted an experiment to evaluate the superimposition accuracy. The results show that the accuracy was highest in the space near the keypoints that had been used to compute the coordinate transformation matrix, and the accuracy was even higher when the number of keypoints was small. This indicates that this method is more suitable for localized training such as hand rehabilitation than for whole-body training. Since this method can be used only with a PC with a camera, it is expected to be widely used for rehabilitation support.

1. INTRODUCTION

Augmented Reality (AR) is a technology that superimposes virtual objects on images of the real world, enabling users to have perceptually rich experiences. Various applications of AR have been studied and put to practical use, and rehabilitation support system for motor function recovery are one of them. The rehabilitation support system enables patients to interact with virtual objects and enjoy games while training for functional recovery [1]. The ultimate goal of this study is to realize a simple AR-based rehabilitation support system using only a personal computer (PC) with a video camera.

In order to superimpose a virtual object at an appropriate position in the image for the interaction, it is important to understand the positional relationship between the image coordinate system of the camera capturing the image and the three-dimensional (3D) coordinate system of the real world, i.e., to compute the transformation matrix from the 3D coordinate system to the image coordinate system. Many previous studies of AR-based rehabilitation support systems have employed stereo cameras or depth sensors to obtain the transformation matrix [1].

Recently, a simple monocular 3D pose estimation method has been developed by deep neural networks (DNN), and there are growing expectations for its application to AR-based rehabilitation [2]. The DNN-based 3D pose estimation method estimates the 3D coordinates of keypoints on the human body (e.g. human joints) only by using a single camera. The method includes direct regression of the keypoint coordinates [3], estimation with 3D heat map [4], [5], and estimating 3D pose from 2D pose (Lifting based 3D pose estimation) [6]-[13]. There are two types of 3D coordinate systems used in the estimation: a local coordinate system (root-relative coordinate system) [3]-[7] whose origin is a specific part of the subject (e.g., the center of the waist), and a camera coordinate system [8]-[13]. In the latter case, it is also necessary to estimate the distance from the camera coordinate system to the human, which requires information on the intrinsic parameters of the camera.

To realize a simple AR, this study utilizes a DNN that estimates the 3D pose of a subject based on the root-relative coordinate system to calculate the matrix that transforms the coordinate system to the image coordinate system in real time (see Figure 1). This study uses MediaPipe Pose [7] , which is capable of high-speed processing, as such a DNN. However, owing to errors in the output of the DNN, a certain amount of error can be

*** Corresponding Author:** kazumoto@hiro.kindai.ac.jp

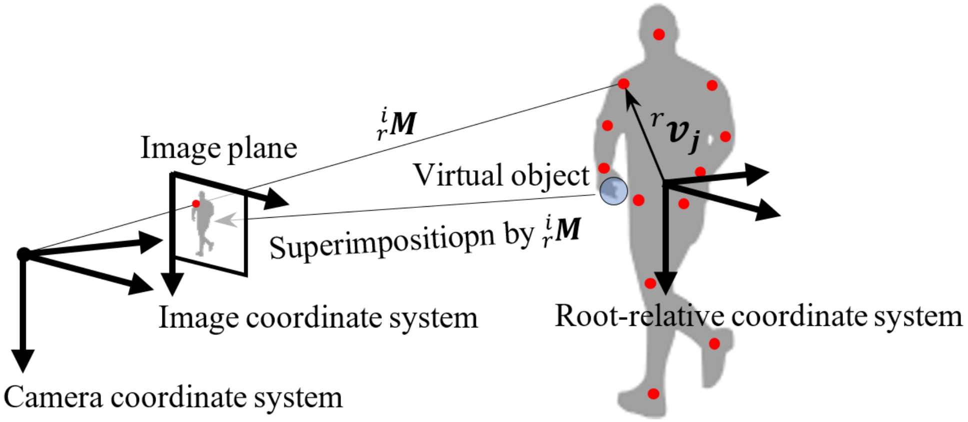
Image plane
${}^{i}_{r}\boldsymbol{M}$
${}^{r}\boldsymbol{v}_{\boldsymbol{j}}$
Virtual object
Superimpositiopn by ${}^{i}_{r}\boldsymbol{M}$
Image coordinate system
Root-relative coordinate system
Camera coordinate system

Figure 1. The coordinate transformation matrix (${}^{i}_{r}\boldsymbol{M}$ in the figure) and the simple AR. The transformation matrix is calculated by using the 3D keypoint coordinates estimated by DNN and their image coordinates and is used to superimpose a virtual object placed virtually at an arbitrary location in 3D space onto the image.

expected in superimposition using the transformation matrix calculated from the output. The purpose of this paper is to clarify the feasibility of using 3D pose estimation network for AR-based rehabilitation support system in terms of superimposition accuracy.

There are three main methods for computing the coordinate transformation matrix that have been used in previous studies on AR-based rehabilitation support systems: color markers and color cameras [14], stereo cameras [15], and depth sensors [16], [17]. [14] uses position information from a color marker attached to the hand, but it is not 3D. [15] calculates the transformation matrix by calibrating stereo cameras. [16] and [17] use depth sensors to measure the positions of keypoints on the fingers and the whole body, respectively, to obtain the transformation matrix. Compared to them, our approach is novel in that the transformation matrix is obtained by a DNN-based 3D pose estimation network. Furthermore, our method uses only a PC with a camera and requires no calibration or other preparations, making it a simple AR that anyone can use.

Several studies have applied 3D pose estimation networks to measurements required for rehabilitation training, such as joint angle measurements. [18] evaluated the applicability of a 3D pose estimation network from experiments on the measurement of the active range of motion in the shoulder. However, from the literature review, no study has reported the applicability of 3D pose estimation networks to the calculation of the coordinate transformation matrix required for AR. The main contributions of this paper are as follows:

1) The method for calculating the coordinate transformation matrix using a 3D pose estimation network is proposed.

2) The relationship between the accuracy of the transformation matrix and the keypoint coordinates obtained from the 3D pose estimation network is experimentally clarified.

3) It is shown that practical accuracy of AR for rehabilitation training can be obtained by selecting appropriate keypoints according to the rehabilitation target area.

The rest of the paper is structured as follows. Section 2 describes the proposed method for calculating the coordinate transformation matrix and a simple AR method using the matrix. The experiment for evaluating the proposed method and the results are provided in Section 3. In Section 4, the possibility of the proposed method for AR-based rehabilitation is discussed. Section 5 concludes the paper and provides future work.

2. Method

The method to compute the coordinate transformation matrix from the keypoint coordinates output by a 3D pose estimation network is proposed in Subsection 2.1. This method uses MediaPipe as the 3D pose estimation network. Subsection 2.2 then describes an AR method that uses the computed matrix.

2.1. Transformation Matrix Calculation

The equation for perspective transformation of 3D keypoints expressed in the root-relative coordinate system to images is as follows.

$$s\,{}^{i}\boldsymbol{v}_{j} = {}^{i}_{r}\boldsymbol{M} \cdot {}^{r}\boldsymbol{v}_{j} \tag{1}$$

, where ${}^{i}_{r}\boldsymbol{M}$, ${}^{i}\boldsymbol{v}_{j}$, ${}^{r}\boldsymbol{v}_{j}$, and s are the coordinate transformation matrix, the image coordinates, 3D keypoint coordinates, and a non-zero scalar, respectively. Each of these coordinates is expressed in terms of homogeneous coordinates. Rewrite Equation (1) using vector and matrix elements:

$$s\begin{pmatrix} {}^{i}\boldsymbol{v}_j(x) \\ {}^{i}\boldsymbol{v}_j(y) \\ 1 \end{pmatrix} = \begin{pmatrix} c_{11} & c_{12} & c_{13} & c_{14} \\ c_{21} & c_{22} & c_{23} & c_{24} \\ c_{31} & c_{32} & c_{33} & 1 \end{pmatrix} \begin{pmatrix} {}^{r}\boldsymbol{v}_j(x) \\ {}^{r}\boldsymbol{v}_j(y) \\ {}^{r}\boldsymbol{v}_j(z) \\ 1 \end{pmatrix} \tag{2}$$

, where the element c_{33} of ${}^{i}_{r}\boldsymbol{M}$ is set to 1 since the equation is up to scale. Equation (3) is obtained by eliminating s in Equation (2).

$$\begin{pmatrix} {}^{i}\boldsymbol{v}_j(x) \\ {}^{i}\boldsymbol{v}_j(y) \end{pmatrix} = \begin{pmatrix} {}^{r}\boldsymbol{v}_j(x) & {}^{r}\boldsymbol{v}_j(y) & z & 1 & 0 & 0 & 0 & 0 & -{}^{r}\boldsymbol{v}_j(x){}^{i}\boldsymbol{v}_j(x) & -{}^{r}\boldsymbol{v}_j(y){}^{i}\boldsymbol{v}_j(x) & -{}^{r}\boldsymbol{v}_j(z){}^{i}\boldsymbol{v}_j(x) \\ 0 & 0 & 0 & 0 & {}^{r}\boldsymbol{v}_j(x) & {}^{r}\boldsymbol{v}_j(y) & {}^{r}\boldsymbol{v}_j(z) & 1 & -{}^{r}\boldsymbol{v}_j(x){}^{i}\boldsymbol{v}_j(y) & -{}^{r}\boldsymbol{v}_j(y){}^{i}\boldsymbol{v}_j(y) & -{}^{r}\boldsymbol{v}_j(z){}^{i}\boldsymbol{v}_j(y) \end{pmatrix} \begin{pmatrix} c_{11} \\ c_{12} \\ c_{13} \\ c_{14} \\ c_{21} \\ c_{22} \\ c_{23} \\ c_{24} \\ c_{31} \\ c_{32} \\ c_{33} \end{pmatrix} \tag{3}$$

The matrix in Equation (3) is a 2×11 matrix using the 3D and image coordinates of one keypoint, so if there are N keypoints, the matrix is $2\mathrm{N} \times 11$. Therefore, for $6 \leq \mathrm{N}$, the matrix becomes a tall matrix, and its Generalized Inverse Matrix is used to obtain the elements c_{11}~c_{33} of the coordinate transformation matrix. Since DNNs that estimate 3D keypoints generally estimate 13 or more keypoints [3]-[13], the transformation matrix can be obtained as long as 6 keypoints are visible, even if keypoint occlusion occurs.

2.2. Simple AR

Once the coordinate transformation matrix is obtained, the coordinates of the position where a virtual object is superimposed on the image are calculated using Equation (4).

$$s\,{}^{i}\boldsymbol{v}_o = {}^{i}_{r}\boldsymbol{M} \cdot {}^{r}\boldsymbol{v}_o \tag{4}$$

, where ${}^{i}\boldsymbol{v}_o$ and ${}^{r}\boldsymbol{v}_o$ denote homogeneous image coordinates and homogeneous 3D coordinates of a virtual object. The computation of the transformation matrix and superimpositions is done for each image frame.

3. Experiments

Experiments were conducted to evaluate the effectiveness of the simple AR based on 3D pose estimation in terms of superimposition accuracy. In general, the accuracy of superimposition is evaluated by the re-projection error, which is the error between the image coordinates transformed from the 3D coordinates of a point in the real world by the transformation matrix and the actual image coordinates of the point. On the other hand, the simple AR method uses the 3D coordinate system recognized by the 3D pose estimation network (MediaPipe Pose in this study), so the 3D coordinates expressed in this coordinate system are used to evaluate the superimposition accuracy. There is no point other than keypoints at which such 3D coordinates can be obtained. Furthermore, since the human keypoints are not visible and therefore do not appear in the image, the image coordinates of the keypoints estimated by the 3D pose estimation network must be used. Therefore, the difference between the image coordinates transformed by the transformation matrix from the 3D coordinates of a keypoint estimated by the 3D pose estimation network and the image coordinates of the keypoint estimated by the same network is evaluated.

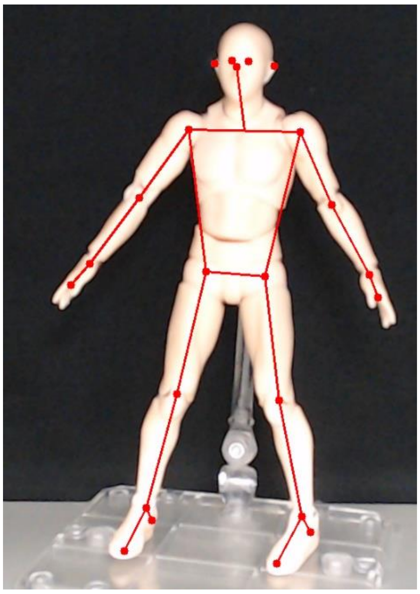

Figure 2. Twenty-three keypoints used in the experiment

In the experiment, the following 23 keypoints (see also Figure 2) were used among the keypoints estimated by MediaPipe Pose:

Nose, Eye (right, left), Ear (right, left), Shoulder (right, left), Elbow (right, left), Wrist (right, left), Index-finger (right, left), Waist (right, left), Knee (right, left), Ankle (right, left), Heel (right, left), Toe (right, left).

The input videos to the MediaPipe Pose were taken from TNT15 [19], a motion capture dataset that is freely available for research purposes. Each video was taken by 8 RGB-cameras from 8 different directions of a subject's activity in a laboratory. There are 4 subjects and 5 activities consisting of walking, running on the spot, rotating arms, jumping and skiing exercises, and punching (i.e., 20 different videos). The duration of each activity is about 10 seconds, and the frame size of the videos is 800 x 600. In our experiment, in order to use the left and right keypoints of the human body equally and to increase the number of data, we created videos with the video frames inverted left and right, and used a total of 40 types of videos (original and inverted).

The transformation matrix was calculated by expanding the matrix in Equation (3) to $2N \times 11$ matrix by using the estimated 3D coordinates of N keypoints. Here, four different experiments (Expt 1-4) were performed, depending on which keypoints were selected for the computation of the transformation matrix. The keypoints selected for each experiment are listed in Table 1. In each experiment, the re-projection error of the keypoints shown in Table 2 was evaluated over the five motions. An example of the re-projection in Expt 1 is shown in Figure 3, and the results of each experiment are shown in Figure 4, 5, 6, and 7, respectively.

The frames per second (fps) of the pose estimation was almost 15 on a personal computer (Windows OS, Intel Xeon Silver 4214 CPU @ 2.20GHz, Nvidia GeForce RTX 2080 Ti GPU), and there was no obvious decrease in computation time, even after the addition of the transformation matrix calculation and the re-projection process.

Table 1. Keypoints used for the transformation matrix calculation in the experiment. Keypoints other than Nose are both left and right

Expt 1	Expt 2	Expt 3	Expt 4
Nose Eye Ear Shoulder Elbow Wrist Index-finger Waist Knee Ankle Heel Toe	Nose Elbow Wrist Waist Knee Ankle	Shoulder Elbow Wrist Index-finger	Waist Knee Ankle Toe

Table 2. Keypoints for which re-projection errors were calculated in the experiment

Expt 1	Expt 2	Expt 3	Expt 4
Nose Eye Ear Shoulder Elbow Wrist Index-finger Waist Knee Ankle Heel Toe	Nose Eye Ear Shoulder Elbow Wrist Index-finger Waist Knee Ankle Heel Toe	Shoulder Elbow Wrist Index-finger	Waist Knee Ankle Toe

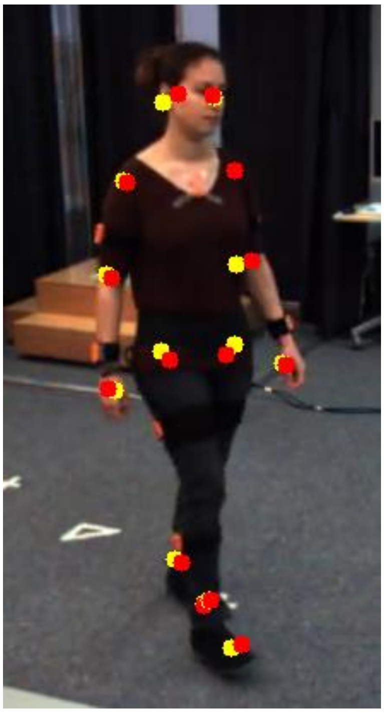

Figure 3. Example of re-projection. Red filled circles indicate the 2D positions of the keypoints estimated by MediaPipe Pose. The yellow filled circles indicate the re-projected positions from the 3D coordinates of the keypoints estimated by the network.

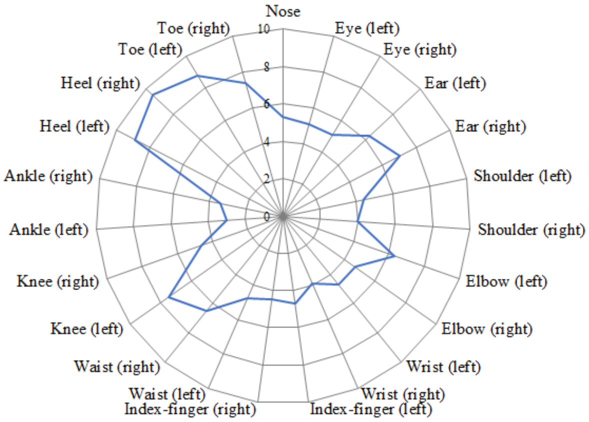

Figure 4. Re-projection errors in Expt 1. Numerical values are in pixels.

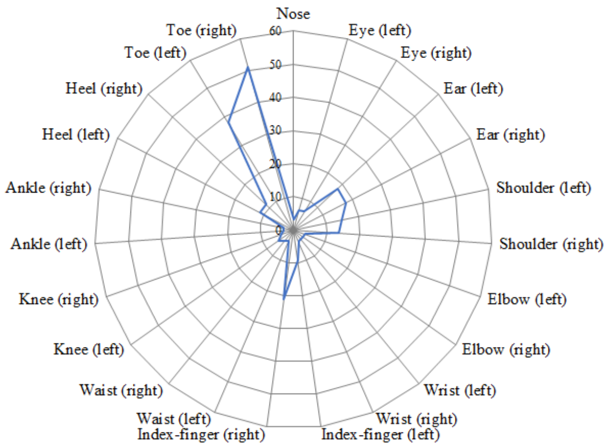

Figure 5. Re-projection errors in Expt 2. Numerical values are in pixels. Note that the maximum value of the scale is 60.

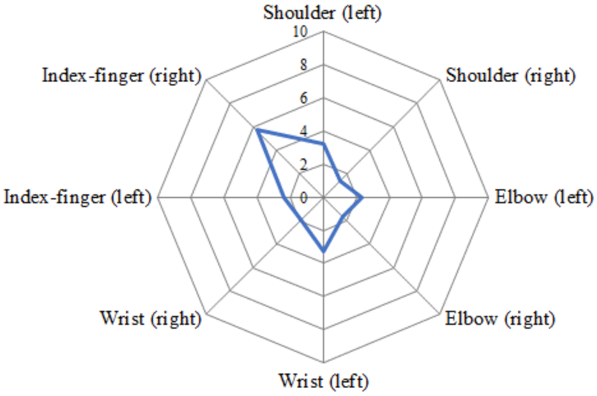

Figure 6. Re-projection errors in Expt 3. Numerical values are in pixels.

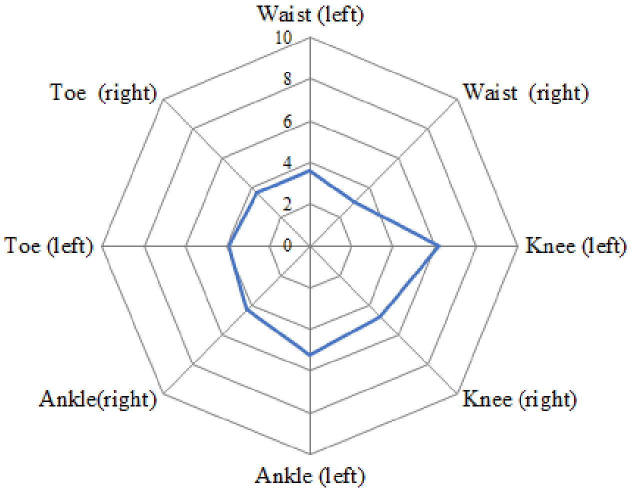

Figure 7. Re-projection errors in Expt 4. Numerical values are in pixels.

4. Discussion

Figure 4 shows that Heel (right) has the largest re-projection error of 9.5 pixels when the transformation matrix is calculated using the 3D coordinate estimation results for the all keypoints. Estimating the amount of error in real space, since the image is 800 pixels wide and covers an area of 2,000 mm (TNT15_documentation on [19]), the amount of error is $9.5 / 800 \times 2000 = 23.8$ mm. Since the median (50th percentile) of hand breadth of American women and men is 93.0 mm and 102.0 mm, respectively [20], this error amount is about 1 / 3.9 and 1 / 4.3 of the hand breadth, respectively. Since the hand breadth is the width of four fingers except thumb, although this is a rough estimate, the magnitude of the error is approximately equal to the width of one finger.

Figure 5 shows that the overall error increases when the number of keypoints used to calculate the transformation matrix is reduced. In particular, the errors for the keypoints not used in the calculation (Ear, Shoulder, Index-finger, Heel and Toe) are large. However, Eye was also not used in the calculation, but its error was not much worse. It may be calculated in such a way that the accuracy of the transformation matrix is increased in the space near the keypoints used for the calculation. Accordingly, since Eye is located close to Nose used in the calculation, its accuracy was probably preserved.

However, Figure 6 and 7 do not show an increase in error compared to the case where all keypoints are used (Figure 4), but rather a decrease for Shoulder, Elbow, Wrist, Waist, and Toe, even though the number of keypoints used to calculate the transformation matrix has been reduced, respectively. Calculating the transformation matrix using the Generalized Inverse Matrix of the expanded matrix ($2N \times 11$) in Equation (3) is equivalent to finding the least-squares solution of the equation. Therefore, if the keypoints used in this calculation are distributed in a small space, an optimized matrix can be obtained within that range, and the accuracy is not reduced but rather increased within that range.

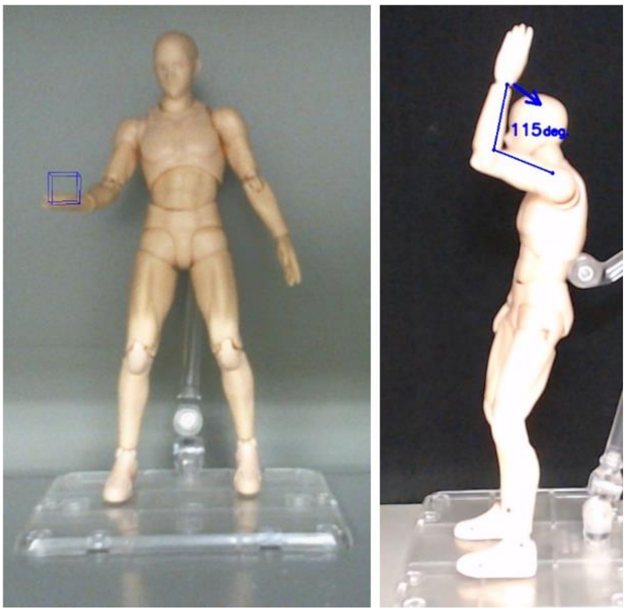

Figure 8. Example of simple AR assuming rehabilitation support. Left: Virtual object and right hand interaction. Right: Angle indication in left elbow bending exercise.

If the position of the superimposed virtual object is off by as little as one finger, there should be no problem in interaction with virtual object and its display. Based on the above considerations, the experiments revealed that, as an

AR system for rehabilitation support, it is better to design the system that superimposes virtual objects (see Figure 8) using the transformation matrix obtained from keypoints ($6 \leq \mathrm{N}$) near the body part of the target of functional recovery training.

5. Conclusion

In order to construct a simple AR system for rehabilitation support, we proposed a method to calculate the coordinate transformation matrix required for superimposition of virtual objects using MediaPipe Pose, which estimates the 3D coordinates of keypoints, and experimentally verified the superimposition accuracy using this method. The results showed that the accuracy was highest in the space near the keypoints that had been used to compute the transformation matrix, and the accuracy was even higher when the number of keypoints was small. Consequently, this method is more suitable for localized training such as hand rehabilitation than for whole-body training. The fps for the whole process was almost 15. Since rehabilitation training is generally performed with slow movement, it can be said that the processing speed is enough. In the future, we will develop an AR system for localized training and conduct practical verification.

On the other hand, the coordinates estimated by MediaPipe Pose are represented by a root-relative coordinate system with the waist as the origin, so it is not suitable for training that evaluates waist motion. MediaPipe Pose was employed for this study because it is a lightweight DNN with high processing speed, but another DNN that estimates the world coordinates of keypoints will be investigated to expand the scope of the application.

Acknowledgments

This work was supported by JSPS KAKENHI Grant Number 21K12166.

Conflicts of Interest

The author declares no conflict of interest.

References

[1] Y. Gu *et al.*, “A Review of Hand Function Rehabilitation Systems Based on Hand Motion Recognition Devices and Artificial Intelligence,” *Brain Science*, vol. 12, 1079, 2022.

[2] T. Hellsten, J. Karlsson, M. Shamsuzzaman, and G. Pulkkis, “The Potential of Computer Vision-based Marker-less Human Motion Analysis for Rehabilitation,” *Rehabilitation Process and Outcome*, vol. 10, pp. 1–12, 2021.

[3] B. Tekin, I. Katircioglu, M. Salzmann, V. Lepetit, and P. Fu, “Structured Prediction of 3D Human Pose with Deep Neural Networks, *Proc. BMVC2016*,” pp. 130.1-130.11, 2016.

[4] H. Zhou, C. Hong, Y. Han, P. Huang, and Y. Zhuang, “MH Pose: 3D Human Pose Estimation Based on High-quality Heatmap,” *Proc. BIG-DATA2021*, pp. 3215-3222, 2021.

[5] S. Parajuli and M. K. Guragai, “Human Pose Estimation in 3D Using Heatmaps,” *Proc. AISP2022*, pp. 1-4., 2022

[6] J. Martinez, R. Hossain, J. Romero, and J. J. Little, “A Simple yet Effective Baseline for 3D Human Pose Estimation,” *Proc. ICCV2017*, pp. 2640-2649, 2017.

[7] “MediaPipePose,” [Online]. Available: https://google.github.io/mediapipe/solutions/pose. [Accessed: 10-Jan-2023].

[8] D. Tome, C. Russell, and L. Agapito, “Lifting from the Deep: Convolutional 3D Pose Estimation from a Single Image,” *Proc. CVPR2017*, pp. 2500-2509, 2017.

[9] W. Li, H. Liu, H. Tang, P. Wang, and L. V. Gool, “MHFormer: Multi-hypothesis Transformer for 3D Human Pose Estimation,” *Proc. CVPR2022*, pp. 13147-13156, 2022.

[10] G. Moon, J. Y. Chang, and K. M Lee, “Camera Distance-aware Top-down Approach for 3D Multi-person Pose Estimation from a Single RGB Image,” *Proc. ICCV2019*, pp. 10133-10142, 2019.

[11] Y. Cheng, B. wang, B. Y., and R. T. Tan, “Graph and Temporal Convolutional Networks for 3D Multi-person Pose Estimation in Monocular Videos,” *Proc. AAAI-21*, pp. 1157-1165, 2021.

[12] L. Jin, C. Xu, X. Wang, Y. Xiao, Y. Guo, X. Nie, and J. Zhao, “Single-stage is Enough: Multi-person Absolute 3D Pose Estimation,” *Proc. CVPR2022*, New Orleans, USA, 13076-13085, 2022.

[13] Y. Zhan, F. Li, R. Weng, and W. Choi, “Ray3D: Ray-based 3D Human Pose Estimation for Monocular Absolute 3D Localization,” *Proc. CVPR2022*, pp. 13116-13125, 2022.

[14] H. M. Hondori, M. Khademi, L. Dodakian, S. C. Cramer, and C. V. Lopes, “A Spatial Augmented Reality Rehab System for Post-stroke Hand Rehabilitation,” Medicine Meets Virtual Reality, vol. 20, pp. 279-285, 2013.

[15] D. Avola, L. Cinque, G. L. Foresti, and M. R. Marini, "An Interactive and Low-cost Full Body Rehabilitation Framework Based on 3D Immersive Serious Games," *Journal of Biomedical Informatics*, vol. 89, pp. 81–100, 2019.

[16] Z.-R. Wang, P. Wang, L. Xing, L.-P. Mei, J. Zhao, and T. Zhang, " Leap Motion-based Virtual Reality Training for Improving Motor Functional Recovery of Upper Limbs and Neural Reorganization in Subacute Stroke Patients," *Neural Regeneration Research*, vol. 12(11), pp. 1823- 1831, 2017.

[17] Y. Tokuyama, R.-P.-C. J. Rajapakse, S. Yamabe, K. Konno, and Y.-P. Hung, "A Kinect-Based Augmented Reality Game for Lower Limb Exercise," *Proc. Cyberworlds2019*, pp. 399-402, 2019.

[18] 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, 112826, 2023.

[19] “Multimodal Motion Capture Dataset (TNT15) ,” [Online]. Available: https://www.tnt.uni-hannover.de/project/TNT15/. [Accessed: 10-Jan-2023].

[20] C. C. Gordon *et al.*, “Anthropometric Survey of U.S. Army Personnel: Methods and Summary Statistics,” U.S. Army Natick Soldier Research, 2014.