Three-Dimensional Foot Contact Position on a Smart Fitness Trampoline with a Upward Looking Wide View Imaging System

Sekyung Park[1] Junkyu Park[1] Jongsik Ahn[2] Boram Cho[1] Suwoong Lee[1,*] Minyoung Kim[2,*]

[1] *Advanced Mechatronics Group, Korea Institute of Industrial Technology, Cheonan, Republic of Korea*
lee@kitecth.re.kr
[2] *Department of Electrical and Electronic Engineering, the Kyungpook National University, Daegu, Republic of Korea*
minyoung.kim2@gmail.com

***Abstract*—This study estimates the position and depth of the user's foot contacts of playing on a trampoline by acquiring the image of the footprint shadow and processing it to enable quantitative analysis of jumping (as a fitness exercise) or to link it to computer games. First, a hardware prototype of a jumping fitness monitoring system is proposed, consisting of a trampoline, an upward-looking wide-angle fish-eye camera module, and a signal processing embedded board. Second, image-based three-dimensional foot contact position estimation (I3D-FCE) algorithms are proposed, comprising the foot contact position estimation algorithm in the transverse plane and the foot contact depth estimation algorithm in the vertical direction. The foot contact positions were estimated using the binary image of the foot shadow obtained by extracting the binary masks of the trampoline and foot. The foot contact depth was estimated using the similarity ratio of the footprint shadow sizes. To verify the usefulness of the proposed system and the accuracy of the proposed contact position estimation algorithms, a series of experiments were conducted using a robotic manipulator. The experimental results show that the X and Y coordinates of the foot contact position and the Z coordinate of the foot contact depth were successfully estimated within an average error of 3.40 (mm), 3.26 (mm), and 4.49 (mm), respectively.**

***Keywords*—Image Processing, Embedded board, Smart Fitness**

I. Introduction

Trampolines are often used not only for recreational activities but also for exercise and rehabilitation assistance systems [1] [2] [3] [4]. Among them, jumping fitness is a one-person trampoline exercise program performed with music [5] [6]. In this program, the person exercising follows the movements of a professional instructor to simultaneously strengthen the person's aerobic condition, muscle strength, and balance. However, a trampoline equipped with a motion recognition system that estimates the amount of exercise can be enjoyed by individuals at home without the help of a professional instructor. A home-based trampoline system can quantify exercise intensity data based on motion concordance between instructor and user, providing users with feedback on exercise effectiveness [7]. This systems are in line with online health care for COVID-19 prevention, and the demand for online health care technologies is growing [8]. In order to build user data that can be analyzed accordingly, user motion information is required.

Various studies have reported on motion recognition of people exercising on a trampoline. R. Kajastila and PW Connolly conducted a study to recognize and analyze the whole-body motion of a user with a camera [9] [10], and M. Tiator carried out a study to link body motion recognized with a camera and virtual reality (VR) system [11]. In these studies, the whole-body motion was recognized with a camera to determine the degree of motion correspondence, concentrating on the linkage with VR data. To quantify the exercise effects, T. Helten and D. Eager installed a three-axis acceleration sensor at the waist of the user to analyze the intensity and cycle of exercise [12] [13]. A study on the classification of movements, such as walking and running, using a distance sensor installed under a trampoline was reported by H. Mori [7]. As such, existing studies have recognized the motion of a person exercising on a trampoline through a variety of sensor equipment to roughly classify motion. However, providing more precise data on exercise intensity and combining high-level game or entertainment content in the VR environment requires a solution for a more accurate position of motion analysis is needed.

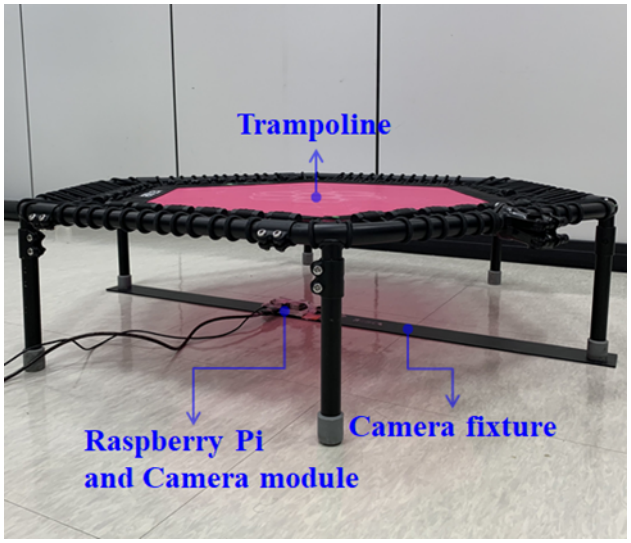
In this paper, we propose an algorithm for estimating the foot contact position in the transverse plane and the foot contact depth in the vertical direction by processing the image of the foot shadow contacting the trampoline. We present a hardware prototype for a jumping fitness monitoring system.

II. Image-based Three-dimensional Foot Contact Position Estimation

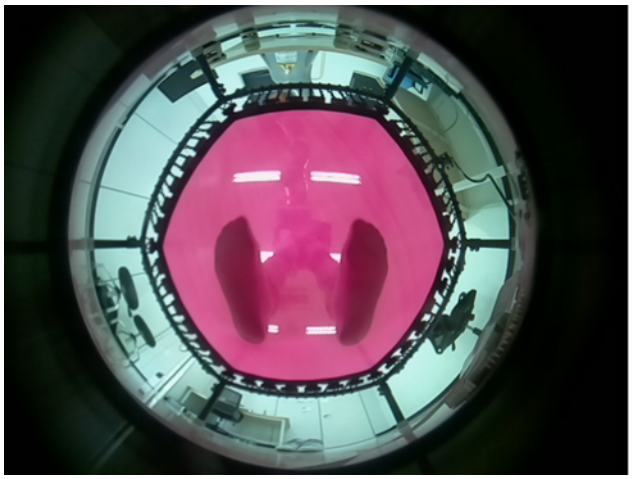
A. System Overview

Fig. 1 shows a prototype of a jumping fitness system employed with the image-based three-dimensional foot contact position estimation (I3D-FCE) algorithm. This system includes a jumping fitness trampoline (J6H130 FLEXI, W × D × H = 1360 × 1360 × 285 (mm), maximum load of 130 (kgf)) from

Jumping Inc., an embedded board (Raspberry Pi 4 B, with a Broadcom BCM2711 SoC 1.5 GHz Quad-Core 64-bit CPU, and 4 GB of RAM) for real-time image processing, and a wide-angle fish eye camera module (220° wide angle fish eye lenses for Raspberry Pi Camera-LS-32220, 5 million pixels, 640 × 480 pixels, 60/90 fps). A single-camera module was employed to simplify the distance measurement system and accelerate the processing. The camera module was attached to the bottom of the trampoline to capture the foot shadow. The extracted shadows were used to estimate the contact position and the depth of the foot.



(a)



(b)

Fig. 1. (a) Trampoline system equipped with a sensor at the center of the camera fixture and (b) trampoline image captured with a wide-angle lens.

Fig. 2 shows a flowchart of the I3D-FCE algorithm. Assuming X, Y, and Z are the actual contact positions on the trampoline, $\hat{X}$, $\hat{Y}$, and $\hat{Z}$ represent the contact positions estimated by the I3D-FCE algorithm. P_x and P_y denote the estimated positions in the pixel coordinate system, while $x^{'}$ and $y^{'}$ represent the positions converted from P_x and P_y, respectively, in the world coordinate system. z is the estimation depth value computed from the similarity ratio formula. Fig. 2(a) illustrates the process of extracting P_x and P_y, and Fig. 2(b) shows the process of obtaining the estimated $\hat{X}$, $\hat{Y}$, and $\hat{Z}$ positions after extracting $z^{'}$. Details are provided in Sections B and C.

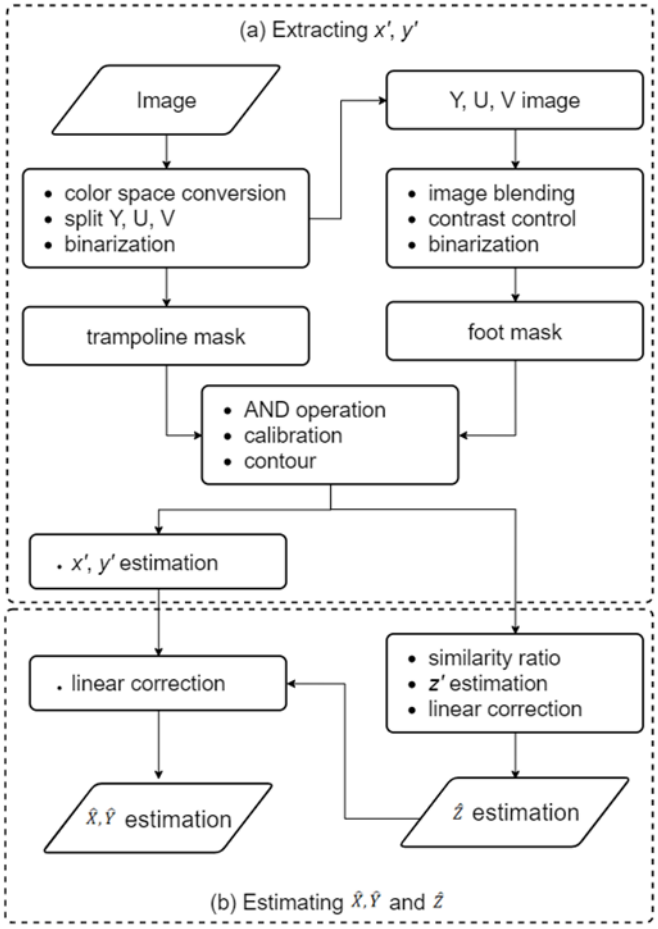


Fig. 2. $\hat{X}$, $\hat{Y}$, and $\hat{Z}$ coordinate estimation sequence

B. Estimation of Two-Dimensional Contact Position in Transverse Plane

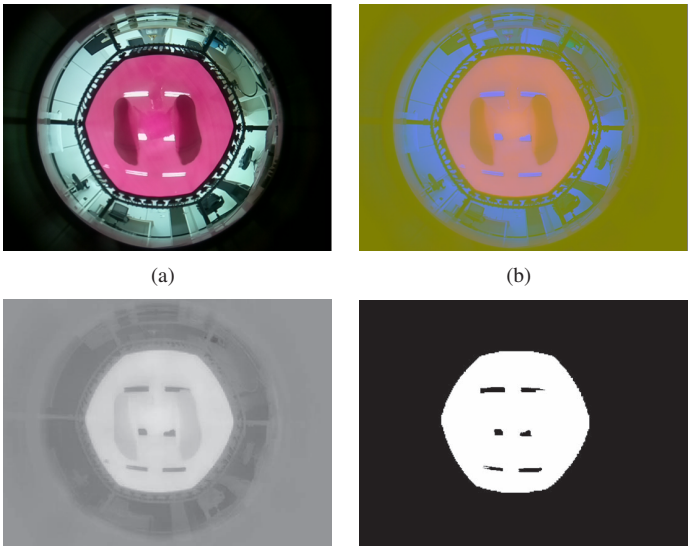
The proposed method consists of extracting the trampoline and foot masks and estimating the two-dimensional (2-D) foot contact position ($x^{'}$,$y^{'}$).

In the trampoline mask extraction step, the red pad area in Fig. 1(b) is determined as the region of interest (RoI) and is appropriately extracted from the input image. The binarization method using R-channel has difficulty in separating a bright background area. Therefore, the color space conversion method expressed by (1) [14] was applied.

$$\begin{bmatrix} Y \\ U \\ V \end{bmatrix} = \begin{bmatrix} 0.299 & 0.587 & 0.114 \\ -0.16874 & -0.33126 & 0.5 \\ 0.5 & -0.41869 & -0.08131 \end{bmatrix} \begin{bmatrix} R \\ G \\ B \end{bmatrix} \quad (1)$$

In (1), Y indicates luminance, U indicates the color component difference between the blue component and luminance

(Blue-Y), and Vindicates the color component difference between the red component and luminance (Red-Y).
The RoI was determined with a trampoline mask by performing Otsu-thresholding in the V-channel, where the red color information was emphasized [15] [16]. Fig. 3 shows the images acquired in each process.



(a) (b) (c) (d)

Fig. 3. (a) Original RGB image; (b) YUV Image; (c) V-channel image; (d) Trampoline mask extraction image.

Next, to extract the foot mask, the foot shadow must be contrasted with the trampoline area. For this, we defined a single-channel image, $Gray_{adj}$ in which the red information was intensified. Assuming that $Gray_Y$, $Gray_U$, and $Gray_V$ are the single-channel images of Y, U, and V, respectively, $Gray_{adj}$ can be defined by (2).

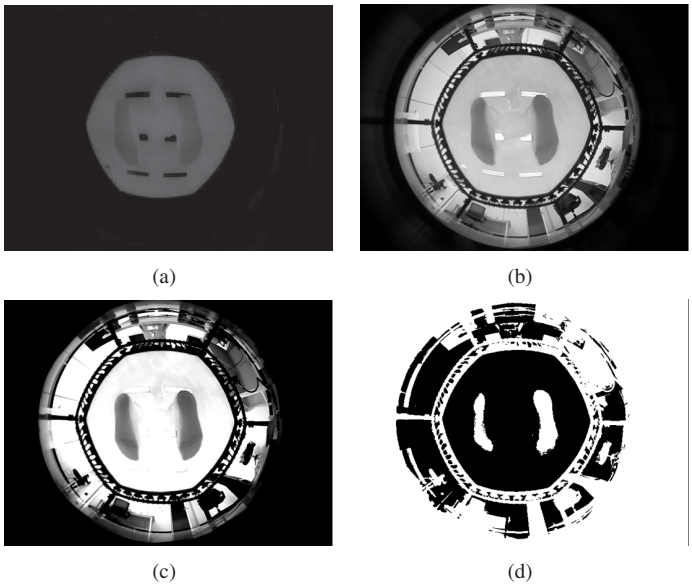
$$Gray_{adj} = Gray_V - Gray_U + Gray_Y \tag{2}$$

Fig. 4(a) shows the result of minimizing the color information, excluding the trampoline, by removing the information of $Gray_U$ with the blue color information highlighted in $Gray_V$. Next, adding the luminance $Gray_Y$ produces an image in which the contrast between the trampoline area and foot shadow is more clear, as shown in Fig. 4(b). Subsequently, the contrast ratio of $Gray_{adj}$ was amplified using contrast control, and the foot mask was extracted through Otsu-thresholding. Fig. 4 shows the images from each process.

During the 2-D position estimation, a binary image is an output by applying (3).

$$Dst(x, y) = Mask_{Foot}(x, y) \cap Mask_{Trampoline}(x, y) \tag{3}$$

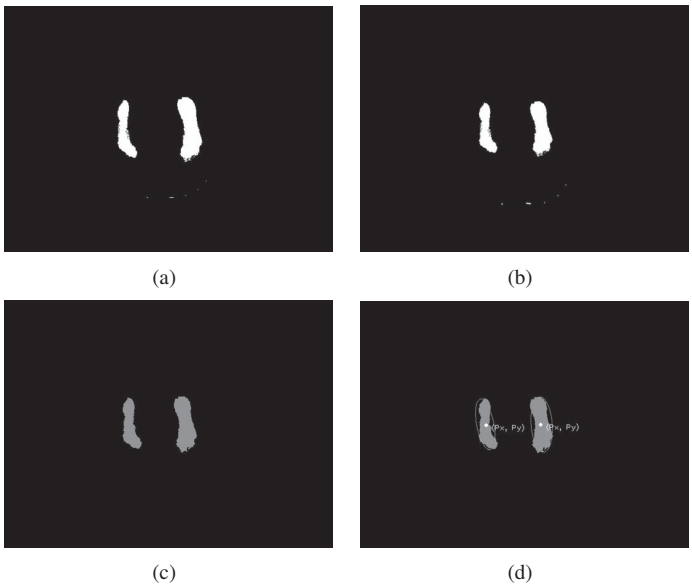
Where x and y are the image coordinates; $Mask_{Foot}$ and $Mask_{Trampoline}$ denote the foot mask and trampoline masks, respectively, and $Dst(x, y)$ is the result of the AND operation. Image distortion is removed from the acquired binary image using omnidirectional camera calibration [15] [16]. To



(a) (b) (c) (d)

Fig. 4. (a) Image of minimized color information; (b) $Gray_{adj}$-channel image; (c) Image of contrast ratio amplification; (d) Image of foot mask extraction.

increase the processing speed, calibration is performed on a single-channel image rather than a three-channel image.
In the distortion-free image, the shadow is recognized as an ellipse through contours [17] [18]. The center point of the extracted ellipse is defined as the contact position (P_x, P_y). The positions are converted into the world coordinate system as ($x^{'}$, $y^{'}$) through pixel-size values. Using this, the 2-D position can be finally estimated. Images for each step are shown in Fig. 5.



(a) (b) (c) (d)

Fig. 5. (a) Image of feet shadow; (b) Distortion-free image after calibration; (c) Contour image; (d) Estimation image.

C. Estimation of Contact Depth in the Vertical Direction

To estimate the contact depth of the foot in the vertical direction, we introduced a similarity ratio formula. When the user jumps on the trampoline, the foot shadow size observed in the image changes. Knowing the actual size of the feet and the free-standing height of the trampoline, the depth of the trampoline can be estimated from the change in the foot shadow size, as shown in Fig. 6. Specifically, it is determined based on data that can be obtained in advance, such as the height of the trampoline and the size of the user's foot, and the depth is estimated by using these data and the length of the foot shadow long axis obtained from the contour.

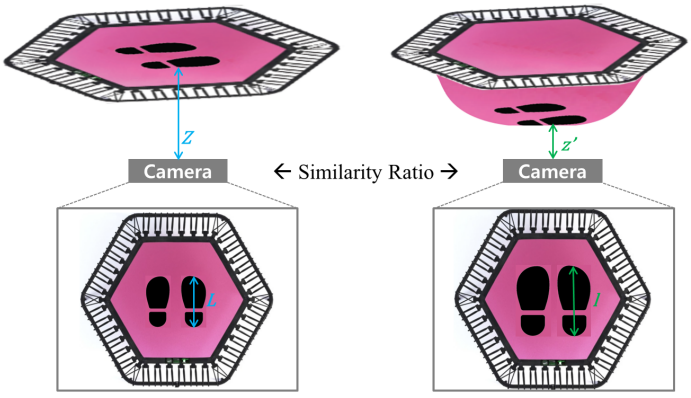


Fig. 6. Schematic representation of estimating the contact position in the z-axis.

The size of the user's foot and distance between the camera and trampoline are denoted by L and Z, respectively. If the foot size extracted through the image is denoted by l, then z' can be derived from (4) and (5).

$$L : Z = l : z^{'} \tag{4}$$

$$z^{'} = \frac{Z * l}{L} \tag{5}$$

Using the similarity ratio formula, the foot contact depth can be estimated using a single-camera module without stereo vision. However, as the field of view varies with the distance from the camera to the object, a distance-based correction for each coordinate is required. Therefore, to correct the extracted z' value, the linear correction coefficients W_z and B_z were calculated using portions of the data. Then, $\hat{Z}$ was calculated using the linear correction method expressed by (6).

$$\hat{Z} = W_z * z^{'} + B_z \tag{6}$$

Subsequently, to obtain the linear coefficient of x' and y' for each Z, through linear correction of the coefficients, an adjusted correction function $\hat{Z}$ was obtained. Assuming that $W_x(\hat{Z})$, $B_x(\hat{Z})$, $W_y(\hat{Z})$, and $B_y(\hat{Z})$ denote the corresponding x' and y' correction functions for$\hat{Z}$the 2D position ($\hat{X}$,$\hat{Y}$) can be calculated from (7) and (8).

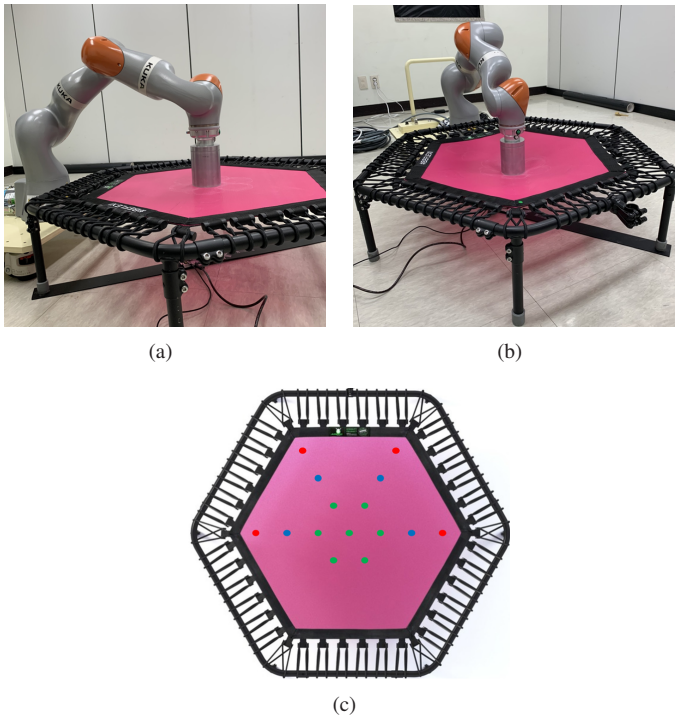
$$\hat{X} = W_x(\hat{Z}) * x^{'} + B_x(\hat{Z}) \tag{7}$$

$$\hat{Y} = W_y(\hat{Z}) * y^{'} + B_y(\hat{Z}) \tag{8}$$

III. Experiment and Results

A. Experimental Setup

To verify the I3D-FCE algorithm, it is necessary to compare the actual and estimated values of the trampoline and foot contact position. In this study, to overcome the difficulty in measuring the position of a user's foot contact position, experimental data were acquired using a manipulator (KUKA LBR iiwa 14 R820, cooperative robot, payload of 140 (kfg), precision of 2) that can quantify the motion position value on a millimeter scale. The experimental data obtained using the robot were used as the actual coordinates of the contact position. Fig. 7 shows the experimental environment with a trampoline and manipulator. A contact jig (round shape of 100 mm diameter) was attached to the manipulator to replace the foot shadow. Considering the movable range and diameter of the manipulator jig, a 2-D contact position was selected. The trampoline center was set as the origin, and 2-D reference positions were selected within movable distances of 0, 100, 200, and 300 mm from the origin. The 15 positions considered in the experiment are shown in Fig. 7(c).



(a) (b)

(c)

Fig. 7. Experimental setting using a manipulator for (a) z = 0 and (b) z ¿ 0; (c) Contact position scheme.

As the trampoline moves away from the center, the elastic force of the trampoline decreases, causing a difference in

the depth, which can be measured at each reference point. Considering this, the contact images were measured at a vertical depth of 0 and 30 mm at the red points; 0, 30, and 60 mm at the blue points; and 0, 30, 60, and 90 mm at the green points. A total of 48 3-D references were selected, and an experimental database was built by collecting 15 data points per position. 30, 60, and 90 mm were set as the maximum depth in each location because the trampoline could not be pressed deeper owing to the load weight of the manipulator. Data were measured 70 times at each of the 48 3-D reference positions to establish an experimental database. From the 3,360 images, 960 were used to obtain the linear correction coefficients for x, y, and z, and the remaining 2,400 were used for the experiment.

B. Results

The error between the actual position and the estimated position, which was obtained through the manipulator and I3D-FCE algorithm, was calculated using the root mean square error (RMSE) given in (9). P_i denotes the actual contact position (X-Y-Z); $\hat{P}_i$ denotes the position estimated by I3D-FCE ($\hat{X}$-$\hat{Y}$-$\hat{Z}$)

$$RMSE = \sqrt{\sum_{i=1}^{N} \frac{\left(P_i - \hat{P}_i\right)^2}{N}} \tag{9}$$

Table I presents the error in the 3-D point and each coordinate axis. It can be observed that the error rate is slightly higher in the z-axis than in the x- and y-axes, confirming that the estimation for the z-axis affects the overall RMSE.

TABLE I
STANDARD ERROR OF X-Y-Z ESTIMATION POSITION

	RMSE (mm)		
	Min	**Max**	**Average**
X-Y-Z	1.13	17.61	4.86
X	0.01	23.84	4.75
Y	0.02	16.01	3.90
Z	0.07	22.63	4.41

Table II presents a more detailed analysis of the average error corresponding to the distance between the origin and 2-D contact point. Similarly, Table shows the average error corresponding to the distance of the vertical depth from the origin. In Table , the points at distances of 0 and 100, 200, and 300 mm from the origin correspond to the green, blue, and red points shown in Fig. 7(c), respectively. Table II shows that the largest error occurs at a point located 300 mm from the origin.

This trend can also be observed in Table III; the error is relatively large at depths of 0 and 30 (mm). This error includes the error in the area outside the trampoline, corresponding to the red dot shown in Fig. 7(c).

TABLE II
STANDARD ERROR CORRESPONDING TO X-Y DISTANCE

Point Color	**Distance (mm)**	**RMSE (mm)**			
		X	**Y**	**Z**	**X-Y-Z**
Green	0.0	1.42	2.22	4.16	3.16
	100.0	3.14	3.20	4.03	3.82
Blue	200.0	5.63	4.40	3.84	4.96
Red	300.0	9.96	6.11	6.53	8.69

TABLE III
STANDARD ERROR CORRESPONDING TO Z DISTANCE

Distance (mm)	**Average RMSE (mm)**			
	X	**Y**	**Z**	**X-Y-Z**
0.0	4.97	4.47	4.38	5.21
30.0	5.84	4.39	4.75	5.48
60.0	3.97	3.73	4.64	4.50
90.0	3.23	1.95	3.43	3.37

The relatively large number of errors in the verified outer part of the trampoline relates closely to the trampoline structure. A force applied in the vertical depth direction at the outer part of the trampoline often causes interference between the trampoline and foot masks, which can lead to relatively large errors compared to those at other positions. In addition, an omnidirectional camera calibration process was performed to compensate for the characteristics of the wide-angle lens; but errors may occur in the external area because there are limitations due to the use of a wide-angle range.
However, jumping fitness exercises are normally performed at the center of the trampoline because the user needs to exercise in the most stable position. Therefore, it can be considered that the position estimation error for the outer exercise position is within the acceptable range for the jumping fitness system.
Nevertheless, to link the exercise effectiveness with various content such as VR and games in the future, high position estimation accuracy is required, so a more accurate position estimation needs to be performed at all positions that users have contact with.

IV. CONCLUSION

In this study, we proposed the I3D-FCE algorithm to estimate the coordinates of the foot contact position during trampoline motion. First, a prototype of a jumping fitness system that included a jumping fitness trampoline, an embedded processing board, and a wide-angle fish-eye camera module was constructed. Then, the 2-D contact position in the transverse plane was estimated using the binary image of the foot shadow acquired by extracting the binary masks

of the trampoline and foot. The contact depth was estimated using the similarity ratio of the shadows. A manipulator was employed to verify the estimated value of the foot contact position experimentally. The X and Y coordinates of the foot contact position and the Z coordinate of the foot contact depth had estimated average errors of 4.75 (mm), 3.90 (mm), and 4.41 (mm), respectively. The 3-D estimation position had an average error of 4.86 (mm).

Based on the location estimation algorithm developed in this study, we plan to research advanced algorithms suitable for jumping fitness exercises by obtaining more location data and using various jig sizes. Further work includes building a more realistic environment, improving location accuracy through experiments, and adding the weight of humans to the estimated position value to calculate the amount of exercise and calories burnt, and linking them to jumping fitness games and VR content.

ACKNOWLEDGMENT

This work was supported by the KITECH RD Program of the Development of Soft Robotics Technology for Human-Robot Coexistence Care Robots (PEO20052)

REFERENCES

[1] G. Şahin, E. Demir, H. Aydın, G. Şahin, E. Demir, and H. Aydın, "Does mini-trampoline training more effective than running on body weight, body fat, vo2 max and vertical jump in young men," *International Journal of Sports Science*, vol. 6, no. 1, pp. 1–5, 2016.

[2] C. Witassek, N. Nitzsche, and H. Schulz, "The effect of several weeks of training with mini-trampolines on jump performance, trunk strength and endurance performance.," *German Journal of Sports Medicine/Deutsche Zeitschrift fur Sportmedizin*, vol. 69, no. 2, 2018.

[3] M. Sadeghi, G. Ghasemi, and M. Karimi, "Effect of 12-week rebound therapy exercise on static stability of patients with spinal cord injury," *Journal of sport rehabilitation*, vol. 28, no. 5, pp. 464–467, 2018.

[4] P. Daneshvar, G. Ghasemi, V. Zolaktaf, and M. T. Karimi, "Comparison of the effect of 8-week rebound therapy-based exercise program and weight-supported exercises on the range of motion, proprioception, and the quality of life in patients with parkinson's disease," *International journal of preventive medicine*, vol. 10, 2019.

[5] "Iumping fitness." https://www.jumping-fitness.com/.

[6] "Fitforfun." https://www.jumping.fitness/.

[7] H. Mori, K. Shiratori, T. Fujieda, and J. Hoshino, "Versatile training field: the wellness entertainment system using trampoline interface," in *ACM SIGGRAPH 2009 Emerging Technologies*, pp. 1–1, 2009.

[8] S.-G. Lim, S.-H. Jung, and J.-H. Huh, "Visual algorithm of vr e-sports for online health care," in *Healthcare*, vol. 9, p. 824, Multidisciplinary Digital Publishing Institute, 2021.

[9] R. Kajastila, L. Holsti, and P. Hämäläinen, "Empowering the exercise: A body-controlled trampoline training game," *International journal of computer science in sport*, vol. 13, no. 1, pp. 6–23, 2014.

[10] P. W. Connolly, G. C. Silvestre, and C. J. Bleakley, "Automated identification of trampoline skills using computer vision extracted pose estimation," *arXiv preprint arXiv:1709.03399*, 2017.

[11] M. Tiator, O. Köse, R. Wiche, C. Geiger, and F. Dorn, "Trampoline jumping with a head-mounted display in virtual reality entertainment," in *International Conference on Intelligent Technologies for Interactive Entertainment*, pp. 105–119, Springer, 2017.

[12] T. Helten, H. Brock, M. Müller, and H.-P. Seidel, "Classification of trampoline jumps using inertial sensors," *Sports Engineering*, vol. 14, no. 2-4, pp. 155–164, 2011.

[13] D. Eager, C. Chapman, K. Bondoc, *et al.*, "Characterisation of trampoline bounce using acceleration," in *Proceedings: the 7th Australasian Congress on Applied Mechanics (ACAM 7), 9-12 December 2012, the University of Adelaide, North Terrace Campus/National Committee on Applied Mechanics of Engineers Australia*, p. 1062, Engineers Australia, 2012.

[14] M. Podpora, G. P. Korbas, and A. Kawala-Janik, "Yuv vs rgb-choosing a color space for human-machine interaction.," in *FedCSIS (Position Papers)*, pp. 29–34, 2014.

[15] N. Otsu, "A threshold selection method from gray-level histograms," *IEEE transactions on systems, man, and cybernetics*, vol. 9, no. 1, pp. 62–66, 1979.

[16] "Image thresholding." https://docs.opencv.org/master/d7/d4d/tutorial_py_thresholding.html.

[17] "Camera calibration and 3d reconstruction." https://docs.opencv.org/3.0.0/d9/d0c/group__calib3d.html#ga93efa9b0aa890de240ca32b11253dd4a.

[18] D. Scaramuzza, *Omnidirectional vision: from calibration to root motion estimation*. PhD thesis, ETH Zurich, 2007.