JST: Smart Systems and Devices
Volume 32, Issue 1, January 2022, 001-008

A Real-Time Tracking Algorithm for Human Following Mobile Robot using 3D Sensor

Hoang Hong Hai
Hanoi University of Science and Technology, Hanoi, Vietnam
Email: hai.hoanghong@hust.edu.vn

Abstract

Detecting and tracking a particular person are considered the main tasks of a mobile robot. In this paper, we propose a real-time mobile robot system using 3D Kinect sensor for automatically detecting, tracking, and following humans. This method is based on depth information, skeleton, and color of humans from 3D camera. Firstly, the depth image is taken from 3D Kinect to segment the individual region. After that, we calculate the body length, shoulder length, and arm length in combination with the color of target's clothes to gather as the material for the tracking task. Finally, the mobile robot which is controlled by voice command can recognize and follow the single target person. The effectiveness and robustness of the proposed method are evaluated in comparison with the method based on single skeleton or color of objective. Moreover, our proposed method can identify the target again when it disappears and appears again in the frame. All experiments are implemented with the support of model designed to integrate 3D Kinect camera on a wheeled mobile robot. The velocity and direction of the wheeled mobile robot are controlled by a proportional-integral-derivative controller to keep a constant velocity all the time. The experiment result is shown that the proposed system has worked effectively, stably, and flexibly and the success rate is more than 90%.

Keywords: Detection, 3D depth image, tracking, mobile robot, real-time.

1. Introduction

Recently, with the development of image identification technology and 3D sensor technology, mobile robot research has made remarkable advancements. Mobile robots that can follow people at any time have become more trendy. As long as the robot can identify the targets that need to be followed, the robot can be operated by the user without having to manipulate the robot. 3D sensor Microsoft Kinect [1,2] is a popular, and relatively cheap device, which has multiple applications in the robotics field such as detecting and tracking objects. The RGB-D Kinect camera can provide an easy method of detecting up to six people at the same time in one frame [3]. The traditional method requires a person to stay within the Field of View (FoV) of a camera to be tracked continuously and cannot re-identify the same person if the object disappears or be blocked from the camera. This leads to the inconvenience in applying for real-time life systems. To tackle this problem of person re-identifying, we use depth image and colour image, both of which were provided by RGB-D Kinect camera. The people who have the same skeleton size and same colour of clothes are identified by the same Tracking ID [4,5].

In computer vision, there are many classification methods used to distinguish objects. They are based on information about the shape or colour of objects. Commonly methods such as SURF, SIFT, HOG, Haar and LBP are applied in object recognition [6-10]. The recent development in machine learning and deep learning approach has a great impact on the accuracy of object detecting and tracking. However, the long latency time is still a challenge when applying these techniques to a lightweight, mobile computer. Therefore, using information about the colour of tracked person's clothes is suitable for tackling the problem of tracking a specified person. In addition, Kinect also provides skeleton tracking of the person in frame, we use this feature to get clothes region, measure skeleton size to identify target person [11,12].

In this paper, we introduce a method in combining Kinect's RGB sensor and IR sensor to solve the problem of the inability to re-identify persons when they are out of Kinect's FoV. Firstly, the depth image information of the object is captured by the Kinect sensor. Then the body length, shoulder length, and arm length in combination with colour of target's clothes to gather as the material for the tracking task. The input of the object's motion data to the computer bases on the image identification technology. The mobile robot is controlled by its environment is built by software. The system synchronously controls the robot through the motion data in which the robot can recognize and follow the single target person. This proposed method can be applied for human tracking robot to recognize,

ISSN: 2734-9373
https://doi.org/10.51316/jst.155.ssad.2022.32.1.1
Received: February 23, 2021; accepted: September 16, 2021

follow person and re-identify the target when that person disappears from viewing and returns, based on clothes' color and skeleton size. The structure of the paper is presented as follows. After the introduction in Section 1, Section 2 shows the method to perform human region segmentation depending on the depth image. Section 3 contains a description of the proposed tracking human method. Section 4 shows the experimental results. Finally, Section 5 presents discussion while conclusions and future works are shown in Section 6.

2. Human Region Segmentation Based on Depth Image

This section presents the method used for extracting human region out of the environment. Camera RGB-D Kinect gets both depth image and RGB image as shown in Fig. 1, but they do not have the same coordinates joints due to the different arrangements of RGB sensor and IR sensor on Kinect. Therefore, to use the depth and colour information in the right way, we must synchronize the coordinates of two formats of image. To overcome this problem, we use depth images for extracting a person out of the background and create a mask to storage the person region. Then, visualizing on a new green background, we take the colour of person region from colour image based on calibration of depth image and RGB image. Pushing it to the mask, where we store the depth person region, we can verify the colour and depth information of the person in the same coordinate.

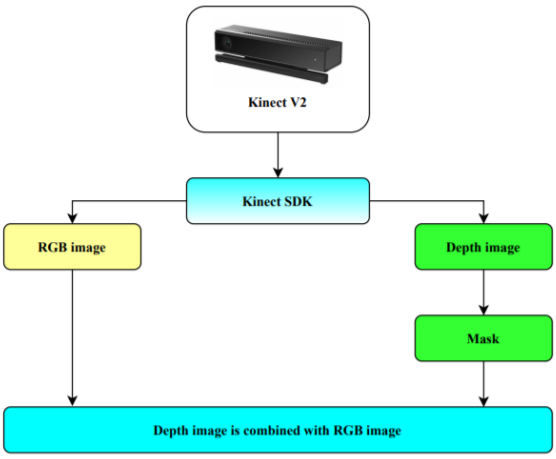

Fig. 1. Human region segmentation process

Fig. 2. Depth and player index bits

After camera calibration in depth và RGB, using the support of Kinect SDK library released by Microsoft, we easily analyze depth images and detect human region. It can identify 6 people at the same time and gives an ID for each person in depth frame. Concretely, each pixel in depth image is represented in a format of 16 bits register as in Fig. 2.

Bits 0 to 2 hold Player Index and bits 3 to bits 15 contain the depth value. To check if the pixel belongs to a human, we use AND operator with a mask of the form 0000 0111 to get the last 3 bits as shown in Fig. 2. The value of these 3 bits is obtained whether the pixel is human pixel or not. If its value is 0 which means no one is in that pixel. Otherwise, if values run from 1 to 6, then the pixel belongs to one of the persons. Collecting all pixels belonging to the person region and ignoring the background, we can separate the human region as shown in Fig. 3.

However, the mask illustrated in Fig. 3 does not carry the color of target region. The next step is the fitting color to this mask. Mapping this mask on a RGB image we get human region segmentation on RGB space. This requires calibration of RGB and IR sensors since these two sensors are in different positions. However, we do not mention that here but focus on the algorithm to follow a human. The result after the segmentation process is shown in Fig. 4. The result contains both depth and color information of the target person.

Fig. 3. Mask was created from depth image

Fig. 4. The result of the segmentation process

3. The Proposed Human Tracking Method

3.1. Clothing Extraction

The contribution of our paper is proposing a method that recognizes and identifies the target person among the person appearing in the vision of the robot. To handle this task, we do not consider the skeleton size but also the color feature of the target. Among color distinguishing algorithms, which can adapt to real-time requirements for mobile robot, we choose a common method based on histogram to calculate the equivalent index between candidate and target human.

The Kinect does not only provide a depth image but also a method to form the skeleton of the human in frame. We use this to determine the area of clothes based on the joints' coordinate. This method is affected by immutable light conditions that make recognition process change easily and become less accurate. It is an important problem, as it has a great impact on re-identifying humans. To handle this problem, both regions of T-shirt and trouser are transformed into HSV color space and only Hue and Saturation channels are used for histogram computing to decrease the effect from environment.

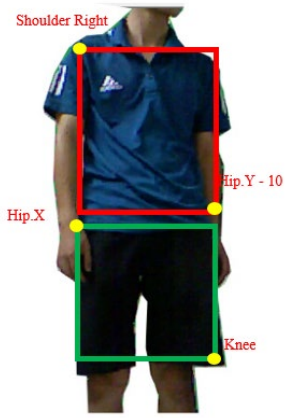

Fig. 5. Human clothes region after segmentation

From the segmented image in Fig. 5, we define two areas of T-shirt and trouser based on skeleton coordinates as shown above. The clothes areas are converted into HSV space, Hue and Saturation channels are used to calculate by the formula:

$$R_{h,s}^{k} = \sum_{i}\sum_{j} \partial_{i,j}^{h,s} \eta Q_{i,j} \tag{1}$$

where k is the region human T-shirt or trousers, h is the number of Hue channels, s is number of bin Saturation channels, $\partial_{i,j}^{h,s} = 1$ if pixel in T-shirt, $Q_{i,j}$ if weight if pixel (i, j), η is constant to normally result.

After calculating histogram of each region, we compare it with histogram of target person from the input dataset depending on the intersection of comparison type.

$$Pc(I, \hat{I}) = \sum_{k=1}^{2} W_k \frac{\sum_h \sum_s \min(R_{h,s}^{k}, \hat{R}_{h,s}^{k})}{\sum_h \sum_s R_{h,s}^{k}} \tag{2}$$

where $Pc(I, \hat{I})$ is score similarity between I and $\hat{I}$, W_k is the weight of T-shirt region and trousers region.

3.2. Skeleton Size Extraction

As mentioned above, Kinect provides a method to identify the skeleton of a human in a frame. Depending on skeleton size we can get additional information such as body length, shoulder length and arm length for the process of identifying humans.

From the model in Fig. 6, we calculate the length of the body, arm, and shoulder

- Body Height = D (Head, shoulder center) + D (shoulder center, hip center) + D (hip right, knee right) + D (Knee right, ankle right)
- Arm length = D (shoulder right, elbow right) + D (elbow right, hip right) + D (hip right, wrist right)
- Shoulder width = D (shoulder right, shoulder left)

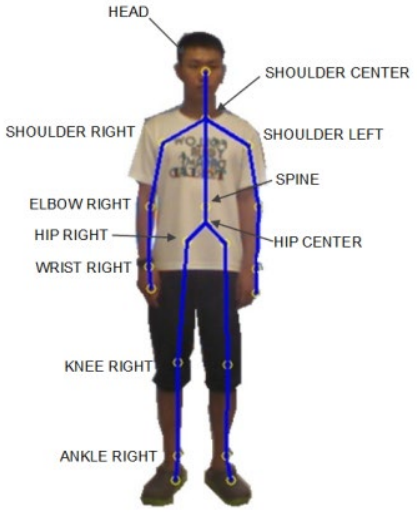

Fig. 6. Joints in Skeleton tracking

Because of error in the real-time process, we assume results follow Gaussian distribution and calculate follow the formula:

$$P_i = \frac{1}{\sigma_i \sqrt{2\pi}} \exp\left(\frac{-(L_i - \mu_i)^2}{2\sigma_i^2}\right) \tag{3}$$

where P_i is length of body, shoulder, arm, μ_i is mean value and σ_i is standard deviation value.

Combine all values of length we have:

$$P_B = \alpha_1 P_{body} + \alpha_2 P_{arm} + \alpha_3 P_{shoulder} \quad (4)$$

where α_1, α_2, α_3 are weights of each region size. Combining both color extraction and skeleton size extraction we get the following formula:

$$P = \beta P_c + (1-\beta) P_B \quad (5)$$

where β is the weight of the proposed method.

This P indicator is used to supervise the target person. Because of considering both body element size and color feature, which are reflected by P, the system can follow exactly target person although the other people appears and obscures target. In other words, P is the confidence index of the target person.

4. The Practicing on Mobile Robot

4.1. Mobile Robot System

The proposed method is integrated into the robot as shown in Fig. 7. The RGB-D Kinect sensor is installed on the top of the wheeled mobile robot. The mobile robot is controlled by an 8-bit AVR microcontroller integrated on an Arduino Uno board. The microcontroller receives the command from Kinect and controls the velocity of motors and makes the mobile robot move stably and follow the human smoothly. Processing depth image, computation skeleton size, and detecting human are processed on Laptop Asus K555L Core I5, 2.2Ghz Ram 8Gb with supporting of Kinect SDK.

To achieve our goal, we develop the algorithm for tracking tasks as detail as Fig. 8. Concretely, when the system starts, it captures a set of samples of target, who is in front of the mobile robot, to calculate target's features. Then a process of consistent considering target's location, target's distance, and comparison between the person on frame and target sample status to decide the command to the mobile robot. The robot will be controlled to make sure that the person in frame is right target and keep the target always in the middle side and has a 1.2 m distance.

Control task and image processing task are divided into two separate threads to avoid data mining congestion for real-time purposes. The commands sent to the robot are encoded into specific formats. Firstly, when the human says "Get Sample", the robot first takes 100 frames of the person in front of the robot to make target human reference. The robot follows the target person when it receives the voice command "Follow me".

Secondly, the segmentation process is taken to get the body region, which is fed into skeleton size and color feature calculation process. The robot's motion is processed and calculated based on the location of the center of the human in the frame, then commands are sent to move the robot and keep the center of the target human always in the middle of frame and the distance from the robot to target person is always from 1.2 to 1.4 meters.

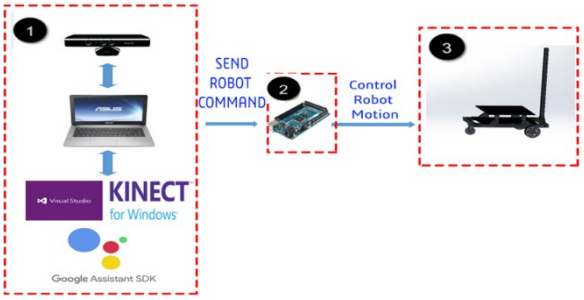

Fig. 7. Control robot system

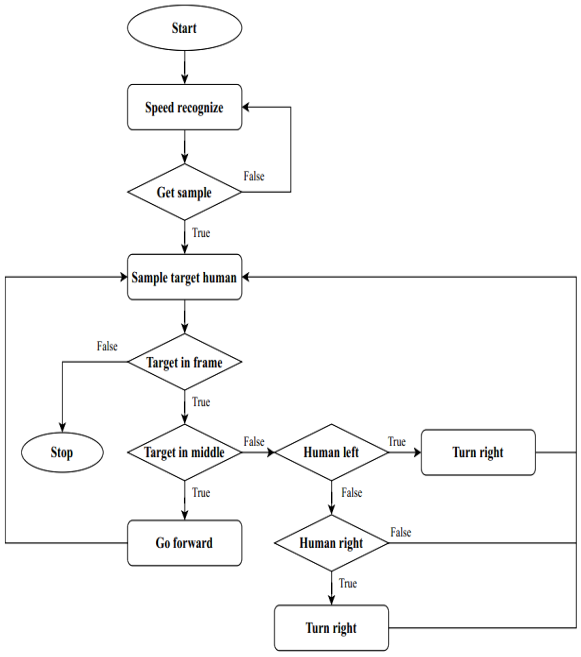

Fig. 8. Algorithm control mobile robot with proposed tracking method

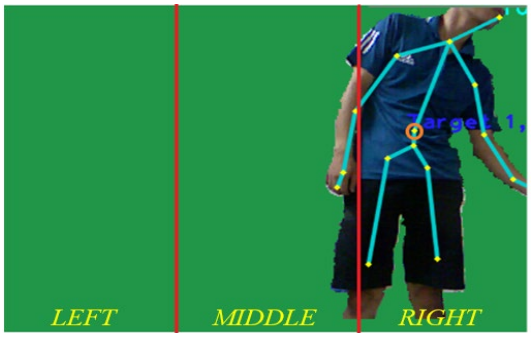

Fig. 9. Frame divided for determining position of the target person

We decide to split the image into three parts in width: left, middle, and right as displayed in Fig. 9. The part for which target's center is between 0 and 210 (pixel width) will represent the left part, and the part between 210 and 430 will be the middle part while the remaining range shall be the right. Depending on the position of the center of the person, the computer continuously sends command to the robot in string format.

Depending on the case whether the user and the obstacle are in the same range or not, there is an output sent to the Arduino to control the movement of the robot: to move left, right, forward, or even stop in the case where the user is located at a distance less than or equal to 1 meter from the robot. In the case in which the user is in the middle range; if the user is located within 1 meter, the robot should stop. Otherwise, check if the closest value is also in that range; if so, then there is an obstacle hampering the trajectory of the robot while tracking the user detected. If no obstacle is detected the robot will continue in its way toward the user as Fig. 10 illustrates. The user is located in the middle part while the obstacle is in the right part, therefore, the robot continues in its way as there is no obstacle. The second case is that when the user is in the left part, the robot here also stops if the user is located within 1 meter. Like the previous case, check whether the closest value is also in the same range as the user so that there is an obstacle that must be avoided.

The robot then moves to the right and continues tracking the user. On the other hand, if there is no obstacle between the user and the robot, the last will move as the user does. The final case, as shown in Fig. 5, is the same as the previous one, but the difference here is that the user is on the right side so the robot will move to the left to avert the obstacle instead of moving to the right.

H< position>#	
Position	**Description**
T	Go forward
R	Turn right
L	Turn left
B	Go backward

Fig. 10. From left to right: LEFT, MIDDLE, RIGHT

As shown in above result, we can easily see that system could detect the target person with high accuracy despite the difference in target person's pose and create a skeleton mask at the same time. When the person has significant difference in skeleton size and clothes' color with the target, the system would not detect and return the ratio representing how they are different from the target.

4.2. PID Control

To guarantee the effectiveness of the system, the stability of velocity plays an important role, which deeply depends on the control of two wheels. The most suitable method for this task seems to be the PID controller. We manipulate and maintain the reliable wheel's speed by generating PID factors. The mobile robot remains at a fixed speed even though it carries a laptop and camera as Fig. 7. For turning PID factors we can use Ziegler-Nichols method or turning by MATLAB software. However, applying Ziegler-Nichols is not considered as a reasonable choice. This traditional method needs to find out the point, at which the robot begins to oscillate and then estimate PID factors based on this value. This task is hard to deal with because of the low encoder equipped on motors. This causes velocity updating to be not fast enough to figure out the ultimate gain factor. It is easy to confuse consistent oscillations with vibrations. Therefore, using MATLAB is a convenient application for this research. This important part is achieved by estimating the transfer function of motors then using it to simulate the controller model which can forecast the reliable PID factors. We first collect 200 velocity samples to feed into the system identify toolbox in MATLAB which uses these samples to calculate the transfer function as Fig. 11. The figure shows that the generated transfer function is fit with fixed signal, which is expected velocity. This transfer function is an element of PID simulation model which allows forecasting PID factors based on turn response and block response. Fig. 12 illustrates the results of turning PID factor. The output of PID simulation model represents for wheel's motor which is exactly equal to expected result in simulation environment. The wheel speed reaches and is stable at set value after just around 2.3 seconds. This result prompts the reliability of this method. However, in the real environment, the mobile robot approximately meets the expected value after just around 2 seconds. As shown in Fig. 13 and Fig. 14, although it is clear that when experiencing with no load, the actual speed of robot in real environment does not dramatically match the set value but is still perfectly close to it. It is more noticeable that when robot brings with hardware, the velocity is acceptable even there are many vibrations. This stable velocity allows our system to guarantee the performance of computer vision tasks and remain system's smoothness.

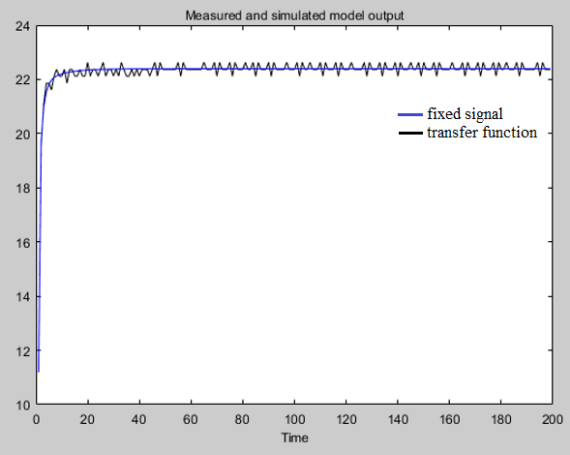

Fig. 11. An illustration of fixed signal and result of the transfer function.

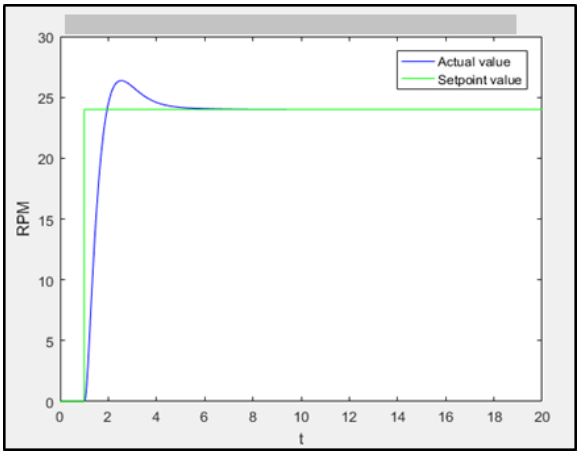

Fig. 12. The simulation of setpoint value and estimation of real value using MATLAB

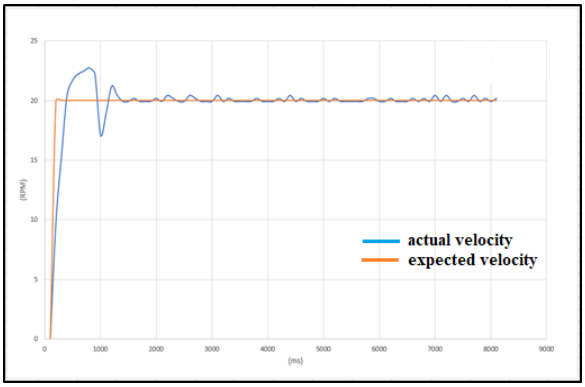

Fig. 13. Comparison of actual velocity with no load and expected velocity

Fig. 14. Comparison of actual velocity with loads and expected velocity

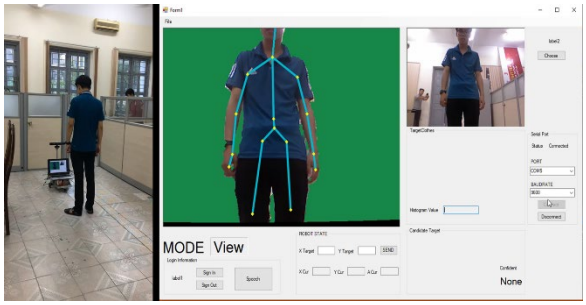

Fig. 15. Mobile robot control through Kinect

With the support of our PID controller, the system performs well in tracking objects. This controller helps the system works smoothy and it obiviously inhances the performance of tracking system. Acquiring images during robot's movement is detrimentally effected, which leads to the poor quality of input data. As a result, the accuracy of recognising people's skeleton is surely severe. This PID controller does not merely solves the problem of robot's movement and robot's velocity, it contributes the efficience of image processing system.

5. Experimental Result

The goal is to use Kinect camera to support human tracking for mobile robots. The proposed method takes advantage of Kinect in segmenting people, determines the distance from human to the camera. The combination of RGB sensor and depth sensor creates a real-time tracking system for human following with high accuracy than the traditional sensor. When the subject performs in the scope of the Kinect, the procedure of the Kinect gets the subject's skeleton information and calculates the arm joint angle range. The experiment situation is shown in Fig. 15.

Fig. 16 illustrates how our system verifies the target among other persons. A comparison of clothes color and skeleton size is processed to check how does the object is similar to the target. The confidence index reflects the comparison and is used to consider what person is the right target. As shown in Fig. 16, if it got confidence index higher than 0.75, it is known as the target. If the target person changes the pose or even turns back around, the indicator index still confirms the target. While other gets index lower it will be ignored and does not make the adversity to the operation of system.

To demonstrate our performance over than traditional approach, we compare our method and method applying only skeleton or color process. We practice 50 times for each approach with different person poses to verify the flexibility. The detail of the result can be figured out in Fig. 17. Obviously, using only skeleton information without comparing color features gets very low accuracy when tracking the target. The target person will be confusingly understood when the other person appears. Applying color features only seemly increase the accuracy of the system. However, it does not respond to our command because considering color features without skeleton causes some mistakes. Concretely, when people have the same clothes color but different in skeleton size, the robot cannot analyze and distinguish it. Therefore, our proposed method takes advantage and fix the disadvantages of the two approaches above by supervising both skeleton size and color feature. Our research gets a significant result when compared with traditional methods.

In terms of inference time, our model can satisfy the continuous operation of tracking tasks. The response speed is 100 ms when human moves continuously without frame lagging as shown in Fig. 18. The accuracy of recognizing the specified person is 92%. The thread is relatively stable without overlapping bandwidth between the streams.

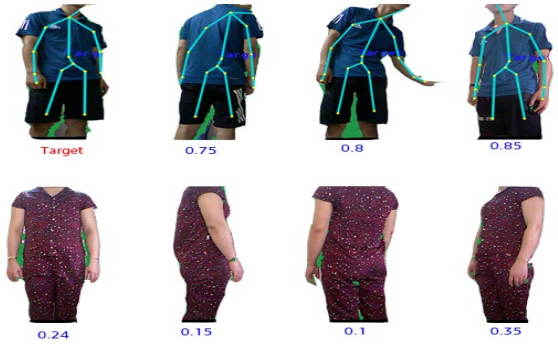

Fig. 16. Result of proposed method

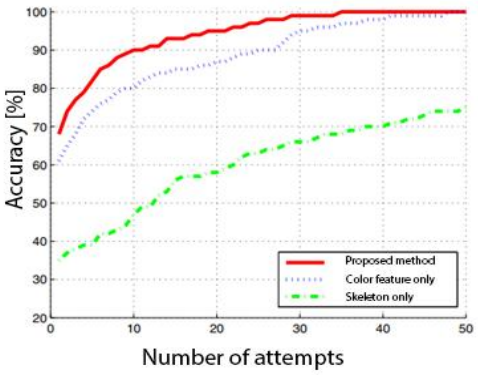

Fig. 17. Comparison of different methods

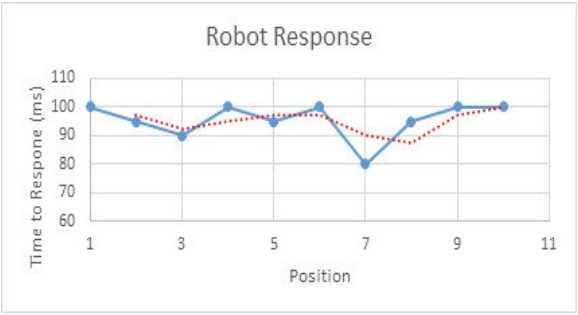

Fig. 18. Response time of the mobile robot

Results show that mobile robot works efficiently in immutable light conditions with different angles. Through these experiments, we confirm that the proposed method can track specific people efficiently with low latency time. Moreover, the effectiveness is also considered in the aspect of movement of target person. The mobile robot always keeps the target forward in 1.2 m of distance and keeps the target at the middle of vision. Although when the person turns around or step backward, move left or right, the system still smoothly operates.

6. Conclusion and Future Work

In this paper, we have implemented a real-time detection, tracking function on a controller mobile wheeled robot, which can be easily adapted as an intelligent system for helping people carry the luggage in venues like public areas such as railway stations, airports, bus stations and museums, etc. By combining both skeleton size in spatial space and color feature of target, our method perfectly guarantees the accuracy in identifying the target. This leads to the stable and continuous operation of mobile robot. Besides, using proportional-integral-derivative controller, the system can work stably and move smoothly. Moreover, analyzing color of person's clothes on HSV color space decreases the effect from environment. Experimental results show that users can easily use voice commands to control the mobile robot for the detection and tracking of human with high accuracy although the person disappears from the vision of the camera and returns.

In the future, we will apply the human detection and tracking on an ARM-based embedded platform, e.g., Raspberry Pi 4. Running on an embedded platform helps decrease the system complexity. In addition, we can add more RGB-D cameras or LIDAR sensor for 360-degree video capturing of the environment.

References

[1] Jarret Web, James Ashley, Beginning Kinect Programing with the Microsoft Kinect SDK 1st Ed, Apress Publishing, USA, 2012

[2] Soroush Falahati, Open NI Cookbook, 1st Ed, Packt Publishing, USA, 2013

[3] Webb, Jarret, Ashley, James, Beginning Kinect Program with the Microsoft Kinect SDK, 1st Ed, Apress, USA, 2012. https://doi.org/10.1007/978-1-4302-4105-8_1

[4] Shiying Sin, Ning, An, Human recoginze for following robot with a kinect sensor, ROBIO, Qingdao, Chinna, 2016, pp.16709654

[5] Abdel- Mehsen Ahmad and Hiba, 3D Sensor-based moving human tracking robot with obstacle avoidance, IMCET, Beirut, Lebanon, 2016, pp.16523074.

[6] Purvi Agarwal, Pranjal Gautam, Anmoal Agarwal, Vijai Singh, Human follower robot using kinect, IRJET, India,2017, pp.2395-0072.

[7] Doan Thi Huong Giang, Vu Hai, Tran Thi Thanh Hai, Ultilizing depth image from kinect sensor: error analysis and its applications, FAIR, Thai Nguyen, Vietnam, 2014.

[8] Cheng-An Yang and Kai-Tai Song, Control design for robotic human-following and obstacle avoidance using an RGB-D camera, ICCAS, Jeju, Korea, 2020, pp.19301946. https://doi.org/10.23919/ICCAS47443.2019.8971754

[9] Zhang, Huang, Rui Jun Yan, Wen Shen Zhou, Long Sheng, Binocular vision sensor (kinect)-based pedestrian following mobile robot, Applied Mechanics and Materials, Trans Tech Publications, Ltd., October 2014 https://doi.org/10.4028/www.scientific.net/AMM.670-671.1326

[10] Shih, Ching-Long, Chao-Cheng Li, A people-following mobile robot using kinect and a laser scanner, Robot Autom Eng J (2018) pp. 1-8 https://doi.org/10.19080/RAEJ.2018.02.555578

[11] Armando Nava, Leonardo Garrido and Ramon F. Brena, Recognizing activities using a kinect skeleton tracking and hidden markov models, MICAI, Tuxtla Gutierrez, Mexico, 2015, pp.15413297. https://doi.org/10.1109/MICAI.2014.18

[12] José-Juan Hernández-López, Ana-Linnet Quintanilla-Olvera, José-Luis López-Ramírez, Francisco-Javier Rangel-Butanda, Mario-Alberto Ibarra-Manzano, Dora-Luz Almanza-Ojeda, Detecting objects using color and depth segmentation with Kinect sensor, Procedia Technology, Vol 3, pp.196-204, 2012 https://doi.org/10.1016/j.protcy.2012.03.021