Patient necessity notification system based on gesture recognition (Kinect V2) and internet of things using selection frame method

F Febriansyah¹, N A Suwastika² and H Fakhrurroja³

¹,² School of Computing, Telkom University, Bandung, Indonesia
³ Technical Implementation Unit for Instrumentation Development, Indonesian Institute of Sciences, Bandung, Indonesia

E-mail: ¹febbyfeysyah@gmail.com, ²anggis@telkomuniversity.ac.id, ³hani002@lipi.go.id

Abstract. The use of gesture recognition integrated with Internet of Things (IoT) as a system for detecting patient needs provides information speed and accuracy for nurses. One tool that is widely used to implement gesture recognition is Kinect (Version 2). For data communication, Kinect will be collaborated with IoT in the process of sending data from the start after gesture recognition to the presentation of information to end users. The problem that arises when Kinect is integrated with IoT is the burden on the network because the average number of frames read by Kinect in one second is 30 frames. In this study, a frame selection method was developed to minimize the number of frames sent to IoT networks by minimizing the number of duplicate frames. The selection frame method on gesture recognition integrated with IoT is then tested for performance with distance and delay parameters. The system implementation results are able to read 100% gesture recognition, and the accuracy of the information sent by the nurse is 100% in the testing environment. The optimal distance that can provide a reading of more than 95% is 2.5 meters. The optimal value of 100% is not achieved because some testing of the falling position cannot read the whole skeleton.

1. Introduction

In health services, promptly and quickly responding to the patient's needs is a service priority. To meet these patient needs, effective and efficient information management is needed. Some cases of patient needs that require fast and accurate information are patient requests for infusions, other patient needs, and emergency calls. On the other hand, nurses can experience a falling incident when the patient is alone in the room. On the other hand, nurses also need specific information so that they can provide optimal services. In health care procedures, when there is a specific call, the nurse can provide the equipment needed by the patient. The nurse can also confirm the patient's request, so that the patient can adjust conditions. Based on the results of a survey of hospitals in the city of Bandung, Indonesia, most hospitals do not have a system that can quickly and accurately manage information for patients and nurses.

To meet the needs of health services in the hospital, a system is needed that is able to recognize the patient's orders or requests, then convey information to the nurse quickly and accurately, then the nurse can confirm the order until the patient's readiness information can be known by the patient without a long wait. Some alternative systems for reading patient requests are bells, voice commands, and gesture recognition. The use of bell is a conventional method that is widely used in hospitals. The disadvantage
of this method is that the nurse does not immediately know what commands are requested by the patient. In addition, confirmation of the availability of nurses cannot be known directly by the patient. When the condition of the room of the nurse center (station) is empty the patient will not be handled.

In Mario Herryn Tambunan's research, et al. [1], in implementing Indonesian voice command recognition-based system, the highest average accuracy obtained was 94.45% only obtained at a distance of 0.5 - 1.5 meters between someone with Kinect V2 sensor, more than that distance the accuracy decreases. In addition, noise in the room or the environment around the system cannot be minimized, resulting in a decrease in the accuracy of the recognized commands. On the other hand, so far there have been no studies that can detect incidents such as falling using speech recognition. Whereas in the research of Natasha Abner, et al. [2], that gesture is a universal language for communication.

Kinect has the ability to detect body movements of someone in front of them based on a skeleton that is read by the sensor. Not only in bright room conditions, but even in dark room conditions all points of one's skeleton can be detected. Kinect can identify and distinguish between right- and left-hand movements. In version 2, the skeleton detection is up to 25 joint points, more than version 1 which is only 20 joint points. In addition, version 2 has increased accuracy and data processing is faster than version 1.

This study implements a system that utilizes Microsoft Kinect Sensor V2 technology that can recognize the patient's body movements as a proposed command and integrate it with IoT so that nurses can monitor each patient requests via a browser that supports HTML5 on each device. One of the limitations of an IoT-based system is how to minimize the burden of sending data on the internet network. To handle this problem, the selection frame method is used. Selection frame is a method for selecting frames that are grouped in duplicates in one type of patient's command needs. Duplicate frames are frames that show the same command as a minimum of two consecutive frames reading. The number of duplicate frames is accommodated into a variable. Based on the number of frames, a logical process can be carried out to identify the order based on the most frames that appear, whether the command has been executed before, and what kind of conditions so that the system can find out that the patient is submitting an order. The selection frame also serves to tell the system which commands are valid to send to networks that are integrated with IoT. The protocol used for data communication is a web socket where the browser must support HTML5. Another minimal system architecture requirement is of course a Microsoft Kinect V2 sensor and a Local Area Network (LAN) that can be connected to the internet network.

To analyze system performance that implements an integrated IoT selection frame, the system will be tested for valid command accuracy with distance and delay parameters starting from the start of gesture recognition until the patient's need notification is displayed on the browser. Distance testing will determine the optimal distance to read someone's gesture accurately. In distance testing, Kinect will be placed in a position where a person's skeleton can be read from the skeleton point of the tail bone, left / right hand, to the head. System performance is measured based on the time needed to read the type of movement, the accuracy of reading the type of movement, sending data to the internet until it is received by the nurse, and the nurse's response in the form of confirmation to the patient.

The systematic presentation of this research is as follows: Chapter 2 shows previous research that is the reference of this research. Chapter 3 describes the flow of the selection frame method. Chapter 4 explains the integration of systems with IoT. Chapter 5 discusses the workflow of the system in general. Chapter 6 describes the analysis of the results of the system testing that has been carried out. Chapter 7 contains the conclusions of our study.

2. Overview

Kinect Research implemented in the healthcare sector have been widely carried out. Among them is for monitoring an elderly [3], detection of patients who walk outside their beds [4,5], detection of patients who fall [6], detection of patient's sleep position [7], and physiotherapy for patients [8].

Research on the detection of fallen patients using Kinect sensors has been carried out by Kawatsu et al. [6] which uses two kinds of algorithms, namely Position Algorithm and Velocity Algorithm. Velocity
Algorithm has disadvantages in some cases, such as when someone accidentally goes up or down a ladder, and when he walks out of sight of the camera, the person will be detected falling. Unlike the Position Algorithm which is not a problem in the case, but has a deficiency if someone deliberately lies on the floor, the person will be detected falling. Based on these comparisons, Position Algorithm is applied in this study given the level of error that can be tolerated in cases where patients move from their beds or chairs.

Research K F Li et al. [9] utilizes the ability of Kinect to receive input movements to translate American Sign Language (ASL) into a text that will be displayed on a web application for someone who has hearing loss. The ASL movement input will be validated according to the vocabulary or phrase that has been recognized by the system. In this study, there are 11 vocabulary that have been built and can be displayed through the web. The research shows that the method of recognition of movements in Kinect is able to produce text output that can be displayed through the web. In this study, only 3 representations of palm movement commands for patient needs were built.

Gesture Recognition in Kinect V2 does not require patients to wear devices in the form of other sensors on their body that cause problems if the patient forgets to wear them. The Speech Recognition in Kinect V2 is not applied to the work of this research because it is not suitable for detecting falls. Then, it can be concluded also based on the research of M H Tambunan et al. [1] that speech recognition by Kinect V2 with Indonesian has several limitations. Among them is that if someone's distance with the Kinect V2 is farther away, then the system will be more difficult to recognize the sound. The optimal distance to recognize sounds is between 0.5 to 1 meter, more than that distance the accuracy of the system decreases. Therefore, this study focuses on the distance that can detect commands more than 1 meter given the extent of room conditions of patients in hospitals in general. This study did not test the parameters in the form of a person's body structure or the influence of the amount of clothing on the accuracy of the system as already done by Xu et al. [10].

Research F Moreno et al. [11] explained in detail about the mechanism in developing web applications that utilize motion sensor technology with Kinect. Starting from image acquisition, image processing, data transmission (networking), to the display of image information that has been captured by the Kinect sensor on a web application. It was also stated on the site [11,12,13] that officially, the development of web applications that support Kinect sensor integrity can use Windows OS through its browser, namely Internet Explorer (with the IE9 version or the most recent). Other browsers can also be used only if they support HTML5. In addition, there are also minimum specifications of PC / Laptop hardware that will be used in the application development process.

Research A. B. H. Mohamed et al. [14] described that in parallel, in the medical field, helping and watching over people who are sick raises its own attractiveness. Especially if you can facilitate them with sensors that are relatively small and affordable, and do not look striking in the patient's environment. It is also concluded by L Liu et al. [5] that a system that utilizes Kinect V2 is needed in the patient's room due to its noninvasive, contact-free, and low-cost advantages.
3. Selection frame

![Selection frame method flowchart.](image)

Kinect V2 is able to capture up to 30 fps (Frames per Second) at any time. With this number, there are 60 frames in 2 seconds which are based on [15] 8,294,400 bytes per frame multiplied by 60 to 497,664,000 bytes. That is a big value and can overload the server if all unselected frames are sent directly. This size only represents a color frame with a quality of 1080p, while to detect a skeleton requires collaboration between depth frames and infrared frames which have a larger size if accumulated. Therefore, out of these 60 frames, the frame selection method can be applied when the frame flow starts to be captured by the system and represents the results of the skeleton movement detection of someone in front of Kinect V2 into text, so that the system performance becomes optimal.

The number of duplicate frames is accommodated by comparing the results of the movement classification text in the current frame with the previous frame. If the number of duplicate frames has reached 60, then 60 duplicate frames will be represented to become a valid command to be sent to the server. The selection frame logic process doesn't just stop there, if a valid command has been sent to the server and the patient is still detected giving the system command, the system will define a state that the patient has just submitted a valid command. This is done to improve the ability of the system to recognize commands that are really needed by the patient. The frame selection method can be described in the logic flow as in figure 1.

4. Internet of things

![Internet of things.](image)

In figure 2, the first step to sending the patient input results in the form of a valid command text is through User Datagram Protocol (UDP). UDP is suitable for sending broadcast messages to clients quickly. Excess like this is useful especially to notify nurses of the patient's needs in a short time. Especially if the patient is in dire need of nurse assistance immediately, such as an emergency call or
when a fall is detected. For data communication in this study, websocket is used i.e. Socket IO that supports simultaneous communication. Real-time communication is needed so that patients can find out quickly about the nurse's willingness to help. So that if a patient has been treated in a short time, the nurse can then handle other patients in need.

Data on patient needs is then pushed to the cloud, namely Ubidots, via LAN. Ubidots distributes patient needs information to browsers that support HTML5 where nurses can be on standby waiting for real-time notification of patient needs. After the patient's information needs appear in the browser, the nurse can confirm their willingness to handle the patient through the button on the user interface that has been provided. This confirmation of willingness is sent back via the push method to Ubidots, and channeled through the websocket and UDP back to the patient.

5. The built system

The process of input and output of the system can be described through figures 3 and 4 above. It can be noted in figure 3, that the system receives input from the patient's movement in Kinect V2. This system will then process captured image data to produce a text that classifies the type of command / movement of the patient's palm. To be able to give an order, patient needs to raise their hand first, at least parallel to the head. This is done to improve accuracy so that the system can distinguish when patients can give orders, and when they can't. So, if the patient wants to give another command, it is recommended to lower his hand first, then lift it again. Before the command is sent to the server, the system first chooses which data frame is worth sending, and which one is not. In handling this, a frame selection method is applied so that the system can find out which commands the patient really wants and does not overload the server.
Furthermore, the results of the classification of the valid command will be sent to the server via the UDP protocol (User Datagram Protocol) with port 41181. The server will send the command to the client via socket. This client is a browser that supports HTML5. There is only one client (nurse) who can receive command information from the patient. The command information will be displayed through the browser in the form of notification of patient needs. Examples include asking for help to replace an infusion of a patient, help with defecation, and make emergency calls. In addition, this system can also provide notifications to nurses automatically when the patient falls to the floor area. Then, the nurse’s confirmation of each patient's needs will be available through the buttons provided in the pop-up notification. The confirmation from the nurse is used as a back response to be sent to the patient through the voice of the active speaker so that the patient can find out directly what the nurse's willingness is in handling their needs.

Kinect V2 supports hand-tracking to represent commands that you want to define based on palm movements. There are several hand-states that have been provided by Kinect V2 including Open, Close, Lasso, Unknown, and Not Tracked. In this system, Open represents the command "Emergency Call", Close for the command "Help Defecate", and Lasso for the command "Infuse Out". While Unknown and Not Tracked are only an indication that the command is not recognized by Kinect V2 and if the skeleton of a person's body still cannot be captured by the sensor.

For detection of falling patients, this system applies Position Algorithm based on the research of Kawatsu et al. [6]. This algorithm calculates the distance between the floor plane and the skeleton point of the patient's body. In this system, the reference point of the skeleton is the patient's tail bone (Spin Base). If the skeleton point is about the floor area, the system will detect it as "Falling Patient".

Figure 5. Hand command (left) and browser notification (right) capture.

Figure 6. Fall detection (left) and browser notification (right) capture.

System testing was carried out in bright room conditions (1298-2390 lux) and there were no objects that prevented the detection of one's body. Commands are sent to the browser (client) through the local network in the building (figures 5 and 6).
6. Evaluation

6.1. Results of testing and analysis of accuracy. This test scenario is carried out by recording the number of commands that have been detected and classified by the system from 20 experiments that have been carried out on the parameters of distance. The experiment consisted of 5 times the command "Emergency Call" (OPEN), 5 times the command "Help Defecate" (CLOSE), 5 times the command "Infuse Out" (LASSO), and 5 times the detection of "Falling Patients". 5 attempts at each of these commands represent the various positions and movements of the patient, both regularly and arbitrarily. The following is a graph of the test results with parameters in the form of distance.

![Accuracy Based on Distance](image)

**Figure 7.** Accuracy test graph based on distance.

This test is conducted to determine the optimal range of distance between object (patient) and Kinect V2 sensor. The patient first adjusts the position in front of the sensor at a distance of 1.5 meters, which is the closest distance to being able to give commands via the sensor. Every time the interval is 0.5 meters, the patient gives an order of 20 trials as previously mentioned, up to the farthest distance of the sensor detection that is 5 meters. The number of commands that are successfully detected and classified by the system is then calculated, so that the accuracy value is obtained at each distance.

Based on the graph of the results of system testing with distance parameters (figure 7), the highest accuracy value is obtained at 95% when the distance between objects (patients) and sensors in the range of 2.5 meters to 3 meters. In this case, the patient's body is not obstructed by other objects except what he wears, so that the whole skeleton point of the patient's body is detected by the system. In addition to optimizing accuracy, patients are strongly recommended to be right in front of the sensor and not back to back with the sensor or face to the side.

It can also be noted in the graph (figure 7), which ranges from 1.5 meters to 2 meters and at a distance of 3.5 meters, has a fairly good accuracy that is worth 90%. But a distance of 1.5 meters is not recommended to be applied. This is because the range of view of the Kinect V2 sensor to detect all the skeleton points of the patient's body is narrower. So it is feared that some parts of the patient's body cannot be detected by the sensor if the body part is outside the maximum range of view of the Kinect V2. This will greatly affect the ability of the system to detect falls or commands desired by the patient.

At a distance of 4 meters, the value of the accuracy obtained decreased considerably to 80%, especially for the "Infusion Out" commands (LASSO), whereas other commands can still be optimally detected by the system. But at a distance of 4.5 meters, the accuracy obtained dropped dramatically to 30%. During this position, the system begins to have difficulty detecting all the skeleton points of the patient's body, so that the desired command is difficult to convey. Until finally at a distance of 5 meters,
the skeleton of the patient's body cannot be detected by the system, although the Kinect V2 RGB Camera can still capture images of objects in front of it.

6.1.2. Data delivery performance testing and analysis. Data transmission performance testing is carried out based on the length of time since the patient's order was sent (server) until the notification on the browser (client) is displayed. Every command available on this system is tested with a data transmission test scenario done via LAN. Figure 8 contains a graph showing the results.

Based on the graph of the results of data delivery performance testing (figure 8), the lowest delay value is obtained at the "Patient Fall" command of 2.8 ms. This means that detection of falling patients has the fastest data delivery performance to be immediately delivered to nurses through a browser. As for hand-tracking commands such as Open, Close, and Lasso, the data delivery performance is slightly slower than the detection of fallen patients, but is still in the millisecond range which is still a short time to convey patient needs.

7. Conclusion
The application of the frame selection method to gesture recognition can eliminate duplicate frames sent to end users through IoT. Delay generated from this delivery process in this environment is still in millisecond units. The reading of the order has been done correctly, including when there is an incident of a patient falling. Based on the results of system testing, the highest accuracy values obtained were 95% in the range of 2.5 meters to 3 meters between objects (patients) and Kinect V2 sensors. This is the best distance to run the system optimally. The command with the fastest data transmission performance is the detection of "Falling Patients".

References
[1] M H Tambunan, Martin, H Fakhruroja and C Machbub 2018 Indonesian speech recognition grammar using kinect 2.0 for controlling humanoid robot 2018 no 978 pp 59–63
[2] N Abner, K Cooperrider and S Goldin-Meadow 2015 Gesture for linguists: a handy primer linguist 2015 Lang. Compass 9 no 11 pp 437–451
[3] S Khawandi, B Daya and P Chauvet 2011 Implementation of a monitoring system for fall detection in elderly healthcare Procedia Comput. Sci. 3 pp 216–220
[4] L Liu and S Mehrotra 2016 Detecting out-of-bed activities to prevent pneumonia for hospitalized patient using Microsoft Kinect V2 2016 Proc. - 2016 IEEE 1st Int. Conf. Connect Heal Appl.
[5] L Liu and S Mehrotra 2016 Patient walk detection in hospital room using Microsoft Kinect V2 2016 Proc. Annu. Int. Conf. IEEE Eng. Med. Biol. Soc. EMBS 2016–Octob pp 4395–4398

[6] C Kawatsu, J Li and C J Chung 2013 Development of a fall detection system with microsoft kinect 2013 Adv. Intell. Syst. Comput. 208 AISC pp 623–630

[7] L Liu and S Mehrotra 2016 Bed angle detection in hospital room using Microsoft Kinect V2 2016 pp 277–280

[8] O Postolache, F Cary, P S Girao and N Duarte 2016 Physiotherapy assessment based on Kinect and mobile APPs 2016 IISA 2015 - 6th Int. Conf. Information, Intell. Syst. Appl

[9] K F Li, K Lothrop, E Gill and S Lau 2011 A web-based sign language translator using 3D video processing 2011 2011 14th Int. Conf. Network-Based Inf. Syst. pp 356–361

[10] H Xu, Y Yu, Y Zhou, Y Li and S Du 2013 Measuring accurate body parameters of dressed humans with large-scale motion using a Kinect sensor 2013 (Switzerland: Sensors) 13 no 9 pp 11362–11384

[11] F Moreno, E Ramirez, F Sans and R Carmona 2015 An open source framework to manage kinect on the web 2015 Proc. - 2015 41st Lat. Am. Comput. Conf. CLEI 2015

[12] Microsoft Kinect Hardware Requirements and Sensor Setup 2016 [Online] Available: https://developer.microsoft.com/en-us/windows/kinect Accessed: 19-Nov-2017

[13] Microsoft Kinect in the Browser 2016 [Online] Available: http://goo.gl/sGW7E1. Accessed: 19-Nov-2017

[14] A B H Mohamed, T Val, L Andrieux and A Kachouri 2012 Using a Kinect WSN for home monitoring: principle, network and application evaluation 2012 Int. Conf. Wirel. Commun. Undergr. Confin. Areas ICWCUCA 2012 pp 9–13

[15] Geekswithblogs Processing Kinect v2 Color Streams in Parallel 2014 [Online] Available: http://geekswithblogs.net/freestylecoding/archive/2014/08/20/processing-kinect-v2-color-streams-in-parallel.aspx Accessed: 17-Oct-2018