Real-time and Seamless WYSIWYAS Navigation System for Smart Device

Yusuke Takatori [*], Shohei Iwasaki [**], Tatsuya Henmi [**], and Hideya Takeo [*]
[*, **]Kanagawa Institute of Technology, Japan
[*]{takatori,takeo}@ele.kanagawa-it.ac.jp, [**]{s1112063,1112040}@ccy.kanagawa-it.ac.jp

Abstract - This paper proposes a real-time and seamless WYSIWYAS (RS-WYSIWYAS) navigation system for smart devices, which is to be used by a pedestrian while walking on a corridor. WYSIWYAS is a navigation concept that provides a user an intuitive guidance that do not need to interpret provided guidance information. The proposed system is implemented as a prototype application of RS-WYSIWYAS navigation system for Android smart device that has a rear camera. To provide WYSIWYAS navigation for the user, a smart device captures consecutive some M-CubITS marker elements assigned to one line by M-sequence on a corridor from a scenic movie. Then the application obtains the location of the captured marker elements in a building and the heading of the smart device. After that, it shows the user the direction to his/her destination in real time and seamlessly. Moreover, in order for the prototype application to become practical, its response and the performance of marker elements recognition are improved. The response of the application is improved by resizing of the captured movie and using native code programming. At the same time, marker recognition performance is improved by pre-masking of a captured movie. The results of experiments indicate that the response and marker recognition performance are improved to the level that a user could accept for practical use.

Keywords: ITS, Indoor navigation system, WYSIWYAS navigation, M-CubITS

1 INTRODUCTION

Because of the spread of smart devices, the demand for pedestrian navigation systems has increased. Most smart devices use GPS for positioning. However, if these devices are used inside a building or underground areas where an electric wave of the GPS does not reach, it is difficult for such devices to pinpoint their location. In addition, it is known that GPS precision and accuracy decreases if a smart device is used in areas dominated by tall buildings [1]. For these reasons, pedestrian navigation systems not dependent on GPS have attracted research attention. In around 2000, many indoor navigation systems have been developed. Indoor navigation system using wireless technology such as radio wave [2], ultrasonic wave [3], [4] or infrared ray [5] uses position of a transmitter. However these system need installation of transmitters and the user needs a dedicated receiver. Recently, a positioning technique using an electric wave emitted by Wi-Fi access points attracts attention [6]. Though these technique are available with the smartphone which is equipped with Wi-Fi device as standard equipment, some problems are left in the precision for this system because the RSSI (Received Signal Strength Indicator) fingerprinting observed by a receiver often includes some vagueness and an estimated position may not be stable. On the other hand, many positioning technologies that recognize a tag located indoors and pinpoint the location of the terminal have been conducted. Tag-based pedestrian navigation system using ID-tag like RFID or two dimension code have been studied [7]-[10]. In most cases of tag-based navigation system, tags are input their location information, then a mobile device reads these information and recognizes its location, and the acquired location information is pointed on a 2D map. However, the guidance is conducted only at tag-installed places and uses 2D map. That is, the user have to look for the ID-tag and understand his/her location on a 2D map. Besides he has to judge the way to go. Because the operation load of the tag-based navigation systems on a user is high, tag-based 2D navigation system is difficult for persons who cannot read a 2D map well. To realize a more intuitive navigation system, Kurihara proposed an indoor navigation system using augmented reality (AR) marker [11]. The AR marker proposed by Kato [12] is a two-dimensional symbol that allows a digital camera to determine its position and rotation relative to the surface of the marker. A smart device with camera reads the AR marker and overlays an arrow on the AR marker shown in the captured frame. Therefore, this navigation system does not need to interpret a two-dimensional map. However, its navigation concept is still a tag-based navigation. Like fixed guidepost, AR markers are located at each turning point. Therefore, similar to the conventional tag-based navigation systems, a user must perceive the position of the AR markers during use. To reach a destination, the user must first locate the nearest AR marker.

Furthermore, Hasegawa proposed a navigation concept called "WYSIWYAS" (What you see is what you are suggested) [13]-[15]. WYSIWYAS is a fundamental design concept of human-machine interface (HMI) for personalized and intuitive navigation. WYSIWYAS navigation is designed to provide intuitive navigation information to the user without awareness of navigation infrastructures. Then, it overlays a direction arrow to the destination on a scenic image that is captured by the camera device. Additionally, Hasegawa proposed the positioning system M-CubITS, which allows a mobile device with a camera function to determine its position and orientation. This system uses multimodal markers with 1-bit information (0 or 1), which are disposed in a line along a passage in accordance with an m-sequence. When a user takes a picture of a passage that includes a number of disposed M-CubITS markers, the bit sequence of markers is extracted. At this time, the user do not have to consider the arrangement or the position of the markers. Following this, the extracted bit sequence is checked against a database, and the position and the orientation of the mobile device is obtained. By applying the M-CubITS positioning system to the WYSIWYAS navigation concept, intuitive navigation without awareness of navigation infrastructures is achieved. Yamashita and Manabe have built pedestrian navigation systems based on the "WYSIWYAS navigation concept with M-CubITS positioning system" on a mobile phone with a camera device [16]-[18]. Masuda has built navigation systems based on the "WYSIWYAS navigation concept with M-CubITS positioning system" for vehicles [19].

Nevertheless, since each conventional WYSIWYAS navigation application running on a mobile phone captures a still picture [16]-[18], they cannot provide real-time and seamless navigation for a user. They urge a user to stop and to take a still picture when he/she wants to know the direction of the destination. It requires a series of operations (it impairs the real-time response of the system) and it is repeated every time the user wants the navigation information (it impairs the seamless of the system). It is insufficient from the point of view of the design concept of WYSIWYAS (What you see is what you are suggested) because if a mobile device has moved, the current scene is different from the captured image before moving. Further, series of operation (Stopping and taking a still image) is a big burden on the user when he/she uses it in a building that has a lot of branch points or entrances of rooms. This may lead to some problems that effect on the usability of it. First, this could impede other pedestrian traffic because users stay on the passage way. Second, when marker recognition was failed, it's necessary to do the same operation once again. Third, for a person using a wheelchair the photography operation is troublesome. This is because it is not available during movement. Fourth, the intuition of the navigation is spoiled because the conventional system cannot support the posture change of the user. To realize more intuitive and high usability WYSIWYAS navigation, it is necessary to guide the user seamlessly in real-time to remove these problems.

In this paper, a prototype system of a real-time and seamless WYSIWYAS (RS-WYSIWYAS) navigation system that does not require a user to know the direction of the destination has been developed. To achieve real-time and seamless WYSIWYAS navigation, our proposed system processes a captured camera-preview movie, and superimposes the navigation information on it. An Android tablets with rear camera are used for the proposed system. A smart tablet captures a scene movie and extracts consecutive some M-CubITS marker elements (classified by color "red" or "green" according to the bit information) in order to recognize the location of the marker elements and the direction, then overlays an arrow symbol on the screen, indicating the direction of the destination. Meanwhile, because the resolution of the camera becomes higher in recent years, it causes higher image processing time that causes severe application delay. Moreover if there are non-marker objects of similar color to M-cubITS marker elements, the application misunderstands them as marker elements. As a result the correct location of marker elements cannot obtained and the device provide wrong navigation.

In this paper, section 2 explains conventional WYSIWYAS navigation system that offers uses intermittent WYASISYAS navigation. In section 3, the prototype of RS-WYSIWYAS navigation system is described. In section 4, usability improvements of the prototype application are conducted. The application's response is improved by resizing the captured movie (sequential pictures) and using native code programming. Moreover marker recognition performance is improved by applying pre-masking processing to a captured movie. This omits the information of unwanted areas where M-CubITS marker elements do not exist, and extracts marker sequence information. In section 5, this paper is concluded.

2 WYSIWYAS NAVIGATION SYSTEM

2.1 Concept of WYSIWYAS Navigation System

WYSIWYAS is a concept of Human Machine Interface (HMI) for virtual navigation systems [13]-[15]. When users employ smart devices with a WYSIWYAS navigation system, they capture images using the WYSIWYAS navigation application. Then the application displays an arrow symbol that points to the final destination on the captured image. Figure 1 shows a usage scene example of WYSIWYAS navigation system. If a user goes to the direction that is indicated by the arrow symbol, they will get to their destination. That is, a smart device does not need to know the accurate and precise own location information. To implement these features, the WYSIWYAS navigation system is required to recognize the relationship between the destination and location of the captured image, and direction of the smart device.

2.2 M-CubITS

M-CubITS is a positioning scheme that places multimodal marker elements (as shown in Fig. 2) according to M-sequences along a corridor, detects a row of M-CubITS elements with a camera, compares the row with a database, and determines the location and direction of the captured marker elements. M-CubITS marker elements are designated as either "0" or "1" and distinguished by the color of the figure. An example of a marker sequence arrangement is shown in Fig. 3. Marker elements are arranged to one line according to an M-sequence code. The top side of an isosceles triangle points to the forward direction. Because an M-sequence code is generated from the Linear Feedback Shift Register (LFSR) of m-stages with maximum code length 2^m-1, the device can recognize its unique position by observing m-chips (Fig. 3). For instance, if a passageway 1 km long is covered by markers that contain 1 bit of information placed 1 meter apart, the required number of shift register stages is 10 (maximum code length:1023 chips). Therefore, marker positioning is required to capture at least 10 markers in an image.

In the case of Fig. 3, when a smart device captures markers from the second to the fourth, it gets a bit-sequence "110". After that, the smart device finds the same bit-sequence from the row of an output bit of LFSR that is recorded to a map database. Then the smart device understands that location ID 2, 3 and 4 markers come out on the screen. Furthermore the smart device judges the direction where the user advances from the location ID of the present location and the location ID (it has been enrolled in map data) of the destination.

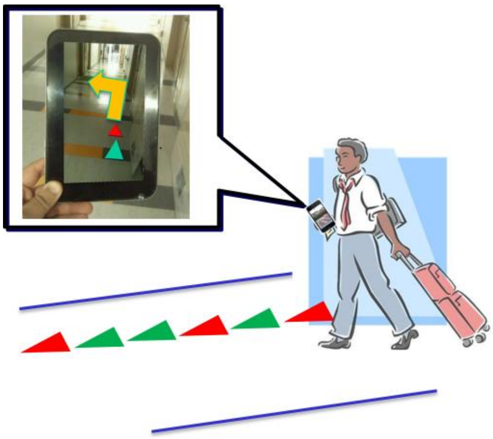


Figure 1: Usage scene example of WYSIWYAS navigation

Red: "1" Green: "0"

Figure 2: Marker elements for M-CubITS

After the marker elements installed into the facility, the location of each marker elements is related to the map data. When renewing guide information, only the map data is updated.

3 REAL-TIME AND SEAMLESS WYSIWYAS NAVIGATION SYSTEM [20]

3.1 Overview of Real Time and Seamless WYSIWYAS Navigation

To indicate the direction to the destination in real-time and seamlessly, the proposed system basically conducts conventional image processing to each frame of a captured movie. That is, multiple images captured successively should be processed in a short period. The device recognizes the sequence of the captured marker elements and its direction, and displays an arrow symbol on the original scenic image. The developing environment of this navigation application is the Eclipse [21] with Android SDK [22]. Besides, some image processing modules use the OpenCV library for the Android [23]. Although the process of marker recognition is based primarily on the scheme in [16], some of processes use techniques different from Ref. [16] in consideration of the processing time.

In this study, we adopted the M-CubITS marker elements shown in Fig. 2. These elements are either red or green; this binary-information is used as chip information for the M-sequences. The shape of the elements is triangular to facilitate recognizing the direction of the M-sequence.

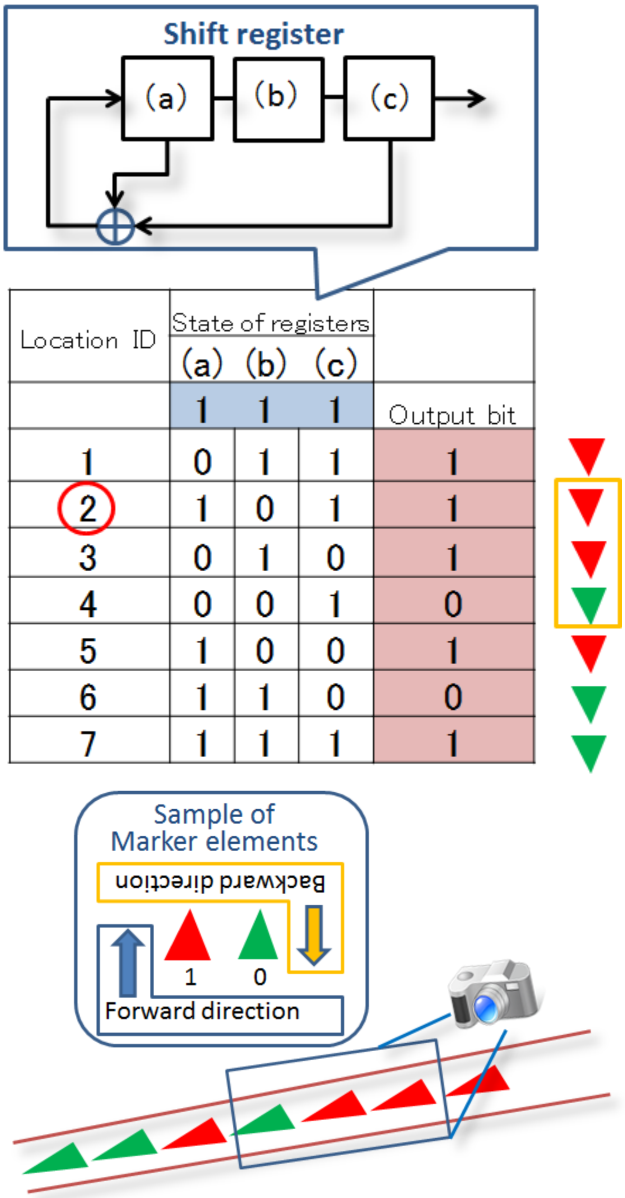


Figure 3: Example of marker sequence arrangement

3.2 Flow of Marker Recognition Process

The process for marker element recognition is as follows:
(1) Real-time image capturing by a rear camera
(2) Particular color extraction

The marker-colored areas that are colored green or red are extracted from the captured image and converted into an HSV (color model) image. In this process, the captured image becomes binary; red or green areas are designated as "1"; otherwise, they are designated as "0." The color-extracted image is shown in Fig. 4 (This image is an 8bit gray scale image, and the brightness value of the marker-colored area is given 255 and the brightness value of other area is given 0). To reduce the noise of the binary image, a smoothing and morphology operation is conducted. An example of a noise-reduced image is shown in Fig. 5.

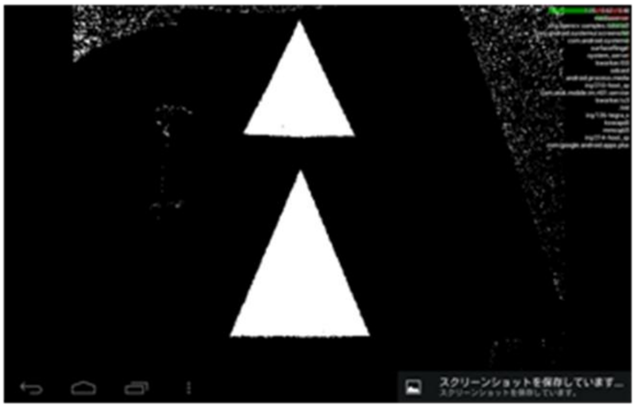


Figure 4: Particular color extracted image

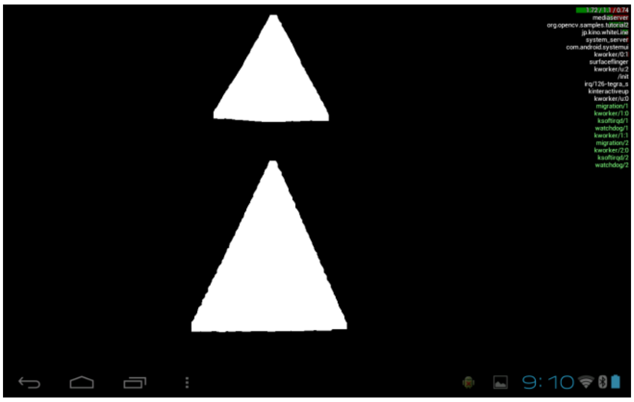


Figure 5: Noise reduction image

(3) Contour extraction

Contours of marker element are extracted and labeled with numbers. The label numbers are assigned in order from the bottom to the top of the image. In the marker element recognition process (described later), marker element information is acquired in a specific order, starting with the marker with the smallest label number. The contour image is shown in Fig. 6.

(4) Marker element recognition

To extract information from a marker element, the centrobaric coordinate of the contour is calculated. After that, a horizontal line that passes through the centrobaric coordinate is drawn, and two intersection coordinates with the contour line are calculated. A horizontal line with intersection coordinates (coordinates A and B) is drawn in Fig. 7. Then, the most common pixel color between coordinates A and B is judged as the marker color.

The direction of the extracted sequence of the marker elements is determined as follows:

- The application calculates the gradient of a line that passes through centrobaric coordinates on the first and second marker elements.
- It calculates two coordinates that intersect the horizontal line that passes through the centrobaric coordinate and contour line of the first marker element.
- By drawing two vertical lines from these intersection coordinates to the line that passes through the centrobaric coordinates, two line segments result.

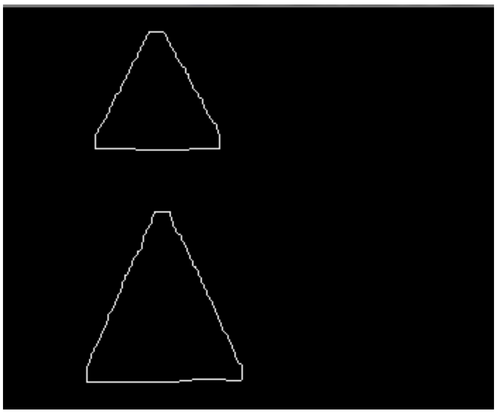


Figure 6: Contour extraction

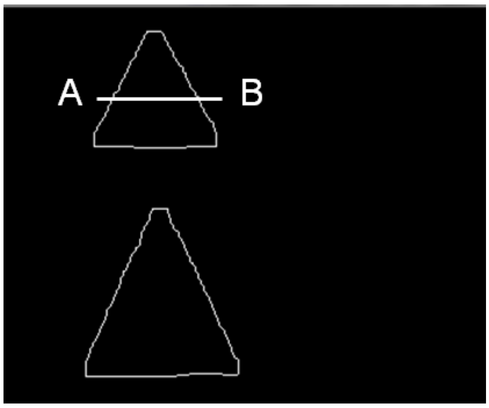


Figure 7: Recognition of marker element color

- These line segments are compared to determine the direction of marker elements. These are shown in Fig. 8. If the gradient of the line is a near right angle, the direction of the marker elements is determined by comparing the number of pixels within the contour line at y = Y_g and y = Y_g-5 (pixel).

(5) Navigation arrow symbol display

Using the marker color and the direction information, the device determines the sequence of marker elements. Then the system judges the direction to go, and overlay a guiding arrow symbol on the original scenic image. If the application cannot extract enough number of marker elements, the sequence of marker elements cannot be specified and arrow symbol is not updated in the processed frame.

3.3 M-CubITS Marker Database

Application of RS-WYSWYAS navigation needs the location of the marker elements that comes out captured movie. To obtain this information, the application references from a marker database. With regard to this system, it is assumed to be used in public or private areas. In public areas, because marker elements are used for public infrastructure, it is thought that a marker database can be used freely through the Internet via mobile networks. On the other hand, in private areas, a building manager might not want to disclose marker data about the building to unauthorized individuals. Moreover, mobile phone might be restricted within such private areas. In these cases, it is assumed that smart-device would access the M-CubITS database via the

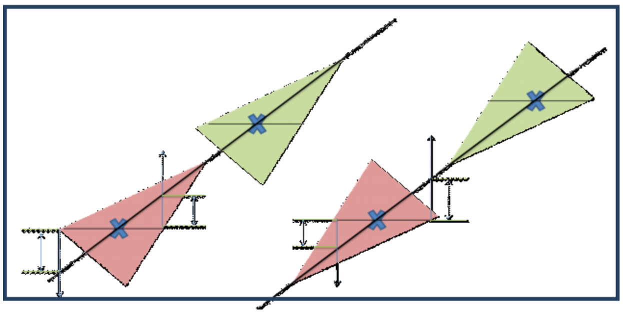


Figure 8: Determining marker direction

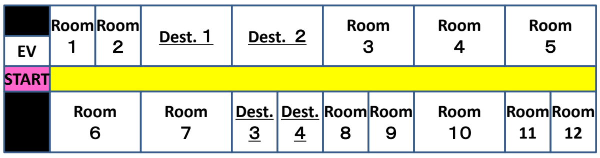


Figure 9: Diagram of experimental area

Table 1: Device specifications
(Toshiba Regza Tablet AT570, 2012)

CPU		NVIDIA® Tegra® 3 Mobile Processor
	Frequency	1.30GHz
	Number of cores	NVIDIA®4-PLUS-1™ Quad core
	Cache memory	1MB
Memory	Capacities	1GB(On board)
Camera		Front camera (1.2MP)
		Rear camera (8MP)

wireless local-private network prepared in the building. In this paper, we constructed an environment of WYSIWYAS navigation system for a corridor in a building. The prototype application performs navigation from the state where the M-CubITS data have already been downloaded in the device, assuming that use of the M-CubITS database has been allowed by the manager of the building.

3.4 Indoor Experiment of Prototype RS-WYSIWYAS Navigation

3.4.1 Overview of Experiment

To evaluate the usability of the proposed system, an M-CubITS WYSIWYAS navigation system was prepared for a building at Kanagawa Institute of Technology. In this experiment, an M-sequence generated from a 7-bit LFSR was adopted, and the marker elements were arranged 0.5 meter apart. A passageway of approximately 60 m was covered, and a smart device recognizes the captured marker positions on the passageway by capturing an image of at least seven marker elements. We made a marker arrangement where the triangle marker element (40 cm × 40 cm and made of cloth) lined up for a corridor (40m length), and have made a database which corresponds to this marker sequence. In this experiment, we used an Android tablet, the Toshiba REGZA Tablet AT570. The specifications of this device are listed in Table 1. The proposed application was installed on the tablet. A subject first selects a destination from among four predetermined locations (classrooms). Then, he/she faces the rear

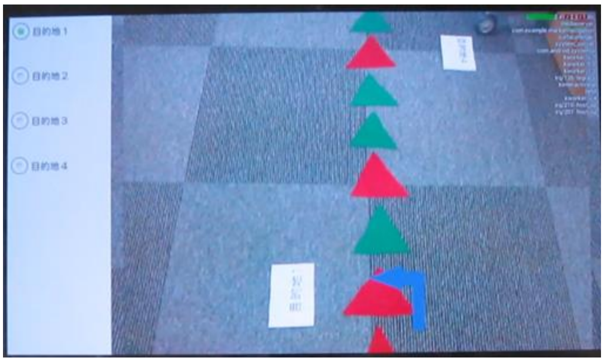


Figure 10: Navigation application user interface

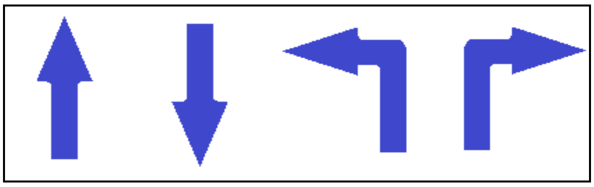


Figure 11: Arrow patterns

camera to the corridor and capture a movie with at least seven marker elements. The application then determines the location of the marker elements, searches for the destination, and displays a guidance arrow symbol on the captured movie to indicate which direction the user should go. In this experiment, because all subjects belongs to Kanagawa Institute of Technology, they might already know the location of each destination room. Therefore, we gave each destination a name that was different from any actual room name. Figure 9 shows a diagram of the experiment area. Figure 10 shows what the user will see on his/her device. There are four buttons on the left-hand side of the screen; touching one selects a destination. These destinations (Dest.1, 2, 3, 4 in Fig. 9) are at almost the same position from the starting point. During navigation, the device indicates the direction of the destination by a guidance arrow symbol; arrow patterns are shown in Fig. 11. For this experiment, a right turn arrow or a left turn arrow points entrance of rooms; it indicates that the subject has arrived at the room of destination. Subjects (12 peoples) conducted the following two experiments in order.

a) Intermittent Navigation (conventional system)

A subject stands at starting point and selects a random destination from the device. Then, the subject turns over the smart device in order to avoid looking at the screen. Next, a staff member shows the subject a diagram (Fig. 9) for a brief time period, permitting the subject to memorize the location of his or her selected destination. The subject then starts looking for the destination without watching the device. If the subject becomes lost, he/she stops, and faces the smart device to nearby marker elements and observes the guidance arrow. Then, the subject turns over the smart device again and continues looking for the destination.

b) RS Navigation (proposed)

The subject stands at starting point again and selects a different destination . He/she faces the camera of the smart device to marker elements and observes the guidance arrow. The subject continues looking for the destination while displaying guidance on the device screen.

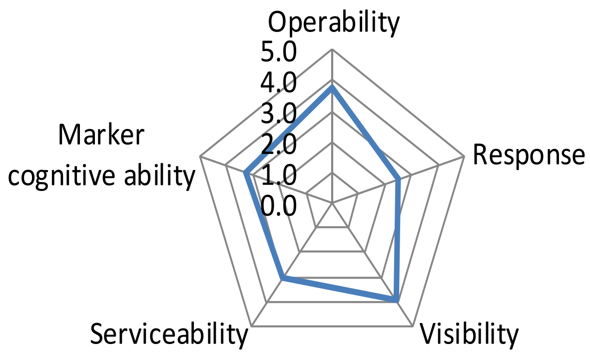


Figure 12: Result of Q1

Table 2: Result of Q2-Q4 and travel time

subjects	Q.2	Q.3	Q.4	Travel Time[s] (intermittent)	Travel Time[s] (RS)
A	Y	Y	RS	362	62
B	Y	N	RS	65	51
C	Y	N	RS	263	85
D	Y	N	intermittent	41	85
E	Y	N	RS	52	40
F	Y	N	RS	170	50
G	N	Y	RS	193	118
H	Y	Y	intermittent	89	115
I	Y	Y	RS	48	65
J	Y	Y	RS	52	39
K	Y	N	RS	278	61
L	Y	Y	intermittent	300	248

During the subject conducts an experiment, travel time from start to destination was measured. After each experiment, subjects answered questionnaires. The questions were as follows.
[Q.1] Please rate the following on a scale of one to five (Very Good: 5, Good: 4, Fair: 3, Poor: 2, Very Poor: 1).
{Operability/ Response/ Visibility/ Serviceability/ Marker cognitive ability}
[Q.2] Is the navigation easy to understand? (yes/ no)
[Q.3] Is the load of the proposed system heavy? (yes/no)
[Q.4] Which application is easier to use? (intermittent / RS)
[Q.5] Did you have any trouble with the applications? (Free writing)

3.4.2 Results

The questionnaire results of Q1 is shown in Fig. 12 and answers of Q2, Q3, Q4 and travel times are listed in table 2. Subjects who answered the free writing questions of Q5 provided the following answers:

- “It is difficult to capture seven marker elements.”
- “This application must improve its ability to recognize marker elements without error, the frame rate of the preview screen and the response speed of the arrow display.”
- “Implementation of an audio based command feature is expected.”

In these answers to the questionnaires, the application is given high marks in operability and visibility. On the other hand, response, marker cognitive ability are rated lower. It is assumed that lack of response and marker cognitive ability lead to low serviceability. Therefore, this application needs to improve the response and marker cognitive ability in order to increase the quality of the navigation system.

Moreover, since most of subjects answered that the navigation is easy to understand, this application achieves an intuitive navigation interface for users. The first experiment was performed using the conventional navigation with which the user is shown the direction to the destination when he takes a picture (intermittent navigation). Meanwhile, the second experiment was performed using the proposed application that offers real-time and seamless navigation. Our questionnaire indicated that most subjects navigated smoother in the RS-navigation experiment than that of intermittent navigation. Therefore, the proposed prototype application that captures a movie and processes them in every frame, providing the user with a real-time intuitive navigation system is more beneficial than the conventional scheme. In both experiments, all subjects arrived at their destination. However, there were some cases when the device could not detect marker elements. When the device cannot detect correct marker elements information, wrong directions are shown to the subject. Although most subjects arrived at the destination in about 40 seconds, some subjects took more than 5 minutes to arrive if they had trouble with marker recognition errors.

4 IMPROVEMENT OF USABILITY

4.1 Response Improvement

4.1.1 Resizing Captured Images

There is a concern that the high processing load would cause navigation because the resolution of current smart devices has become larger, delays. To reduce the influence of the image processing load that depends on image size, the size of the captured image is reduced before the main process.

4.1.2 Implementation of Native Methods

In order to reduce processing time, several methods have been replaced with native methods. In the prototype navigation application that works on the Android OS, most of methods are coded as non-native methods except for some image processing methods using the OpenCV library that provides the native method for image processing. Though non-native methods are used for maintenance and portability in the prototype application, they have to be replaced into native method to improve its response. The marker recognition process performed after each image capture is divided into eight parts, as shown in Fig. 13. Then, the response performance of the prototype RS-WYSIWYAS navigation system was analyzed. The prototype RS-WYSIWYAS navigation application was installed into a smart tablet (Nexus 7 2013 model, because the battery life of the terminal which we used ran out and it has stopped production, substituted smart devices are used), and the average processing time for each process was recorded. The top of the data shown in Fig. 14 is the result of the prototype application analysis. This result shows that a large screen size (1920*1200) costs

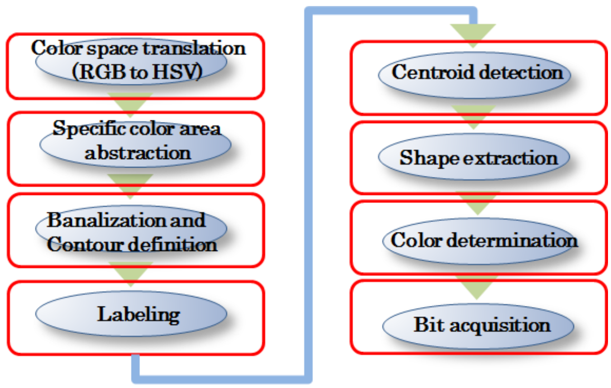


Figure 13: Flow of image processing

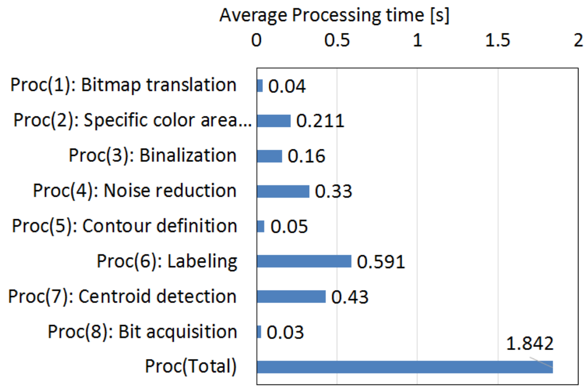


Figure 14: Analysis of processing time of prototype application

approximately 1.8 seconds of the entire processing time. This processing time could result unsatisfactory to users. Moreover, the processing time for labeling process occupies the largest percentage of the total processing time. This process also includes less native library code offered by OpenCV. Therefore, this process is the first candidate of the replacement to native code.

4.1.3 Evaluation Experiment

First, the influence of image resizing is evaluated. The second and third data in Fig. 15 are the results of the marker recognition process with resizing of the captured movie. In these results, the resizing rate of 0.35 shows approximately 0.31 seconds of processing time, which is approximately 1/6 of the prototype application, and this is the shortest processing time among the top three results. This processing time represents 3 Hz of the navigation updating cycle. Recently, most generic GPS receivers for navigation adopt 1 Hz of the positioning updating frequency. Compared with this, the improved application can navigate with higher updating frequency. Meanwhile, although we attempted to test a lower than 0.35 resizing rate, the image processing program cannot output a navigation arrow on the screen constantly because it cannot detect sufficient marker elements from the resized image. This result indicates the trade-off between the marker size and response. That is, though the prototype application has possibility to adopt 0.35 times of size (height: 0.15m, base length: 0.15m) of marker elements, improved application achieves more fast response in exchange for the down-sizing of the marker.

Table 3: Device specifications (Google Nexus 7 2013)

CPU		Snapdragon S4 Pro (APQ8064)
	Frequency	1.5GHz
	Number of cores	quad core
memory		2GB
Camera	Front	120MP
	Rear	500MP
Screen	Size	7inch
	Resolution	1920×1200
OS		Android 4.3

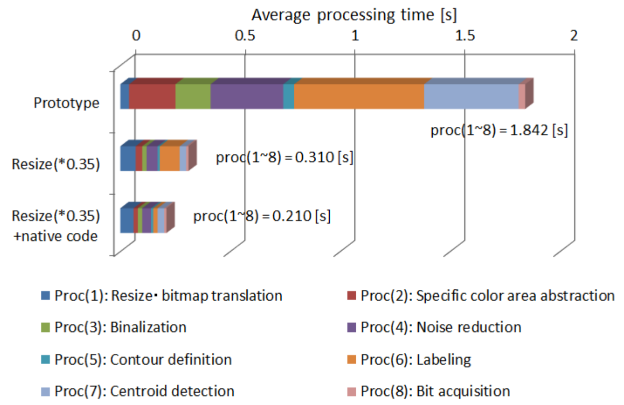


Figure 15: Result of performance evaluation

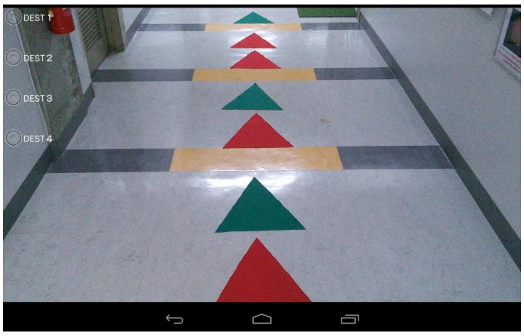
Furthermore, the influence of the replacement of byte codes with native code is evaluated. The results of the processing time of using native codes is shown at the bottom of Fig. 15. This indicates that the labeling processing time is reduced from 0.35s to 0.21s. It also indicates that the total processing time has been reduced to approximately 1/9 of the prototype application; that is, a navigation update of approximately 5 Hz frequency is achieved. Therefore, this prototype system has possibility to show the navigation information to the user by the update interval similar to the case to use GPS.

4.2 Improvement of Marker Element Recognition

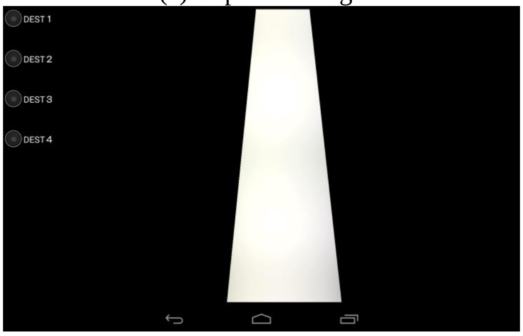
4.2.1 ROI Mask Generation Using Past-extracted Marker Information

Typically, some of objects on the corridor have similar colors of marker elements. For the prototype application, it is difficult to distinguish a non-marker object that has similar colors of marker elements from marker elements placed on the floor. If non-marker objects are extracted and treated as marker elements, the application shows the wrong arrow symbol to the user.

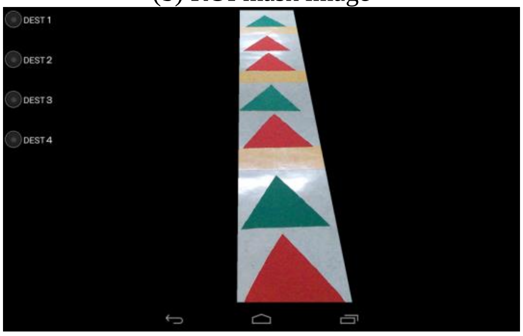
To reduce wrong guidance, pre-masking processing for the captured movie has been implemented. This process allows omitting the information of unwanted areas where M-CubITS marker elements are not placed, and extract only the marker sequence information. To exclude unwanted areas, the marker information extracted in the previous frame is used.



(a) Captured image



(b) ROI mask image



(c) Pre-masked image

Figure 16: Generating pre-masked image

First, we derive a regression line using the centrobaric coordinates of the marker captured previous frame. Next, the Region of Interest (ROI) mask image is generated with a trapezoidal or triangular region along the regression line. In this process, if the regression line crosses the top of the screen, a trapezoidal region is placed along the regression line; meanwhile, if the regression line crosses the side of the screen, a triangular region is allocated along the regression line. Subsequently, a pre-masked image that is used for marker detection is generated by the captured image filtered by the ROI image. Figure 16 shows a generated pre-masked image.

4.2.2 Evaluation Experiment

4.2.2.1 Evaluation Method

To evaluate the improved marker recognition, marker elements are placed on the corridor illustrated in Fig. 9. We used an Android tablet, the Google Nexus 7 (2013 model, in table3). In this corridor, there are some non-marker elements that has similar color of marker elements. Then four types of experiment is conducted.

i) Prototype app. / corridor without non-marker object
ii) Prototype app. / corridor with non-marker object
iii) Improved app. / corridor without non-marker object
iv) Improved app. / corridor with non-marker object

When experiment without non-marker object is conducted, non-marker object that has similar color of marker elements (green and red) are removed or covered with white cloth.

12 subjects conducted 4 types of experiment by following procedure. To take counter balance, half of the subjects (6 persons) conducted prototype application experiments (experiment (i) and (ii)) in the beginning, the rest of subjects (6 persons) conducted the improved application experiments (experiment (iii) and (iv)).

(1) The subject runs the navigation application (prototype or improved) on the smart device.
(2) The subject stands at the starting point.
(3) The subject is told the destination and sets the destination in the application.
(4) The subject searches for the destination according to the navigation instructions.

After subjects finish four experiments, they answer questionnaires. The questions are as follows:

Q.1 Wrong indication time ratio (WITR)

The wrong indication time ratio is defined as a time ratio that perceived as having been presented with the wrong indication. The subject estimates this value from 0% to 100% in step of 10%.

Q.2 Availability

A subject rates the usability of the application through navigation with three choices (fully available/partially available/not available).

Q.3 Acceptable WITR

This is defined as the value of WITR that the user may want to use the application. The subject answers this value from 0% to 100% in step of 10%.

4.2.2.2 Results

Figure 17 shows the WITR in the case where there are few non-marker objects around M-CubITS marker elements on the corridor. In this graph, the WITR using the prototype application distributes broadly. On the other hand, the WITR using the improved navigation application concentrates around 10%. This result indicates that most subjects feel that the improved application provides less wrong indications than the prototype application. Meanwhile, Fig. 18 shows the WITR in the case where there are some non-marker elements on the corridor. In this graph, the distribution of the WITR using the prototype application concentrates around 60% to 100%. It is considered that this degradation of the performance is caused by the marker sequence recognition error. On the other hand, the distribution of the WITR using the improved application concentrates around 0% to 30%. This result is approximately equal to the case where there are some non-marker elements on the corridor. It is indicated that improved application can suppress the influence of surrounding non-marker objects.

Next, the results from questionnaire Q.2 are shown in Table 4. In this question, subjects choose availability of this system. For the prototype system, there are few users judging it to be available enough. Especially, in the case where

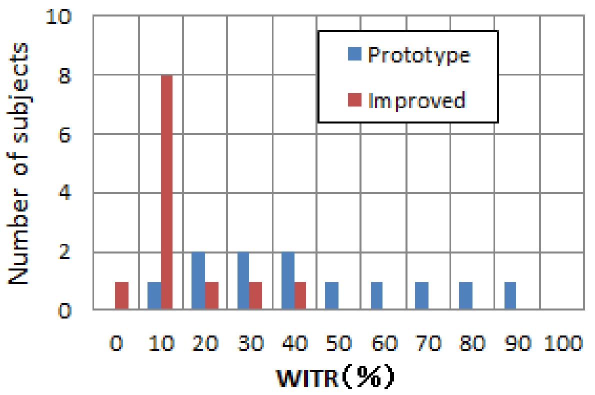


Figure 17: Wrong indication time ratio
(There are few non-marker objects)

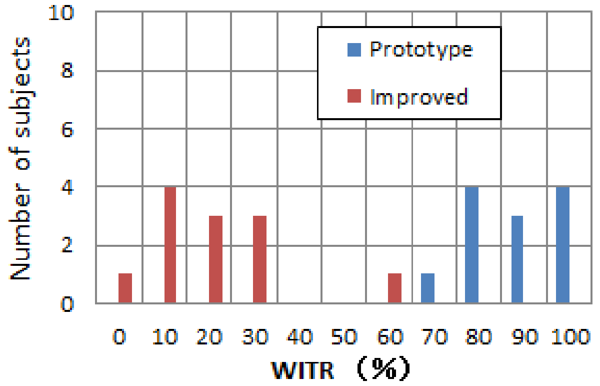


Figure 18: Wrong indication time ratio
(There are some non-marker objects)

Table 4: Availability

	Fully available	Partially available	Little available
Prototype	2	4	6
Prototype (some non-marker object)	0	2	10
Improved	5	4	3
Improved (some non-marker object)	2	7	3

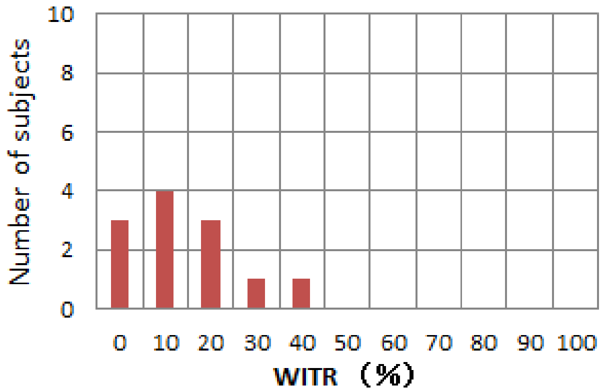


Figure 19: Acceptable WITR

there are some non-marker elements on the corridor, most users feel that the system is insufficient. On the other hand, for the improved system, the availability of subjects is increased. Whether there are non-marker objects or not, 75% of the subjects accept this improved application. However, three subjects feel that the improved application is not acceptable for navigation. This system is effective if it can extract correct marker information in previous frame. However if it cannot extract, it perform same processing as the prototype application. Because the user can recognize a marker on the screen, a function to input a centrobaric coordinates of extracted marker element into the application is expected.

Finally, the result from questionnaire Q.3 is shown in Fig. 19. In this graph, the acceptable WITR is distributed around 10%, and this result shows the same tendency as the time ratio perceived by subjects. On the other hand, three subjects answered that 0% of WITR is required, which corresponds to the results from questionnaire Q.2. Although the improved application reduces marker sequence recognition error, some subjects are not satisfied. This result will be an indicator for usability improvements.

5 CONCLUSION

In this paper, a real-time and seamless WYSIWYAS pedestrian navigation system for smart devices was proposed. In order to provide a user with real-time seamless navigation, the proposed navigation application recognizes consecutive some M-CubITS marker elements on every frame of captured movie, estimates its location and direction, and shows a user the direction that he wants to go. We compared the performance of a prototype of RS-WYSIWYAS navigation application with conventional WYSIWYAS navigation application through navigation experiments in an indoor corridor in which M-CubITS marker elements were arranged. The results of these experiments indicated that the proposed application was evaluated more intuitive and intelligible than the conventional intermittent navigation. Nevertheless, the recognition performance of marker elements and response time of the application were not highly evaluated.

Therefore, in order for the prototype application to become practical, the response and the marker elements recognition performance are improved. From the perspective of the response time, resizing captured images reduced the computation load. In addition, the implementation of native methods also reduced processing time. The results of processing time measurements indicated that a navigation updating cycle of approximately 5 Hz was achieved.

Meanwhile, from the perspective of marker elements recognition, pre-mask processing of captured images using the marker information obtained from a previous frame improved the marker elements recognition performance. The results of questionnaires for the navigation experiments indicated that the improved navigation application reduces the WITR compared with the prototype application. In addition, most subjects considered that the availability of the improved application is higher than the prototype application.

In this study we conducted experiments on a single floor indoor environment. However, generally a building with complicated structure needs the guidance of the user. Therefore, implementation methodology to complex buildings (multiple layer, crossing, etc.) is required. Moreover, to improve the marker recognition performance at the beginning of the navigation will be needed.

6 REFERENCES

[1] J. Soubielle, I. Fijalkow, P. Duvaut, and A. Bibaut, “GPS positioning in a multipath environment,” IEEE Trans. Signal Process. vol. 50, no. 1, pp. 141-150 (2002).

[2] Y. Chen and H. Kobayashi, “Signal Strength Based Indoor Geolocation,” in Proc. IEEE ICC, pp. 436-439 (2002).

[3] J. Moriya and T. Hasegawa “A study of Inverse GPS based positioning system,” Proc. of IEEE ITSC 2004, pp.559-563 (2004).

[4] T. Hada, H. Sunaga, M. Akiyama, S. Ioroi, and H. Tanaka, “Investigation and Demonstration of Local Positioning System using Ultrasonic Sensors for Wide Indoor Areas,” International Conference on Ambient Intelligence 2010, pp.280-284 (2010).

[5] A. Ward, A. Jones and A. Hopper, “A New Location Technique for the Active Office,” IEEE Personal Communications, Vol. 4, No. 5, pp. 42-47 (1997).

[6] K. Kaemarungsi and P. Krishnamurthy, “Modeling of indoor positioning systems based on location fingerprinting,” INFOCOM 2004. Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies, vol.2 Volume: 2, pp. 1012-1022 (2004).

[7] J. Rekimoto and K. Nagao, “The world through computer,” Proceeding of the ACM Symposium on User Interface Software and Technology (UIST95), pp.29-36 (1995).

[8] I. Siio, T. Masui, and K. Fukuchi “Real-world Interaction using the FieldMouse,” CHI Letters, Vol.1, Issue1, pp.113-119 (1999).

[9] I. Siio, “User Position Detection using RFID Tags,” IPSJ-Technical Report 88, pp.45-50 (2000).

[10] H. Matsubara, N. Fukasawa, K. Goto, K. Kawai, T. Sato, T. Aoki, N. Mizukami, K. Fujinami, and A. Shinomiya, “Interactive Guidance System for Passengers,” QR of RTRI 42(4), pp.201-206 (2001).

[11] T. Hasegawa, “A study on ITS and systems innovation,” Technical Report of IEICE, no. ITS 2002-120, pp. 13-17 (2003) (in Japanese).

[12] T. Hasegawa, “ITS platform EUPITS–approach to realization,” Technical Report of IEICE, no. ITS 2003-8, pp. 41-47 (2003) (in Japanese).

[13] K. Kurihara and F. Sato, “Marker Based Indoor Navigation System Sharing Location Information,” DICOMO2014 Symposium, pp.1099-1103 (2014) (in Japanese).

[14] H. Kato and M. Billinghurst, “Marker Tracking and HMD Calibration for a video-based Augmented Reality Conferencing System,” Proc. of the 2nd International Workshop on Augmented Reality, pp.85-94 (1999).

[15] T. Hasegawa, “ITS platform EUPITS–toward realization,” Technical Report of IEICE, no. ITS 2003-26, pp. 29-35 (2003) (in Japanese).

[16] S. Yamashita, and T. Hasegawa, “On the M-CubITS pedestrian navigation system by a camera-equipped mobile phone,” Proc. of ITSC2004, pp.714-717 (2004).

[17] T. Manabe, S. Yamashita, and T. Hasegawa, “On the M-CubITS pedestrian navigation system,” Proc. 9th IEEE Int. Conf. on ITS, pp. 793-798 (2006).

[18] T. Manabe, T. Hasegawa, Y. Matsuoka, S. Furukawa, and A. Fukuda, “On the M-CubITS pedestrian WYSIWYAS navigation using tile carpets,” Proc. 10th IEEE Int. Conf. on ITS, pp. 879-884 (2007).

[19] R Masuda, J Kim, T Hasegawa, “On Direction Position Oriented M-CubITS Vehicle WYSIWYAS Navigation,” IEICE Trans. Vol. J91-A,No.1,pp.11-20 (2008) (in Japanese).

[20] Y. Takatori and H. Takeo, “A real-time seamless WYSIWYAS navigation system for smart devices,” Proc. of ITST2013, pp. 163-168 (2013).

[21] Eclipse, http://www.eclipse.org/

[22] Android SDK, http://developer.android.com/index.html

[23] OpenCV library for Android, http://opencv.org/platforms/android.html

(Received September 30, 2015)
(Revised December 26, 2015)

Yusuke Takatori received his B.E. and M.E. degrees in Electrical and Electronic Systems Engineering from Saitama University, Japan in 2002 and 2004 respectively. He also received a Ph.D. degree in Information and Mathematical Science from Saitama University, Japan in 2007. From 2007 to 2012, he was an Assistant Professor in the Department of Management Science, Faculty of Engineering, and Tokyo University of Science. From 2012 to 2015, he was an Assistant Professor in the Department of Electrical and Electronic Engineering, Kanagawa Institute of Technology. Since 2015, he has been an Associate Professor. His research interests are in the area of Intelligent Transport Systems (ITS). Dr. Takatori is a member of IEEE and IEICE.

Shohei Iwasaki received his B.E. degrees in Electrical and Electronic Systems Engineering from Kanagawa Institute of Technology, Japan in 2015. His research interests include image processing for Intelligent Transport Systems. He is currently working at Daito electron co., ltd.

Tatsuya Henmi received his B.E. degrees in Electrical and Electronic Systems Engineering from Kanagawa Institute of Technology, Japan in 2015. His research interests include image processing for Intelligent Transport Systems. He is currently working at Rokko & Associates, Inc.

Hideya Takeo received B.S. and M.S. degrees from Kanagawa University, Japan in 1984 and 1986 respectively. He also received Ph.D. degree on bio-applications and systems engineering from Tokyo University of Agriculture and Technology, Japan in 2005. From 1986, He worked Fuji Photo Film Co., Ltd. and engaged in research on Medical Imaging Technology. From 2006 to 2009, he was an Associate Professor, and From 2009, he has been a Professor in the Department of Electrical and Electronic Engineering, Kanagawa Institute of Technology. His research interests are in the area of Image processing. He is a member of ITE (The Institute of Image Information and Television Engineers), Jamit(The Japanese society of Medical. Imaging Technology), IIEEJ (The Institute of Image Electronics Engineers of Japan), and MII (Medical Imaging and Information Sciences).