INDOOR AUGMENTED REALITY-BASED MULTI-USER NAVIGATION FOR MERCHANDISE SHOPPING USING OMNI-CAMERAS[†]

[1]*Shu-Lin Yang* (楊舒琳), [2]*Bao-Shuh P. Lin* (林寶樹) *and* [3]*Wen-Hsiang Tsai* (蔡文祥)

[1]Institute of Computer Science and Engineering
[2]Intelligent Information & Communications Research Center
[3]Department of Computer Science
National Chiao Tung University, Hsinchu, Taiwan
Emails: pplinlin@gmail.com, bplin@mail.nctu.edu.tw, whtsai@cis.nctu.edu.tw

ABSTRACT

An indoor multi-user navigation system for the purpose of merchandise shopping based on augmented reality (AR) and computer vision techniques by the use of a mobile device like an HTC Flyer is proposed. An indoor vision infra-structure is set up by attaching multiple fisheye cameras on the ceiling of an indoor environment. For multi-user identification and localization, a vision-based method is proposed, which analyzes images captured from the fisheye cameras to detect multi-color edge marks mounted on mobile devices and movements of multiple users. For AR-based merchandise guidance, the server-side system analyzes the image of merchandise items captured from the mobile-device camera by the SURF algorithm and matches the resulting features against a pre-constructed merchandise image database. Finally, navigation and merchandise information is overlaid onto the images shown on mobile–device screens. Good experimental results show the feasibility of the proposed methods.

Keywords: indoor navigation, augmented reality, user localization, omni-vision, fisheye camera, mobile device.

1. INTRODUCTION

People often encounter an annoying problem in a market place where very few salespersons are roaming among the aisles to help merchandise search works. In such a case, people need a more intuitive navigation system to guide them to get to desired aisles and merchandise items. In this study, it is desired to design an indoor navigation system for merchandise shopping and other similar activities based on *augmented reality* (AR) and *computer vision* (CV) techniques using mobile devices as display tools, as illustrated by Fig. 1.

With augmented reality (AR) techniques, artificial information about an environment and objects in it can be overlaid on an image of the real world acquired by the mobile-device camera. This enhances the real-world image with virtual objects or digital information for convenient inspection on the mobile-device screen.
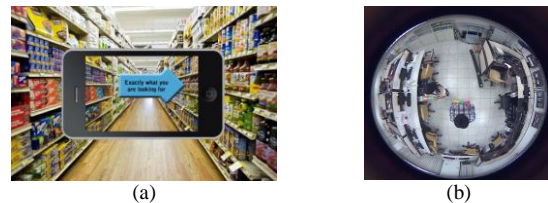

Fig. 1: Proposed AR-based indoor navigation system using computer vision techniques.

Miyashita, et al. [2] designed a museum guide system which uses a markerless tracking technique and an AR platform – Unifeye SDK. Jongbae and Heesung [3] proposed a vision-based indoor navigation system which recognizes the location of a user by applying marker detection and image-sequence matching on images captured with a wearable camera, and displaying relevant navigation information in the AR way. We try to develop an AR system for merchandise shopping and similar activities in this study. There exist other systems using AR techniques for the same purpose. For example, Chen, et al. [4] designed a mobile AR system for retrieving the information about a book by recognizing the book spine without taking it off the bookshelf.

More specifically, we propose a new AR-based navigation system for multiple users using a mobile device for indoor environments in this study. The images captured from a mobile device are *augmented* by the information of respective merchandise names, the planned path to a desired item, or other types of guiding comments, as illustrated in Fig. 1(a). Such a type of AR-based guidance is accomplished by a user localization technique implemented by analyzing images taken from the fisheye cameras affixed on the ceiling in the navigation environment as illustrated in Fig. 1(b) as well as a merchandise recognition technique implemented by analyzing images taken from mobile devices. A brief description of the proposed system is as follows.

At first, a sufficient number of fisheye cameras are installed on the ceiling in the environment. Then, an environment map model is created, which includes the information to be used in the navigation stage. Also, while the images captured with the fisheye cameras are analyzed by a server system to detect the locations and orientations of the multiple users in the environment. Furthermore, for merchandise shopping, the server-side system analyzes the images captured with the camera on the client system which is the mobile device. The server extracts the features of these images with an SURF algorithm [1] and matches them against a pre-constructed database of merchandise features to obtain the merchandise information if the recognition is successful.

Finally, the server system sends the obtained navigation and merchandise information to the client, namely, the mobile device held by the user. Such information then is overlaid onto the real tour sites or merchandise items shown in the current image taken of the environment by the mobile-device camera. In other words, the device displays the navigation information in an AR way.

In summary, the goal of this study is to develop an indoor navigation system with the following capabilities: (1) working in indoor environments, and being able to detect locations and orientations of multiple users; (2) planning a proper path from the user's location to the destination of a desired merchandise item, and updating the path dynamically when the user moves to a location not in the path; (3) integrating real images with virtual augmentations to provide users convenient navigation and merchandise information.

The details of the proposed AR-based indoor navigation system will be introduced in the following, with the configuration of the proposed system and the system process described in Section 2, the process of "learning" an indoor environment in Section 3, a user identification method in Section 4, a newly-proposed multi-user localization method in Section 5, the AR process for guidance of merchandise shopping and a merchandise recognition method in Section 6, the proposed AR techniques for merchandise information overlaying in Section 7, and some experimental results in Section 8, followed by conclusions in the last section.

2. SYSTEM DESIGN AND PROCESSES

2.1 System Design

As shown in Fig. 2, the proposed system is constructed to be of a client-server structure. The server system runs on a virtual machine (VM) of the cloud server, and is connected to the fisheye cameras on the ceiling through an Ethernet. The server sends the navigation and merchandise information to the client system running on the user-held mobile device. On the other hand, when a user enters the environment, the user's client-side system on the mobile device is connected to

the cloud server through a 4G/LTE network, and sends the images captured by the mobile-device camera to the cloud server; meanwhile, it receives relevant information from the server and displays the augmentation information on the mobile-device screen.

2.2 Learning Process

The system operations include two stages — a *learning* process and a *navigation* process. The goal of the learning process is to establish an *environment map*, which includes information about the places available for users to visit, the fish-eye cameras, the merchandise patterns, the magnetic fields, and the obstacle orientations in the environment. The learning process can be decomposed into three phases: learning for path planning, learning for user localization, and learning for merchandise recognition. The former two phases of learning are based on Hsieh and Tsai [5]. And in the process of learning for merchandise recognition, we construct the merchandise patterns, including feature extraction and merchandise information. More details will be described in subsequent sections.
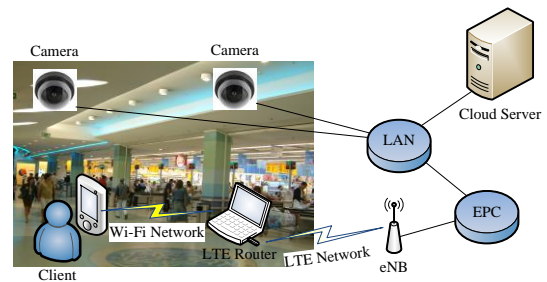

Fig. 2: Network architecture of proposed system.

2.3 Navigation Process

In the navigation stage, the server analyzes the omni-images captured with the fisheye cameras continuously, and sends the analyzed environment information to the client. Meanwhile, the server analyzes the images received from the client, and sends the analyzed merchandise information to the client. The client-side system displays the information on the mobile-device screen. When the user wants to reach a certain destination or merchandise product, the server plans a path consisting of a set of intermediate points, and sends it to the client. Then, the client system displays it on the mobile-device screen as well.

3. LEARNING OF ENVIRONMENTS

3.1 Construction of Environment Map

The environment map is created from an image of the floor plan drawing of the environment, which we call the *floor plan image*, like the example shown in Fig. 3. The environment map includes information about the cameras, target places for visits, merchandise items, magnetic fields, and obstacles in the environment. We can use such

information in the processes conducted in the navigation stage, such as user localization, path planning, and merchandise recognition.

Fig. 3: Floor plan image of experimental environment.

3.2 Camera Calibration

Before the proposed system can be started, the cameras to be used should be calibrated. The process includes two parts: fish-eye camera calibration and mobile-device camera calibration. For the former, we adopt a space-mapping method proposed by Hsieh and Tsai [5] for transformations between the *global coordinate system* (GCS) and the *fisheye-camera image coordinate system* (FICS), using a box as the calibration target as shown in Fig. 4(a).
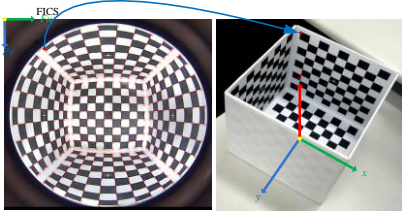

Fig. 4: Space-mapping with a calibration box.

To calibrate the mobile-device camera, a perspective camera model is used [6] by which a 3D space point with coordinates (p_x, p_y, p_z) in a symmetric view frustum as illustrated in Fig. 5(a) in the *camera coordinate system* (CCS) can be transformed into an image point with coordinates (M_u, M_v) in the *mobile-device-camera image coordinate system* (MICS) by the following way:

$$\begin{pmatrix} p'_x \\ p'_y \\ p'_z \\ p'_w \end{pmatrix} = \begin{pmatrix} \frac{h}{w}\cot\frac{\alpha}{2} & 0 & 0 & 0 \\ 0 & \cot\frac{\alpha}{2} & 0 & 0 \\ 0 & 0 & \frac{-(f+n)}{f-n} & \frac{-2fn}{f-n} \\ 0 & 0 & -1 & 0 \end{pmatrix} \begin{pmatrix} p_x \\ p_y \\ p_z \\ 1 \end{pmatrix}; \quad (1)$$

$$\begin{pmatrix} M_u \\ M_v \\ M_z \end{pmatrix} = \begin{pmatrix} w & 0 & 0 \\ 0 & h & 0 \\ 0 & 0 & 1 \end{pmatrix} \begin{pmatrix} 0.5 & 0 & 0 & 0.5 \\ 0 & -0.5 & 0 & 0.5 \\ 0 & 0 & 0.5 & 0.5 \end{pmatrix} \cdot \begin{pmatrix} p'_x/p'_w \\ p'_y/p'_w \\ p'_z/p'_w \\ 1 \end{pmatrix} \quad (2)$$

where α is the angle of the field of view of the frustum in the *y* direction; *w* and *h* are the image width and height, respectively; M_z is the distance of the image plane with respect to the CCS; and *n* and *f* are respectively the nearest and farthest visible distances of the view frustum, which do not affect the mapped image coordinates in the MICS in (3). Therefore, only one unknown variable α

need be calibrated. We adopt a method from [5] which uses a calibration board to estimate the unknown variable α as illustrated in Fig. 5(b).
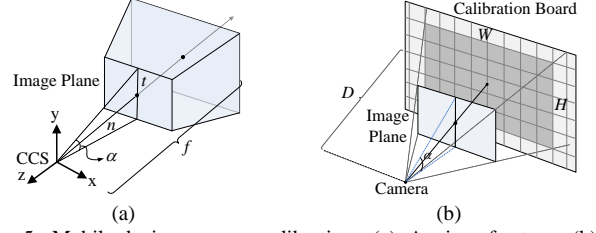

(a)   (b)

Fig. 5: Mobile-device camera calibration. (a) A view frustum. (b) Calibration of the field-of-view angle.

4. USER IDENTIFICATION BY COLOR IMAGE ANALYSIS USING MULTICOLOR EDGE MARKS ON TOP OF CLIENT DEVICES

4.1 Multicolor Edge Mark Detection

The first work in the navigation process is user identification so that the system can localize individual user in the environment. In this study, we propose a method to identify multiple users by the way of attaching a multicolor edge mark on the top of the mobile device held by each user. The material of the edge mark is selected in such a way that the colors on the mark have high saturation and high lightness, and so appear to be prominent in the acquired omni-image. Consequently, the color portions in the image can be segmented out easily. For example, as shown in Fig. 6, we can see the yellow-green-pink edge mark very clearly in the omni-image.

Fig. 6: A clear multicolor edge mark (the yellow-green-pink strip) in the acquired omni-image.

In order to segment the multicolor edge mark out of an omni-image, at first we convert the color space of the omni-image from the RGB space to the HSV one. The HSV color model assigns three color components into a pixel, which respectively are *hue*, *saturation*, and *value*. The hue component may be described with the words we normally think of colors: red, blue, green, etc.; the saturation component refers to the dominance of hue in the color; and the value component indicates the lightness of the color. By the use of the HSV model, we can separate the multicolor edge mark from the omni-image more easily, because the colors of the multicolor edge mark have high saturation and high lightness. In addition, we assume that there are three colors on each multicolor edge mark. We detect and classify them to obtain the

identification number for each user. The classification scheme will be introduced in the next section. In addition, the multicolor edge mark becomes a strip shape in the omni-image, so we can use it to detect as well the orientation of the user (i.e., the direction to which the user is facing). The proposed scheme for user orientation detection will be described in Section 5.

4.2 Multicolor Edge Mark Classification

We detect the color regions in each omni-image taken by a fisheye camera on the ceiling and obtain the three middle points of the color regions. The combination of the three colors is then mapped into different user identification numbers by a pattern classification scheme as described now. First, we construct a mapping table between the three detected color regions and the user identification numbers. In theory, we have $3^3 = 27$ user identification numbers. But we have to consider two "ambiguous" cases which can appear on the edge mark: 1) the three colors on an edge mark are *left-right symmetric* to those on another (e.g., the two color combinations yellow-green-pink and pink-green-yellow) — If the colors of two edge marks are of this case, then when they appear in an image *simultaneously* and *face symmetrically* to each other, then they cannot be differentiated as two distinct marks; 2) more than one identical color is *neighboring* in the mark (e.g., the combination yellow-yellow-pink) — Neighboring colors on the edge mark will be extracted by image processing to be just a *single* region in this study, so the edge mark will be regarded to consist of less than three colors. For example, the combination of the seemingly three colors, yellow-yellow-pink, will be considered to be just a combination of two colors, yellow and pink, only.

Therefore, only nine *effective* combinations of colors are left, instead of 27, for use in user identification when three colors are used on the edge mark, as can be figured out and seen from the mapping table shown in Table 1. Of course, we may use a larger number of colors (larger than 3) on the edge mark to increase the number of effective combinations for identifying more persons. For example, when four distinct colors are used, then it can be figured out that twenty-two effective combinations of them can be used for user identification.

Table 1: Mapping table of combinations of three colors and identification numbers.

ID	1	2	3	4	5	6	7	8	9	-
	GGG	YYY	PPP	YYG	PYG	YYP	PGY	PYG	YPG	PYP
				YGG	PGG	YPP	YGP	GYP	GPY	PGP
				GYY	GPY	PPY				CYG
				GGY	GGP	PYY				GPG
										YGY
										YGY

4.3 Technique for Classification Error Reduction

The proposed user identification method described above has a stability problem which often causes failures in identifying users in our experiment. Therefore, we propose further some techniques to reduce the errors of classification. Specifically, we construct a record to remember the last five results of the identification numbers yielded by the previously-proposed method. If the last five identification numbers are all the same, meaning that the person in question has been stably identified already, then the newly-yielded identification number will be discarded.

In addition, sometimes it might happen that not all the three colors on a multicolor edge mark are detected successfully because the multicolor edge mark might be blocked accidentally by the user's body. To solve this problem, we define a priority sequence to different multicolor edge marks as: the three-color edge mark, two-color edge mark, and one-color edge mark, and use the defined sequence in correcting erroneous classification results. For example, if we detect three colors on a multi-color edge mark, which was classified as a two-color edge mark in the previous image frames, then it will be corrected to be a three-color edge mark in the current frame according to the defined priority.

As a summary of the above discussion on user identification, given a sequence of user identification numbers n_1, n_2, n_3, ..., n_n, we process them into a sequence of corrected user identification numbers n_1', n_2', n_3', ..., n_n' by the following steps: 1) put n_1, n_2, n_3, n_4, n_5 in a record R and set $n_i' = n_i$ for $i = 1, 2, …, 5$; 2) for $i > 5$, if $n_{i-1} = n_{i-2} =…= n_{i-5}$ and $priority(n_{i-1}) > priority(n_i)$, then set $n_i' = n_{i-1}$ to keep the currently-detected identification number; else, set $n_i' = n_{i-1}$ to correct the identification number; 3) put n_i' in R and delete n_{i-5} from R if the number of elements in R is larger than five; 4) repeat Steps 2) and 3) until the last user identification number in the input sequence is processed.

5. MULTI-USER LOCALIZATION IN INDOOR ENVIRONMENTS BY COMPUTER VISION TECHNIQUES

5.1 Multi-user Location Detection

The first step of the proposed multi-user location detection method is background/foreground separation. For this, we capture a background image before running the system. Then, when users enter the environment, they are considered as foreground regions in the image, and detected by connected component labeling. Next, in each found region which is presumably the shape of a user's body, we try to find a "foot point" in the region, which is defined to be the region point closest to the image center. Under the assumption that each user is standing on the ground all the time, this foot point can be used to determine the user' location in the real-world space

because accordingly the axis of each user's body will be perpendicular to the ground so that the axis of his/her body will go through the image center. Consequently, we can detect the users' foot points using this property, and transform them into the GCS as the result of user localization. An example of the results is shown in Fig. .
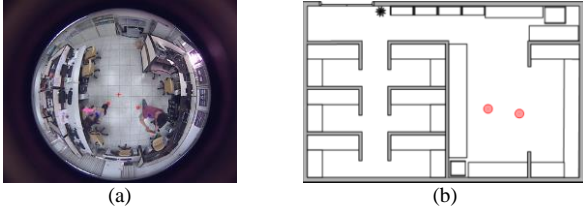

(a)   (b)

Fig. 7: Detected foot points of two users (shown as red circles). (a) The original image captured from the camera with foot points detected. (b) The corresponding positions of the foot points in the environment map.

5.2 User Viewing Orientation Detection

Based on the ideas about user orientation detection described in [5], we propose a method for detection of each user's viewing orientation in this study. In [5], three techniques have been proposed for uses in different situations – *human motion estimation*, *magnetic-field sensing*, and *multicolor edge mark detection*. By using the first technique, the positions of the users are detected at first as described previously in Section 5.1. The results are used next to compute some motion vectors, and the average directions of the vectors then are taken to be the orientations of the users. In addition, we average further the motion vectors computed of five consecutive image frames to get more stable results; and determine whether the users are turning to speed up the work of motion averaging – if a turning is detected, then averaging is conducted by the use of less frames.

In the magnetic-field sensing technique, we utilize an azimuth map which has been established in advance in the learning stage to obtain the desired orientation of each user by interpolation using the magnetic-field data sensed by the built-in magnetometer in the mobile device and the learned magnetic-field data recorded in the azimuth map.

In the multicolor edge mark detection technique, a multicolor edge mark, which is detected as described in Section 4, is used for user-orientation detection. Specifically, we compute the direction vector of the approximating line going through the detected multicolor edge mark for use as the mobile-device orientation which is perpendicular to the user orientation. In more detail, under the assumption that the user holds the device horizontally, the edge mark becomes parallel to the ground. And as shown by the example in Fig. 8, the color edge mark is represented as a yellow-green-pink line, and the red line and the yellow-green-pink line are projected onto identical image points; meanwhile, the vertical projection (shown as the dotted green line) of the multicolor edge mark is parallel to the red line. Therefore, we can determine the orientation of the color edge mark by the orientation of the red line.

6. AR-BASED GUIDANCE FOR MERCHANDISE SHOPPING

In order to provide a more convenient and easy-to-use interface for merchandise shopping and other similar activities, we have to recognize merchandise items and then augment corresponding information on the client-device screen or planned paths to guide the users to the locations of the desired items. More details are described in the following.
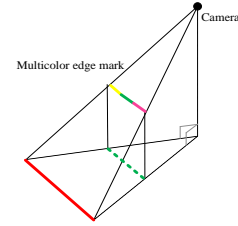

Fig. 8: The red line and the yellow-green-pink multicolor edge mark are projected onto identical image points; and the vertical dotted green projection of the edge mark is parallel to the red line.

6.1 Merchandise Recognition by Speeded-up Robust Features (SURFs)

In the learning stage, we construct a database of merchandise information, including the image feature points, as well as the name, location, brand, and price, of each merchandise item. For this, at first we segment respective merchandise images taken with the mobile-device camera (called client-camera images) into merchandise-item image parts, extract feature points from the results using the SURF extraction algorithm [1], and record the descriptor of the feature points in the database, including the distribution of the first-order Haar wavelet responses in the x and y directions in the image. An example of the results is shown in Fig. 9(a)

In the navigation stage, the server-side system analyzes the client-camera images to extract SURF features. An example of the results is shown in Fig. 9(b).
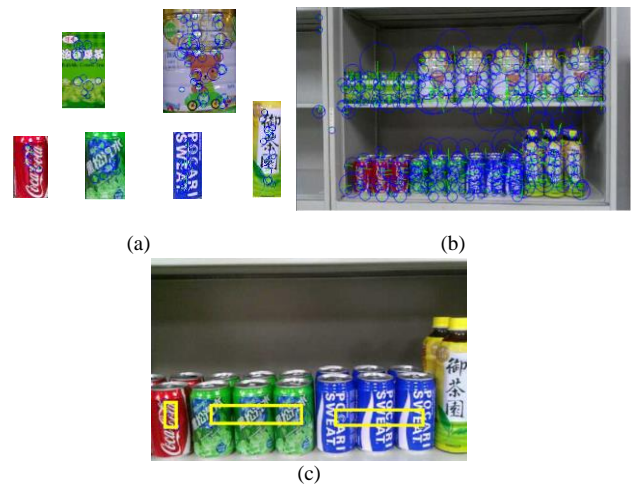

(a)   (b)

(c)

Fig. 9: Merchandise recognition. (a) Feature points of merchandise images (b) Feature points of the client- camera image of a client image

(c) The bounding boxes of matched feature points.

Then, merchandise recognition is conducted. Since many merchandise items are contained in each acquired client-camera image, we match the features of each merchandise-item image part against the pre-constructed database and obtain a set of matched feature points. If the number of points in the set is large enough, then we decide that the merchandise item in the image part is the recognition result.

Next, we have to specify the region of each merchandise items in the acquired client-camera image, so that we can augment the corresponding merchandise information at the correct position on the mobile-device screen. For this, after we obtain the set of matched feature points, we try to find a bounding box to enclose them, as shown by the example in Fig. 9(c). Finally, the server-side system sends the locations of these bounding boxes in the image together with the corresponding merchandise item information to the client-side system for AR display on the mobile-device screen.

6.2 Path Planning Techniques for AR-based guidance

Based on Hsieh and Tsai [5], we propose a technique of path planning for AR-based guidance using an obstacle avoidance map constructed in the learning stage as shown in Fig. 10. At first, a user is allowed to search the name of a merchandise item or a target place to visit using the client system on the mobile device. Then, the server-side system detects the location of the user as the start point and fetches the location of the merchandise item or the target place in the database as the destination to find a path consisting of a sequential set of *path points*. Subsequently, a *path simplification* process is conducted, which consists of two major tasks: *redundant point elimination* and *distance reduction*. The goal of *redundant point elimination* is to find two path points that are non-connected and can instead be connected together without colliding any obstacle, and then to remove the intermediate path points between them. The goal of the second task, *distance reduction*, is to find a direct "shortcut" between *two line segments* where each line segment is formed by connecting two path points. We apply the redundant point elimination and distance reduction processes iteratively on an initially-planned path until the points of the path do not change any further.

In addition, a user might not always follow a planned path in a navigation session. So we have to update the path when he/she walks away from the planned path. To do so (as shown in Fig. 10(b) and Fig. 10(c)), we just find the farthest path point p_s (shown as the red circle in Fig. 10(c)) reachable from the current point p (shown as the green circle in Fig. 10(c)) in the planned path, and then compose the new path as the one consisting of p and then p_s, followed by the remaining points in the old path.

After a path is updated, it contains only one new point, which is the current point p where the user is located, and the remainders in the path are the points of the originally planned path. Therefore, we have to check whether the resulting new line segments are of the simplest form or not. For this, we apply again the path simplification process described previously.
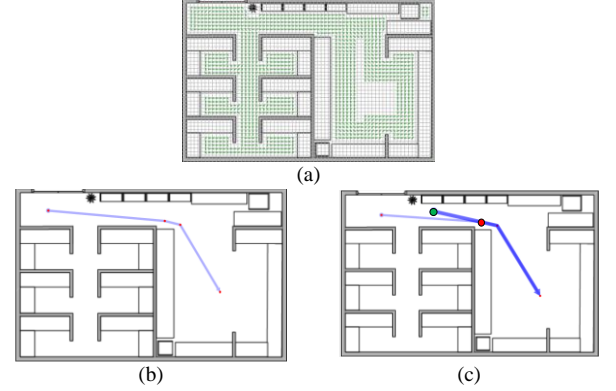

(a)

(b)   (c)

Fig. 10: Path planning techniques. (a) Obstacle avoidance map of the experimental environment (b) Original planned path (c) Updated path.

7. AUGMENTED REALITY TECHNIQUE FOR MERCHANDISE-SHOPPING GUIDANCE

We overlay navigation information onto the real images taken of the current scene to get augmented reality effects. Before this can be done, we have to conduct a mapping from the real-world scene to the image on the mobile-device screen, followed by a display of the augmented navigation and merchandise information. The techniques for carrying out these tasks are described in the following.

7.1 View Mapping between Real World and Client Device

The client system is designed to transform the GCS coordinates of the visited sites appearing in the acquired image into the MICS on the 2D mobile-device screen in order to augment the relevant information. This is done in two steps in this study — from the GCS to the CCS and then from the CCS to the MICS. The latter transformation from a point p in the CCS into a point q in the MICS can be accomplished by Eq. (1) described previously. And the former transformation from a point $a = [a_x\ a_y\ a_z]^T$ in the GCS into a point $p = [c_x\ c_y\ c_z]^T$ in the CCS may be carried out by:

$$p = \begin{pmatrix} right_x & up_x & -forward_x & -c_x \\ right_y & up_y & -forward_y & -c_y \\ right_z & up_z & -forward_z & -c_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} a_x \\ a_y \\ a_z \\ 1 \end{pmatrix} \qquad (3)$$

where the three vectors $up = [up_x,\ up_y,\ up_z]^T$, $right = [right_x,\ right_y,\ right_z]^T$, and $forward = [forward_x,\ forward_y,\ forward_z]^T$ are the three orthonormal bases of the CCS, as illustrated in Fig. 11.
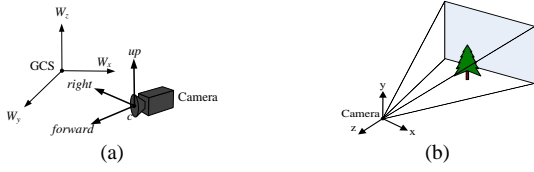
Fig. 11: A camera in the GCS and the CCS. (a) A camera in the GCS with three orthonormal vectors *up*, *right*, and *forward*. (b) The CCS.

7.2 Rendering for Target Place Information

To create AR effects, we overlay the names and distances of the target places onto corresponding positions in the scene image on the mobile-device screen. As illustrated in Fig. 12(a), a target-place range R is defined by four parameters, $\vec{f}$, p, w, and h, in the learning stage where vector $\vec{f}$ and point p specify the direction and position of R, respectively, and w and h specify the width and height of R, respectively. Accordingly, the four corner points of R can be computed and transformed into the MICS using the scheme described previously in Section 7.1. Also, as shown in Fig. 12(b), we clip the resulting coordinates of the corners to fit them into the range of the displayed image size. Finally, the position of the displayed text of the target place's name and distance as illustrated in Fig. 12(c) is computed to be the centroid of the four transformed corner points in the MICS.
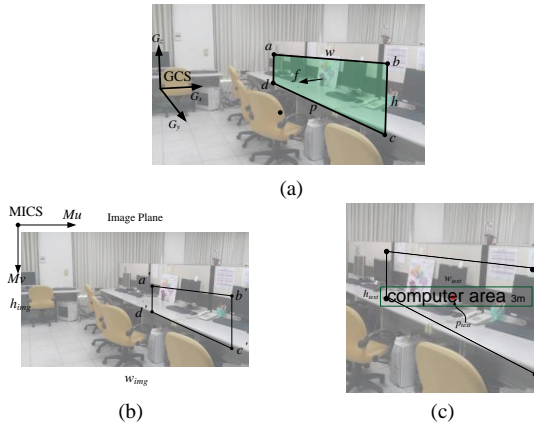

(a)

(b)

(c)

Fig. 12: Transformation of the GCS of a target place into the MICS. (a) Parameters of the target-place range. (b) Clipping transformed results into the range of the image size. (c) Displayed text of the target place.

7.3 Augmentation of Merchandise Information

We have to overlay the name, brand, price of each merchandise item onto the corresponding object which appears in the image taken with the mobile-device camera. To accomplish this, we determine where to display the text on the device screen by computing the MICS coordinates of the corner points a, b, c, and d of the merchandise-enclosing rectangular shape as shown in Fig. 13(a) to obtain the display position of the text of the merchandise name as shown in Fig. 13(b). In addition, we have to display the text of the brand and price under the merchandise name. For this, we obtain the display

position of such text in a way as shown in Fig. 13(c).

7.4 Rendering for Displaying Navigation Path

It is decided in this study that only two line segments of a path will be displayed at a time on the mobile device screen after a user receives the path sent from the server system. The displayed path is composed of thick line segments and an arrow. It is a 3D augmented object which can be rendered by the OpenGL API; therefore, we just have to compute the vertices of the geometric shape of the path. An example is shown in Fig. 14.

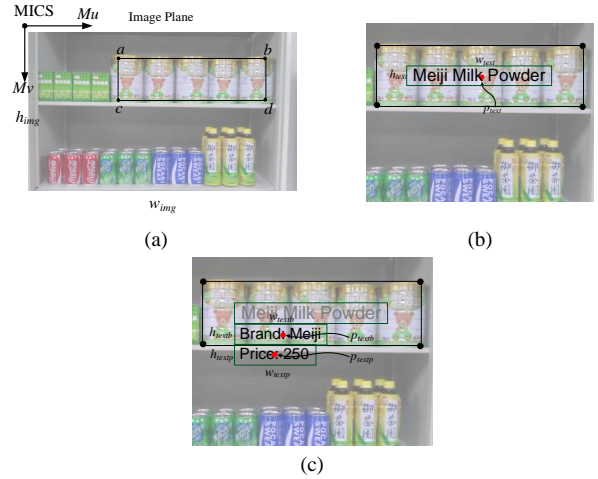(a)   (b)

(c)

Fig. 13: Augmentation of merchandise information. (a) Parameters of a merchandise item (b) Display the name of a merchandise item on the display position p_{text}. (c) Displaying the brand and price of a merchandise item on the display position p_{textb} and p_{textp}.

Fig. 14: Display of the first two line segments of a path.

8. EXPERIMENTAL RESULTS

The environment map of the environment where our experiments were conducted is shown in Fig. 15, which includes six target places (shown as green regions), eight merchandise items (shown as blue texts), and two fisheye cameras (shown as blue circles). The first experimental result we show is that of two users browsing surrounding target places in the environment. The two users stood at the locations shown in Fig. 16(a), and the corresponding augmented images appearing on their mobile-device screens are shown in Figs. 16(b)-(d) and Figs. 16(e)-(g), respectively.

Also, we show the results of merchandise items browsing. Figs. 17(a)-(b) show the results of browsing single merchandise items in the augmented images, and Figs. 17(c)-(d) show the results of browsing multi-type merchandise items in the augmented images.

7

9. CONCLUSIONS

In this study, an indoor navigation system for merchandise shopping and similar activities based on AR and omni-vision techniques using mobile devices has been proposed. Several techniques have been proposed for use in the system, including: 1) a method for multi-user identification in omni-images of indoor environments; 2) a method for multi-user localization in indoor environments by 3D omni-image analysis; 3) a method for recognition of merchandise items by SURF extraction and matching; 4) a method for indoor AR-based guidance for merchandise shopping by overlaying merchandise information on the real objects in scene images; 5) a method for indoor AR-based guidance for merchandise shopping by overlaying a navigation path on the floor in the scene image. The experimental results have revealed the feasibility of the proposed system. Future studies may be directed to 1) communicating images captured the cameras on different users' mobile devices, by which the users can browse different places without going there, providing the capability for processing multiple environment maps, which can be utilized for multi-user localization on different floors or at distinct places of an indoor environment, and so on.

REFERENCES

[1] H. Bay, *et al.*, "SURF: Speeded Up Robust Features," Features," *Computer Vision and Image Understanding (CVIU)*, vol. 110, no. 3, 2008, pp. 346-359.

[2] T. Miyashita, P. Meier, T. Tachikawa, S. Orlic, T. Eble, V. Scholz, A. Gapel, O. Gerl, S. Arnaudov, and S. Lieberknecht, "An Augmented Reality Museum Guide," *Proc. of IEEE International Symposium on Mixed and Augmented Reality (ISMAR)*, Cambridge, UK, 2008, pp. 103-106.

[3] J. Kim and H. Jun, "Vision-Based Location Positioning using Augmented Reality for Indoor Navigation," *IEEE Transactions on Consumer Electronics*, vol. 54, pp. 954-962, 2008.

[4] D. Chen, S. Tsai, C. H. Hsu, J. P. Sinhg and B. Girod, "Mobile Augmented Reality for Books on a Shelf," *Proc. of 2011 IEEE International Conference on Multimedia and Expo*, Barcelona, Spain, pp. 1–6, July 2011.

[5] M. Y. Hsieh, *et al.*, "A Study On Indoor Navigation By Augmented Reality And Down-Looking Omni-Vision Techniques Using Mobile Devices," in *Computer Vision,Graphics, and Image Processing*, Aug 2012.

[6] T. Akenine-Moller, *et al.*, "Pespective Projection," in *Real-Time Rendering*, Third Edition, T. Akenine-Moller, ed., 2008, pp. 92-97.
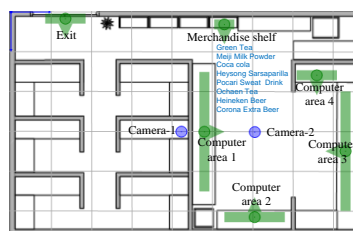
Fig. 15: The environment map of the experimental environment.

(a)

(b)   (c)

(d)   (e)

(f)   (g)

Fig. 16: Two users browsing surrounding target places at certain locations. (a) Place where the two user stood at. (b)-(d) The series of resulting augmented images of the female user. (e)-(g) The series of resulting augmented images of the male user.
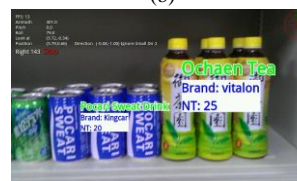

(a)   (b)

(c)   (d)

Fig. 17: Results of browsing merchandise items. (a)-(b) There is single-type merchandise items in the augmented image.(c)-(d) There are multi-type merchandise items in the augmented image.