VIEW MANAGEMENT OF ANNOTATIONS FOR WEARABLE AUGMENTED REALITY

K. Makita, M. Kanbara and N. Yokoya

Nara Institute of Science and Technology, Japan
(koji-ma, kanbara, yokoya)@is.naist.jp

ABSTRACT

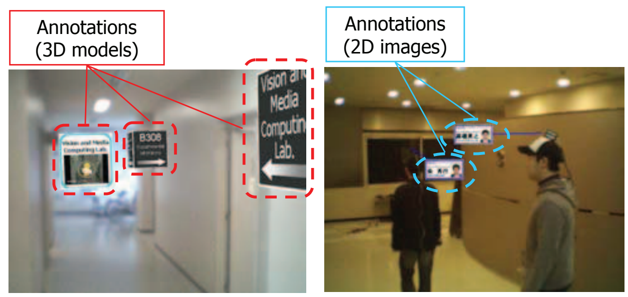
In annotation overlay applications using augmented reality (AR), view management is widely used for improving readability and intelligibility of the annotations. In order to recognize the visible portions of objects in the user's view, the positions, orientations, and shapes of the objects should be known in the case of conventional view management methods. However, it is difficult for a wearable AR system to obtain the positions, orientations and shapes of objects because the target object is usually moving or non-rigid. In this paper, we propose a view management method to overlay annotations of moving or non-rigid objects for networked wearable AR. The proposed method obtains positions and shapes of target objects via a network in order to estimate the visible portions of the target objects in the user's view. Annotations are located by minimizing penalties related to the overlap of an annotation, occlusion of target objects, length of a line between the annotation and the target object, and distance of the annotation in sequential frames. Through experiments, we have proven that the prototype system can correctly provide each user with annotations on multiple users of wearable AR systems.

Index Terms— Augmented reality, Wearable computer, Annotation, View management

1. INTRODUCTION

Augmented reality (AR) can enhance real world by superimposing virtual objects on the user's view. Annotation overlay applications give us additional information for the real world [1, 2, 3]. Annotations are generally overlaid by using a 3D model or a 2D image as shown in Fig. 1.

It is absolutely essential for annotations to be understood as an information about a target object in a real-world scenario. Therefore, it is important to improve the readability and intelligibility of the annotations in the user's view. In annotation overlay applications, view management makes it possible to appropriately overlay annotations for a user so as to intuitively understand annotations [4, 5]. One critical issue of managing a view of annotations for a target object is the avoidance of an overlapping of one annotation with others [6, 7]. The unexpected occlusion of objects by other annotations must also be prevented. In order to overcome these problems, an annotation overlay system needs to estimate the visible portions of target objects in the user's view. In conventional view management methods, position, orientation, and shape of objects are known. However, it is difficult for a wearable AR system to obtain the position, orientation, and shape of objects if the objects are moving or non-rigid. In this paper, we propose a view management method for networked wearable AR to overlay annotations of moving or non-rigid objects. The wearable AR system obtains positions and shapes of target objects via a network shared database framework [3] in order to estimate the visible portions of target objects in the user's view. The annotations are overlaid by minimizing the penalty function that considers overlaps of annotations, occlusion of target objects, distance between the annotation and the target object, and distance of the annotation in sequential frames.



(a) 3D models (b) 2D images

Fig. 1. Examples of annotation overlay images.

2. AR SCENE FOR TESTBED

We have prepared an indoor environment as a testbed for proposed wearable AR system. The environment is conducive to the use of wearable AR systems that generate annotation overlay images by estimating the user's position and orientation using a localization method [8] and obtain annotation information via a wireless network by using a framework [3]. In this environment, wallpapers with pre-printed invisible markers on a ceiling are used in the localization method [8]. Further, the position and orientation errors of the method are $1.3cm$ and $0.5°$. This paper assumes that a wearable computer user is a moving or a non-rigid object. We have prepared 3D object annotation for displaying navigation information and 2D image annotation for displaying user's location and personal information. The 3D object annotation is overlaid as objects fixed in the real world as shown in Fig. 1(a). On the other hand, the 2D image annotation is overlaid by using the proposed view management method as shown in Fig. 1(b).

3. VIEW MANAGEMENT ALGORITHM

The proposed method uses a position and a shape of wearable computer users and follows the three steps given below.

- **Step 1) Estimation of existing region of the users**
- **Step 2) Calculating penalty**
- **Step 3) Annotation overlay**

A detailed description of each step is given in the following sections.

3.1. Estimation of existing region of the users

In this step, the wearable AR system estimates an existing region of a user. The wearable AR system creates a probability map which is used for calculating the penalty for an annotation. Fig. 2 shows an example of a probability map. This method is designed using the concept of probability density, and the location of the user in the real environment is approximated by using primitive models. The details of each step of the estimation are given below.

In the first step, the position of the user in the view of the AR system is estimated by using the position data obtained via a network. In this report, the position of the user's camera is considered to be the user's position. In the next step, the existing region of the user is estimated by using primitive models. This approximation is based on the assumptions that the position of the top of the head is the user's position and that the user is positioned vertically on the floor. In this paper, the model user consists of three parts (head, upper body, and lower body), and rectangular solids are used as the primitive models. Each part of the model is represented a set of rectangular solids that are arranged radially in horizontal direction.

Each part has an existing probability as a parameter. We have concluded that the head has a high probability density because the camera is set on the head. In contrast, the lower body has a low probability density.

Next, each primitive model is drawn by off-screen rendering in order to estimate the probability in the image from the user's view. Each primitive model is drawn with a permeation rate (alpha value) in inverse proportion to the existing probability. The location of the users in the view of the AR system is estimated as described above.

Finally, the probability map is generated using the result of off-screen rendering. This map includes the location of the users as well as that of the 3D object annotations. The regions of 3D object annotations are rendered using colored textures and are not transparent. The color of a point on the image generated by off-screen rendering can be treated as an existing probability of the users and fixed annotations at that point.

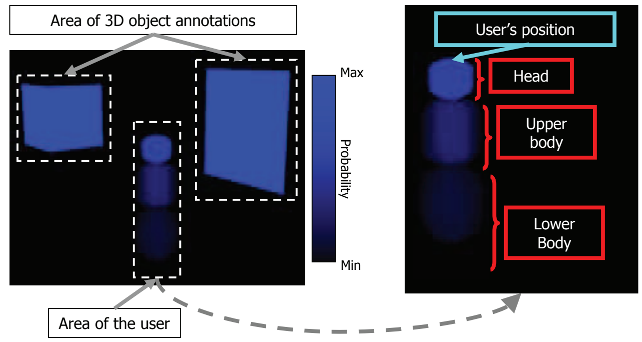


Fig. 2. Example of a probability map.

3.2. Calculating penalty

Many factors need to be considered while managing the overlaying annotations. As the first step of this work, we handle three factors related to readability and intelligibility: the amount of overlapping, the length of a line between the user and its annotation, and distance of the annotation in sequential frames. Annotations of the users are overlaid on the 2D user's view. A penalty when an annotation is overlaid at a position (x, y) is defined as

$$P(x,y) = w_o P_o(x,y) + w_l P_l(x,y) + w_d P_d(x,y) \quad (1)$$

where P_o is a penalty related to overlapping, P_l is a penalty related to the length between the user and its annotation on the view plane, and P_d is a penalty related to the amount of the distance between annotations in sequential frames on the view plane. Further, w_o, w_l, and w_d are weight parameters. Default of each penalty is described as follows.

Calculation of penalty related to overlapping P_o :
The penalty related to overlapping, P_o, includes two types of penalties. One is a penalty P_{anno} related to overlapping between annotations, and the other is the penalty P_{users} related to overlapping between the annotation and users. $P_o(x, y)$ is thus defined as

$$P_o(x,y) = P_{anno}(x,y) + P_{users}(x,y) \quad (2)$$

Fig. 3 shows three patterns of overlaying annotations that include the penalty related to overlapping between annotations. Fig. 3(a) shows an overlapping between 2D images. Fig. 3(b) shows overlapping lines. Fig. 3(c) shows an overlapping between the 2D image and the line. The penalty related to overlapping between annotations is the amount of overlap that is simply calculated as an area of the common area. In contrast, a penalty related to overlapping between the annotation and the users is calculated as a sum of probabilities in the common area.

Fig.4 shows two patterns of overlaying annotation that include the penalty related to overlapping between the annotation and the users. When the annotation is overlaid at a point

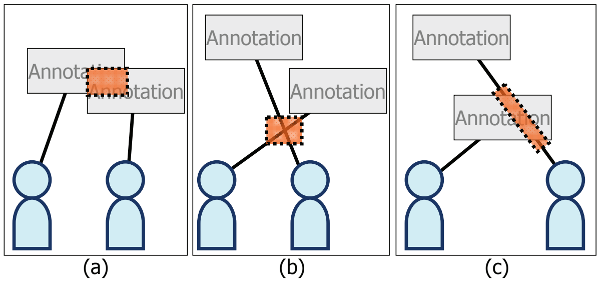


Fig. 3. Penalty related to overlapping between annotations.

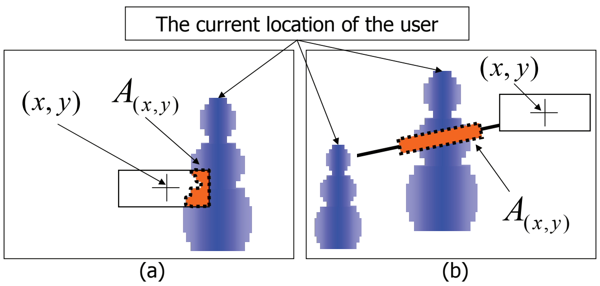


Fig. 4. Penalty related to overlapping between the annotation and users.

(x, y), $P_{users}(x, y)$ is defined as

$$P_{users}(x, y) = \sum_{i \in A_{(x,y)}} C_i \tag{3}$$

where i is a point on the probability map, C_i is the brightness value of the point i, and $A_{(x,y)}$ is a set of points that are occluded if an annotation is placed at (x, y).

Calculation of penalty related to line P_l :
The penalty related to the line between the annotation and the user P_l is proportional to the length of the line. In this study, in order to simplify the regulation of penalty, P_l is defined as $P_l(x, y) = l^2$, where l is the length of the line between the annotation and the user.
Calculation of penalty related to distance P_d :
P_d is a penalty related to the distance of the annotation on the view plane in sequential frames. As described in Fig. 5, P_d is defined as $P_d(x, y) = |B_t - A_t|$. A_{t-1} is the position of the annotation in the previous frame, A_t is the position in the current frame calculated by keeping the relative position between the annotation and the user in the previous frame, and $|B_t - A_t|$ is the distance between B_t and A_t.

3.3. Annotation overlay

Annotations are overlaid by minimizing the penalty defined in section 2.3. In order to minimize the penalty, the following steps are used.

A : Configuration of size and shape of annotation
B : Decision of an arrangement of annotations

A detailed description of these steps is given below.

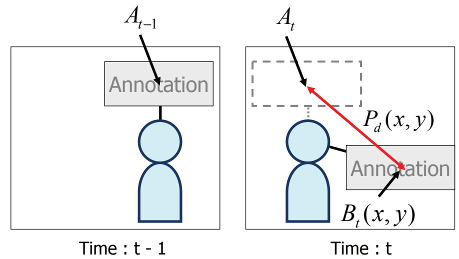


Fig. 5. Penalty related to the distance of the annotation in sequential frames.

A : Configuration of size and shape of annotation
2D images for annotations include a variety of information, and the images are displayed on the screen of the display device of the wearable AR system. Therefore, the size and the shape of the annotation have to be decided in consideration of the contents and the specification of the display device. For example, an image for the annotation that includes text should be sufficiently large for users of the wearable AR system to read and understand the text information. In contrast, if the images are oversized, the amount of overlapping is considerably large. The size and the shape of the annotation are fixed in view of these facts.

B : Decision of the arrangement of annotations
Two approaches for minimizing the penalty are proposed in this paper. One is global minimization and the other is sequential optimization. Global minimization guarantees an arrangement which has the global minimum penalty obtained by checking all the patterns of arrangements of annotations. However, it requires a considerably large amount of computation time. In contrast, sequential optimization works faster by deciding the position of each annotation in order. Although sequential optimization does not guarantee a minimum penalty, it is highly possible that an acceptable approximation is obtained by using this optimization technique. From this standpoint, annotations are overlaid by using the sequential optimization technique.

4. EXPERIMENT

In this experiment, one user of wearable AR system and three wearable computer users are introduced in the environment described earlier. The users' positions are estimated by their wearable PC (CPU: 933 MHz, Memory: 1.0 GB) and an original infrared camera (Captured image: 640×480 pixels, Captured rate: approximately 29 fps) and are transmitted to the wearable AR system. Position data is transmitted using user IDs so that the wearable AR system can identify the user. The user of the wearable AR system is equipped with a wearable PC (CPU: 2.0 GHz, Memory: 1.0 GB), and an original camera unit composed of an infrared camera and a video camera. The infrared camera (Captured image: 1024×768 pixels, Captured rate: approximately 30 fps) is used for estimating

(a) With proposed view management

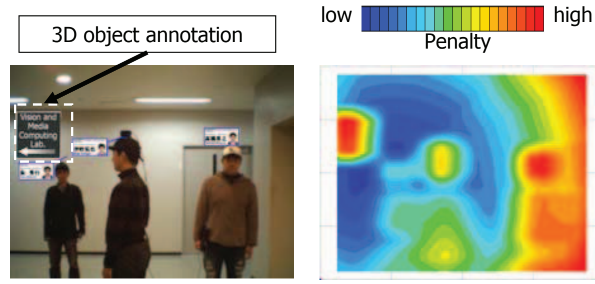
(b) Without proposed view management

Fig. 6. Experimental result of overlaying annotations.

the user's position and orientation. The video camera (Captured image: 800×600 pixels, Captured rate: approximately 15 fps) is used for capturing the user's view. In this experiment, the size of the generated AR images is 640×480 pixels; the annotations are rectangle-shaped and have a fixed size of 81×41 pixels. For this paper, both the upper and lower surfaces of the rectangular solids are square-shaped and measure 5 cm on each side. The number of solids used for representing each part are as follows: head: 25, upper body: 81, and lower body: 169. We have set the parameters of each zone as follows: head: 0.7, upper body: 0.3, and lower body: 0.1.

Fig. 6 shows the images generated when the four users are walking in the area and the user of the AR system receives the location of the other users around him at a rate of approximately 20 times per second. The annotations for the users in Fig. 6(a) are dynamically placed by the view management, and in Fig. 6(b), the annotations are placed by fixing the relative position between the annotation and the user. As shown in Fig. 6(a), the proposed method overlays annotations preventing overlaps in this result. For a faster penalty calculation in this experiment, the penalty is calculated only in an area of 21×21 pixels around the point A_t in Fig. 5 and 16 points around the user's position for preventing a local minimum. The distance between the 16 points and the user is $450 \div l$, where l is the distance between the target user of the annotation and the user of the wearable AR system in the real world. The AR system worked at a rate of 8-12 fps in this experiment.

Fig. 7 shows the comparative result between an annotation overlay image and a penalty distribution. Fig.7(b) shows penalties of all points on the image when an annotation for the user on the left is overlaid in Fig. 7(a). As shown in Fig. 7, the penalty is high in the area of the 3D object annotation and the users, and the annotation for the user on the left is placed at a position whose penalty is low.



(a) (b)

Fig. 7. Comparative result between annotation overlay image and penalty distribution.

5. SUMMARY

This paper proposes a view management method for annotations of moving or non-rigid objects. By using the proposed method, the users of the AR system were intuitively able to understand personal information. We confirmed the feasibility of the method by conducting experiments in an indoor environment. As a future work, we will consider the user's attention in the real world for improving the view management.

6. REFERENCES

[1] G. Reitmayr, E. Eade and T. Drummond: "Semi-automatic Annotations in Unknown Environments," 6th IEEE and ACM Int. Symp. on Mixed and Augmented Reality (ISMAR 2007), pp. 67–70, 2007.

[2] B. Bell, S. Feiner and T. Höllere: "Information at a glance," IEEE Computer Graphics and Applications, Vol. 22, No. 4, pp. 6–9, 2002.

[3] K. Makita, M. Kanbara and N. Yokoya: "Personal information annotation on wearable computer users with hybrid peer-to-peer communication," Proc. 16th Int. Conf. on Artificial Reality and Telexistence (ICAT 2006), pp. 217–227, 2006.

[4] S. Peterson, M. Axholt and S.R. Ellis: "Managing Visual Clutter: A Generalized Technique for Label Segregation using Stereoscopic Disparity," Proc. IEEE Virtual Reality (IEEE VR), pp. 169–176, 2008.

[5] B. Bell, S. Feiner and T. Höllere: "View Management for Virtual and Augmented Reality," ACM Symp. on User Interface Software and Technology (UIST 2001), pp. 101–110, 2001.

[6] R. Azuma and C. Furmanski: "Evaluating Label Placement for Augmented Reality View Management," Proc. 2nd Int. Symp. on Mixed and Augmented Reality (ISMAR 2003), pp. 66–75, 2003.

[7] F. Zhang and H. Sun: "Dynamic Labeling Management in Virtual and Augmented Environments," 9th Int. Conf. on Computer Aided Design and Computer Graphics(CAD/CG 2005), pp. 397–402, 2005.

[8] Y. Nakazato, M. Kanbara and N. Yokoya: "A localization system using invisible retro-reflective markers," Proc. IAPR Conf. on Machine Vision Applications (MVA2005), pp. 140–143, 2005.