An AR Interface to Enable Real-time Preview Design Variations in Actual Environment

Akira Iwaya
Department of Computer Science, University of Tsukuba
Ibaraki, Japan
iwaya@iplab.cs.tsukuba.ac.jp

Shin Takahashi
Faculty of Engineering, Information and Systems, University of Tsukuba
Ibaraki, Japan
shin@cs.tsukuba.ac.jp

Jiro Tanaka
Faculty of Engineering, Information and Systems, University of Tsukuba
Ibaraki, Japan
jiro@cs.tsukuba.ac.jp

Abstract
In this paper, we propose a way to enable users to preview a modified version of objects in the real world with a mobile device's screen using techniques of augmented reality with live video. Here, we applied the methodology to develop a prototype system and an interface that enables users to modify fonts of designs of a poster put in an actual environment and preview it to reduce the problem referred to as "impression inconsistency." From another point of view, this system uses an "interaction through video" metaphor. Tani et al. devised a technique to remotely operate machines by manipulating a live video image on a computer screen. Boring et al. applied it to distant large displays and mobile devices. Our system provides interaction with static, unintelligent targets such as posters and signs through live video.

Author Keywords
AR, interface to modify, preview, live video, interaction with real-world objects

ACM Classification Keywords
H.5.2. [Information Interfaces and Presentation]: User Interfaces: Input Devices and Strategies, Interaction Styles.

Introduction

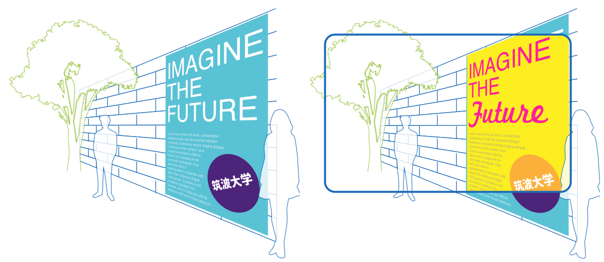

Figure 1: A poster placed in an actual environment (left), and preview of a modified design with AR (right)

To design printed materials, such as posters or flyers, we commonly use desktop publishing (DTP) software, such as Adobe Illustrator. DTP software adopts a WYSIWYG (what you see is what you get) interface, which displays the content (text and graphics) on the screen in a manner almost identical to the final result of the printed document. Thus, users can design documents intuitively, and obtain "what they see" as a printed result. However, the impression of a design when viewing it on a computer screen often differs from that viewed in an actual environment. We call this problem "*impression inconsistency*."

A major reason for this problem is that the impression is influenced by the environment where the printed material is posted. Whether the environment is indoor or outdoor, sun-lit or artificially illuminated, the materials of nearby walls, and surrounding objects, all affect the impression. Thus, often, a font looks good on a computer screen but not so impressive when printed and posted on a wall. A combination of colors that looks good on a screen may look not so eye-catching in the actual environment. If we could design posters at actual size in an actual environment, we could reduce the inconsistency; however, in reality, we have to work sitting in front of computers using DTP software, so we cannot design posters in their real environment.

Our final goal in this research is to design and implement an interactive editing and previewing tool to assess design variations of posters in actual environments where the user can edit in the same way as with DTP software. As a step towards this, we developed a prototype system in which the user can overwrite design elements such as fonts, arrangements, and colors, and preview the modification. In this system, the user views a target printed poster on a wall through the touch-panel display of a smart phone, and changes the size or font of the text in the poster with touch gestures on the image of the poster.

Fig.1 shows an example of a poster placed in an actual environment and an example of modification. The system alleviates the problem of "impression inconsistency."

Essentially, this is an augmented reality (AR) technique in the sense that the user is previewing a virtually modified 'real' poster in an actual environment. Additionally, the system enables the direct manipulation, and even modification, of text objects in printed posters. This method of AR that adds or mutates information of objects in the real world in real time enables users to modify already printed designs while experiencing the actual environment's atmosphere.

From another point of view, this system uses an "interaction through video" metaphor. Tani et al.[1] devised a technique to remotely operate machines by manipulating a live video image on a computer screen. Boring et al.[2] applied it to distant large displays and mobile devices. Boring's Touch Projector allows users to manipulate content on distant displays that are unreachable by pointing the device at the respective display and manipulating its content by touching and dragging objects in a live video. While this allows users to interact with objects through video, the target objects must be *intelligent* in the sense that they are controlled or displayed by computers, so that the objects can be operated remotely. Our system provides interaction with static, 'unintelligent' targets, such as posters and signs, through live video. The technique enables users to feel as if they are modifying the design of targets in the real world, and preview it through the screen of a mobile device.

In situ WYSIWYG editing of a poster in the real world

For the starting point of this study, we focused on the designs of texts because these are the most important elements in design. In this section, we show an AR system to overwrite fonts of printed texts with different fonts, while maintaining the original color. Fig.2 shows an overall view of a user who directs a smart phone at a poster and uses the system in an actual environment.

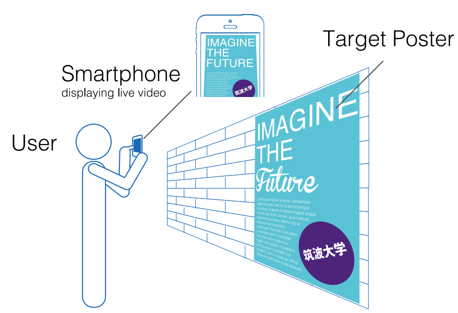

Figure 2: Overall view of using scene

Interaction for Editing

In our system, users can change the font and size of text through touch gestures. In either case, first, the user photographs the poster on the wall using the camera of a smart phone, and adjusts its orientation so that the touch-panel display shows a live video of the target part of the poster.

Second, the user specifies the text area he/she wants to modify by dragging the area diagonally on the touch-panel. Then, the text in the specified area is selected, and the user can change its size and font.

To change the size of the text, the user performs a pinch in/out gesture on the text. In this modification, the color and characters of the text are preserved because the system recognizes the characters in the text, generates a zoomed or shrunk new text image, and overlays it on the area.

To alter the font of the text, the user simply swipes left or right on the text. Then, the other font is applied to the text, and the new text image is overlaid on the camera image so that the user can view the modified version of the poster. Another swipe shows another font on the area. The user can select one of the installed fonts by multiple swipe gestures.

After the user has specified an area of text, the area of the poster is continually overlaid with the modified text image. Even if the user moves the smart phone, the system tracks the area and overwrites the new font image at the appropriate position.

Usage scenario of the system

Our target users are designers of printed posters. This section describes a scenario of the usage of our system in which the designer checks the result of a printed poster.

As described above, when using the existing DTP process, if the impression of the printed material is not what it is supposed to be, the designer must make modifications using the DTP software and print the material again to evaluate those changes. With our system, however, the designer can preview the modification instantly in an actual environment.

As an example, a designer created a poster of a university with DTP software. He printed it, put it on the wall of a university building, and checked how it appeared. Then, he felt impression inconsistency: specifically that it was more "low impact-looking than it ought to be." So, he changed the font of text, and decided to use our system to preview the result of the change before printing a modified poster.

He launched the system on his smart phone, directed it at the poster, and selected the region to modify by dragging on the touch-panel (the left of Fig.3). He swiped the text region left several times to select an appropriate font. Each time he swiped on the text, the text image of the new font appeared at the region (the right of Fig.3). Although the original font was Helvetica, many fonts, such as Futura and Trajan, are available. He was satisfied with Marketing Script.

Figure 3: Preview of modified design by AR

After the designer had selected a new font for the text, he previewed it from various points of view by moving around the poster (Fig.4). He was satisfied with the preview, and printed the modified poster again as the final version (Fig.5).

Figure 4: Preview of modification

Our system alleviates the impression inconsistency problem because the user can preview the modified version of the poster before printing, which reduces the necessary number of cycles of printing, checking, and modifying.

Another usage scenario of our system is in learning designs. Our system can be applied not only to the

Figure 5: Final result

designer's own posters but also to other posters in the local environment. By trying various modifications on posters encountered locally, designers can 'practice' poster designs in daily life.

Implementation

We have developed a prototype AR system that runs on a smart phone (iPhone 6, 4.7-inch screen, 1334 x 750, 326 dpi). For image processing, we use OpenCV. The working system is shown in Fig.6 and Fig.7.

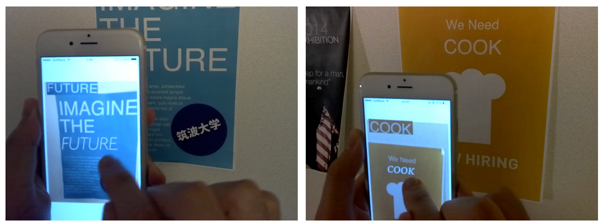

Figure 6: Working system on iPhone 6

Figure 7: Choosing fonts by swipe gesture

The user selects the word "FUTURE" on the left of Fig.6 and a word "COOK" on the right and checks the preview of the modified font. Fig.7 shows changing fonts by means of a swipe gesture.

The key function of the system is to recognize the characters in the text area that the user specifies by the dragging operation on a real-time video from the device's camera. Then, the system replaces this with a new font text image.

The system uses the inpaint[3] algorithm to remove the texts and recover the background behind the texts. In a related work, Kauai et al. studied diminished reality that enabled removing objects in a user-selected region[10].

Determining the text region specified by the user
The user selects a text region by a dragging operation on the touch-panel screen. The system recognizes the user's input as a rectangular region in the camera image. While the user continues dragging, a rectangular rubber-band filled with translucent white color is displayed on the screen to inform the user of the region selected. When the user lifts their finger from the screen, the region to be selected is fixed. Fig.8 shows the user selecting a text region by dragging. In this situation, the user is shooting the poster perpendicular to the wall, so that the obtained area of the original poster is almost rectangular. If the user shoots from an angle, the area obtained will be distorted. The current implementation does not handle this situation well, but we are implementing a function to correct the distortion to track the region and overlay text appropriately.

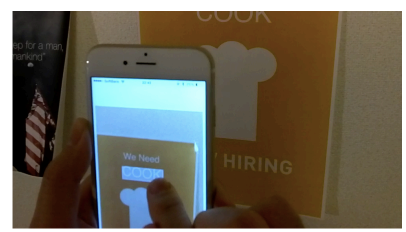

Figure 8: User is selecting a text region by dragging

Figure 9: Poster displayed as if background of the text is reconstructed

Figure 10: Final image with the region selected by the user replaced with new fonts

Text recognition
The image of the selected region is then binarized with an appropriate threshold, and passed to an optical character recognition (OCR) engine to recognize the characters in the area. Our prototype system incorporates Tesseract[1], an open source OCR engine, that runs on iOS[2].

Extracting the color of the text
The system must extract the color of the target text to render an overlay text image with a different font while preserving the color. First, the system extracts the exact character image area by creating a mask image. Then, it calculates the most frequent color in the masked image, which is used as the color of the text.

Producing a background image by removing the text
Before rendering a new text image, the system generates a pure background image of the area. In our system, the clipped text image region is processed with the image inpainting algorithm to erase the text and to generate a background image. Fig.9 shows a poster with the text "COOK" eliminated.

Tracking the text region
To appropriately overlay a new generated text image in the real-time camera input image, the system must track the selected region. To track the region, we used Clustering of Static-Adaptive Correspondences for Deformable Object Tracking (CMT)[4]. CMT is an object-tracking algorithm that is robust for scaling and rotating.

[1]Tesseract, https://code.google.com/p/tesseract-ocr/(2014.11.24)
[2]Tesseract-OCR-iOS, https://github.com/gali8/Tesseract-OCR-iOS (2014.12.1)

Overlaying the background image and the text image
Finally, the system overlays the generated background image, and renders the characters on it. The size of the characters is calculated so that the height of the characters fits the selected region. Users can change the font by swipe gestures, which are animated as if they are flipping through a series of font candidates. Fig.10 shows the resulting screen image in which the user previews the modified poster. In Fig.10, the font of the text is changed from Helvetica to Times New Roman, while the size and color of the text are preserved.

Preliminary Evaluation

Prototype System Evaluation
We tested our system using several posters. Our system works in real-time and with almost no delay when no selected area is selected (approximately 25 FPS). However, after the user selected a region and tracking started, the frame rate declined to approximately 5 FPS. This FPS refers to frames per second of live video, not the rendering times of the user interface. As a future project, a freezing video image function, similar to that used in Touch Projector[2], may alleviate this problem.

The accuracy of the text recognition depends on Tesseract. When the text is relatively large and captured sharply, the system recognizes characters correctly in most cases.

User Interview
We asked three graduate school students majoring in design to use our prototype system, and then conducted interviews to hear what they thought about the system. The posters used in their trial includes that shown in Fig.11. provided by us, and other posters in their possession.

Figure 11: A poster used in user interviews

We obtained the following comments. The graduate students stated that the good points of the system were that "it would be great if we could use this system in public spaces to investigate whether public signs or posters are visually attractive" and "we can easily understand visual impressions including a surrounding environment intuitively." Moreover, they insisted that the system would be helpful for posters or graphic designs not created by them, because they could immediately modify designs they found and liked to their style. Modifying existing good designs aids in learning design.

Regarding the interface, they said "it is intuitive and simple but not suitable for professional use because refining details while aiming the device is hard because devices and live video are wiggling and tracking function should be improved." "Saving the image of the modified design is a required function." They said that it was sometimes difficult to select a region of text while conducting live video, due to the movement of the device while held in the user's hand. Regarding solutions to this problem of aiming the device at the poster for a prolonged period, a study suggested guidelines for AR user experiences. Grubert et al. proposed a design guideline for a hybrid of a poster and live video interface, especially public posters experienced in a mobile context[7]. One of the guidelines is to "Keep AR experiences after the user has stopped aiming the camera at the poster." We will follow this and keep the poster editable while a user is not aiming the mobile device in future work. Also, we are planning a pause/play function for the live video, invoked by double-tapping the screen.

Related Work

There are some studies that have used live video to interact with or expand real object information. Rohs proposed a virtual coordinate to match visual space and real space using two markers and a system with coordinates that also visually detected phone movement, which enables applications such as item selection and interaction with large-scale displays[5]. However, our system is equipped with a tracking algorithm, and so is a marker-less AR system capable of application with any poster. Schoning et al. presented a method of laying out virtual buildings according to physical maps and enabled measurement of distances on the map by means of live video capturing of the map with a pen.[6]. The proposed method overlays buildings and automatically appears as the user sweeps over the map. However, with our system, the user interacts with the live video by means of drag, pinch, and swipe gestures and modifies existing content directly.

Several methods of making rendered objects appear similar to real objects have been proposed. Agusanto et al. reported a method of rendering 3D models photorealistic for AR using environmental illumination[8]. In our study, a modified design must be overlaid into an existing environment to enable previewing without feeling uncomfortable to reduce impression inconsistency. Lee et al. reported a method to preview designed products by making a rough mockup with CNC and coloring it by AR, considering occlusions such as hands[9]. However, our interest is in printed materials, and not only previewing but also modifying designs. Our work and that of Lee et al. has in common the use of AR to visualize designs in their actual environments. Lee et al. focused on already existing 3D models whereas our system must generate an overlay image taking into consideration its contents (e.g., fonts, colors, text size, and background).

Conclusions

We have described an AR interface that facilitates modification of objects in the real world using a live video from a mobile device's camera, which was developed from the ideas of Tani et al. and Boring et al. We also developed a prototype application system that enables one to preview and modify designs in their actual environment to reduce the impression inconsistency resulting from differences between impressions on a computer display versus a printed poster.

Future studies will include examining differences between AR devices, and improving the system to enable the modification of not only fonts but also other design elements, such as positions and color combinations. We will also consider the size of the device screen and the device camera's angle of view and investigate what users feel when viewing actual posters using the camera screen by means of user research. Finally, we must evaluate the usefulness of the system in actual professional designers' daily operations. We will perform these evaluations in future work.

References

[1] Tani, M., Yamaashi, K., Tanikoshi, K., Futakawa, M., and Tanifuji, S., "Object-oriented video: interaction with real-world objects through live video", In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 593-598, 1992.

[2] Boring, S., Baur, D., Butz, A., Gustafson, S., and Baudisch, P., "Touch projector: mobile interaction through video", In Proceedings of the SIGCHI Conference on Human Factors in Computing Systems, pp. 2287-2296, 2010.

[3] Bertalmio, M., Bertozzi, A. L., and Sapiro, G., "Navier-stokes, Fluid Dynamics, and Image and Video Inpainting", In Proceedings of the 2001 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, Vol.I, pp.I355-I362, 2001.

[4] Nebehay, G., and Pflugfelder, R. , "Clustering of Static-Adaptive Correspondences for Deformable Object Tracking", In Proceedings of the 2015 IEEE Computer Vision and Pattern Recognition, 2015.

[5] Rohs, M., "Real-world interaction with camera phones." Ubiquitous Computing Systems. Springer Berlin Heidelberg, pp. 74-89, 2005.

[6] Schoning, J. , Kruger, A., and Muller, H. J., "Interaction of mobile camera devices with physical maps", In Adjunct Proceedings of the 4th International Conference on Pervasive Computing, pp. 121-124, 2006.

[7] Grubert, J., Grasset, R., and Reitmayr, G., "Exploring the design of hybrid interfaces for augmented posters in public spaces", In Proceedings of the 7th Nordic Conference on Human-Computer Interaction: Making Sense Through Design, pp. 238-246, 2012.

[8] Agusanto, K., Li, L., Chuangui, Z., and Sing, N. W., "Photorealistic rendering for augmented reality using environment illumination" , In Proceedings of the 2nd IEEE/ACM International Symposium on Mixed and Augmented Reality, pp. 208-216, 2003.

[9] Lee, W., and Park, J., "Augmented Foam: A Tangible Augmented Reality for Product Design", In Proceedings of the 4th IEEE/ACM International Symposium on Mixed and Augmented Reality, pp.106-109, 2005

[10] Kawai, N., Sato, T., and Yokoya, N., "Diminished Reality Considering Background Structures", In Proceedings of the 2013 IEEE International Symposium on Mixed and Augumented Reality, pp. 259-260, 2013.