C-Band: A Flexible Ring Tag System for Camera-Based User Interface

Kento Miyaoku[1], Anthony Tang[2], and Sidney Fels[2]

[1] NTT Cyber Solutions Laboratories, NIPPON TELEPHON AND TELEGRAPH
1-1 hikarinooka Yokosuka-shi Kanagawa, Japan
`miyaoku.kento@lab.ntt.co.jp`
[2] University of British Columbia,
Vancouver, British Columbia, Canada
`{tonyt,ssfels}@ece.ubc.ca`

Abstract. This paper proposed a new visual tag system for enhancing real-world media interaction using handheld camera devices. This paper also described performance evaluations of the prototype, and its applications. C-Band is based on a ring with a color pattern code. A C-Band tag can provide a self-contained URL, and is flexible enough to allow various aesthetic designs for the tag's surface. Furthermore, the tag's structure is useful for building interactive techniques. Taken together, these features suggest that C-Band is an effective method to build various attractive camera-based media interactions.

Keywords: Visual Tag, Color-Difference Signal, Camera, Mobile Terminal.

1 Introduction

One of the most promising goals in mobile computing is linking real-world objects such as paper, panels, labels and screens to digital data. This will allow real world objects to be used as input keys to trigger various applications [1,2] Among the many approaches to realizing this vision, we are most interested in camera-based techniques because of the ubiquity of mobile phones and other devices with high-resolution cameras (Figure 1).

Visual tags such as QR code [3] are very attractive since they can contain a practical amount of application data such as a URL, and the computational cost for decoding is quite low so they can be mounted on even low power mobile devices such as cell phones. However, current existing visual tags are aesthetically inflexible. They often can not fit into surrounding graphical design. This limitation reduces the popularity of using visual tags in camera-based interaction systems.

To overcome this limitation, we proposed a new color ring based visual tag system called "C-Band". Using a color difference-based encoding provided by the ring gives C-Band tags sufficient design flexibility to allow integration with various graphic designs (Figure 2). We built a prototype using a preliminary algorithm and evaluated its basic performance. The evaluations show that a C-Band tag containing 28 bytes

R. Shumaker (Ed.): Virtual Reality, HCII 2007, LNCS 4563, pp. 320–328, 2007.

can be effectively extracted from a 640×480 pixel resolution image. These features suggest that C-Band is an effective method to build various camera-based interactions. We describe some possible applications in this paper.

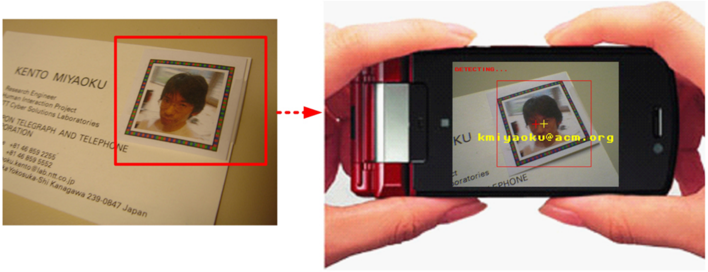

Fig. 1. Linking real world objects with digital data

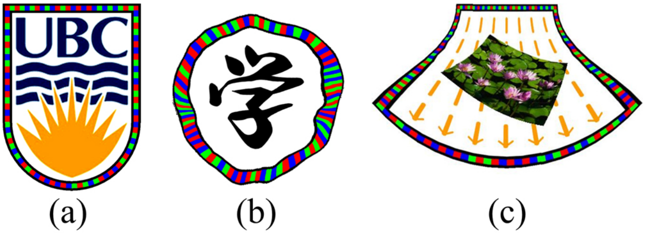

Fig. 2. Feasible C-Band tag designs

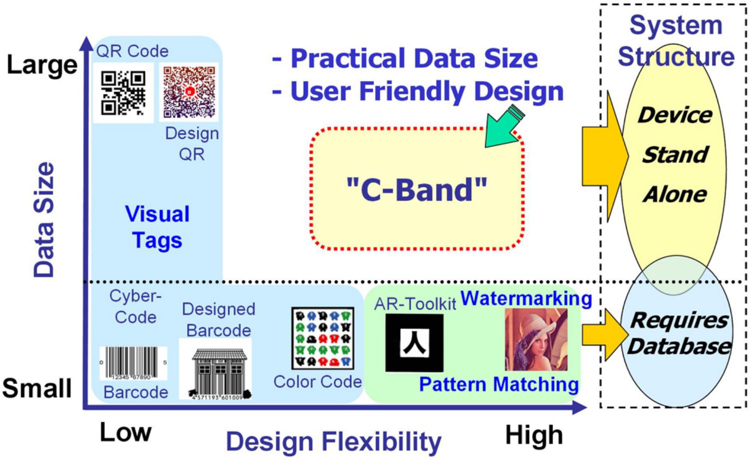

Fig. 3. Design space of camera based tagging systems

2 Related Work

There are three main approaches to embedding camera readable data into an object/image: (1) visual tags: placed on the object and scanned [3-7]; (2) digital image watermarking: dormant and imperceptible information is embed via digital encoding [8,9], and (3) image feature pattern matching: the image itself is used as data [10] .

Some explicit visual tag systems support a very large data size so that the tag can contain various kinds of data such as URLs, e-mail addresses, and so on. This flexibility enables the tag system to stand alone (i.e. no database lookup), and facilitates building many applications in a simple and cost-effective way. Also, systems to decode visual tags are usually very simple and processed rapidly so that they can be executed on common cell phone devices. However, their designs are aesthetically inflexible and cannot provide any human-readable semantic information. Moreover, they often disrupt the attractiveness of the overall graphical design.

Watermarking techniques [8,9] and image feature pattern matching techniques [10] can place data into any image without compromising the aesthetic quality. This means that these techniques integrate tagged areas with human-readable semantic information. However, the size of data that can be embedded with these techniques is usually relatively small and static, thereby requiring a database to look up the information associated with the data, i.e. the ID. The database requirement makes these approaches less flexible and increases the overall complexity of the system.

In summary, existing techniques that allow common cameras to scan data create a tradeoff between data size and aesthetic flexibility (Figure 3). DataGlyph [11] can well integrate relatively large amounts of data into images. Unfortunately, it is expensive to extract DataGlyph from images captured by common image sensors. Its response speed is a problem when it runs on common cell phone devices.

These observations indicate that the visual tag system still the most reasonable approach to build phone camera based interaction systems, as some interactions requires high response speeds. If design flexibility could be added to visual tags, it would have great potential to promote the use of camera based interactions. Our proposal, the new visual tag called "C-Band", achieves both practical data size and design flexibility.

3 C-Band: A Color Ring Visual Tag System

Since visual tags must have a pattern area that expresses computer-readable data, the main challenge is to harmonize the pattern area with the target. Our approach utilizes the frames commonly placed around figures to indicate, to emphasize, or to mark off areas. These frames are modified to yield a visual encoding scheme that is based on frame color modulation. The scheme does not rely on geometric information for decoding; instead, the scanner simply detects sequential color differences. Consequently, the tags can have various shapes and surface designs (Figure 2).

3.1 Structure of C-Band Tag

Figure 2 shows examples of C-Band tags. A C-Band tag consists of a content area and a frame surrounding the content area. The ring surface is divided into color sectors, and the number of color sectors depends on the size of the embedded data. Each ring has black edges to enhance the accuracy of ring component extraction. A thin white margin between the ring and the content area is usual to ensure reliable detection.

3.2 Color Difference-Based Encoding Method

To achieve tag shape flexibility, C-Band uses a simple hue-difference based encoding method using 3 colors to encode information [12]. With this method, a hue-increase between colors of a pair of successive color-sectors on the ring expresses a single bit; the hue-increase of $2/3\pi$ indicates “0”, and $4/3\pi$ indicates “1”. For example, using a color wheel representation with (R)ed (hue=0 or 2π), (G)reen (hue=$2/3\pi$) and (B)lue (hue=$4/3\pi$), color changes from R→G, G→B and B→R indicate ‘0’, and color changes from R→B, B→G, and G→R indicate ‘1’. A binary sequence that includes n-bit data and c-bit checksum can be transformed into a color pattern using this encoding method. The pattern expressing $n+c$ color differences is placed on the ring in a counter-clockwise direction. As a sector is required to delineate the start and end of the pattern, the ring is divided into $n+c+2$ sectors to express n-bit data.

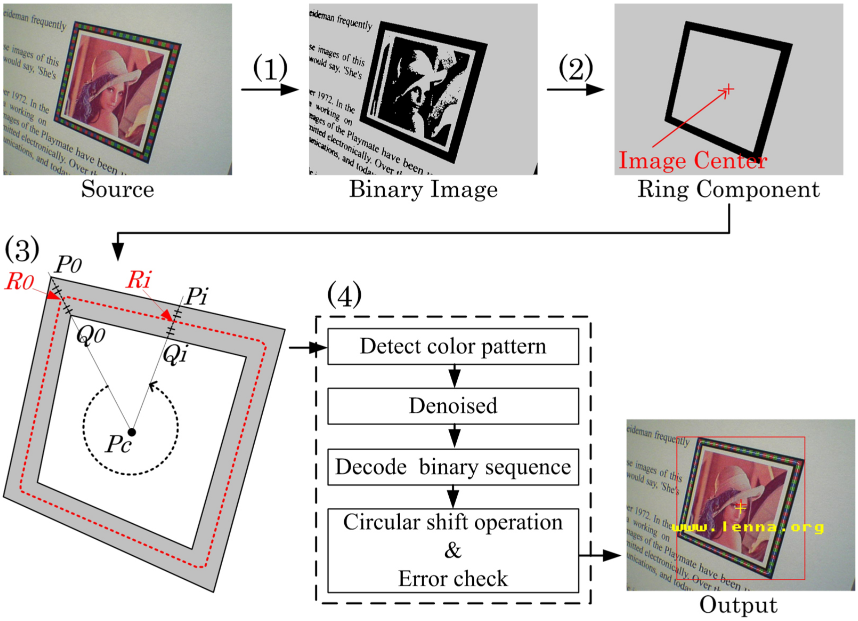

Fig. 4. C-Band Tag Detection Process

3.3 Detection Processes

We designed a preliminary algorithm to detect and decode a C-Band tag. When the user captures a tag, the tag is detected and interpreted by the algorithm. The algorithm, outlined in Figure 4, consists of the following four stages:

(1) Binarization
The original RGB color image is transformed into a binary image by applying a preset threshold to the luminosity of each pixel. The luminosity, Y, is calculated by the equation: $Y = 0.299*R+0.587*G+0.114*B$.

(2) Ring component extraction
After labeling all connected components in the image, a ring component candidate is selected by considering the shape and the positional relationship with the image center and other components.

(3) Ring skeleton extraction
Pixels at the boundary between different colors including the edge are often mixed making it difficult to accurately detect the color changes at these points. To decrease the effect of color mixing, we use a thinning algorithm to extract the most reliable pixels. We use the following algorithm:
The pixel sequence R_i on the ring skeleton is extracted. First, P_c inside the ring component is selected. The pixel sequence P_i $(i = 0,..,N-1)$ is the sequence of ordered pixels appearing on the outer edge of the ring component in a counter-clockwise direction. For each i $(i = 0,..,N-1)$, the intersection point Q_i between the segment P_iP_c and the inner edge of the ring component is detected, and then, the middle point R_i of the segment P_iQ_i is extracted as a color pixel near the middle of the sector.

(4) Color Pattern Decoding
Each color of R_i is classified into one of the three primary colors (R,G,B) simply using the maximum color value. We do not use hue for this process to reduce the computational cost. From this, we obtain the RGB color sequence that appears around the ring (such as {R, R, R, G, G, G, B, B, B, *R*, G, G, G}) as the pixel sequence R_i.

This sequence is denoised and reduced to form the RGB color pattern for decoding: first, repeated colors mark a sector and are reduced to a single color in the sequence; second, if a color sector only has a single pixel, we remove the sector from the sequence as noise. In the example above, the color sequence is converted into the color pattern of {R, G, B, G} (note that *R* is removed as noise). The obtained color pattern is transformed into a binary sequence by applying the above transitions rules.

Then, every code obtained by a circular shift operation to the binary sequence is checked. If no error is found, the scanner decides that a tag and its code were detected which yields the data of the C-Band tag.

4 Evaluation of Basic Performance

We prototyped the C-Band system (Figure 5) using a Windows PC and a webcam (Sony PCGA-UVC11A, Lens: F3.4 focal length f=5mm (35mm film conversion f=40mm); Focus range: 3cm to infinity). We used standard PC equipment for ease of implementation; in practice, the algorithm is simple enough to run on a standard mobile phone. Our prototype uses Microsoft's DirectShow technology, and can process white-balanced images (provided by the webcam) at 15-30fps depending upon camera resolution. In our tests, the tags expressed 16-bit CRC check bits. When the data size is n-bit, the ring of the tag is divided into $n+18$ sectors. The tags were printed by a small common inkjet printer (HP Deskjet 5740).

4.1 Performance Under Indoor Lighting

The basic detection performance of the system was tested under indoor fluorescent lighting. The system processed 1000 image frames in which a tag was captured continuously. We measured how many times the system could detect and decode the tag's data correctly. Note that, because of the CRC check bits, the system never incorrectly decoded a tag (a false positive). Thus, detection reliability is dependent only on the percentage of correct detections (the hit rate) in the 1000 trials.

There is a tradeoff between camera resolution and detectable data size, because data size depends on how many color sectors are used: each color sector must be captured with enough pixels to detect its color accurately. This experiment was intended to determine the smallest detectable data size for each resolution under ideal conditions.

We tested square tags containing data ranging from 4 bytes to 32 bytes. The height of each printed tag was 5cm. The width of the ring of these tags was 1/14 of the height of the tag, or 3.6mm. The tags were detected by a camera set 7cm in front of the tag. In this case, the captured tag was in the center of the image and the height of the tag is about 80% of the height of the image. We tested camera resolutions of 640×480, 320×240 and 160×120 pixels. Figure 5 shows the results of these tests.

With images of 640×480 pixel resolution, the system achieved over 90% detection reliability for tags of up to 28 bytes. 320×240 pixels images attained 90% reliable detection for the square tag of up to 16 bytes, and the oval tag of up to 20 bytes. Finally, 8 byte tags were reliably detected almost 100% of the time even with 160x120 pixel resolution.

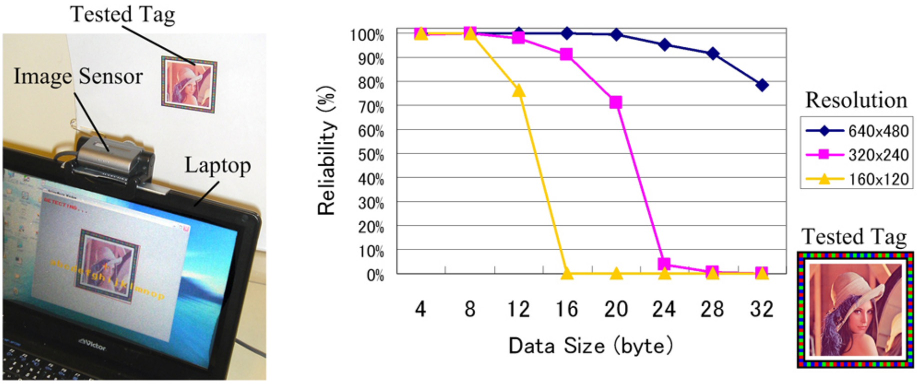

Fig. 5. Relationship between detection reliability and data size

4.2 Processing Speed

The processing speed of the detection system was checked by using a small laptop (Victor MP-XP7310) which has a Pentium M 1Gz CPU. Under the same condition as the experiment in Section 4.1, 1000 captured images were processed and the average processing time for each image was calculated. The processing times were 68.0msec, 14.8msec and 3.5msec for 640×480 pixels, 320×240 pixels and 160×120 pixels, respectively. The labeling process in the ring component extraction occupies the longest processing time, and ring skeleton extraction is also takes long. Extrapolating this performance to current cell phone technology (100MHz to 425MHz processors), we would expect a 2 to 10 times increase in processing time; however, implementing the image processing functions on a GPU (Graphics Processing Unit) - commonly found in newer mobile phones - would significantly decrease the processing time.

5 Potential Applications

With a prototype C-Band system in place, we explored a variety of application contexts for camera-based interaction. Due to space limitations, we describe the most salient and interesting explorations here.

(1) Paper Keypad
A paper keypad is a paper slip which shows several tags that contain a character code, a symbol code or a command code (Figure 6). The paper keypad can be used as a portable input method for small camera devices. Since existing visual tags can not contain a figure on their surface, an embedded character must be shown next to a visual tag. C-Bands around each letter allow the desired characters to be pointed at and selected directly, improving text entry speed and accuracy.

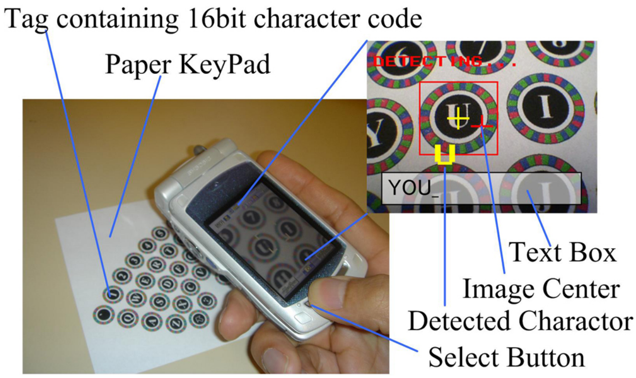

Fig. 6. Paper Keypad

(2) Large screen interaction
C-Band tags can also contain animation as semantic information which is not possible with the other tagging systems. We built a whack-a-mole game (Figure 7) to exploit

this feature. In the game display, nine C-Band tags indicate the holes of the moles. Each C-Band tag (hole) has a small ID. A user points and shoots by using a camera device. Since C-Band tags can occupy a large space in the game UI as holes, they can be detected much better and from wider ranges than is possible by placing small matrix codes beside the holes. Test users enjoyed this game and generally seemed to like the feel of the UI graphics design.

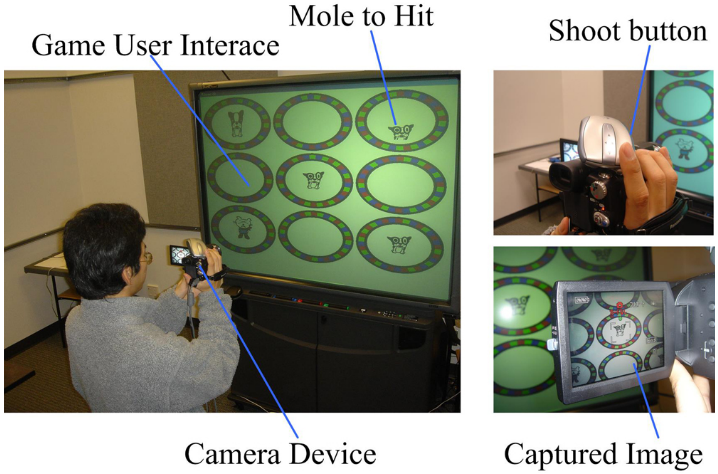

Fig. 7. C-Band based Whack-a-Mole game screen

(3) Hot Frame - Physical URL

The evaluations showed that the tag can directly provide a short URL which can be expressed by 28 bytes or fewer. A C-Band frame works both as a URL source and as a cue to indicate a hotspot area. As shown in Figure 2 (a), C-Band can also be used as a part of the graphical design of a frame. QR-code tags require more than 1cm space if the camera has 0.3mm resolution. If the figure is restricted to a 3cm×3cm space it is not efficient to use a QR-code as shown in Figure 8. However, C-Band can use this space effectively by forming its shape to fit around the figure. This feature makes C-Band a reasonable way to attach a physical URL to small figures like trademarks.

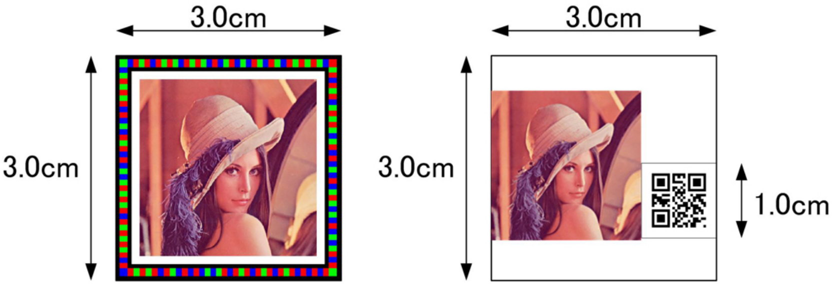

Fig. 8. Adding a URL tag to small, fixed space

6 Conclusion

We created a new flexible visual tag system, "C-Band", to promote the use of camera-based interaction. Experiments under indoor lighting with a proof-of-concept system showed that a tag with up to a 28 byte data can be detected well by a 640×480 pixel camera. We also showed that C-Band allows designers to be flexible and creative in embedding C-Band tags into graphical objects (Figure 2). We believe that these features of C-Band will encourage the adoption of camera-based interaction via visual tags. We continue to work on resolving the remaining issues with this technology. The most important task is to evaluate its performance under various lighting conditions. We confirmed that the prototype could work under some outdoor lighting conditions, but more extensive tests are needed. Also, we will improve the detection algorithm to increase robustness, as the current algorithm, while functional, is quite rudimentary. We plan to install C-Band into smart phones and PDAs to make the system truly practical.

Acknowledgments. The authors would like to thank members of the HCT Lab and NTT Cyber-Solutions Laboratories for their support. This work was done while the first author was visiting the University of British Columbia.

References

1. Brush, A.J.B., Turner, T.C., Smith, M.A., Gupta, N.: Scanning Objects in the Wild: Assessing an Object Triggered Information System. In: Proc.Ubicomp2005, pp. 305–322 (2005)
2. Rekimoto, J., Nagao, K.: The World through the Computer: Computer Augmented Interaction with Real World Environments. In: Proc. UIST'95, pp. 29–36 (1995)
3. QR-code.com, http://www.qrcode.com/
4. Semacode, http://semacode.org/
5. Rekimoto, J., Ayatsuka, Y.: CyberCode: Designing Augmented Reality Environments with Visual Tags. In: Proc. DARE2000, pp. 1–10 (2000)
6. Toye, E., Sharp, R., Madhavapeddy, A., Scott, D.: Using Smart Phones to Access Site-Specific Services. IEEE Pervasive Computing 4(2), 60–66 (2005)
7. ColorZip, http://www.colorzip.com/
8. Alattar, A.M.: Smart images using digimarc's watermarking technology. in Security and Watermarking of Multimedia Contents II, Proc. SPIE, vol. 3971, pp. 264–273 (2000)
9. Nakamura, T., Katayama, A., Masashi Yamamuro, M., Sonehara, N.: Fast Watermark Detection Scheme for Camera equipped Cellular Phone, In: Proc. NUM2004, pp. 101–108 (2004)
10. Kato, H., Billinghurst, M.: Marker tracking and HMD calibration for a video-based augmented reality conferencing system. In: Proc. IWAR1999, pp. 85–94 (1999)
11. Moravec, K.L.C.: A grayscale reader for camera images of Xerox DataGlyphs. In: Proc. BMVC, pp. 698–707 (2002)
12. C-blink: A Hue-Difference-Based Light Signal Marker for Large Screen Interaction via Any Mobile Terminal. In: Proc. UIST 2004, pp. 147–156 (2004)