Digital Watermarking Method to Extract Watermarks from Printed Matters with Cell Phone by Using Finder Patterns and Alignment Pattern of QR Code

Motoi Iwata, Kenji Yamanaka and Koichi Kise
Graduate School of Engineering Osaka Prefecture University
1-1 Gakuen-cho, Nakaku, Sakai, Osaka 599-8531, Japan
Email: iwata@cs.osakafu-u.ac.jp

***Abstract*—There are some watermarking methods to be able to extract a watermark even from a printed watermarked image with cell phones. In such a use, the speed of extracting watermark should be fast for users' convenience, where QR codes are used for the similar purpose. However, QR codes require its own space in addition to contents and its appearance of QR codes is unattractive. On the other hand, digital watermarking requires no additional space, and its appearance is visually attractive. Nakamura's method used a black frame for identifying a watermarked area. The problem of Nakamura's method is a large constraint on original images and inconvenient extraction way. Therefore, we employ the finder pattern and the alignment pattern of QR codes for position detection. The advantage of the proposed method is a small constraint on design and convenient way of real time extraction. In the experiment, we implement the proposed method as an Android application and evaluate its performance including its real extraction time on a cell phone. The experimental results confirm that the proposed method can extract a correct watermark in 2 seconds, where it is enough fast in practice.**

I. Introduction

Digital watermarking is a technique for embedding any information into digital contents so that the degradation caused by embedding is imperceptible, where the embedded information is called watermark. There are some watermarking methods to be able to extract a watermark even from a printed watermarked image with cell phones. In such a use, the speed of extracting watermark should be fast for users' convenience, where QR codes are used for the similar purpose. QR code is a 2-dimensional code developed by Denso wave Inc. [1]. The characteristic points of QR code are representing large amount of information bits and fast processing to read the information. However, QR codes require its own space in addition to contents. Moreover, the mosaic-like black and white appearance of QR codes is unattractive because humans cannot understand its patterns. This problem has not been solved although some methods colorize QR code or introduce an illustration partially. On the other hand, digital watermarking requires no additional space, and its appearance is visually attractive because the contents themselves represent information. Nakamura's method could extract a watermark from a printed image with the camera of a cell phone by using a black frame for identifying a watermarked area [2]. The problem of Nakamura's method is a large constraint on original images and impossibility to correct rotation of more than 90 degrees. Moreover it is users' inconvenience that Nakamura's method requires to take a picture one by one for each watermark extraction. Therefore, we employ the finder pattern and the alignment pattern as same as QR codes for position detection so that we can identify the watermarked area by detecting these patterns in the extracting process. The advantage of the proposed method is a small constraint on design and capability of correcting rotation of more than 90 degrees. Moreover the implementation as Android application realizes a real time extraction only by holding a cell phone up over a printed watermarked image. In the experiment, we implement the proposed method as an Android application and evaluate its performance including its real extraction time on a cell phone. The experimental results confirm that the proposed method can extract a correct watermark in 2 seconds, where this time constraint is based on the psychological research in [3]. This result shows that the proposed method is enough fast in practice. The rest of this paper consists of four sections. Firstly We describe the conventional method in Sec. II. Next we describe the proposed method in Sec. III. Then we show and discuss the performance of the proposed method in Sec. IV. Finally we conclude this paper in Sec. V.

II. Conventional method

In this section, we describe a conventional method proposed by Nakamura et al. in [2]. Nakamura's method could extract watermarks from printed watermarked images by using cell phones, where the computational time of one extracting procedure was in one second. There are various distortions on printed watermarked images compared with the corresponding watermarked images. Therefore extractor should have robustness against such distortions because they prevent extracting correct watermarks. Nakamura's method employed side trace algorithm (STA) proposed in [4] to correct geometrical distortions caused by camera angle, where a black frame was added to printed watermarked images. The conventional method also employed 2-dimensional sine pattern modulation so as to be robust against non-linear distortions caused by waving paper surface.

A. Embedding procedure

The inputs of embedding procedure are an original image I, a watermark of length L bits, and an embedding strength a, where I is a color image of size $W \times H$ pixels $I = \{I^R_{x,y}, I^G_{x,y}, I^B_{x,y}\} (0 \le x < W, 0 \le y < H)$.

978-1-4799-2190-4/13 $26.00 © 2013 IEEE
DOI 10.1109/ACPR.2013.79

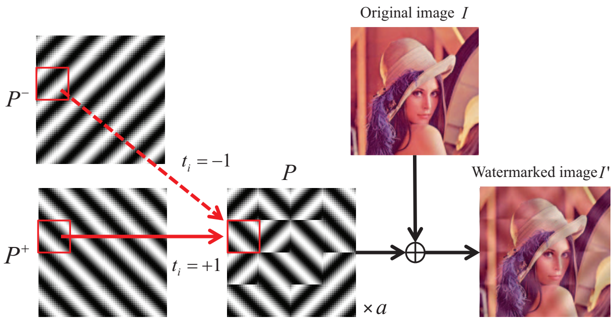

Fig. 1. Embedding procedure by using 2-dimensional sine pattern.

1) Error correcting code: An error correcting code is made from the watermark by using the watermark as information symbols. Then the error correcting code is regarded as a watermark bit sequence of length n, that is, $\boldsymbol{c} = \{c_j | c_j \in \{0,1\}, 0 \leq j < n\}$.

2) Spread spectrum modulation: Firstly, the watermark bit sequence $\boldsymbol{c}$ is transformed to a sequence of length N^2 as following:

$$\boldsymbol{b} = \overbrace{c_0, \ldots, c_0}^{l}, c_1, \cdots, c_{n-2}, \overbrace{c_{n-1}, \ldots, c_{n-1}}^{l} \quad (1)$$

where $l = N^2/n$ and N is a pre-defined integer value. Moreover $c_j = 0, 1$ are transformed to $b_k = -1, 1$ in $\boldsymbol{b} = \{b_k \in \{-1,1\} | 0 \leq k < N^2\}$.

Secondly, the direct sequence $\boldsymbol{s} = \{s_k | 0 \leq k < N^2\}$ is obtained by

$$s_k = b_k r_k \quad (2)$$

where r_k is an element of a pseudo-random sequence $\boldsymbol{r}$. $\boldsymbol{r}$ is constructed from $\boldsymbol{r^{(j)}} = \{r_i^{(j)} | r_i^{(j)} \in \{-1,+1\}, 0 \leq i < l, \sum_{i=0}^{l-1} r_i^{(j)} = 0\}$ for $0 \leq j < n$ in the following manner.

$$\boldsymbol{r} = r_0^{(0)}, \ldots, r_{l-1}^{(0)}, \cdots, r_0^{(n-1)}, \ldots, r_{l-1}^{(n-1)} \quad (3)$$

Finally, the elements of $\boldsymbol{s}$ are randomly rearranged. Then embedding sequence $\boldsymbol{t} = \{t_k | 0 \leq k < N^2\}$ is obtained. This rearrangement should be saved for extracting procedure.

3) 2-dimensional sine pattern modulation: Figure 1 shows the abstract of the embedding procedure. Firstly the 2-dimensional sine patterns P^- and P^+ in the left side of Fig. 1 are generated. The size of P^- and P^+ is same as the original image, where let F be the relative frequency of the patterns against the size of the image. In other word, F is the number of sine curves in one horizontal or vertical line in the patterns. P^- is same as P^+ rotated 90 degree.

Next the watermark pattern $P = \{P_{x,y}^{(h,v)}\}$ is generated from P^- and P^+ in the following manner. $\{P_{x,y}^{-(h,v)}\}$ and $\{P_{x,y}^{+(h,v)}\}$ are respectively obtained by dividing both of P^- and P^+ into blocks of which the number is $N \times N$, where (h, v) is the location of a block ($0 \leq h < N, 0 \leq v < N$) and (x, y) is the location of the corresponding pixel in the image. Then $\{P_{x,y}^{-(h,v)}\}$ or $\{P_{x,y}^{+(h,v)}\}$ is selected as the element of P by the following equation.

$$P_{x,y}^{(h,v)} = \begin{cases} P_{x,y}^{-(h,v)} & \text{if } t_{h+vn} = -1 \\ P_{x,y}^{+(h,v)} & \text{if } t_{h+vn} = +1 \end{cases} \quad (4)$$

where t_{h+vn} is the element of embedding sequence obtained in the previous section and $h + vn$ means raster scan order.

4) Adding embedding pattern to original image: The watermarked image $I' = \{I'^R_{x,y}, I'^G_{x,y}, I'^B_{x,y}\}$ is obtained by adding the watermark pattern P to the original image I as follows :

$$\begin{aligned} I'^R_{x,y} &= I^R_{x,y} + aP_{x,y} \\ I'^G_{x,y} &= I^G_{x,y} + aP_{x,y} \\ I'^B_{x,y} &= I^B_{x,y} + aP_{x,y} \end{aligned} \quad (5)$$

where a is the embedding strength pre-defined in the beginning of Sec. II-A.

B. Syncronization with frame

A watermarked photograph is obtained by taking a picture of a printed watermarked image by a camera. The watermarked photograph includes geometrical distortions caused by camera angle. Side trace algorithm (STA) with a black frame proposed in [4] is used for correcting the geometrical distortions. In STA, a black frame is firstly added to a watermarked image in advance. Then the watermarked image with the black frame is printed out and the watermarked photograph is taken from it. STA can detect four corners of the area of the watermarked image from the watermarked photograph. Finally the target image for extraction is obtained by applying projective transformation to the area surrounded by the four corners.

STA uses the difference of contrast between the black frame and a background. Therefore STA cannot correctly detect the area of the watermarked image when there is little contrast between the frame and the background or too many edges in the background or the watermarked image. Moreover STA cannot correct the rotation over 90 degree. As a result, the background and layout of printed materials are limited.

C. Extracting procedure

1) Transform to gray scale and normalize of size: The target color image for extraction is transformed to gray scale by calculating the mean of R, G, and B at each pixel. Then the size of the target gray scale image for extraction is normalized to $4F \times 4F$, where F is the relative frequency defined in Sec. II-A3. The employment of normalized size $4F$ makes the following pre-processing effective.

2) Pre-processing: Pre-processing is composed of convolution process and clip process. At first, the convolution process using the operator shown in Fig. 2 is applied to the normalized image. The sine patterns in both of P^- and P^+ are emphasized by using the operator because the wavelength of sine curves in P^- and P^+ is equal to 4 due to the normalized width and height $4F$. Next clip process makes pixel values 1, -1 or 0 based on the sign of the pixel values. Figure 3 shows the example after pre-processing, where the pixel values in Fig. 3 are replaced $\{-1, 0, 1\}$ with $\{0, 128, 256\}$ respectively.

-1		+1
+1		-1

Fig. 2. Operator for pre-processing filter.

Fig. 3. Example after pre-processing.

3) 2-dimensional sine pattern demodulation: The pre-processed image obtained in Sec. II-C2 is divided into blocks of which the number is $N \times N$, where let $M = 4F/N$ be the width and height of the blocks. The correlation values $E^-_{h,v}$ and $E^+_{h,v}$ of P^- and P^+ are respectively obtained for all blocks in the following manner. At first, $e^{-(h,v)}_{x',y'}$ and $e^{+(h,v)}_{x',y'}$ are respectively calculated by the convolution between the block and the convolution matrix C^- and C^+ shown in Fig. 4, where $0 \le x' < M$ and $0 \le y' < M$. Next $E^-_{h,v}$ and $E^+_{h,v}$ are obtained from $e^{-(h,v)}_{x',y'}$ and $e^{+(h,v)}_{x',y'}$ by the following equation.

$$E^-_{h,v} = \sum_{x'=0}^{M-1} \sum_{y'=0}^{M-1} |e^{-(h,v)}_{x',y'}| \quad (6)$$

$$E^+_{h,v} = \sum_{x'=0}^{M-1} \sum_{y'=0}^{M-1} |e^{+(h,v)}_{x',y'}| \quad (7)$$

Finally the extraction matrix $D = \{d_{h,v}\}$ of size $N \times N$ is obtained by the following equation.

$$d_{h,v} = E^+_{h,v} - E^-_{h,v} \quad (8)$$

4) Weighting: In this section, weight $w_{h,v}$ of $d_{h,v}$ is calculated in the following manner so that the components of 2-dimensional sine pattern in original images are reduced because such components are regarded as strong noise against 2-dimensional sine pattern added in embedding procedure. At first, the mean $q_{h,v}$ of pixels p in a block $B_{h,v}$ is calculated for all blocks in the image before clip process in Sec. II-C2.

$$q_{h,v} = \frac{1}{M^2} \sum_{p \in B_{h,v}} |p| \quad (9)$$

Then the weight $w_{h,v}$ is calculated from $q_{h,v}$ by the following equation.

$$w_{h,v} = f(q_{h,v}) \quad (10)$$

where $f(x)$ is monotone decreasing function.

5) Spread spectrum demodulation: The weighted extraction matrix $D' = \{w_{h,v} d_{h,v}\}$ is transformed to 1-dimensional sequence $\boldsymbol{g} = \{g_k | 0 \le k < N^2\}$ in raster scan order. Then a sequence $\boldsymbol{h} = \{h_k | 0 \le k < N^2\}$ is obtained by the rearrangement of $\boldsymbol{g}$ in the inverse order of the random arrangement in

		-1		
	-1		+2	
-1		+2		-1
	+2		-1	
		-1		

C^-

		-1		
	+2		-1	
-1		+2		-1
	-1		+2	
		-1		

C^+

Fig. 4. Convolution matrix for 2-dimensional sine pattern demodulation.

Sec. II-A2. Next partial sequences $\boldsymbol{x^{(j)}} = \{x_i^{(j)} | 0 \le i < l\}$ of $\boldsymbol{h}$ are obtained by the following sequence.

$$x_i^{(j)} = h_{i+jl} \quad (11)$$

where $l = N^2/n$ is the length of $\boldsymbol{x^{(j)}}$. Moreover normalized $\boldsymbol{x^{(j)}}$ is defined as a extraction sequence, where the mean and variance of the normalized $\boldsymbol{x^{(j)}}$ is 0 and 1 respectively. Finally watermark bits c'_j are extracted based on the correlation value ρ_j between pseudo random sequence $\boldsymbol{r^{(j)}}$ and $\boldsymbol{x^{(j)}}$ by the following equation.

$$c'_j = \begin{cases} 0, & \text{if } \rho_j < 0 \\ 1, & \text{if } \rho_j \ge 0 \end{cases} \quad (12)$$

$$\rho_j = \sum_{i=0}^{l-1} x_i^{(j)} r_i^{(j)} \quad (13)$$

where $\boldsymbol{r^{(j)}}$ is same as the pseudo random sequence $\boldsymbol{r}$ used for embedding procedure in Sec. II-A2.

6) Error correcting decode: A watermark of length L is obtained by error correcting decode of $\boldsymbol{c'} = \{c'_j\}$.

III. PROPOSED METHOD

In this section, we describe the proposed method and the implementation of it as Android application. In Sec. III-A and Sec. III-B, error correcting procedure and location detection with finder pattern and alignment pattern of the proposed method are described respectively. Then we describe the implementation as Android application in Sec. III-C.

A. Error correcting coding and decoding

In the proposed method, we use the following manner of error correcting coding and decoding instead of the manner described in Sec. II-A1 and Sec. II-C6 respectively. In error correcting coding, we obtain a watermark bit sequence of length 32 bits from a watermark of length 4 bits in the following manner. At first, 16 kinds of watermark bit sequences $\boldsymbol{\phi^m}$ of length 32 are obtained from 16 rows in Hadamard matrix, where $0 \le m < 16$. The hamming distance between the 16 watermark bit sequences $\boldsymbol{\phi^m}$ is equal to 16 for any combination of them except for same sequences because the rows of Hadamard matrix are orthogonal each other. The watermarks of length 4 bits are assigned to 16 watermark bit sequences one by one. In error correcting decoding, we calculate each hamming distance between an extracted watermark bit sequence and all watermark bit sequences described above. Then the watermark corresponding to the watermark bit sequence which has a minimum hamming distance if the minimum hamming distance is smaller than T, where T is a pre-defined constant value. We regard a extraction as extraction failur if the hamming distance is larger than T.

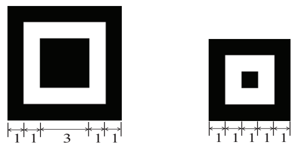

Fig. 5. Finder pattern(left) and alignment pattern(right).

Fig. 6. Example of a watermarked image with patterns for correction.

B. Location detection with finder pattern and alignment pattern

We employ the finder pattern and alignment pattern of QR code. Figure 5 shows a finder pattern and an alignment pattern. The arrangement of black and white pixels in a finder pattern is 1:1:3:1:1 for all angles. We can detect four corners of the distorted watermarked photograph by adding three finder patterns and one alignment pattern to a watermarked image in advance, where the direction of the watermarked photograph is also detected. Figure 6 shows an example of a watermarked image with finder patterns and an alignment pattern. Shown in Fig. 6, three finder patterns are arranged at the upper-left, upper-right and lower-left corners, while an alignment pattern is arranged at the lower-right corner. It helps the detection of the direction of the watermarked image that only one alignment pattern is arranged at one of the corners. Figure 7 and 8 show the comparison the watermarked images of the proposed method with those of the conventional method. Shown in Fig. 7, the design of the watermarked image of the conventional method becomes worse than that of the proposed method. Moreover, shown in Fig. 8, the proposed method can detect the watermarked area in an image by using finder patterns and an alignment pattern, while the conventional method cannot detect it because some edges in the image interfere the detection.

C. Implementation as Android application

In this section, we describe the implementation of the proposed method as Android application. The application extracts a watermark from a printed watermarked image only by holding the Android cell phone up over the printed watermarked image in real time. Therefore the application continues to focus its camera on the printed watermarked image and to extract a watermarks any number of times as possible. Moreover the extraction on the application does not need any communication to other devices. Then the application can work offline. The extracting procedure is composed of the location detection, the correction by projective transformation

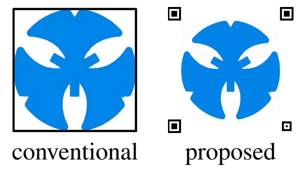

Fig. 7. Example for applying each method to a logo image.

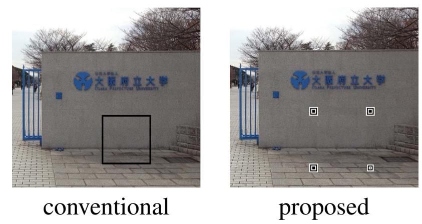

Fig. 8. Applying each method to a part of an image.

and watermark extraction. We describe information on display screen in addition to above three processes.

1) Location detection: We use Zxing to detect the coordinates of four corners of the watermarked area in the distorted watermarked photograph, where Zxing is the library for barcode reader released by Google,. Here the methods for detecting the coordinates of the center of finder patterns and an alignment pattern in Zxing are used.

2) Correction by projective transformation: The corrected watermarked area is obtained by the projective transformation the area surrounded by center points of three finder patterns and one alignment pattern to the square of pre-difined size.

3) Watermark extraction: We use the corrected watermarked area obtained in the previous section as the target image for extraction. The extracting procedure before error correcting decode is same as that of the conventional method. We obtain a watermark from the extracted watermark bit sequence in the manner described in Sec. III-A

4) Information on display screen: Figure 9 shows a screen shot of the proposed Android application. The green square in the center of the display screen is displayed at all time. The application detects the watermarked area displayed as a pink square when finder patterns and an alignment pattern come into the green square. After the succession of the detection of the watermarked area, the gray-scale target image for extraction is displayed at the upper-left of the display screen. Then the extraction procedure is applied to the gray-scale target image for extraction. The extracted watermark is displayed at upper-center of the display screen if the extraction result is not extraction failure.

IV. Experiments

A. Processing time

We investigated the processing time of extraction procedure on a real Android cell phone, where the processing time is the mean value of 100 times trial. Galaxy S III GT-I9300 is used for the experiment as Android cell phone. Table I shows the mean processing time of the extraction on the real Android cell phone. Shown in Tab. I, the mean of total processing time

Fig. 9. Display screen.

TABLE I. MEAN PROCESSING TIME (100 TIMES TRIAL).

Process	time [ms]
Location detection	97.99
Correction	232.77
Watermark extraction	215.70
Total	546.46

per a extraction is 546 [ms]. The extraction procedure can work 3 times per two seconds, which is allowed time to use comfortably. This result confirms that the implementation of the proposed method is enough fast to use in practice.

B. Tradeoff between image quality and robustness

We investigated the extraction rates to evaluate the tradeoff between image quality and robustness, where the used parameters were $1 \leq a \leq 12$, $N = 32$, $F = 48$, $f(x) = 1/x + 0.25$ and $T = 5$. The extraction rates were calculated for each original image with each a as the mean of 100 times trial. Extraction rate means the rate of correct extraction to all extraction, where there is no incorrect extraction without extraction failure. We used two color images "Lena" and "OPU" of size 512×512 pixels as original images, where the used sizes of printed watermarked images were 5 and 15 centimeters squares by the laser printer EPSON LP-M5000. The cell phone was held up over the printed watermarked image so that the camera of the cell phone caught the whole of each images. Figure 10 shows the original image "OPU" and the watermarked images with $a = 4, 8, 12$. Shown in Fig. 10, the degradation caused by embedding procedure was not so perceptible. The degradation in watermarked images with $a = 8, 12$ was not so perceptible in normal viewing although it was perceptible in close viewing. Table II shows the extraction rates. Shown in Tab. II, the extraction rates for printed size 15[cm] were higher than those for 5[cm]. The reason is that smaller printed size causes relatively larger effect to prevent extracting from the degradation just by printing and waving paper surface and the camera shake by closer shot. The extraction rates of "OPU" was lower than those of "Lena" for the same embedding strength a. It is because the watermark embedded into the white background of "OPU" disappears after printing. This problem can be solved by using appropriate a to the feature of original images because the proposed method requires no embedding strength a for extracting. A watermark can be correctly extracted in 2 seconds if one of three extractions succeeds because three extractions can work in 2 seconds according to the previous section. Therefore extraction rates over 64% is enough to achieve an extraction rate over 95% in 2 seconds extraction. From this viewpoint, the following embedding strengths a are suitable, that is, $a = 5$ for "Lena" of 5[cm], $a = 3$ for "Lena" of 15[cm], $a = 9$ for "OPU" of 5[cm] and $a = 5$ for "OPU" of 15[cm].

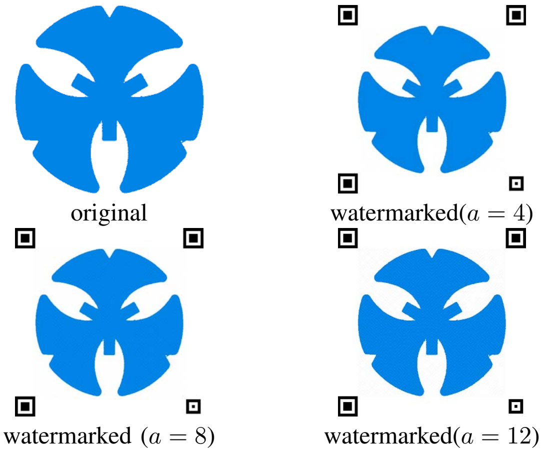

Fig. 10. Original image and watermarked images (OPU)

TABLE II. EXTRACTION RATE[%](100 TIMES TRIAL).

a	Lena		OPU	
	5cm	15cm	5cm	15cm
1	0	0	0	0
2	0	18	0	0
3	0	87	0	9
4	0	95	0	53
5	64	100	2	98
6	83	100	24	100

a	Lena		OPU	
	5cm	15cm	5cm	15cm
7	97	100	44	100
8	100	100	55	100
9	100	100	96	100
10	100	100	100	100
11	100	100	100	100
12	100	100	100	100

V. CONCLUSION

We have proposed a new watermarking method using finder patterns and an alignment pattern of QR code for correction. Moreover we implemented the proposed method as Android application so as to use it on a real cell phone. Then we investigated the performance of processing time and tradeoff between image quality and robustness of our method on a real cell phone. The experimental results confirm that the implemented application can extract a correct watermark in 2 seconds with appropriate embedding strengths a, where the appropriate a cause imperceptible degradation on watermarked images. Our future work should be consideration to the shape and arrangement of patterns for correction, a kind of error correcting code and enlarging the amount of watermark bits.

ACKNOWLEDGMENT

This research was supported in part by the Grant-in-Aid for Young Scientists (b)(24700175) from Japan Society for the Promotion of Science (JSPS) and A-STEP FS stage : Exploratory Research (AS242Z02577H) from Japan Science and Technology Agency.

REFERENCES

[1] QRcode.com, http://www.qrcode.com/en/

[2] T.Nakamura, A.Katayama, M.Yamamuro, and N.Sonehara, "Fast watermark detection scheme from analog image for camera-equipped cellular phone," IEICE Trans. D-II, vol.J87-D-II, no.12, pp.2145–2155, 2004.

[3] R.B.Miler, "Response time in man-computer conversational transactions," Proc. AFIPS Fall Joint Computer Conference, vol.33, pp.267–277, 1968.

[4] A.Katayama, T.Nakamura, M.Yamamuro, and N.Sonehara, "A new high-speed corner detection method: side trace algorithm(STA) for i-appli to detect watermarks," IEICE Trans. D-II, vol.J88-D-II, no.6, pp.1035–1046, 2005.