A Visible Video Data Hiding Scheme Based on Fade-In and Fade-Out Effects Utilizing Barcodes

Tetsuya KOJIMA\(^{1}\), Senior Member and Kento AKIMOTO\(^{1}\), Nonmember

**SUMMARY** Data hiding techniques are usually applied into digital watermarking or digital fingerprinting, which is used to protect intellectual property rights or to avoid illegal copies of the original works. It has been pointed out that data hiding can be utilized as a communication medium. In conventional digital watermarking frameworks, it is required that the difference between the cover objects and the stego objects are quite small, such that the difference cannot be recognized by human sensory systems. On the other hand, the authors have proposed a ‘hearable’ data hiding technique for audio signals that can carry secret messages and can be naturally recognized as a musical piece by human ears. In this study, we extend the idea of the hearable data hiding into video signals by utilizing the visual effects. As visual effects, we employ fade-in and fade-out effects which can be used as a kind of visual rendering for scene transitions. In the proposed schemes, secret messages are generated as one-dimensional barcodes which are used for fade-in or fade-out effects. The present paper shows that the proposed schemes have the high accuracy in extracting the embedded messages even from the video signals captured by smartphones or tablets. It is also shown that the video signals conveying the embedded messages can be naturally recognized by human visual systems through subjective experiments.

**key words:** video data hiding, visual effect, fade-in, fade-out, barcode

1. **Introduction**

Data hiding is a technique to embed secret messages into multimedia contents such as images, audio and video signals without being recognized by human sensory systems\(^{[2]}\). The cover objects are the original works which does not convey any messages, while the stego objects are defined as the works where secret messages are embedded by data hiding techniques. Data hiding techniques are usually applied into digital watermarking or digital fingerprinting, which is used to protect intellectual property rights or to avoid illegal copies of the original works. On the other hand, it has been pointed out that data hiding can be utilized as a communication medium. In other words, messages are secretly embedded into multimedia contents that can be recorded by receivers such as smartphones and tablets. The extracted messages can be displayed on the screens of such receivers.

Therefore, the message is transmitted without any cables or electro-magnetic waves, but with multimedia contents. The authors have proposed data hiding techniques for such communications purposes, for example, disaster prevention broadcasting systems using audio data hiding\(^{[3]}\), \(^{[4]}\), an audio digital watermarking based on distortion effects\(^{[5]}\), \(^{[6]}\), and so on.

In data hiding techniques, the property called ‘transparency’ is usually quite important. In other words, it is required that the difference between the cover objects and the stego objects are quite small, such that the difference cannot be recognized by human sensory systems. On the other hand, a ‘hearable’ data hiding scheme for musical pieces called “Tone Code” has been proposed by the authors\(^{[7]}\). In this scheme, secret messages are modulated into one of the musical components such as chords, rhythms, melodies, and so on. The stego musical piece can be naturally recognized by human ears as one of the components of the musical piece played on additional electronic instruments such as synthesizers. In this scheme, the stego object is apparently recognized as a different one from the cover objects even by human recognition systems. In this sense, we call this scheme as ‘hearable’ data hiding.

In this study, we extend the idea of the hearable data hiding into video signals by utilizing the visual effects. As visual effects, we employ fade-in and fade-out effects which can be used as a kind of visual rendering for scene transitions. In the proposed schemes, secret messages are generated as one-dimensional barcodes which are used for fade-in or fade-out effects. The stego video signals conveying secret messages are apparently different from the cover video signals. It is expected, however, that these visual transition effects can be naturally perceived by human visual systems since the barcodes conveying messages gradually appear or disappear with time. In this sense, the proposed scheme can be called as a ‘visible’ video data hiding\(^{[1]}\).

The present paper shows that the proposed schemes have high accuracy in extracting the embedded messages even from the video signals captured by smartphones or tablets. It is also shown that the video signals conveying the embedded messages can be naturally recognized by human visual systems through subjective evaluation experiments. These results imply that the proposed schemes can be applied into actual solutions such as video advertising on the digital signages or on the web. To the best of the authors’ knowledge, the concept of such ‘visible’ video data hiding techniques utilizing barcodes have not been proposed in pre-
vious studies.

2. Proposed Video Data Hiding Scheme

In this section, we propose a visible video data hiding scheme based on fade-in and fade-out effects utilizing one-dimensional barcodes. In the proposed scheme, a barcode conveying a secret message is displayed gradually with time as fade-in and fade-out effects at a scene transition of the video signal. In the following, we describe the embedding and extracting procedures of the proposed scheme, and discuss the embedding capacity of the scheme.

2.1 One-Dimensional Barcodes

We employ Code 128 as a one-dimensional barcode. Code 128 is one of the high-density barcodes and defined in ISO/IEC 15417:2007 [8]. It can encode all of the 128 ASCII characters as well as the Latin-1 characters defined in ISO/IEC 8859-1 [9]. In Code 128, a barcode consists of four components, that is, a start symbol, a message, a check digit, and the stop symbol. Each symbol except for the start symbol in the barcode is composed of three bars and three spaces. The width of each bar or space is 1, 2, 3 or 4 units. The sum of the bar width must be even units, while the sum of the space width must be odd units. The total size of bars and spaces is 11 units. For example, the character ‘A’ is represented by the 11-bit pattern ‘10100011000’, where the first unit is the start symbol and the check digit of the barcode is composed of three bars and three spaces. The width of each bar or space is 1, 2, 3 or 4 units. The sum of the bar width must be even units, while the sum of the space width must be odd units. The total size of bars and spaces is 11 units. For example, the character ‘A’ is represented by the 11-bit pattern ‘10100011000’, where the first unit is the start symbol and the check digit of the barcode is composed of three bars and three spaces. Note that the sum of the 1’s and 0’s are even (4) bits and odd (7) bits, respectively.

There are three types of start symbols in Code 128, that is, Code-A, Code-B and Code-C. In this study, we employ Code-C. In Code-C, two-digit decimal numbers from 00 to 99 are employed instead of ASCII codes. The stop symbol is denoted by ‘1100011101011’ of 13 units length. The check digit is calculated modulo 103 from the remainder of a division of the message by 103. Let $l_M$ be the digit length of $M$, that is, $l_M = \lfloor \log_{10} M \rfloor + 1$. Every two digits of $M$ from the least significant digit is transformed into an 11-unit barcode in Code-C of Code 128. In this paper, each two-digit number of $M$ is called a message symbol corresponding to an 11-unit barcode. If $l_M$ is odd, a ‘0’ is added at the left of the most significant digit. The barcode width $L$ is, then, given as

$$L = 11 \times \left(2 + \left\lceil \frac{l_M}{2} \right\rceil \right) + 13,$$

(2)

where ‘2’ implies the start symbol and the check digit of length 11 units while ‘13’ implies the length of the stop symbol. A unit size of the barcode corresponds to a one-pixel width in a frame of the cover video signal.

2.2.2 Expansion of the Barcode Width

The barcode width is expanded into $L_k = k \times L$, where $k$ is a positive integer called an expansion rate. When $W$ stands for the frame width of the cover video signal, it is obvious that the barcode width should satisfy

$$L_k \leq W.$$

(3)

The value of $k$ should be carefully chosen. If $k$ is too small, a barcode is shown in a narrow area at the center of the frame. This may make the fade-in or fade-out effects quite uncomfortable. In addition, a small $k$ implies that the lengths of bars and spaces of the barcodes are also small, which may
make it difficult to capture the shown barcodes successfully by the camera at the receiver. Therefore, the value of $k$ should be as large as possible satisfying Eq.(3). In the experiments given in this paper, we take two different values for $k$, and compare the results for both cases.

2.2.3 YCC Transformation of a Video Frame

Every frame of the cover video signal is transformed from RGB format into YCbCr format. The barcode is embedded only into Y component. In the following, we call the Y component of a frame as ‘an image’.

2.2.4 Embedding as a Fade-In Effect

The barcode is displayed at the center of the frame of the cover video signal. In other words, the widths of the right and the left margin outside of the barcode is even. The Y component of every pixel at the first frame of the cover video signal is set to zero. Then, $v$ columns are randomly chosen from the frame of the cover video signal except for the columns corresponding to the bars of the barcode. The selected $v$ columns are replaced with Y components of the corresponding parts in the frame of the cover video signal. When all columns except for those corresponding to the bars of the barcode are replaced with the original Y components of the cover signal, the whole barcode can appear on the frame. A whole barcode is displayed for $t_d$ frames since it is easy to capture the barcode from the video signal. After $t_d$ frames, randomly chosen $v$ columns from the frames of the cover video signal except for the columns corresponding to the bars of the barcode are set to zero. This procedure repeats until Y components of all of the columns except for those corresponding to the bars of the barcodes are set to zero. Then the fade-out effect completes and the whole frame is turned into black. In Fig. 3, the transition from right-hand-side to the left-hand-side corresponds to the fade-out effect.

A sample video clip of the stego video signal including both of fade-in and fade-out effects is attached to this paper. It can be seen that two different scenes are switched through fade-out and fade-in effects in the sample clip.

As shown in the sample clip, the barcode conveying a message gradually appears or disappears on a cover video signal in the proposed method. The scene transitions realized by the proposed method are quite similar to fade-in or fade-out effects obtained by overlaying random vertical stripes. While the whole barcode is shown for $t_d$ frames, such frames seem completely same as the video signal obtained by overlaying the barcode on the cover video signal with maintaining the spaces of the barcode transparent. Therefore, it is expected that the scene transition of the stego video signal can be naturally recognized by human visual systems. On the other hand, some errors may occur in decoding the barcode since the spaces of the barcode is not completely white, but transparent. This is not the case for the standard barcodes printed in black and white. In addition, note that there is a trade-off between the extraction accuracy and the visual image quality. If $t_d$ is large, the barcode is shown for a long time, which makes the transition

2.2.5 Embedding as a Fade-Out Effect

First, $v$ columns are randomly chosen from the frame of the cover video signal corresponding to the bars of the barcode. The Y components of the selected $v$ columns are all set to zero. When all columns corresponding to the bars of the barcode are turned into black, the whole barcode can appear on the frame. A whole barcode is displayed for $t_d$ frames similarly as the fade-in effect. After $t_d$ frames, randomly chosen $v$ columns from the frames of the cover video signal except for the columns corresponding to the bars of the barcode are set to zero. This procedure repeats until Y components of all of the columns except for those corresponding to the bars of the barcodes are set to zero. Then the fade-out effect completes and the whole frame is turned into black. In Fig. 3, the transition from right-hand-side to the left-hand-side corresponds to the fade-out effect.

![Fig. 2](image)

Fig. 2 An example of a snapshot of a stego video signal [1].

![Fig. 3](image)

Fig. 3 An example of the fade-in and fade-out effects [1].
effects quite unnatural. On the contrary, it is expected that the message can be correctly extracted by employing a large $t_d$. Therefore, the staying time $t_d$ of the barcode should be as small as possible while the message can be successfully extracted.

2.3 Extracting Procedure

The extracting procedure of the proposed scheme can be given as follows. In the embedding procedure given in Sect. 2.2, the case for fade-in and fade-out effects are slightly different. In extracting, however, a common procedure can be applied for the both effects. Note that it is assumed that the parameters such as $k$, $L_k$, $t_d$, $v$, and the embedded bit length are shared in advance between the embedding and the extracting sides. Especially, the sharing of $k$ and the bit length is important to avoid the case where the wrong messages with short lengths can be accidentally extracted. The following procedure is applied to all of the frames of the recorded video signal.

2.3.1 YCC Transformation of a Video Frame

The process same as Sect. 2.2.3 is applied to a frame of the recorded video signal.

2.3.2 Space Synchronization of the Frame

In this study, we consider the case where the stego video signal is recorded by smartphones or tablets. In this case, the frame size of the recorded video signal is not always same as that of the cover video signal since the video signals cannot be always recorded directly from the front. In the proposed method, the frame sizes of the cover and the stego video signals should be completely same in order to extract the embedded message correctly. Therefore, the frame size of the stego video signal must be correctly estimated. We call this process as ‘space synchronization’ in this paper. In this study, we employ a method called STA (Side Trace Algorithm)\cite{10},\cite{11} utilizing a kind of projective transformation. STA algorithm assumes that the displayed stego video signals are rectangular, and centered in the camera frame. In this algorithm, a recorded frame is first scanned from both of the left-end and right-end to find the left and the right edges of the rectangular-shaped stego video signal. Once the left and the right edges are found, the recorded frame is scanned upward and downward to identify the four corners of the stego video signal. After the four corners are identified, the stego video signal is mapped to a rectangle of the designated size. See Refs.\cite{10},\cite{11} for the details of the algorithm.

2.3.3 Decoding of the Barcode

After capturing a barcode from the recorded video signal, the transmitted message is decoded by the standard decoding algorithm for Code 128. For decoding the embedded barcode, we employ \textit{pyzbar}\cite{12}, a Python-based library that can read one-dimensional barcodes and QR codes. After space synchronization by STA, it is possible to obtain and decode one-dimensional barcodes by \textit{pyzbar}. In the proposed method, first, the start symbol is retrieved. If the start symbol is not Code-C, the recorded frame is rejected and the next frame is employed for decoding. If the start symbol is Code-C, the next 11-bit barcodes to the right of the start symbol are transformed into a sequence $M'$ of two-digit decimal numbers. Then $M'$ is transformed into the binary sequence of the recovered message $m' \triangleq \{m'_0, m'_1, \ldots, m'_{n-1}\}$. This process is repeated until the check digit and the stop symbol are detected. If the decoded message is same as the one obtained from the previous frame, the message is discarded. If there are any decoding errors, no messages are output. Therefore, the embedded message cannot be extracted in this case.

2.4 Capacity

In the proposed scheme, the capacity, that is, the maximum size of the embeddable message depends on the frame size of the cover video signal. Here we discuss the maximum bit length $N_b$ according to the frame width $W$ and the expansion rate $k$ of the barcodes. The minimum size of a barcode is 11 units corresponding to 11 pixels. The additional 35 pixel is required to represent a message since the start symbol, the check digit and the stop symbol are needed. As defined in Sect. 2.2.1, a message symbol is a two-digit number divided from the decimal number $M$ corresponding to the original message. Therefore, the maximum number of message symbols $N_{m}$ embeddable by the proposed scheme can be given as

$$N_m = \left\lfloor \frac{W - 35k}{11k} \right\rfloor.$$  \hskip 0.5cm (4)

where $k$ is the given expansion rate. A message symbol is represented as a two-digit decimal while the large decimal number obtained by concatenating these two-digit numbers corresponds to the whole embedded message. In this regard, the capacity of the proposed scheme can be evaluated as

$$N_b = \left\lfloor \log_2 100^{N_m} \right\rfloor,$$  \hskip 0.5cm (5)

in bits. For example, if the frame size is 1,280 pixels and $k = 3$, $N_m = 35$ symbols can be embedded and the capacity is $N_b = 232$ bits.

3. Experimental Results

3.1 Simulation Results

The performance of the proposed scheme is evaluated through computer simulations. We assume that the stego signals are shown on the displays or digital signages, and that the embedded messages are extracted from the video
signals recorded by smartphones or tablets. In the simulations, however, the messages are extracted directly from the stego video signals without capturing them by smartphones or tablets. As the cover video signals, we use the video clips provided in NHK Creative Library [13] as well as the signals where the same still images are repeated. In the latter case, the images are selected from the standard images at SIDBA [14]. We employ 10 repeated still images from SIDBA and 5 video clips from NHK Creative Library shown in Figs. 4 and 5, respectively. The parameters for the simulations are shown in Table 2.

As shown in Table 2, the parameters \( k = 2 \) and 3 are used for SIDBA images while \( k = 3 \) and 4 are used for the video clips from NHK Creative Library. For example, assume that \( k = 2 \) is employed for SIDBA images. In this case, a 13-bit message corresponds to a four-digit decimal numbers according to Eq. (1). Therefore, the barcode consists of two message symbols as well as start, stop symbols and check digit. The barcode width is given as \( 4 \times 11 + 13 = 57 \) units according to Eq. (2). If the expansion rate \( k \) is set to 2, the barcode width is 114 pixels while the width is 171 pixels for \( k = 3 \).

In addition, the staying time of barcodes are set to 1. As described in Sect. 2.2.5, the parameter \( t_d \) should be kept as small as possible while the extraction of the message is
successful. In the preliminary experiments, we have confirmed that there are few extraction errors even in the case of $t_d = 1$. Note that the parameter $v$, that is, the appearance rate of the barcode is related to the total frame length required for fade-in and fade-out effects. For example, assume that the total lengths of bars and spaces are equal in a barcode. If we consider the fade-in effect for a video clip from NHK Creative Library under the assumption of $k = 4$, there are 510 columns each corresponding to bars and spaces of the barcode. Since $v = 30$ as shown in Table 2, 17 frames are required to replace the spaces of the barcode with the Y components of the cover video signal. Then, the whole barcode is shown for a single frame, followed by another 17 frames required to removing all the black columns. Therefore, the total frame length for a single fade-in effect is 35 frames in this case, which implies that the fade-in effect takes around 0.5 seconds in time.

The results for the both cases are shown in Tables 3 and 4. In these tables, ✓ and X stand for the case where the messages are successfully extracted, and the case where the messages are not correctly extracted, respectively. In general, the decoding of Code 128 results in three cases, that is, (i) the case where the message is correctly decoded, (ii) the case where no message is detected, and (iii) the case where a wrong message is decoded. In the proposed method, the case (iii) can hardly happen since a piece of the barcodes gradually appears or disappears in fade-in and fade-out effects. In addition, it is assumed that the parameter $k$ and the embedded bit length are shared in advance between the embedding and the decoding sides. It is quite rare that such a barcode with partially missing parts can satisfy the check digit even if the message symbols of the barcode are mistakenly decoded as a wrong message. Actually, there are no cases (iii) observed in our experiments. Therefore, the case X corresponds to the case (ii) only in Tables 3 and 4.

From these results, the extraction of the message is almost successful when the barcode width is relatively short. Even in the short barcode case, the results for the video signals J and N are not successful. Figure 6 shows a captured frame of the cover video signal J and the corresponding frame of the stego video signal. It is considered that some of the dark parts of the cover video signals are mistakenly recognized as the bars of the barcode since the luminance of the whole image is quite low. As shown in Fig. 5, it should be noted that the video clip N also include lots of dark pixels which seem almost completely black.

In addition, when the barcode width is long, the extraction can be failed against our expectation described in Sect. 2.2.2. In these cases, the space width of the barcode becomes long as the whole barcode width. Therefore, some dark parts are also mistakenly recognized as the bars of the barcode. Especially in the video signals B and C, some vertical straight lines with dark colors may be mistakenly recognized as the bars of the barcode. This implies that the proposed scheme has high accuracy in extraction when the luminance of the cover video signals is not very low and when there are no vertical straight lines that can be seen like black bars.

### Results for the Video Signals Captured by Cameras

Now we evaluate the performance of the proposed scheme when the displayed video signals are captured by smartphones or tablets. We employ the four video clips, K, L, M and O shown in Fig. 5, where the extraction is successful in Table 4. In this experiment, we capture the video clips shown on the display directly from the front (0°), and also capture them diagonally ($-20°$, $+20°$) as shown in Fig. 7. The examples of the snapshots of the captured video signals are shown in Fig. 8. Note that the whole stego video signals are captured in order to avoid the clipping effects. The parameters for the simulations are shown in Table 5. We use ASUS VZ229H 21.5-inch monitor and Sony Xperia XZs as the display and the smartphone, respectively. The specifica-

| Table 3 | Simulation results for the repeated still images from SIDBA. |
|---------|---------------------------------------------------------|
| Barcode | A | B | C | D | E | F | G | H | I | J |
| width   |   |   |   |   |   |   |   |   |   |   |
| 110     | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | X |
| 165     | ✓ | X | X | ✓ | ✓ | ✓ | ✓ | ✓ | ✓ | X |

| Table 4 | Simulation results for the video clips from NHK Creative Library. |
|---------|---------------------------------------------------------------|
| Barcode | K | L | M | N | O |
| width   |   |   |   |   |   |
| 759     | ✓ | ✓ | ✓ | X | ✓ |
| 1,012   | ✓ | ✓ | ✓ | X | ✓ |

![Fig. 6](image_url) The sample frames of the cover and the stego video signals for a case of unsuccessful extraction.

![Fig. 7](image_url) Angles to capture the displayed video signals in the experiment.
Table 5 Parameters for the experiments using the captured video signals.

|                          | Stego video signal | Captured video signal |
|--------------------------|--------------------|-----------------------|
| Resolution               | 1,280 × 720 pixels | 1,920 × 1,080 pixels  |
| Frame rate               | 60 fps             | 60 fps                |
| Embedded bit length      | 132 bits           |                       |
| Distance from the camera  | 51 / 71 cm         |                       |
| Capturing angles         | −20°, 0°, +20°     |                       |

Table 6 Specifications of the rear camera of the employed smartphone (Sony Xperia XZs).

|                        | Sony IMX400 Exmor RS |
|------------------------|----------------------|
| Sensor model           | 1 / 2.3"             |
| Sensor format          |                      |
| Pixel size             | 1.22 µm              |
| Aperture               | f/2.0                 |
| Focal length           | approx. 4.4 mm       |
|                        | approx. 25 mm (35 mm equivalent) |

Table 7 MOS scale for the subjective evaluation of the stego images.

| Scale | Description       |
|-------|-------------------|
| 5     | Completely no incongruity |
| 4     | Almost no incongruity |
| 3     | Moderate           |
| 2     | Some incongruity   |
| 1     | Strong incongruity |

Table 8 The distribution and the average of the MOS scores.

| Barcode width | 759 pixels | 1,012 pixels |
|---------------|------------|--------------|
| 5             | 31.5%      | 36.8%        |
| 4             | 42.1%      | 47.4%        |
| 3             | 15.8%      | 10.5%        |
| 2             | 10.5%      | 5.3%         |
| 1             | 0.0%       | 0.0%         |
| Averaged score| 3.95       | 4.16         |

The proposed data hiding scheme is one of the ‘visible’ data hiding. This implies that the stego video signals are obviously different from the cover video signals since the visible barcodes convey the embedded information. Therefore, we execute the subjective evaluation of the image quality instead of the objective evaluation. MOS shown in Table 7 is employed as the evaluation scale. Subjects are 19 students of ages from 18 to 22 years old. The stego video signals are same as those employed in Sect. 3.2. The experiments are executed for two barcode widths, that is, 765 and 1,020 pixels.

The distributions of the answered evaluation scores and their averages are shown in Table 8. These results show that 70% to 80% answers are positive in the both cases. It implies that the fade-in and fade-out effects in the stego video signals can be naturally recognized as visual effects by human visual systems.

4. Conclusion

The present paper proposes a new visible video data hiding scheme based on the the visual effects. As visual effects, we employ fade-in and fade-out effects, which are obtained by one-dimensional barcodes. The experimental results show that the embedded secret messages can be successfully extracted from the stego video signals even in the case where the video signals are captured by the camera functions of smartphones or tablets. In addition, it is shown that the fade-in and fade-out effects in the stego video signals can be naturally recognized as visual effects by human visual systems through the subjective evaluation experiments. These results imply that the proposed scheme has high effectiveness as a visible video data hiding technique and it can be applied into the applications like advertisements on the digital signage or on the web. To the best of the authors’ knowledge, there has been no proposals of the visible video data hiding techniques based on the visual effects such as fade-in or fade-out effects using one-dimensional barcodes.

As shown in Sect. 3.1, there are some cases where the embedded message cannot be successfully extracted because of the characteristics of the cover signals as well as error outputs by the barcode decoder. In our future study, we are planning to solve this problem by automatically selecting appropriate scenes from the cover video signals, where
the proposed scheme can be successfully applied.

In the present paper, we only use several video signals in the experiments. It is very important to assess the effectiveness of the proposed scheme by using various video clips in extraction of the secret messages as well as subjective evaluations. Especially in the subjective evaluations, it is required to compare the stego video signal generated by the proposed method with the video signals with fade-in and fade-out effects obtained by overlaying random vertical stripes. It is also considered that the performance of the proposed method can be affected by some conditions, such as averaged brightness of each video frame, subjects’ movements in each frame, capturing angles, exposures, camera shakes as well as video compressions. It is quite important to evaluate the performance of the proposed method under these various environments.

Besides, the authors are planning to apply the proposed method into the version using two-dimensional barcodes such as QR codes in the future study.

Acknowledgments

This work was supported by JSPS KAKENHI Grant Number 18K11609. The authors sincerely thanks to Prof. Akira Tanaka and Prof. Raito Matsuzaki, National Institute of Technology, Tokyo College, Japan, for their fruitful discussions and advices.

References

[1] T. Kojima and K. Akimoto, “Visible video data hiding techniques based on visual effects utilizing barcodes,” Proc. 2020 International Symposium on Information Theory and Its Applications (ISITA 2020), Kapolei, HI, USA (Virtual), pp.514–518, Oct. 2020.
[2] F.A.P Petitcolas, R.J. Anderson, and M.G. Kuhn, “Information hiding — A survey,” Proc. IEEE, vol.87, no.7, pp.1062–1078, July 1999.
[3] T. Kojima, T. Tachikawa, A. Oizumi, Y. Yamaguchi, and U. Parampalli, “A disaster prevention broadcasting based on data hiding scheme using complete complementary codes,” Proc. 2014 International Symposium on Information Theory and Its Applications (ISITA 2014), Melbourne, Australia, pp.45–49, Oct. 2014.
[4] T. Kojima, A. Oizumi, and U. Parampalli, “Properties of an emergency broadcasting system based on audio data hiding,” Proc. 11th International Conference on Intelligent Information Hiding and Multimedia Signal Processing (IIHMSP 2015), Adelaide, Australia, pp.142–145, Sept. 2015.
[5] Y. Matsunaga, T. Kojima, N. Aoki, Y. Dobashi, and T. Yamamoto, “A digital watermarking technique for music data using distortion effect,” Acoustic Letters, vol.39, no.1, pp.37–39, Jan. 2018.
[6] Y. Matsunaga, T. Kojima, N. Aoki, Y. Dobashi, and T. Yamamoto, “Some evaluations on a digital watermarking technique for music data using distortion effect,” IEICE Trans. Inf. & Syst., vol.E102-D, no.6, pp.1119–1125, June 2019.
[7] K. Kamada, T. Kojima, and U. Parampalli, “Tone code: A novel method for covert communications based on musical components,” Proc. 2016 International Symposium on Information Theory and Its Applications (ISITA 2016), Monterey, CA, USA, pp.340–344, Oct. 2016.
[8] The International Organization for Standardization, “ISO/IEC 15417:2007, Information Technology — Automatic Identification and Data Capture Techniques — Code 128 Bar Code Symbology Specification,” https://www.iso.org/standard/43896.html, accessed March 16, 2021.
[9] The International Organization for Standardization, “ISO/IEC 8859-1:1998, Information Technology — 8-bit Single-byte Coded Graphic Character Sets — Part 1: Latin Alphabet no.1,” https://www.iso.org/standard/28245.html, accessed March 16, 2021.
[10] 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 (in Japanese),” IEICE Trans. Inf. & Syst., vol.J88-D-II, no.6, pp.1035–1046, June 2005.
[11] T. Nakamura, S. Yamamoto, R. Kitahara, T. Miyatake, A. Katayama, T. Yasuno, and N. Sonehara, “Video watermarking scheme for real-time detection on camera phones (in Japanese),” The Journal of the Institute of Image Electronics Engineers of Japan, vol.36, no.4, pp.426–434, 2007.
[12] The Python Package Index, “pizbar 0.1.8,” https://pypi.org/project/pizbar/, accessed July 10, 2021.
[13] Japan Broadcasting Corporation (Nippon Hos¯o Ky¯okai), “NHK creative library,” https://www.nhk.or.jp/archives/creative/, accessed March 28, 2020.
[14] Signal and Image Processing Institute, “The USC-SIPI image database,” Ming Hsieh Department of Electrical Engineering, USC Viterbi School of Engineering, http://sipi.usc.edu/database/, accessed March 28, 2020.

Tetsuya Kojima received the B.E., M.E., and D.E. degrees in information engineering from Hokkaido University, Sapporo, Japan, in 1992, 1994 and 1997, respectively. From 1997 to 2001, he was with the Graduate School of Information Systems, the University of Electro-Communications, Tokyo, Japan as a research associate. In 2001, he joined the Department of Computer Science, National Institute of Technology, Tokyo College, Tokyo, Japan, as a research associate, and is currently a professor. From April, 2010 to March, 2011, he was a visiting researcher at the University of Melbourne, Australia. Dr. Kojima has served as a committee member of various academic societies as well as an organizing committee member of many international conferences including a general co-chair of the Eighth International Workshop on Signal Design and its Applications in Communications (IWSDA ’17) held in Sapporo, Japan. He has served as IEICE Tokyo Section Chair in 2021. His research interests include the sequence design and its applications to communication systems, information hiding as well as other applications of information theory. He is a member of IEEE.

Kento Akimoto graduated from the Advanced Course of Mechanical and Computer Systems Engineering, National Institute of Technology, Tokyo College, Japan, and received the B.E. degree from National Institution for Academic Degrees and University Evaluation, Japan in 2020. He has studied communication engineering and multimedia information hiding. Since 2020, he has been with NHK Technologies, Inc., Japan.