Velody – A Novel Method for Music Steganography

Camilla Vaske, Mattias Weckstén, Eric Järpe
School of Information Technology
and Electrical Engineering
Halmstad University
Box 823, Halmstad, Sweden
e-mail: camvas14@student.hh.se, mattias.wecksten@hh.se, eric.jarpe@hh.se

***Abstract*—This study describes a new method for musical steganography utilizing the MIDI format. MIDI is a standard music technology protocol that is used around the world to create music and make it available for listening. Since no publicly available method for MIDI steganography has been found (even though there are a few methods described in the literature), the study investigates how a new algorithm for MIDI steganography can be designed so that it satisfies capacity and security criteria. As part of the study, a method for using velocity values to hide information in music has been designed and evaluated, during which the capacity of the method is found to be comparable with similar methods. In an audibility test, it is observed that audible impact on the music can not be distinguished at any reasonable significance level, which means that also a security criterion is met.**

Keywords-teganography; MIDI; algorithms; velocity; capacity, audibility; music; computer security; information hiding technology; secret message embeddingteganography; MIDI; algorithms; velocity; capacity; audibility; music; computer security; information hiding technology; secret message embeddingS

I. Introduction

The best way of hiding a person is in a crowd of people. A book is best hidden in a library among many other books. So a message may best be hidden in a carrier which is a common and innocent kind of object, such as a piece of music or an image, which commonly are sent in emails without any hidden messages inside.

In communication the secrecy of the message transmitted is commonly an important aspect, e.g. in health care systems, finance reporting systems, government conduct systems etc but also for less official purposes. Steganography is the art of hiding messages. The information constituting the message is not changed, only hidden in the information constituting other objects such as images, text or sound. Steganography differs from cryptography by since the latter protects the information rather than hides it [1]. Reasons for using steganography as apposed to cryptography may be e.g. to watermark, means for transmitting whistle blower evidence leakage or use it as a tool for IT forensics as well as for illegal activities such as hiding criminal communication, industrial espionage or malware [2]. It may be useful in countries with repressive regimes where communication by means of cryptography would catch the attention of the repressing authorities, or in countries where cryptography is prohibited [3].

Steganography and cryptography complement each other and can therefore make powerful combination for solutions to communication security [4]. Due to its nature, the amount of use of steganography is harder to estimate [1] and therefore the need for steganography methods has not been documented to the same extent as the need for cryptographic methods. Nevertheless the use of steganography is well documented through the history (see [5]–[8]). Modern steganography is by many authorities considered as an alternative for citizens subject to repression in totalitarian states.

This paper is divided as follows. First, in Section II, the concepts of music steganography in general and MIDI (acronym for Musical Instrument Digital Interface) stegranography in particular are presented. Next, in Section III Velody, the method proposed is defined and evaluation criteria described. The proposed method is evaluated according to the criteria in Section IV. Finally the results are discussed and conclusions are drawn in Section V.

II. Model

In [9] different methods for sound steganography were compared. Conclusions from that study were that music files are better carriers than other sound types from a security perspective. Common formats, such as MP3, also mitigate the risk of raising suspicion [10]. Since music is a form of sound, music files may be used as carriers in the same way as other sound files. Examples of sound steganography techniques are *least significant bit coding*, *parity coding*, *echo hiding*, *phase coding*, *spread spectrum* and *tone insertion* [11], [12]. Nevertheless, the music is usually satisfying certain properties such as tempo and melody which can be further used to conceal information. Also, music streaming services provide an enormous supply of material to hide messages in [13].

In music steganography the natural properties of music can be used for steganography purposes apart from the sound steganography methods. For instance musical notes can be encoded by binary substitution where one note represents the value 0 and another note represents the value 1. Then a musical piece can be composed so that the hidden message is contained according to that rule. Alternatively, if the messages are short, a musical piece can be chosen so that it fits the message and the substitution method [14]. Another example of how the

978-1-5386-1038-1/17/$31.00 ©2017 IEEE

form of the music can be used for steganography is stated in the *Live Musical Steganography Project* where a known piece of music is modified so that certain notes deviate from the original. The message is then retrieved by calculating the difference between that musical piece and the original according to a certain algorithm [15]. The method StegIbiza [13] also falls into this category. There, indistinguishably small tempo changes are exploited to hide a message. However, it is concluded that in spite of being a lucrative area of research the focus of steganography is increasingly on image and network carriers rather than sound and music.

III. Method

The goal of this study is to establish an algorithm of a new method for music steganography. It will be evaluated according to standard quality criteria and a possibly new quality measure defined to quantify a security property. Here two MIDI steganography methods are considered. One very promising project for music steganography is StegIbiza where the idea is to convert Morse code messages into changes of bpm (beats per minute) in MIDI dance music. For modulation of the tempo, a short Morse signal is represented by a positive change of bpm and a long signal is represented by a negative change of bpm. The pieces of music used in the study were prepared both using a DAW (digital audio workstation) and a DJ controller. Evaluation of the method proved that tempo changes of 1% could not be distinguished by a group of testers [13]. Other alternatives of music steganography are [16]–[19].

A. Information Hiding in Standard MIDI Files Based on Velocity Reference Values

A recently developed MIDI steganography approach was presented in [20] in which velocity reference values were used to embed a message into the music. The method involves using the first notes of certain beat strengths in a measure as reference values for remaining notes in the same measure. The message is embedded in these notes by slightly changing their velocity according to the reference values. The study suggests that this approach prevents the velocity changes from being heard because the changes are limited.

B. Standard MIDI File Steganography

A method for steganography utilizing redundancy of MIDI file events was introduced in [21]. The method can be used to hide information in SMFs (Standard MIDI Files) without the sounds of the files being changed. An average embedding rate of 1.1% was calculated during evaluation of the method using a batch of 300 MIDI files.

C. The proposed steganography method Velody

Since there is an immense supply of MIDI songs available on the internet, and the MIDI format offers a straight forward possibility for modifying features (such as note velocity, duration etc) the proposed method Velody will be defined for MIDI files as carriers. Other methods for MIDI steganography have been suggested in the literature but no implemented and ready to use method, neither commercial, freeware nor open source, was found on the internet.

The MIDI (Musical Instrument Digital Interface) format is a protocol for describing music as a series of *note on* and *note off* events. Other features, such as note value, velocity, pitch, may also be defined in a MIDI file. Due to the simplicity and extensibility of the protocol it has become a popular choice, especially among amateur composers, professional composers and film music composers to mention a few.

The proposed steganography program Velody is a combination of different parts, some of which are well-known open source programs. The steganography program is the following bash script file:

```
@echo off
set /p midName="Name of MIDI file (without extention): "
set /p message="Message: "
call midicsv.exe %midName%.mid %midName%.txt
call steg %midName%.txt "%message%"
call csvmidi.exe steg%midName%.txt steg%midName%.mid
```

where `midicsv.exe` is the program translating music code from MIDI format to CSV (Comma Separated Value) files, `steg` is the actual steganography program given below and `csvmidi.exe` is the program translating the steganographed music code from CSV format back to MIDI. The CSV file is henceforth referred to as an *event list*.

Assuming the preceding step with `midicsv` has yielded the event list `E`, the steganography part `steg` is described by the following pseudo code for hiding the message $\mathtt{M} = \{M_1, M_2, \ldots, M_n\}$,

1) For each note event in `E` there is a velocity value. Denote by $\{V_{i,j} : i = 1, 2, \ldots, N$ and $j = 1, 2, \ldots, T\}$ the positive velocity values where i indicates its order number in the list of events (with the exception that the velocity of the first note for each instrument track j is denoted by $V_{0,j}$), and j denotes the instrument track number. According to this, checking that the size of the carrier event list is sufficiently large means checking that $n \leq N$.
2) `M` is transformed from text to binary digits (binary ASCII) `M'`.
3) The binary message `M'` is encrypted rendering $\mathtt{C} = \{C_1, C_2, \ldots, C_n\}$ using a given encryption/decryption key `K`.
4) Define velocity reference values $\{R_j : j = 1, 2, \ldots, T\}$ (one for each instrument track) as follows,
   a) If $2 \leq V_{0,j} \leq 126$, then $R_j = V_{0,j}$,
   b) If $V_{0,j} = 1$, then $R_j = 2$,
   c) If $V_{0,j} = 127$, then $R_j = 126$.
5) Construct the output event list `E'` such that it is the same as `E` but with velocities $V'_{i,j}$ defined by
   a) $V'_{i,j} = R_j - 1$ if $C_i = 0$,
   b) $V'_{i,j} = R_j + 1$ if $C_i = 1$,
   c) $V'_{i,j} = R_j$ if $i > n$.
6) Return the event list `E'`.

D. The proposed desteganography method

The desteganography program similarly combines `midicsv` translating music code from MIDI format to an event list and from that event list extracts the hidden message. The desteganography program is the bash script file with contents as follows,

```
@echo off
set /p midName="Name of MIDI file (without extention): "
call midicsv.exe \%midName\%.mid \%midName\%.txt
call desteg
return message
```

Assuming the preceding step with `midicsv` has yielded an event list `E`, the desteganography part `desteg` is described by the following pseudo code for retrieving the message `M`,

1) The steganographed file event list `E` with its positive velocity values $\{V_{i,j} : i = 1, 2, \ldots, N$ and $j = 1, 2, \ldots, T\}$ (as defined in the steganography program) is read into the desteganography program.
2) The velocity reference values $\{R_j : j = 1, 2, \ldots, T\}$ (one for each instrument track) are defined as follows,
   a) If $2 \leq V_{0,j} \leq 126$, then $R_j = V_{0,j}$,
   b) If $V_{0,j} = 1$, then $R_j = 2$,
   c) If $V_{0,j} = 127$, then $R_j = 126$.
3) The encrypted message `C` $= \{C_1, C_2, \ldots, C_n\}$ is retrieved from the velocities $V_{i,j}$ in `E` according to
   a) $C_i = 0$ if $V_{i,j} < R_j$,
   b) $C_i = 1$ if $V_{i,j} > R_j$,
   c) Nothing is done if $V_{i,j} = R_j$.
4) The encrypted message `C` is decrypted rendering the binary ASCII message `M`' using the given encryption/decryption key `K`.
5) `M`' is translated back from binary ASCII to the corresponding ASCII characters to cleartext `M`.
6) Return the decrypted message `M`.

E. Steganography example

Let us consider hiding the message "HI" in a MIDI recording of the Fugue in G minor by J.S. Bach as an illustration of how Velody works. First the cleartext HI is transformed into its binary corresponding ASCII sequence: `M` = 10010001001001. Then the encryption using XOR and, say, the key `K` = 01000001010100 is `C` = 10010001001001 + 01000001010100 (mod 2) = 11010000011101, i.e. $C_1 = 1, C_2 = 1, C_3 = 0$, $C_4 = 1$, and so on.

Now, the Fugue by Bach starts as illustrated in Figure 1. Running the corresponding MIDI file with `midicsv.exe` yields an event list as given in Figure 2. There the first note events are the three oboe notes and their velocities are 68, 66, 75. Then the oboe track has number $j = 1$ and $V_{0,1} = 68$, $V_{1,1} = 66$ and $V_{2,1} = 75$. After the oboe there is the flute with track number $j = 2$ and velocities $V_{0,2} = 59$, $V_{3,2} = 69$ and $V_{4,2} = 58$ followed by four English horn (on track $j = 3$) notes with velocities $V_{0,3} = 90$, $V_{5,3} = 82$, $V_{6,3} = 85$ and $V_{7,3} = 87$ and so on. Then the first reference values are consequently $R_1 = V_{0,1} = 68$, $R_2 = V_{0,2} = 59$ and $R_3 = V_{0,3} = 90$. Next step is defining the new velocities $V'_{1,1} = R_1 + 1 = 69$ since $C_1 = 1$, $V'_{2,1} = R_1 + 1 = 69$ since $C_2 = 1$, $V'_{3,2} = R_2 - 1 = 58$ since $C_3 = 0$, $R_{4,2} = R_2 + 1 = 60$ since $C_4 = 1$, and so on.

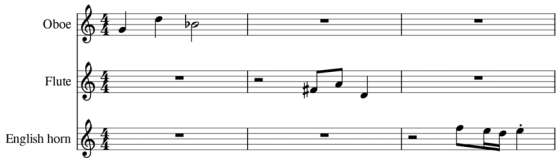

Figure 1. *The first three measures of the music score of the Fugue in G minor by J.S. Bach.*

```
0, 0, Header, 1, 4, 480
1, 0, Tempo, 500000
1, 0, Time_signature, 4, 2, 24, 8
2, 0, Start_track
2, 0, Title_t, "Oboe"
2, 0, Program_c, 0, 0
2, 0, Control_c, 0, 10, 64
Note Start  2, 0, Note_on_c, 0, 67, 68    Note Velocity
Note End    2, 480, Note_off_c, 0, 67, 64
2, 480, Note_on_c, 0, 74, 66
2, 960, Note_off_c, 0, 74, 64
2, 960, Note_on_c, 0, 70, 75
2, 1680, Note off c, 0, 70, 64
3, 0, Start_track
3, 0, Title_t, "Flute"
3, 2400, Program_c, 1, 0
3, 2400, Control c, 1, 10, 64
3, 2880, Note_on_c, 1, 66, 59
3, 3120, Note_off_c, 1, 66, 64
3, 3120, Note_on_c, 1, 69, 69
3, 3360, Note_off_c, 1, 69, 64
3, 3360, Note_on_c, 1, 62, 58
3, 3840, Note_off_c, 1, 62, 64
4, 0, Start_track
4, 0, Title_t, "English horn"
4, 4320, Program_c, 2, 0
4, 4320, Control_c, 2, 10, 64
4, 4800, Note_on_c, 2, 77, 90
4, 5040, Note_off_c, 2, 77, 64
4, 5040, Note_on_c, 2, 76, 82
4, 5160, Note_off_c, 2, 76, 64
4, 5160, Note_on_c, 2, 74, 85
4, 5280, Note_off_c, 2, 74, 64
4, 5280, Note_on_c, 2, 76, 87
4, 5520, Note off c, 2, 76, 64
```

Figure 2. *The event list that corresponds to a MIDI version of the Fugue.*

Thus, the new event list `E'` is constructed as `E` in all instances except all velocities $V_{i,j}$ are traded for their corresponding $V'_{i,j}$. Finally, by using `csvmidi.exe` the output MIDI file is constructed by transforming the event list `E'` to MIDI format.

F. Speed

To measure the speed of Velody the script method reading a message to hide and a carrier MIDI file directly into the program the average execution times of the proposed steganography as well as the desteganography procedures were calculated.

G. Capacity

To evaluate a capacity aspect of the method, i.e. the size of hidden message in proportion to size of carrier file, 100 MIDI files were randomly chosen from the site `http://www.midiworld.com`. For each of these files

the capacity for hiding a message according to the proposed method Velody was calculated and the proportion to carrier file sizes were calculated. The average of these proportion values were compared to the similar, previously mentioned but not publicly available method [20] for steganography using velocity values of MIDI files.

H. Security

One major advantage with the proposed Velody method is that it does not require a reference song to be sent together with the steganographed song. Nevertheless it would be a drawback if there were obvious audible peculiarities about the steganographed song. To clearly free Velody of such suspicions the following evaluation experiment is performed. A batch of 10 songs (see Table I) were chosen. In each of them the message

TABLE I. SONGS FOR THE SECURITY TEST

Title	*Composer*	*Duration*
1. Summertime Blues	Cochran and Capeheart	2:03
2. Pokemon bike ride theme	Masuda	1:28
3. Flashdance	Moroder	2:00
4. Blue suede shoes	Perkins	1:41
5. Pirates of the Carribean	Bruns	1:02
6. Pokemon	Siegler and Loeffler	1:03
7. Duck Tales	Mueller	0:55
8. Please please me	Lennon and McCartney	2:02
9. X files	Snow	0:46
10. Oh Susanna	Foster	1:51

THEOWLSARENOTWHATTHEYSEEM

was hidden. A security aspect is whether people are able to distinguish that a song has been tampered with. To this end 25 people were requested to participate in a test. Of these, 10 people accepted the invitation and took the test. Before the test they were instructed what differences the steganography has to a song. Then each test person was given each song as a pair: one was the original carrier version and one was the steganographed version. Then they were asked to guess which song was the steganographed version. Under the null hypothesis they guess completely at random making their probability of being right 50%. Denoting an incorrect guess by 0 and a correct one by 1, the number of correct guesses for song i is $X_i = \sum_{j=1}^{10} Y_{ij}$ where Y_{ij} is the value of the guess of person j of song i. This renders 10 binomial tests, each resulting in a p-value p_i, $i = 1, 2, \ldots, 10$.

However, in order to indicate the overall audibility security of the method these p-values need to be combined into a joint verdict for the hypothesis of whether the proposed steganography method is easily revealed or not. To this end the Fisher's method for combination of p-values the test statistic

$$U = -2\sum_{i=1}^{n} \ln p_i$$

is used for testing $H_0 : p = 0.5$ against $H_1 : p > 0.5$ at level α of significance. Under the null hypothesis $U \in \chi^2_\alpha(2n)$.

In order complicate steganalysis, velocities are encoded to two levels for each instrument track. Also the payload is encrypted using the XOR operator with a given key before it is embedded. Decryption is performed using the same key. In the audibility part of the evaluation the method tested did not include the cryptography step. Since this could not possibly favour the proposed method in respect of audibility security the results are considered to be relevant anyway also for the method which does include the encryption step.

IV. RESULTS

A. Speed

Using the whole batch of 100 randomly chosen MIDI files from the proposed steganography process executed in 0.86102 seconds per MIDI file on average. The desteganography process took 0.60947 seconds of execution time on average.

B. Capacity proportion

The embedding capacity proportion of size of hidden message relative to size of carrier file among the 100 MIDI files was 574 characters per MIDI file ($4\,595$ bits). This compares to [20] where 6 MIDI files were chosen and an average of 372 characters per MIDI file ($2\,491$ bits) could be hidden. However, the average size of the files presented in [20] were 19.3 kB while the average size of our 100 MIDI files was almost twice as much: 38.4 kB. This means that the estimated hiding capacity ratio of Velody was 1.5% while the corresponding estimate of the method in [20] was 1.9%, slightly better. Compared to [21], with an average embedding rate of 1.1%, Velody was found to have a somewhat larger capacity.

C. Security

Binomial hypothesis tests were made to establish to what extent the 10 test persons could discern audibly deviations in the 10 pairs songs (each pair constituting of one original version and one steganographed version). This resulted in 100 guesses, i.e. 10 guesses for each of the 10 songs, see Table II. Since under the null hypothesis that a person guesses completely at random as opposed to guessing with a higher probability at the steganographed version.

TABLE II. SCORE FROM THE AUDIBILITY SECURITY TEST

Song	Correct answers	p-value
1	5	0.6230
2	5	0.6230
3	5	0.6230
4	5	0.6230
5	6	0.3770
6	4	0.8281
7	3	0.9453
8	8	0.0547
9	6	0.3770
10	5	0.6230

This result was then subject to the Fisher's method for combining p-values. Here it turned out that $U = 14.9358$ which is nowhere near e.g. $\chi^2_{0.05}(20) = 31.4104$. As a matter

of fact the p-value of this overall test based on the test statistic value 14.9358 is 0.78. This p-value may clearly bust any suspicion that proposed steganography method Velody could be audible discerned.

As to steganalysis, Velody is still clearly vulnarable to detection since the two velocity level pattern for each instrument track is fairly easily recognized. Retrieval of the message is essentially impossible without the encryption key though.

V. Discussion

For watermarking and community documentation by people living in states with repressive regimes, the possibility of communicating by, not only secret but also, hidden channels are crucial. The area has received an increasing attention in the research society but is still viewed upon with scepticism among many authorities dealing with computer security by whom steganography is believed to be a technique essentially not used at all.

In this paper a novel method, Velody, for MIDI steganography by means of modifying velocity values is presented. It does not require an original file to be attached for reference to extract the hidden message as is the case with many other MIDI steganography methods.

One important aspect of quality is the capacity for hiding messages, i.e. the ratio between the size of the maximum message possible to hide according to the method and the size of the carrier file. It turns out that Velody has a slightly lower capacity for hiding messages compared to a recently developed method [20] and a slightly higher capacity as compared to the method presented in [21].

One of the most strongly desired quality properties of a steganography method is likely the security aspect. Here the risk of detecting steganographic manipulation of a MIDI song compared to the original song was tested. In a test with 10 people listening and guessing which was the manipulated version systematically for 10 MIDI tunes the p-value for rejecting that people were guessing at random was a striking 0.78. This is a radical indication that the songs treated with the proposed method Velody show no audible flaws even when compared to the original songs (which are not even present when using the method according to the intended procedure).

A possible improvement, though, could be to choose the velocity values not just constantly one unit below and one unit above the reference, but rather in a more clever way so as to imitating the values used before treatment with the proposed method. Another way could be to hide more bits per note thus raising the capacity of the method. This would be a supposedly rather straightforward improvement for future development. A valuable addition would be to evaluate this change by somehow measuring how song quality is changed from this treatment. How to measure this could be a more challenging task, which nevertheless could be of great value.

It would be of great interest to see how other solutions for music steganography rate according to the audibility security test suggested and deployed in this paper by comparing the final p-value of one method to that of Velody and others.

VI. Conclusion

A new music steganography method is developed. It shows high speed, capacity and level of secrecy according to evaluations by comparing the results to corresponding ones of contemporary methods. A possibly new way of evaluating audible effects caused by the steganography method is developed. Further development of the velocity values so that a more irregular structure is achieved would further fortify the security and steganalysis resilience of Velody.

References

[1] R. L. Biradar and A. Umashetty, "A survey paper on steganography techniques," *High Impact Factor*, vol. 9, no. 1, pp. 721–722, 2016.

[2] W. Mazurczyk, K. Szczypiorski, A. Janicki, and H. Tian, "Trends in modern information hiding: techniques, applications, and detection," *Security Comm. Networks*, vol. 9, no. 8, pp. 703–704, 2016.

[3] F. Fridrich, *Steganography in Digital Media: Principles, Algorithms, and Applications*. New York: Cambridge University Press, 2009.

[4] A. Cheddad, C. Condell, and P. Curran, C Mc Kevitt, "Digital image steganography: Survey and analysis of current methods," *Signal processing*, vol. 90, no. 3, pp. 727–752, 2010.

[5] S. Singh, *Kodboken*. Stockholm: Norstedts förlag, 1999.

[6] F. A. P. Petitcolas, R. J. Anderson, and M. G. Kuhn, "Information hiding - a survey." *Proceedings of the IEEE, special issue on protection of multimedia content*, vol. 87, no. 7, pp. 1062–1078, 1999.

[7] U. Tatar and T. Mataracolu. (2011) Analysis and implementation of distinct steganographic methods. [Online]. Available: https://arxiv.org/abs/1108.2153

[8] T. Sharp, "An implementation of key-based digital signal steganography," *I. S. Moskowitz (Ed.): IH 2001, LNCS 2137*, pp. 13–21, 2001.

[9] F. Djebbar, B. Ayad, K. A. Meraim, and H. Hamam, "Comparative study of digital audio steganography techniques," *EURASIP Journal on Audio, Speech, and Music Processing*, vol. 2012, no. 25, 2012.

[10] M. Noto. (2001) Mp3stego: Hiding text in mp3 files. [Online]. Available: https://www.sans.org/reading-room/whitepapers/stenganography/mp3stego-hiding-text-mp3-files-550

[11] W. Bender, D. Gruhl, N. Morimoto, and A. Lu, "Audio steganalysis based on negative resonance phenomenon caused by steganographic tools," *IBM Systems Journal*, vol. 35, no. 3,4, pp. 313–336, 1996.

[12] R. Tanwar and M. Bisla, "Audio steganography," in *2014 International Conference on Reliability Optimization and Information Technology (ICROIT)*, feb 2014, pp. 322–325.

[13] K. Szczypiorski. (2016) Stegibiza: New method for information hiding in club music. [Online]. Available: https://arxiv.org/abs/1608.02988

[14] J. R. Krenn. (2004) Steganography: Past, present, future. [Online]. Available: https://www.sans.org/reading-room/whitepapers/stenganography/steganography-past-present-future-552

[15] L. Hutchinson. (2014) Live musical steganography. [Online]. Available: http://scholarcommons.sc.edu/senior_theses/20

[16] J. Corinna. (2004) Steganography v – hiding messages in midi songs. [Online]. Available: https://www.codeproject.com/Articles/5390/Steganography-V-Hiding-Messages-in-MIDI-Songs

[17] K. Yamamoto and M. Iwakiri, "A standard midi file steganography based on fluctuation of duration," in *International Conference on Availability, Reliability and Security, 2009. ARES '09.*, mar 2009, pp. 774–779.

[18] N. Adli, "Three steganography algorithms for midi files," in *Proceedings of 2005 International Conference on Machine Learning and ybernetics, 2005*, nov 2005, pp. 2401–2404.

[19] P. Bao and X. Ma, "Mp3-resistant music steganography based on dynamic range transform," in *Proceedings of 2004 International Symposium on Intelligent Signal Processing and Communication Systems, 2004. ISPACS 2004*, jun 2005, pp. 266–271.

[20] D.-C. Wu and M.-Y. Chen, "Information hiding in standard midi files based on velocity reference values," *International Journal of Network Security*, vol. 18, no. 2, pp. 274–282, 2016.

[21] D. Inoue and T. Matsumoto, "Scheme of standard midi files steganography and its evaluation," in *Electronic Imaging 2002*. International Society for Optics and Photonics, 2002, pp. 194–205.