Computers, Materials & Continua
DOI:10.32604/cmc.2021.016635
Article

Tech Science Press

General Steganalysis Method of Compressed Speech Under Different Standards

Peng Liu[1], Songbin Li[1,*], Qiandong Yan[1], Jingang Wang[1] and Cheng Zhang[2]

[1]Institute of Acoustics, Chinese Academy of Sciences, Beijing, 100190, China
[2]The University of Melbourne, Melbourne, VIC3010, Australia
[*]Corresponding Author: Songbin Li. Email: lisongbin@mail.ioa.ac.cn
Received: 07 January 2021; Accepted: 17 February 2021

Abstract: Analysis-by-synthesis linear predictive coding (AbS-LPC) is widely used in a variety of low-bit-rate speech codecs. Most of the current steganalysis methods for AbS-LPC low-bit-rate compressed speech steganography are specifically designed for a specific coding standard or category of steganography methods, and thus lack generalization capability. In this paper, a general steganalysis method for detecting steganographies in low-bit-rate compressed speech under different standards is proposed. First, the code-element matrices corresponding to different coding standards are concatenated to obtain a synthetic code-element matrix, which will be mapped into an intermediate feature representation by utilizing the pre-trained dictionaries. Then, bidirectional long short-term memory is employed to capture long-term contextual correlations. Finally, a code-element affinity attention mechanism is used to capture the global inter-frame context, and a full connection structure is used to generate the prediction result. Experimental results show that the proposed method is effective and better than the comparison methods for detecting steganographies in cross-standard low-bit-rate compressed speech.

Keywords: Cross-standard; compressed speech; steganalysis; attention

1 Introduction

Data hiding is a technique of embedding secrets into digital media imperceptibly, and different types of media data are considered for steganography, including image [1,2], text [3,4], and video [5,6]. In recent years, with the continuous growth of network bandwidth and the enhancement of network convergence, network streaming media services for communication have undergone unprecedented development. Since Voice over Internet Protocol (VoIP) technology [7,8] has been widely used for real-time communication, it has become an excellent carrier for transmitting secret information over the Internet. VoIP steganography is a means of imperceptibly embedding secret information into VoIP-based cover speech. There are many VoIP speech codecs, including G.711, G.723.1, G.726, G.728, G.729, internet Low Bitrate Codec (iLBC), and the Adaptive Multi-Rate (AMR) codec. Most of them, including G.723.1, G.729, AMR, and iLBC, are low-bit-rate speech codecs that use analysis-by-synthesis linear predictive coding (AbS-LPC) [9].

At present, most methods of speech steganography utilize AbS-LPC low-bit-rate speech codecs to embed secret information for covert communication. Therefore, it is essential to develop a powerful steganalysis method to analyze low-bit-rate speech streams.

Information-hiding methods based on low-bit-rate speech streams can be divided into three categories according to the embedding position: The first category uses a pitch synthesis filter for information hiding [10–16], the second uses a LPC synthesis filter to hide information [17–22], and the third embeds information by directly modifying the value of some code elements in the compressed speech stream [23–30].

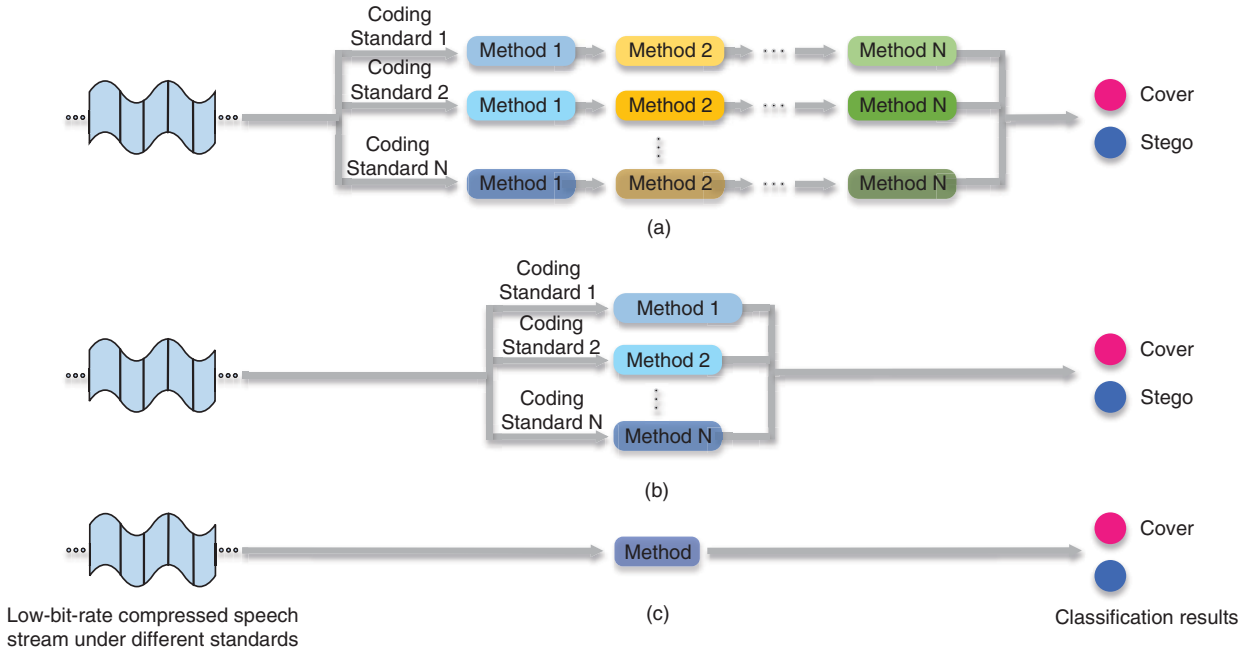

Figure 1: Difference between different levels of general steganalysis methods: (a) Non-general steganalysis method; (b) C_1- and (c) C_2-level general steganalysis methods

The existing steganalysis methods for AbS-LPC low-bit-rate compressed speech steganography are specifically designed for a specific coding standard or category of steganography methods. Thus, they lack generalization capacity. When general steganalysis is required, it is complex and time-consuming to enumerate all the steganalysis methods that correspond to the steganographic methods, which makes it difficult to meet the requirements of practical applications. In this paper, the generality of steganalysis algorithms is divided into two levels: one is general for different steganography algorithms under the same compression standard, and the other is general for steganography algorithms under different standards. For interpreting the idea of the proposed method, the generality of the first one is referred to as C_1 and that of the second one as C_2. The general steganalysis algorithm of the C_1 level can effectively detect different information-hiding algorithms (e.g., quantization index modulation [31]) under the same standard, such as G.729. The general steganalysis method of the C_2 level can detect different information-hiding algorithms under an arbitrary standard. For example, to achieve general steganalysis of different coding standards, if non-general steganography detection methods are used, it is necessary to jointly use multiple steganalysis methods for different coding standards and different steganography methods,

as shown in Fig. 1a. As demonstrated in Fig. 1b, different methods must be combined under different coding standards when using the steganalysis methods of the C_1 level. As shown in Fig. 1c, only one detection method of the C_2 level is needed. Obviously, the ideal steganalysis method is to achieve C_2-level generality, which is also the research focus of this paper.

Since speech signals are encoded by different encoding standards, the number of code elements (CEs) and their connotations are quite different. Therefore, it is unrealistic to perform C_2-level general steganalysis directly based on the original compressed speech stream. In this paper, the compressed speech stream of different coding standards is first converted into an intermediate feature representation. Then, a classification network based on a CE affinity attention mechanism is built to accomplish steganalysis.

2 Proposed Method

The architecture of the proposed steganalysis method is illustrated in Fig. 2. It can be divided into two parts: Intermediate feature representation and the steganalysis network. The intermediate feature representation is mainly used to convert compressed speech data under different coding standards into a general intermediate feature representation, and the steganalysis network performs steganalysis based on the intermediate feature. The details are described below.

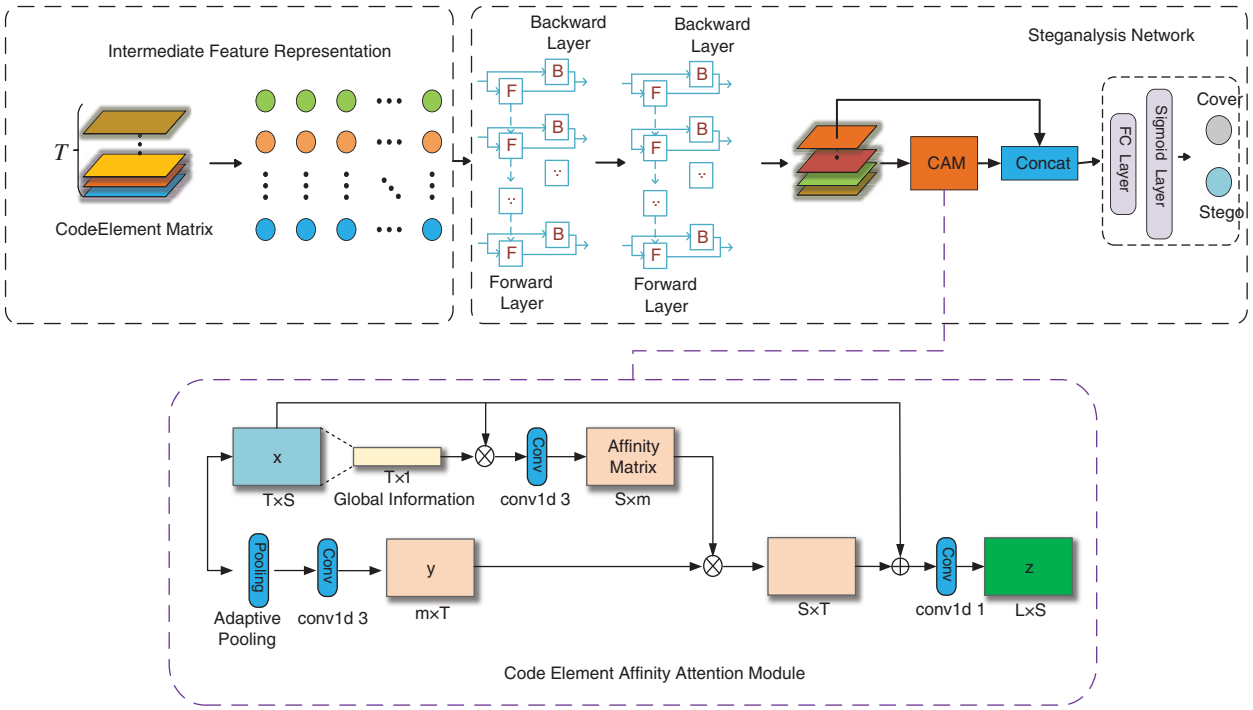

Figure 2: Architecture of the proposed method. It consists of two parts: Intermediate feature representation and steganalysis network. The code elements of a speech are first converted to an intermediate feature representation. Then, a steganalysis network based on a code-element affinity attention module is employed to detect whether the speech contains hidden information

2.1 Intermediate Feature Representation

Assuming that one must detect m types of coding standards at the same time, the CE matrix $\boldsymbol{X}^i$ corresponding to the ith coding standard can be expressed as

$$\boldsymbol{X}^i = \begin{bmatrix} x^i_{1,1} & \cdots & x^i_{1,N_i} \\ \vdots & \ddots & \vdots \\ x^i_{T,1} & \cdots & x^i_{T,N_i} \end{bmatrix} \tag{1}$$

where N_i is the number of CEs in a frame corresponding to the ith coding standard, and x^i_{T,N_i} is the value of the N_ith CE in frame T. To detect different coding standards at the same time, the CE matrices corresponding to m coding standards are concatenated to obtain a synthetic CE matrix $\boldsymbol{X}$:

$$\boldsymbol{X} = \begin{bmatrix} x^1_{1,1} & \cdots & x^1_{1,N_1} & & x^m_{1,1} & \cdots & x^m_{1,N_m} \\ \vdots & \ddots & \vdots & \cdots & \vdots & \ddots & \vdots \\ x^1_{T,1} & \cdots & x^1_{T,N_1} & & x^m_{T,1} & \cdots & x^m_{T,N_m} \end{bmatrix} \tag{2}$$

where x^m_{T,N_m} is the value of the N_mth CE in frame T corresponding to the mth coding standard.

To convert the values of CEs into a form that is easy to use by the neural network, one-hot coding is utilized to map each CE into a feature vector. For a CE that occupies n bits, its coded value range is $0 - 2^{n-1}$. In one-hot encoding, a vector with a length of 2^n is used to represent this CE. If the coded value of this CE is u, the one-hot representation can be denoted as

$$\mathcal{H}_{\text{one-hot}}(x) = \left\{c_0, \quad c_1, \quad \cdots, \quad c_{2^{n-1}}\right\} \tag{3}$$

where

$$c_i = \begin{cases} 1, & if \ \ i = u \\ 0, & else \end{cases} \tag{4}$$

After one-hot coding, a group of independent CE one-hot representations are obtained, which are then aggregated according to the order of the original CEs to form a long feature vector, called a multi-hot vector. This process is called multi-hot coding. Taking the example in which there are M code elements in a frame, the length of the corresponding multi-hot vector can be calculated as

$$\ell = \int_{i=1}^{M} 2^{d_i} \tag{5}$$

where d_i denotes the number of bits that the ith CE occupies.

However, some CEs may occupy too many bits, which will greatly increase the amount of calculation of the model. When the one-hot coding operation is conducted on these code elements, the length of the one-hot vector will become very large. This explosive dimension increase is more than can be afforded, so dimensionality reduction is needed. Therefore, a frequency count method is employed on these CEs. Experiments prove that this is a simple but very effective coding method. Specifically, for each CE that occupies more than 8 bits, the occurrence frequency of its

every coded value is counted. Then, the coded values are arranged in order of frequency, and the first 255 values selected. These values are encoded as 0–254. The other coded values are encoded as 255. In this way, the coded value of all CEs can be mapped into 0–255.

A sparse representation $\boldsymbol{R}$ can be obtained by applying multi-hot encoding. However, the sparse representation will bring an additional computational cost to the model, which is unfavorable for the real-time requirements of steganalysis. Inspired by natural language processing tasks, an embedding method for each CE is introduced. First, dictionaries are built for each CE to convert the multi-hot vectors to more compact representations into the intermediate feature. The parameters are randomly initialized in the dictionaries with the normal distribution. It is hoped that such an embedding representation can be obtained that has a strong robustness to the different embedding rates. Then, a large dataset consisting of different stego data and cover data is built to pre-train the dictionaries. At the pre-training stage, two-layer bidirectional long short-term memory (Bi-LSTM) [32] and a full connection layer are used, followed by a sigmoid activation function. Dictionaries and the training network are trained together. In addition, the dictionaries are fixed once the training is done. Before utilizing the steganalysis network to classify the input sample, the matrix $\boldsymbol{R}$ will be converted into the embedding matrix $\boldsymbol{E}$ based on the trained dictionaries.

2.2 Steganalysis Network

Since the front and back frames of a speech sample can influence each other, a two-layer Bi-LSTM is first employed to capture long-term contextual correlations of $\boldsymbol{E}$, and a better representation of the frame vector is generated. However, Bi-LSTM can only capture long-range dependencies, which lack local CE information. Inferring inter-frame context information from local CE information can simultaneously capture both intra- and inter-frame relationships, which is very important for low-bit-rate compressed speech steganalysis tasks. Global context information is useful for extracting a wide range of inter-frame dependencies and providing a comprehensive understanding of the entire input speech sequence, while local CE information plays a key role in understanding the secret information embedded in different CE positions. Based on this theory, the CE affinity attention module is proposed, which adaptively infers the global context information between frames under the guidance of the codeword affinity representation.

The architecture of the CE affinity attention module is illustrated in Fig. 2. It consists of two branches: the first branch is used to calculate the local affinity attention vector, and the second deals with the feature representation $\boldsymbol{y}$ at a single scale. Moreover, the second branch determines the amount of information contained in the local affinity vectors. Both branches will be described in detail below.

In this paper, the output features calculated by Bi-LSTM are defined as $\boldsymbol{O}$, where T indicates the number of frames of the input data, and S the feature dimension. In the first branch, the features $\boldsymbol{O}$ are first calculated by a global average pooling operation to obtain the global information representation $\boldsymbol{g}(\boldsymbol{O})$, which can express the global inter-frame information. The process can be defined as

$$\boldsymbol{g}(\boldsymbol{O_i}) = \frac{1}{S}\sum_{j=1}^{S} o_{i,j}, \quad i \in (1, 2, \ldots, S) \tag{6}$$

where $o_{i,j}$ denotes the feature value at the jth position of the ith frame. Then, a frame-wise multiplication between global information $\boldsymbol{g}(\boldsymbol{O_i})$ and input features $\boldsymbol{O}$ is employed to obtain a new global-guided feature representation $\widetilde{\boldsymbol{O}}$, which can be calculated by

$$\widetilde{\boldsymbol{O}}_j = \boldsymbol{g}(\boldsymbol{O})_j * \boldsymbol{O}_j, \quad j \in (1, 2, \ldots, T) \tag{7}$$

After $\widetilde{\boldsymbol{O}}$ is obtained, it is reshaped into $\overline{\boldsymbol{O}}$. Then, a one-dimensional (1D) convolution (kernel size 3 and stride 1) followed by a ReLU activation function is used to convert the global-guided feature representation into an affinity vector $\boldsymbol{A}$, where M denotes the area size of codeword affinity. In the second branch, adaptive average pooling and a 1D convolution are first applied on input feature $\boldsymbol{O}$ to obtain $\boldsymbol{y}$. Then, $\boldsymbol{y}$ is reshaped into the size $M \times T$ to match that of the affinity vector. $\boldsymbol{A}$ and $\boldsymbol{y}$ are then multiplied together and the results reshaped to obtain an adaptive context matrix $\boldsymbol{z}^M$, which includes local codeword information and global inter-frame correlations. The mathematical description of this process can be defined as

$$\boldsymbol{z}^M = \sum_{j=1}^{M} a_j y_j \tag{8}$$

where $a_j \in \boldsymbol{A}$ indicates the affinity factor.

To endow the features used for classification with both long-range dependencies and global inter-frame context, the features output from Bi-LSTM and the codeword affinity module are integrated to form a more powerful feature representation. Then, the features are inputted into a classification that consists of two layers of full connection and a sigmoid activation function. A prediction probability value p, which determines whether the hidden message exists in an input speech sequence, is then obtained:

$$Output\ \ Result = \begin{cases} cover, & p < 0.5 \\ stego, & p \geq 0.5 \end{cases} \tag{9}$$

3 Equations and Mathematical Expressions

Seven thousand speech segments were collected from the Internet, including samples from seven human voice categories, to form the speech database. Each category contains 1,000 speech segments. The seven categories are Chinese man, Chinese woman, English man, English woman, French, German, and Japanese. Each human voice category contains samples from more than five individuals. The duration of each speech segment is 10 s, and each segment is formatted as a mono PCM file with an 8,000-Hz sampling rate and 16-bit quantization. The speech segments in each category are divided into a training dataset and a testing dataset at a 4:1 ratio. The training dataset is used to conduct parameters adjustment of the model, and the test dataset is used to evaluate the model performance. The G.723.1 (6.3 kbit/s) and G.729 codecs are used to evaluate the performance of the proposed method.

Both the training and testing stages were executed on a GeForce GTX 2080 graphical processing unit with 11 Gb of graphics memory. PyTorch was used to help implement the model and algorithm. In addition, in the process of training the neural network, Adam was used as the optimizer with a learning rate of 1×10^{-4}, and the cross-entropy chosen as the loss function. The maximal training epoch was 200, and the batch size in the training process was 16.

As mentioned above, three main categories of steganography methods exist for AbS-LPC low-bit-rate compressed speech. To comprehensively test the performance of the proposed model, a representative method [15,17,24] was chosen for each steganography category. For simplicity, the chosen methods are denoted "ACL" [15], "CNV" [17], and "HYF" [24]. It should be noted that the ACL and HYF methods are designed for the G.723.1 standard, and the CNV method was used for steganography under the G.729 standard; All three methods were used for steganography under the G.723.1 standard.

To the best of our knowledge, no general method has been designed for the detection of steganographies in cross-standard AbS-LPC low-bit-rate compressed speech. The MFCC-based steganalysis method [33] can, in theory, detect any type of steganography based on the decoded audio/speech data. In this sense, this method is believed to be general as well. Besides, Hu et al. [34] proposed a SFFN-based general steganalysis method for specialized coding standards. In the present paper, these methods are used as comparison algorithms with which to evaluate the proposed method.

The embedding rate is defined as the ratio of the number of embedded bits to the total embedding capacity. Experiments on the three steganography methods for the G.723.1 standard were conducted under five different embedding rates (20%–100%). The experimental results are shown in Tab. 1. For ACL, the detection accuracy of the MFCC method is only 51.58% when the embedding rate is 20%, slightly better than a random guess. As a comparison, the detection accuracy of the proposed method is 98.96%, far exceeding that of the MFCC method. However, the detection accuracy of SFFN achieves 99.54%, 0.58% higher than the proposed method. When the embedding rate is 40% or above, both SFFN and the proposed method have a detection accuracy of 100%. For HYF and CNV, when the embedding rate is 20%, the detection accuracies of the proposed method are 35.73% and 37.26% higher, respectively, than that of MFCC. By contrast, the detection accuracies of SFFN are 8.48% and 12% higher than that of MFCC, respectively. When the embedding rate is 80% or above, SFFN can achieve detection accuracies greater than 95%, while the proposed method can achieve the same accuracy when the embedding rate is only 20%.

Table 1: Detection accuracies of 10 s of speech with different embedding rates for G.723.1 standard. Results in bold are for the proposed method

Steganography method	Steganalysis method	Embedding rate (%)				
		20	40	60	80	100
HYF	MFCC	61.66	63.96	66.84	68.91	71.38
	SFFN	70.14	82.79	92.04	96.43	99.17
	Proposed	**97.39**	**99.75**	**99.96**	**100**	**100**
CNV	MFCC	59.17	60.16	62.89	64.25	68.76
	SFFN	71.17	83.71	92.28	95.82	99.46
	Proposed	**96.43**	**99.96**	**99.96**	**100**	**100**
ACL	MFCC	51.58	52.33	52.4	56.34	61.47
	SFFN	**99.54**	**100**	**100**	**100**	**100**
	Proposed	98.96	**100**	**100**	**100**	**100**

Since the ACL and HYF methods are designed for the G.723.1 standard, the CNV method is used for steganography under the G.729 standard. Experiments on the CNV method were conducted under five different embedding rates (20%–100%). The experimental results are shown in Tab. 2, from which it can be seen that the proposed method performs better than MFCC and SFFN at all embedding rates. When the embedding rate is 20%, the detection accuracies of the proposed method are 32.73% higher than that of MFCC and 6.74% higher than that of SFFN. When the embedding rate is 80% or above, SFFN can achieve detection accuracies greater than 99%, while the proposed method can achieve the same accuracy when the embedding rate is only 40%.

Table 2: Detection accuracies of 10 s of speech with different embedding rates for G.729 standard. Results in bold are for the proposed method

Steganography method	Steganalysis method	Embedding rate (%)				
		20	40	60	80	100
CNV	MFCC	62.37	64.55	65.95	67.25	69.03
	SFFN	88.36	96.85	98.53	99.51	99.97
	Proposed	**95.1**	**99.64**	**99.89**	**100**	**100**

In summary, the proposed method achieves the best results at all embedding rates under the G.723.1 and G.729 standards, except for a 20% embedding rate and ACL steganography under the G.723.1 standard, which is 0.58% lower than that of SFFN. The experimental results indicate that the proposed steganalysis method can be effective for detecting steganographies in cross-standard low-bit-rate compressed speech.

4 Conclusions

In this paper, a common method for detecting steganographies in cross-standard low-bit-rate compressed speech based on intermediate feature representation is proposed. To detect multiple coding standards at the same time, the code element (CE) matrices corresponding to m coding standards are first concatenated to obtain a synthetic CE matrix. Then, one-hot coding is utilized to convert this matrix into a form that is easy to use by a neural network. Inspired by the ideas in natural language processing, dictionaries are built for each CE by transforming them into intermediate features to achieve more compact representations. These features are inputted into the resulting steganalysis network to obtain the final classification result. Experimental results indicate the superiority in accuracy and performance of the proposed method.

Funding Statement: This work is supported partly by Hainan Provincial Natural Science Foundation of China under Grant No. 618QN309, partly by the Important Science & Technology Project of Hainan Province under Grant Nos. ZDKJ201807 and ZDKJ2020010, partly by the Scientific Research Foundation Project of Haikou Laboratory, Institute of Acoustics, Chinese

Academy of Sciences, and partly by the IACAS Young Elite Researcher Project (QNYC201829 and QNYC201747).

Conflicts of Interest: The authors declare that they have no conflicts of interest to report regarding the present study.

References

[1] R. Das, M. Baykara and G. Tuna, "A novel approach to steganography: Enhanced least significant bit substitution algorithm integrated with self-determining encryption feature," *Computer Systems Science and Engineering*, vol. 34, no. 1, pp. 23–32, 2019.

[2] Y. Wang, Z. Fu and X. Sun, "High visual quality image steganography based on encoder-decoder model," *Journal of Cyber Security*, vol. 2, no. 3, pp. 115–121, 2020.

[3] L. Y. Xiang, S. H. Yang, Y. H. Liu, Q. Li and C. Z. Zhu, "Novel linguistic steganography based on character-level text generation," *Mathematics*, vol. 8, no. 9, pp. 1558, 2020.

[4] Y. J. Tong, Y. L. Liu, J. Wang and G. J. Xin, "Text steganography on RNN-generated lyrics," *Mathematical Biosciences and Engineering*, vol. 16, no. 5, pp. 5451–5463, 2019.

[5] Z. Li, L. Meng, S. Xu, Z. Li, Y. Shi *et al.,* "A hevc video steganalysis algorithm based on pu partition modes," *Computers, Materials & Continua*, vol. 59, no. 2, pp. 563–574, 2019.

[6] H. Tang, X. Yang, Y. Zhang and K. Niu, "A mv-based steganographic algorithm for H.264/avc without distortion," *Computers, Materials & Continua*, vol. 63, no. 3, pp. 1205–1219, 2020.

[7] B. Goode, "Voice over internet protocol (VoIP)," *Proceedings of the IEEE*, vol. 90, no. 9, pp. 1495–1517, 2002.

[8] R. J. B. Roslin, O. O. Khalifa and S. S. N. Bhuiyan, "Improved voice over internet protocol for wireless devices," in *7th Int. Conf. on Computer and Communication Engineering*, Kuala Lumpur, Malaysia, pp. 498–503, 2018.

[9] D. O. Shaughnessy, "Linear predictive coding," *IEEE Potentials*, vol. 7, no. 1, pp. 29–32, 1988.

[10] B. Geiser and P. Vary, "High rate data hiding in ACELP speech codecs," in *IEEE Int. Conf. on Acoustics, Speech and Signal Processing*, Las Vegas, NV, USA, pp. 4005–4008, 2008.

[11] A. Nishimura, "Data hiding in pitch delay data of the adaptive multi-rate narrow-band speech codec," in *Fifth Int. Conf. on Intelligent Information Hiding and Multimedia Signal Processing*, Kyoto, Japan, pp. 483–486, 2009.

[12] H. Miao, L. Huang, Z. Chen, W. Yang, AI-hawbani *et al.,* "A new scheme for covert communication via 3G encoded speech," *Computers & Electrical Engineering*, vol. 38, no. 6, pp. 1490–1501, 2012.

[13] S. Yan, G. Tang and Y. Chen, "Incorporating data hiding into G.729 speech codec," *Multimedia Tools and Applications*, vol. 75, no. 18, pp. 11493–11512, 2016.

[14] Y. Ren, H. Wu and L. Wang, "An AMR adaptive steganography algorithm based on minimizing distortion," *Multimedia Tools and Applications*, vol. 77, no. 10, pp. 12095–12110, 2018.

[15] Y. F. Huang, C. Liu, S. Y. Tang and S. Bai, "Steganography integration into a low-bit rate speech codec," *IEEE Transactions on Information Forensics and Security*, vol. 7, no. 6, pp. 1865–1875, 2012.

[16] Y. Z. Ren, H. Y. Yang, H. X. Wu, W. P. Tu and L. N. Wang, "A secure AMR fixed codebook steganographic scheme based on pulse distribution model," *IEEE Transactions on Information Forensics and Security*, vol. 14, no. 10, pp. 2649–2661, 2019.

[17] B. Xiao, Y. F. Huang and S. Y. Tang, "An approach to information hiding in low bit-rate speech stream," in *IEEE Global Telecommunications Conf.*, New Orleans, LO, USA, pp. 1–5, 2008.

[18] Y. F. Huang, H. Z. Tao, B. Xiao and C. C. Chang, "Steganography in low bit-rate speech streams based on quantization index modulation controlled by keys," *Science China Technological Sciences*, vol. 60, no. 10, pp. 1585–1596, 2017.

[19] H. Tian, J. Liu and S. B. Li, "Improving security of quantization-index-modulation steganography in low bit-rate speech streams," *Multimedia systems*, vol. 20, no. 2, pp. 143–154, 2014.

[20] P. Liu, S. B. Li and H. Q. Wang, "Steganography in vector quantization process of linear predictive coding for low-bit-rate speech codec," *Multimedia Systems*, vol. 23, no. 4, pp. 485–497, 2017.

[21] P. Liu, S. B. Li and H. Q. Wang, "Steganography integrated into linear predictive coding for low bit-rate speech codec," *Multimedia Tools and Applications*, vol. 76, no. 2, pp. 2837–2859, 2017.

[22] Y. Z. Ren, W. M. Zheng and L. N. Wang, "SILK steganography scheme based on the distribution of LSF parameter," in *Asia-Pacific Signal and Information Processing Association Annual Summit and Conf.*, Honolulu, HI, USA, pp. 539–548, 2018.

[23] T. Z. Xu and Z. Yang, "Simple and effective speech steganography in G. 723.1 low-rate codes," in *Int. Conf. on Wireless Communications & Signal Processing*, Nanjing, China, pp. 1–4, 2009.

[24] Y. F. Huang, S. Y. Tang and J. Yuan, "Steganography in inactive frames of VoIP streams encoded by source codec," *IEEE Transactions on Information Forensics and Security*, vol. 6, no. 2, pp. 296–306, 2011.

[25] J. Liu, K. Zhou and H. Tian, "Least-significant-digit steganography in low bitrate speech," in *IEEE Int. Conf. on Communications*, Ottawa, Canada, pp. 1133–1137, 2012.

[26] R. S. Lin, "An imperceptible information hiding in encoded bits of speech signal," in *Int. Conf. on Intelligent Information Hiding and Multimedia Signal Processing*, Adelaide, SA, Australia, pp. 37–40, 2015.

[27] Z. J. Wu, H. J. Cao and D. Z. Li, "An approach of steganography in G.729 bitstream based on matrix coding and interleaving," *Chinese Journal of Electronics*, vol. 24, no. 1, pp. 157–165, 2015.

[28] S. F. Yan, G. M. Tang, Y. F. Sun, Z. Z. Gao and L. Q. Shen, "A triple-layer steganography scheme for low bit-rate speech streams," *Multimedia Tools and Applications*, vol. 74, no. 24, pp. 11763–11782, 2015.

[29] X. S. Peng, Y. F. Huang and F. F. Li, "A steganography scheme in a low-bit rate speech codec based on 3D-sudoku matrix," in *8th IEEE Int. Conf. on Communication Software and Networks*, Beijing, China, pp. 13–18, 2016.

[30] Y. J. Jiang and S. Y. Tang, "An efficient and secure VoIP communication system with chaotic mapping and message digest," *Multimedia Systems*, vol. 24, no. 3, pp. 355–363, 2018.

[31] S. B. Li, Y. Z. Jia and C. C. J. Kuo, "Steganalysis of qim steganography in low-bit-rate speech signals," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 25, no. 5, pp. 1011–1022, 2017.

[32] G. Liu and J. B. Guo, "Bidirectional LSTM with attention mechanism and convolutional layer for text classification," *Neurocomputing*, vol. 337, no. 14, pp. 325–338, 2019.

[33] Q. Z. Liu, A. H. Sung and M. Y. Qiao, "Temporal derivative-based spectrum and mel-cepstrum audio steganalysis," *IEEE Transactions on Information Forensics and Security*, vol. 4, no. 3, pp. 359–368, 2009.

[34] Y. T. Hu, Y. H. Huang, Z. L. Yang and Y. F. Huang, "Detection of heterogeneous parallel steganography for low bit-rate voip speech streams," *Neurocomputing*, vol. 419, no. 1, pp. 70–79, 2020.