International Journal of Research in Engineering and Science (IJRES)
ISSN (Online): 2320-9364, ISSN (Print): 2320-9356
www.ijres.org Volume 10 Issue 4 ǁ 2022 ǁ PP. 39-48

Multi-Stage Residual Hiding for Image-Into-Audio Steganography

Gali Dileep kumar, *Department of Electrical and Electronics and Communication Engineering, GITAM, University, India.*

Tadi Rajesh Kumar, *Department of Electrical and Electronics and Communication Engineering, GITAM University, India.*

N.K.L. Prasanna, *Department of Electrical and Electronics and Communication Engineering, GITAM University, India.*

G. Sai Teja, *Department of Electrical and Electronics and Communication Engineering, GITAM University, India.*

Dr. D. Madhavi, *Department of Electrical and Electronics and Communication Engineering, GITAM University, India.*

Abstract
The extensive use of audio communication technology has sped up the transmission of audio data over the Internet, making it a popular carrier for covert communication. We describe a cross-modal steganography method for hiding image material in audio carriers while maintaining the cover audio's perceptual integrity in this study. The first network encodes the decreasing multilevel residual errors inside different audio subsequence with the corresponding stage sub-networks in our framework, while the second network decodes the residual errors to produce the final revealed results from the modified carrier with the relevant stage sub-networks. Because of the gradual sparse feature of residual faults, the suggested framework's multi-stage architecture not only makes payload capacity control more flexible, but it also makes concealment easier. Qualitative research suggests that human listeners are unaffected by carrier alterations, and that the decoded visuals are extremely understandable.

Keywords: *Audio steganography, residual hiding, multi-stage network, convolutional neural networks*

Date of Submission: 01-04-2022 Date of acceptance: 14-04-2022

I. INTRODUCTION

A file may contain more information than it appears to have. A file may appear normal to the untrained eye, but expert recipients can extract additional information from it. Steganography, a technique for concealing hidden messages in digital carriers to promote covert communication by utilizing the redundancy of human perceptions, has recently been widely investigated to protect personal data. Steganography's secrecy appeals to a wide range of applications, including copyright certification [1] and covert communication [2].

The embedding approach has a big impact on faithful concealing. A wide range of steganography settings and methods have been proposed to achieve perfect hiding performance. For example, due to their simplicity, Least Significant Bit (LSB) replacement algorithms [3, 4, 5] have become particularly popular for steganography. Then sophisticated approaches like HUGO [6, WOW [7], and S-UNIWARD [8] appeared, which encoded messages in complicated textures by minimizing a well-crafted distortion function and obtained superior performance. The aforementioned methods, on the other hand, typically rely on domain knowledge to detect features for hiding secret messages in cover carriers, resulting in low payload capacities and apparent distortion changing.

Deep neural networks (DNN) have been applied to steganography in recent years, with a heavy concentration on images. DNN-based approaches have been investigated to learn the signal properties implicitly rather than specifying domain knowledge explicitly. Generative Adversarial Networks (GAN) were used to apply deep learning to steganography for the first time [9, 10, 11, 12, and 13]. Adversarial training not only increases resistance to additional types of attacks, but it also improves visual performance. Several steganography techniques [1, 14, and 15] that embed a picture inside another image have been suggested to improve the payload capacities of steganography. Furthermore, as various audio applications become more prominent, researchers are paying increasing attention to speech steganography. In [16] optimises two neural networks to implant the message in the cover audio and retrieve the message from the changed carrier. [17]

proposes a GAN-based audio-to-audio architecture in which an encoder network and a decoder network are developed for frequency domain information embedding and extraction utilizing the short-time Fourier transform.

In comparison to hand-crafted embedding approaches, the aforementioned deep network-based steganography algorithms have outperformed them. These frameworks, however, nevertheless have the following flaws: (1) The majority of these methods appear to be ineffective when it comes to steganography between different data modalities. (2) Because of the diversity of information, hiding the secret message directly is challenging and generally results in visible artefacts.

To address the drawbacks of the aforementioned approaches, we offer a Deep neural network-based Image-To-Audio Steganography (DITAS) framework, as illustrated in Fig. 1, that consists of two multi-stage networks: the hidden network and the revealing network. The revealing network decodes the residual errors from the modified carrier with the appropriate stage sub-networks to produce the final revealed findings, while the hiding network encodes the decreasing residual errors inside distinct audio sub sequences with the corresponding stage sub-networks. is the suggested framework.

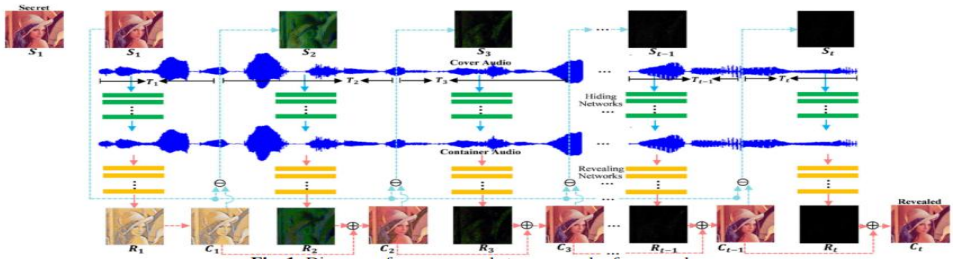

Figure.1.1 shows a diagram of the steganography framework.

Hides the secret message in a step-by-step method, making the process easier. The suggested technique outperforms competing algorithms on benchmark datasets, according to experimental results.

The following list consists of the most important contributions:

1) We offer a revolutionary image-to-audio steganography framework based on deep learning that out performs current approaches in terms of hiding capabilities.

2) The proposed technique can regulate the payload capacity more flexibly while also making the hiding process easier by hiding the residual faults of several tiers.

3) Because our architecture embeds residual faults in separate audio sub sequences, the secret image can be restored to some extent even if part of the carrier is destroyed.

II. PROPOSED METHOD

2.1 Hiding Network

Given a secret image S0 of size w×h and a cover audio A l, where w, h are the secret image's width and height, and l is the cover audio's dimension. Using a multi-stage network, we gradually integrate multilevel residual defects of the secret image into the cover audio. Specifically, from the cover audio sequence, t non-overlapping audio sub sequences are initially selected, which are expressed mathematically as T1, T2,..., Tt, and the dimension of each subsequence is wh. The suggested framework is made up of t stages that are used to embed residual mistakes from the hidden image into these t sub sequences. More specifically, we hide the residual error between the original secret image and the revealed results from the previous i-1 stages disclosing sub networks for the i-th stage. The process of concealment can be described as follows:

$$\tilde{T}_i = H(Si, Ti; \theta Hi)$$

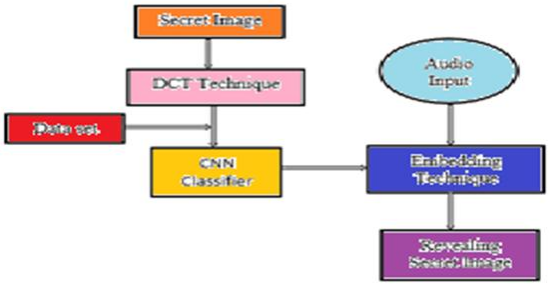

Figure 2.1.1 Block diagram of Proposed Method

where H denotes the hiding network's operation and Hi denotes the parameter of the i-th stage hiding sub-network. T˜ i is the concealed result (Container) that preserves the perceptual fidelity of the cover audio T˜ i, and Si is the i-th level residual error of the secret picture to be hidden. The residual error Si is calculated as follows:

$$S_i = S_0 - C_{i-1}, \qquad C_{i-1} = \sum_{j=0} R_j$$

where S0 is the initial secret picture, Ri I > 0) is the disclosed result of the i-th stage revealing sub-network, as shown in the next chapter, and R0 is a zero tensor of the same size as S0. The total sum of revealed results from previous I steps disclosing sub-networks is denoted by Ci.

2.2 Revealing Network

The concealed audio subsequence T˜ I is produced after the hiding stage. In the revealing stage, the purpose of the revealing network is to extract the secret image from the hidden audio sequence. The revealing procedure can be described as follows, given the secret audio subsequence T˜ i.

Ri = R (T˜ i; θ Ri)

Where R denotes the revealing network's activity and Ri denotes the parameter of the i-th stage revealing sub-network. T I yields Ri, which is the revealed residual result. We combine the residuals Ri together to produce the final revealed once we acquire the residuals Ri.

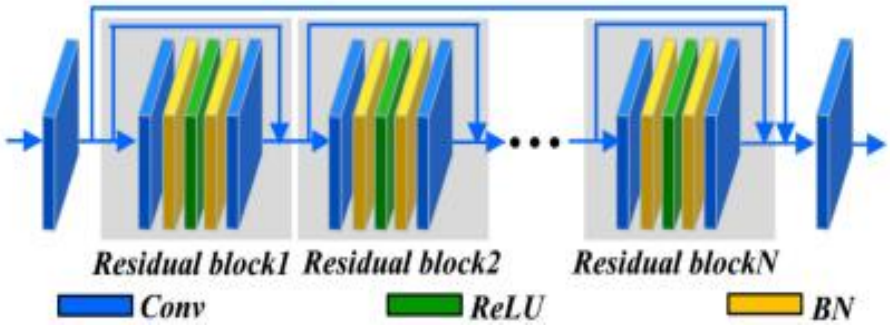

Figure.2.2.1 shows the suggested residual block's architecture.

Ci = Pi j=0 Rj is the end outcome. It's worth noting that the suggested system embeds multi-level residuals in separate audio sub sequences, allowing the revealed result to be recovered from the container sequence in stages. In other words, the sub sequences are independent between each other and even if some sub sequences are lost, the secret image can also be revealed to some extent, which improves the robust ness of the proposed method.

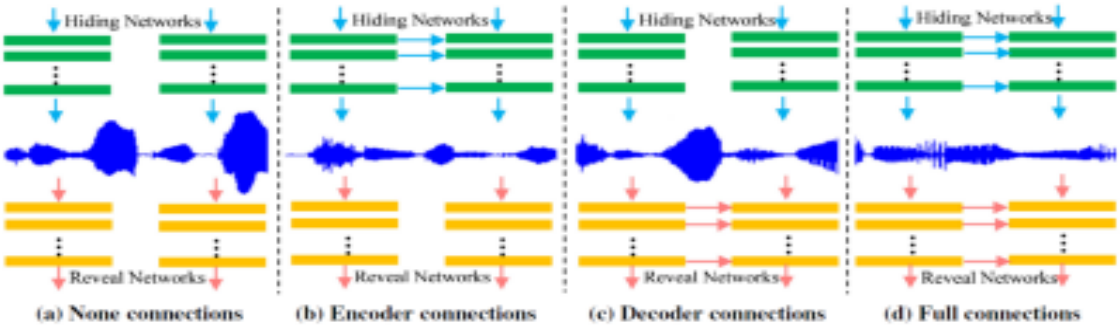

Fig. 2.2.2 Four experimental versions' structural details.

2.3 Loss Function

We have two main tasks in the proposed framework: one is secret picture concealing, and the other is secret image extraction.

As a result, the container and the extracted entities are both constrained by two loss items. Furthermore, because the proposed framework has t phases, the whole loss function can be represented as

L (θ Hi , θ Ri) = Xt i=1 L Hi (θ Hi) + λi L Ri (θ Ri)

LHi conceals the loss of the i-th stage concealing sub-network for information concealment, while LRi reveals the loss of the i-th stage revealing sub-network for information extraction. The parameters of them are Hi and Ri. The regularisation parameter I is used to manage the tradeoff between them. The hiding loss is defined as follows:

$$L\ Hi\ (\theta\ Hi) = 1\ N\ X\ N\ i=1\ k\ H\ (Si, Ti; \theta\ Hi) - Ti\ k\ \|\ ^{2}_{2}$$

$$L\ Ri\ (\theta\ Ri) = 1\ N\ X\ N\ i=1\ k\ R\ (T\tilde{}\ i; \theta\ Ri) - Si\ k\ \|\ ^{2}_{2}$$

Where $T\tilde{}\ i = H\ (Si, Ti; \theta\ Hi)$ indicates the container audio sequences.

III. EXPERIMENTAL RESULTS AND ANALYSIS

3.1 Implementation and training details

Pre-processing is required for the one dimensional audio subsequence Ti in order to insert the secret image into the cover audio using convolutional methods. Two methods of audio data pre-processing are used specifically: 1) Audio data is directly transformed into a new tensor. 2) To convert audio from the audio domain to the frequency domain, the Short-Time Fourier Transform (STFT) is used.

Following the pre-processing, a tensor with the size of w × h is generated for Ti, which may be simply concatenated with the secret entity Si as the input of the i-th stage hidden sub network. The proposed residual block is used in the proposed hiding and revealing networks, as shown in Fig.2. Each convolutional layer for each stage hidden sub-network consists of 64 kernels of size 33, except for the last layer, which contains a single kernel to ensure dimensional consistency between the output and the cover audio tensor. Similarly, the revealing network's final convolutional layer includes 3 kernels to extract the final revealed residuals, while the remainder layers have 64 kernels.

On a Titan X GPU, we train our model with Pytorch [18], a Python toolkit. All network parameters are optimized via adaptive moment estimation (Adam) [19]. We also set λi=0.8

Fig. 3.1.2 Our framework's intermediate visual outputs.

0.8. To determine the best number of stages (t), we train a model with 10 stages, and Fig.4 depicts the relationship between performance and stage numbers, demonstrating that as the number of stages increases, performance development slows down, and we choose t = 5 in our model. It's worth noting that the proposed architecture can be trained from beginning to end.

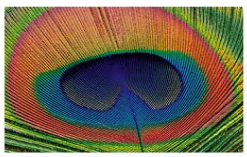

Figure.3.1.1 First colour image as one input image in one case

Figure 3.1.2 First Black and white image as input image in one case

Figure.3.1.3 Second colour image as one input image in one case

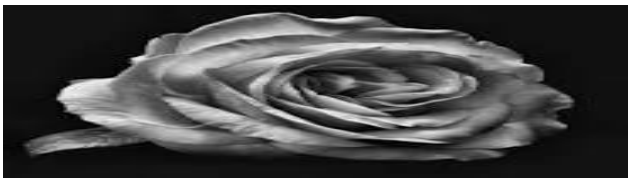

Figure 3.1.4 Second Black and white image as input image in one case

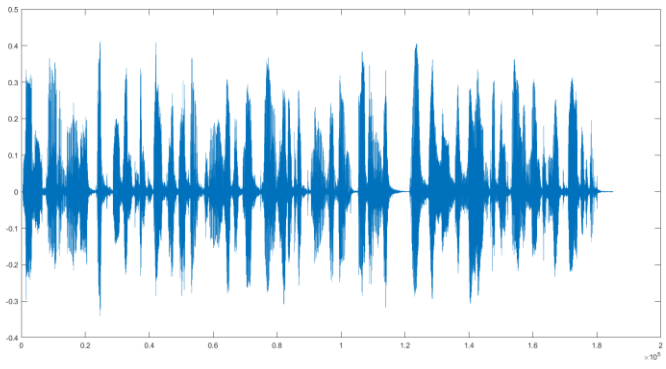

Figure 3.1.5 input audio file

We employ the VOC2012 [20] training set as the hidden picture and the LJ Speech dataset [21] as the cover audio for training. We chose two types of images as the secret image for testing: natural images (containing Set14 [22], LIVE1 [23], Val of VOC2012 [20], and Val of Image Net [14]) and facial photos (using CelebA dataset [12]). Furthermore, the cover audio is taken from the TIMIT audio dataset [24]. The patch size is set to 6464, which is randomly cut from the training dataset, and the batch size is set to 16. For all layers, the learning

rate is set at 1e-4 and then reduced by a factor of three every 20 epochs. For 200 epochs, we train the entire network.

Figure 3.1.6 Images encoded in given audio file

3.2 Output

The output is obtained as given below after the audio signal is decoded by the receiver.

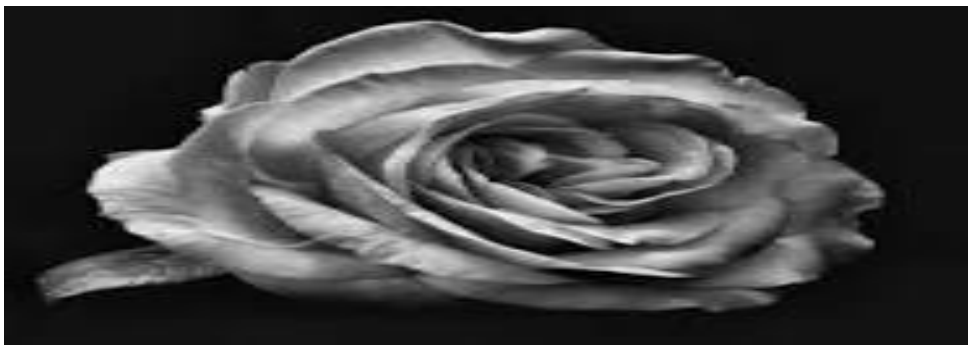

Figure.3.2.1 Revealed images as output images in all cases.

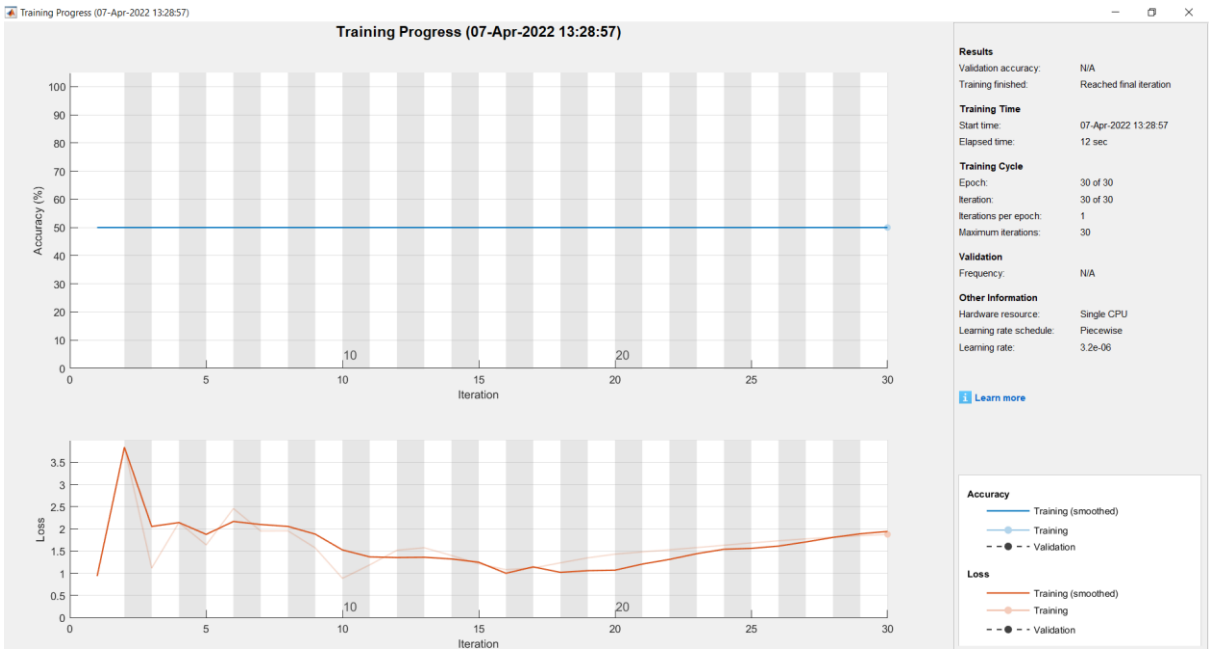

Figure 3.2.2 Training progress of given first color image data

Epoch	Iteration	Time Elapsed (00:00:00) hrs	Mini batch accuracy	Mini batch Losses	Base Learning rate
01	01	00:00:02	50.00%	1.0519	0.0100
30	30	00:00:13	50.00%	1.1556	3.200e^-06

Figure 3.2.3 Input data normalization and accuracy and losses of given first color image data

Figure 3.2.4 Training progress of given black and white image data

Epoch	Iteration	Time Elapsed (00:00:00) hrs	Mini batch accuracy	Mini batch Losses	Base Learning rate
01	01	00:00:01	50.00%	1.0019	0.0100
30	30	00:00:12	50.00%	1.0556	3.00e^-06

Figure 3.2.5 Input data normalization and accuracy and losses of given black and white image data

Figure 3.2.6 Training progress of given second color image data

Epoch	Iteration	Time Elapsed (00:00:00) hrs	Mini batch accuracy	Mini batch Losses	Base Learning rate
01	01	00:00:01	50.00%	0.9364	0.0100
30	30	00:00:12	50.00%	1.8756	3.200e^-06

Figure 3.2.7 Input data normalization and accuracy and losses of given second color image data

Figure 3.2.8 Training progress of given second black and white image data

Epoch	Iteration	Time Elapsed (00:00:00) hrs	Mini batch accuracy	Mini batch Losses	Base Learning rate
01	01	00:00:02	50.00%	1.0090	0.0100
30	30	00:00:14	50.00%	1.8700	3.200e^-06

Figure 3.2.9 Input data normalization, accuracy and losses of given second black and white image data

IV.CONCLUSION

In this paper, we present a deep learning-based cross-modal image-to-audio steganography platform. Instead of explicitly hiding the secret image, the suggested solution embeds the secret image's residual flaws into the cover audio in a multi-stage fashion. The suggested method's hiding process causes residual errors to become increasingly sparse as the number of stages increases, which not only makes payload capacity control more flexible, but also makes hiding easier due to the sparsity of residual errors.

REFERENCES.

[1]. Shumeet Baluja, "Hiding images in plain sight: Deep steganography," Advances in Neural Information Processing Systems (NIPS), pp. 2069–2079, 2017.
[2]. Eric Cole and Ronald D. Krutz, "Hiding in plain sight: Steganography and the art of covert communication," 2003.
[3]. Ron G. Van Schyndel, Andrew Z. Tirkel, and Charles F. Osborne, "A digital watermark," in IEEE International Conference on Image Processing (ICIP), 1994.
[4]. Raymond B. Wolfgang and Edward J. Delp, "A watermark for digital images," IEEE International Conference on Image Processing (ICIP), 1996.
[5]. M. Asad, J. Gilani, and A. Khalid, "An enhanced least significant bit modification technique for audio steganography," in International Conference on Computer Networks and Information Technology, 2011, pp. 143–147.
[6]. Toma´s Pevn ˇ y, Tom ´ a´s Filler, and Patrick Bas, "Using high-dimensional image models to perform highly undetectable steganography," vol. 6387, pp. 161–177, 2010.
[7]. Vojtech Holub and Jessica Fridrich, "Designing steganographic distortion using directional filters," in IEEE Workshop on Information Forensic and Security, 2012.
[8]. Vojtech Holub, Jessica Fridrich, and Tom s Denemark, ˇ "Universal distortion function for steganography in an arbitrary domain," Eurasip Journal on Information Security, pp. 1–13, 2014.
[9]. Jiren Zhu, Russell Kaplan, Justin Johnson, and Li Fei-Fei, "Hidden: Hiding data with deep networks," Proceedings of the European Conference on Computer Vision (ECCV), pp. 657–672, 2018.
[10]. Ian J Good fellow, Jean Pouget-Abadie, Mehdi Mirza, Xu Bing, and Yoshua Bengio, "Generative adversarial nets," in International Conference on Neural Information Processing Systems (NIPS), 2014.
[11]. Denis Volkhonskiy, Ivan Nazarov, and Evgeny Burnaev, "Steganographic generative adversarial networks," 2017.
[12]. Haichao Shi, Jing Dong, Wei Wang, Yinlong Qian, and Xiaoyu Zhang, "Ssgan: Secure steganography based on generative adversarial networks," 2017.
[13]. Jamie Hayes and George Danezis, "Generating steganographic images via adversarial training," in Advances in Neural Information Processing Systems 30, pp. 1954– 1963. Curran Associates, Inc., 2017.
[14]. Atique Ur Rehman, Rafia Rahim, M Shahroz Nadeem, and Sibt Ul Hussain, "End-to-end trained cnn encode decoder networks for image steganography," 2017.

[15]. Pin Wu, Yang, and Xiaoqiang Li, "Image into-image steganography using deep convolutional network," in Advances in Multimedia Information Processing - PCM 2018 - 19th Pacific-Rim Conference on Multimedia, Hefei, China, September 21-22, 2018, Proceedings, Part II. 2018, vol. 11165 of Lecture Notes in Computer Science, pp. 792–802, Springer.
[16]. Felix Kreuk, Yossi Adi, Bhiksha Raj, Rita Singh, and Joseph Keshet, "Hide and speak: Deep neural networks for speech steganography," ArXiv preprint ArXiv: 1902.03083, 2019.
[17]. Dengpan Ye, Shunzhi Jiang, and Jiaqin Huang, "Heard more than heard: An audio steganography method based on gan," ArXiv preprint ArXiv: 1907.04986, 2019.
[18]. Soumith Chintala Adam Paszke, Sam Gross and Gregory Chanan, "Pytorch: Tensors and dynamic neural networks in python with strong gpu acceleration," 2017.
[19]. Diederik Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," Computer Science, 2014.
[20]. Shiqi Dong Ru Zhang and Jianyi Liu, "Invisible steganography via generative adversarial networks," 2018.
[21]. Keith Ito, "The lj speech dataset. https://keithito.com/- ljspeech-dataset/," 2017.
[22]. Wuzhen Shi, Feng Jiang, Shengping Zhang, and Debin Zhao, "Deep networks for compressed image sensing," IEEE International Conference on Multimedia and Expo (ICME), pp. 877–882, 2017.
[23]. Wenxue Cui, Feng Jiang, Xinwei Gao, Shengping Zhang, and Debin Zhao, "An efficient deep quantized compressed sensing coding framework of natural images," in 2018 ACM Multimedia Conference on Multimedia Conference, MM 2018, Seoul, Republic of Korea, October 22-26, 2018. 2018, pp. 1777–1785, ACM. [24] Victor Zue, Stephanie Seneff, and James Glass, "Speech database development at mit: Timits and beyond," Speech Communication, vol. 9, no. 4, pp. 351–356, 1990