ISSN 2303-4521
Periodicals of Engineering and Natural Sciences *Original Research*
Vol. 10, No. 3, June 2022, pp.380-386

Text to speech using Mel-Spectrogram with deep learning algorithms

Abdulamir A. Karim[1], Suha Mohammed Saleh[2]
[1]Computer Science Department, Technology University, Iraq
[2]Computer Science Department, Technology University, Iraq

ABSTRACT

The purpose of text to speech (TTS), sometimes called speech synthesis, is to synthesize a natural and intelligible speech for a given text. A wide range of applications uses TTS technologies in media, chatbots, and entertainment, among other fields, making it a hot topic for the research community. Recently, the progress achieved by artificial intelligence, especially in deep learning and neural networks, enables TTS to produce a high-quality synthesized speech. However, despite the success achieved, currently, available works suffer from the need for very long training and inference time, which makes it dominated by big tech companies. This paper proposes a model based on convolutional neural networks (CNN) and gated recurrent units (GRU). The proposed model can work even in low computational environments and requires low training time. The MOS achieved is 4.26, higher than the MOS performed by state-of-the-art methods.

Keywords: First keyword, Second keyword, Third keyword, Fourth keyword, Fifth keyword

Corresponding Author:

Suha Mohammed Saleh
Computer Science Department, Technology University
Baghdad, Iraq
suhaa74.2016@gmail.com

1. Introduction

Text to speech TTS aims to generate a natural speech indistinguishable from human and audio records using a specific text [1]. TTS has a wide range of applications in many fields (entertainment [2], media [3], teaching [4], chatbots, and robotics [5], among others). Therefore, it has been a hot topic for researchers in many fields of artificial intelligence AI [6-8], such as speech processing and natural language processing NLP [9]. In addition, For building a TTS model, knowledge in many fields such as machine learning ML, signal processing [10], acoustics [11], and linguistics [12] is required. The development in deep learning DL helped build many TTS models based on neural networks [13-19]. Many methods were suggested since the first state-of-the-art algorithm, the WaveNet [14], achieved acceptable results using GPU computers for training and inferencing, aiming to build an effective model that requires lower computational power and can be implemented on slower CPU computers and embedded systems such as mobile phones. The traditional speech synthesis methods used complex techniques and required extracting linguistic features. Extracting these features requires domain knowledge and is expensive in terms of man power and time-consuming. Furthermore, the synthesized speech has glitches and pronunciation problems which make it looks unnatural [20]. In recent years, the progress in neural network fields helped produce end-to-end speech synthesis models. For instance, Tacotron 1 [15] and Tacotron 2 [21] are neural network-based models which replace the extracting linguistic features steps with a neural network, then a vocoder based on the Griffin Lim algorithm [22] is used to synthesize the audio, the resulted audio quality was enhanced noticeably for the mentioned method. However, the main disadvantages of these methods are the long training time and the need for high computational power [23]. Inspired by the success achieved in the previously mentioned works, this paper proposes a neural network-based model that requires lower training time and computational power to make it available for both research and manufactured fields. The main contribution of this paper can be summarized as follow:

1- A simple 1D residual CNN and GRU are used instead of the Griffin Lim algorithm, which speeds up the synthesis time.
2- The synthesized audios are higher in quality and more natural than the previous state-of-the-art methods.

The remainder of their paper can be explained as follows:
Section 2 provides a brief of studies relevant to TTS categorization. Section 3 describes the methods used. Section 4 presents the experiment. Section 5 discusses the obtained results. Finally, section 6 concludes.

2. Related works

The traditional methods before using AI required human feature extraction preprocessing steps, and the results were weak in terms of naturalness and intelligibility. This section focuses on models which use neural networks as the backbone for TTS. Wavenet [14] is the first neural network-based speech synthesized. In Wavenet, the linguistic features extracted are used directly to generate the waveform. In DeepVoice [24], a variant version of the Wavenet model is used with fewer parameters to reduce the training time. Besides, a deep neural networks model with connectionist tampered classification (CTC) loss is used for phoneme boundary detection. In DeepVoice 2 [21], the quality achieved by the DeepVoice 1 model is enhanced by introducing a post-processing neural vocoder. Tacotron [15] proposed an end-to-end model directly generating the characters' speech using a from-scratch trained model. Tacotron 2 [21] used the attention mechanism to help in producing the Mel-spectrogram frames from the input characters. Furthermore, a variant Wavenet version is used to conditionally generate the waveform samples from the Mel-spectrogram frames. FastSpeech 1 and 2 [25-26] proposed a simplified acoustic feature extraction module to generate the waveform from the Mel-spectrograms. Finally, in Clarinet [27], a fully end-to-end text-to-wave model is proposed, a fully convolutional-based model on the original Wavenet model. Despite all the high results achieved by the presently mentioned works, the TTS field is still an open field for research that requires improving the synthesized audio quality and reducing the training and inference time and can be used in low computer power devices [28-32].

3. Materials and methods

The proposed model to synthesize audio for a given text can be illustrated in Figure 1. The input text passes throw many steps: 1) linguistic feature extraction from the text by converting the text characters into phonemes using text analysis; 2) generating the acoustic features from the linguistic phonemes in the forms of Mel-spectrogram; 3) generating the waveform from the Mel-spectrograms using the vocoder. The following sections explain each part of the proposed model briefly.

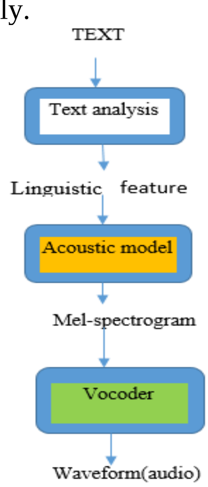


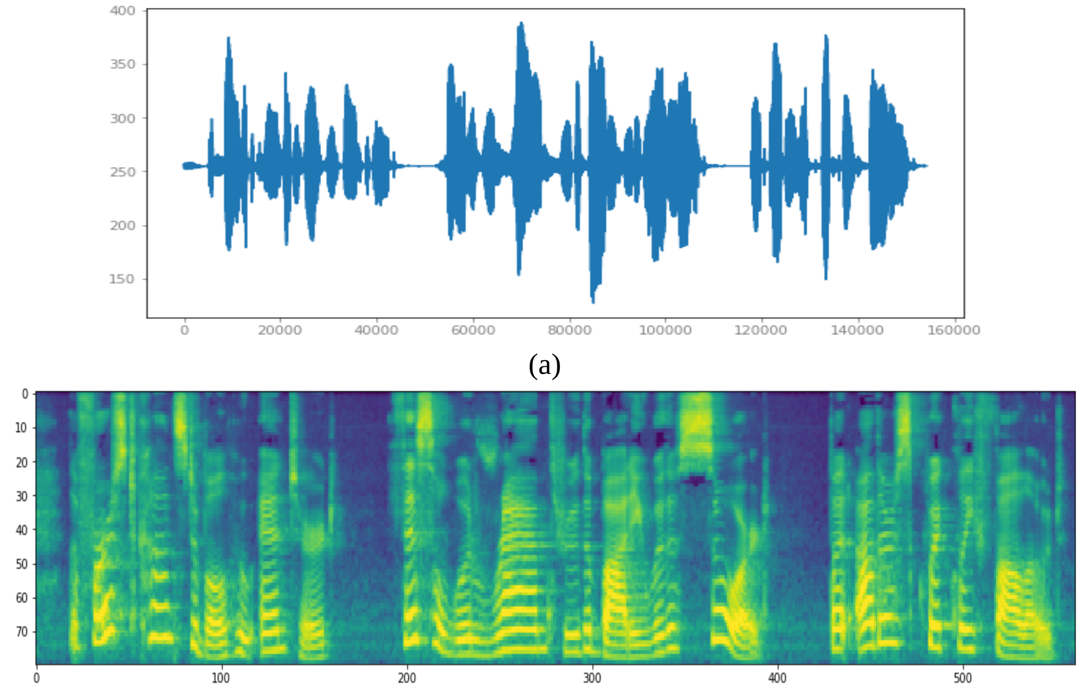
Figure 1. The proposed TTS model

3.1. Text analysis TA

TA is considered the front-end part of text synthesis. This step extracts essential linguistic features from the text to provide pronunciation and prosody information required for natural speech synthesis. In addition, many regular expression features are used in TA, such as:
Text normalization function: converts the row of written text into spoken words (for instance, the year 2022 is normalized into two thousand twenty-two, and abbreviations such as Mrs. are normalized into mister).
White space collapsing: to identify the steps. Lower_case converter function: converts the text into lower case.Convert into ASCII code function Prosody prediction function: identifies the speech stress, rhythm, and intonation. After the above processing, the linguistic features extracted be ready for the next step to generate the Mel-spectrogram.

3.2. Acoustic model AM

The most widespread form of acoustic features is the Mel-spectrograms used by the vocoder to generate the last waveform of the synthesized audio. The spectrogram is a way of inferring audio data and converting it into an image where the vertical axis represents the audio frequency. The horizontal axis represents the time, while the color intensity represents the frequency amplitude at a particular time point. Short-time Fourier transformation (STFT) algorithm is used to obtain the spectrogram from the audio. In Mel-spectrogram, the frequencies are converted into a Mel-scale[33-38]. According to the University of California, the Mel-scale can be defined as "a perceptual scale of pitches judged by listeners to be equal in distance one another". Figure 2 shows the audio sample and its corresponding Mel-spectrogram from the dataset used in this paper.



(a)

(b)

Figure 1. (a) Audio sample from the used dataset. (b) The Mel-spectrogram for the above audio sample

3.3. Vocoder

The vocoder aims to convert the generated Mel-spectrograms into final waveform synthesized audio. Using the output of multi residual 1D convolutional neural network CNN layers concatenated with multi 2D CNN layers for up-sampling. Finally, dense and GRU layers are used to generate waveforms conditioned on extracted linguistic features. The detailed constriction of the vocoder used in this work is shown in Figure 3.

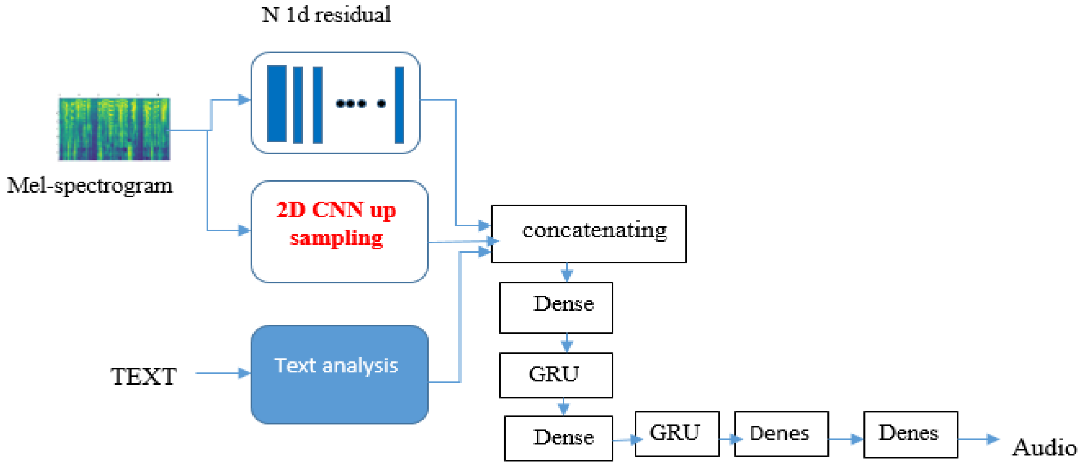


Figure 2. The vocoder structure

3.4. Dataset

The proposed model was trained on the publicly available LJ Speech dataset (keithito.com). The dataset consists of 13,100 1-10 second-length files for a single female speaker reading parts of seven different books. In addition, the transcript of each audio is provided in text file form. The total audio length is about 24 hours. The whole dataset was used to train the model except for 50 randomly chosen utterances used for evaluation.

3.5. Model training

The training process was conducted using Google Colab with a GPU environment activated. For 20 epochs and 16 batch sizes using Adam optimizer with 0.0001 for the learning rate. The Python programming language is used with the TensorFlow library.

3.6. Loss function

Negative log-likelihood NLL is used to calculate the loss in this work. First, −log(y) is calculated, where y is a prediction corresponding to the ground truth after the Softmax Activation Function was applied. Then, the loss for a mini-batch is computed by taking the mean or sum of all items in the batch. Since a negative value is returned for the log of a number >= 0 and < 1, we add a negative sign to convert it to a positive number, hence the *negative* log likelihood. At 0, the function returns ∞ (−log(0)=∞), and at one returns 0 (−log(1)=0), so very wrong answers are heavily penalized. The function plot is as follows in Figure 3.

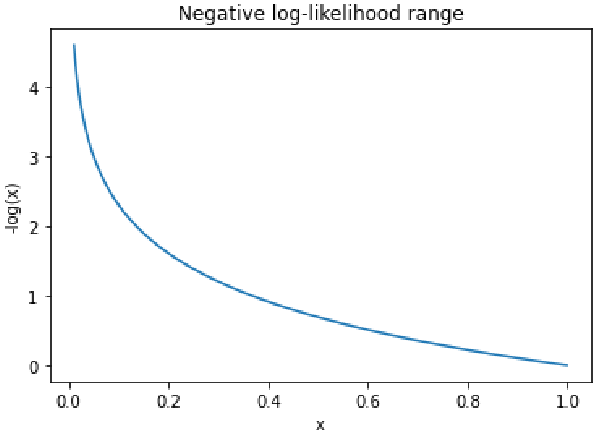


Figure 3. NLL function plot

3.7. Evaluation

To evaluate the results, the mean opinion score MOS is used. MOS is a numerical measurement that represents the human evaluation of an event. It is widely used in telecommunications to rank voice and video quality. A 1 to 5 scale is mainly used, where 1 means bad, and 5 means excellent; then the average number of other human-scored individuals is calculated.

4. Results and discussion

A survey with 30 people was conducted to measure and compare the synthesized audio quality. The 25 audio files mentioned earlier, kept for testing, are used in this survey by synthesizing the audio using the proposed model. Finally, the comparison was made using the ground truth audios and the results obtained by the Tacotron 2 model, as shown in Table 1.

Table 1. The results of MOS for the proposed model compared with the results of the Tscotron 2 model and the ground truth audios

Model	**MOS**
Tacotron 2	4.14 ± 0.05
The proposed model	4.25 ± 0.05
Ground truth	4.42 ± 0.05

Fifteen participants evaluated anonymized outputs to rate them into five categories: "Bad", "Poor", "Good", "Fair", and "Excellent". The results show that the synthesized audio obtained by the proposed model outperformed the results obtained by the Tacotron 2 model and came closer to the ground truth result. Furthermore, the training time for a single training step is ~ 0.3 seconds, which is about 4.75 times faster than that of the Tacotron 2 (~ 1.7 seconds) with the same batch size of 16. In comparison, the inference time is varied between 0.015- 0.018 seconds for text between (12- 36) words compared with 0.085- 0.167 seconds when using the Tacotron 2 model under the same training environment. The relatively low inference time of the proposed model makes it suitable to be used in mobile devices or low computational power devices. The loss diagram for the training process is shown in Figure 4, and it can be noticed that the loss decreased significantly from 3.516 at the first epoch to 1.472 at the 20th epoch.

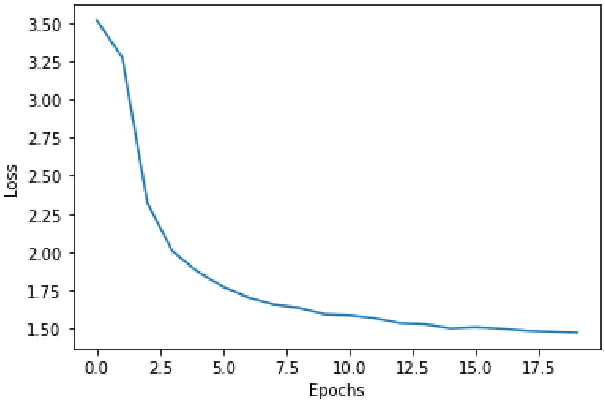


Figure 4. Loss values per epoch for the training process

5. Conclusions

This paper proposed a neural network-based audio synthesized model. CNN and GRU layers are used in this model in the vocoder part. The generated audios were closer to the ground truth audios than the state-of-the-art Tocotron 2 model, as shown in the results section. Furthermore, the training and inference time increased significantly, believing that the proposed model can be used with low computational power devices. Future work includes investigating the proposed method for multiple speakers' text synthesizing and adding more techniques for pitch prediction to add more naturalness to the generated audios.

Declaration of competing interest

The authors declare that they have no any known financial or non-financial competing interests in any material discussed in this paper.

Funding information

No funding was received from any financial organization to conduct this research.

References

[1] P. Taylor, *Text-to-speech synthesis*, vol. 9780521899277. 2009. doi: 10.1017/CBO9780511816338.

[2] S. Kato, Y. Yasuda, X. Wang, E. Cooper, S. Takaki, and J. Yamagishi, “Modeling of Rakugo Speech and Its Limitations: Toward Speech Synthesis That Entertains Audiences,” *IEEE Access*, vol. 8, 2020, doi: 10.1109/ACCESS.2020.3011975.

[3] O. Bendel, “The synthetization of human voices,” *AI and Society*, vol. 34, no. 1, 2019, doi: 10.1007/s00146-017-0748-x.

[4] J. Matoušek *et al.*, “Speech and web-based technology to enhance education for pupils with visual impairment,” *Journal on Multimodal User Interfaces*, vol. 14, no. 2, 2020, doi: 10.1007/s12193-020-00323-1.

[5] A. Kotov, N. Arinkin, L. Zaidelman, and A. Zinina, “Linguistic approaches to robotics: From text analysis to the synthesis of behavior,” in *Communications in Computer and Information Science*, 2019, vol. 943. doi: 10.1007/978-3-030-05594-3_16.

[6] “2021 Index IEEE/ACM Transactions on Audio, Speech, and Language Processing Vol. 29,” *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 29, 2022, doi: 10.1109/taslp.2022.3147096.

[7] C. D. Manning, H. Schütze, and G. Weikurn, “Foundations of Statistical Natural Language Processing,” *SIGMOD Record*, vol. 31, no. 3, 2002, doi: 10.1145/601858.601867.

[8] A. Ligeza, “Artificial Intelligence: A Modern Approach,” *Neurocomputing*, vol. 9, no. 2, 1995, doi: 10.1016/0925-2312(95)90020-9.

[9] A. Galassi, M. Lippi, and P. Torroni, “Attention in Natural Language Processing,” *IEEE Transactions on Neural Networks and Learning Systems*, vol. 32, no. 10, 2021, doi: 10.1109/TNNLS.2020.3019893.

[10] R. Dastres and M. Soori, “A Review in Advanced Digital Signal Processing Systems,” *International Journal of Electrical and Computer Engineering*, vol. 15, no. 3, 2021.

[11] A. Ozcelik, J. Rich, and T. J. Huang, “Fundamentals and applications of acoustics in microfluidics,” in *Multidisciplinary Microfluidic and Nanofluidic Lab-on-a-chip*, 2022. doi: 10.1016/b978-0-444-59432-7.00016-9.

[12] L. E. B. Key and B. P. Noble, *Course in General Linguistics*. 2017. doi: 10.4324/9781912281732.

[13] H. Ze, A. Senior, and M. Schuster, “Statistical parametric speech synthesis using deep neural networks,” 2013. doi: 10.1109/ICASSP.2013.6639215.

[14] A. van den Oord *et al.*, “WaveNet: A Generative Model for Raw Audio Based on PixelCNN Architecture,” *arXiv*, 2016.

[15] Y. Wang *et al.*, “Tacotron: Towards end-To-end speech synthesis,” in *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, 2017, vol. 2017-August. doi: 10.21437/Interspeech.2017-1452.

[16] N. Kalchbrenner *et al.*, “Efficient neural audio synthesis,” in *35th International Conference on Machine Learning, ICML 2018*, 2018, vol. 6.

[17] W. Ping *et al.*, “Deep Voice 3: 2000-Speaker Neural Text-to-Speech,” 2018.

[18] N. Li, S. Liu, Y. Liu, S. Zhao, and M. Liu, “Neural speech synthesis with transformer network,” 2019. doi: 10.1609/aaai.v33i01.33016706.

[19] H. B. Moss, V. Aggarwal, N. Prateek, J. Gonzalez, and R. Barra-Chicote, “BOFFIN TTS: Few-Shot Speaker Adaptation by Bayesian Optimization,” in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 2020, vol. 2020-May. doi: 10.1109/ICASSP40776.2020.9054301.

[20] Prof. M. Patole, A. Pandey, K. Bhagwat, M. Vaishnav, and S. Chadar, “A Survey on ‘Text-to-Speech Systems for Real-Time Audio Synthesis,’” *International Journal of Advanced Research in Science, Communication and Technology*, 2021, doi: 10.48175/ijarsct-1400.

[21] J. Shen *et al.*, “Natural TTS Synthesis by Conditioning Wavenet on MEL Spectrogram Predictions,” in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 2018, vol. 2018-April. doi: 10.1109/ICASSP.2018.8461368.

[22] N. Perraudin, P. Balazs, and P. L. Sondergaard, “A fast Griffin-Lim algorithm,” 2013. doi: 10.1109/WASPAA.2013.6701851.

[23] J. Liu, Z. Xie, C. Zhang, and G. Shi, “A novel method for Mandarin speech synthesis by inserting prosodic structure prediction into Tacotron2,” *International Journal of Machine Learning and Cybernetics*, vol. 12, no. 10, 2021, doi: 10.1007/s13042-021-01365-x.

[24] H. Zhang, A. Wang, D. Li, and W. Xu, “DeepVoice: A voiceprint-based mobile health framework for Parkinson’s disease identification,” in *2018 IEEE EMBS International Conference on Biomedical and Health Informatics, BHI 2018*, 2018, vol. 2018-January. doi: 10.1109/BHI.2018.8333407.

[25] V. Popov *et al.*, “Fast and lightweight on-device TTS with Tacotron2 and LPCNet,” in *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, 2020, vol. 2020-October. doi: 10.21437/Interspeech.2020-2169.

[26] Y. Ren *et al.*, “FastSpeech: Fast, robust and controllable text to speech,” in *Advances in Neural Information Processing Systems*, 2019, vol. 32.

[27] W. Ping, K. Peng, and J. Chen, “Clarinet: Parallel wave generation in end-to-end text-to-speech,” 2019.

[28]A. Sallomi, S. A. Hashem, and Y. Mezaal, “A novel theoretical model for cellular base station radiation prediction,” *Int. J. Simul. Syst. Sci. Technol.*, 2019.

[29] Y. S. Mezaal and M. Al-Ogaidi, “Performance analysis of negative group delay network using MIMO technique,” *TELKOMNIKA*, vol. 18, no. 5, p. 2572, 2020.

[30] Y. Mezaal, “Investigation of PAPR reduction technique using TRC-SLM integration,” *Int. J. Simul. Syst. Sci. Technol.*, 2019.

[31] M. S. Shareef, T. Abd, and Y. S. Mezaal, “Gender voice classification with huge accuracy rate,” *TELKOMNIKA*, vol. 18, no. 5, p. 2612, 2020.

[32] B. H. Majeed, and L. F. Jawad, "Computational Thinking (CT) Among University Students," *International Journal of Interactive Mobile Technologies,* vol. 16, no. 10, 2022.

[33] H. Tauma, and N. Alseelawi, "A Novel Method of Multimodal Medical Image Fusion Based on Hybrid Approach of NSCT and DTCWT," International journal of online and biomedical engineering, vol. 18, no. 3, 2022.

[34] H. Alrikabi, and H. Tauma, "Enhanced Data Security of Communication System using Combined Encryption and Steganography," International Journal of Interactive Mobile Technologies, vol. 15, no. 16, pp. 144-157, 2021.

[35] H. Salim, and A. Ibtisam A. Aljazaery, "Encryption of Color Image Based on DNA Strand and Exponential Factor," International journal of online and biomedical engineering(iJOE), vol. 18, no. 3, 2022.

[36] S. H. Abbood, M. S. Rahim, and A. M.Alaidi, "DR-LL Gan: Diabetic Retinopathy lesions synthesis using Generative Adversarial Network," *International journal of online and biomedical engineering,* vol. 18, no. 3, 2022.

[37] H. TH., and N. A. Jasim, "Design and Implementation of Smart City Applications Based on the Internet of Things," International Journal of Interactive Mobile Technologies (iJIM), vol. 15, no. 13, pp. 4-15, 2021.

[38] I. A. Aljazaery, H. T. Salim, and M. R. Aziz, "Combination of Hiding and Encryption for Data Security," International Journal of Interactive Mobile Technologies, vol. 14, no. 9, pp. 34-47, 2020.