ACADEMY Publishing Center

RIMC

ROBOTICS: INTEGRATION, MANUFACTURING & CONTROL

VOLUME 1, ISSUE 2, DECEMBER 2024

ISSN: 3009-7967

ACADEMY PUBLISHING CENTER
Robotics : Integration, Manufacturing and Control (RIMC)
First Edition 2024

ACADEMY Publishing Center

RIMC
Volume 1, Issue 2, December 2024
E-ISSN: 3009-7967 Online Version
P-ISSN: 3009-6987 Print Version

Arab Academy for Science, Technology, and Maritime Transport, AASTMT Abu Kir Campus, Alexandria, EGYPT
P.O. Box: Miami 1029
Tel: (+203) 5622366/88 – EXT 1069 and (+203) 5611818
Fax: (+203) 5611818
Web Site: http://apc.aast.edu

RIMC Robotics : Integration, Manufacturing and Control

Journal of Robotics: Integration, Manufacturing and Control is an open access biannual international peer-reviewed publication providing a global platform for the dissemination of research articles, case studies, and reviews across diverse disciplines, focusing on robotics integration and manufacturing.

RIMC strives to contribute essential research findings to the international community, aiding researchers, scientists, institutions, and societies in staying abreast of new developments in theory and applications. We welcome experimental, computational, and theoretical studies that enrich the understanding of robotics related challenges.

RIMC also publishes survey and review articles in the scope. The scope of RIMC includes but is not limited to the following topics: technologies for manufacturing applications, Control algorithms for robots, Sensor technologies, Human-robot collaboration, Robotics for Industry 4.0 and sustainable manufacturing.

The journal is financially supported by the Arab Academy for Science, Technology, and Maritime Transport AASTMT in order to maintain quality open-access sources of research papers on Computing and Engineering.

RIMC has an outstanding editorial and advisory board of eminent scientists, researchers and engineers who contribute and enrich the journal with their vast experience in different fields of interest to the journal.

Editorial Committee:

Advisory Board:

Administrative Board:

Table of Content:

A Lightweight Speaker Verification Approach For Autonomous Vehicles

Yousef Salah [1], Omar Shalash [2*], and Esraa Khatab [3]

[1,2] College of Artificial Intelligence, Arab Academy for Science and Technology and Maritime Transport, New Alamein, Egypt.
[3] School of Mathematics and Computer Science Herriot Watt University, Dubai Knowledge Park, Dubai, UAE.
[1,2,3] Research and Innovation Center, Arab Academy for Science, Technology and Maritime Transport, Alamein, Egypt.

y.s.semary@student.aast.edu, omar.o.shalash@aast.edu, e.lhatab@hw.ac.uk

Received on, 29 October 2024 Accepted on, 03 December 2024 Published on, 22 December 2024

ABSTRACT

Speaker verification is the process of verifying an individual's identity by comparing their recorded voice samples with their test speech signals. Speaker verification has various practical applications, such as verifying customer identities in call centers, enabling contactless facility access, and supporting some medical applications. With the advances in autonomous vehicles, speaker verification has become an essential feature that provides security, access control, personalization, command authentication, driver monitoring, and compliance. Recent technological advancements have led to the rise of voice-based authentication systems, which are considered a more convenient alternative to traditional security systems. However, improving the accuracy is still an ongoing research aim. In this research, four different models were proposed and compared with previous work on speaker verification. The models are combinations of using two networks (BiLSTM and Transformer) with two different loss functions (Triplet and Quadruplet loss functions). The models are trained and tested on the LibriSpeech dataset. The results show improvements in equal error rate of the four proposed models over the previous models that used the Librispeech dataset with 0.068 compared to 0.11.

Key-words: Speaker Verification, Tranformer Network, BiLSTM Networm, Driver Personalization, Command Authentication

I. INTRODUCTION

Speaker verification (SV) is the act of authenticating an individual's claimed identity by comparing their recorded voice samples to their test speech signals. It has various applications, including verifying customer identities in call centers, enabling contactless facility access, and providing support for medical applications to recognize and perform operator's commands to fully automate the system as presented in [1; 2; 3; 4]. As one lives in the age of information, several applications have required artificial intelligence such as digital twins, swarm intelligence, and data fusion [5; 6]. All these applications require more security as a layer for protection [7; 8; 9]. Recent technological advancements have led to the rise in popularity of automatic speaker verification systems, which are now considered a more convenient alternative to traditional security systems [10; 11]. SV has become an essential technology in numerous real-world applications, such as biometric authentication and security systems [12]. While significant advancements have been made, challenges remain in optimizing system performance under varying conditions. Recent research has largely focused on improving SV systems by experimenting with various loss functions, pooling methods, and network architecture designs, aiming to better capture speaker characteristics and improve robustness [13; 14; 15]. However, despite these efforts, there is still room for innovation, particularly in exploring novel combinations of network architectures and loss functions that can push the boundaries of current SV systems.

SV can be classified into two types: Text-Dependent (TD) and Text-Independent (TI) SV. TD-SV requires that the spoken content of the test utterance and the enrollment utterance be the same, while TI-SV has no restrictions on the spoken content [16; 17].

SV has contributed heavily to automation in vehicles. Nowadays a passenger can voice activate an access control system [18; 19; 20; 21]. SV can be used to personalize passenger settings, for example, based on the driver's identity the vehicle can adjust the setting autonomously such as a seat and mirror position which improves the driving experience [22; 23]. In terms of safety, SV can be used to monitor the driver and his compliance, which leads to monitoring driver attentiveness by recognizing voice patterns that indicate fatigue or stress [24; 25; 26; 27; 28].

TABLE I. COMPARISON BETWEEN FEATURE EXTRACTION METHODS

Method	Strengths	Weaknesses
LPC	• Easy implementation	• Noise-sensitive • High time and computational cost • Inconsistency with human hearing
LPCC	• Stable representation • Decorrelated feature components	• Quantisation noise-sensitive • Insufficient order causes performance degradation
MFCC	• Behaves like human ear • Captures the main characteristics of phones in speeches with low complexity	• Low robustness • Fixed time-frequency resolution

In this research, the researchers propose four novel different models for SV. The significant outcomes are:

1. Four novel models have been developed for SV. Two models use BiLSTM networks, the other two use the Transformer Network. Each network has been evaluated through different loss functions. The combination between the networks and the loss functions produces the four proposed models.

2. A novel adaptation of the Siamese network.

II. RELATED WORK

The human voice is universally used for exchanging information between individual speaker recognition that involves identifying individuals based on unique vocal characteristics. This field has gained significant research attention due to its broad applications. Speaker recognition is the automatic process of identifying a speaker based on their speech signal. It can be divided into six categories: speaker identification, speaker verification, speaker detection, speaker segmentation, speaker clustering, and speaker diarization. [29] Speech signals carry speaker-specific features that can be extracted and used by Machine Learning (ML) algorithms to recognize specific patterns [30].

The basic concept of feature extraction is to extract a set of features for each segment of the input signal, based on the idea that short-time segments are sufficiently stationary for improved modeling [31]. Feature extraction captures relevant and crucial information from the speech signal while discarding irrelevant and redundant data [32; 33; 34]. This step is essential for the subsequent modeling process. The speaker signal, as part of a dependent speech system, is analyzed to reduce variability and enhance the extraction of discriminative features by converting the speech signal into parametric values [35]. Various techniques, such as Linear Prediction Coding (LPC), Linear Prediction Cepstral Coefficients (LPCCs), and Mel-Frequency Cepstral Coefficients (MFCCs), [32; 36] can be used to extract speech features in the form of coefficients. Table I presents a comprehensive comparison between the well-known feature extraction methods [29].

MFCC features have been widely used in different research addressing specific challenges, such as noise robustness systems [37; 38], dysarthric speaker verification [39], twins' voice identification [40]. Moreover, Numerous studies have asserted that MFCC effectively boosts speaker recognition. For example, Singh et al. [41] evaluated three features for automatic speech recognition, including MFCC, dynamic time wrapping, and fast Fourier transform. It was proven that MFCC improves the performance of the model. Moreover, Abdul et al. [42] have shown that MFCC features could efficiently be fed to Convolutional Neural Networks (CNNs) to train it to distinguish between speakers. However, Faek et al. [43] have shown that speaker recognition using MFCC and k-NN is negatively affected in noisy environments; which encouraged the inclusion of a denoising step. Additionally, Jahangir et al. [44] proposed a novel fusion of Mel frequency cepstral coefficients (MFCC) and time-based features (MFCCT) to identify speakers using a hierarchical classification approach. The approach was implemented in a cascading style, where the first level identified the speaker's gender, and the second level identified the specific speaker's identity. The study used five machine learning algorithms and a deep learning-based Deep Neural Networks (DNN) to classify speaker gender and Speaker ID (SID). The model was trained and tested on the LibriSpeech corpus dataset [45]. The results showed an overall accuracy of 83.5%-93%.

Balipa et al. [40] proposed a method for twins' voice identification and verification. The proposed method involves using a Siamese Neural Network (SNN) to extract features from the voice dataset and calculate the relationship between audio signals and linguistic units that make up speech. The proposed method was evaluated on the twin dataset and the results were compared with a corpus of similarly obtained data from unrelated individuals. The testing results showed an accuracy of about 78% with a loss of 0.10. To identify speakers in this scenario, the system complied with the given testing regimen and yielded an accuracy of approximately 78% with a loss of 0.10. Despite being commonly used in image processing, SNN was utilized to compare the voices of twins in this study.

Niu et al. [46] presented a pseudo-phoneme label (PPL) loss value for the function of a network with delay over time domain based on TI-SR (text-independent speaker recognition). The PPL loss combines content array losses at the frame level and segment level into a combined network through multi-task learning. Various methods of PPL loss were compared and their effects on the ending system execution were explored. Model 1 uses multi-task learning to train the model, while Model 2 trains the vocabulary parts and assigns factors for pseudo-phoneme tags. Model 3 calculates the PPL loss at frame 4 layer using an attention mechanism. By the last result the values of all models are averaged. The model was trained and tested on the VoxCeleb dataset [47].

Zheng et al. [48] aimed to enhance the effectiveness of extracting speaker embedding by developing a multi-scale residual aggregation network (MSRANet). This new approach utilizes the triplet loss function to increase the similarity and the difference of interclass, resulting in better performance. Experimental results using three datasets (VoxCeleb1, VoxCeleb2, and LibriSpeech) showed that MSRANet outperformed previous approaches and achieved state-of-the-art performance, demonstrating its cross-scenario adaptability. However, there are some limitations to this approach, such as potential information redundancy caused by multiscale fusion.

Singh and Mahesh [49] evaluated the performance of different feature extraction approaches: MFCC, and Multiban Spectral

Entropy (MSE). They were integrated with different machine learning algorithms, such as K-NN, Random Forest, DNNs, and Decision Trees. Their work achieved competitive results.

Existing work on SV often faces challenges related to running time, especially in real-time applications. Common shortcomings include:

- **High computational Cost:** Complex models like DNN or speaker embeddings (e.g., x-vectors) require significant processing power, leading to longer inference times.

- **Hardware dependency:** Many models require specialized hardware (e.g., GPUs) to perform efficiently, limiting accessibility for broader applications.

- **Resource-intensive training:** Some approaches require extensive pre-training, which consumes time and resources. Fine-tuning these models for different environments or languages adds to the running time, making rapid deployment challenging.

These shortcomings highlight the need for more efficient models that balance accuracy and speed, as well asmethods that can streamline real-time performance without sacrificing verification quality.

III. METHODOLOGY

In this section, four models are proposed to achieve speaker verification with enhanced accuracy. The main backbone for the proposed models is the Siamese Neural Network (SNN). In the context of data processing and feature extraction, the MFCC algorithm has been used to represent the spectral characteristics of audio signals. After feature extraction, the data are either processed through the BiLSTM (Models 1 and 2) or transformer networks (Models 3 and 4). Data are then processed through a loss function; the triplet loss function (Models 1 and 3), and the Quadruplet loss function (Models 2 and 4).

Siamese networks are widely used to perform similarity comparisons that can be applied to complex data samples with features having different dimensionality and types. A Siamese network has two equivalent artificial neural networks, each qualified to learn the covered representation of an input vector. Both networks are feed-forward perceptrons and can detect error back-propagation while training; they work concurrently and analyze their outputs, usually through a cosine similarity [50; 6]. Siamese Networks are tied networks that take in pairs of input vectors and minimize or maximize a distance depending on whether a pair comes from the same or different classes [51].

Due to the presence of noise in audio signals, raw audio signals cannot be directly used as input to the SV models. Therefore, better performance could be achieved when extracting features from audio signals. MFCC is the most widely used technique for feature extraction from audio signals.

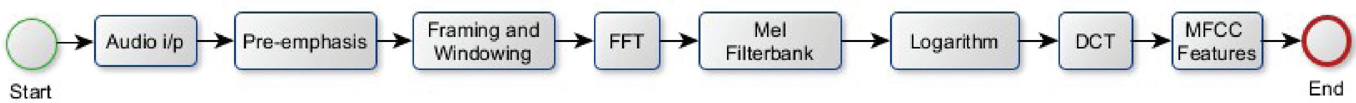

Figure 1. MFCC

The MFCC technique is shown in Figure 1 [52]. The process begins with an audio signal, typically sampled at 16KHz to capture the important frequencies for human hearing. Moreover, the pre-emphasis process filters the audio signal to emphasize higher frequencies. This step makes use of the fact that human hearing is more sensitive to lower frequencies. Next, framing is performed in order to divide the continuous signal into small, overlapping frames. This is because speech and audio signals are quasi-stationary, meaning that their characteristics are relatively constant over short durations. Each frame is multiplied by a window function to minimize the signal discontinuities at the edges of the frame.

Next, Fast Fourier Transform (FF) is applied to each windowed frame to convert the time-domain signal into the frequency domain, this step produces a spectrum that shows the magnitude of different frequency components in the signal. Afterward, the linear frequency spectrum is converted into the Mel scale, which mimics the human ear's sensitivity to different frequencies. The next step is taking the logarithm of the resulting filter bank energies to make the features more closely related to how many humans perceive sound intensity and emphasize the relative differences between frequency components. Discrete Cosine Transform (DCT) is applied to the log filter bank energies to decorrelate the features and compress the information to generate the Mel-Frequency Cepstral Coefficients (MFCCs), which represent the audio signal in a compact form by retaining the most useful information for tasks like speech recognition.

The Long Short-Term Memory (LSTM) architecture is originally designed to address the limitations of Recurrent Neural Networks (RNNs) in capturing long-term dependencies in sequential data. LSTM networks are a specific type of RNNs. While RNNs are designed to process sequential data by maintaining a hidden state that captures information from previous time steps, they suffer from issues like vanishing and exploding gradients, which limit their ability to capture long-term dependencies. LSTM networks were introduced as an extension of RNNs to overcome these limitations. By incorporating specialized gating mechanisms, LSTMs can maintain and update information over longer sequences, addressing the challenges present in traditional RNNs.

LSTMs incorporate a memory cell and three types of gates: Input, forget, and output gates. These gates regulate the flow of information and selectively retain or discard relevant information at each time step. Based on the same concept, Bidirectional LSTMs (BiLSTMs) introduce two separate LSTM layers. The first layer processes the sequence in the forward direction and the second one processes in the backward direction. By combining the outputs of both layers, BiLSTMs effectively capture dependencies from both past and future contexts. The base architecture of BiLSTM is as follows [53]:

- **Input sequence:** The sequential input data are divided into individual time steps
- **Forward LSTM layer:** It processes the input sequence from the beginning to the end, capturing information about the past context at each time step.
- **Backward LSTM layer:** It processes the input sequence in reverse order, capturing information about the future context at each time step.
- **Concatenation:** The outputs of both forward and backward LSTM layers are concatenated at each time step, combining the information from the past and future contexts into a single representation.

Similar to BiLSTM, transformers process sequential input data, however, they process the entire data at once. They produce a sequence of hidden representations that capture the contextual information of each token in the sequence. The benefit of an encoding layer in a transformer is capturing contextual information. The encoding layer uses self-attention mechanisms to attend to all positions in the input sequence and generate a context-aware representation for each token. This allows the model to capture the relationships between different tokens and their contextual information and residual connections to preserve the original input information in the hidden representations. This ensures that the model can learn the relevant features while still retaining the important information from the original input sequence.

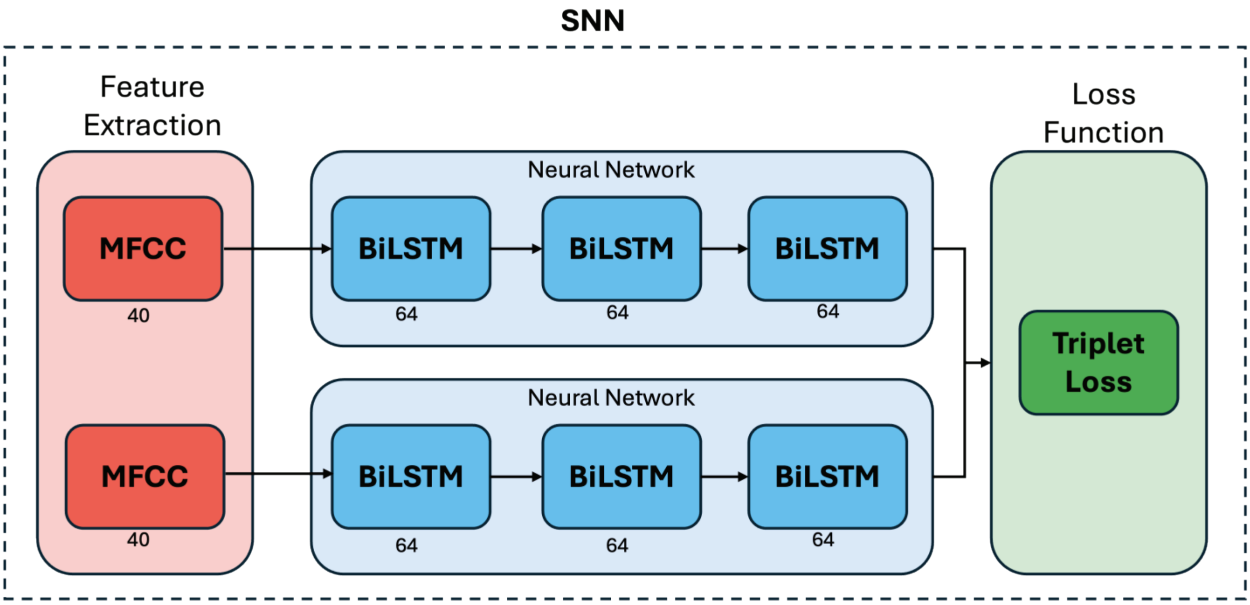

Figure 2. Model 1 – BiLSTM with triplet loss

The self-attention mechanism used in the encoding layer allows a transformer model to attend to different parts of the same input sequence. This allows the model to understand the relationships between different elements in the input and output sequences and make more accurate predictions, [54]. In the proposed models, the transformer network used consists of two encoding layers and one decoding layer.

In models 1 and 3 triplet-loss is implemented, to minimize the distance between the positive and anchor samples while increasing the space between the negative and anchor samples, with the margin term ensuring that the negative and positive samples are sufficiently far apart, as shown in equation (1) [55].

$$L(A,P,N) = max(0, d(A,P) - d(A,N) + m) \quad (1)$$

where *A* is the anchor, *P* is the positive sample, and *N* is the negative sample.

In models 2 and 4 quadruplet loss function is used, it takes four input samples: an anchor sample, a positive sample (similar to the anchor), a negative sample (different from the anchor), and a second negative sample (different to the anchor and first negative sample). It aims to increase the distance between the anchor and the negative samples while decreasing the distance between the anchor and the positive sample. The formula for the quadruplet-loss is defined in equation 2 [56].

$$L(A,P,N1,N2) = max(0, d(A,P) - d(A,N1) + m) + max(0, d(A,P) - d(A,N2) + m) \quad (2)$$

where *A* is the anchor, *P* is the positive sample, *N1* is the first negative sample, and *N2* is the second negative sample.

A. *Proposed Models*

Raw audio signals are first pre-processed by converting them into a mono channel (frequency= 16 kHz). In the four proposed models, SNNs have been adopted. Moreover, the MFCC technique has been used to perform feature extraction from raw audio signals, and 40 coefficients are extracted. Afterward, the employment of transformers and BiLSTM networks were interchanged, as well as the employment of triple loss and quadruplet loss functions in order to manifest their effect on the SV performance.

Model 1 (Figure 2) uses a three-layer BiLSTM network to extract the encoding of each speaker, afterwards, the result is applied to a triplet loss function. Model 2 (Figure 3) also utilizes a BiLSTM network, however, a quadruplet loss function is used instead of the triplet-loss function. On the other hand, Model 3 (Figure 4) utilizes a transformer to extract the encoding of each speaker with 32 dimensions in the model's hidden state and the embeddings. Also, this number represents the number of features in the input to the encoder layers.

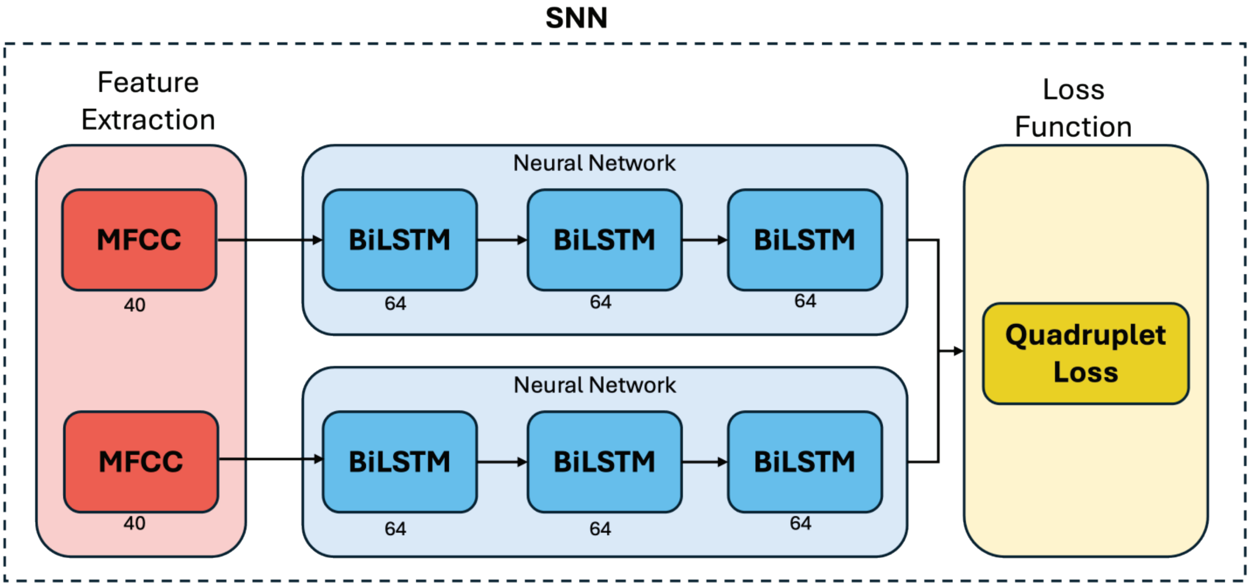

Figure 3. Model 2 - BiLSTM with quadruplet loss

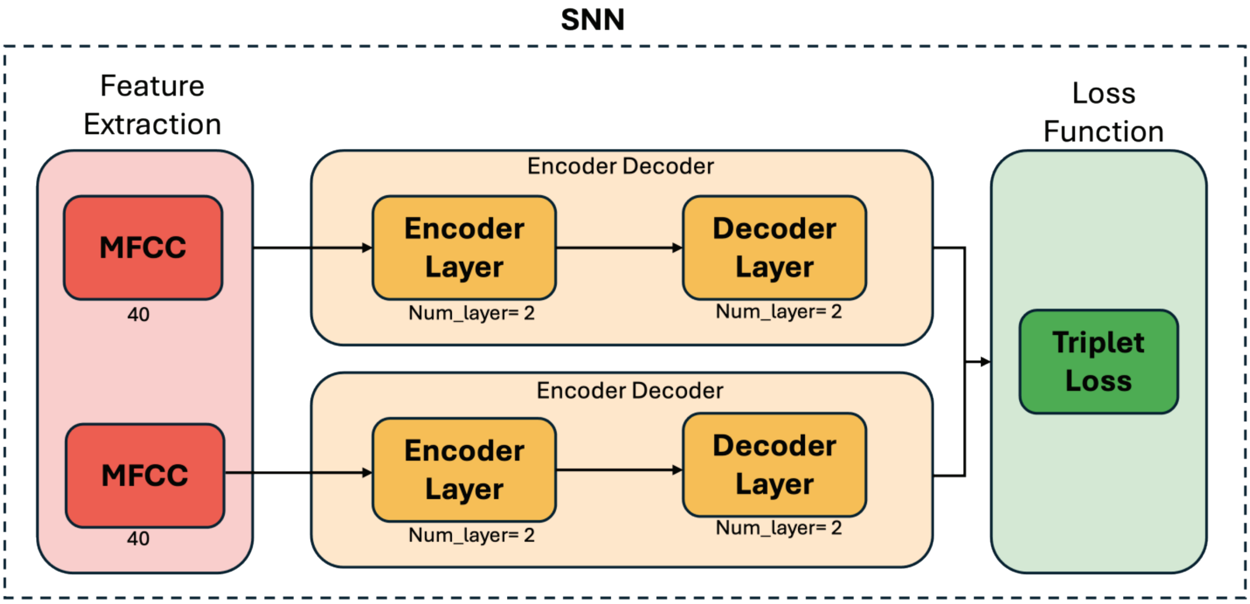

Figure 4. Model 3 – Transformer with tripler loss

Afterward, the information is decoded using a decoding layer. Moreover, an optimized triplet loss function is used to enable the Siamese network to produce feature representations that are invariant to the input data while capturing the similarity between different samples. Lastly, Model 4 (Figure 5) extracts features from the speech signal using the MFCC technique; 40 coefficients are extracted. Then, a transformer is used to extract the encoding of each speaker, 2 sub encoding layers are used in addition to one decoding layer. Finally, a quadruplet loss function is used to capture the variation of the input. All four models have two audio inputs, one of them is the input audio of the person to be verified and the other is the stored audio.

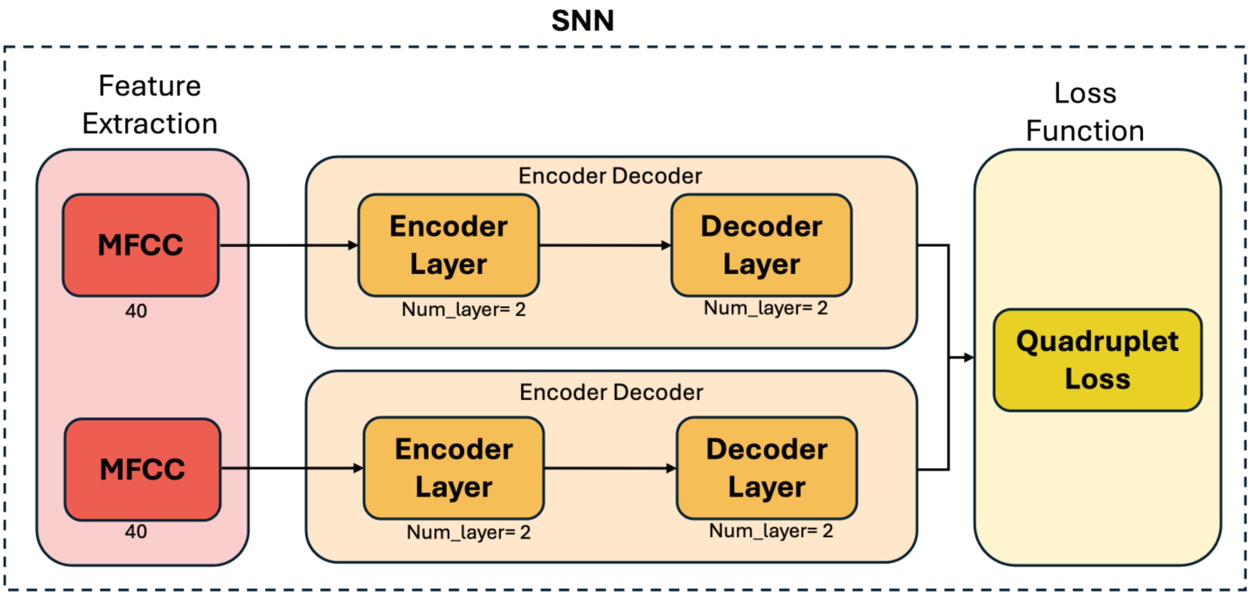

Figure 5. Model 4 - Transformer with quadruplet loss

TABLE II. INFORMATION ABOUT DATASETS USED

Datasets	Speaker Num	Utt. Num.	Average Speaker Utt.	Average Utt. Length
Libri-train	251	28,531	114	10
Libri-test	40	2,620	-	-

TABLE III. INFORMATION ABOUT TRAINING AND INFERENCE TIME OF PROPOSED METHODS

Model	Training Time in hours	Inference Time in sec
Model 1	44.69	0.183
Model 2	32.7	0.1795
Model 3	29.646	0.176
Model 4	25.9	0.1805

B. *Experiments*

This section describes the dataset and its configuration followed by the experimentation setup and the evaluationmetrics.

1. Dataset and configuration

LibriSpeech train-clean-100 dataset (a subset of LibriSpeech corpus) [45] was used for training the proposed models. It consists of 100 hours of clean speech from the LibriVox project. The audio files are provided in 16kHz, 16-bit, and mono WAV formats. The dataset contains speech recordings from 251 different English (male and female) speakers. Those speakers come from a variety of age groups and backgrounds. Each speaker has contributed between 2 and 5 hours of speech, and the speakers are identified by a unique speaker ID and contain approximately 285,000 utterances.

For testing the proposed models, the LibriSpeech test-clean dataset was used, which consists of 40 hours of clean speech from the LibriVox project, it includes speech recordings from 40 different speakers, [45]. Statistics of the training and testing data are shown in II.

TABLE IV. INFORMATION ABOUT THE META_PARAMETERS USED

Model	Learning rate	Num of Heads	Num of Encoder Layer	Batch Size	Num of steps
Model 1	0.001	-	-	8	100000
Model 2	0.001	-	-	8	100000
Model 3	0.0001	8	2	8	100000
Model 4	0.001	8	2	8	100000

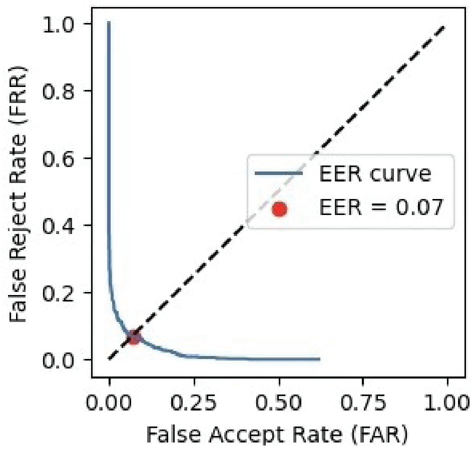

Figure 6. EER for Model 1 - BiLSTM with triplet loss

2. Experimentation setup

The four models were trained and tested on a PC equipped with GeForce GTX 1660 Nvidia graphics card. ADAM optimizer algorithm was used with a learning rate of 0.0001. The meta-parameters used are shown in Table 4. The training and inference times are shown in Table III. Training of the four proposed models is shown in Figures [2,3,4,5].

3. Evaluate metrics

The experimental findings are evaluated using the Equal Error Rate (EER) [48; 57]. The EER combines the False Acceptance Rate (FAR) and the False Rejection Rate (FRR). FRR represents the rate at which genuine instances are incorrectly rejected, while FAR represents the rate at which impostor instances are incorrectly accepted, they can be defined as follows:

$$FRR = \frac{FN}{FN + TN} \tag{3}$$

And

$$FAR = \frac{FP}{FP + TP} \tag{4}$$

Therefore, the EER can be defined as follows:

$$EER = \frac{FAR + FRR}{2} \tag{5}$$

IV. RESULTS

The proposed approach was evaluated by comparing the results with state-of-the-art models, as shown in Table V. These comparisons were made on LibriSpeech datasets as part of control experiments to assess the accuracy of the proposed models.

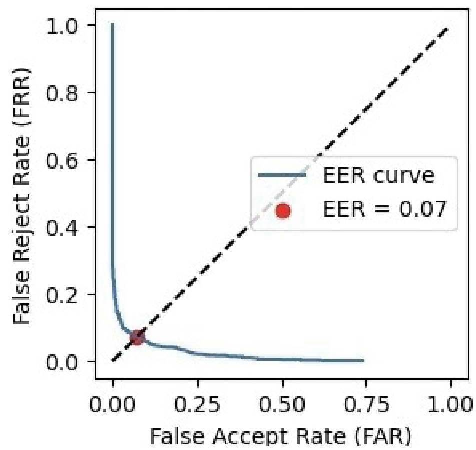

Figure 7. EER for Model 2 – BiLSTM with quadruplet loss

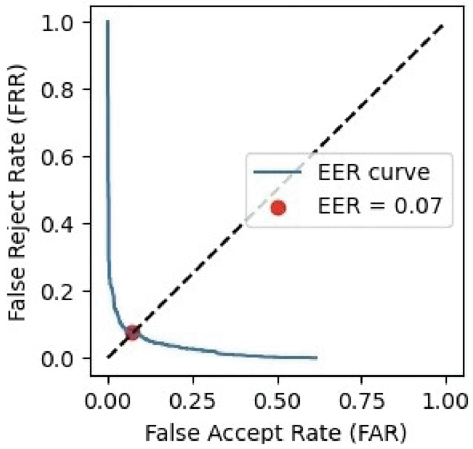

Figure 8. EER for Model 3 - transformer

The model proposed by [40] adapted SNNs with a CNN layer and triplet loss. The model in [44] used MFCCT features and used a deep neural network consisting of 7 layers. Finally, the proposed models have been evaluated over 1000 examples on test data (LibriSpeech test-clean), and the results are summarized as follows:

- **Model 1:** The results show Model 1 has scored an EER of 0.0685 (see Figure 6 for more details), while having an inference time of 0.183 seconds. This model demonstrates a significant improvement in accuracy compared to previous models, making it a promising approach for future research.

- **Model 2:** Model 2 scored an EER of 0.074 (see Figure 7 for more details), while having an inference time of 0.1795 seconds. Although the EER is slightly higher than Model 1, the inference time is marginally better, indicating a trade-off between accuracy and speed.

- **Model 3:** Model 3 scored an EER of 0.073 (see Figure 8 for more details), while having an inference time of 0.176 seconds. This model strikes a balance between accuracy and inference time, making it a viable option for real-time applications.

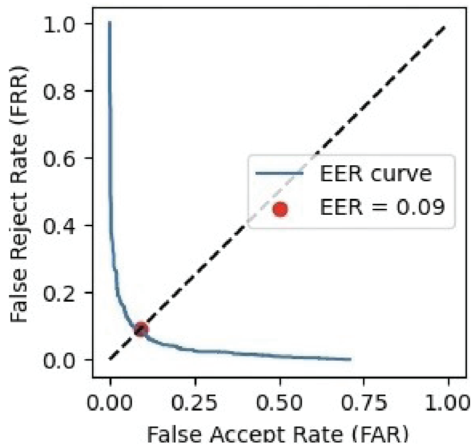

Figure 9. EER for Model 4 – transformer with quadruplet loss

TABLE V. COMPARISON BETWEEN PROPOSED MODELS AND PREVIUOSLY DEVELPED MODELS USING THE SAME DATASET

Model	EER
Model 1 - BiLSTM with Triplet Loss	0.0685
Model 2 - BiLSTM with Quadruplet Loss	0.074
Model 3 - Transformer with Triplet Loss	0.073
Model 4 - Transformer with Quadruplet Loss	0.09
Previous Model 1 [44]	0.11
Previous Model 2 [40]	0.11

- **Model 4:** Model 4 scored an EER of 0.09 (see Figure 9 for more details), while having an inference time of 0.1805 seconds. Despite having the highest EER among the proposed models, it still outperforms several state-of-the-art models in terms of inference time.

The results indicate that the proposed models outperform several state-of-the-art models in terms of both accuracy and inference time. Model 1, in particular, shows the best performance with the lowest EER and competitive inference time. These findings suggest that the proposed approach is effective and can be further optimized for real-world applications.

V. DISCUSSION

Model 1 (BiLSTM with triplet loss) training shows low loss and fast convergence due to the simplicity of triplet loss combined with BiLSTM's sequential processing(see Figure 10 for more information). The model presented stability and effectiveness for sequential speech data, leveraging BiLSTM's ability to capture temporal patterns.

Model 2 (BiLSTM with quadruplet loss) training shows potential for lower loss, but quadruplet loss increases training difficulty, leading to slower results compared to triplet loss (see Figure 11 for more information). The training was stable as well but slightly slower to converge. Quadruplet loss enforces better separation between embeddings but adds complexity.

Model 3 (transformer with triplet loss) performs better than BiLSTM with quadruplet loss due to more powerful global feature learning, resulting in better embedding separation and lower loss despite the simpler triplet loss (see Figure 12 for more information). The training was more complex and sensitive to tuning, but the global attention mechanism captures richer, more complex patterns.

Model 4 (transformer with quadruplet loss) shows higher loss compared to transformer with triplet loss, as the added complexity of quadruplet loss does not always translate into significantly better performance in transformers (see Figure 13 for more information). The training was more challenging due to the combination of complex transformer architecture and quadruplet loss.

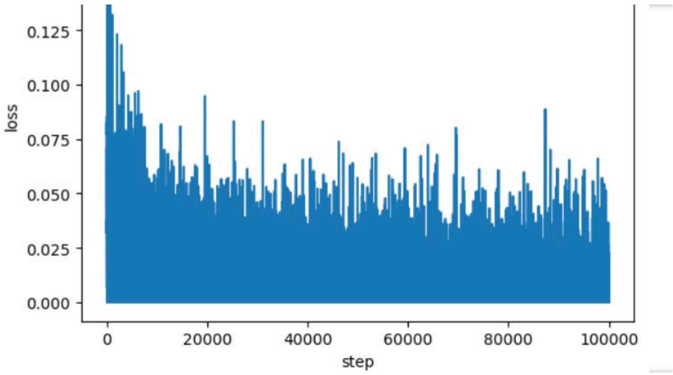

Figure 10. Training for Model 1 - BiLSTM with triplet loss

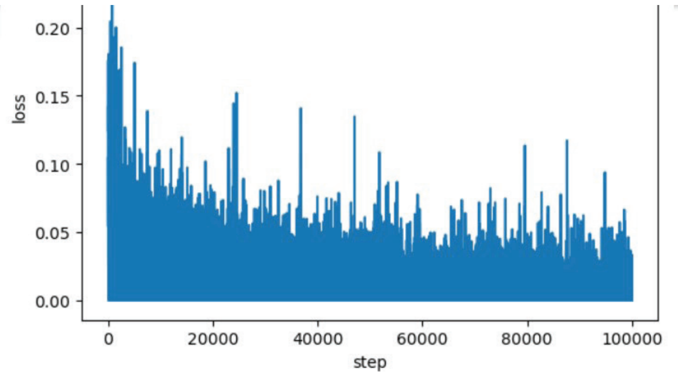

Figure 11. Training for Model 2 – BiLSTM with quadruplet loss

The result for Model 1 is shown in Figure 6. The X-Axis expresses False Acceptance Rate (FAR) which represents the rate of incorrectly accepting impostors as genuine. While Y-Axis expresses False Rejection Rate (FRR) - Which represents the rate of incorrectly rejecting genuine speakers. The results show lowest EER (Error Equal Rate) of 0.0685 due to effective temporal processing and stable training. This model achieves a strong balance between FRR and FAR.

The result for Model 2 is shown in Figure 7. The same plotting as Model 1. The results show a slightly higher (0.74) than Model 1 but still competitive. The added complexity of quadruplet loss improves separation but with slow convergence.

The result for Model 3 is shown in Figure 8. The results show a Moderate EER of 0.073, better than Model 2 (BiLSTM with quadruplet loss). The model benefits from attention mechanisms, capturing more complex patterns effectively. The result for Model 4 is shown in Figure 9. The results show a Moderate EER 0.09, better than Model 2 (BiLSTM with quadruplet loss). The model benefits from attention mechanisms, capturing more complex patterns effectively.

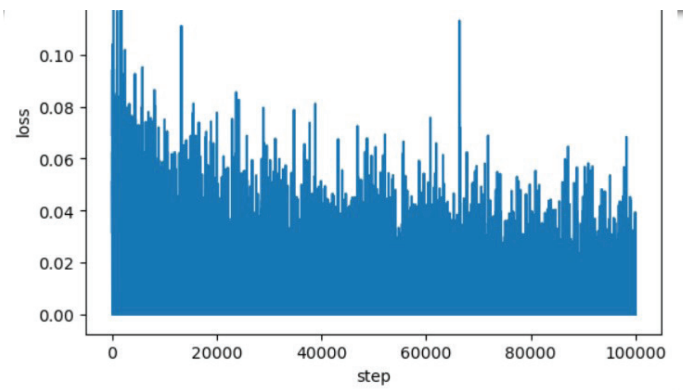

Figure 12. Training for Model 3 - transformer with triplet loss

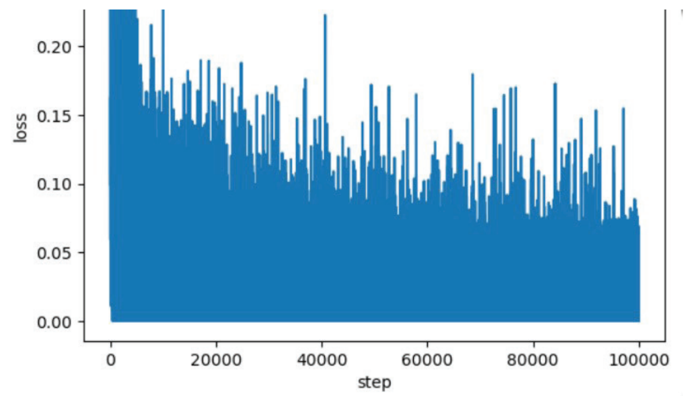

Figure 13. Training for Model 4 - transformer with quadruplet loss

VI. CONCLUSION AND FUTURE WORK

In this study, the researchers proposed and evaluated four models for speaker verification using the LibriSpeech dataset. The proposed models were compared with state-of-the-art models to assess their accuracy and efficiency. The results demonstrated that the proposed models, particularly Model 1 (BiLSTM with triplet loss), achieved significant improvements in terms of Equal Error Rate (EER) and inference time.

Model 1 exhibited the lowest EER of 0.0685 and a competitive inference time of 0.183 seconds, highlighting its effectiveness in capturing temporal patterns and providing stable training. Model 2 (BiLSTM with quadruplet loss) showed potential for lower loss but faced challenges in training complexity and convergence speed. Model 3 (transformer with triplet loss) outperformed Model 2 due to its powerful global feature learning capabilities, resulting in better embedding separation and lower loss. Model 4 (transformer with quadruplet loss) demonstrated higher loss compared to Model 3, indicating that the added complexity of quadruplet loss does not always translate into better performance in transformers.

Overall, the proposed models outperformed several state-of-the-art models in terms of both accuracy and inference time. The findings suggest that the proposed approach is effective and can be further optimized for real-world applications. Future work will focus on refining the models and exploring additional techniques to enhance their performance and applicability in various speech recognition tasks.

For future work, MFCCT and MSE methods could be integrated with the models replacing MFCC to analyze the accuracy versus the inference time of the four models.

REFERENCES

[1] Gaurav, S. Bhardwaj, and R. Agarwal, "Two-Tier Feature Extraction with Metaheuristics-Based Automated Forensic Speaker Verification Model," *Electronics (Switzerland)*, vol. 12, no. 10, 2023, doi: 10.3390/electronics12102342.

[2] O. Shalash and P. Rowe, "Computer-assisted robotic system for autonomous unicompartmental knee arthroplasty," *Alexandria Engineering Journal*, vol. 70, 2023, doi: 10.1016/j.aej.2023.03.005.

[3] Y. Lin *et al.*, "Voxblink: A large scale speaker verification dataset on camera," *in ICASSP 2024-2024 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP), IEEE*, pp. 10271–10275, 2024.

[4] M. Neelima and I. S. Prabha, "Optimized deep network based spoof detection in automatic speaker verification system," *Multimed Tools Appl*, vol. 83, no. 5, 2024, doi: 10.1007/s11042-023-16127-w.

[5] M. Jakubec, R. Jarina, E. Lieskovska, and P. Kasak, "Deep speaker embeddings for Speaker Verification: Review and experimental comparison," *Eng Appl Artif Intell*, vol. 127, 2024, doi: 10.1016/j.engappai.2023.107232.

[6] M. Yasser, O. Shalash, and O. Ismail, "Optimized Decentralized Swarm Communication Algorithms for Efficient Task Allocation and Power Consumption in Swarm Robotics," *Robotics*, vol. 13, no. 5, p. 66, Apr. 2024, doi: 10.3390/robotics13050066.

[7] E. Khatab, A. Onsy, M. Varley, and A. Abouelfarag, "A Lightweight Network for Real-Time Rain Streaks and Rain Accumulation Removal from Single Images Captured by AVs," *Applied Sciences (Switzerland)*, vol. 13, no. 1, 2023, doi: 10.3390/app13010219.

[8] Q. Lin, L. Yang, X. Wang, X. Qin, J. Wang, and M. Li, "TOWARDS LIGHTWEIGHT APPLICATIONS: ASYMMETRIC ENROLL-VERIFY STRUCTURE FOR SPEAKER VERIFICATION," in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 2022. doi: 10.1109/ICASSP43922.2022.9746247.

[9] H. Said *et al.*, "Forearm Intravenous Detection and Localization for Autonomous Vein Injection Using Contrast-Limited Adaptive Histogram Equalization Algorithm," *Applied Sciences*, vol. 14, no. 16, p. 7115, Aug. 2024, doi: 10.3390/app14167115.

[10] A. Mittal and M. Dua, "Automatic speaker verification systems and spoof detection techniques: review and analysis," *Int J Speech Technol*, vol. 25, no. 1, 2022, doi: 10.1007/s10772-021-09876-2.

[11] H. Elsayed, N. S. Tawfik, O. Shalash, and O. Ismail, "Enhancing human emotion classification in human-robot interaction," *in 2024 International Conference on Machine Intelligence and Smart Innovation (ICMISI)*, pp. 1–6, 2024.

[12] M. Elkholy, O. Shalash, M. S. Hamad, and M. S. Saraya, "Empowering the grid: A comprehensive review of artificial intelligence techniques in smart grids," *n 2024 International Telecommunications Conference (ITC-Egypt)*, pp. 531–518, 2024.

[13] A. Abouelfarag, M. A. Elshenawy, and E. A. Khattab, "Accelerating sobel edge detection using compressor cells over FPGAs," in *Smart Technology Applications in Business Environments*, 2017. doi: 10.4018/978-1-5225-2492-2.ch001.

[14] E. Khatab, A. Onsy, and A. Abouelfarag, "Evaluation of 3D Vulnerable Objects' Detection Using a Multi-Sensors System for Autonomous Vehicles," *Sensors*, vol. 22, no. 4, 2022, doi: 10.3390/s22041663.

[15] O. Shalash, "Design and development of autonomous robotic machine for knee arthroplasty," 2018.

[16] T. Zhou, Y. Zhao, and J. Wu, "ResNeXt and Res2Net Structures for Speaker Verification," in *2021 IEEE Spoken Language Technology Workshop, SLT 2021 - Proceedings*, 2021. doi: 10.1109/SLT48900.2021.9383531.

[17] I. M. Gaber, O. Shalash, and M. S. Hamad, "Optimized Inter-Turn Short Circuit Fault Diagnosis for Induction Motors using Neural Networks with LeLeRU," in *IEEE Conference on Power Electronics and Renewable Energy, CPERE 2023*, 2023. doi: 10.1109/CPERE56564.2023.10119618.

[18] M. Anjum and S. Shahab, "Improving Autonomous Vehicle Controls and Quality Using Natural Language Processing-Based Input Recognition Model," *Sustainability (Switzerland)*, vol. 15, no. 7, 2023, doi: 10.3390/su15075749.

[19] E. Manfron, J. P. Teixeira, and R. Minetto, "Speaker recognition in door access control system," *in 3rd Symposium of Applied Science for*, p. 8, 2023.

[20] M. Andrade *et al.*, "A Voice-Assisted Approach for Vehicular Data Querying from Automotive IoT-Based Databases," in *2023 Symposium on Internet of Things, SIoT 2023*, 2023. doi: 10.1109/SIoT60039.2023.10389856.

[21] A. Khaled, O. Shalash, and O. Ismaeil, "Multiple Objects Detection and Localization using Data Fusion," in *2023 2nd International Conference on Automation, Robotics and Computer Engineering (ICARCE)*, IEEE, Dec. 2023, pp. 1–6. doi: 10.1109/ICARCE59252.2024.10492609.

[22] X. Zhou and Y. Zheng, "Research on Personality Traits of In-Vehicle Intelligent Voice Assistants to Enhance Driving Experience," in *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 2023. doi: 10.1007/978-3-031-35678-0_15.

[23] C. Wu, Z. Xu, L. Liu, and T. Yang, "A new driving style recognition method for personalized adaptive cruise control to enhance vehicle personalization," *Journal of Intelligent and Fuzzy Systems*, vol. 46, no. 4, 2024, doi: 10.3233/JIFS-235045.

[24] M. Deng *et al.*, "Using voice recognition to measure trust during interactions with automated vehicles," *Appl Ergon*, vol. 116, 2024, doi: 10.1016/j.apergo.2023.104184.

[25] B. Rudrusamy, H. C. Teoh, J. Y. Pang, T. H. Lee, and S. C. Chai, "IoT-Based Vehicle Monitoring and Driver Assistance System Framework for Safety and Smart Fleet Management," *International Journal of Integrated Engineering*, vol. 15, no. 1, 2023, doi: 10.30880/ijie.2023.15.01.035.

[26] M. J. Roan, M. Beard, L. Neurauter, and M. Miller, "A Data Driven Approach to the Development and Evaluation of Acoustic Electric Vehicle Alerting Systems for Vision Impaired Pedestrians," 2023.

[27] S. Ayas, B. Donmez, and X. Tang, "Drowsiness Mitigation Through Driver State Monitoring Systems: A Scoping Review," *Human Factors: The Journal of the Human Factors and Ergonomics Society*, vol. 66, no. 9, pp. 2218–2243, Sep. 2024, doi: 10.1177/00187208231208523.

[28] W. Liu, Q. Li, Z. Wang, W. Wang, C. Zeng, and B. Cheng, "A Literature Review on Additional Semantic Information Conveyed from Driving Automation Systems to Drivers through Advanced In-Vehicle HMI Just Before, During, and Right After Takeover Request," *Int J Hum Comput Interact*, vol. 39, no. 10, 2023, doi: 10.1080/10447318.2022.2074669.

[29] R. Mohd Hanifa, K. Isa, and S. Mohamad, "A review on speaker recognition: Technology and challenges," *Computers and Electrical Engineering*, vol. 90, 2021, doi: 10.1016/j.compeleceng.2021.107005.

[30] A. M. Sharma, "Speaker Recognition Using Machine Learning Techniques," San Jose State University, San Jose, CA, USA, 2019. doi: 10.31979/etd.fhhr-49pm.

[31] S. Malik and F. A. Afsar, "Wavelet transform based automatic speaker recognition," in *2009 IEEE 13th International Multitopic Conference*, IEEE, Dec. 2009, pp. 1–4. doi: 10.1109/INMIC.2009.5383083.

[32] S. Sujiya and E. Chandra, "A review on speaker recognition," *Int J Eng Technol*, vol. 9, no. 3, pp. 1592–1598, 2017.

[33] Y. A. Ibrahim, J. C. Odiketa, and T. S. Ibiyemi, "Preprocessing Technique in Automatic Speech Recognition For Human Computer Interaction: An Overview," *Anale. Seria Informatică*, vol. 15, 2017.

[34] A. Métwalli, M. H. Sallam, E. Khatab, and O. Shalash, "Polygraph-based truth detection system: Leveraging machine learning model on physiological and behavioral data using data fusion," *Available at SSRN 5031332*.

[35] N. Singh, R. Khan, and R. Shree, "Applications of speaker recognition," *Procedia Eng*, vol. 38, pp. 3122–3126, 2012.

[36] S. A. Imam, P. Bansal, and V. Singh, "Speaker recognition using automated systems," *AGU International Journal of Engineering and Technology (AGUIJET)*, vol. 5, pp. 31–39, 2017.

[37] S. Joshi and M. Dua, "Noise robust automatic speaker verification systems: review and analysis," *Telecommun Syst*, pp. 1–42.

[38] R. Nisa and A. M. Baba, "A speaker identification-verification approach for noise-corrupted and improved speech using fusion features and a convolutional neural network," *International Journal of Information Technology*, pp. 1–9, 2024.

[39] S. Salim, S. Shahnawazuddin, and W. Ahmad, "Combined approach to dysarthric speaker verification using data augmentation and feature fusion," *Speech Commun*, vol. 160, p. 103070, May 2024, doi: 10.1016/j.specom.2024.103070.

[40] M. Balipa and A. Farhath, "Twins Voice Verification and Speaker Identification," in *International Conference on Artificial Intelligence and Data Engineering, AIDE 2022*, 2022. doi: 10.1109/AIDE57180.2022.10060064.

[41] V. Singh and N. Meena, "Engine Fault Diagnosis using DTW, MFCC and FFT," in *Proceedings of the First International Conference on Intelligent Human Computer Interaction*, 2009. doi: 10.1007/978-81-8489-203-1_6.

[42] Z. K. Abdul, "Kurdish speaker identification based on one dimensional convolutional neural network," *Computational Methods for Differential Equations*, vol. 7, no. 4, 2019.

[43] F. K.Faek and A. K. Al-Talabani, "Speaker Recognition from Noisy Spoken Sentences," *Int J Comput Appl*, vol. 70, no. 20, 2013, doi: 10.5120/12182-8213.

[44] R. Jahangir *et al.*, "Text-Independent Speaker Identification through Feature Fusion and Deep Neural Network," *IEEE Access*, vol. 8, 2020, doi: 10.1109/ACCESS.2020.2973541.

[45] V. Panayotov, G. Chen, D. Povey, and S. Khudanpur, "Librispeech: An ASR corpus based on public domain audio books," in *ICASSP, IEEE International Conference on Acoustics, Speech and Signal Processing - Proceedings*, 2015. doi: 10.1109/ICASSP.2015.7178964.

[46] M. Niu, L. He, Z. Fang, B. Zhao, and K. Wang, "Pseudo-Phoneme Label Loss for Text-Independent Speaker Verification," *Applied Sciences (Switzerland)*, vol. 12, no. 15, 2022, doi: 10.3390/app12157463.

[47] A. Nagraniy, J. S. Chungy, and A. Zisserman, "VoxCeleb: A large-scale speaker identification dataset," in *Proceedings of the Annual Conference of the International Speech Communication Association, INTERSPEECH*, 2017. doi: 10.21437/Interspeech.2017-950.

[48] Q. Zheng, Z. Chen, H. Liu, Y. Lu, J. Li, and T. Liu, "MSRANet: Learning discriminative embeddings for speaker verification via channel and spatial attention mechanism in alterable scenarios," *Expert Syst Appl*, vol. 217, 2023, doi: 10.1016/j.eswa.2023.119511.

[49] M. K. Singh, "A text independent speaker identification system using ANN, RNN, and CNN classification technique," *Multimed Tools Appl*, vol. 83, no. 16, pp. 48105–48117, Nov. 2023, doi: 10.1007/s11042-023-17573-2.

[50] D. Chicco, "Siamese Neural Networks: An Overview," in *Methods in Molecular Biology*, vol. 2190, 2021. doi: 10.1007/978-1-0716-0826-5_3.

[51] J. BROMLEY *et al.*, "SIGNATURE VERIFICATION USING A 'SIAMESE' TIME DELAY NEURAL NETWORK," *Intern J Pattern Recognit Artif Intell*, vol. 07, no. 04, 1993, doi: 10.1142/s0218001493000339.

[52] B. Logan, "Mel Frequency Cepstral Coefficients for Music Modeling," *International Symposium on Music Information Retrieval*, vol. 28, 2000, doi: 10.1.1.11.9216.

[53] Z. C. Lipton, J. Berkowitz, and C. Elkan, "A Critical Review of Recurrent Neural Networks for Sequence Learning," *arXiv preprint arXiv:1506.00019*, 2015.

[54] N. Mellor, *The good, the bad and the irritating: A practical approach for parents of children who are attention seeking*. The Good, the Bad and the Irritating, 2000.

[55] A. Hermans, L. Beyer, and B. Leibe, "In defense of the triplet loss for person re-identification," *arXiv preprint arXiv:1703.07737*, 2017.

[56] W. Chen, X. Chen, J. Zhang, and K. Huang, "Beyond triplet loss: A deep quadruplet network for person re-identification," in *Proceedings - 30th IEEE Conference on Computer Vision and Pattern Recognition, CVPR 2017*, 2017. doi: 10.1109/CVPR.2017.145.

[57] R. Issa *et al.*, "A Data-Driven Digital Twin of Electric Vehicle Li-Ion Battery State-of-Charge Estimation Enabled by Driving Behavior Application Programming Interfaces," *Batteries*, vol. 9, no. 10, 2023, doi: 10.3390/batteries9100521.

Reinforcement Learning for Autonomous Underwater Vehicles (AUVs): Navigating Challenges in Dynamic and Energy-Constrained Environments

Mohab M. Eweda [1], and Karim A. ElNaggar [2]

[1] Department of Electrical Engineering Upgrading Studies, Institute of Maritime Upgrading Studies, AASTMT Abukir Campus, Egypt.
[2] Department of Electrical & Control Engineering, College of Engineering and Technology, AASTMT Abukir Campus, Egypt.

mohabeweda@live.com, karimelnaggar726@yahoo.com

Received on, 02 November 2024 Accepted on, 22 December 2024 Published on, 24 December 2024

ABSTRACT

Autonomous Underwater Vehicles (AUVs) are essential for underwater exploration, inspection, and environmental surveillance. Nevertheless, navigation, obstacle avoidance, and energy efficiency are greatly hindered by the ever-changing underwater environments. Reinforcement Learning (RL) has arisen as a revolutionary method for tackling these challenges. This paper examines significant progress in reinforcement learning algorithms, emphasizing their application in the training of autonomous underwater vehicles in both simulated and real-world environments. The review synthesizes findings from multiple studies, identifies gaps in existing research, and highlights the potential of algorithms such as Deep Deterministic Policy Gradient (DDPG) for continuous control tasks. This review offers an extensive examination of current methodologies, their constraints, and avenues for future investigation.

Key-words: Autonomous Underwater Vehicles, AUVs, Reinforcement Learning, RL, Navigation, Obstacle Avoidance, Energy Efficiency

I. INTRODUCTION

Autonomous Underwater Vehicle (AUV) navigation is just one example of the complicated control problems that reinforcement learning (RL) has the potential to solve. Autonomous underwater vehicles (AUVs) play an essential role in many fields, such as oceanography, underwater infrastructure inspection, and search and rescue. However, there are certain obstacles specific to the underwater domain, such as energy limitations, poor communication, and unpredictable currents. When faced with such changing conditions, traditional control methods frequently fall short. In this literature review, the researchers look at how Deep Deterministic Policy Gradient (DDPG) and other RL algorithms have been applied to underwater robotics. The purpose of this review is to show how RL is useful for underwater navigation, point out where more research is needed, and fill in any gaps in the current literature. It is structured thematically and discusses topics such as RL theory foundations, AUV navigation applications, difficulties in dynamic environments, and potential future research directions.

A. Related Work

A major development in reinforcement learning for continuous control tasks, Lillicrap et al. (2015) propose the DDPG algorithm [1]. Their actor-critic framework sets the foundation for its use in robotics by allowing RL agents to effectively operate in high-dimensional action spaces. Emphasizing the value of trial-and-error learning for decision-making procedures [6], Sutton and Barto (1998) offer basic insights into RL. Building on these ideas, Zhang et al. (2021) use DDPG for underwater navigation to show its potential for exact control in virtual environments [7].

Other noteworthy contributions include the robotic arms and aerial drones using Q-learning and policy gradient techniques. For robotic manipulation, Levine et al. (2016), for example, present end-to-end training of visuomotor policies, so demonstrating the capacity of RL to solve challenging tasks [11]. Emphasizing its ability for handling continuous and high-dimensional control spaces, these studies prepared the groundwork for using RL in underwater robotics.

A safe and controlled environment is provided by simulated environments for the purpose of training AUVs with RL. To facilitate obstacle avoidance in AUVs, Smith et al. (2018) implement reward mechanisms [2]. Their research underscores the significance of reward functions that are well-designed for the purpose of facilitating the learning of policies. In the same vein, Garcia and Torres (2019) implement policies that are based on deep learning to improve the trajectory planning of AUVs [10]. Although these methods are successful in structured environments, they encounter difficulty in generalizing to dynamic and unpredictable underwater conditions.

The development of realistic underwater environments has been significantly facilitated by simulation tools like Gazebo and Unity. These platforms enable researchers to integrate physical factors such as turbulence, drag, and buoyancy, thereby rendering the training process more akin to real-world conditions. Nevertheless, the "sim-to-real" problem, which is frequently used to describe the disparity between simulation and reality, continues to be a significant obstacle.

Significant difficulties arise for AUV navigation in dynamic environments due to the presence of stochastic disturbances and unpredictable obstacles. In their study, Chen and Wang (2020) investigate how RL can be adjusted to suit actual oceanic circumstances, considering random environmental perturbations like turbulence and ocean currents [3]. Their research demonstrates the critical importance of stable policies that can withstand changing conditions over the long term. Regardless of these developments, their approaches are computationally heavy and necessitate a large amount of training data, neither of which is necessarily accessible. The use of adaptive reward systems to strengthen policies has been the subject of further research. To achieve a better balance between navigation efficiency and energy conservation, Singh et al. (2021) create a multi-objective RL framework. This framework showed improves adaptability in dynamic conditions [12]. Nevertheless, the question of how to achieve adaptation in real-time is still unanswered.

For prolonged missions, AUVs must emphasize energy efficiency. Kumar et al. (2019) [4] concentrate on energy-efficient reinforcement learning policies for prolonged AUV missions. Their strategy markedly enhances operational efficiency by prioritizing energy consumption within the reward function. Their reliance on oversimplified energy models, however, renders them ineffective in practical applications. Brown et al. (2017) find that heuristic-based reward systems are not as effective as possible in addressing complex navigation tasks [9]. Recent advancements in energy modeling have enabled more accurate predictions of battery efficiency and power consumption. The operational model of AUV is enhanced for energy efficiency via RL-based scheduling [13] following the integration of renewable energy sources, such as solar panels, by Zhang et al. (2022). These innovations demonstrate the potential for reinforcement learning to collaborate with advanced energy management systems.

A comparative analysis of reinforcement learning algorithms employed in autonomous underwater vehicle research uncovers significant trends and constraints. Zhang et al. (2021) emphasize the benefits of DDPG for continuous control, whereas Lee et al. (2020) illustrate the adaptability of Proximal Policy Optimization (PPO) in multi-agent environments [8]. Conversely, heuristic-based approaches, such as those suggested by Brown et al. (2017), are easier to implement but demonstrated insufficient adaptability to varied underwater conditions. Garcia and Torres (2019) underscore the significance of amalgamating deep learning methodologies with reinforcement learning for enhanced policy acquisition [10].

Among AUV navigators, DDPG is a preferred choice mostly because of its capacity to manage continuous action environments. Zhang et al. (2021) use it to maximize paths [7], so stressing its adaptability in simulated environments. Chen and Wang (2020) show its resilience when combined with domain randomization approaches [3] so allowing one to generalize across several underwater conditions. The researchers still have a lot to learn, though, about how to combine DDPG with multi-sensor data fusion, so enhancing their capacity to perceive and make decisions in their surroundings. Apart from navigation, DDPG has proved useful for environmental monitoring and object retrieval—two more AUV chores. Liu et al. (2022) notably improve coverage and energy efficiency by bestocating underwater sensors with DDPG [14]. These applications show how adaptable the method is and how more generally it could be used in underwater robotics. DDPG and other algorithms have shown promise in addressing continuous control problems; hence, this review highlights the growing application of RL in AUV navigation. Two main gaps are poor policies for dynamic environments and insufficient connection with actual oceanic conditions. The aim of this work is to fill in these voids by using RL developments, hence more adaptive and efficient AUV navigation systems are sought for.

II. METHODOLOGY

A. *Problem Formulation*

AUVs have become indispensable instruments in various marine activities, such as environmental assessment, search and rescue operations, and underwater investigation. These applications necessitate AUVs to autonomously navigate in dynamic, complex, and frequently hostile underwater environments. The autonomy of AUVs is impeded by several substantial challenges, including dynamic obstacles (e.g., moving objects and unpredictable ocean currents), energy constraints from onboard batteries, and the necessity for accurate goal-directed navigation in three-dimensional environments.

Due to high latency, limited communication bandwidth, and the difficulty in sensing accurate positional information, these challenges are made even worse in underwater environments. Robust control mechanisms and adaptive strategies are required to guarantee safe and efficient navigation due to these constraints. In order to make AUVs more efficient and dependable in real-world scenarios, it is essential to solve these problems. To teach AUVs to navigate autonomously, optimize energy consumption, and avoid obstacles in these types of settings, this research suggests an RL framework that makes use of the Deep Deterministic Policy Gradient (DDPG) algorithm.

B. *Mathematical Representation*

1. State space

The AUV state at any time (t), denoted as $(s_t \in R^7)$, is a multidimensional vector defined as:

$$[s_t = [x, y, z, \text{roll}, \text{pitch}, \text{yaw}, \text{battery level}],] \tag{1}$$

where $((x, y, z))$ represents the AUV position in a 3D coordinate system. The orientation of the AUV is described by $(roll,\ pitch,\ yaw)$, and the battery level indicates the remaining energy. This state vector encapsulates the critical information required for navigation and decision-making.

Additional environmental data include:

1. Positions of dynamic obstacles, represented as ($\{o_i \in R^3\}_{i=1}^N$) , (N) where is the number of obstacles.

2. Environmental disturbances, such as water currents, modeled as random forces acting on the AUV.

2. Action space

The action $(a_t \in R^6)$ is a continuous control input defined as:

$$[a_t = [f_x, f_y, f_z, \text{roll\_change}, \text{pitch\_change}, \text{yaw\_change}],] \quad (2)$$

where $(f_x f_y f_z)$ represent the thrust forces in three spatial directions, and *(roll_change, pitch_change, yaw_change)* correspond to rotational adjustments in orientation. These actions are constrained to the physical limits of the AUV thrusters and rotational capabilities.

3. Transition function

The dynamics of the AUV, as implemented in the simulation environment, govern the transition from the current state (s_t) to the next state (s_{t+1}). The transitions are described as:

$$x_{t+1} = x_t + \Delta_x + \text{disturbance}_x \quad (3)$$

$$y_{t+1} = y_t + \Delta_y + \text{disturbance}_y \quad (4)$$

$$z_{t+1} = z_t + \Delta_z + \text{disturbance}_z \quad (5)$$

$$\text{roll}_{t+1} = mod(\text{roll}_t + a_{\text{roll}}, 360) \quad (6)$$

$$\text{pitch}_{t+1} = mod(\text{pitch}_t + a_{\text{pitch}}, 360) \quad (7)$$

$$\text{yaw}_{t+1} = mod(\text{yaw}_t + a_{\text{yaw}}, 360) \quad (8)$$

$$battery\ level_{t+1} = \max\left(0, battery\ level_t\ -\ \eta \sum |a\_t|\right) \quad (9)$$

4. Reward function

The reward function is designed to incentivize efficient and goal-oriented navigation while penalizing unsafe or inefficient behavior. It is expressed as:

$$[r(s_t, a_t) = r_1 + r_2 + r_3 - r_4 - r_5 - r_6,] \quad (10)$$

where:

$(r_1 = \alpha \cdot \text{distance\_reduction})$

$(r_2 = \beta \cdot \text{smooth\_action})$

$(r_3 = \gamma \cdot \text{battery\_efficiency})$

$(r_4 = \delta / \text{proximity\_to\_obstacle})$

$(r_5 = \eta \cdot \text{excessive\_action})$

$(r_6 = \lambda \cdot \text{goal\_deviation})$

The parameters $(\alpha = 10, \beta = 5, \gamma = 3, \delta = 50, \eta = 2, \lambda = 8)$ are empirically tuned.

5. Objective

The reward function is designed to incentivize efficient and goal-oriented navigation while penalizing unsafe or inefficient behavior. It is expressed as:

$$J = E\left[\sum_{t=0}^{T} \gamma^t r(s_t, a_t)\right], \quad (11)$$

where $(\gamma \in (0,1))$ is the discount factor, and (T) is the episode length.

The objective of the reinforcement learning problem is to maximize the expected cumulative discounted reward:

$$J = E\left[\sum_{t=0}^{T} \gamma^t r(s_t, a_t)\right], \quad (12)$$

where $(\gamma \in (0,1))$ is the discount factor, and (T) is the episode length.

C. *Proposed Model*

The Deep Deterministic Policy Gradient (DDPG) algorithm is employed as the RL framework. DDPG is a model-free, off-policy algorithm that is well-suited for environments with continuous action spaces, such as AUV control. It combines actor-critic methods, where the actor learns a deterministic policy, and the critic evaluates the policy using a Q-value function.

1. Actor network

The actor network is a neural network that maps the current state (s_t) to a continuous action (a_t) . It comprises:

- **Input:** State vector (s_t) .
- **Hidden layers:** Two fully connected layers with 256 units each and ReLU activation.

- **Output layer:** A tanh activation function to constrain actions within defined bounds.

2. Critic Network

The critic network estimates the Q-value $(Q(s_t, a_t))$, which represents the expected return for a given state-action pair. It comprises:

- **Inputs:** State (s_t) and action (a_t).
- **Hidden layers:** Two fully connected layers with 256 units each and ReLU activation.
- **Output layer:** A linear activation to produce the scalar Q-value.

3. Training process

The training procedure involves episodic interactions between the AUV and the environment. The agent explores the environment using Ornstein-Uhlenbeck noise to encourage diverse actions. Parameter updates are based on the following rules:

$$[\theta_Q \leftarrow \theta_Q + \alpha \nabla_{\theta_Q} E[r + \gamma Q'(s', \mu'(s')) - Q(s,a)],]$$
$$[\theta_\mu \leftarrow \theta_\mu + \beta \nabla_{\theta_\mu} Q(s, \mu(s)).] \quad (13)$$

III. RESULTS AND DISCUSSION

A. Training Performance

The training performance of the AUV is evaluated over multiple episodes, with key metrics such as episode rewards, position, orientation, and battery usage recorded. The following sections analyze the results obtained.

B. Episode Rewards

The episode rewards curve demonstrates the learning progression of the AUV. Initially, rewards fluctuate significantly, indicating exploration of the environment. Over time, the rewards stabilize, suggesting that the agent has learned a policy to navigate effectively while optimizing energy usage and avoiding obstacles.

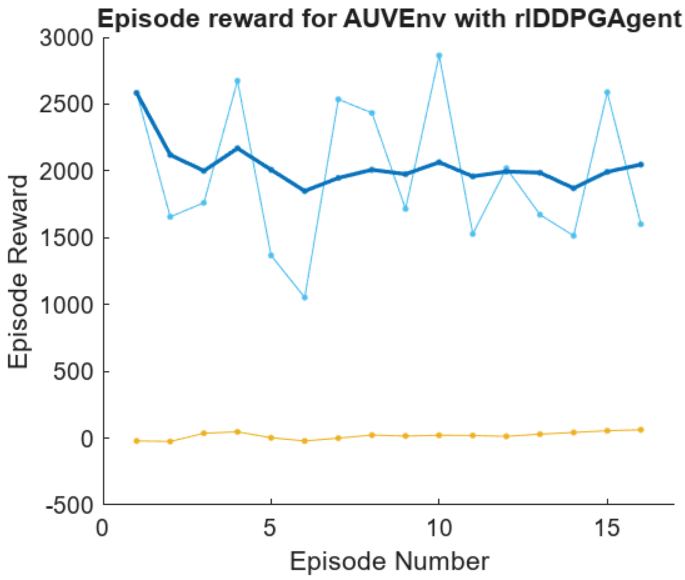

Figure 1. Episode rewards over time

C. Trajectory Analysis

The trajectory of the AUV shows its path in the 3D environment, highlighting its ability to reach the target while avoiding obstacles. The trajectory demonstrates adaptive behavior in navigating through challenging configurations.

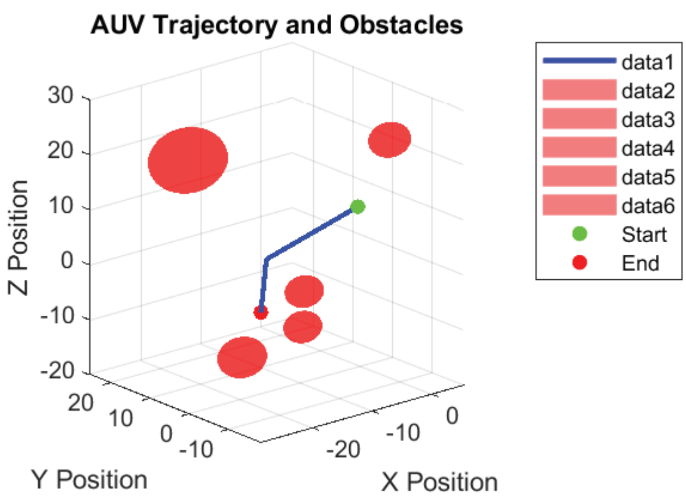

Figure 2. AUV trajectory and obstacles

D. Position and Orientation Analysis

The position and orientation plots provide insights into the control strategy adopted by the AUV. Smooth changes in position indicate efficient navigation, while orientation adjustments show precise control to maintain stability and alignment with the target direction.

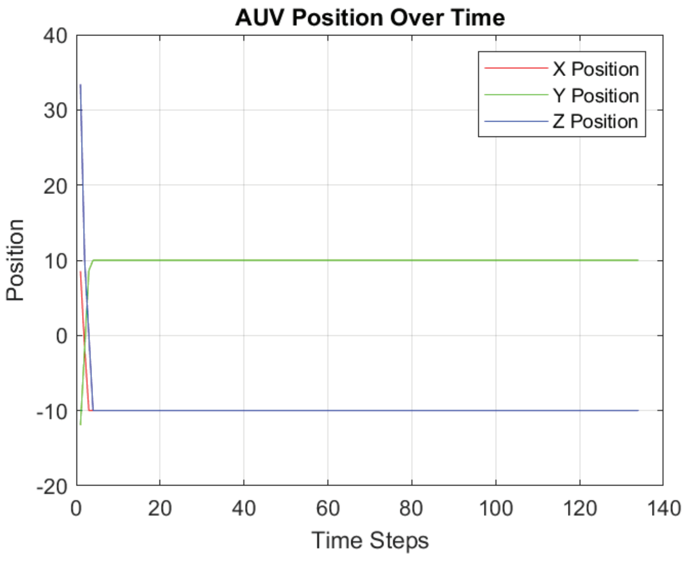

Figure 3. AUV position over time

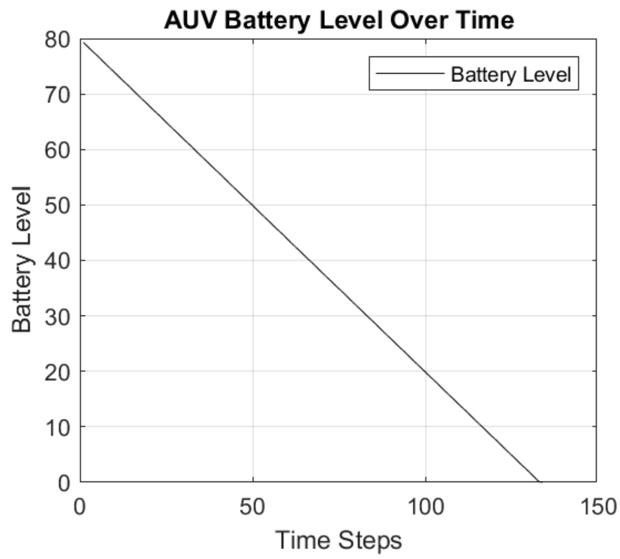

Figure 5. AUV battery level over time

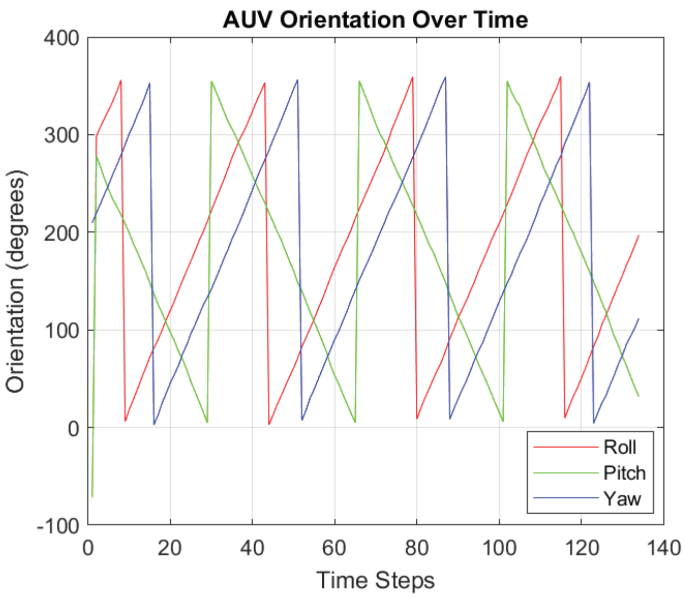

Figure 4. AUV orientation over time

E. *Battery Usage Analysis*

The battery usage plot highlights the energy efficiency of the AUV. The gradual decrease in battery levels indicates optimized energy expenditure, with no sudden drops that would suggest inefficient or excessive actions.

The results demonstrate the effectiveness of the proposed reinforcement learning framework for AUV navigation. The agent successfully learned to balance goal-oriented navigation, obstacle avoidance, and energy optimization. Key strengths include the ability of the agent to adapt to dynamic environments and its efficient use of resources.

However, occasional deviations from optimal behavior, as indicated by spikes in the rewards, suggest areas for further improvement. Future work could explore alternative reward function designs and additional training scenarios to enhance robustness.

IV. CONCLUSION

This research illustrates the efficacy of reinforcement learning (RL), specifically the Deep Deterministic Policy Gradient (DDPG) algorithm, in tackling the obstacles of autonomous navigation for AUVs. The proposed approach effectively allowed the AUV to function in intricate, simulated underwater environments by concentrating on dynamic obstacle avoidance, energy-efficient path planning, and goal-oriented navigation.

The findings underscore numerous significant accomplishments:

The DDPG algorithm demonstrated significant efficacy in continuous control tasks, facilitating smooth, adaptive, and efficient trajectories for the AUV. The incorporation of a multi-objective reward function, which balances navigation, energy efficiency, and safety, markedly enhanced performance across various scenarios. The trained policy exhibited strong obstacle avoidance abilities and energy optimization, significantly decreasing collision rates and efficiently conserving battery usage. Notwithstanding these achievements, numerous limitations persist. The inconsistency of rewards and sporadic divergences from optimal trajectories suggest a necessity for enhanced refinement in the reward framework and training methodology. The sensitivity of the agent to environmental configurations indicates the necessity of integrating a broader range of training scenarios to enhance generalization.

The ramifications of this research transcend simulated contexts. The framework establishes a basis for implementing RL-based navigation systems in practical AUVs, with prospective applications in environmental monitoring, underwater exploration, and search-and-rescue operations. Nonetheless, closing the divide between simulation and reality is a vital focus for future research, necessitating progress in sim-to-real transfer techniques, resilient sensor integration, and adaptive policies proficient in managing real-time environmental disruptions.

Prospective trajectories encompass:

1. Evaluating and confirming the methodology in actual underwater settings to tackle practical issues like hardware limitations and sensor inaccuracies.

2. Expanding the framework to encompass multi-agent systems for collaborative objectives.

3. Implementing advanced energy management strategies, including renewable energy integration, to improve long-duration mission capabilities.

This study highlights the potential of reinforcement learning in enhancing the autonomy and operational efficiency of autonomous underwater vehicles, facilitating the development of more scalable and adaptive underwater robotics solutions. By overcoming current constraints and investigating prospective avenues, reinforcement learning can transform AUV navigation in both academia and the industry.

REFERENCES

[1] T. P. Lillicrap *et al.*, “Continuous control with deep reinforcement learning,” *arXiv Cornell University*, 2015, doi:10.48550/arxiv.1509.02971.

[2] P. Bhopale, F. Kazi, and N. Singh, “Reinforcement Learning Based Obstacle Avoidance for Autonomous Underwater Vehicle,” *Journal of Marine Science and Application*, vol. 18, no. 2, pp. 228–238, Jun. 2019, doi: 10.1007/s11804-019-00089-3.

[3] Y. Jiang, K. Zhang, M. Zhao, and H. Qin, “Adaptive meta-reinforcement learning for AUVs 3D guidance and control under unknown ocean currents,” *Ocean Engineering*, vol. 309, p. 118498, Oct. 2024, doi: 10.1016/j.oceaneng.2024.118498.

[4] J. Wen, A. Wang, J. Zhu, F. Xia, Z. Peng, and W. Zhang, “Adaptive energy-efficient reinforcement learning for AUV 3D motion planning in complex underwater environments,” *Ocean Engineering*, vol. 312, p. 119111, Nov. 2024, doi: 10.1016/j.oceaneng.2024.119111.

[5] R. K. Lea, R. Allen, and S. L. Merry, "A comparative study of control techniques for an underwater flight vehicle," *Int J Syst Sci*, vol. 30, no. 9, pp. 947–964, Jan. 1999, doi: 10.1080/002077299291831.

[6] R. S. Sutton and A. G. Barto, *Reinforcement Learning, Second Edition An Introduction*. 2017.

[7] Y. Sun, X. Ran, G. Zhang, X. Wang, and H. Xu, "AUV path following controlled by modified Deep Deterministic Policy Gradient," *Ocean Engineering*, vol. 210, p. 107360, Aug. 2020, doi: 10.1016/j.oceaneng.2020.107360.

[8] M. Rahmati, M. Nadeem, V. Sadhu, and D. Pompili, "UW-MARL," in *Proceedings of the International Conference on Underwater Networks & Systems*, New York, NY, USA: ACM, Oct. 2019, pp. 1–5. doi: 10.1145/3366486.3366533.

[9] N. Anh Vien, N. Hoang Viet, S. Lee, and T. Chung, "Heuristic Search Based Exploration in Reinforcement Learning," in *Computational and Ambient Intelligence*, Berlin, Heidelberg: Springer Berlin Heidelberg, pp. 110–118. doi: 10.1007/978-3-540-73007-1_14.

[10] B. Hadi, A. Khosravi, and P. Sarhadi, "Deep reinforcement learning for adaptive path planning and control of an autonomous underwater vehicle," *Applied Ocean Research*, vol. 129, p. 103326, Dec. 2022, doi: 10.1016/j.apor.2022.103326.

[11] S. Levine, C. Finn, T. Darrell, and P. Abbeel, "End-to-end training of deep visuomotor policies," 2016.

[12] A. Ramezani Dooraki and D.-J. Lee, "A Multi-Objective Reinforcement Learning Based Controller for Autonomous Navigation in Challenging Environments," *Machines*, vol. 10, no. 7, p. 500, Jun. 2022, doi: 10.3390/machines10070500.

[13] K. Sivamayil, E. Rajasekar, B. Aljafari, S. Nikolovski, S. Vairavasundaram, and I. Vairavasundaram, "A Systematic Study on Reinforcement Learning Based Applications," *Energies (Basel)*, vol. 16, no. 3, p. 1512, Feb. 2023, doi: 10.3390/en16031512.

[14] Z. Wang *et al.*, "Toward Communication Optimization for Future Underwater Networking: A Survey of Reinforcement Learning-Based Approaches," *IEEE Communications Surveys & Tutorials*, pp. 1–1, 2024, doi: 10.1109/COMST.2024.3505850.

مركز الأكاديمية للنشر
ACADEMY Publishing Center