Undersea Diver Communication System Using Speech/Text Conversion

Yue Rong, Peng Chen

School of Electrical Engineering, Computing and Mathematical Sciences
Curtin University

Introduction

Underwater acoustic communication systems have low data rate due to the limited bandwidth of the underwater acoustic channel. This makes real-time speech communication between divers challenging. To overcome the bandwidth challenge, in this paper, we present an undersea diver communication system, which uses speech-to-text conversion at the transmitter to achieve speech compression. The text message is transmitted to the receiver, and the message is converted back to speech at the receiver end.

The average speech rate is around 140 words per minute in normal conversation. In underwater diver scenarios, the speech rate could be even lower. Considering that the average word length is 5.5 characters, and ASCII (8 bit) is used for character encoding, the data rate of the speech text is around $140 \times 5.5 \times 8/60 = 103$ bits per second (bps). In contrast, the data rate of the state-of-the-art low-rate audio codecs such as Lyra and Opus is in the order of several thousand bps [1]. Thus, the data rate after speech-to-text conversion is one order lower than the rate of the most advanced speech codec, making the converted text suitable for transmission through bandwidth limited underwater acoustic channel.

In this paper, we present a prototype diver communication system developed by our team. The core parts of the prototype are real-time offline speech-to-text and text-to-speech converters based on Wav2vec 2.0 [2] and PicoTTS [3], respectively, implemented on a Raspberry Pi board. The prototype has a small form factor, lightweight and is portable by divers. We discuss the hardware and software design at the transmitter end and the receiver end. We demonstrate the performance of this real-time system, validating its efficacy through an underwater acoustic communication experiment conducted recently in the Success Boat Harbour in Fremantle, WA. Initial trial results show promising outcomes of the proposed diver communication system.

System Design

The system design encompasses both the transmitting and receiving processes, with a focus on its real-time capabilities.

System Architecture

The data transmission workflow of the system is illustrated in Figure 1. The diver speech signal is first recorded and processed to filter out the bubble noise, breathing noise and background noise. Then the pre-processed speech signal is passed to a speech recognition engine where the speech is converted to text. The transcription is passed to the transmitter modem as ASCII characters, which are sent through the underwater acoustic channel to the receiver modem.

At the receiver end, the ASCII text is converted to speech signal through a text-to-speech engine. Then, the speech signal is played back.

This research was supported by the Defence Science Centre, an initiative of the State Government of Western Australia.

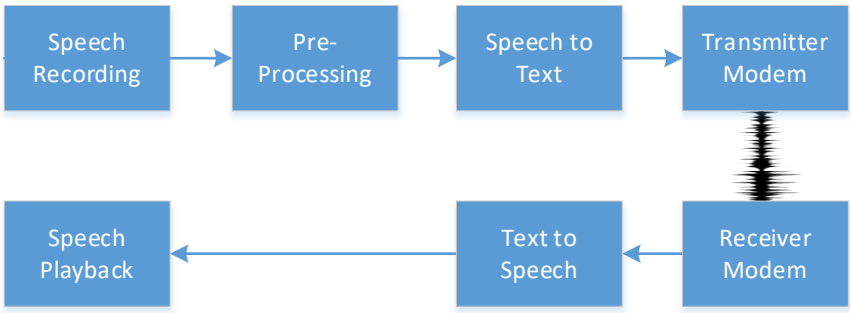

Figure 1: System Architecture.

Speech-to-text Conversion

Wav2vec2.0 is a state-of-the-art speech representation model developed by Facebook AI Research (FAIR). The architecture of the Wav2vec2.0 model is shown in Figure 2. The key innovation of Wav2vec2.0 lies in its self-supervised learning (SSL) approach. Instead of relying on labelled training data where the model is trained on pairs of audio and corresponding transcriptions, Wav2vec2.0 leverages a large amount of unlabelled data. It does so by training the model to predict the context of masked portions of the audio waveform. This pre-training phase helps the model learn meaningful representations directly from raw audio signals. The self-supervised pre-training is followed by fine-tuning on a smaller labelled dataset for the specific speech recognition task. The model is adjusted to the specific characteristics of the labelled data, allowing it to perform well on the target speech recognition task.

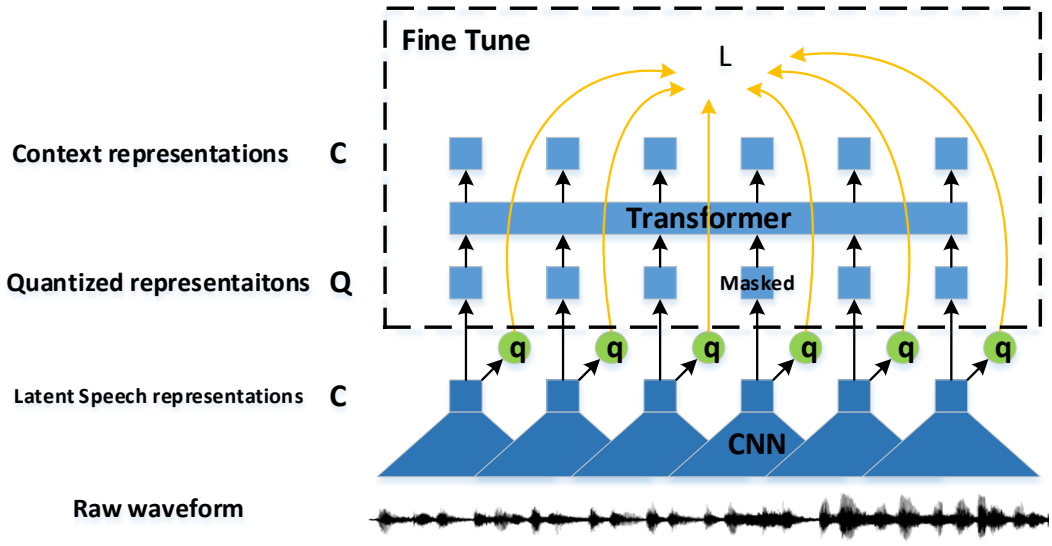

Figure 2: Wav2vec2.0 model.

Software Flowchart

The flowchart of the system software is illustrated in Figure 3. The system starts with two parallel processes, one for receiving signals while the other for transmitting signals.

Note that the system has included the replay function, which is useful in case the incoming speech is not fully comprehended by the diver due to reasons such as lack of diver's attention or high background noise level. Alternatively, we can include warning sounds such as a beep to alert the diver of the incoming speech. After each receiving process, the target file for replay is updated.

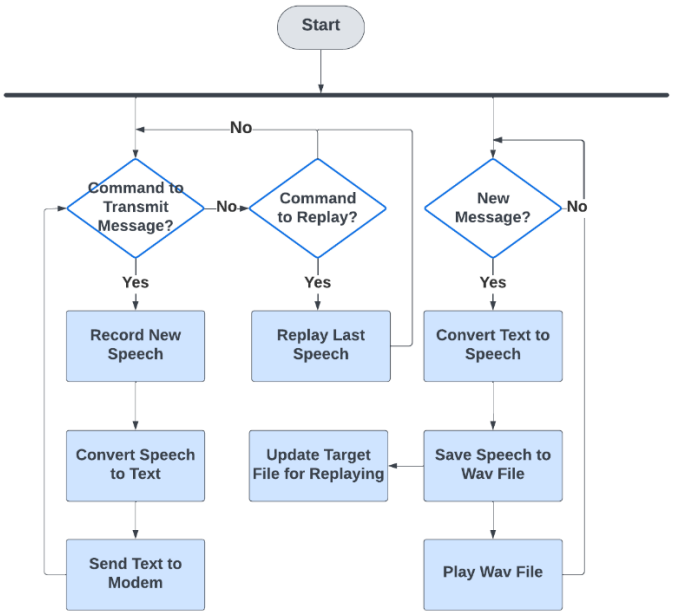

Figure 3: System software flowchart.

Prototype Development

Hardware Configuration

The hardware configuration of the prototype system is shown in Figure 4. The microphone and speaker from a diver mask are connected to a Raspberry Pi board, together with a push-to-talk button. This button can release two commands with short press for starting new speech and long press for playing back the last received message.

The Raspberry Pi board shown in Figure 5 can be inserted into a watertight pressure housing designed and made by our team. Two amplifiers are applied to enhance the speech signal, one for earphones and the other for the microphone.

A pair of Delphis acoustic modems manufactured by Succorfish Ltd [4] are used to transmit and receive signals underwater. One modem is connected to the Raspberry Pi board, while the other modem is interfaced with a laptop computer through a RS 232 cable as shown in Figure 4.

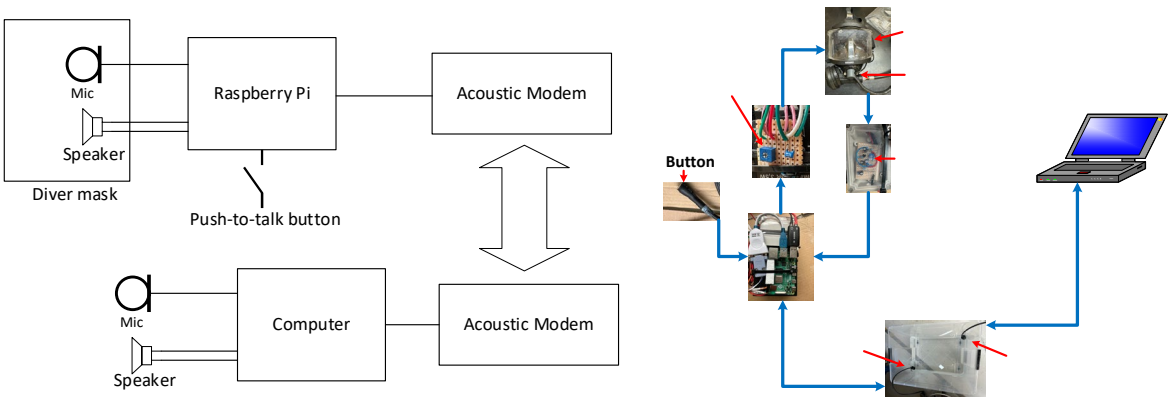

Figure 4: Hardware configuration of the prototype system.

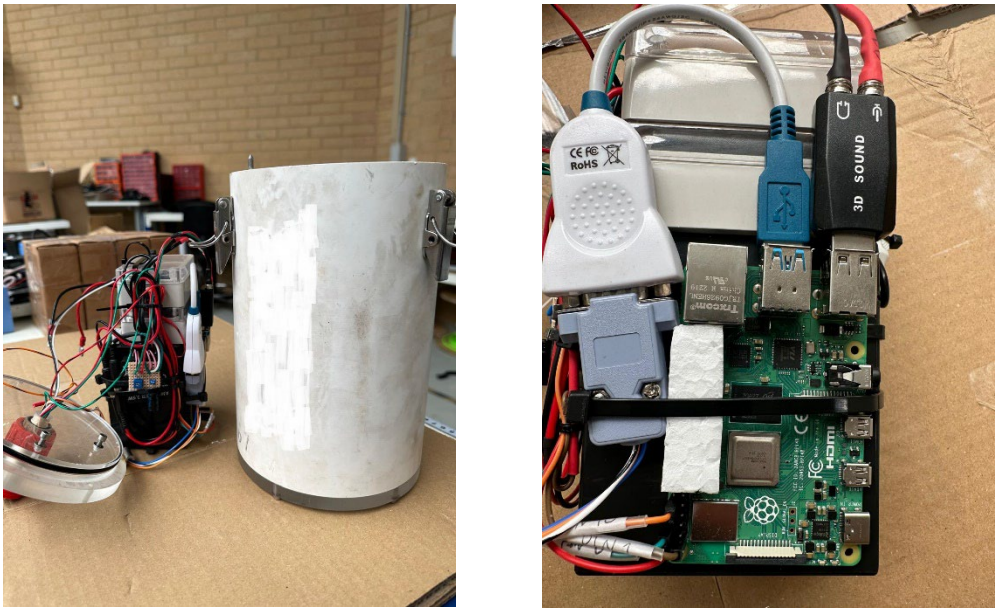

Figure 5: Raspberry Pi board and canister.

Processor Board

Raspberry Pi is a series of small, affordable, single-board computers. They are designed to be cost-effective, making them accessible by a wide range of users, including students, hobbyists, and professionals. Raspberry Pi 5 is selected as the developing board of the prototype. Its specifications are as follows:

CPU	Broadcom BCM2712 2.4GHz quad-core 64-bit Arm Cortex-A76
Memory	8 GB
GPU	800 MHz VideoCore VII
USB	2x USB 3.0 + 2x USB 2.0
Input power	Type-C (5.1V 5A)
Weight	32 g
Storage	micorSD card, can support NVMe SSDs

Acoustic Modem

The Delphis acoustic modems are low-cost, low-power, and have a small form factor, making them idea for portable diver applications. The specifications are listed below:

Frequency band	24-32 kHz
Source level	168 dB re 1μPa @ 1m
Raw data rate	640 bits/s
Maximal range	2 km sea water; 3.5 km fresh water
Depth rating	350 m
Supply voltage	5V
Current	2.5 mA listening; 5 mA receiving; maximal 300 mA transmitting

Software Development

The system software is developed using Python. At the transmitter side, the Wav2vec2.0 speech-to-text conversion is adopted to transcribe speech to text. The resulting text message is transmitted under the binary terminal mode of the Delphis modem. The Python pyserial library is used to communicate between the Raspberry Pi and the Delphis underwater acoustic communication modem.

The receiver modem also works under the binary terminal mode and applies the Pico text-to-speech (TTS) engine to convert text to speech. Compared with other tools such as the Google text-to-speech (gTTS), PicoTTS provides a better speech quality. In addition, it can also be trained to generate personalised speech.

Trial Results

An initial trial of the system was carried out in the Success Boat Harbour in Fremantle, Western Australia. The location of the trial is shown in Figure 6. During this trial, a diver talked through a microphone located in the AGA mask. The speech signal was converted to text by the Raspberry Pi board located in a watertight canister carried by the diver (see Figure 6), and the text signal was transmitted by one Delphis modem to another Delphis modem. The received signal was converted to speech and played back to the speaker on a laptop computer located on a jetty. This setup simulates the scenario of communication between a diver and the topside.

Figure 6: The location and setup of the harbour test of the prototype system

In this underwater communication experiment, a total of 41 speech files were recorded, with 35 of them successfully received by the receiver. The word error rate (WER) of each speech is shown in Figure 7. With the original Wav2ve2.0 –960 hours model, the overall WER for these recordings is calculated to be 41.6%.

Additionally, the recorded speech underwent off-line evaluation using an improved Wav2vec 2.0 model, which was fine-tuned through a set of 100 sentences from the same diver. The WER for each speech using the fine-tuned Wav2vec2.0 model is shown in Figure 7. It can be seen that through fine-tuning, the average WER is reduced to 16%.

During the trial, the laptop computer also sent speech to the diver. Noises generated by wind and boat activities were present during the trial. The experimental results indicate the successful functioning of the prototype system in a real undersea environment.

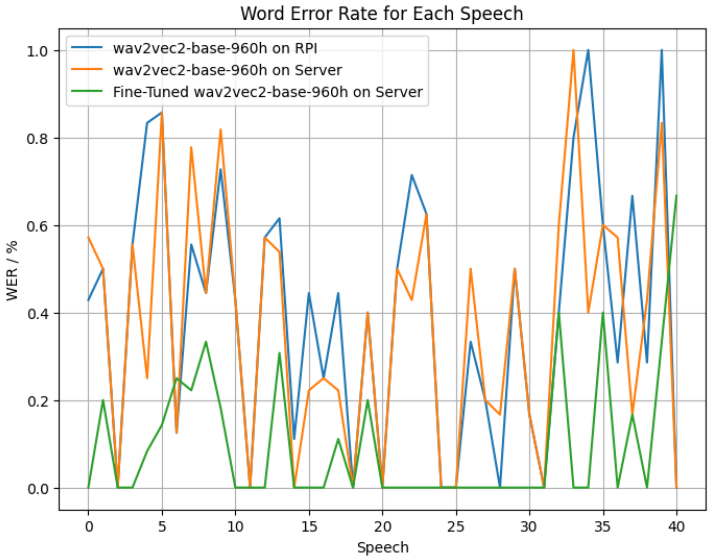

Figure 7: WER of each speech.

Conclusions

We have presented a speech-to-text conversion-based diver undersea communication system. Initial trial results show promising outcomes of the proposed diver communication system. Future works include improving the reliability of the speech-to-text conversion algorithm and the integration of the system to diver equipment.

References

[1] N. Zeghidour, A. Luebs, A. Omran, J. Skoglund and M. Tagliasacchi, "SoundStream: An End-to-End Neural Audio Codec," *https://arxiv.org/abs/2107.03312,* 2021.

[2] A. Baevski, H. Zhou, A. Mohamed and M. Auli, "Wav2vec 2.0: A Framework for Self-Supervised Learning of Speech Representations," *https://arxiv.org/abs/2006.11477,* 2020.

[3] "Pico Text-to-Speech," [Online]. Available: https://github.com/ihuguet/picotts.

[4] "Succorfish Delphis," [Online]. Available: https://succorfish.com/products/delphis/.