IEEE Access
Multidisciplinary ⋮ Rapid Review ⋮ Open Access Journal

Received January 10, 2022, accepted January 26, 2022, date of publication February 1, 2022, date of current version February 17, 2022.

Digital Object Identifier 10.1109/ACCESS.2022.3148410

Frame Boundary Detection and Deep Learning-Based Doppler Shift Estimation for FBMC/OQAM Communication System in Underwater Acoustic Channels

PUSHPA KOTIPALLI, ADI SURENDRA MOHANRAJU M., AND PRAVEENA VARDHANAPU
Department of Electronics and Communications Engineering, Shri Vishnu Engineering College for Women (A), West Godavari District, Bhimavaram, Andhra Pradesh 534202, India

Corresponding author: Pushpa Kotipalli (pushpak@svecw.edu.in)

This work was supported by the Naval Research Board, Defence Research and Development Organization, India, under Grant NRB/4003/PG/427.

ABSTRACT Underwater acoustic (UWA) communication is the only option for underwater long-distance wireless communication. However, its data rate is very low, in tens of kbps, and the time-varying nature of the UWA channel exhibits a Doppler effect. The data rate can be increased using a multicarrier technique, such as Filter Bank Multicarrier (FBMC). The FBMC-OQAM system does not require a cyclic prefix (CP) to eliminate intersymbol interference (ISI). Therefore, compared to other multicarrier techniques, the FBMC-OQAM system achieves the maximum spectral efficiency. Despite the well-designed filter, synchronisation errors cause performance degradation in the FBMC-OQAM system. In this paper, we propose frame boundary and Doppler frequency estimation methods for the FBMC-OQAM system under UWA channels. In this study, we propose a preamble-based frame boundary. The conjugate symmetry property is used in the frame boundary estimation. The performance of the proposed technique was studied under different UWA channels with different Doppler scale. The proposed time synchronisation technique yielded multiple peaks under UWA channels. A method for identifying the peak that represents the frame boundary is proposed. We compared the performance of proposed timing metric with existing three reference timing metrics. The proposed timing metric exhibited superior over two metrics and almost identical performance with the Liming's metric with less computational complexity. In addition, we propose a convolutional neural network (CNN)-based Doppler scale estimation method. The proposed technique was shown to estimate a wide range of Doppler scale factors. The performance of the proposed technique was studied by considering different UWA channels at different SNRs. The proposed CNN architecture is analysed by changing the CNN parameters. Stacked autoencoders were adopted for Doppler scale estimation and the performance of the proposed CNN-based Doppler scale estimation method was compared with the stacked autoencoders based Doppler scale estimation method. Simulation results show that the proposed CNN-based Doppler Scale estimator exhibits good performance in UWA channels at different SNRs and Doppler scale factors.

INDEX TERMS Filter bank multicarrier (FBMC), orthogonal quadrature amplitude modulation (OQAM), underwater acoustic communication (UWA communication), convolutional neural network (CNN), frame synchronization, Doppler frequency, deep learning.

I. INTRODUCTION

Coastal waters extending from the shoreline up to 200 m depth beyond which the seafloor slope increases is commonly referred to as "Shallow Water" Shallow water covers 8% of oceans and seas. Marine habitats are mostly present in the shallow waters. The underwater sound speed changes with temperature, salinity, and pressure (depth). In the shallow underwater acoustic (UWA) channel, the sound speed becomes independent of depth in the winter because

The associate editor coordinating the review of this manuscript and approving it for publication was Filbert Juwono.

temperature, wind, and waves in shallow waters largely affect the temperature and salinity and make them independent of depth. The shallow UWA channel environment is complex and exhibits multipath propagation, as shown in Fig. 1. The propagation of acoustic signals strongly affected the absorption loss and scattering loss. The acoustic signal is mainly dominated by spreading loss in short ranges, whereas it depends on both absorption and spreading losses over long ranges [1].

The speed of sound changes with the depth of the UWA channel, but it remains constant over the shallow water depth. The impulse response of a shallow UWA channel depends on the reflection and refraction properties of the channel, as well as the geometry of the channel. Fading occurs due to reflections from the seafloor, sea surface, and underwater obstacles. Transmitted acoustic waves experience scattering when they encounter small objects, such as water bubbles and fish shoaling [2]. Therefore, as a consequence of multipath propagation, multiple copies of acoustic signals reach the receiver at different delays with different path losses and cause signal smearing due to inter-symbol interference (ISI). The channel dispersion length changes with the geometry of the channel, and the velocity of the underwater acoustic signal is low (∼1500 m/s). This nature of the shallow water environment turns the UWA channel into doubly selective channels, that is, the transmitted signal while passing through the UWA channel, undergoes dispersion in both time and frequency domains [3].

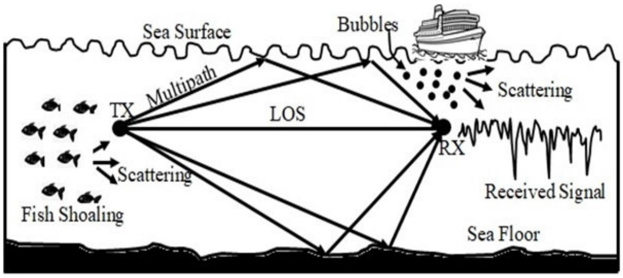


FIGURE 1. Underwater acoustic environment.

UWA channel characteristics include ambient noise, frequency-dependent attenuation, limited acoustic bandwidth, and a non-negligible Doppler effect. Similar to free-space channels, Doppler frequency is generated in the UWA channel owing to the relative motion of the underwater transmitting and receiving vehicles [4]. Apart from vehicular movement, the UWA channel experiences the Doppler effect owing to two more factors, such as drifting (unintentional motion of transmitter, receiver, or both) and surface wave motion [4].

A. RELATED WORKS

In this study, we consider a shallow water environment in the frame boundary and Doppler shift estimation. Underwater long-distance wireless communication is possible only with acoustic waves. As acoustic waves are at low frequencies, the available bandwidth is low. To improve the data rate, multicarrier techniques, such as ''Orthogonal Frequency Division Multiplexing (OFDM)'' and ''Filter Bank Multicarrier (FBMC)'', are employed. To achieve high spectral efficiency, the offset QAM (OQAM) technique was adopted in the FBMC system. The OQAM-based FBMC (FBMC-OQAM) system uses well-localized band-limited pulse shaping filters [5]–[7] which overlap in the time domain and require the satisfaction of the orthogonal condition of the filter in the real domain [8]. Therefore, in FBMC-OQAM, in-phase and quadrature-phase modulation symbols are transmitted separately with a half-symbol duration offset between them [9], [10]. The real orthogonal condition requirement in FBMC-OQAM leads to intrinsic imaginary interference in the dispersive channels [11]. The OQAM technique solves the intrinsic interference problem through a phase offset between adjacent real symbols. The FBMC-OQAM system does not require a cyclic prefix (CP) to eliminate intersymbol interference (ISI). Therefore, compared to other multicarrier techniques, the FBMC-OQAM system achieves the maximum spectral efficiency [12]. The FBMC-OQAM system generates a very low out-of-band (OOB) energy level compared to other filter bank models [12]. FBMC-OQAM was proposed for applications such as cognitive radio and uplink of multiuser multicarrier systems [9], and for future mobile communications, an alternative waveform to OFDM in 5G scenarios such as device-to-device (D2D)-enabled machine-type communications [13]–[15]. In addition, it was considered as a potential candidate for physical layer data communication in future CR networks [9], [16].

The impact of time and frequency synchronisation errors on the performance of filter bank multicarrier transmission was investigated in [17]–[24]. Therefore, accurate time and frequency synchronisation techniques must be designed to handle synchronisation errors. Davide Mattera [25] analysed the performance of an FBMC-PAM system in the presence of a CFO. By comparing the SIRs of OFDM and FBMC-OQAM systems, they proved that the FBMC-OQAM system outperforms the OFDM system when the same number of subcarriers is considered for both multicarrier systems.

Synchronisation techniques can be broadly divided into two categories. These are blind synchronisation techniques and data-aided synchronisation techniques. Mattera and Tanda proposed a blind synchronisation technique for symbol boundary estimation in [26] and blind symbol timing and CFO estimation in [27]. Both techniques exploit the approximate conjugate symmetry property of the FBMC-OQAM signal and require multiple symbols. The second-order cyclostationarity property of the OFDM-OQAM signal was used for blind CFO estimation [28]. As blind synchronisation methods converge slowly, they are not suitable for practical implementation. Therefore, data-aided synchronisation techniques are adopted.

Data-aided synchronisation techniques are further classified into correlation-based and conjugate-symmetry-based techniques. Several correlation-based synchronisation techniques have been investigated. Fusco [29] considered a training sequence made up of identical blocks, and Zeng [30] designed a preamble with periodic properties for the proposed joint time and frequency synchronisation method. As the training sequence consists of several identical parts, a strong correlation between adjacent blocks still exists in the presence of a time offset (TO). Therefore, in the presence of TO, these techniques did not yield sharp peaks at the symbol boundary.

Seo *et al.* [31] proposed time synchronisation based on discretely inserting synchronisation symbols into FBMC data symbols. This method provides interference cancellation, but its complexity is high. A pilot-based fine synchronisation method was proposed in [32]. This is useful only after a coarse initial synchronisation is accomplished. Chung [33], [34] designed a constant-amplitude zero-autocorrelation (CAZAC) sequence-based preamble for timing synchronisation. It has a distortion-compensating property and achieves higher synchronisation accuracy. In [35], the authors provided an overview of the existing time and frequency synchronisation algorithms for the FBMC-OQAM system. Stitz *et al.* [36] proposed carrier frequency offset (CFO) and fractional time offset (FTO) estimation methods using scattered pilots. FTO is estimated using the phase shift between two successive subcarriers of the same symbol, whereas CFO is estimated using the phase shift between two subcarriers at the same locations in two adjacent OFDM symbols [35].

Cho and Ma [37] proposed a joint time and frequency synchronisation technique. They modified the conjugate-symmetric (CS) property by considering null subcarriers and suggested a method for proper peak detection for time offset estimation. In addition, they proposed a ratio-based (non-correlation method) CFO estimator. In the frequency domain frame detection method [38], the authors proposed an autocorrelation metric for repeated pilots of the FBMC-OQAM symbol. The training symbol is generated by loading data onto even-numbered subcarriers only, and weighted correlation is applied to the resultant identical sub-blocks for joint estimation of time and frequency offsets [39]. A long training sequence consisting of nine FBMC symbols which are generated using a modified Zadoff–Chu sequence is used for joint time and frequency synchronisation [40].

The conjugate symmetry property has been applied for timing synchronisation, that is, the real data are transmitted on subcarriers of the preamble symbol in [41]. Li *et al.* [42] generated a conjugate symmetric sequence using a pilot symbol and two auxiliary data symbols and proposed a symbol timing algorithm based on the conjugate symmetry property. In [43]–[45], the authors proposed timing synchronisation techniques by using specially designed synchronisation sequences with a phase-weighted conjugate symmetric structure. Repeated conjugate-symmetric property in two consecutive FBMC-OQAM symbols has been modified by considering null subcarriers and an algorithm for proper peak detection for joint estimation of TO and CFO [37].

Doppler frequency and Doppler rate are estimated for the FBMC-OQAM system using the correlation between identical parts of a specially designed preamble [46]. The Doppler scaling factor and channel parameters were estimated using high-resolution harmonic estimation methods [47].

UWA signal processing techniques have been introduced in [48], [49]. Zhou *et al.* considered zero-padded OFDM to minimise the transmission power and proposed a null subcarrier-based Doppler compensation technique [50], who proposed non-uniform Doppler compensation via resampling and uniform compensation on the residual Doppler [51]. The authors [52] used Preamble with HFM signals to estimate the frame boundary of a signal in a UWA channel and proved that it outperforms the existing LFM-based approach with high frame boundary detection ability. A brief overview of signal processing methods related to a single carrier and OFDM is presented in [49]. A two-step procedure for decision-feedback equalisation was proposed for filtered multitone modulation (FMT), and its performance was analysed [53]. Amini *et al.* [54], [55] showed that FBMC outperforms OFDM for underwater acoustic communications.

The cross-section of the cyclic spectral correlation function of the received signal was used for the CFO estimation [56]. FrFT is applied to linear frequency modulated (LFM) signals in the preamble to determine the Doppler shift [57]. FrFT is applied to the LFM signal and hyperbolic frequency modulation (HFM) signal for the determination of joint time and frequency synchronisation [58]. A time and frequency synchronisation algorithm based on FrFT on the preamble with a symmetrical triangular linear frequency modulation (STLFM) signal is proposed for OFDM systems in UWA channels [59], [60]. Instead of a cyclic prefix (CP), zero padding (ZP) is used with OFDM and proposed for channel equalisation for the UWA channel [61]. The preamble with a hyperbolic frequency modulation (HFM) signal is used for timing position and Doppler factor estimation [62]. By applying a fractional Fourier transform to the preamble containing a chirp signal, the authors [63] proposed that the time and frequency synchronisation for the signal is the UWA channel. A method of matched filter banks along with weighted summation and adjacent peaks algorithms is used for Doppler scale estimation of OFDM signals in UWA channels [64]. The authors [65] studied the maximum likelihood approach [66] based Doppler scaling coefficient of the OFDM signal in the UWA channel. The accuracy of the algorithm depends on the proper selection of the training sequence length and window length. The training symbol is generated by using two chirp symbols, in which the first symbol consists of an up-chirp and the second symbol consists of a down-chirp. At the receiver, the time-difference-of-arrival (TDoA) between two received chirp symbols is measured, and the Doppler shift is determined from this time offset [67].

Neural networks, which are trained using the Levenberg-Marquardt rule, are used in Doppler frequency estimation and show that they are faster than backpropagation methods [68]. With powerful feature extraction capabilities, convolutional neural networks (CNNs) have become unique and superior models in applications related to computer vision, such as autonomous driving and human–computer interaction. In recent years, CNNs have been applied in signal processing. Yann LeCun and Léon Bottou described the CNN in detail [69]. Deep learning (DL) has been used for channel estimation and equalisation in FBMC systems [70]. Ninkovic [71] presented a performance and complexity analysis of packet detection and CFO estimation using both conventional and DL-based approaches. Kojima [72] developed spectrogram images by concatenating PSD values over multiple OFDM symbols and applied spectrogram images as input to the CNN for joint classification of SNR and Doppler shift.

B. PAPER CONTRIBUTIONS

Our main goal is to estimate the frame boundary and Doppler frequency for the FBMC-OQAM system for communication through UWA channels with a low minimum error probability. Deep learning-based techniques have a proven track record of extracting hidden features from the input image for image classification and are more appropriate for attaining accurate Doppler frequency estimation results. Therefore, we considered CNN for Doppler feature extraction. We considered a specially designed preamble [73] suitable for the FBMC-OQAM system and developed images for input to the CNN. Our major contributions in this study are as follows:

1) To present a suitable method to estimate the frame boundary in the UWA environment, we considered the periodic conjugate-symmetry present in the preamble. A procedure for selecting an appropriate peak that falls on the frame boundary is also suggested.
2) To present Doppler frequency estimation, we considered a CNN-based approach, and images were developed using the periodic conjugate symmetry property in the preamble.
3) The superior performance of the proposed methods is presented through simulation results.

The remainder of this paper is organised as follows. Section II introduces the FBMC-OQAM system model. Section III presents the frame boundary estimation and DL-Doppler shift estimation methods. Section IV describes the proposed CNN architecture. Section V presents the simulation results of the performance of the proposed methods for the FBMC system in the presence of the UWA channel, and Section VI presents the conclusions.

II. SYSTEM MODEL OF FBMC

In FBMC, initially randomly generated bits are sent to symbol mapper i.e. OQAM symbol mapper to map random bits to OQAM symbols. The OQAM symbol is generated by providing an offset of half of the symbol duration (T) to the QAM symbol so that the real and imaginary parts are transmitted separately with an offset of T/2. These OQAM symbols are parallel to the serial converter, and then modulation is applied using IFFT which acts as a multicarrier modulator. The IFFT produces time-domain samples, and these samples are given to the synthesis filter bank. A block diagram of the FBMC-OQAM system is shown in Fig. 2. The synthesis filter bank used in this block was a PHYDYAS prototype filter. The prototype filter is designed using certain frequency domain coefficients, and the shifted versions of this prototype filter form a filter bank to remove ICI and maintain orthogonality between sub-channels.

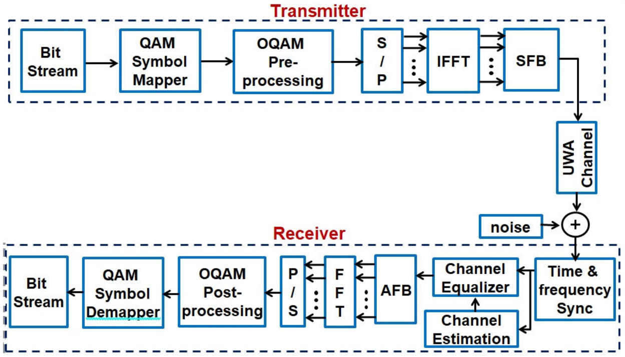


FIGURE 2. Block diagram of FBMC-OQAM system.

The output of the synthesis filter bank was passed through an underwater acoustic channel. At the channel, some noise is added owing to the Doppler spread and multipath effects. This signal is received by an MMSE equaliser which is used to compensate for the channel effects. The equalized signal is passed through an analysis filter bank i.e. PHYDYAS prototype filter bank which detects the original samples. These samples are passed through a serial-to-parallel converter and then passed through FFT to recover the data symbols because FFT acts as a multicarrier demodulator. These recovered data symbols are passed through the OQAM de-mapper to obtain the original bits.

Baseband equivalent FBMC signal is given by

$$s(t) = \sum_{m=0}^{\infty} \sum_{k=o}^{N-1} d_{(m,k)} g_{(m,k)}(t) \tag{1}$$

where N is the number of subcarriers, $d_{(m,k)}$ is the real-valued offset QAM (OQAM) symbol transmitted over the k^{th} subcarrier, and the m^{th} time slot, $g_{(m,k)}(t)$ represents the time-frequency translated version of the symmetrical real-valued prototype filter $g(t)$ which is given by

$$g_{(m,k)}(t) = g\left(t - m\frac{T_0}{2}\right) \exp\{2\pi k F_0 t\} e^{j\theta_{(m,k)}} \tag{2}$$

Here, $T_0/2$ is the time interval between real-valued symbols, F_0 is the subcarrier spacing, and $T_0 = NT_s$, where T_s is the sampling interval. Also, $F_0 = 1/T_0$. The additional phase shift connected with the OQAM method is represented by $\theta_{(m,k)}$, and it depends on the time and frequency slots as

$$\theta_{(m,k)} = \frac{\pi}{2}(m+k) \tag{3}$$

We considered the PHYDYAS prototype filter with an overlapping factor *of* $K = 4$. The PHYDYAS prototype filter is defined as:

$$g(t) = \begin{cases} \frac{1}{\sqrt{A}}\left(1 + 2\sum_{k=1}^{K-1}(-1)^k P_k \cos\left(\frac{2\pi kt}{KT_0}\right)\right); \\ \quad t \in [0, KT_0] \\ 0; \quad \text{elsewhere} \end{cases} \tag{4}$$

where $\mathrm{P}_0 = 1$, $\mathrm{P}_1 = 0.97196$, $\mathrm{P}_2 = 0.707$, $P_3 = \sqrt{1 - P_1^2}$, and $A = KT_0\left(1 + 2\sum_{k=1}^{K-1} P_k^2\right)$. The peak of the PHYDYAS prototype filter occurs at $t = KT_0/2$, whereas for the filter with overlapping factor $K = 4$, the peak occurs at $t = 2T_0$.

We considered the UWA channel with channel impulse response

$$h(t) = \sum_{l=0}^{L-1} h_l e^{j2\pi f_d^l t}\delta(t - \tau_l) \tag{5}$$

where h_l, τ_l, and f_d^l represent the complex amplitude, path delay, and Doppler frequency along the l^{th} channel path, $l = 0, 1, 2, \ldots, L - 1$, and L is the total number of channel paths in the UWA channel.

Now the FBMC signal is passed through the UWA channel. The channel output becomes

$$y(t) = \sum_{l=0}^{L-1} h_l e^{j2\pi f_d^l t} \sum_{m=0}^{\infty}\sum_{k=o}^{N-1} d_{(m,k)} g\left[t - \tau_l - \frac{mT_0}{2}\right] \times e^{j2\pi kF_0(t-\tau_l)}\, e^{j\theta_{(m,k)}} \tag{6}$$

We assume equal Doppler frequency shift along all channel paths. That is

$$f_d^l = f_d; \quad l = 0, 1, \ldots, L - 1 \tag{7}$$

By substituting (7) and $F_k = kF_0$ in (6) and adding additive white Gaussian noise $w(t)$, the received signal $y(t)$ can be expressed as

$$y(t) = \sum_{m=0}^{\infty}\sum_{k=o}^{N-1} d_{(m,k)} e^{j\theta_{(m,k)}} \sum_{l=0}^{L-1} h_l e^{-j2\pi F_k \tau_l} g\left[t - \tau_l - \frac{mT_0}{2}\right] \times e^{j2\pi(F_k + f_d)t} + w(t) \tag{8}$$

From this channel distorted and Doppler-affected noisy FBMC signal, we have to determine the frame boundary and Doppler shift.

III. PROPOSED METHOD

We propose a preamble-based frame boundary estimation method and a deep learning-based carrier frequency offset. The preamble data are known to the receiver. It is used for time and frequency synchronisation, and channel estimation. The carrier frequency offset is caused by (i) the mismatch between the transmitted carrier frequency and local oscillator frequency and (ii) the Doppler frequency shift due to the mobility of either the transmitter, receiver, or both. In this study, we consider the Doppler frequency shift estimation.

The frame structure of the proposed FBMC system is shown in Fig. 3. The frame consists of a preamble, followed by a data payload with FBMC symbols. The first sample index, $n = 0$, represents the frame boundary. The length of the FBMC symbol is represented by L_p.

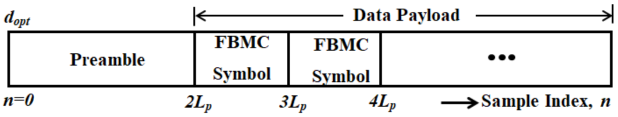


FIGURE 3. Frame structure.

A. PREAMBLE GENERATION

We considered the preamble sequence structure proposed by Ma [73]. By using a Zadoff-Chu (ZC) sequence, an FBMC signal with an overlapping factor $K = 4$ and signal length $L_p = 4N$, where N represents the number of subcarriers, is generated. Then, the FBMC signal is copied four times and all five copies of FBMC signals are superimposed after shifting each successive FBMC signal by N samples, as shown in Fig. 4 [73], to generate a preamble sequence $x[n]$.

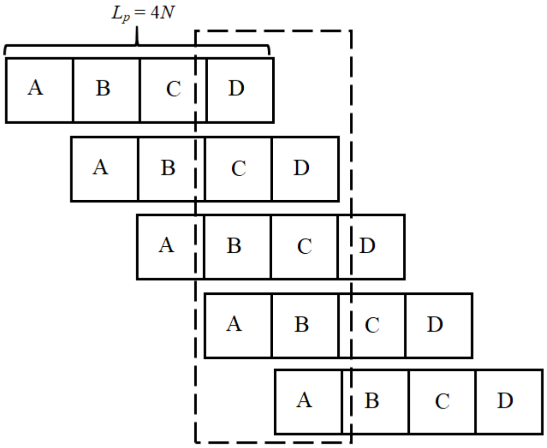


FIGURE 4. Superimposition of five copies of ZC sequence based FBMC signals to generate the Preamble sequence [73].

The FBMC symbol was generated using the simulation parameters listed in Table 1. FBMC symbol length is $L_p = KN = 4N = 512$. Five copies of the FBMC symbol were superimposed, as shown in Fig. 2. Mathematically, we can represent this as follows: Let the four equal parts of the FBMC symbols be A, B, C, and D, and be expressed as

$$s[n] = \{A, B, C, D\} \tag{9}$$

A, B, C, and D are expressed as

$$A = s[n]; \quad n = 0, 1, \ldots, N - 1 \tag{10}$$
$$B = s[n + N]; \quad n = 0, 1, \ldots, N - 1 \tag{11}$$
$$C = s[n + 2N]; \quad n = 0, 1, \ldots, N - 1 \tag{12}$$
$$D = s[n + 3N]; \quad n = 0, 1, \ldots, N - 1 \tag{13}$$

TABLE 1. Simulation parameters.

Parameter	Value
No. of subcarriers, N	128
No. of FBMC symbols, M	10
Modulation method for Data payload	4 - OQAM
Modulation method for the preamble	ZC-sequence
Prototype Filter	PHYDYAS filter
Overlapping factor, K	4
Filter Length, $L_p = KN$	512
Carrier Frequency, f_c	12 KHz
Sampling Frequency, $f_s = 3f_c$	36 KHz
Preamble Length, 2 L_p	1024
No. of subcarriers, N	128
Modulation method for the preamble	ZC-sequence
Prototype Filter	PHYDYAS filter
Overlapping factor, K	4
Filter Length, $L_p = KN$	512
Carrier Frequency, f_c	10 KHz
Sampling Frequency, $f_s = 3\,f_c$	30 KHz
Preamble Length, 2 L_p	1024

The preamble is now expressed as

$$x[n] = \left\{ \begin{matrix} A, A+B, A+B+C, A+B+C+D, \\ A+B+C+D, B+C+D, C+D, D \end{matrix} \right\} \tag{14}$$

As shown in Fig. 4, the length of the preamble sequence $x[n]$ is $2L_p = 8N$ and it consists of two identical parts. The identical parts of the preamble sequence are mathematically expressed as

$$x_1[n] = x[n+3N]; \quad n = 0, 1, \ldots, N-1 \tag{15}$$
$$x_2[n] = x[n+4N]; \quad n = 0, 1, \ldots, N-1 \tag{16}$$
$$x_1[n] = x_2[n]; \quad n = 0, 1, \ldots, N-1 \tag{17}$$

A plot of the amplitude of the preamble sequence is shown in Fig. 5.

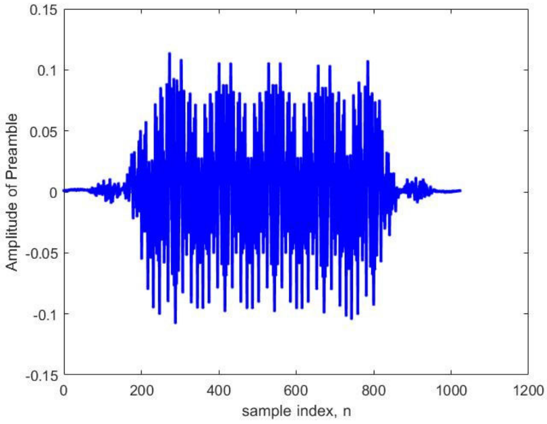


FIGURE 5. Proposed preamble in time domain.

B. PROPOSED FRAME BOUNDARY ESTIMATION METHOD

The two identical parts $x_1[n]$ and $x_2[n]$ exhibit conjugate symmetry properties, as shown below.

Let us divide each identical part of the preamble sequence into two equal-length sequences as

$$x_{11}[n] = x_1[n]; \quad n = 0, 1, \ldots, \frac{N}{2} - 1 \tag{18}$$
$$x_{12}[n] = x_1\left[n + \frac{N}{2}\right]; \quad n = 0, 1, \ldots, \frac{N}{2} - 1 \tag{19}$$
$$x_{21}[n] = x_2[n]; \quad n = 0, 1, \ldots, \frac{N}{2} - 1 \tag{20}$$
$$x_{22}[n] = x_2\left[n + \frac{N}{2}\right]; \quad n = 0, 1, \ldots, \frac{N}{2} - 1 \tag{21}$$

The above-mentioned four sequences exhibit conjugate symmetry property. That is

$$x_{11}\,[n] = x_{11}^*\left[\frac{N}{2} - n\right]; \quad n = 0, 1, \ldots, \frac{N}{4} - 1 \tag{22}$$
$$x_{12}\,[n] = x_{12}^*\left[\frac{N}{2} - n\right]; \quad n = 0, 1, \ldots, \frac{N}{4} - 1 \tag{23}$$
$$x_{21}\,[n] = x_{21}^*\left[\frac{N}{2} - n\right]; \quad n = 0, 1, \ldots, \frac{N}{4} - 1 \tag{24}$$
$$x_{22}\,[n] = x_{22}^*\left[\frac{N}{2} - n\right]; \quad n = 0, 1, \ldots, \frac{N}{4} - 1 \tag{25}$$

With the help of equations (15) to (21), we can express equations (22) to (25) as

$$x\,[3N+n] = x^*\left[\frac{7N}{2} - n\right]; \quad n = 0, 1, \ldots, \frac{N}{4} - 1 \tag{26}$$
$$x\left[\frac{7N}{2} + n\right] = x^*\,[4N-n]; \quad n = 0, 1, \ldots, \frac{N}{4} - 1 \tag{27}$$
$$x\,[4N+n] = x^*\left[\frac{9N}{2} - n\right]; \quad n = 0, 1, \ldots, \frac{N}{4} - 1 \tag{28}$$
$$x\left[\frac{9N}{2} + n\right] = x^*\,[5N-n]; \quad n = 0, 1, \ldots, \frac{N}{4} - 1 \tag{29}$$

We considered the conjugate symmetry property stated in (26) for the timing metric, and we propose the timing metric for frame boundary estimation as

$$P_1[d] = \sum_{m=0}^{N/4-1} x\,[3N+m+d]\,x\left[\frac{7N}{2} - m + d\right]; \quad d = 0, 1, \ldots \tag{30}$$

The normalized timing metric is given by

$$M_1[d] = \frac{P_1[d]}{E_1[d]} \tag{31}$$

where $E_1[d]$ represents the average power of the sequence and it is expressed as

$$E_1[d] = \left| \sum_{m=0}^{N/4-1} x\,[3N+m+d] \right|^2 \tag{32}$$

$M_1[d]$ yields multiple peaks at regular intervals of *N/2*.

To understand the performance of the proposed timing metric, we generated a frame consisting of a preamble followed by a data payload with 10 FBMC symbols using the simulation parameters listed in Table 1. The length of the FBMC symbol is 512, and that of the preamble is 1024. Then, we applied the proposed timing metric (31) to the frame, and the resultant magnitude plot of the proposed metric is shown in Fig. 6. In the preamble part of Fig. 3, we obtain several significant peaks with amplitude 1, and the data part also consists of significant peaks at regular intervals of 448 samples and each peak with adjacent peaks at 97 and 64 samples from it.

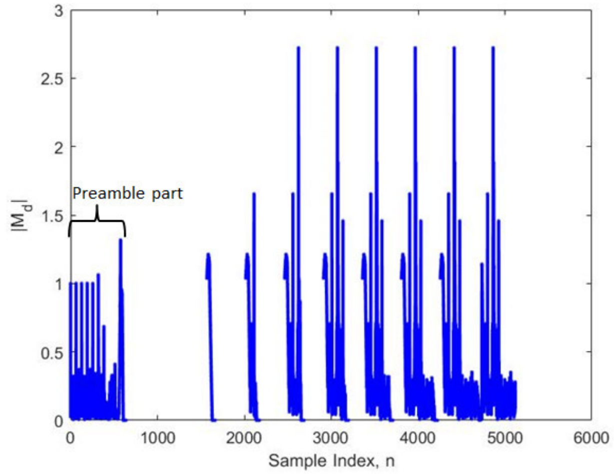


FIGURE 6. Timing metric on the transmitted frame.

To understand the peaks in the preamble part in detail, we considered the enlarged version of the preamble part of Fig. 3, as shown in Fig.7. This plot consisted of seven significant peaks at regular intervals of N/2 = 64. The first six peaks have equal amplitudes and are equal to 1, whereas the 7th peak amplitude is approximately 0.6. The first peak appeared at n = 0 which represents the frame boundary. Therefore, the proposed timing metric indicates the time index of the first peak as the frame boundary.

C. DOPPLER SCALE ESTIMATION

Shuo Ma [73] proposed a fractional Doppler frequency shift algorithm by considering the correlation between two identical parts of the preamble sequence.

$$\begin{aligned} Q[d_{opt}] &= \sum_{m=0}^{N-1} x_1^*\left[m+d_{opt}\right] x_2\left[m+d_{opt}\right] \\ &= \sum_{m=0}^{N-1} x^*\left[3N+m+d_{opt}\right] x\left[4N+m+d_{opt}\right] \end{aligned} \tag{33}$$

Now the Doppler shift factor is determined as

$$\widehat{\varepsilon} = \frac{3\left|Q[d_{opt}]\right.}{2\pi} \tag{34}$$

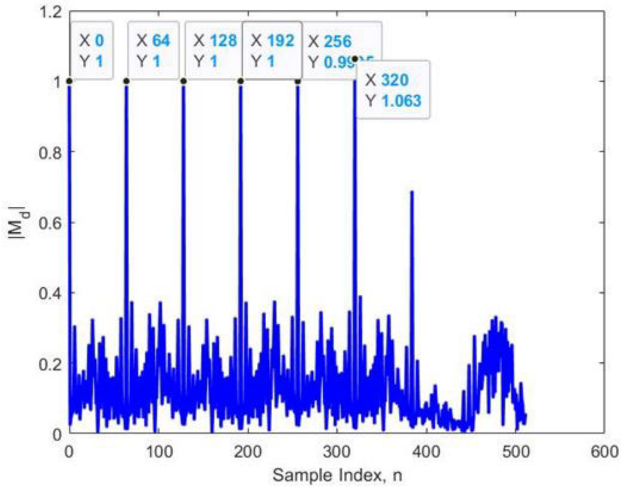


FIGURE 7. Timing metric on the preamble.

D. CRLB FOR ESTIMATION OF DOPPLER SCALE

The Cramer-Rao Lower Bound (CRLB) sets a lower bound on the variance of any unbiased estimator [74]. Assuming perfect timing synchronization, we have estimated the Doppler scale using (33) and (34).

As presented in (15) and (16), $x_1[n]$ and $x_2[n]$ represent two identical parts of the preamble $x[n]$ and are used in Doppler scale estimation. Let us consider AWGN channel. Then the received preamble is given by

$$r[n] = e^{\frac{i2\pi\varepsilon n}{K_1}} x[n] + \upsilon[n]$$

where ε is the normalized frequency offset given by

$$\varepsilon = \frac{\Delta f}{f_c}$$

where Δf is the frequency offset and f_c is the carrier frequency, and $\upsilon[n]$ is the nth time domain AWGN saimple added to the FBMC preamble with mean zero and variance σ^2. Assuming perfect frame boundary at $d_{opt} = 0$ and following (33), the correlation between $(n+3N)^{\text{th}}$ sample and $(n+4N)^{\text{th}}$ sample of the received preamble sequence r[n] is

$$\begin{aligned} z[n] &= r^*[n+3N].r[n+4N] \\ &= x^*[n+3N].x[n+4N]e^{\frac{i2\pi\varepsilon N}{K_1}} \\ &\quad + x^*[n+3N].\upsilon[n+4N]e^{\frac{-i2\pi\varepsilon(n+3N)}{K_1}} \\ &\quad + x[n+4N].\upsilon^*[n+3N]e^{\frac{i2\pi\varepsilon(n+4N)}{K_1}} \\ &\quad + \upsilon^*[n+3N].\upsilon[n+4N] \end{aligned}$$

which can be written as

$$z[n] = x^*[n+3N].x[n+4N]e^{\frac{i2\pi\varepsilon N}{K_1}} + \overline{\upsilon[n]}$$

Now, following the procedure given in [75], we derived the CRLB for the variance of frequency offset estimation and it is given as

$$\left(var\left(\hat{\varepsilon}\right)\right)_{CRLB} = \frac{k_1^2}{(2\pi)^2 N^3 (SNR)} \tag{35}$$

For $N = 128, k_1 = 3$

$$\left(var\left(\hat{\varepsilon}\right)\right)_{CRLB} = \frac{1.087 \times 10^{-7}}{SNR} \tag{36}$$

This algorithm works excellently in the AWGN channel and yields erroneous results in the presence of the UWA channel. Therefore, we propose a deep learning (DL)-based Doppler shift estimation algorithm.

E. PROPOSED DL-BASED DOPPLER SHIFT ESTIMATION METHOD

As discussed in the timing metric and Doppler shift estimation, the preamble exhibits both correlation and conjugate properties. These properties are considered to be features of deep learning.

From Conjugate Symmetry properties, we consider four features as shown below.

$$y_1[n] = x\left[3N+n\right]x\left[\frac{7N}{2}-n\right]; \quad n = 0, 1, \ldots, \frac{N}{4}-1 \tag{37}$$

$$y_2[n] = x\left[\frac{7N}{2}+n\right]x\left[4N-n\right]; \quad n = 0, 1, \ldots, \frac{N}{4}-1 \tag{38}$$

$$y_3[n] = x\left[4N+n\right]x\left[\frac{9N}{2}-n\right]; \quad n = 0, 1, \ldots, \frac{N}{4}-1 \tag{39}$$

$$y_4[n] = x\left[\frac{9N}{2}+n\right]x\left[5N-n\right]; \quad n = 0, 1, \ldots, \frac{N}{4}-1 \tag{40}$$

We can form the grid using these four conjugate symmetry-related features as

$$Y = \left\{y_1^T[n], y_2^T[n], y_3^T[n], y_4^T[n]\right\} \tag{41}$$

where $(\bullet)^{\mathrm{T}}$ represents the transpose of the sequence $(\bullet)$.

From the correlation property, we consider four features of length $(N/4)$ equal to the feature length mentioned above. The correlation sequence is written as

$$z[n] = x_1^*[n]x_2[n] = x^*[3N+n]x[4N+n]; \quad n = 0, 1, \ldots, N-1 \tag{42}$$

This sequence is divided into four parts with equal lengths to represent the four correlation features.

$$z_k[n] = z\left[n+\frac{kN}{4}\right]; \quad n = 0, 1, \ldots, \frac{N}{4}-1; \quad k = 0, 1, 2, 3 \tag{43}$$

The correlation feature grid is given by

$$Z = \left\{z_1^T[n], z_2^T[n], z_3^T[n], z_4^T[n]\right\} \tag{44}$$

We considered combining conjugate symmetry and correlation feature grids for Doppler shift estimation using deep learning.

$$F_1 = \{Y, Z\} \tag{45}$$

The features of the preamble are shown in Fig. 8. The amplitude of the conjugate symmetry-related features is shown in Fig. 8(a) to Fig. 8(d), and the amplitudes of the correlation-related features are shown in Fig. 8(e) Fig. 8(h). A convolutional neural network (CNN) was used in deep learning.

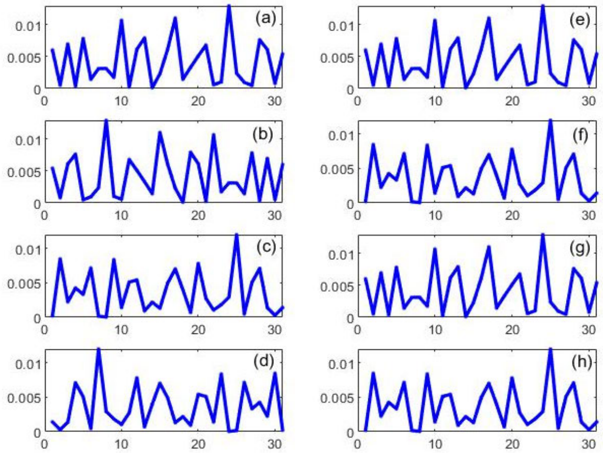


FIGURE 8. Features of the transmitted preamble. (a)–(d) represent four features due to Conjugate Symmetry property and (e)–(h) represent four features due to correlation property.

F. PRE-PROCESSING

CNN works only on real data, whereas our feature grid consists of complex values. Therefore, it is necessary to convert complex values to real data. During pre-processing, we divided each complex feature vector into two real feature vectors representing the real and imaginary parts of the complex vector separately. After pre-processing, the Doppler feature grid suitable for the CNN is given by:

$$F = \{Y_R, Y_I, Z_R, Z_I\} \tag{46}$$

where Y_R and Y_I represent the real and imaginary parts of feature grid Y. Similarly, Z_R and Z_I represent the real and imaginary parts of the feature grid Z.

IV. DESCRIPTION OF CNN MODEL

Initially, a CNN was developed for computer vision, where a two-dimensional image database was analysed for classification-related applications [76]. A CNN is a special form of a multi-layer neural network. The CNN architecture consists of several convolutional layers, ReLU layers, pooled layers, a fully connected layer, a softmax layer, and a classification layer stacked together [77].

We propose two versions of CNN architectures. CNN architecture – 1 is as shown in Fig. 9. The basis of the CNN is the convolutional layer, which performs the convolution operation on the input image. In this study, we consider a feature grid of size $N/4 \times 8 \times 1$ as an input image to the CNN model.

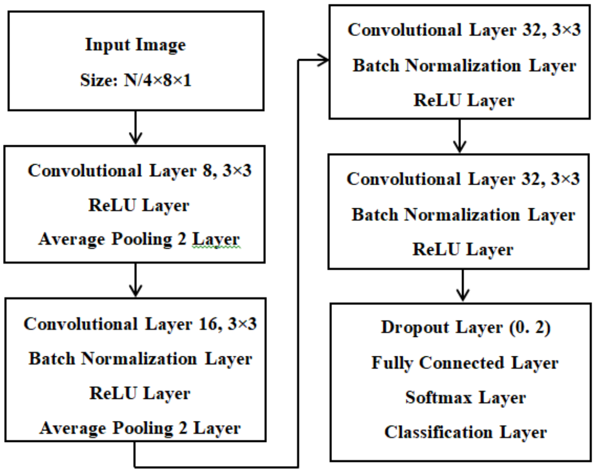


FIGURE 9. CNN architecture -1 with feature grid as input image.

The proposed CNN architecture - 1 is presented in five sections, where each section represents a set of layers. The first section consists of a convolutional layer, ReLU layer, and average pooling layer. The convolutional layer passes this image through a set of convolutional filters to create feature maps from the features hidden in the image. The convolutional layer consists of eight filters, each with a typical size of 3 × 3. This is followed by an activation function, known as a rectified linear unit (ReLU), which maps all negative sample values to zero so that features with only positive sample values are carried forward to the next layer. Then, the average pooling layer computes the average pixel values in the rectangular regions. We considered a pool size of 2, that is, the height and width of the rectangular regions were both 2. In addition, the stride i, the step size of 2, is considered so that the pooling regions do not overlap.

The output of the first section is applied to the second section which consists of four layers: convolutional layer, batch normalisation layer, ReLU layer, and average pooling layer. A convolutional layer with 16 filters, each with a filter size of 3 × 3. The batch normalisation layer performs normalisation along the mini-batches. It speeds up the training involved in the convolutional layer and makes learning easier. The batch normalisation layer is followed by the second ReLU layer and the second average pooling layer. The average pooling layer output is applied to the third section which consists of three layers: convolutional layer, batch normalisation layer, and ReLU layer. The convolutional layer consists of 32 filters, each with a filter size of 3 × 3, followed by the batch normalisation layer and ReLU layer. The three layers of the third section are repeated to form the fourth section. The first layer of the final section is the dropout layer. We considered a dropout layer with a dropout probability of 0.2. This means that the dropout layer randomly selects 20% of the pixels of the image and nullifies them. A fully connected layer utilises the features extracted from the previous stages and predicts the class of the image, whereas the softmax layer performs a multi-classification task. It produces classification results which include the probability of each classification class. The final layer of the proposed CNN is the classification layer. It computes the cross-entropy loss for each classification class obtained from the softmax layer and performs weighted classification on mutually exclusive classes.

We propose CNN architecture – 2 as shown in Fig. 10. It is the simplified version of CNN architecture – 1. It consists of one convolutional layer with 32 filters, each with a filter size of 3 × 3. It is followed by the batch normalisation layer, ReLU layer and Dropout Layer with 20% dropout. Dropout layer is followed by fully connected layer, softmax layer and finally classification layer.

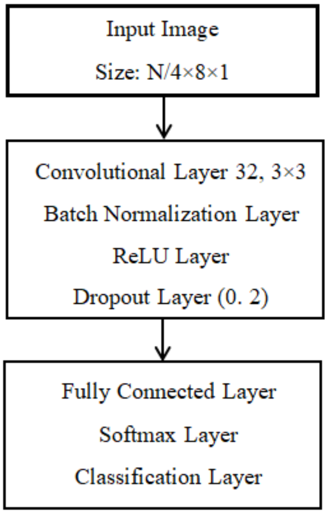


FIGURE 10. CNN architecture – 2.

V. UWA CHANNEL SIMULATION

Underwater and acoustic channels present a huge challenge for high-speed and reliable communication because they are not only vulnerable to interference and noise but are also highly affected by impediments, Doppler shift (moving environment), and changing communication environment dynamics. The transmitted radio signals are likely to be diffracted, scattered, reflected, and attenuated during their transmission. Therefore, when the acoustic signals reach the receiver, they may have different transmission paths and random phases, and their power is attenuated. The UWA channel exhibits three main characteristics: low speed of sound that varies with medium conditions, time-and frequency-dependent attenuation, and time-varying multipath propagation that depends on boundary conditions [78]. In an underwater acoustic channel, the Doppler frequency shift is caused by (i) the drifting of the transmitter/receiver due to water movement, (ii) the reflections caused by the surface wave motion, and (iii) if either the transmitter, receiver, or both are mobile and move at a particular speed. Among the three types of Doppler scaling factors, the Doppler scaling factor due to the mobility of the vehicle dominates the remaining two. However, Doppler cannot be neglected due to drifting and surface waves.

We simulated the UWA channel using the Stojanovic channel model [4]. This model assumes flat bathymetry, surface shape, and sound speed profile, and provides good results for flat bathymetry. To obtain large-scale simulations, we used Bellhop [79]. The UWA channel geometry considered in the channel simulation is shown in Fig.11. We considered a bathymetry of 100 m depth, one transmitter, and one receiver located at 80 m and 40 m from the bottom of the channel (seafloor), as shown in Fig. 10. The transmitter and re.ceiver were separated by a range of 1000 m.

The varying sound speed profile (SSP) is considered with sound speed changing in the range of 1539 m/s to 1534 m/s at a UWA channel depth of 100 m. Additionally, the surface of the UWA channel is considered non-flat.

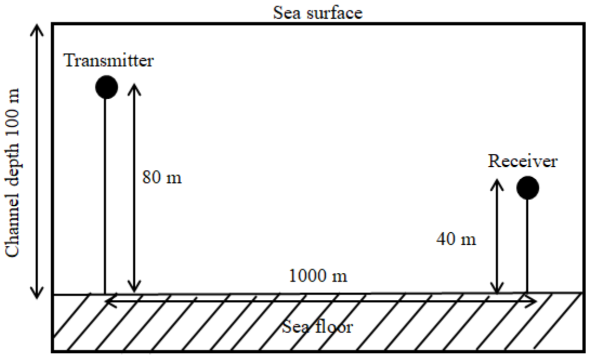


FIGURE 11. UWA channel geometry.

With the ray-tracing option, Bellhop produces an Eigen ray file containing a fan of rays emanating from the source. Bellhop computes the arrival time and amplitude data. Doppler scaling factors due to drifting and surface wave motion are included in the channel simulation, and the Doppler scaling factor due to vehicular motion is applied externally. The instantaneous channel gain of the simulated channel model is shown in Fig. 12.

Simulations were conducted in MATLAB over 1000 iterations with 360 different UWA channel impulse responses. Six paths were considered in the UWA channel simulations. As shown in Fig. 13, the simulated channel consists of six paths with approximate delays of 0 ms, 0.2 ms, 1.1ms, 2, 4, and 8 ms. These delays change slowly with time owing to the Doppler effect.

Fig. 13 shows the normalised channel impulse responses for four different iterations. In Fig. 13(a), the 3rd path is the strongest path, the first three paths are strong in Fig. 13(b), 2nd and 3rd paths are strong in Fig. 13(c) and 2nd path is the strongest, as shown in Fig. 13(d). The path delay by the longest significant path is called the total multipath delay spread. In the simulations shown in Fig. 13, the total multipath delay spread was 8 ms.

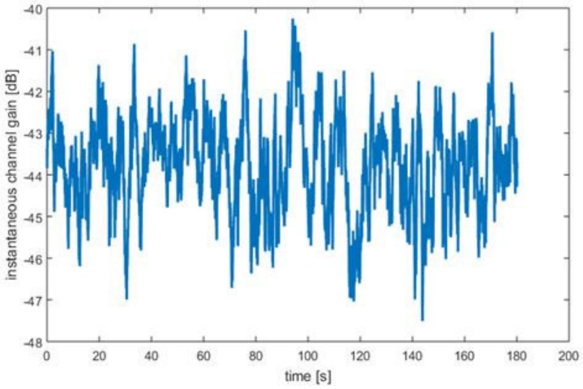


FIGURE 12. Instantaneous channel gain of UWA channel.

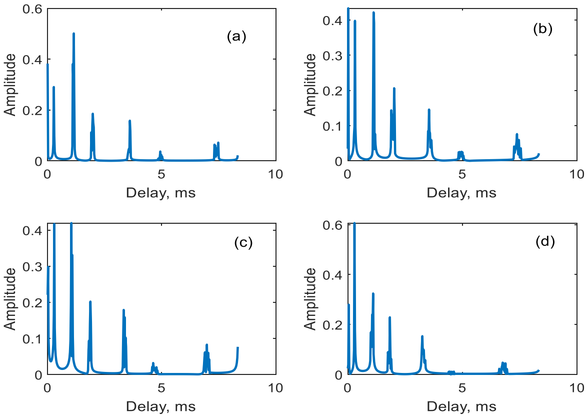


FIGURE 13. (a)–(d): UWA channel impulse response at four different time instances.

We repeated the experiment for different SNR values. A total of 1000 different channel and noise realisations were conducted at each SNR value with SNR ranging from 0 dB to 40 dB in steps of 5 dB.

For the impulse response of the channel along each path, the normalised channel impulse responses of Fig. 13 are shown for a 2 ms delay period in Fig. 14. The channel impulse response along each path changed in each iteration. As shown in Fig. 14, the delay spread of individual paths may be different but less than the total multipath delay spread.

VI. SIMULATION RESULTS

A. PERFORMANCE OF TIMING METRIC

With the simulation parameters in Table 1, a frame with the preamble followed by a data payload with 10 FBMC symbols was generated. The performance of the timing metric in the ideal case is shown in Fig. 6. The time offset of 10 samples was added to the frame, then it was passed through the UWA channel and a Doppler shift was applied at the channel output. Then, additive white Gaussian noise at SNR = 0 dB was added. Hence, the received signal is time-shifted and distorted

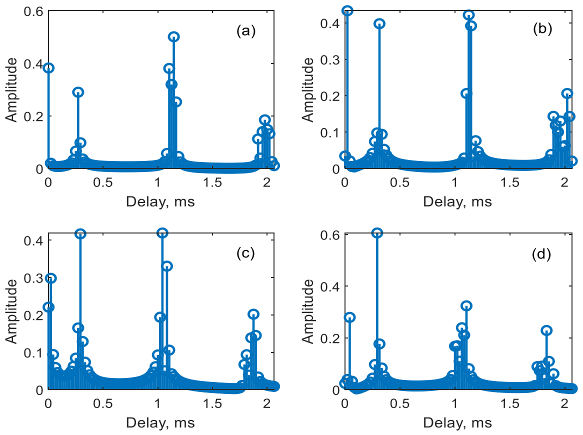


FIGURE 14. (a)–(d): Normalized channel impulse responses (a)-(d) of Fig. 13 over 2 ms period.

owing to the channel, and the Doppler affects the transmitted noisy signal. The timing metric was applied to the received signal, and the resultant magnitude plot is shown in Fig. 15.

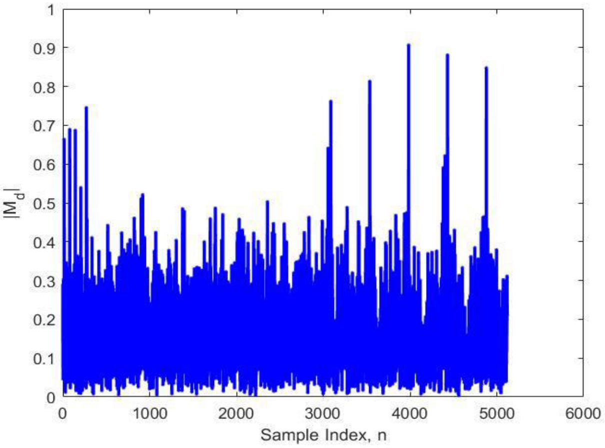


FIGURE 15. Timing metric on UWA channel affected and Doppler shifted received signal.

The enlarged version of the preamble part of Fig. 15 is shown in Fig. 16.

Fig. 16 shows several significant peaks. Significant peaks at sample indices {26, 90, 154, 218, 282} exhibit periodicity of N/2 = 64 samples with different amplitudes due to channel distortion. The first peak occurred at time index n = 26. That means the proposed metric produces a significant peak along the first channel path. Therefore we used threshold level V_T for peak identification.

A threshold level V_T is set in such a way that the miss detection probability P(M) and false detection probability P(Fa) are very low as compared to correct detection probability P(C). With the proposed timing metric amplitude more than threshold level V_T, if the first peek of the timing metric falls at the frame boundary i.e., at the time index $d = d_{opt}$, it is called the correct detection, else it is called a false alarm. If the proposed timing metric amplitude is less than threshold level V_T, it is called miss detection. Mathematically, probability of correct detection is defined as

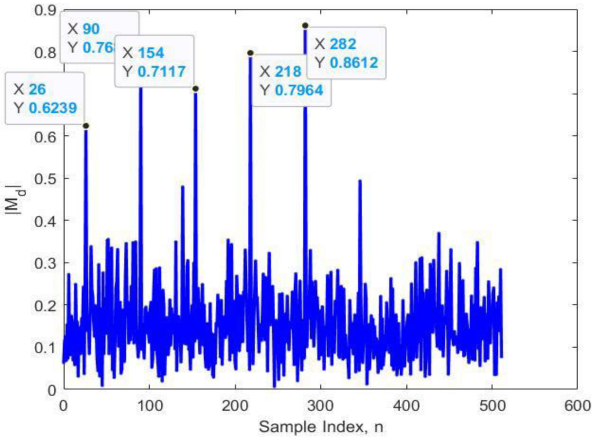


FIGURE 16. Enlarged version of preamble part of Fig. 15.

$$P(C) = \Pr\{|M(d)| \geq V_T\}; \quad d = d_{opt} \tag{47}$$

The probability of false alarm is defined as

$$P(Fa) = \Pr\{|M(d)| \geq V_T\}; \quad d \neq d_{opt} \tag{48}$$

Similarly, the probability of miss detection is defined as

$$P(M) = \Pr\{|M(d)| < V_T\}; \quad d \neq d_{opt} \tag{49}$$

We noted the number of correct detections, number of false detections, and missed detections of frame boundaries by varying the SNR in steps of 5 dB from 0 dB to 40 dB. The probability of correct detection is calculated as the ratio of the number of correct detections to the total number of detections. The probability of false alarm and probability of missed detection were calculated similarly.

Fig. 17 shows the probability of correct, false, and missed detections to the SNR.

B. COMPARISON OF PROPOSED TIMING METRIC WITH EXISTING TIMING METRICS

We compared the performance of the proposed timing metric with three reference methods. The first method is Schmidl & Cox [80] method. Cox's timing metric estimates the frame based on the correlation between two repeated sections of the preamble. FBMC symbol does not contain CP but, as shown in Fig. 4, the preamble contains three partially identical sections on either side of two identical sections. Therefore, the Schmidl & Cox method on two identical halves of the preamble produces a plateau and causes ambiguity in frame boundary detection. For the frame boundary detection in the presence of plateau, averaging method Schmidl and Cox proposed an averaging method in which 90% of points on right and left of the maximum point are found and an average of them is considered as frame boundary. The second reference method is Chung's [39] method and. it is the improved version

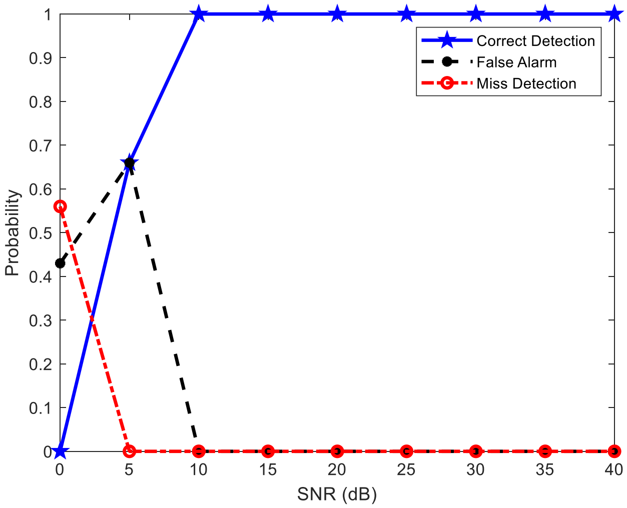


FIGURE 17. Probability of correct detection, false alarm and miss detection w.r.to SNR (dB).

of Smidl & Cox's method, and it is based on the weighted correlation between two repeated sections of the preamble. Our proposed method is based on conjugate symmetry present in the preamble sequence and identification of the first significant peak. The third reference method is Liming's timing metric [81] and it is based on weighted conjugate symmetric sequence.

We compared the performances of the four methods under consideration through MATLAB simulations. We added complex Gaussian noise to the received signal. The mean of the noise is zero while its variance depends on SNR. We conducted the simulation experiments 1000 times each time with different noise realizations. We considered the root-mean-square error (RMSE) as a metric for checking the accuracy of the timing metrics. RMSE is defined as

$$RMSE = \sqrt{\frac{\sum_{i=1}^{N} \left(\lambda_{FB} - \widehat{\lambda_i}\right)}{N}}$$

where λ_{FB} represents the actual frame boundary and $\widehat{\lambda_i}$ repsents i^{th} estimated frame boundary and N represents the total number of observations.

We varied SNR in steps of 3 dB from 0 dB to 30 dB and determined the RMSE of the timing metric at each SNR value. With the AWGN channel, the proposed method and Liming methods yielded exact frame boundary i.e., zero RMSE for all SNR values from 8 dB onwards and small RMSE for SNR $\leq$ 8 dB as shown in Fig. 18. With AWGN channel, RMSE of Cox's method varies in the range 66 to 41.5 for SNR changes from 0 dB to 30 dB. Chung's method exhibited superior performance as compared to Cox's method and RMSE of Chung's method varies from 26 to 8.7 over SNR = 0 to 30 dB.

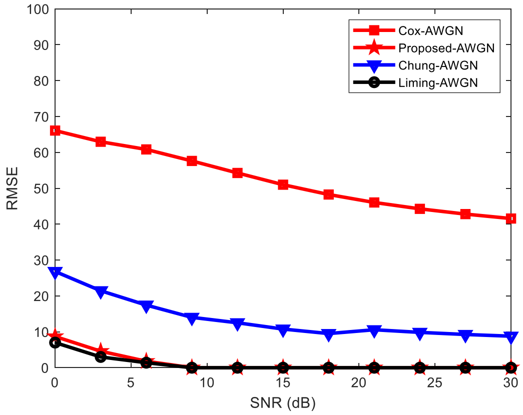


FIGURE 18. RMSE of time estimation versus SNR in AWGN channel.

We repeated the simulation experiments 1000 times each time with different channel and noise realizations for two different Doppler scale values $\varepsilon = 0$ and $\varepsilon = 0.006$.

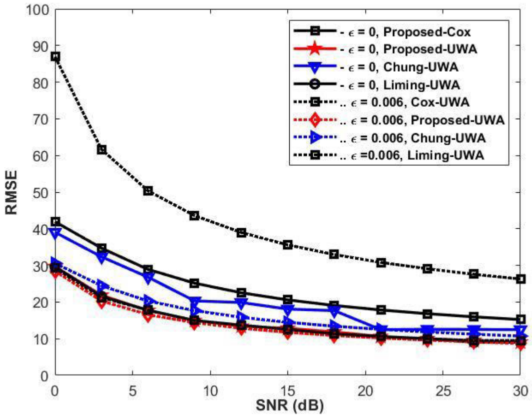


FIGURE 19. RMSE of time estimation versus SNR in UWA channel.

As shown in Fig. 19, Cox's method exhibits the highest RMSE as compared to the other three methods, and its RMSE increases as the Doppler scale increases. Chung's method exhibits better performance than the Cox's method. Whereas the proposed method and the Liming method exhibited nearly similar performance irrespective of the Doppler scale values. RMSE of all the methods increased with UWA channels.

The computational complexity of Schmidl & Cox's method and the proposed method is the same because both require N complex multiplications and additions per sample. The computational complexities of the Chung's method and the Liming's method are 2N complex multiplications and N additions each.

C. GENERATION OF DATASET

An A-frame with a preamble followed by a data payload was generated. The data payload consists of ten FBMC-OQAM symbols. A time-shifted frame was passed through the UWA channel to a frame time offset of 10 samples. A Doppler frequency offset is added to the channel output. At the receiver, we first estimated the frame boundary using the proposed timing metric, and then a feature grid of size 31 × 16, where 31 is the number of channels and 16 is the number of samples, was generated by applying the correlation and conjugate symmetry properties on the preamble. Its Doppler scale is considered as the label.

We considered the maximum underwater vehicle speed to be 150 km/h, which results in a Doppler scale equal to 0.027. The Doppler scale was varied from 0 to 0.027 in steps of 0.001. For each Doppler scale, 1000 feature grids were generated by considering 1000 different UWA channels. For 28 Doppler scales, we generated 28000 feature grids and 28000 corresponding labels. All 28000 feature grids were preprocessed and applied to train the CNN. With this process, we prepared a dataset consisting of 28000 pairs of feature grids and their corresponding labels.

D. PERFORMANCE OF CNN-BASED DOPPLER SHIFT ESTIMATION

Out of 28000 feature grids of the dataset, 256 feature grids were randomly selected for validation, 256 feature grids were randomly selected for testing, and the remaining were used for training the CNN.

The proposed CNN consists of 18 layers, including an input image layer, four convolution layers, four ReLU layers, three batch normalisation layers, two average pooling layers, one dropout layer, one fully connected layer, one softmax layer, and a classification layer, as shown in Fig. 3. Although the basic architecture of our proposed CNN is similar to the well-known computer vision-related networks, we empirically determined the exact configuration based on pilot studies on Doppler signal input data.

Zero-padding was used in the Doppler feature grid before each convolution. The number of filters in the convolution layers is 8, 16, 32, and 32, respectively, and all filters have a filter size of 3 × 3. Average pooling was performed on 2 × 2 regions with a stride of two. Batch normalisation was used to normalise the mini-batches across each channel. The ReLU layer eliminates negative values in the feature maps produced by the convolution layer. The dropout layer is used with a probability of dropout chances of 0.2 and this layer nullifies 20% of randomly selected data in the feature maps. The network was trained using a stochastic gradient descent with momentum (SGDM) algorithm with an initial learning rate of 0.0003. The network was validated using a validation frequency of 21 iterations. The layers, size of kernels, and number of kernels for each layer used in the proposed CNN architecture are listed in Table 2.

The stochastic gradient descent with momentum (SGDM) optimiser with the following parameters was used:

- the number of epochs = 5,
- Learning rate = 0.0003 and
- Mini batch size = 256 (number of feature grids and corresponding labels

TABLE 2. The layers, size of kernels, number of kernels for each layer used in the proposed CNN architecture.

Layer	Name	Type	Kernel size, number of kernels, strides and padding
1	'imageinput'	Image Input	31×16×1
2	'conv_1'	Convolution	8 3×3×1 convolutions with stride [1 1] and padding 'same'
3	'relu_1'	ReLU	ReLU
4	'avgpool2d_1'	Average Pooling	2×2 average pooling with [1 1] and padding [0 0 0 0]
5	'conv_2'	Convolution	16 3×3×8 convolutions with stride [1 1] and padding 'same'
6	'batchnorm_1'	Batch Normalization	Batch normalization with 16 channels
7	'relu_2'	ReLU	ReLU
8	'avgpool2d_2'	Average Pooling	2×2 average pooling with [1 1] and padding [0 0 0 0]
9	'conv_3'	Convolution	32 3×3×16 convolutions with stride [1 1] and padding 'same'
10	'batchnorm_2'	Batch Normalization	Batch normalization with 32 channels
11	'relu_3'	ReLU	ReLU
12	'conv_4'	Convolution	32 3×3×32 convolutions with stride [1 1] and padding 'same'
13	'batchnorm_3'	Batch Normalization	Batch normalization with 32 channels
14	'relu_4'	ReLU	ReLU
15	'dropout'	Dropout	20% dropout
16	'FC1'	Fully Connected	28 fully connected layer
17	'SoftMax'	Softmax	Softmax
18	'Output'	Classification Output	Crossentropyex with '0' and 27 other classes

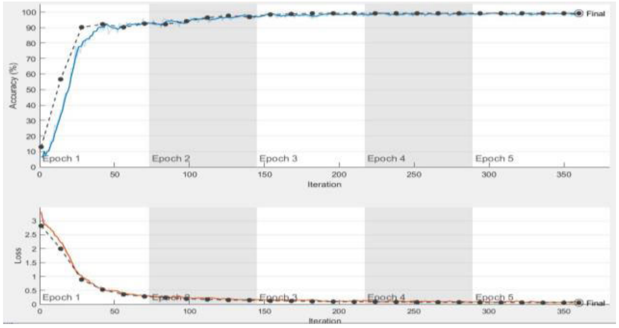


FIGURE 20. Training Progress at SNR = 5 dB.

Fig. 20 shows the training progress at an SNR of 5 dB. It consists of the percentage validation accuracy and data validation loss curves with respect to the number of iterations. The training process on this dataset required an elapsed time of 55 s. Within the elapsed time, it covered five epochs. The number of iterations per epoch is 72. The proposed CNN architecture with a validation frequency of 14 iterations and a 0.0003 learning rate yielded 99.22% validation accuracy by the time it completed the 5th epoch.

The data loss validation curve (in Fig. 20) does not have a plateau at the beginning; hence, the weight initialisation is good. The initial data loss is 3 which is a low value and exponentially reduces to 0.5, by the 60th iteration (within the first epoch) and then gradually reduces to zero by the time it completes the 5th epoch, that is, 360 iterations. This indicates that the selected learning rate is sufficient for the loss validation curve to reach a local minimum within a few epochs.

The confusion matrix is illustrated in Fig. 21. Randomly selected 256 feature grids containing different Doppler scales. For example, nine feature grids relate to zero Doppler scale, seven feature grids correspond to Doppler scale equal to 0.001, and so on. The applied Doppler scales are called true classes. After the CNN training progresses, the estimated Doppler scale is called the predicted class. Our proposed method predicted 253 true classes, and three classes were predicted as their adjacent class; that is, the proposed CNN-based Doppler scale estimation method achieved 98.83% testing accuracy.

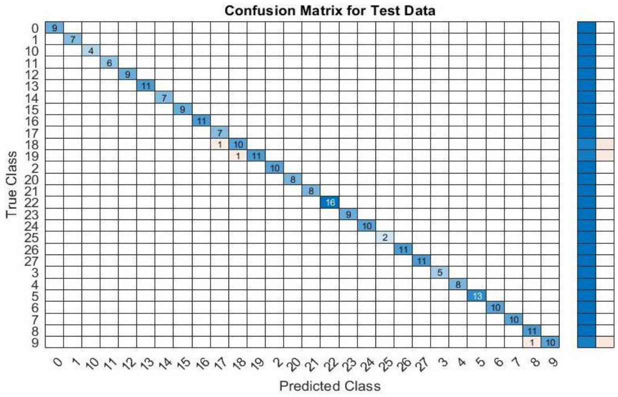


FIGURE 21. Confusion Matrix of CNN classifier-1 at SNR = 5 dB.

The performance of the proposed CNN architecture-1 and CNN architecture-2 based classifiers were tested using different parameter values. Test accuracy depends on several parameters. The test accuracy is determined by varying the number of epochs, learning rate, batch size, and SNR. The parameters are listed in Table 3.

TABLE 3. List of by-default parameters of CNN.

Parameter	Value
SNR	0 dB
Number of Epochs	10
Learning Rate	0.0003
Batch Size	1024

Fig. 22(a) shows the test accuracy of the classifiers with respect to the learning rate. Test accuracy of CNN classifier-1 and CNN classifier-2 is 72% and 48% respectively at a learning rate of 0.0001. The test accuracy increased up to 0.0005 and then decreased. The number of epochs was varied from 1 to 15 in steps of 1, and test accuracy was measured for each epoch value. Fig. 22(b) shows the performance of the proposed CNN classifiers with respect to the number of epochs. The test accuracy increased up to the 7th epoch and then decreased. Test accuracy of CNN classifier-1 is higher than that of CNN classifier-2 at all epoch values from 1 to 15. For the default value of the number of epochs, that is, ten epochs, the test accuracy of CNN classifier-1 was 82%, whereas the test accuracy of CNN classifier-2 was 77% for the number of seven epochs.

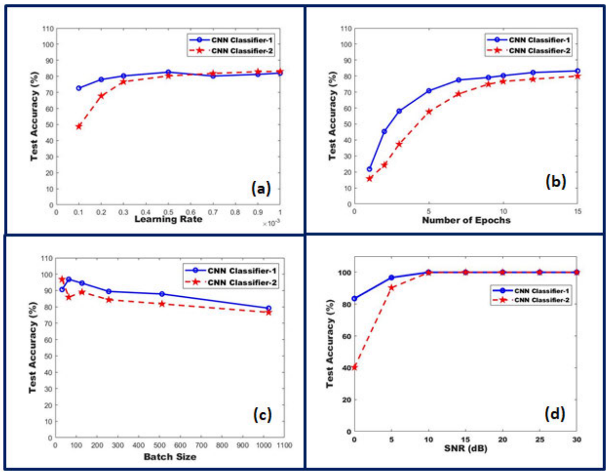


FIGURE 22. Performance of CNN classifiers with respect to (a) Learning Rate, (b) Maximum Number of Epochs, (c) Batch Size, and (d) SNR.

The batch size was selected from the set of values {32, 64, 128, 256, 512, 1024}. As the batch size decreased, the number of iterations per epoch increased and took more time. At the lower batch sizes (32 or 64), CNN yields high validation accuracy, but the training accuracy shows large variation as compared to the validation accuracy at each iteration. The training progress curve closely followed the validation curve at a higher batch sizes (128 or above). Fig. 22(c) shows the test accuracy of the batch size. For the set of {32, 64, 128, 256, 512, 1024}, the set of the test accuracies exhibited by the CNN classifier-1 is {90.762, 96.87, 94.53, 89.45, 87.89, 79.20}, whereas the set of the test accuracies exhibited by the CNN classifier-2 is {96.87, 85.93, 89.06, 84.37, 81.84, 76.66}. For the same set of batch sizes, test accuracy exhibited by CNN classifier-2 With five epochs, batch size = 1024, learning rate = 0.0003, SNR was varied from 0 dB to 30 dB in steps of 5 dB, and the testing accuracy of the proposed CNN classifiers is measured at each SNR. As shown in Fig. 22(d), the accuracies of CNN classifier-1 and CNN classifier-2 at 0 dB SNR are 83.5% and 76.66% respectively and accuracies of both classifiers are 100% from SNR = 10 dB onwards.

E. STACKED AUTOENCODERS BASED DOPPLER SHIFT ESTIMATION

Mingqian Liu et al [82] proposed stacked autoencoders based wireless signal classification. Choi-Williams distribution (CWD) was used to generate two-dimensional time-frequency images corresponding to wireless signals. This feature classification method needs further processing like conversion of time-frequency images to binary images, application of cropping algorithm for searching presence of signals, and adjustment of cropped time-frequency images to the required image size by a bicubic interpolation algorithm. The entire process is very complex and time-consuming.

In this paper, we consider stacked autoencoders for Doppler frequency estimation. Instead of applying CWD

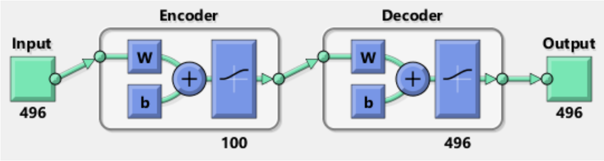


FIGURE 23. Block diagram of autoencoder-1.

and processing algorithms on the input signal, we consider feature grids as input images to the stacked autoencoder network for comparing its performance with the performance of CNN classifier. Feature grids are pre-processed to acquire grey image properties i.e., pixels are positive integers in the range 0 to 255.

Complex feature grid is converted to two real grids containing real and imaginary parts of the complex signal grid. The magnitude of each grid is taken to have all positive values. Grids contain fractional values; they are multiplied with 10^4 and rounded off to the next integers.

We considered two types of stackednets. Architecture of Stackednet – 1 consists of two sparse autoencoders joined with softmax layer as shown in Fig. 23. Input feature grid size is 31 × 16 i.e, input array to the stackednet is 496 samples. The hidden size of autoencoder-1 is 100 and hidden size of autoencoder-2 is 50.

Each sparse autoencoder of stackednet-1 was trained separately in an unsupervised manner and softmax layer was trained. Then all the sections were cascaded to form stacked autoencoder network. Finally stacked network was trained in a supervised manner.

The number of neurons in the hidden layer of autoencoder 1 was set to 100. The coefficient for L_2 weight regularizer in the loss function with weight regularization factor 0.004 was used for the weights of the network. The sparsity regularizer controls the sparsity of the output from the hidden layer. Sparsity Proportion and sparsity regularization are the parameters of sparsity regularizer and they are set to 0.15 and 4 respectively. Autoencoder 1 was trained with the maximum number of epochs equal to 100. As shown in Fig. 20, the autoencoder consists of an encoder followed by a decoder. The encoder maps an input of size 496 samples to 100 samples using 100 hidden neurons. Decoder follows the reverse mapping i.e., reconstructs the input size of 496 samples.

The autoencoder-1 was trained with the training dataset consisting of 8000 received feature vectors each with 496 samples and it produces the same size data at the output. The decoded data of size 496 × 8000 from the autoencoder-1 was further encoded to 100 × 8000 size for getting new feature vectors.

The block diagram of autoencoder-2 is as shown in Fig. 24. Similar to the autoencoder-1, autoencoder-2 was designed with the following parameters in Table 4.

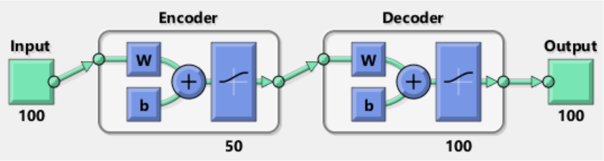


FIGURE 24. Block diagram of autoencoder-2.

TABLE 4. List of parameters values for Autoencoder-2.

Number of neurons of hidden layer	50
Maximum number of epochs	50
L_2 weight regularization	0.002
Sparsity Proportion	0.1
Sparsity Regulization	4

Input dataset to the autoencoder-2 is the encoded output of autoencoder-1 which is of the size 100 × 8000. After training the autoencoder-2, we get the encoded output of the autoencoder-2 with a size 50 × 8000.

In dataset was generated as described in subsection B of Simulation Results in section VI by considering 10 different Doppler scales from zero in steps of 0.001. The softmax layer follows the Autoencoder-2 and is the final layer in the proposed architecture of the Stackednet-1. The softmax layer activation function returns the probability of each of ten classes of Doppler shift. The block diagram of the softmax layer is shown in Fig. 25. Input to the softmax layer consists of 50-dimensional feature vectors. With the help of labels of training data, The softmax layer was trained in a supervised manner. It converts the input dataset of size 50 × 8000 to 10 classes with corresponding probabilities.

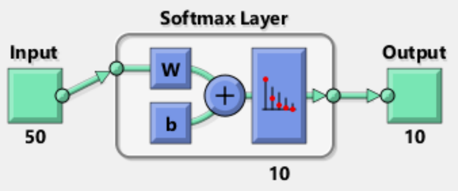


FIGURE 25. Block diagram of softmax layer.

The complete architecture of stackednet-1 is shown in Fig. 26. 1024 feature vectors from 10 different classes were applied to stackednet-1 and tested its performance.

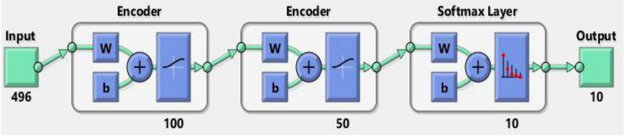
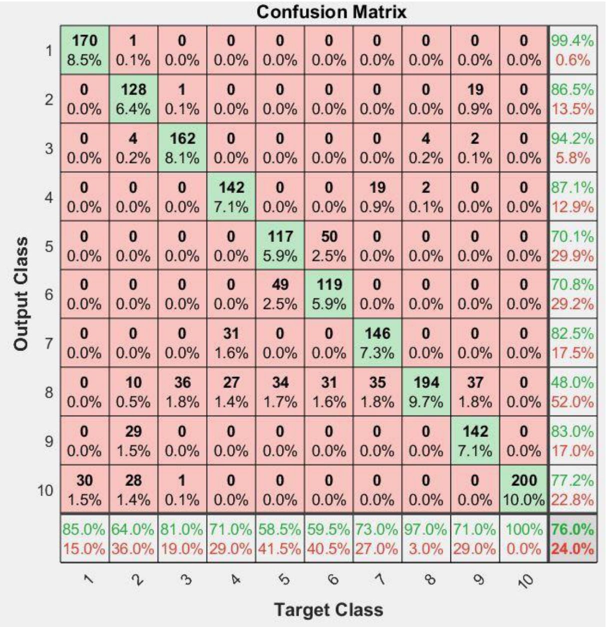


FIGURE 26. Architecture of Stackednet -1.

A Confusion matrix is used to describe the performance of the stackednet-1. Fig. 27 shows the confusion matrix of the stackednet-1 classifier at SNR = 5 dB. It shows overall test accuracy of 76%.



Confusion Matrix

Output Class	1	2	3	4	5	6	7	8	9	10	
1	170 8.5%	1 0.1%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	99.4% 0.6%
2	0 0.0%	128 6.4%	1 0.1%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	19 0.9%	0 0.0%	86.5% 13.5%
3	0 0.0%	4 0.2%	162 8.1%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	4 0.2%	2 0.1%	0 0.0%	94.2% 5.8%
4	0 0.0%	0 0.0%	0 0.0%	142 7.1%	0 0.0%	0 0.0%	19 0.9%	2 0.1%	0 0.0%	0 0.0%	87.1% 12.9%
5	0 0.0%	0 0.0%	0 0.0%	0 0.0%	117 5.9%	50 2.5%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	70.1% 29.9%
6	0 0.0%	0 0.0%	0 0.0%	0 0.0%	49 2.5%	119 5.9%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	70.8% 29.2%
7	0 0.0%	0 0.0%	0 0.0%	31 1.6%	0 0.0%	0 0.0%	146 7.3%	0 0.0%	0 0.0%	0 0.0%	82.5% 17.5%
8	0 0.0%	10 0.5%	36 1.8%	27 1.4%	34 1.7%	31 1.6%	35 1.8%	194 9.7%	37 1.8%	0 0.0%	48.0% 52.0%
9	0 0.0%	29 1.5%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	142 7.1%	0 0.0%	83.0% 17.0%
10	30 1.5%	28 1.4%	1 0.1%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	200 10.0%	77.2% 22.8%
	85.0% 15.0%	64.0% 36.0%	81.0% 19.0%	71.0% 29.0%	58.5% 41.5%	59.5% 40.5%	73.0% 27.0%	97.0% 3.0%	71.0% 29.0%	100% 0.0%	**76.0%** **24.0%**

Target Class

FIGURE 27. **Confusion Matrix of stackednet-1 classifier at SNR = 5 dB.**

We propose an architecture of the stackednet-2 as shown in Fig. 28. It consists of one sparse autoencoder followed by the softmax layer. The hidden size of the autoencoder is 100.

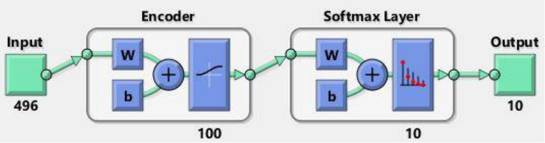
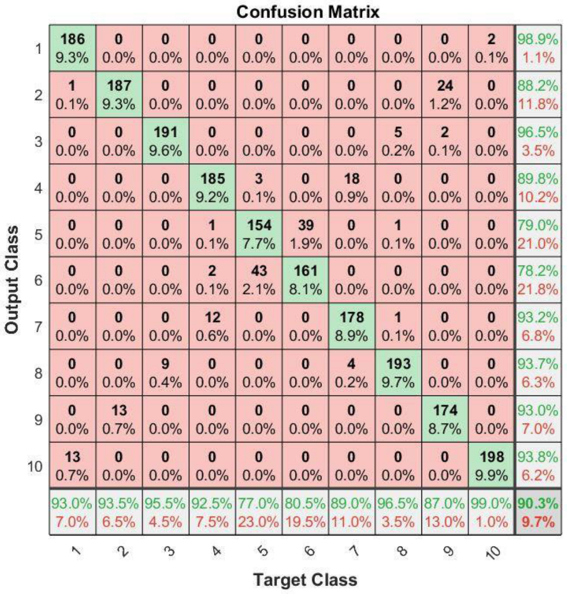


FIGURE 28. **Architecture of Stackednet-2.**

The stackednet-2 was trained for the same dataset and tested its performance for the same set of feature vectors that were used for the stackednet-1. It exhibited overall test accuracy of 90.3% at SNR= 5 dB which is very high as compared to the test accuracy of 76% of the stackednet-1.

Performance of proposed CNN classifiers and stackednets was compared in terms of test accuracy with respect to SNR. SNR was varied 0 dB to 30 dB in terms of 5 dB and test accuracy of each of the four classifiers was estimated. Fig. 29 shows the plot of the test accuracy with respect to SNR for the CNN classifier-1, CNN classifier-2, stackednet-1 and stackednet-2. At SNR = 0 dB, CNN classifier-1 exhibits superior performance as compared to other classifiers. Its test accuracy was 83% which is 5% more than the test accuracy of CNN classifier-2.

Test accuracies of stackednet-1, and stackednet-2 are 31.6% and 40.2% respectively. Test accuracy of stackednet-2 is s 90.3% which is the same as that of CNN classifier-2. CNN classifier-2 yields 97% test accuracy at SNR = 5 dB, and it is the highest accuracy among all the four classifiers. Except stackednet-1, the remaining three classifiers exhibit 100% test accuracy from SNR = 10 dB onwards.



Confusion Matrix

Output Class	1	2	3	4	5	6	7	8	9	10	
1	186 9.3%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	2 0.1%	98.9% 1.1%
2	1 0.1%	187 9.3%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	24 1.2%	0 0.0%	88.2% 11.8%
3	0 0.0%	0 0.0%	191 9.6%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	5 0.2%	2 0.1%	0 0.0%	96.5% 3.5%
4	0 0.0%	0 0.0%	0 0.0%	185 9.2%	3 0.1%	0 0.0%	18 0.9%	0 0.0%	0 0.0%	0 0.0%	89.8% 10.2%
5	0 0.0%	0 0.0%	0 0.0%	1 0.1%	154 7.7%	39 1.9%	0 0.0%	1 0.1%	0 0.0%	0 0.0%	79.0% 21.0%
6	0 0.0%	0 0.0%	0 0.0%	2 0.1%	43 2.1%	161 8.1%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	78.2% 21.8%
7	0 0.0%	0 0.0%	0 0.0%	12 0.6%	0 0.0%	0 0.0%	178 8.9%	1 0.1%	0 0.0%	0 0.0%	93.2% 6.8%
8	0 0.0%	0 0.0%	9 0.4%	0 0.0%	0 0.0%	0 0.0%	4 0.2%	193 9.7%	0 0.0%	0 0.0%	93.7% 6.3%
9	0 0.0%	13 0.7%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	174 8.7%	0 0.0%	93.0% 7.0%
10	13 0.7%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	0 0.0%	198 9.9%	93.8% 6.2%
	93.0% 7.0%	93.5% 6.5%	95.5% 4.5%	92.5% 7.5%	77.0% 23.0%	80.5% 19.5%	89.0% 11.0%	96.5% 3.5%	87.0% 13.0%	99.0% 1.0%	**90.3%** **9.7%**

Target Class

FIGURE 29. **Confusion Matrix of stackednet-2 classifier at SNR = 5 dB.**

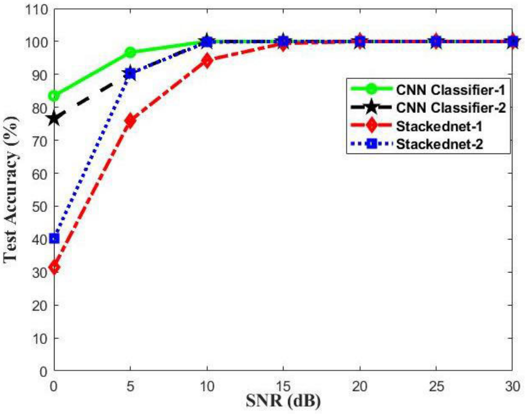


FIGURE 30. **Comparison of performance of CNN and stackednet classifiers.**

We considered the off-line training time in the computational complexity analysis. The off-line training time of CNN classifier-1 is 428.71 seconds, the off-line training time of CNN classifier-2 is 197.97 seconds, the off-line training time of the stackednet-1 is 220.29 and the off-line training time of the stackednet-2 is 217.32 seconds. From this computational complexity analysis, it can be seen that the computational complexity of the proposed CNN classifier-2 is the lowest among all the four classifiers.

VII. CONCLUSION

This paper proposes frame boundary and Doppler scale estimation algorithms for the FBMC-OQAM underwater communication system. The proposed algorithms and computer simulation results were as follows.

1) A data-aided frame boundary estimation technique suitable for the FBMC-OQAM system in the UWA

channel is presented. Specifically, we developed a technique based on the periodic conjugate symmetry present in the preamble. Owing to its periodic nature, the timing metric produces multiple peaks. In particular, the proposed timing metric yielded six significant peaks during the preamble, and the first peak indicated the frame boundary. The performance of the proposed frame boundary estimation method was compared with timing metrics related to Schmidl & Cox, Chung, and Liming. The accuracy of the timing metrics was checked with the RMSE parameter. For the demonstration of the performance of timing metrics, simulations were conducted for AWGN and UWA channels. From the simulation results, it was found that the RMSE of the proposed metric is very close to the RMSE of Liming's method and also, the computational complexity of the suggested method is lesser than that of Liming's method.

2) The FBMC-OQAM system is prone to a severe Doppler effect in the UWA channel. We recommend two types of CNN classifiers for Doppler scale estimation. The frame boundary indicates the first sample of the preamble. We applied the correlation and conjugate-symmetry properties to the preamble and generated feature grids. After pre-processing, feature grids were applied to the CNN, and the Doppler scale was estimated using a CNN-based classification approach.
3) The PHYDYAS shaping filter was considered in the FBMC-OQAM symbol generation. The proposed time and frequency algorithms exhibit superior performance in the AWGN and UWA channels. The proposed timing metric performs well for an SNR $\geq$ 5 dB. At SNR = 5 dB, it exhibits a 67% correct probability of accurate frame boundary detection, 33% false frame boundary detection, and zero miss detection. The false detection-related peaks fall along the strongest path of the channel. The probability of correct frame detection was 100% for an SNR $\geq$ 10 dB. The proposed CNN classifiers-based Doppler scale estimation algorithm yielded 100% testing accuracy for SNR $\geq$ 10 dB.
4) Stacked autoencoders were considered for Doppler scale estimation. Feature grids were preprocessed to form a feature vector dataset. We propose two types of Stackednets. They were trained with the dataset and tested. At 10 dB SNR, The stackednet-1 yielded 94% testing accuracy whereas stackednet-2 yielded 100% testing accuracy.
5) CNN classifier-1 exhibits superior performance, but its computational complexity is very high. Computational complexity of the stackednet-1 is low, but its testing accuracy is low. CNN classifier-2 and stackednet-2 exhibit the same testing accuracy for SNR $\geq$ 5 dB and their computational complexities are also around 200 seconds.

ACKNOWLEDGMENT

The authors would like to acknowledge the encouragement given by the management of the Shri Vishnu Education Society and the contributions made by students.

REFERENCES

[1] Y. Kularia, S. Kohli, and P. P. Bhattacharya, "Analysis of acoustic channel characteristics for underwater wireless sensor networks," *Int. J. Comput. Sci., Inf. Technol. Control Eng.*, vol. 3, no. 2, pp. 11–24, Apr. 2016.

[2] H. Dol, M. A. Ainslie, M. Colin, and J. Janmaat, "Simulation of an underwater acoustic communication channel characterized by wind-generated surface waves and bubbles," in *Proc. Meetings Acoust. ECUA*, vol. 17, 2012, Art. no. 070054. [Online]. Available: http://acousticalsociety.org/

[3] G. Ranjani and G. Sadashivappa, "Characterization of underwater acoustic channels," in *Proc. Int. Conf. Appl. Theor. Comput. Commun. Technol. (iCATccT)*, Oct. 2015, pp. 523–528, doi: 10.1109/ICATCCT.2015.7456940.

[4] P. Qarabaqi and M. Stojanovic, "Statistical characterization and computationally efficient modeling of a class of underwater acoustic communication channels," *IEEE J. Ocean. Eng.*, vol. 38, no. 4, pp. 701–717, Oct. 2013, doi: 10.1109/JOE.2013.2278787.

[5] J. Zhang, M. Zhao, J. Zhong, P. Xiao, and T. Yu, "Optimised index modulation for filter bank multicarrier system," *IET Commun.*, vol. 11, no. 4, pp. 459–467, 2017.

[6] A. Viholainen, T. Ihalainen, T. H. Stitz, M. Renfors, and M. Bellanger, "Prototype filter design for filter bank based multicarrier transmission," in *Proc. 17th Eur. Signal Process. Conf.*, 2009, pp. 1359–1363.

[7] J. A. Prakash and G. R. Reddy, "Efficient prototype filter design for filter bank multicarrier (FBMC) system based on ambiguity function analysis of Hermite polynomials," in *Proc. Int. Mutli-Conf. Autom., Comput., Commun., Control Compressed Sens. (iMac4s)*, Mar. 2013, pp. 580–585, doi: 10.1109/iMac4s.2013.6526477.

[8] P. Siohan, C. Siclet, and N. Lacaille, "Analysis and design of OFDM/OQAM systems based on filterbank theory," *IEEE Trans. Signal Process.*, vol. 50, no. 5, pp. 1170–1183, May 2002.

[9] B. Farhang-Boroujeny, "OFDM versus filter bank multicarrier," *IEEE Signal Process. Mag.*, vol. 28, no. 3, pp. 92–112, May 2011, doi: 10.1109/MSP.2011.940267.

[10] J. Dou, Z. Zhang, J. Dang, L. Wu, Y. Wei, and C. Sun, "Properties and achievable data rate of a cyclic prefix based imperfect reconstruction filter bank multiple access system," *IET Commun.*, vol. 10, no. 17, pp. 2427–2434, Nov. 2016.

[11] H. Jamal and D. W. Matolak, "Dual-polarization FBMC for improved performance in wireless communication systems," *IEEE Trans. Veh. Technol.*, vol. 68, no. 1, pp. 349–358, Jan. 2019, doi: 10.1109/TVT.2018.2879573.

[12] R. Nissel and M. Rupp, "OFDM and FBMC-OQAM in doubly-selective channels: Calculating the bit error probability," *IEEE Commun. Lett.*, vol. 21, no. 6, pp. 1297–1300, Jun. 2017, doi: 10.1109/LCOMM.2017.2677941.

[13] B. Farhang-Boroujeny, "Filter bank multicarrier modulation: A waveform candidate for 5G and beyond," *Adv. Electr. Eng.*, vol. 2014, pp. 1–25, Dec. 2014, doi: 10.1155/2014/482805.

[14] C. Sexton, Q. Bodinier, A. Farhang, N. Marchetti, F. Bader, and L. A. DaSilva, "Enabling asynchronous machine-type D2D communication using multiple waveforms in 5G," *IEEE Internet Things J.*, vol. 5, no. 2, pp. 1307–1322, Apr. 2018, doi: 10.1109/JIOT.2018.2806184.

[15] C. Sexton, Q. Bodinier, A. Farhang, N. Marchetti, F. Bader, and L. A. DaSilva, "Coexistence of OFDM and FBMC for underlay D2D communication in 5G networks," in *Proc. IEEE Globecom Workshops (GC Wkshps)*, Dec. 2016, pp. 1–7, doi: 10.1109/GLOCOMW.2016.7848863.

[16] H. Zhang, D. Le Ruyet, D. Roviras, Y. Medjahdi, and H. Sun, "Spectral efficiency comparison of OFDM/FBMC for uplink cognitive radio networks," *EURASIP J. Adv. Signal Process.*, vol. 2010, no. 1, pp. 1–11, Dec. 2010.

[17] P. K. Remvik and N. Holte, "Carrier frequency offset robustness for OFDM systems with different pulse shaping filters," in *Proc. IEEE Global Telecommun. Conf. (GLOBECOM)*, vol. 1, Nov. 1997, pp. 11–15.

[18] T. Fusco, A. Petrella, and M. Tanda, "Sensitivity of multi-user filter-bank multicarrier systems to synchronization errors," in *Proc. 3rd Int. Symp. Commun., Control Signal Process.*, Saint Julian's, Malta, Mar. 2008, pp. 393–398.

[19] T. H. Stitz, T. Ihalainen, and M. Renfors, "Practical issues in frequency domain synchronization for filter bank based multicarrier transmission," in *Proc. 3rd Int. Symp. Commun., Control Signal Process.*, Mar. 2008, pp. 411–416, doi: 10.1109/ISCCSP.2008.4537260.

[20] H. S. Sourck, Y. Wu, J. W. M. Bergmans, S. Sadri, and B. Farhang-Boroujeny, "Effect of carrier frequency offset on offset QAM multicarrier filter bank systems over frequency-selective channels," in *Proc. IEEE Wireless Commun. Netw. Conf.*, Apr. 2010, pp. 1–6.

[21] Y. Medjahdi, M. Terre, D. Le Ruyet, D. Roviras, and A. Dziri, "The impact of timing synchronization errors on the performance of OFDM/FBMC systems," in *Proc. IEEE Int. Conf. Commun. (ICC)*, Jun. 2011, pp. 1–5.

[22] H. Lin, M. Gharba, and P. Siohan, "Impact of time and carrier frequency offsets on the FBMC/OQAM modulation scheme," *Signal Process.*, vol. 102, pp. 151–162, Sep. 2014.

[23] W. Chung, B. Kim, M. Choi, H. Nam, H. Yu, S. Choi, and D. Hong, "Synchronization error in QAM-based FBMC system," in *Proc. IEEE Mil. Commun. Conf.*, Oct. 2014, pp. 699–705.

[24] J. Louveaux, M. Fuhrwerk, D. Mattera, M. Renfors, and M. Tanda, "FBMC synchronization techniques," in *Orthogonal Waveforms and Filter Banks for Future Communication Systems*, 2017, pp. 221–255, doi: 10.1016/b978-0-12-810384-5.00010-4.

[25] D. Mattera, M. Tanda, and M. Bellanger, "CFO sensitivity and efficient joint estimation for the short filter multicarrier system FBMC-PAM," *Signal Process.*, vol. 164, pp. 10–19, 2019, doi: 10.1016/j.sigpro.2019.05.031.

[26] T. Fusco and M. Tanda, "Blind frequency-offset estimation for OFDM/OQAM systems," *IEEE Trans. Signal Process.*, vol. 55, no. 5, pp. 1828–1838, May 2007.

[27] D. Mattera and M. Tanda, "Blind symbol timing and CFO estimation for OFDM/OQAM systems," *IEEE Trans. Wireless Commun.*, vol. 12, no. 1, pp. 268–277, Jan. 2013, doi: 10.1109/TWC.2012.121112.120296.

[28] T. Fusco, A. Petrella, and M. Tanda, "Non-data-aided carrier-frequency offset estimation for pulse-shaping OFDM/OQAM systems," *Signal Process.*, vol. 88, no. 8, pp. 1958–1970, Aug. 2008.

[29] T. Fusco, A. Petrella, and M. Tanda, "Data-aided symbol timing and CFO synchronization for filter bank multicarrier systems," *IEEE Trans. Wireless Commun.*, vol. 8, no. 5, pp. 2705–2715, May 2009, doi: 10.1109/TWC.2009.080860.

[30] Y. Zeng and M. W. Chia, "Joint time-frequency synchronization and channel estimation for FBMC," in *Proc. IEEE 25th Annu. Int. Symp. Pers., Indoor, Mobile Radio Commun. (PIMRC)*, Sep. 2014, pp. 438–442, doi: 10.1109/PIMRC.2014.7136205.

[31] B. Seo, D. Sim, T. Lee, and C. Lee, "Efficient time synchronization method with adaptive resource configuration for FBMC systems," *IEEE Trans. Commun.*, vol. 68, no. 9, pp. 5563–5574, Sep. 2020.

[32] T. Stitz, T. Ihalainen, A. Viholainen, and M. Renfors, "Pilot-based synchronization and equalization in filter bank multicarrier communications," *EURASIP J. Adv. Signal Process.*, vol. 2010, no. 1, Dec. 2010, Art. no. 741429.

[33] W. Chung, C. Kim, S. Choi, and D. Hong, "Synchronization sequence design for FBMC/OQAM systems," *IEEE Trans. Wireless Commun.*, vol. 15, no. 10, pp. 7199–7211, Oct. 2016, doi: 10.1109/TWC.2016.2598809.

[34] J. Hu, Y. Chen, and Q. Huang, "Synchronization of FBMC based on CAZAC sequence," in *Proc. IEEE 2nd Int. Conf. Softw. Eng. Service Sci.*, Jul. 2011, pp. 116–119, doi: 10.1109/ICSESS.2011.5982268.

[35] J. Louveaux, M. Fuhrwerk, D. Mattera, M. Renfors, and M. Tanda, "FBMC synchronization techniques," in *Orthogonal Waveforms and Filter Banks for Future Communication Systems*, 2017, pp. 221–255, doi: 10.1016/b978-0-12-810384-5.00010-4.

[36] T. Stitz *et al.*, "Pilot-based synchronization and equalization in filter bank multicarrier communications," *EURASIP J. Adv. Signal Process.*, vol. 2010, no. 741429, pp. 1–11, 2010, Art. no. 9, doi: 10.1155/2010/741429.

[37] H. Cho and X. Ma, "Generalized synchronization algorithms for FBMC-OQAM systems," *IEEE Trans. Veh. Technol.*, vol. 67, no. 10, pp. 9764–9774, Oct. 2018, doi: 10.1109/TVT.2018.2864304.

[38] C. Thein, M. Schellmann, and J. Peissig, "Analysis of frequency domain frame detection and synchronization in OQAM-OFDM systems," *EURASIP J. Adv. Signal Process.*, vol. 2014, no. 1, p. 83, Dec. 2014.

[39] W. Chung, B. Kim, I. Jung, J. Kim, H. Yu, S. Choi, and D. Hong, "Synchronization method for FBMC systems," in *Proc. IEEE 86th Veh. Technol. Conf. (VTC-Fall)*, Sep. 2017, pp. 1–5, doi: 10.1109/VTC-Fall.2017.8287939.

[40] M. K. Al-Haddad and H. T. Ziboon, "Joint carrier frequency and symbol timing synchronization sequence for FBMC based system," *Phys. Commun.*, vol. 42, Oct. 2020, Art. no. 101165, doi: 10.1016/j.phycom.2020.101165.

[41] K. Pushpa, C. N. Kishore, and Y. Yoganandam, "A new method for frame synchronization in OFDMA mode of WMAN," in *Proc. IEEE Region 10 Conf.*, Nov. 2008, pp. 1–6, doi: 10.1109/TENCON.2008.4766711.

[42] L. Li, L. Ding, Y. Wang, P. Wu, and J. Zhang, "Joint symbol timing and channel estimation for FBMC-OQAM systems," *IEEE Access*, vol. 7, pp. 131326–131337, 2019, doi: 10.1109/ACCESS.2019.2939383.

[43] H. Zheng, J. Sun, X. Zhang, C. Zhang, and Y. Liu, "Timing synchronization algorithm of FBMC system based on phase-weighted conjugate symmetric structure," in *Proc. 6th Int. Conf. Intell. Comput. Signal Process. (ICSP)*, Apr. 2021, pp. 337–341, doi: 10.1109/ICSP51882.2021.9408802.

[44] L. Li, Y. Wang, and L. Ding, "Time synchronization sequence with weighted conjugate symmetry property for FBMC-OQAM systems," in *Proc. IEEE Mil. Commun. Conf. (MILCOM)*, Oct. 2018, pp. 1–6, doi: 10.1109/MILCOM.2018.8599831.

[45] X. Liu, D. Qu, and T. Jiang, "A new data-aided synchronization scheme for FBMC-OQAM systems," in *Proc. 8th Int. Conf. Wireless Commun. Signal Process. (WCSP)*, Oct. 2016, pp. 1–5, doi: 10.1109/WCSP.2016.7752565.

[46] S. Ma, Y. He, G. Cui, W. Wang, and H. Wang, "Joint Doppler frequency and rate estimation in FBMC-OQAM system," in *Proc. 8th Int. Conf. Wireless Commun. Signal Process. (WCSP)*, 2016, pp. 1–5, doi: 10.1109/WCSP.2016.7752650.

[47] A. Y. Kibangou, C. Siclet, and L. Ros, "Joint channel and Doppler estimation for multicarrier underwater communications," in *Proc. IEEE Int. Conf. Acoust., Speech Signal Process.*, Mar. 2010, pp. 5630–5633, doi: 10.1109/ICASSP.2010.5495258.

[48] R. J. Vaccaro, "The past, present, and the future of underwater acoustic signal processing," *IEEE Signal Process. Mag.*, vol. 15, no. 4, pp. 21–51, Jul. 1998, doi: 10.1109/79.689583.

[49] A. C. Singer, J. K. Nelson, and S. S. Kozat, "Signal processing for underwater acoustic communications," *IEEE Commun. Mag.*, vol. 47, no. 1, pp. 90–96, Jan. 2009, doi: 10.1109/MCOM.2009.4752683.

[50] B. Li, S. Zhou, M. Stojanovic, L. Freitag, and P. Willett, "Multicarrier communication over underwater acoustic channels with nonuniform Doppler shifts," *IEEE J. Ocean. Eng.*, vol. 33, no. 2, pp. 198–209, Apr. 2008, doi: 10.1109/JOE.2008.920471.

[51] B. Li, S. Zhou, M. Stojanovic, L. Freitag, and P. Willett, "Non-uniform Doppler compensation for zero-padded OFDM over fast-varying underwater acoustic channels," in *Proc. OCEANS-Eur.*, Jun. 2007, pp. 1–6, doi: 10.1109/OCEANSE.2007.4302478.

[52] F. Wei, X. Xiaomei, Z. Lan, and C. Yougan, "A frame synchronization method for underwater acoustic communication on mobile platform," in *Proc. Int. Conf. Image Anal. Signal Process.*, 2010, pp. 518–522, doi: 10.1109/IASP.2010.5476062.

[53] J. Gomes and M. Stojanovic, "Performance analysis of filtered multitone modulation systems for underwater communication," in *Proc. OCEANS*, Oct. 2009, pp. 1–9, doi: 10.23919/OCEANS.2009.5422175.

[54] P. Amini, R.-R. Chen, and B. Farhang-Boroujeny, "Filterbank multicarrier for underwater communications," in *Proc. 49th Annu. Allerton Conf. Commun., Control, Comput. (Allerton)*, Sep. 2011, pp. 639–646.

[55] P. Amini, R.-R. Chen, and B. Farhang-Boroujeny, "Filterbank multicarrier communications for underwater acoustic channels," *IEEE J. Ocean. Eng.*, vol. 40, no. 1, pp. 115–130, Jan. 2015, doi: 10.1109/JOE.2013.2291139.

[56] M. Xiao, Z. Xin, G. Yizhou, and Z. Qing, "Research on carrier frequency estimation of underwater acoustic communication signals," in *Proc. IEEE Int. Conf. Signal Process., Commun. Comput. (ICSPCC)*, Sep. 2018, pp. 1–4, doi: 10.1109/ICSPCC.2018.8567725.

[57] X. Han, W. Ge, J. Yin, and G. Yu, "Mobile underwater acoustic communication with constantly changing Doppler effect," in *Proc. OCEANS-MTS/IEEE Kobe Techno-Oceans (OTO)*, May 2018, pp. 1–5.

[58] J.-C. Zheng, H.-X. Sun, C.-F. Yang, and J. Liu, "A joint synchronization parameter estimation in fractional Fourier transform orthogonal frequency division multiplexing underwater acoustic communication system," *Adv. Mech. Eng.*, vol. 8, no. 3, 2016, doi: 10.1177/1687814016638310.

[59] N.-N. Huang, H. Lei, and J.-C. Zheng, "A precise synchronization scheme using in the underwater acoustic communication system based on FRFT-OFDM," *Int. J. Signal Process. Syst.*, vol. 3, no. 1, pp. 1–7, 2014.

[60] F. Yuan, Z. Jia, J. Li, and E. Cheng, "STLFM signal based adaptive synchronization for underwater acoustic communications," *IEEE Access*, vol. 7, pp. 28734–28748, 2019, doi: 10.1109/ACCESS.2019.2901357.

[61] E. Sang, X. Xu, G. Qiao, and J. Su, "Study on ZP-OFDM for underwater acoustic communication," in *Proc. Int. Conf. Neural Netw. Signal Process.*, Jun. 2008, pp. 299–302, doi: 10.1109/ICNNSP.2008.4590360.

[62] Z. Ling, L. Xie, and H. Chen, "Joint Doppler scale estimation and timing synchronization in underwater acoustic communications," in *Proc. IEEE Int. Conf. Signal Process., Commun. Comput. (ICSPCC)*, Sep. 2019, pp. 1–6, doi: 10.1109/ICSPCC46631.2019.8960868.

[63] J. A. Sidorkina, R. M. Dyabirov, B. I. Shakhtarin, and A. V. Bychkov, "A proposed chirp synchronisation algorithm based on the fractional Fourier transform for underwater acoustic communication," in *Proc. Syst. Signal Synchronization, Generating Process. Telecommun. (SYNCHROINFO)*, 2019, pp. 1–4, doi: 10.1109/SYNCHROINFO.2019.8814166.

[64] K. A. Balakhonov, B. I. Shakhtarin, and R. M. Dyabiov, "Experimental research of Doppler scale estimation algorithms for OFDM signals in fast time-varying underwater acoustic channel," in *Proc. Systems Signal Synchronization, Generating Process. Telecommun. (SYNCHROINFO)*, 2018, pp. 1–5, doi: 10.1109/SYNCHROINFO.2018.8456952.

[65] B. I. Shakhtarin, K. A. Balakhonov, and K. S. Kalashnikov, "OFDM signal synchronization algorithm in the multipath hydroacoustic channel," *J. Commun. Technol. Electron.*, vol. 63, no. 4, pp. 354–360, Apr. 2018, doi: 10.1134/S1064226918040113.

[66] D. Landstrom, J. M. Arenas, J.-J. van de Beek, P. O. Borjesson, M.-L. Boucheret, and P. Odling, "Time and frequency offset estimation in OFDM systems employing pulse shaping," in *Proc. ICUPC*, vol. 1, Oct. 1997, pp. 279–283.

[67] R. Diamant, A. Feuer, and L. Lampe, "Choosing the right signal: Doppler shift estimation for underwater acoustic signals," in *Proc. 7th ACM Int. Conf. Underwater Netw. Syst.*, 2012, pp. 1–8, doi: 10.1145/2398936.2398971.

[68] S. E. Noel, H. H. Szu, and Y. J. Gohel, "Doppler frequency estimation with wavelets and neural networks," *Proc. SPIE*, vol. 3391, pp. 150–157, Mar. 1998, doi: 10.1117/12.304865.

[69] Y. LeCun, L. Bottou, Y. Bengio, and P. Haffner, "Gradient-based learning applied to document recognition," *Proc. IEEE*, vol. 86, no. 11, pp. 2278–2324, Nov. 1998.

[70] X. Cheng, D. Liu, C. Wang, S. Yan, and Z. Zhu, "Deep learning-based channel estimation and equalization scheme for FBMC/OQAM systems," *IEEE Wireless Commun. Lett.*, vol. 8, no. 3, pp. 881–884, Jun. 2019, doi: 10.1109/LWC.2019.2898437.

[71] V. Ninkovic, A. Valka, D. Dumic, and D. Vukobratovic, "Deep learning-based packet detection and carrier frequency offset estimation in IEEE 802.11ah," *IEEE Access*, vol. 9, pp. 99853–99865, 2021, doi: 10.1109/ACCESS.2021.3096853.

[72] S. Kojima, K. Maruta, Y. Feng, C.-J. Ahn, and V. Tarokh, "CNN-based joint SNR and Doppler shift classification using spectrogram images for adaptive modulation and coding," *IEEE Trans. Commun.*, vol. 69, no. 8, pp. 5152–5167, Aug. 2021, doi: 10.1109/TCOMM.2021.3077565.

[73] S. Ma, Y. He, G. Cui, W. Wang, and H. Wang, "Joint Doppler frequency and rate estimation in FBMC-OQAM system," in *Proc. 8th Int. Conf. Wireless Commun. Signal Process. (WCSP)*, Oct. 2016, pp. 1–5, doi: 10.1109/WCSP.2016.7752650.

[74] S. M. Kay, *Fundamentals of Statistical Signal Processing: Estimation Theory*. Upper Saddle River, NJ, USA: Prentice-Hall, 1993.

[75] D. Sen, S. Chakrabarti, and R. V. R. Kumar, "A technique for frequency synchronization in multi-band OFDM in realistic UWB channel," *Circuits, Syst. Signal Process.*, vol. 28, no. 6, pp. 993–1016, Dec. 2009.

[76] V. Tiwari, C. Pandey, A. Dwivedi, and V. Yadav, "Image classification using deep neural network," in *Proc. 2nd Int. Conf. Adv. Comput., Commun. Control Netw. (ICACCCN)*, Dec. 2020, pp. 730–733, doi: 10.1109/ICACCCN51052.2020.9362804.

[77] S. Khan, H. Rahmani, S. A. A. Shah, and M. Bennamoun, *A Guide to Convolutional Neural Networks for Computer Vision*. San Rafael, CA, USA: Morgan & Claypool, 2018.

[78] B. Henson, J. Li, Y. V. Zakharov, and C. Liu, "Waymark baseband underwater acoustic propagation model," in *Proc. Underwater Commun. Netw. (UComms)*, 2014, pp. 1–5, doi: 10.1109/UComms.2014.7017132.

[79] M. Porter. (2011). *The BELLHOP Manual and User's Guide: Preliminary Draft*. [Online]. Available: http://oalib.hlsresearch.com/Rays/HLS-2010-1.pdf

[80] T. M. Schmidl and D. C. Cox, "Robust frequency and timing synchronization for OFDM," *IEEE Trans. Commun.*, vol. 45, no. 12, pp. 1613–1621, Dec. 1997.

[81] L. Li, Y. Wang, and L. Ding, "Time synchronization sequence with weighted conjugate symmetry property for FBMC-OQAM systems," in *Proc. IEEE Mil. Commun. Conf. (MILCOM)*, Oct. 2018, pp. 1–6, doi: 10.1109/MILCOM.2018.8599831.

[82] M. Liu, G. Liao, N. Zhao, H. Song, and F. Gong, "Data-driven deep learning for signal classification in industrial cognitive radio networks," *IEEE Trans. Ind. Informat.*, vol. 17, no. 5, pp. 3412–3421, May 2021, doi: 10.1109/TII.2020.2985715.

PUSHPA KOTIPALLI received the Ph.D. degree in electronics and communications engineering from Osmania University, Hyderabad, Andhra Pradesh, India, in 2010.

She worked as an Engineer with Astra Microwave Products Ltd., Hyderabad, from 1992 to 1997. Since 1997, she has been working in different engineering colleges, became an Associate Professor, in 2000, and a Professor, in 2007. She is currently working as a Professor with the ECE Department, Shri Vishnu Engineering College for Women, Bhimavaram, Andhra Pradesh. She published two books, three book chapters, and 20 refereed journal articles. She delivered 40 video lectures for SONET, Government of Andhra Pradesh, in 2003, and presented 15 papers in conferences. She completed one DST project and handled one NRB of DRDO project on "Underwater Acoustic Communications." She guided student teams towards completing three IEEE EPICS Projects. She delivered three webinars on underwater acoustic communications. Her research interests include signal processing in RF communications and underwater acoustic communications, and deep learning techniques. She developed "Olfaction Dysfunction Tester" for which her team received the Presidential Award GYTI 2018. She reviewed several articles related to the IEEE Communication Society and Ph.D. thesis. She is a paper setter to different autonomous institutes. She chaired conferences and organized technical fests and workshops. She acted as the Vice-Chair to Women-in-Engineering (WIE) Affinity Group of the IEEE Hyderabad Section, in 2011. As an Organizing Co-Chair, she organized 'IEEE R10 Women In Engineering Congress–WIECON 2012' organized by the Shri Vishnu Engineering College for Women, Bhimavaram, in collaboration with the IEEE Hyderabad Section and IEEE R10, in March 2012.

ADI SURENDRA MOHANRAJU M. received the M.Tech. degree in mobile computing technology, with a focus on signal processing, from the School of Electronics, Devi Ahilya University, in 2013. He is currently employed as an Assistant Professor at the Shri Vishnu Engineering College for Women. He is a Co-Principal Investigator to a DRDO sponsored project funded by the Naval Research Board. His research interests include signal processing for underwater communications, machine learning, and deep learning in signal and image processing.

PRAVEENA VARDHANAPU received the M.Tech. degree in communication systems from the SRKR Engineering College. She is currently working as a Junior Research Fellowship (JRF) for the NRB-DRDO project on "FBMC-Based Underwater Acoustic Communications," with the Department of Electronics and Communication Engineering, Shri Vishnu Engineering College for Women. Her research interests include communications and signal processing.