PCI Classification in 5G-NR: Deep Learning Unravels Synchronization Signal Blocks

Md Rabiul Hossain and Marwan Krunz
Department of Electrical and Computer Engineering, University of Arizona, Tucson, USA
{mrhossain, krunz}@arizona.edu

***Abstract*—Accurate detection of the Physical Cell Identity (PCI) is critical for rapid synchronization and connection establishment in 5G New Radio (5G-NR) systems. This paper introduces a deep learning-based approach for PCI classification, aiming to mitigate the computational complexity associated with traditional methods that rely on decoding the Synchronization Signal Block (SSB). Our approach processes only time-domain baseband samples of the downlink signal, arranged in fixed-length windows. These windows are inputted into a pre-trained Convolutional Neural Network (CNN), which classifies the samples into one of several known PCI values (representing nearby cells) or into an 'other' category (representing all non-nearby cells, as well as windows that do not contain SSB samples). Because PCI-related information is contained only in the SSB symbols of a frame, it is possible for an input window to include no or a few SSB samples. Accordingly, in labeling the training set, we use a threshold $\mathbf{T}_{\text{train}}$ on the fraction of the samples within a window: if $\mathbf{T}_{\text{train}}$ or more of the samples belong to an SSB of cell with a target PCI value, the true label for that window is set to that PCI value; otherwise, it is set to 'other.' A separate threshold $\mathbf{T}_{\text{test}}$ is used for labeling the test windows. We also study another labeling mechanism whereby only samples of the third OFDM symbol in an SSB (which contains the Secondary Synchronization Signal) is used to determine the label. Our analysis considers two commonly used SSB formats that correspond to 15 and 30 kHz subcarrier spacings, respectively. Extensive simulations are conducted which reveal that the proposed classifier can reliably (above 98%) identify the PCI value of a captured signal even under Signal-to-Noise Ratio (SNR) values as low as -10 dB. This performance comes with a significant reduction in computational complexity as it bypasses the need for traditional SSB decoding procedures used for PCI estimation in 5G networks.**

***Index Terms*—Physical cell identity, 5G-NR, deep learning, synchronization signal block, PSS, SSS.**

I. Introduction

THE advent of 5G New Radio (5G-NR) heralds a new era in wireless communication, offering unparalleled data speeds, minimal latency, and extensive machine-type communications [1], [2]. The synchronization process is an integral component of 5G systems, as it ensures seamless initial discovery and reliable communication between User Equipment (UE) and Base Stations (BS) [3]–[6]. It is used by the UE to acquire the Timing Offset (TO) and the Carrier Frequency Offset (CFO), as well as retrieving the Physical Cell Identifier (PCI) [4], [7]. The PCI uniquely identifies a cell within the network, enabling the UE to connect to the appropriate cell. Precise identification of PCI is crucial for the synchronization mechanism, significantly influencing network performance, e.g., reducing error probabilities during initial access, minimizing interference, and facilitating handovers, thereby enhancing the overall user experience [8].

Machine Learning (ML) plays an important role in 5G-NR, enabling dynamic spectrum sharing, performance enhancement in coexistence scenarios, and fostering advancements in areas like adaptive beamforming [9], resource allocation, interference management, predictive analytics, and network optimization [10], [11]. This paper exploits ML for estimating the PCI, bypassing the conventional approaches for synchronization which rely on signal decoding. This strategy leads to a significant reduction in complexity and opens the door for integrating ML-based PCI detection in various use cases such as smart repeaters, SSB-based passive coherent location RADAR systems, UAV detection, and others.

Cell search in the 5G system requires decoding the Synchronization Signal Blocks (SSBs) [12], [13]. Using the Global Synchronization Channel Number (GSCN) within a specified frequency band, a UE executes a cell search by detecting both the Primary Synchronization Signal (PSS) and the Secondary Synchronization Signal (SSS) [14]. SSB consists of the PSS and SSS along with the Physical Broadcast Channel (PBCH) [6], [15], [16]. PSS in 5G-NR utilizes one of three pre-defined m-sequences, each with a length of 127, transmitted on the first OFDM symbol of the SSB. It assists in determining the physical layer identity groups and facilitates initial synchronization and symbol-level time alignment [6]. The SSS is used to identify the specific cell within the identified PSS group. There are 336 possible sequences. Combination of these two signals (PSS identifying the broader group and SSS pinpointing the exact cell within that group) enables precise calculation of the PCI [17]. This calculation, integral to the UE's ability to differentiate between various network cells, is essential to maintain effective communication and network coherence in a dynamic environment of 5G connectivity.

Typically, PCI detection occurs in tandem with the synchronization process. The algorithm devised for PCI detection, detailed in [18], relies on estimating the CFO and SSS sequence. This estimation is achieved by finding the maximum correlation between the received signal (after frequency offset compensation) and all possible SSS sequences. In [6], a timing synchronization algorithm based on PSS was proposed which incorporates an enhanced coarse synchronization method using Fourier transform with a fine synchronization approach based

on the triple auto-correlation algorithm. However, this comes at the cost of higher computational complexity, twice as much as the conventional approach. Improved PCI detection probability through precoding vector switching transmit diversity and receive diversity was proposed in [19]. It was shown that receive diversity with up to four antennas significantly improves PSS and SSS sequence detection, and boosts the PCI detection probability in high Signal-to-Noise Ratio (SNR) regions. Another approach [5] utilized quasi-omni pseudo-random sounding beams and a novel signal processing algorithm for initial cell discovery, synchronization, and fine-resolution beam training. This approach relied on energy detection for initial access, outperforming directional initial access in the cell discovery rate. In [20], a deep-learning-based initial access method was introduced for millimeter-wave MIMO systems. It employed a Convolutional Neural Network (CNN) for enhanced PSS detection. However, the approach requires preprocessing the received signal using Fast Fourier Transform, followed by network training, resulting in higher computational complexity than conventional approaches.

Traditional PCI estimation processes rely predominantly on SSB decoding, which involves high computational complexity due to intensive correlation operations on the receiving end. In this paper, we introduce a CNN and Convolutional Long-Short-Term Memory (ConvLSTM) architecture for PCI detection in a 5G-NR network using only time-domain samples, i.e., prior to FFT, bypassing the SSB decoding stage. It substantially reduces computational complexity compared to traditional methods and effectively retrieves the PCI value from the received signal, even in the presence of a larger frequency offset. Given LTE's simpler synchronization architecture, our PCI estimation approach is also well-suited for LTE systems. The main contributions of the paper are as follows:

- We create a comprehensive dataset of 5G-NR signal waveforms using MATLAB's 'Communication Toolbox' and '5G Toolbox'. The dataset features 12 distinct PCIs (99 to 110) with varying sector and group IDs and two SSB formats (Case-A and Case-B).
- Two deep learning classifiers (CNN and ConvLSTM) are developed for PCI classification. Focusing on the less complex yet efficient CNN architecture, we achieve strong PCI classification performance across SNR values ranging from -10 to 0 dB.
- We continue to evaluate the trained model's performance with the received signal incorporating CFO, showcasing its capability to handle real-world signal imperfections.
- Finally, our deep learning-based approach is compared with the traditional methods that decode the SSB from the baseband signals, revealing a significant advantage in terms of computational complexity with the CNN approach.

The rest of the paper is organized as follows. In Section II, we start with the background of the synchronization process in 5G systems. We also explain the conventional approach of PCI detection. Section III introduces the CNN model, dataset generation, and proposed PCI estimation procedure. Simulation-based performance evaluation of the proposed approach is provided in Section IV. Finally, Section V concludes this paper.

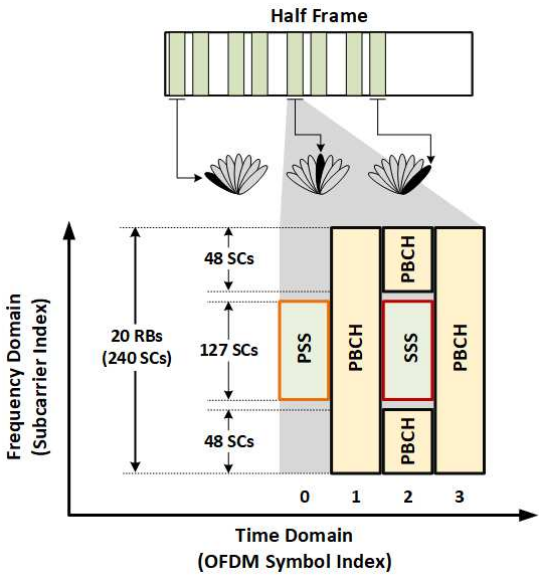


Fig. 1. Structure of an SSB in 5G-NR.

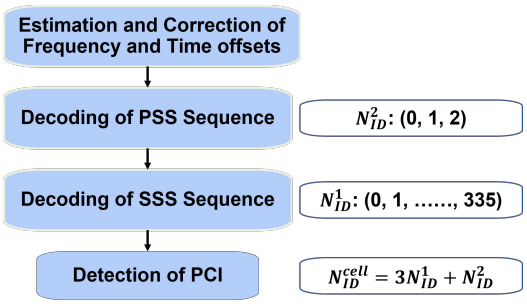


Fig. 2. Classical procedure for PCI estimation.

II. Preliminaries: Synchronization Process in 5G-NR

A. Synchronization Signal Block (SSB)

The synchronization procedure relies on periodic transmission of SSBs by a gNB (gNodeB). In contrast to LTE, the gNB transmits SSB in intermittent bursts with periodicities of 5, 10, 20, 40, 80, and 160 ms, depending on the specific configuration of the network (20ms is the default period). In 5G-NR, a synchronization signal burst (SS burst) is a set of SSBs within a beam-sweep. Each SSB within an SS burst is assigned a distinctive identifier known as the SSB index, which is transmitted through a dedicated beam in a specific direction in the time domain. A UE may only detect and read a single SSB from a particular beam direction without considering other SSBs transmitted from the same cell. Fig. 1 illustrates the structure of an SSB burst transmission pattern. In the time domain, an SSB consists of four OFDM symbols that are each mapped to 240 contiguous subcarriers in the frequency domain. Synchronization signals (PSS and SSS) are in the first and

TABLE I
SSB TIME-DOMAIN RESOURCE ALLOCATION

Indices of OFDM starting symbols of the candidate SSBs						
Frequency Range - 1				**Frequency Range - 2**		
SCS of SSB	**OFDM Symbols Position**	**($f_c \leq 3$ GHz)**	**($3 \leq f_c \leq 6$) GHz**	**SCS of SSB**	**OFDM Symbols Position**	**($f_c \geq 6$ GHz)**
Case-A 15 kHz	$\{2, 8\}$+14n	n = 0, 1 $\{2, 8, 16, 22\}$	n = 0, 1, 2, 3 $\{2, 8, 16, 22, 30, 36, 44, 50\}$	**Case-D 120 kHz**	$\{4, 8, 16, 20\}$+28n	n = 0, 1, 2, 3, 5, 6, 7, 8, 10, 11, 12, 13, 15, 16, 17, 18 $\{4, 8, 16, ...512, 520, 524\}$
Case-B 30 kHz	$\{4, 8, 16, 20\} + 28$n	n = 0 $\{4, 8, 16, 20\}$	n = 0, 1 $\{4, 8, 16, 20, 32, 36, 44, 48\}$	**Case-E 240 kHz**	$\{8, 12, 16, 20,$ $32, 36, 40, 44\}$+56n	n = 0, 1, 2, 3, 5, 6, 7, 8 $\{8, 12, 16, ...484, 488, 492\}$
Case-C 30 kHz	$\{2, 8\}$+14n	n = 0, 1 $\{2, 8, 16, 22\}$	n = 0, 1, 2, 3 $\{2, 8, 16, 22, 30, 36, 44, 50\}$			

third OFDM symbols. The Sub-Carrier Spacing (SCS) can be different than the SCS of a carrier for faster synchronization [16]. Different SSB transmission patterns, based on the SCS of SSB, are observed in time-domain, with flexibility in their frequency domain placement. Table I illustrates all possible SSB burst patterns, specifying the number of SSBs within a burst and the starting OFDM symbol position of these SSBs for FR1 and FR2 of 5G-NR.

B. Primary and Secondary Synchronization Signals

PSS and SSS play crucial roles in synchronizing a UE to a base station. The PSS is used to identify frame boundary and detect the cell ID sector, $N_{\text{ID}}^{(2)}$. It is composed of one of three possible 127-symbol m-sequences [17]. One of these sequences is strategically positioned on the first symbol of each SSB, spanning 127 subcarriers. The SSS is selected from 336 distinct 127-symbol gold sequences, and is positioned on the third symbol of each SSB. The structure of these 336 gold sequences is provided in [17].

5G-NR supports 1008 unique Cell IDs, systematically organized into 336 groups. Each group is defined by a 'Cell ID Group' ($N_{\text{ID}}^{(1)}$, ranging from 0 to 335) and further divided into three sectors, identified by the 'Cell ID Sector' ($N_{\text{ID}}^{(2)}$, taking values 0, 1, or 2). UEs discern $N_{\text{ID}}^{(2)}$ from the PSS and $N_{\text{ID}}^{(1)}$ from the SSS, leading to the calculation of the primary cell (PCI) as: $N_{\text{ID}}^{\text{cell}} = 3N_{\text{ID}}^{(1)} + N_{\text{ID}}^{(2)}$. This identification approach bears similarity to that of 4G networks. However, it differs from 4G in the specific attributes and the generation patterns of the synchronization signal sequences. These differences are pivotal in enhancing the effectiveness of the 5G network, particularly in terms of the speed and accuracy of cell detection.

C. Classical Synchronization Procedure and PCI Estimation

In 5G-NR, the PCI detection process involves several key steps, as shown in Fig. 2. Initially, the receiver identifies the OFDM symbol timing and extracts the PSS sequence through cross-correlation of the received signal and each of the three reference PSS sequences. Following frequency offset estimation and correction, the receiver performs OFDM demodulation and extracts the resource elements corresponding to the SSS from the received grid. It then correlates these elements with each possible locally generated SSS sequence to determine $N_{\text{ID}}^{(1)}$. The indices of the strongest PSS and SSS sequences are combined to determine the PCI, which is essential for the demodulation reference signal (DM-RS) and PBCH processing.

TABLE II
PARAMETERS USED TO GENERATE 5G TRACES

Parameters	**Value**
Channel Bandwidth	20 MHz
Frequency Range	FR1
Numerology	0, 1
# of Subframes	10
# of Resource Block	106
SCS of SSB	Case-A & Case-B
PCI ID	99, 100, ..., 110
SSB Burst Periodicity (Training)	5 msec
SSB Burst Periodicity (Testing)	20 msec
SNR (dB)	$0, -2, -4, -6, -8, -10$

III. PROPOSED PCI DETECTION APPROACH

A. Dataset Description

The 5G-NR signal waveforms are generated using MATLAB's 'Communication Toolbox' and '5G Toolbox'. These toolboxes enable the specification of key signal attributes such as baseband I and Q values, channel bandwidth, modulation, SCS, and allocated resource blocks. Leveraging these defined features, we simulate diverse waveforms for 5G-NR systems across various parameter configurations outlined in Table II in accordance with standard specifications. Amid the array of potential features, particular emphasis is placed on the baseband I/Q samples at the receiver, augmented with noise, serving as the input for the classifier. These I/Q samples, accessible prior to signal decoding, offer a good representation of the actual waveform.

In this study, we explore the impact of two distinct SCS formats, 'Case-A' and 'Case-B,' within the context of SSB. Within each case, we investigate two variants of SSB burst periodicities, specifically 5ms and 20ms (default periodicity). By leveraging these parameters, we generate traces for 12 distinct PCIs ranging from 99 to 110, incorporating different sector and group IDs. Given the pool of 1008 possible PCIs resulting from PSS and SSS combinations, we strategically aim to curate this subset, which encapsulates the full spectrum of possible combinations, thus enabling a comprehensive assessment of the resilience of our PCI detection methodology.

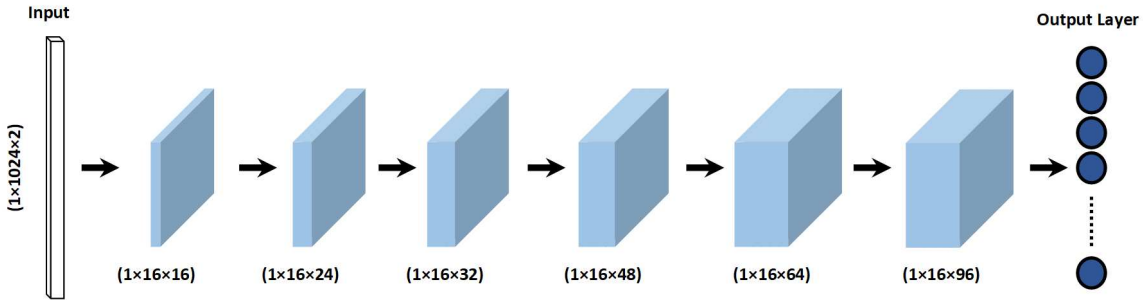


Fig. 3. Architecture of the proposed CNN model for PCI classification.

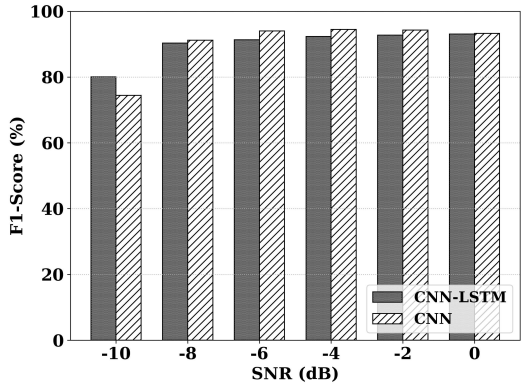


Fig. 4. F1-score comparison between CNN and ConvLSTM at different SNR values considering 'Case-A' SSB format.

TABLE III
HYPER-PARAMETERS FOR PROPOSED PCI CLASSIFIER

Parameters	Value
Max Training Epochs	10
Initial Learning Rate	0.2
Learn Rate Schedule	Piecewise
Learn Rate Drop-Factor	0.8
Learn Rate Drop Period	7
Batch Size	128
Filter Size	[1, 16]
Optimizer	Stochastic gradient descent with momentum
Activation Function	ReLU
Loss	Categorical Cross-entropy

Each PCI was meticulously analyzed with respect to a specific SSB Burst periodicity and one of the SCS formats, resulting in the generation of 300 individual traces per combination. Consequently, our research endeavors yield a comprehensive dataset comprising $14,400$ traces with 10 frames in each, capturing the intricacies and variations across the investigated parameters. The dataset exhibits diversity through the incorporation of varying payloads across the traces. Channel randomization, utilizing the AWGN channel, is applied to each of these traces. The traces for each PCI are further categorized into 6-SNR values after performing channel realizations. Each trace realization comprises $3,072,000$ I/Q pairs, with each OFDM symbol corresponding to $2,192$ I/Q pairs for the Case-A format of SSB and $1,097$ I/Q pairs for the Case-B configuration.

B. Classifier Design

Recent advances in deep learning demonstrated the efficacy of CNNs and ConvLSTM networks in capturing complex data patterns in wireless systems [20]–[23]. This section provides a comparative analysis of these two architectures, focusing on their application in PCI classification. CNNs are known for their effectiveness in processing spatial data through hierarchical feature extraction. Their architecture, characterized by convolutional layers, activation functions, pooling layers, and fully connected layers, is well-suited for tasks that involve identifying patterns and features in images or other grid-like structures. Our proposed CNN model, shown in Fig. 3, consists of six convolutional layers, each coupled with batch normalization, ReLU activation, and max pooling. Noteworthy is the replacement of the traditional max pooling layer with an average pooling layer in the final convolutional layer, thereby introducing a nuanced method for feature aggregation. SoftMax activation function is used in the output layer of the model, thereby giving probabilistic class prediction. The hyper-parameters of this CNN classifier are summarized in Table III.

On the other hand, ConvLSTM networks extend CNNs' capabilities by incorporating LSTM units to address both spatial and temporal dependencies, making them particularly powerful for sequential data where temporal dynamics are important to capture. The proposed architecture of ConvLSTM used in our work integrates the same CNN structure previously discussed but adds an LSTM layer to enhance the model's ability to handle classification tasks. Following the convolutional block, the ConvLSTM model employs average pooling to aggregate global features, which are then flattened. An LSTM layer with 256 hidden units is subsequently used to capture temporal dependencies within these features. Combining the strengths of convolutional and sequential processing, the architecture achieves robust PCI classification performance.

Fig. 4 presents an F1-score comparison between CNN and ConvLSTM models for PCI classification across different SNR (-10 to 0 dB) levels. Surprisingly, the two models exhibit comparable performance except at SNR = -10 dB, where ConvLSTM shows a marginally superior performance. The CNN model offers a clear advantage in computational efficiency

over the ConvLSTM due to its simpler architecture. Given our emphasis on minimizing the computational complexity of traditional PCI detection methods, CNN's efficiency and straightforward design make it the preferred choice for further analysis.

C. Window Labeling Strategy

To perform PCI classification, we arrange the time-domain baseband I/Q samples of the downlink signal in fixed-length windows. During the training process, we allow the windows to overlap by adjusting the stride of the sliding window. In the window labeling process, we explore two approaches. The first approach labels a window based on the percentages of samples that fully or partially belong to SSB, spanning four OFDM symbols. The second approach focuses on a specific segment of the SSB, considering solely the OFDM symbol that contains the SSS. Window labeling depends on a threshold value: T_{train} and T_{test} for the training and testing phase, respectively. This threshold indicates the percentage of I/Q pairs associated with the SSB or SSS for the two approaches, respectively. If an I/Q sample within a window corresponds to the SSB/SSS of a particular PCI and the percentage of such samples is greater than or equal to T_{train}, the window is labeled with that PCI value. Otherwise, it is labeled as 'Other'. During the training phase, we set $T_{\text{train}} = 100\%$, and our observations indicate that the model exhibits accurate window classification performance during testing for $T_{\text{test}} = 50\%$ and 100%. As mentioned above, two types of SSB burst periodicity, 5 ms and 20 ms (default periodicity), are considered. During the training phase, we focus solely on the 5 ms periodicity due to the higher concentration of SS blocks in each trace. However, we maintain the default 20 ms SSB burst periodicity during testing.

IV. Results and Discussion

A. Impact of Model Depth

This segment presents the rationale behind our selection of a six-layer CNN architecture. Our analysis assesses the model's efficacy by selectively focusing on a subset of training and test windows, where all samples are uniquely associated with the third OFDM symbol of an SSB ('Case-A'). The outcomes are succinctly illustrated in Fig. 5, wherein we deploy CNN configurations of varying depths. Our analysis reveals that CNN architecture comprising six layers significantly outperforms alternative configurations and strikes a balance between model complexity and computational efficiency. We optimize the hyper-parameters specifically for the 6-layered CNN model, which could be a plausible explanation for the observed superior performance. The consistent outperformance of precision, recall, and F1 scores, particularly for lower SNR, reinforces our selection of the 6-layered CNN model.

B. SNR Impact on Proposed Model's Performance

We explore two distinct strategies for training the model in the context of SNR selection using the 'Case-A' SSB format. Firstly, our training dataset encompasses a range of

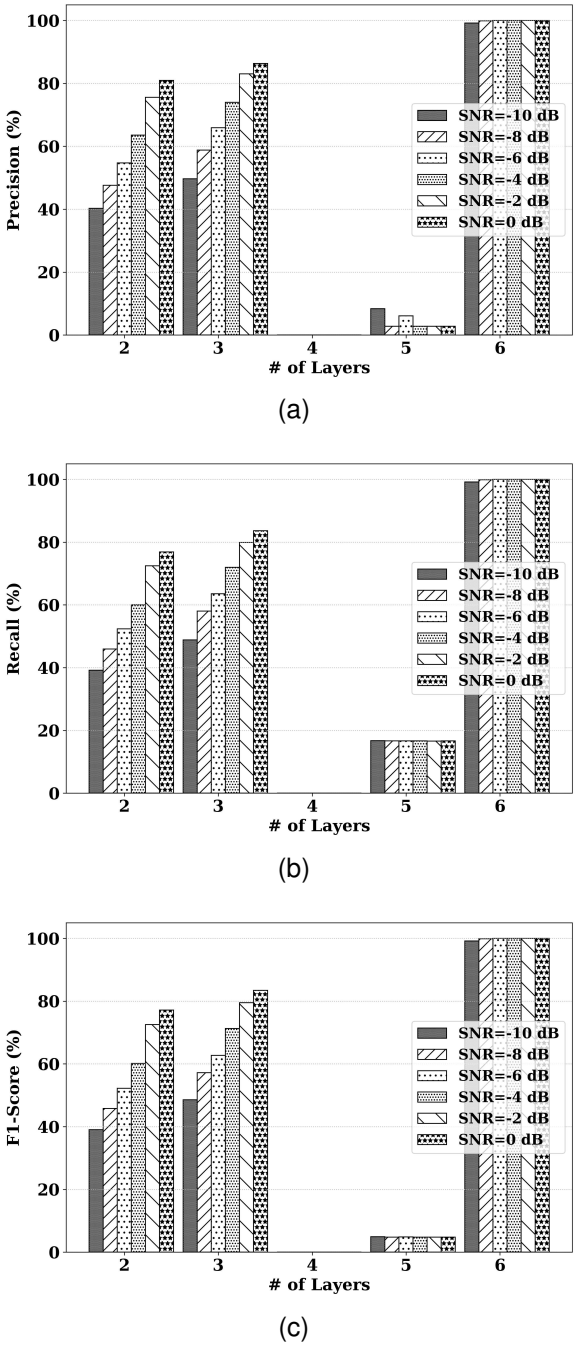


Fig. 5. Impact of convolutional layers on precision, recall, and F1-score for PCI classification considering different SNRs.

SNR values, where the SNR values (in dB) are in the set, $S = \{-10,\ -8,\ -6,\ -4,\ -2,\ 0\}$, providing a comprehensive representation of noise levels. Subsequently, our second approach solely incorporates traces corresponding to the lowest SNR, -10 dB. Both the training and testing datasets include windows containing SSB samples, as well as those without SSB samples, labeled as 'Other.' During model evaluation for both cases, testing traces contain the entire range of SNR values within our predefined set, S. As shown in Table IV, training with a dataset covering an SNR range from -10

TABLE IV
PCI DETECTION ACCURACY AT DIFFERENT SNRS

Inference SNR (dB)	Without Decision Threshold						With Decision Threshold ($\alpha = 95\%$)							
	All SNRs (dB)			Lowest SNR (dB)			All SNRs (dB)				Lowest SNR (dB)			
	Precision (%)	Recall (%)	F1-Score (%)	Precision (%)	Recall (%)	F1-Score (%)	Precision (%)	Recall (%)	F1-Score (%)	Window Removal (%)	Precision (%)	Recall (%)	F1-Score (%)	Window Removal (%)
0	100.00	100.00	100.00	74.44	99.90	84.89	100.00	100.00	100.00	0.21	95.92	99.99	97.87	1.45
−2	99.86	100.00	99.93	79.07	99.92	87.90	100.00	100.00	100.00	0.45	98.61	100.00	99.29	1.62
−4	97.74	99.56	98.63	81.62	99.86	89.60	99.85	100.00	99.93	3.91	99.43	100.00	99.71	2.24
−6	93.75	95.27	94.44	81.66	99.29	89.32	99.91	99.43	99.67	19.51	99.62	100.00	99.81	8.24
−8	81.47	75.24	77.87	79.21	87.57	82.97	98.64	89.46	93.72	43.86	99.44	96.71	98.05	32.31
−10	66.10	47.86	53.97	71.25	61.00	65.16	96.15	51.01	64.28	58.05	97.04	70.52	80.92	56.58

to 0 dB generally yields improved performance compared to training solely with the dataset of SNR = −10 dB, regardless of the decision threshold α. This is because the wider SNR range provides a more nuanced representation of varying noise conditions, enhancing the model's ability to generalize across diverse scenarios. However, this expanded range also increases training time and computational demands due to the larger dataset. Our findings indicate that, without the decision threshold, the first approach in which training traces cover the SNR range from 0 to −10 dB performs better. Considering the decision threshold, $\alpha = 95\%$, the CNN model trained exclusively on traces of SNR=−10 dB delivers performance on par with models trained across all SNRs. The recall value indicates that all the labeled windows are correctly identified in both cases except at SNR=−10 dB. When we train our model solely with the lowest SNR, it tends to misclassify windows labeled as 'Other,' especially windows from the traces with high SNR. However, when we set the α as 95%, the proposed approach performs efficiently across a range of SNRs, including extremely low SNR levels. When the decision threshold is applied, a significant proportion of windows labeled with their respective PCIs are excluded: 32.31% and 56.58% for SNRs of −8 dB and −10 dB, respectively which is less than the alternate approach. Our trained model achieves perfect classification accuracy for SNR values above 0 dB. Therefore, we focus our analysis on the more challenging SNR range from −10 to 0 dB. Consequently, we advocate focusing on training data comprising solely the lowest SNR, as it yields comparable performance over large datasets, effectively balancing performance with limited resources.

C. Comparison of Complete SSB vs. Partial SSB

In this section, we evaluate the CNN model under two different procedures for labeling the input windows: The first method relies on the samples of the entire synchronization signal block ('Case-A') for labeling the window, whereas the second method uses only the samples of the third OFDM symbol in an SSB (which contains the SSS). In this analysis, we consider a subset of the training and test windows by considering only those windows whose 100% of their samples belong to an SSB for the first approach or to the third OFDM symbol of an SSB for the second one i.e., $T_{\text{train}} = T_{\text{test}} = 100\%$.

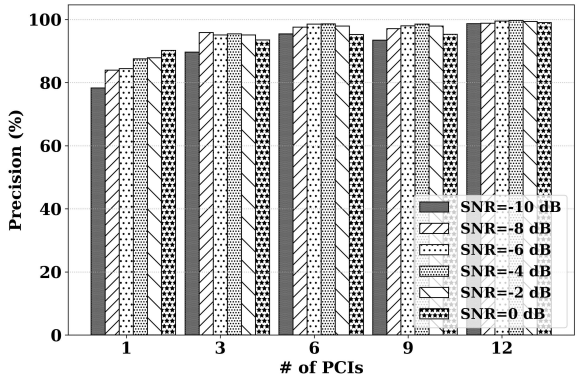
TABLE V
PCI DETECTION ACCURACY FOR TWO LABELING APPROACHES

Inference SNR (dB)	Complete SSB			3rd OFDM symbol of SSB		
	Precision (%)	Recall (%)	F1-Score (%)	Precision (%)	Recall (%)	F1-Score (%)
0	99.65	99.65	99.65	100.00	100.00	100.00
−2	96.82	96.82	96.82	100.00	100.00	100.00
−4	88.19	88.20	88.17	100.00	100.00	100.00
−6	76.45	76.08	76.19	100.00	100.00	100.00
−8	63.47	62.61	62.81	99.56	99.56	99.56
−10	49.20	48.05	48.17	96.19	96.17	96.17

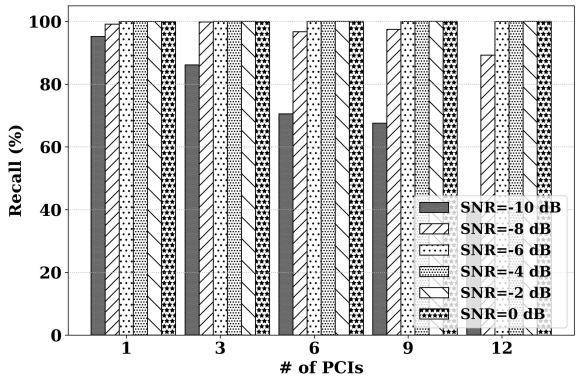
TABLE VI
PCI DETECTION CONSIDERING CASE-B SSB FORMAT

Inference SNR (dB)	without Decision Threshold			with Decision Threshold ($\alpha = 95\%$)		
	Precision (%)	Recall (%)	F1-Score (%)	Precision (%)	Recall (%)	F1-Score (%)
0	84.75	99.95	91.44	93.17	99.98	96.37
−2	87.25	99.96	93.02	94.76	99.98	97.26
−4	89.10	99.73	93.97	97.59	99.99	98.76
−6	90.11	97.67	93.63	98.78	99.29	99.03
−8	92.16	86.81	89.27	99.53	94.18	96.75
−10	85.40	59.34	68.89	98.71	63.22	75.31

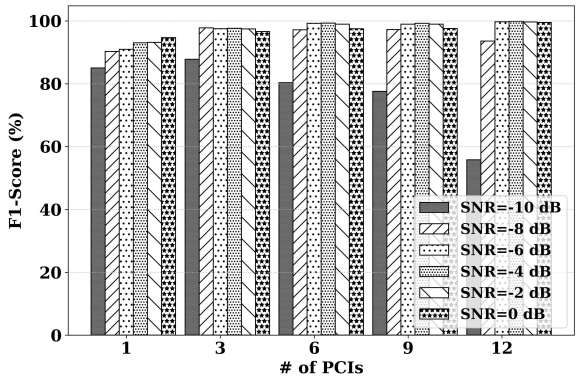
We aim to determine which component - SSB or SSS - offers superior performance in PCI detection, excluding the windows labeled as 'Others,' and a comparative analysis of performance between the two methodologies is delineated in Table V. It is discernible from the results that incorporating the third OFDM symbol while training the model, the PCI detection approach demonstrates greater resilience. In extremely low SNR scenarios, the effectiveness of PCI detection deteriorates significantly when utilizing the entire SSB for training. One contributing factor could be the first OFDM symbol of SSB where PSS is mapped, as it remains unchanged for one-third of all possible PCI values (0 to 1007). Conversely, the third OFDM symbol, where the SSS is, exhibits uniqueness for each PCI. This distinctiveness underpins the superior performance of the model, even under challenging low SNR conditions. Impressively, it maintains a precision of 96% even at an SNR=−10dB.



(a)



(b)



(c)

Fig. 6. PCI classification performance vs. number of PCIs in the training datasets at different SNR values considering 'Case-A'- SSB format and decision threshold, $\alpha = 95\%$.

D. Performance Evaluation for Different PCI Sets

In order to find the robustness of our PCI detection approach, we investigate the model's performance by varying the number of PCIs in the training dataset. The number varies in the set is $\{1, 3, 6, 9, 12\}$. The model's performance is tested using all 12 PCIs. We consider the full set of training and test windows (windows belonging to the third OFDM symbol of an SSB and windows that do not). We set $T_{\text{train}} = 100\%$ and $T_{\text{test}} = 50\%$. Fig. 6 demonstrates a high precision score for all sets. When we train the model with just one PCI, its detection ability is lower than when we train it with 12 PCIs. However, as the number of PCIs increases, the recall performance decreases significantly at SNR = -10 dB. Apart from this, the model works effectively for all other SNR conditions. The main takeaway is that if a model is designed to spot a specific number of PCIs, it can do so accurately without knowing about any other PCIs nearby in a network.

TABLE VII
PCI DETECTION ACCURACY FOR JOINT TRAINING OF TWO SSB FORMATS

Inference SNR (dB)	Training Traces Considering Different SSB Formats Jointly		
	Precision (%)	Recall (%)	F1-Score (%)
0	88.70	88.21	85.65
−2	90.10	81.91	81.17
−4	91.42	74.06	74.30
−6	91.08	68.15	69.71
−8	87.78	56.05	62.99
−10	73.27	31.28	39.20

E. Analysis of Different SSB Formats

All the earlier analyses are for only one type of SSB format (Case-A). We explore the capability of our model by training it with another SSB format, 'Case-B,' and six different PCIs. The PCI detection ability is tabularized in Table VI in terms of precision, recall, and F1 score. The results clearly show that the model trained on traces having 'Case-B' format excels at detecting PCI. Expanding on our progress, we have taken a further step by training the model with traces from two distinct SSB formats to see how it performs collectively. It prepares the classifier for scenarios where it needs to handle PCIs with varying SSB formats. This investigation considers $T_{\text{train}} = 100\%$ for training and $T_{\text{test}} = 50\%$ for testing. Analysis from Table VII reveals that the model yields commendable performance but not as the scenarios exclusively addressing a single SSB format. We consider six PCI values (99 to 104) for two distinct SSB formats (total of 12 classes). Similar to the scenario depicted in Fig. 6, we noticed a notable drop in recall performance at SNR of -10 dB when dealing with a larger class (12 PCIs). The reason for a marginal reduction in precision is that the model occasionally misclassifies cases where the PCI is same, but the SCS of SSB differs.

F. Frequency Offset Analysis

A critical challenge in 5G NR networks is the impact of CFO on received signals. Frequency offset often originates from oscillator inaccuracies and environmental factors, such as temperature, which can shift the transmission carrier frequency. To support high data rates in 5G systems for vehicular speeds up to 500 km/h, Doppler shifts can induce CFOs up to 2 kHz at the 4.2 GHz band−corresponds to 13% of the subcarrier spacing [24]. We consider a testing dataset with four CFO values: 0, 15, 30, and 45 kHz to evaluate the effect of CFO on our proposed model. We utilize a pre-trained model, which

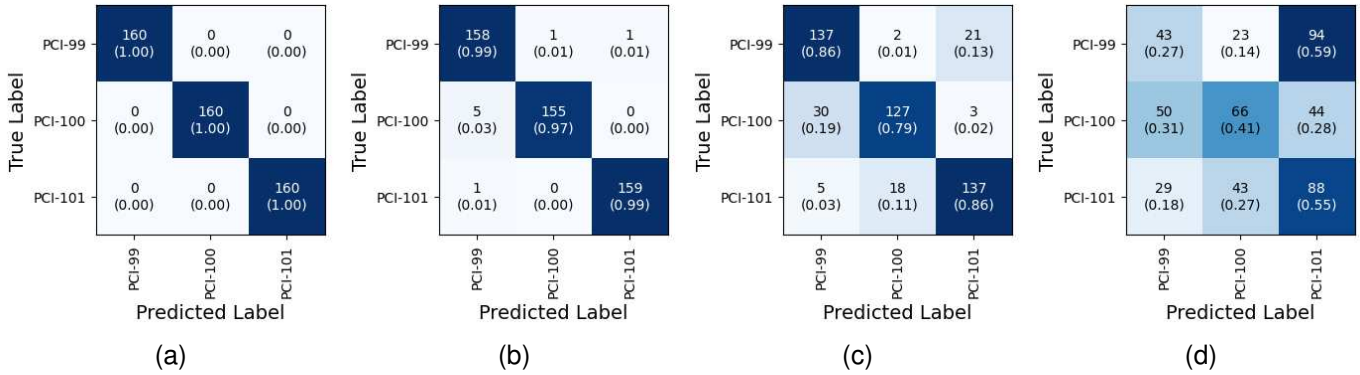


Fig. 7. Confusion matrices for PCI classifier considering different frequency offset at SNR = 0 dB (a) 0 KHz, (b) 15 KHz, (c) 30 KHz, and (d) 45 KHz.

TABLE VIII
PCI DETECTION ACCURACY FOR DIFFERENT CARRIER FREQUENCY OFFSET AT DIFFERENT SNRS

Inference SNR (dB)	Carrier Frequency Offset											
	CFO = 0 (kHz)			CFO = 15 (kHz)			CFO = 30 (kHz)			CFO = 45 (kHz)		
	Precision (%)	Recall (%)	F1-Score (%)	Precision (%)	Recall (%)	F1-Score (%)	Precision (%)	Recall (%)	F1-Score (%)	Precision (%)	Recall (%)	F1-Score (%)
0	100.00	100.00	100.00	98.36	98.33	98.34	83.71	83.54	83.54	41.39	41.04	40.43
−2	100.00	100.00	100.00	92.13	92.08	92.09	68.00	67.50	67.26	29.72	30.83	30.00
−4	100.00	100.00	100.00	82.19	81.87	81.85	56.72	56.04	55.99	34.47	35.21	34.57
−6	100.00	100.00	100.00	67.96	67.92	67.84	43.70	43.54	43.43	33.52	33.75	33.58
−8	98.34	98.33	98.33	51.43	51.25	51.31	39.56	39.58	39.47	32.77	32.29	32.46
−10	91.31	91.25	91.26	42.68	42.29	42.35	37.72	37.71	37.45	32.59	32.29	32.31

was developed using a training dataset based on the 'Case-A' SSB format and three PCI values (99, 100, 101). Importantly, the CFO is not included in the training data but introduced exclusively during inference. We take $T_{\text{test}} = 50\%$ for this analysis. Fig. 7 illustrates that our proposed model achieves perfect PCI classification accuracy at SNR = 0 dB in the absence of CFO. Even with a CFO of 15 kHz, the classifier maintains an accuracy of about 99%. The classification accuracy drops to 85% at a CFO of 30 kHz. However, the model's performance deteriorates significantly when the CFO reaches 45 kHz. The overall performance of our proposed approach across various SNRs is summarized in Table. VIII. The model's ability to classify PCI decreases when we lower the SNR value if CFO exists in the received signal.

G. Complexity Analysis

In our proposed method, the computational complexity mainly comes from the 2D convolutional layer in the CNN block. The input feature map has dimensions of $1 \times 1024 \times 2$, with input channels $C_{\text{in}} = 2$. The convolution is performed using a kernel of size 1×16 (kernel height, $K_{\text{H}} = 1$ and width, $K_{\text{W}} = 16$). The method produces output feature maps with dimensions of $H_{\text{out}} \times W_{\text{out}} \times C_{\text{out}}$, where height $H_{\text{out}} = 1$, width $W_{\text{out}} = 1024$ and the number of output channels C_{out} varies across six layers with values of 16, 24, 32, 48, 64, and 96, respectively. The overall computational complexity of the 2D convolutional layer is determined by the number of operations required, which is given by the expression-

$$O(C_{\text{out}} \times H_{\text{out}} \times W_{\text{out}} \times K_{\text{H}} \times K_{\text{W}} \times C_{\text{in}})$$

For the given parameters, this complexity is approximately $O(9.17 \times 10^6)$.

Detecting the PCI in 5G-NR signals involves several steps, each contributing to the overall computational complexity. The process begins with detecting the PSS, which helps determine the cell's identity within a group. There are three possible PSS sequences, and a receiver correlates the received signal with each of these sequences to identify the cell ID sector, $N_{\text{ID}}^{(2)}$. In calculating the complexity for each correlation, there are two FFT and one IFFT operations of length N, each requiring $N \log_2 N$ operations. Thus, the complexity of the traditional PSS detection per frame is about $O(3 \times 16.79 \times 10^6)$. Only the computational complexity of PSS detection is 5.5 times higher than our proposed method. Additionally, detecting PSS and SSS requires estimating and correcting frequency and timing offsets and performing OFDM modulation to generate reference signals for correlation, further adding to the computational complexity of the process. Traditional methods are computationally intensive, whereas our proposed approach significantly reduces the computational burden, making it far more efficient for PCI detection.

V. CONCLUSIONS

This paper introduced a CNN-based approach for extracting the PCI value directly from the time-domain 5G-NR signal,

bypassing conventional decoding. Rigorous performance evaluation, considering different SCS of SSBs, multiple PCIs, frequency offsets, and SNR values, demonstrated the robustness of the proposed approach in accurately predicting the PCI value. The proposed method involved processing time-domain baseband signals using windows of fixed sizes and leveraging a pre-trained CNN classifier to retrieve the PCI. Based on the simulation results, we noted that considering the 3^{rd} OFDM symbol of the SSB consistently delivers better performance. This was true even under the most challenging SNR conditions and situations where the received signal's SSB can be of any subcarrier spacing format. Additionally, the number of nearby cells constrained the model's performance, particularly at SNR = -10 dB. The proposed machine learning-driven PCI approach can be integrated into 5G-NR devices that require low computational overhead, such as smart repeater systems. It can also facilitate seamless handover procedures and effective synchronization at the UE.

ACKNOWLEDGEMENT

This research was supported in part by NSF (grants # 2413009, 2229386, 2425535, and 1822071), by the Broadband Wireless Access & Applications Center (BWAC), and by the WISPER Center. Any opinions, findings, conclusions, or recommendations expressed in this paper are those of the author(s) and do not necessarily reflect the views of NSF.

REFERENCES

[1] Y.-H. You and H.-K. Song, "Efficient sequential detection of carrier frequency offset and primary synchronization signal for 5G NR systems," *IEEE Transactions on Vehicular Technology*, vol. 69, no. 8, pp. 9212–9216, Aug. 2020.

[2] S. Parkvall, E. Dahlman, A. Furuskar, and M. Frenne, "NR: The new 5G radio access technology," *IEEE Communications Standards Magazine*, vol. 1, no. 4, pp. 24–30, 2017.

[3] J. Rodríguez-Fernández, "Joint synchronization and compressive channel estimation for frequency-selective hybrid mmWave MIMO systems," *IEEE Transactions on Wireless Communications*, vol. 21, no. 1, pp. 548–562, Jan. 2022.

[4] A. Omri, M. Shaqfeh, A. Ali, and H. Alnuweiri, "Synchronization procedure in 5G NR systems," *IEEE Access*, vol. 7, pp. 41 286–41 295, Mar. 2019.

[5] H. Yan and D. Cabric, "Compressive initial access and beamforming training for millimeter-wave cellular systems," *IEEE Journal of Selected Topics in Signal Processing*, vol. 13, no. 5, pp. 1151–1166, Sep. 2019.

[6] D. Wang, Z. Mei, H. Zhang, and H. Li, "A novel PSS timing synchronization algorithm for cell search in 5G NR system," *IEEE Access*, vol. 9, pp. 5870–5880, Jan. 2021.

[7] Y.-H. You, J.-H. Park, and I.-Y. Ahn, "Complexity effective sequential detection of secondary synchronization signal for 5G new radio communication systems," *IEEE Systems Journal*, vol. 15, no. 3, pp. 3382–3390, Sep. 2021.

[8] J.-L. Yin, M.-C. Lee, W.-H. Hsiao, and C.-C. Huang, "A novel network resolved and mobile assisted cell search method for 5G cellular communication systems," *IEEE Access*, vol. 10, pp. 75 331–75 342, Jul. 2022.

[9] M. Krunz, I. Aykin, S. Sarkar, and B. Akgun, "Online reinforcement learning for beam tracking and rate adaptation in millimeter-wave systems," *IEEE Transactions on Mobile Computing*, vol. 23, no. 2, pp. 1830–1845, Feb. 2023.

[10] T. P. Fowdur and B. Doorgakant, "A review of machine learning techniques for enhanced energy efficient 5G and 6G communications," *Engineering Applications of Artificial Intelligence*, vol. 122, Jun. 2023.

[11] A. Mughees, M. Tahir, M. A. Sheikh, and A. Ahad, "Towards energy efficient 5G networks using machine learning: Taxonomy, research challenges, and future research directions," *IEEE Access*, vol. 8, pp. 187 498–187 522, Oct. 2020.

[12] S. Won and S. W. Choi, "A tutorial on 3GPP initial cell search: Exploring a potential for intelligence based cell search," *IEEE Access*, vol. 9, pp. 100 223–100 263, Jul. 2021.

[13] Z. Lin, J. Li, Y. Zheng, N. V. Irukulapati, H. Wang, and H. Sahlin, "SS/PBCH block design in 5G new radio (NR)," in *Proc. of the IEEE Globecom Workshops (GC Wkshps)*, Abu Dhabi, UAE, Dec. 2018.

[14] 3GPP, "NR; User Equipment (UE) radio transmission and reception," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 38.101, May 2022, version 17.5.0.

[15] 3GPP, "NR; Physical layer; General description," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 38.201, Sep. 2018, version 15.0.0.

[16] A. Chakrapani, "On the design details of SS/PBCH, signal generation and PRACH in 5G-NR," *IEEE Access*, vol. 8, pp. 136 617–136 637, Jul. 2020.

[17] 3GPP, "NR; Physical channels and modulation," 3rd Generation Partnership Project (3GPP), Technical Specification (TS) 38.211, Jul. 2020, version 16.2.0.

[18] D. Inoue, K. Ota, M. Sawahashi, and S. Nagata, "Physical cell ID detection using joint estimation of frequency offset and SSS sequence for NR initial access," in *Proc. of the IEEE Vehicular Technology Conference*, Helsinki, Finland, Apr. 2021.

[19] K. Ota, A. Shimura, M. Sawahashi, and S. Nagata, "Physical cell ID detection probability using synchronization signals for NR radio interface," in *Proc. of the International Symposium on Wireless Personal Multimedia Communications (WPMC)*, Chiang Rai, Thailand, Nov. 2018, pp. 620–625.

[20] M. Wang, D. Hu, L. He, and J. Wu, "Deep-learning-based initial access method for millimeter-wave MIMO systems," *IEEE Wireless Communications Letters*, vol. 11, no. 5, pp. 1067–1071, Mar. 2022.

[21] H. Han, T. Shen, Y. Chen, W. Lu, S. Zheng, X. Wang, and X. Yang, "A ConvLSTM-based blind receiver for physical layer wireless communication," *IEEE Transactions on Vehicular Technology*, vol. 73, no. 5, pp. 7351–7356, May 2024.

[22] Q. Wang, B. Su, C. Wang, L. P. Qian, Y. Wu, and X. Yang, "ConvLSTM-based spectrum sensing at very low SNR," *IEEE Wireless Communications Letters*, vol. 12, no. 6, pp. 967–971, Jun. 2023.

[23] W. Zhang, M. Feng, M. Krunz, and A. Hossein Yazdani Abyaneh, "Signal detection and classification in shared spectrum: A deep learning approach," in *IEEE INFOCOM 2021 - IEEE Conference on Computer Communications*, Vancouver, BC, Canada, May 2021.

[24] P. Siyari, H. Rahbari, and M. Krunz, "Lightweight machine learning for efficient frequency-offset-aware demodulation," *IEEE Journal on Selected Areas in Communications*, vol. 37, no. 11, pp. 2544–2558, Nov. 2019.