Sub-Decimeter VLC 3D Indoor Localization With Handover Probability Analysis

SEYED MOHAMMAD SHEIKHOLESŁAMI, FAHIMEH FAZEL, JAMSHID ABOUEI, (Senior Member, IEEE), AND KONSTANTINOS N. PLATANIOTIS, (Fellow, IEEE)

1Department of Electrical and Computer Engineering, University of Tehran, Tehran 1417935840, Iran
2Department of Electrical Engineering, Yazd University, Yazd 8915818411, Iran
3The Edward S. Rogers Sr. Department of Electrical and Computer Engineering, University of Toronto, Toronto, ON M5S 3G8, Canada

Corresponding author: Jamshid Abouei (abouei@yazd.ac.ir)

The work of Jamshid Abouei and Konstantinos N. Plataniotis was supported in part by the Natural Sciences and Engineering Research Council of Canada (NSERC). The work of Jamshid Abouei was performed when he was a visiting professor in the Department of Electrical and Computer Engineering, University of Toronto, Toronto, Canada.

ABSTRACT This paper addresses the problem of Visible Light Communication (VLC)-based indoor localization and handover, where mobile users communicate with hybrid VLC/mmWave Access Points (APs). The VLC system consists of multiple Light-Emitting Diodes (LEDs) treated as VLC transmitters, multiple Photodiodes (PDs) on the user’s smart device, and multiple mmWave Radio Frequency (RF) transmitters used as complementary APs for the VLC system in the case of blockage. We propose a Convolutional Neural Network (CNN)-based algorithm consisting of offline and online modes. In the offline mode, we gather a data set by dividing the environment into fixed-sized elements where the received VLC signal along with the data attained from the smart device at each element represent a sample to train a CNN model for indoor localization. In the online mode, users employ the received VLC signals to estimate their locations. We then propose a virtual soft handover process according to the Coordinated Multiple Point (CoMP) transmission, where the HandOver Margin (HOM) and Waiting Time (WT) are dynamically set based on the change in Signal-to-Noise-Ratio (SNR) values in consecutive time slots. We derive a closed-form expression for the average effective throughput during the handover process, which shows the algorithm’s superior performance compared to conventional soft and hard handovers. Simulation results show an average positioning error of 4.31 centimeters for the proposed localization algorithm in a $5 \times 4 \times 3$ m$^3$ smart environment.

INDEX TERMS Indoor localization, visible light communication (VLC), convolutional neural network (CNN), coordinated multiple point (CoMP)-based soft handover, fountain code.

I. INTRODUCTION Highly precise and robust user localization has become an inseparable part of 6G and beyond wireless networks to support various location-based services in many emerging applications such as smart homes and automatic factories [1]. Although the Global Positioning System (GPS) is widely used for outdoor applications such as road transport services, it is not able to provide sub-meter level accuracy in an indoor environment as satellite signals are scattered and attenuated by indoor roofs and walls [2]. In the past decade, Radio Frequency (RF) systems such as WiFi have been widely used for indoor localization, by employing high-level software and hardware resources [3]. However, RF systems suffer from low energy efficiency and certain weakness in security, and cannot solely satisfy the Quality of Service (QoS) in the next generation of wireless networks due to the wide employment of Internet of Things (IoT)-based mobile devices (IMDs) for smart indoor services.

Visible Light Communication (VLC) has emerged as a promising technology that is employed as a complementary technique for RF systems in indoor localization. VLC yields more benefits in comparison with the RF technology, such as higher data rate, tremendous bandwidth, and higher security with fewer health risks [4]. Besides, it is more energy-efficient than RF and can be implemented using the commonly-used Light-Emitting Diodes (LEDs) as transmitters and Photodiodes (PDs) as receivers [5]. The combination of VLC technology with millimeter-Wave (mmWave), known as hybrid VLC/mmWave, provides short-range communications with a high level of secrecy rate for serving IMDs. However, mobile users need to maintain their
connectivity during their movements between the coverage area of VLC cells in an indoor environment [6]. In this regard, two types of handover, namely Horizontal HandOver (HHO) and Vertical HandOver (VHO), should be utilized in hybrid VLC/mmWave indoor area, where HHO refers to a handover between two consecutive VLC transmitters, while VHO refers to the handover between a VLC transmitter and an RF transmitter.

Another practical challenge in such hybrid VLC/mmWave indoor networks is that the Received Signal Strength (RSS) may either change rapidly due to the ping-pong handover or reduce significantly in the receiving end due to user mobility, path loss, and reflection, resulting in a high detection error. To tackle this issue, channel coding is regarded as a popular solution that can control the error at the receiver. Although fixed-rate channel coding can guarantee a reliable transmission in traditional wireless systems with slight changes in channel conditions, it does not work properly in channels with unstable conditions that happen more in VLC indoor service applications. In this situation, rateless fountain codes are applied to adjust the channel code rate on the fly based on the channel conditions to provide reliable transmissions. They have exhibited strong capabilities in reaching the above targets over various channel conditions in indoor environments. The fountain encoding process is extremely simple and has a tight power budget.

This paper addresses the aforementioned challenges by proposing a Convolutional Neural Network (CNN)-based localization algorithm for an indoor environment in the presence of multiple VLC transmitters and the fountain code. The CNN model takes the Line of Sight (LoS) and Time of Arrival (ToA) of the received VLC signal as well as the data obtained from the gyroscope sensors as inputs to estimate the 3 Dimensional (3D) location of the user in the output. The proposed algorithm utilizes the estimated location of users to maintain their connection. Moreover, a virtual soft handover algorithm based on the Coordinated Multiple Point (CoMP) transmission is proposed to prevent the ping-pong effect during the handover process.

A. RELATED WORKS
Motivated by the literature, we review the recent works in the use of VLC technology for indoor localization and the handover process in VLC systems as follows.

1) INDOOR LOCALIZATION IN VLC SYSTEMS

Generally, existing VLC indoor localization techniques in the literature are categorized into three groups: i) triangulation [7]–[10], ii) vision analysis [11]–[14], and iii) fingerprinting [6], [15]–[17]. The triangulation localization techniques employ the geometric properties of triangles to locate the user. These techniques are divided into two classes of angulation and lateration. Angulation techniques, such as Angle Difference of Arrival (ADoA) [7] and Angle of Arrival (AoA) [8], measure the angles between the user and multiple VLC transmitters to estimate the location of the user. The lateration techniques, such as RSS and Time of Arrival (ToA), measure the distance between the user and multiple VLC transmitters to calculate the location of the user. It was shown in [10] that a combination of these techniques can improve the accuracy of VLC-based indoor localization.

Vision-based indoor localization algorithms utilize the camera on other smart devices to find the relationship between the 3D position of objects (e.g., the LEDs) in the real world and their 2D projection on the image sensor. Authors in [14] proposed a hybrid RSS/AoA algorithm to estimate the position of the user by employing image processing and gyroscope measurements in an VLC-based indoor environment with a single LED.

Fingerprinting methods include offline and online stages for indoor localization. In the offline mode, information of the received signal, such as RSS and AoA, is collected in a data set. Then, in the online stage, the user is localized by comparing the measured data and the collected data set using different techniques, such as k-Nearest Neighbor (kNN) and probabilistic methods. Authors in [15] used a fusion of multiple classifiers to estimate the position of the user using RSS fingerprints in a VLC-based indoor environment. Reference [16] used principal component analysis to extract important features from the flicker frequency spectra. Then, it utilized kNN clustering and Artificial Neural Network (ANN) to identify the light source from its flickers. Authors in [18] applied two machine learning techniques, namely second-order linear regression and Kernel Ridge Regression Machine Learning (KRRML) with sigmoid function data preprocessing for VLC-based indoor localization. Additionally, [19], [20] employ Weighted K-Nearest Neighbor (WKNN) to estimate the position of the receiver in a VLC indoor environment, while [21] uses extreme machine learning for visible light positioning.

To the best of our knowledge, VLC-based triangulation techniques can reach a high positioning accuracy by employing at least two VLCs in an indoor environment at the cost of complex geometric analysis. In addition, vision-based localization algorithms have a simple mechanism and can be implemented on any camera-equipped smart device. However, it cannot be applicable in smart devices without a camera, like a smartwatch. Moreover, frequent use of camera is energy consuming for smart devices. Recently, Deep Learning (DL) algorithms are widely used in VLC systems, particularly, for smart indoor localization purposes. It is demonstrated in [22] that applying deep learning in VLC indoor positioning techniques not only improves the accuracy of the positioning but also reduces the computational complexity of indoor localization compared to conventional schemes since deep learning techniques learn from a series of samples labeled with their 2D/3D locations to estimate a new sample, while the conventional techniques, such as triangulation algorithms, employ complicated geometric computations for user positioning.
2) HANDOVER PROCESS IN VLC INDOOR SERVICES

Different handover management techniques have been studied in the literature including i) applying the resource allocations and placements of VLC transmitters [23]–[28], ii) setting a Waiting Time (WT) before the handover process [29]–[32], and iii) setting a threshold value for HandOver Margin (HOM) [33], [34]. The authors in [27], [28] employed Multipath-Transmission Control Protocol (MPTCP) for vehicular VLC to tackle the handover problem. The authors in [23], [24] determined the steerable and directional VLC devices’ placements such that the overlapped area between adjacent transmitters becomes large enough for successful handovers. In [25], [26], a centralized controller adopted the soft handover process among different VLC transmitters by determining the subcarrier assignment to users subject to QoS constraints. The authors in [30] examined the vertical handover in hybrid VLC/RF systems as a Markov decision process aiming to optimize the Quality of Experience (QoE) and time to trigger the handover. The authors in [29] aimed to optimize the waiting time before the vertical handover process based on the rate of blocking and the recovery of VLC optical wireless channels. Authors in [31], [32] employ reinforcement learning to optimize the WT before the handover process. References [33], [34] derived a mathematical expression related to QoS parameters of users in a VLC network. Then, a threshold value for the performance degradation of VLC was computed to determine triggering the vertical handover between VLC and WiFi networks.

To the best of our knowledge, there is no prior research on setting the WT before the handover process, HOM, and average effective throughput analysis in a VLC network. This paper addresses this gap by proposing a CoMP-based soft handover algorithm to jointly set the WT and HOM while applying the rateless fountain code for reliable transmission. Moreover, unlike existing researches, the average effective throughput is derived during the handover process.

B. MAIN CONTRIBUTIONS

In this paper, we consider a hybrid VLC/mmWave system in an indoor environment with multiple VLC transmitters located on the ceiling. We assume that each user is carrying a smart device (e.g., smartphone or smartwatch) equipped with VLC receivers, i.e., PDs, along with inertial sensors, such as accelerometers and gyroscopes. We propose a DL-based algorithm for user localization by processing the received VLC signals and the data obtained from the sensors on the smart device. Besides, we analyze the handover process to maximize the average effective throughput. The key contributions of this paper are summarized as follows.

- We propose a CNN-based indoor localization algorithm containing two stages. In the first stage (offline mode), the environment is divided into several elements with equal sizes. The VLC signal characteristics in the elements are measured and stored in a data set. Then, the data set is used as the training set of a CNN model that estimates the 3D coordinates of the user. In the second stage (online mode), the user uses the received VLC signal and the data obtained from the gyroscope sensor available on the smart device as the inputs of the CNN model to estimate its 3D location. The Central Control Management (CCM) is responsible for controlling the transmitted signal from VLC sources, their encoding process by the fountain code, and the handover process during the movement of users.

- We handle the handover process by employing a virtual soft handover algorithm according to the CoMP transmission. Applying the indoor localization algorithm, a Handover VLC List (HVL) is formed, and the VLC transmitter with the minimum horizontal distance from the user is chosen as the serving transmitter. When the user moves close to the cell edge, the handover process is invoked. Then, the HVL is updated, and the cooperating VLC transmitter is determined according to the updated HVL. In the second stage, HOM and WT are dynamically tuned based on the alternation of Signal-to-Noise-Ratio (SNR) values related to the serving and cooperating VLC transmitters in consecutive time slots.

- Different from existing researches, the closed-form expressions for the average effective throughput are derived for two cases in the VLC-based indoor environment, and each case for transmission is determined. More precisely, the user dynamically switches between two cases during the handover process to maximize the average effective throughput.

- We run some simulations in a smart environment (e.g., a smart living room) for different element sizes. The results show that the accuracy of the CNN model increases as the element size is reduced. In addition, the proposed localization algorithm reaches the average positioning error of 4.31 centimeters over the test set for element size 5 × 5 × 5 cm³, where more than 95.7% of test samples have an error less than 10 centimeters.

- Finally, we evaluate the performance of handover algorithms in terms of average effective throughput and Bit Error Rate (BER). It is shown that the proposed scheme outperforms the conventional soft and hard handovers in terms of the mentioned performance metrics. Moreover, it is depicted that increasing the users velocity leads to increasing HOM and decreasing WT.

C. ARTICLE OUTLINE

The rest of the paper is structured as follows. Section II presents the smart environment system model and assumptions. Section III, contains the proposed VLC-based indoor localization algorithm. Then, in Section IV, the algorithm for the CoMP-based soft handover process is proposed to maximize the average effective throughput. In Section V, the simulation results indicate the validity of the proposed...
algorithm. Finally, we summarize our findings and some conclusions in Section VI.

II. SMART INDOOR ENVIRONMENT MODEL

In this work, we consider a hybrid VLC/mmWave indoor environment like a smart living room with the size $A \times B \times C$, where $C$ indicates the height. The indoor area is equipped with $N_T$ VLC transmitters, indexed by $T_v = \{T_{v_1}, \ldots, T_{v_{N_T}}\}$, where each transmitter $T_{v_i} \in T_v$ is positioned at coordinates $L_{v_i}$, and has a layout of $L$ LEDs located on the ceiling at height $C$. LEDs are used for both illumination and transmission purposes and they are equipped with VLC modules.

We also consider a mmWave RF transmitter used as a complementary Access Point (AP) for the VLC system in the case of blockage. Based on the position of VLC lights, we divide the indoor area into $N_T$ regions, indexed by the set $R = \{R_1, \ldots, R_{N_T}\}$, where each region $R_i \in R$ is overlaid with the $i$th VLC transmitter $T_{v_i} \in T_v$, under the real condition of $R_i \cap R_j = \emptyset$ and $\bigcup_{i} R_i = R$ (see Fig. 1). In addition, we consider $N_U$ Internet of Things Mobile Devices (IMDs) moving on the floor, indexed by the set $U = \{u_1, \ldots, u_{N_U}\}$, where each IMD is equipped with an array of $P$ PDs. Moreover, the gyroscope sensor on the IMD measures the rotation angles over the three axis, i.e., $\omega_{u_m}, \rho_{u_m}, \delta_{u_m}$ (see Fig. 1). In the proposed system model, the output signals of LEDs and the RF link along with fountain encoding process are controlled by a Central Control Management (CCM) unit. The IMDs can receive their signals from either VLC transmitters or mmWave RF AP although they prefer to communicate with VLC transmitters due to their higher data rates and energy efficiency. More precisely, IMDs connect to the RF AP if the corresponding VLC channel is blocked in downlink transmission. For the uplink transmission, IMDs take the advantage of the RF AP to access the CCM.

The received VLC signal at IMD $u_m$, $1 \leq m \leq N_U$, located at coordinates $L_{u_m} = (x_{u_m}, y_{u_m}, z_{u_m})$, is obtained as

$$r_{u_m} = H_{T_{u_m}, u_m} x_{u_m}^* + \sum_{m \neq u^*} H_{T_{u_m}, u_m} x_{u_m}^* + n_{u_m},$$

where $T_{u_m}$ refers to the current transmitter serving IMD $u_m$, and $x_{u_m}^* \in \mathbb{C}^{L \times 1}$ denotes the pre-coded signal vector related to the transmitter index $v_{u_m}$. In the proposed network model, we denote $s_{v_{u_m}} \in \mathbb{C}^{L \times 1}$ as the signal transmitted from desired VLC transmitter where each entry of $s_{v_{u_m}}$ is composed of $K$ symbols. Then, the pre-coded signal $x_{u_m}^*$ is obtained after applying fountain encoder, and each entry of $x_{u_m}^*$ is composed of $M$ symbols. In addition, $\mathbb{I}_{u_m}$ in (1) is the set of possible interference terms caused by other VLC transmitters when IMD $u_m$ is located in the overlapped area of VLC cells. We utilize the Alamouti code and maximum likelihood detection technique to decouple the signal transmitted from several VLCs without applying a complicated decoding technique. In addition, $H_{T_{u_m}, u_m} \in \mathbb{C}^{P \times L}$ is the channel matrix between VLC transmitter $T_{v_m}$ and IMD $u_m$. Finally, $n_{u_m} \in \mathbb{C}^{P \times 1}$ is the additive noise vector at the receiver $u_m$, where each component of $n_{u_m}$ is modeled as a zero-mean Gaussian noise consisting of three independent terms namely, shot, thermal, and background noise. Therefore, the variance of the noise is obtained as follows:

$$\sigma_m^2 = \sigma_{\text{shot}}^2 + \sigma_{\text{bg}}^2 + \sigma_{\text{thermal}}^2,$$

where the variance of the shot noise would be as $\sigma_{\text{shot}}^2 = 2qRP_{\text{VLC}}$, where $q$, $R$, and $P_{\text{VLC}}$ indicate the electric charge, photodiode responsivity, and transmitter power, respectively. In addition, the variance of the background noise, denoted
by $\sigma_{bg}^2$ and the thermal noise variance, $\sigma_{\text{thermal}}^2$, are given by:

$$\sigma_{bg}^2 = 2gI_{bg},$$

$$\sigma_{\text{thermal}}^2 = \frac{4KT_k}{R_F},$$

(3)

(4)

where $I_{bg}$, $K$, $T_k$, and $R_F$ are the background current of the PD, background light power, Boltzmann’s constant, absolute temperature, and receiver’s resistance.

There are two types of VLC links between the receivers and transmitters: i) direct links, or equivalently, Line of Sight (LoS), where the signal is received directly from LEDs without any reflection, and ii) reflection links, in which signals are received after several bounces. Thus, the channel impulse response between $j^{th}$ LED of the transmitter $T_{vj}$ and $i^{th}$ PD of the IMD $u_m$ can be written as follows:

$$h_{im,jn} = h_{im,jn}^{\text{dir}} + h_{im,jn}^{\text{ref}},$$

(5)

where $h_{im,jn}^{\text{ref}} = \sum_{z \in \mathcal{Z}} h_{im,jn}^{\text{ref},z}$ represents the summation of all possible number of reflections, denoted by $\mathcal{Z}$ with the cardinality of $|\mathcal{Z}| = N$, in which $h_{im,jn}^{\text{ref},z}$ is the impulse response of the link with $z$ reflections.

The DC channel gain of the direct VLC link between $j^{th}$ LED of transmitter $T_{vj}$ and $i^{th}$ PD of IMD $u_m$ is obtained as follows [35]:

$$L_{im,jn} = \frac{(w + 1)g(\phi_{im,jn})A_r \cos^w(\theta_{im,jn}) \cos(\phi_{im,jn})}{2\pi d_{im,jn}^2},$$

(6)

where $w = -\frac{\ln 2}{\ln \cos(\theta_{1/2})}$ is the Lambertian factor where $\theta_{1/2}$ denotes the semiangle at half power for LEDs, $A_r$ is the area of the receiver, $\theta_{im,jn}$ and $\phi_{im,jn}$ represent the irradiance and incident angles, respectively (see Fig. 1). In addition, $d_{im,jn}$ refers to the Euclidean distance between $j^{th}$ LED of the transmitter $T_{vj}$ and $i^{th}$ PD of the user $u_m$. Denoting $\Phi_c$ and $n_c$ as the receiver Field of View (FoV) semiangle and the refractive index, respectively, the gain of the optical concentrator, $g(\phi_{im,jn})$, is obtained as follows:

$$g(\phi_{im,jn}) = \begin{cases} \left(\frac{n_c}{\sin \Phi_c}\right)^2, & 0 \leq \phi_{im,jn} \leq \Phi_c \\ 0, & \text{otherwise}. \end{cases}$$

(7)

Since a reflection link experiencing $N$ bounces is modeled as a sequence of $N$ direct links, the DC channel gain for a reflection link is calculated as $L_{\text{ref}}^N = \prod_{z=1}^N L_{c,z+1}$, where $L_{c,z+1}$ represents the path loss of $z^{th}$ bounce.

We assume that all LEDs of each transmitter send the same signal. As mentioned in (1), we apply the space-time block code (Tarokh’s scheme) to remove the interference caused by other VLC transmitters. In the special case with two VLC transmitters, the encoder makes a block from symbols and maps them to two VLC transmitters based on the following matrix form expression:

$$\bar{X} = \begin{bmatrix} x_{T_{v1}} & x_{T_{v2}} \\ -x_{T_{v2}} & x_{T_{v1}} \end{bmatrix}.$$  

(8)

Then, each entry of the above matrix is transmitted through all LEDs of the same transmitter. Accordingly the received signal at the receiver is achieved as:

$$\begin{bmatrix} r_{1m}^1 \\ r_{2m}^1 \end{bmatrix} = \begin{bmatrix} H_{T_{v1},u_m}x_{T_{v1}} + H_{T_{v1},u_m}x_{T_{v2}} + n_{1m} \\ -H_{T_{v1},u_m}x_{T_{v1}} + H_{T_{v1},u_m}x_{T_{v2}} + n_{2m} \end{bmatrix},$$

(9)

where $r_{1m}^1$ and $r_{2m}^1$ are the received signals during two consecutive time-slots, while we assume that channel matrix coefficients are constant during transmitting two signals. Applying the encoding technique, we can decouple the signal transmitted from each transmitter as follows:

$$\begin{bmatrix} \bar{x}_{1m} \\ \bar{x}_{2m} \end{bmatrix} = \begin{bmatrix} H_{T_{v1},u_m} & H_{T_{v1},u_m} \\ -H_{T_{v1},u_m} & H_{T_{v1},u_m} \end{bmatrix} \begin{bmatrix} r_{1m}^1 \\ r_{2m}^1 \end{bmatrix}.$$  

(10)

$$\bar{x}_{1m} = \begin{bmatrix} H_{T_{v1},u_m} \bar{x}_{T_{v1}} + H_{T_{v1},u_m} \bar{x}_{T_{v2}} + n_{1m} \end{bmatrix}. $$  

(11)

Finally, we utilize the maximum likelihood detection at the receiver to detect the signal transmitted from the VLC transmitter. According to the above scheme and in the presence of other interferer transceivers, the received SNR at user $u_m$ is calculated as

$$\gamma_{T_{v1},u_m} = \frac{\bar{y}}{\|H_{T_{v1},u_m}x_{T_{v1}}\|^2},$$

(12)

where $\gamma$ is the average SNR received at the transmitter. Table 1 indicates all notations utilized in the paper.

### III. PROPOSED CNN-BASED INDOOR LOCALIZATION ALGORITHM

In this section, we develop a DL-based fingerprinting algorithm for user localization by employing the received VLC signals and the data attained from available sensors on the smart devices. The proposed algorithm contains offline and online modes. The offline mode comprises two steps, namely data acquisition and training. In the data acquisition step, we generate a training data set by dividing the environment into 3D elements of the same size. The received VLC signal characteristics at the center of each element with the gyroscope data from the smart devices are stored as a training sample. In the training step, the collected data set is used for training a CNN model that estimates the 3D location by taking the mentioned features of the data set as its inputs.

The main reason for using CNN is that it can effectively extract the hidden structure and inherent features of the data. In the online mode, the smart device uses the received VLC signal characteristics and the gyroscope data as the inputs of the CNN model for localization. Moreover, in case of VLC blockage, the inertial sensors available on the smart device are used to estimate the location of the user according to its movement and the previous accurate location.

### A. OFFLINE MODE DESIGN

The offline mode design is summarized in two steps, namely data acquisition and training, as follows:
Y.D. the CNN model, denoted by

1) DATA ACQUISITION

In this step, we aim to generate the training data set for the CNN model, denoted by \( D = (X, Y) \), where \( X \) and \( Y \) represent the features and labels, respectively. Denoting \( N_s \) as the number of training samples, each sample \( x^{(k)} \) in the feature set \( X = [x^{(1)}, \ldots, x^{(N_s)}]^T \) contains the LoS channel gain and the ToA between the VLC transmitters and the receiver as well as the angular position of the smart device. In addition, \( Y = [y^{(1)}, \ldots, y^{(N_s)}]^T \), where \( y^{(k)} = (x^{(k)}, y^{(k)}, z^{(k)})^T \), \( k = 1, \ldots, N_s \), is the 3D coordinates of the center of \( k^{th} \) sample. It is worth to mention that we assume a noise-free environment when generating the training data, while, we test the algorithm with noisy inputs. In order to provide a comprehensive data set which improves the generalization of the proposed algorithm, we take samples from different parts of the environment. Towards this goal,
we partition the indoor environment into $N_v$ cubic elements with equal size of $d_x \times d_y \times d_z$ (e.g., $20\,\text{cm} \times 20\,\text{cm} \times 20\,\text{cm}$), where $d_x$, $d_y$, and $d_z$ denote the length of each element in the corresponding axis (see lines 1-12 of Algorithm 1). It should be noted that these elements are different from $R_i \in \mathbb{R}$ region that is covered by $i^{th}$ VLC transmitter.

Then, the Euclidean distance between the center of the transmitter and the center of $k^{th}$ element, denoted by $d_{T_{vn}}^{(k)}$, is calculated (see line 15). Next, the irradiance angle between sample $k$ and each transmitter $T_{vn}$, denoted by $\theta_{T_{vn}}^{(k)}$, is obtained according to (line 16 of Algorithm 1): 

$$
\theta_{T_{vn}}^{(k)} = \cos^{-1}\left(\frac{C - c_{T_{vn}}^{(k)}}{d_{T_{vn}}^{(k)}}\right). 
$$

Denoting $\delta^{(k)}$, $\rho^{(k)}$, and $\omega^{(k)}$ as the rotation angles of $k^{th}$ sample obtained from the gyroscope sensor on the smart devices, the normal vector of $k^{th}$ sample, denoted by $\mathbf{n}^{(k)}$, is obtained in (14), as shown at the bottom of the page. Thus, the incident angle is obtained as follows:

$$
\phi_{T_{vn}}^{(k)} = \cos^{-1}\left(\frac{\mathbf{n}^{(k)} \cdot \mathbf{L}_{T_{vn}}^{(k)}}{|\mathbf{n}^{(k)}||\mathbf{L}_{T_{vn}}^{(k)}|}\right), 
$$

where $\mathbf{L}_{T_{vn}}^{(k)} = L^{(k)} - L_{T_{vn}}$, and $L^{(k)}$ represents the location of $k^{th}$ sample. Throughout this subsection, we omit subscript $m$ from the channel gain described in (6), and denote $L_{i,j,n}$ as the gain of the LoS link between the $i^{th}$ PD of sample $k$ and $j^{th}$ PD from transmitter $T_{vn}$. Accordingly, if sample $k$ lies in the coverage area of transmitter $T_{vn}$, i.e., $\phi_{T_{vn}}^{(k)} \leq \Phi_c$, the Euclidean distance between the LEDs of transmitter $T_{vn}$ and PDs of a receiver located at the center of $k^{th}$ element, denoted by $d_{i,j,n}^{(k)}$ is calculated. Moreover, the corresponding irradiance and incident angles, $\theta_{i,j,n}^{(k)}$ and $\phi_{i,j,n}^{(k)}$ are obtained similar to (13) and (15) (see lines 18-21 of Algorithm 1). In line 22, the LoS channel gain $t_{i,j,n}^{(k)} \forall i = 1, \ldots, P, \forall j = 1, \ldots, L$, is obtained according to (6) and stored in $x^{(k)}$. Additionally, the ToA between $i^{th}$ LED of transmitter $T_{vn}$ and $j^{th}$ PD of a receiver located at the center of $k^{th}$ element, denoted by $t_{i,j,n}^{(k)}$, is calculated as follows:

$$
t_{i,j,n}^{(k)} = c \frac{d_{i,j,n}^{(k)}}{d_{i,j,n}^{(k)}},
$$

where $c = 3 \times 10^8 \,\text{m/s}$ denotes the speed of light. Finally, the feature set, $X$, and labels, $Y$, form a set of input-output pairs that are stored in $D$ as the training data set (lines 29-31 of Algorithm 1).

2) TRAINING (METHODOLOGY)

The proposed algorithm proceeds with training a CNN model as shown in Fig. 2. The model takes $x^{(k)}$ as inputs and delivers the estimated 3D position, $(\tilde{x}^{(k)}, \tilde{y}^{(k)}, \tilde{z}^{(k)})$, in the output. The first step in the proposed indoor localization model is the normalization process, which is considered an important step in training DL models, whose main purpose is to change the values of features to a common scale resulting in faster convergence. In this paper, we use the z-score normalization for each feature, denoted by $x_q^{(k)}$, as follows:

$$
\tilde{x}_q^{(k)} = \frac{x_q^{(k)} - \mu_q}{\sigma_q}, \quad k = 1, \ldots, N_v, 
$$

where $\mu_q$ and $\sigma_q$ denote the mean and standard deviation of $q^{th}$ feature in data set $D$, respectively.

While the features corresponding to the received VLC signals are used as the input of the CNN, the data obtained from the gyroscope sensor on the smart device is used as the input of the Fully-Connected (FC) layers. More precisely, the features corresponding to the LoS and ToA of the VLC signals are reshaped to form two vectors with size $1 \times L'$ where $L' = P \times N_T \times L$, and the two vectors are corresponding to the LoS and ToA. The convolutional layers with the max pooling layers are considered efficient feature extractors.

\[
\mathbf{n}^{(k)} = \begin{bmatrix}
\cos(\delta^{(k)}) & -\sin(\delta^{(k)}) & 0 \\
\sin(\delta^{(k)}) & \cos(\delta^{(k)}) & 0 \\
0 & 0 & 1
\end{bmatrix}
\begin{bmatrix}
1 & 0 & 0 \\
0 & \cos(\rho^{(k)}) & -\sin(\rho^{(k)}) \\
0 & \sin(\rho^{(k)}) & \cos(\rho^{(k)})
\end{bmatrix}
\begin{bmatrix}
\cos(\omega^{(k)}) & 0 & \sin(\omega^{(k)}) \\
0 & 1 & 0 \\
-\sin(\omega^{(k)}) & 1 & \cos(\omega^{(k)})
\end{bmatrix}
\begin{bmatrix}
0 \\
0 \\
1
\end{bmatrix}.
\]
which also reduce the dimension of the data. While each sample of $x^{(k)}$, $k = 1, \ldots, N_k$, is reshaped to form a $1 \times L' \times 2$ sample, the angular position is used as a separate input for FC layers. In other words, the output of the convolutional layers is flattened, and then, it is concatenated with the data obtained from the gyroscope sensor on the smart device (i.e., $\delta^{(k)}$, $\rho^{(k)}$, $\omega^{(k)}$). Finally, the resulting vector is used as inputs of FC layers which deliver the estimated location ($\tilde{x}^{(k)}$, $\tilde{y}^{(k)}$, $\tilde{z}^{(k)}$).

It is worth mentioning that the training procedure requires a considerable amount of energy, and cannot be implemented in smart devices, therefore, the offline stage is performed in the CCM unit instead of the user equipment. When the user enters the smart indoor area, the architecture of the CNN model along with the trained weights are transmitted to the user through VLC links.

B. ONLINE MODE DESIGN

In the online mode design, the user utilizes the trained CNN to estimate its absolute position using the received VLC signal and the data obtained from smart device sensors as inputs. In case of VLC blockage, the inertial data of the smart device is used to estimate the user’s movement. The accelerometer in smart device is able to measure the acceleration and direction of the user’s movement. Using the measured acceleration of user $u_m$, denoted by $\alpha_{u_m}$, the number of steps, represented by $n_{step}^{(m)}$, as well as the direction of each step, denoted by $\mathbf{g}_{u_m}^{(m)}$, the user’s movement can be estimated. The size of $i^{th}$ step, represented by $\ell_{u_m}^{(m)}$, can be estimated as a function of the acceleration at $i^{th}$ step as $\ell_{u_m}^{(m)} = \gamma(\alpha_{u_m})$. Therefore, the position of user $u_m$ after $i^{th}$ step from initial position $L_{u_m}$ can be estimated as follows:

$$L_{u_m}^{(i+1)} = L_{u_m}^{(i)} + \ell_{u_m}^{(m)} \times \mathbf{g}_{u_m}^{(m)},$$

where $L_{u_m}^{(i)}$ is the position of the user at $i^{th}$ step.

IV. PROPOSED HANDOVER ALGORITHM BASED ON INDOOR POSITIONING IN HYBRID VLC/MMWAVE

In this section, we aim to apply the proposed indoor localization algorithm introduced in Section III to identify the nearest VLC transmitter to the user during the handover process. Toward this goal, we first form a Handover VLC List, denoted by $\text{HVL}_{u_m}$, which is composed of VLC candidates providing services for the user $u_m$. In other words, the VLC $T_{v_i} \in \mathbb{T}$ in $\text{HVL}_{u_m}$ is selected if the user $u_m$ is located in the communication range of VLC $T_{v_i}$, expressed as:

$$\text{HVL}_{u_m}(t) = \{T_{v_i} \in \mathbb{T} \mid d_{T_{v_i}, u_m}^h \leq T\},$$

where $t$ refers to the updated iteration and $T$ is a threshold value related to the communication range of each VLC. In addition, $d_{T_{v_i}, u_m}^h$ is the horizontal distance between VLC transmitter $T_{v_i}$ and user $u_m$ that indicates user $u_m$ is located in the region $\mathcal{R}_n$ related to transmitter $T_{v_i}$. In this situation, the transmitter with the minimum $d_{T_{v_i}, u_m}^h$ in the set $\text{HVL}_{u_m}(t)$ is selected as the current serving VLC transmitter, denoted by $T_{v_i}^c$, expressed as:

$$T_{v_i}^c = \arg\min_{T_{v_i} \in \text{HVL}_{u_m}} d_{T_{v_i}, u_m}^h.$$  

Whenever the handover process is invoked during the user’s movement, we update $\text{HVL}_{u_m}$, and pick another transmitter with the minimum $d_{T_{v_i}, u_m}^h$, as the cooperative VLC transmitter, denoted by $T_{v_i}^c$, where $T_{v_i}^c \neq T_{v_i}^s$.

In the second stage, HOM and WT are dynamically set based on the change in the SNR value at the receiver during two consecutive time slots to avoid successive handovers. The proposed handover algorithm is summarized in Algorithm 2, which includes two types of handovers: hard handover and CoMP-based soft handover. In the first case, the current serving VLC transmitter terminates its communication with the user, and an adjacent transmitter, known as cooperative VLC transmitter, communicates with the same user whenever the handover is triggered. In the second case, the user connects to the cooperative VLC transmitter, while its connection with the current serving VLC transmitter is still active. Denoting $t_{\text{CoMPHO}}$ and $t_{\text{HardHO}}$ as the waiting time before the hard handover and CoMP-based soft handover, respectively, the proposed handover process in Algorithm 2 is described as the following four steps:

- **Step 1 (Initialization):** First, we initialize the algorithm by setting $a = 0$, $b = 0$, $t_{\text{CoMPHO}} = 0$, $t_{\text{HardHO}} = 0$, and $t = 0$, where $a$ and $b$ are defined in Section IV-B (See lines 1-3).

- **Step 2 (Serving VLC selection):** In this step, we determine the serving VLC transmitter by forming the set $\text{HVL}_{u_m}$ as in (19). Then, the serving VLC transmitter is chosen according to (20) (See lines 4-6).

- **Step 3 (Cooperating VLC selection):** Whenever the handover is invoked, the set $\text{HVL}_{u_m}(t)$ is updated. Then, the VLC transmitter in set $\text{HVL}_{u_m}(t)$ with the minimum $d_{T_{v_i}, u_m}^h$ is chosen as the cooperating VLC transmitter (see lines 7-10).

- **Step 4 (Handover type selection):** The type of handover is determined based on $\text{HOM}(t)$ and $\text{WT}(t)$ as follows: First, it is assumed that $\text{HOM}(t) = 0$ dB and $\text{WT}(t) = 0$ s. Then, $\text{HOM}(t)$ and $\text{WT}(t)$ are updated based on the change of SNR values, denoted by $\gamma_{\Delta T} = \frac{\gamma_{T_{sv}^{0}} - \gamma_{T_{sv}^{(t-1)}}}{t(t-1)}$, and the inverse of $\gamma_{\Delta T}$, respectively. The higher the value of $\Delta T$, the higher the HOM value and the lower the WT value should be set to minimize unnecessary handovers (See lines 11-12). Finally, the handover type is chosen based on the following conditions: i) the hard handover is performed if $\gamma_{T_{sv}^{0}} \geq \gamma_{T_{sv}^{r}} + \text{HOM}$ and $t_{\text{HardHO}} \geq \text{WT}$, ii) the CoMP-based soft handover is performed if $\gamma_{T_{sv}^{0}} \geq \gamma_{T_{sv}^{r}} + \text{HOM}$, $t_{\text{HardHO}} < \text{WT}$, and $t_{\text{CoMPHO}} \geq \text{WT}$ or $\gamma_{T_{sv}^{0}} + \text{HOM} \geq \gamma_{T_{sv}^{r}}$ and $t_{\text{CoMPHO}} \geq \text{WT}$, otherwise, the user connects to the mmWave RF transmitter.

- **Output:** We can determine the type of handover (hard handover or CoMP-based soft handover) and the VLCs
to which the user connects whenever the handover process is invoked. In addition, the user connects to the mmWave RF transmitter if the VLC channels are not accessible. Moreover, the threshold values for HOM and WT are dynamically set in each time slot. By doing so, the ping-pong effect caused by the successive handovers is prevented.

To analyze the performance of the proposed handover algorithm based on the VLC indoor positioning, we derive a closed-form expression for Average Effective Throughput (AET), and then we investigate the dynamical adjustment of HOM and WT to prevent the ping-pong effect caused by the user’s speed during its movement. Finally, the computation complexity of the proposed handover algorithm is calculated.

### A. AVERAGE EFFECTIVE THROUGHPUT ANALYSIS

To analyze the performance of the proposed handover algorithm, we derive a closed-form expression for the Average Effective Throughput (AET). AET is defined as the expectation of conditional probability of the effective number of symbols transmitted successfully given packet success rate probability, computed as \( c \times r_c \), multiplied by packet success rate probability, denoted by \( P_{\text{succ}}^{(\ell)} = 1 - P_e \), where \( P_e \) represents the Packet Error Rate (PER). Accordingly, AET is expressed as:

\[
\eta^{(\ell)} = \mathbb{E} \left[ \frac{c \times r_c \times P_{\text{succ}}^{(\ell)}}{N_A} \right], \quad \ell \in \{1, 2\},
\]

where superscript \( (\ell) \) refers to the type of the handover process during the user movement, i.e., \( \ell = 1 \) refers to the hard handover, while \( \ell = 2 \) shows the CoMP-based soft handover. Further, \( c = \log_2(M) \) and \( r_c \) represent the number of transmitted symbols and the code rate, respectively. In addition, \( N_A \) represents the average number of packet retransmissions. Note that the users send 1 bit Acknowledgment (ACK) whenever they successfully receive their requested packets, otherwise, they send Negative ACK (NACK) to the VLC transmitter. In case of NACK, the VLC transmitter retransmits the missing packet to the user. Employing fountain code paves the way for the transmission, meaning that the fountain decoder has the ability to recover the block of \( K \) encoded data packets after receiving the block of an encoded packet with size \( M, M > K \). Finally, the user sends an ACK when the whole block of packets is received correctly. In the following two steps, we determine the parameters involved in computing the average effective throughput. Then, in the third step, we derive a closed-form expression for the average effective throughput.

1) PACKET ERROR RATE

Generally, the PER of the link between transmitter \( T_v \) and user \( u_m \), denoted by \( \text{PER}_{T_v, u_m} \), is proportional to Q function, i.e., \( \nu Q(\sqrt{\sigma^2}) \) with constant values \( \nu \) and \( \sigma \). However, there is not a closed-form expression for PER in coded modulations, thus, we utilize the upper bound for \( \text{PER}_{T_v, u_m} \), expressed as:

\[
\text{PER}_{T_v, u_m} = \frac{\nu \exp\left(-\frac{\sigma \gamma_{T_v, u_m}}{2}\right)}{2}, \quad \nu \gamma_{T_v, u_m} \geq \gamma_h,
\]

otherwise, \( \gamma_h = \frac{1}{\sigma} \ln(\nu) \) refers to the threshold SNR value. Next, the PER relates two types of handover is obtained.

**Case 1:** In the first type of handover, the user connects to only one VLC transmitter when the handover is triggered. In this case, the PER is computed as

\[
\text{PER}^{(1)} = \text{PER}_{T_v, u_m},
\]
Case 2: In this case, the user connects to two VLC transmitters simultaneously. In other words, the user is served by the serving VLC transmitter $T_v^s$, while the cooperating VLC transmitter $T_v^c$ collaborates in transmitting the signal to the user, and the PER is computed as follows:

$$PER^{(2)} = PER_{T_v^s, u_m} \times PER_{T_v^c, u_m}.$$ (24)

In fact, in the second case, the error occurs whenever neither $T_v^s$ nor $T_v^c$ transmit the signal.

2) CODING MODE SELECTION

We describe the coding mode selection consisting of code rate and modulation in two handover cases:

Case 1: In this case, we utilize adaptive coding mode selection to increase the average effective throughput. Each mode $(i)$ is selected if the following inequality is satisfied:

$$\gamma_j^{(i)} \leq \gamma_{\theta_{th}, u_m} \leq \gamma_j^{(i+1)}, \quad i \in [1, \ldots, N_{\text{mode}}],$$ (25)

where $N_{\text{mode}}$ refers to the number of coding modes. In addition, $\gamma_j^{(i)}$ is considered as an ascending order so that the inequality $\gamma_j^{(j)} \leq \gamma_j^{(j+1)}$, $\forall j \in [1, N_{\text{mode}} - 1]$ is satisfied. More precisely, no transmission is occurred for $\gamma \leq \gamma_j^{(1)}$. Moreover, $\gamma_j^{(N_{\text{mode}}+1)} = \infty$ indicates that the coding mode is chosen for each value of $\gamma \in [\gamma_j^{(1)}, +\infty)$.

Case 2: In this case, the coding mode is selected such that the average effective throughput $\eta^{(2)}(i)$, defined in (29), is maximized, i.e.,

$$i = \arg \max_{1 \leq i \leq N_{\text{mode}}} \eta^{(2)}(i).$$ (26)

3) EFFECTIVE THROUGHPUT DERIVATION

Recalling (6), we rewrite the channel gain of direct link between $j$th LED of the transmitter $T_v$ and $j$th PD of the IMD $u_m$ as follows:

$$L_{in,j} = \frac{(w+1)g(\phi_{in,j})A_rH_d^{w+1}}{2\pi(H_d^2 + r_d^2)^{w+2}},$$ (27)

where $\cos(\theta_{in,j}) \simeq \cos(\phi_{in,j}) = \frac{H_d}{\sqrt{H_d^2 + r_d^2}}$, in which $H_d$ and $r_d$ denote the vertical and horizontal distances between $j$th LED of transmitter $T_v$ and $m$th user, respectively. In fact, the user only changes the horizontal direction of its smart device during its movement. Moreover, we utilize the Random Way Point (RWP) model for the 2D user’s movement, expressed as follows:

$$f_{r}(r) = \sum_{x=1}^{n} \gamma_x r_x^{\beta_x} \frac{\beta_x}{R_p^{\beta_x+1}},$$ (28)

where $f_{r}(r)$ is the probability distribution function (pdf) of the user’s movement in 2D plane. Generally, for such a model, $n = 3, \gamma = \frac{1}{73} \times [324, -420, 94]$, and $\beta = [1, 3, 5]$ are considered [36]. Substituting (7) in (27), $L_{in,j}$ is re-written as follows:

$$L_{in,j} = \begin{cases} \frac{C_r}{r_d^2 \gamma_{\theta_{th}, u_m}}, & 0 \leq \phi_{in,j} \leq \phi_c \\ 0, & o.w. \end{cases}$$ (29)

where $C_r = \frac{(w+1)A_rH_d^{w+1}}{2\pi} \frac{(n_{\text{in,j}})^2}{N_{\text{in,j}}}$. To calculate the pdf of the $L_{in,j}$, it is more straightforward to compute its Cumulative Distribution Function (CDF) based on (29), expressed as (30), as shown at the bottom of the next page, where $(a)$ comes from the fact that the cosine function is a monotonically decreasing function on the interval $(0, \pi)$. Then, $Pr[L_{in,j} \leq \ell]$ is re-expressed as (31), as shown at the bottom of the next page, where $(a)$ comes from the formula $P(X \cap Y) = P(X)P(Y)$ for two independent random variables $X$ and $Y$, and $u(\ell)$ refers to the unit step function. Using the fact that $Pr[X \geq x] = 1 - F_X(x)$ and $Pr[a \leq X \leq b] = F_X(b) - F_X(a)$, the CDF expression of $F_{L_{in,j}}(\ell) = Pr[L_{in,j} \leq \ell]$ is re-written as (32), as shown at the bottom of the next page. In the next step, we take derivative from the CDF of $L_{in,j}$ with respect to $\ell$ to compute its pdf, denoted by $f_{L_{in,j}}(\ell)$. Using the formula $\frac{\partial}{\partial \ell} F_X(\sqrt{\ell}) = \frac{X'}{2X} f_x(\sqrt{\ell})$, we have (33), as shown at the bottom of the next page, where in step $(a)$, we use $f_{r}(r) = \sum_{x=1}^{n} \gamma_x r_x^{\beta_x} \frac{\beta_x}{R_p^{\beta_x+1}}$ as the pdf of the random way point model for the user movement. Applying the binomial theorem defined as $\left( \left( \frac{C_r}{\ell} \right)^{\frac{r_d^2}{\gamma_{\theta_{th}, u_m}}} - H_d^2 \right)^{2\ell_{\frac{r_d^2}{\gamma_{\theta_{th}, u_m}}}} = \sum_{j=0}^{\ell} (\frac{\ell_{\frac{r_d^2}{\gamma_{\theta_{th}, u_m}}}}{H_d^2})^{\frac{r_d^2}{\gamma_{\theta_{th}, u_m}}} - j \left( \frac{C_r}{\ell} \right)^{2j}$, the pdf $f_{L_{in,j}}(\ell)$ is re-expressed as follows:

$$f_{L_{in,j}}(\ell) = \sum_{x=1}^{n} A_x \sum_{j=0}^{\ell} (-H_d^2)^{\frac{r_d^2}{\gamma_{\theta_{th}, u_m}}} - j \left( \frac{C_r}{\ell} \right)^{2j} \frac{\ell_{\frac{r_d^2}{\gamma_{\theta_{th}, u_m}}}}{\ell_{\frac{r_d^2}{\gamma_{\theta_{th}, u_m}}} - x}$$

$$+ \delta(\ell) \left( 1 - F_r \left( H_d \tan(\phi_c) \right) \right),$$ (34)

where $A_x = \frac{\gamma_x C_r^2}{(w+1)R_p^{\beta_x+1}}$. It is very complex to consider all LEDs of a VLC transmitter and all PDs of a user in a VLC system to compute $\gamma_{in,j}$. Therefore, we assume that the VLC system contains one LED as a VLC transmitter and one PD in the receiver, which is a practical assumption due the short distance between LEDs of a transmitter or between PDs of a receiver. Accordingly, the pdf distribution of $\gamma_{in,j} = \tilde{\gamma} \ell^2$ is computed as (35), as shown at the bottom of the next page, where $\kappa_j = \frac{1}{\ell^2}(\frac{(-H_d^2)^{\frac{r_d^2}{\gamma_{\theta_{th}, u_m}}}}{\ell} - C_r^{\frac{2j}{\gamma_{\theta_{th}, u_m}}})$. Finally, we obtain the average effective throughput as the expectation of conditional probability of the effective number of signals transmitted successfully given the pdf of $\gamma_{in,j}$, expressed as (36), as shown at the bottom of the next page, where $(a)$ comes from the formula $\eta(\ell) = \mathbb{E} \left[ \frac{c_{\gamma,j} \gamma_{in,j}}{N_{\gamma,j}} \right], \quad \ell \in \{1, 2\}$. Then $\eta(\ell)$ is re-expressed as (37), as shown at the bottom of the next page, where $Gamma(s, x)$ is an incomplete gamma function, defined
as \( \Gamma(s, x) = \int_x^{+\infty} t^{s-1} e^{-t} \, dt \). With a similar argument as mentioned above, \( \eta^{(2)} \) is computed by Wolfram|Alpha as \( \frac{C_r}{(H_d^2 + r_d^2)^{\frac{w+3}{2}}} \leq \ell, \; \frac{H_d}{\sqrt{(H_d^2 + r_d^2)^{\frac{w+3}{2}}}} \geq \cos \phi_c \) \( \text{Pr} \{ \ell \geq 0, \frac{H_d}{\sqrt{(H_d^2 + r_d^2)^{\frac{w+3}{2}}}} \leq \cos \phi_c \} \), where \( \arg \max_{1 \leq j \leq \text{Node}} \eta^{(2)} \). Moreover, \( \text{PER}_{\ell} = \text{PER}_{T_i,a_n} \times \text{PER}_{T_j,a_n} \) is hold in step (b).

\[ F_{L_{\text{in}, j_n}} (\ell) = \text{Pr} [L_{\text{in}, j_n} \leq \ell] = \text{Pr} \left\{ \frac{C_r}{(H_d^2 + r_d^2)^{\frac{w+3}{2}}} \leq \ell, \; 0 \leq \phi_{\text{in}, j_n} \leq \phi_c \} + \text{Pr} \{ \ell \geq 0, \phi_{\text{in}, j_n} \geq \phi_c \} \right\} \]

\[ \implies \text{Pr} \left\{ \frac{C_r}{(H_d^2 + r_d^2)^{\frac{w+3}{2}}} \leq \ell, \; \cos \phi_{\text{in}, j_n} \geq \cos \phi_c \} + \text{Pr} \{ \ell \geq 0, \cos \phi_{\text{in}, j_n} \leq \cos \phi_c \} \right\} \]

\[ \leq \text{Pr} \left\{ \frac{C_r}{(H_d^2 + r_d^2)^{\frac{w+3}{2}}} \leq \ell, \; \cos \phi_{\text{in}, j_n} \geq \cos \phi_c \} + \text{Pr} \{ \ell \geq 0, \cos \phi_{\text{in}, j_n} \leq \cos \phi_c \} \right\} \]

**B. DYNAMICALLY SETTING OF HOM AND WT**

In the final step of the handover analysis, we investigate the performance of the proposed algorithm based on HOM and WT. As mentioned before, we define \( \text{HOM}(t) \) as the rate of the change in the SNR value in two consecutive time slots multiplied by constant value \( a \). In addition, \( \text{WT}(t) \) is defined.
as the constant value $b$ divided by the rate of the change in SNR value in two sequential time slots, both expressed as:

$$\text{HOM}(t) = a \times \left( \gamma_{T'_v,u_m}(t) - \gamma_{T'_v,u_m}(t-1) \right), \quad (39)$$

$$\text{WT}(t) = \frac{b}{\gamma_{T'_v,u_m}(t) - \gamma_{T'_v,u_m}(t-1)}, \quad (40)$$

where $t$ refers to the current time slot. In fact, HOM$(t)$ should be proportional to $(\gamma_{T'_v,u_m}(t) - \gamma_{T'_v,u_m}(t-1))$ to ensure that $T'_v$ is appropriate enough for the handover process during the user’s movement at either high or slow speed. However, WT$(t)$ is proportional to the inverse of $(\gamma_{T'_v,u_m}(t) - \gamma_{T'_v,u_m}(t-1))$, meaning that if the user moves away fast, WT$(t)$ should be small to handle the handover rapidly, otherwise, the connection between the user and the VLC drops for a long time.

### C. Computational Complexity Analysis of Algorithm 2

In this subsection, we present the computational complexity analysis of Algorithm 2 by the following proposition.

**Proposition 1:** The computational complexity of Algorithm 2 is of order $O(t\text{PL})$, in which $t$ represents the total number of time-slots.

**Proof:** Regarding the pseudocode in Algorithm 2, there is only one “for” loop. More precisely, the computational complexity of Algorithm 2 mainly depends on lines 10 of the algorithm, where $\gamma_{T'u_m,u_m}$ is computed. To this end, we first obtain the received signal at user $u_m$, where its computational complexity is of order $O(P\text{L})$. Then, we apply the Tarokh scheme on the received signal, as expressed in lines 10 and 11 of Algorithm 2, and its computational complexity is of order $O(P\text{L})$. After detecting the signal transmitted from the VLC transmitter, $\gamma_{T'u_m,u_m}$ is calculated as in Eq. (12), where its computational complexity is of order $O(P\text{L} + P)$. Accordingly, the overall computational complexity related to line 10 of Algorithm 2 is calculated as $O(3P\text{L} + P)$.

The corresponding computational complexity for other lines is of order $O(1)$. Suppose that $t$ is total time slots in the “for” loop. Accordingly, the total computational complexity of Algorithm 2 is calculated by multiplying the aforementioned complexity terms by $t$, i.e., $\max(O(3t\text{PL} + tP))$, or equivalently, of order $O(t\text{PL})$.

### V. Simulation Results

In this section, we evaluate the performance of the proposed indoor localization algorithm and compare it with the existing fingerprinting algorithms, namely, Multi-Layer Perceptron (MLP) [22], Artificial Neural Network (ANN), WKNN [19], and kRRML [18] in terms of the positioning accuracy. Furthermore, we compare the virtual COMP-based handover algorithm with two other algorithms, i.e., soft handover [37] and hard handover [38].

### A. Simulation Setup

We consider a smart living room with size $5 \times 4 \times 3$ m$^3$ consisting of $N_{T} = 4$ VLC transmitters. They are positioned on the ceiling, where each transmitter has $L = 4$ LEDs spaced out evenly with distance of 5 centimeters. The coordinates of the center of each VLC transmitter is shown in Table 2. Each LED has the semi-angle of half power at $\theta_{1/2} = 60^\circ$ resulting in Lambertian factor $w = \frac{\ln(2)}{\ln(\cos(\Phi_c))} = 2$. In addition, each user $u_m$, as a receiver, has $P = 4$ PDs positioned on the smart device where each PD has FoV of $\Phi_c = 85^\circ$ and area of $A_r = 1$ cm$^2$. Moreover, the receiver (e.g., smartphone) is equipped with a gyroscope that measures the angle between the device and the floor. The noise characteristics related to the VLC-based environment along with other simulation parameters are listed in Table 2.

Regarding the mobility of the users, we consider the random way point model, described in (28), where the user moves with average velocity $v_u = 2$ m/s with direction in range $[-\pi, \pi]$ for 1000 seconds. Although the magnitude of

$$\eta^{(2)} = \mathbb{E} \left[ b^+_v r_{c^*} (1 - P_e^{(2)}) \right] = \int_{\gamma_{c^*}}^{+\infty} \int_{\gamma_{c^*}}^{+\infty} b^+_v r_{c^*} (1 - P_e^{(2)}) \gamma_{c^*} \mathbb{J}_{\gamma_{c^*}} \gamma_{c} \mathbb{J}_{\gamma_{c}} \gamma_{c^*} \mathbb{J}_{\gamma_{c^*}} \gamma_{c} \mathbb{J}_{\gamma_{c}}$$

$$= b^+_v r_{c^*} + \sum_{x=1}^{n} A_x \sum_{j=0}^{\theta_{1/2} - 1} b^+_v r_{c^*} \gamma_{c^*}^{\frac{v_{c^*}}{w} + \frac{1}{2}} \gamma_{c}^{\frac{v_{c}}{w} + \frac{1}{2}} \left( \gamma_{c^*}^{\frac{v_{c^*}}{w} + \frac{1}{2}} \gamma_{c}^{\frac{v_{c}}{w} + \frac{1}{2}} \right)$$

$$+ \sum_{x=1}^{n} A_x \sum_{j=0}^{\theta_{1/2} - 1} b^+_v r_{c^*} \gamma_{c^*}^{\frac{v_{c^*}}{w} + \frac{1}{2}} \gamma_{c}^{\frac{v_{c}}{w} + \frac{1}{2}} \left( \gamma_{c^*}^{\frac{v_{c^*}}{w} + \frac{1}{2}} \gamma_{c}^{\frac{v_{c}}{w} + \frac{1}{2}} \right)$$

$$+ \sum_{x=1}^{n} A_x \sum_{j=0}^{\theta_{1/2} - 1} b^+_v r_{c^*} \gamma_{c^*}^{\frac{v_{c^*}}{w} + \frac{1}{2}} \gamma_{c}^{\frac{v_{c}}{w} + \frac{1}{2}} \left( \gamma_{c^*}^{\frac{v_{c^*}}{w} + \frac{1}{2}} \gamma_{c}^{\frac{v_{c}}{w} + \frac{1}{2}} \right)$$

$$\times \left( \sum_{x=1}^{n} A_x \sum_{j=0}^{\theta_{1/2} - 1} b^+_v r_{c^*} \gamma_{c^*}^{\frac{v_{c^*}}{w} + \frac{1}{2}} \gamma_{c}^{\frac{v_{c}}{w} + \frac{1}{2}} \left( \gamma_{c^*}^{\frac{v_{c^*}}{w} + \frac{1}{2}} \gamma_{c}^{\frac{v_{c}}{w} + \frac{1}{2}} \right) \right)$$

$$= \sum_{x=1}^{n} A_x \sum_{j=0}^{\theta_{1/2} - 1} b^+_v r_{c^*} \gamma_{c^*}^{\frac{v_{c^*}}{w} + \frac{1}{2}} \gamma_{c}^{\frac{v_{c}}{w} + \frac{1}{2}} \left( \gamma_{c^*}^{\frac{v_{c^*}}{w} + \frac{1}{2}} \gamma_{c}^{\frac{v_{c}}{w} + \frac{1}{2}} \right)$$

$$\times \left( \sum_{x=1}^{n} A_x \sum_{j=0}^{\theta_{1/2} - 1} b^+_v r_{c^*} \gamma_{c^*}^{\frac{v_{c^*}}{w} + \frac{1}{2}} \gamma_{c}^{\frac{v_{c}}{w} + \frac{1}{2}} \left( \gamma_{c^*}^{\frac{v_{c^*}}{w} + \frac{1}{2}} \gamma_{c}^{\frac{v_{c}}{w} + \frac{1}{2}} \right) \right), \quad (38)$$
the speed is constant, the direction of the user’s movement is time-variant, and the walking time in a specific direction is considered as a random variable in the range $[1, 2]$ seconds. Unlike existing literature, we propose a CoMP-based handover process based on HOM and WT before the handover process, which are dynamically set whenever the handover is triggered. In other words, HOM is calculated as the rate of change in SNR value in two successive time-slots multiplied by $a = 0.1$. In addition, WT is computed as the inverse of the rate of change in SNR value in two consecutive time slots by $b = 0.1$.\[37, 38\].

### B. EVALUATION OF PROPOSED INDOOR LOCALIZATION ALGORITHM

Considering the aforementioned setup, we first simulate the proposed indoor localization algorithm, and evaluate its performance by measuring the mean positioning error on the test set. We use Algorithm 1 to generate the required data sets, i.e., training set and test set, then, we simulate the proposed CNN architecture for indoor localization in Fig. 2. All the data sets in our simulation are collected using MATLAB, while, deep learning and machine learning packages in Python, namely, Keras and Scikitlearn [39] are used to train the estimation models. To generate the training data in the offline stage, we divide the smart living room into equal-sized elements with $d_x \times d_y \times d_z$, and save the 3D coordinates of the center of each element in $y^{(k)}$ as described in lines 1-12 of Algorithm 1. In addition, we randomly choose 10 percent of the training set as the validation set which is considered a useful means to tune the hyperparameters of the model. Among different patterns for generating the test set, we choose the interpolation between the training points as they provide the worst case scenario for evaluating the algorithm since they have the maximum mean distance from the adjacent training samples. Fig 4(a) illustrates the training and test points on the X-Y plane of a 1 m $\times$ 1 m area at the corner of the room at the height around $z = 1$ m with $d_x = d_y = d_z = 20$ cm, while Fig. 4(b) shows the same points from the X-Z plane view.

For both the training and the test sets, the LoS gain of the VLC channels, $L_{i,j,n}^{(k)}$, and the ToA, $t_{i,j,n}^{(k)}$, are calculated according to (6) and (16), and stored in $X^{\text{train}}$ and $X^{\text{test}}$, respectively (line 29 of Algorithm 1). In addition, $Y^{\text{train}}$ and $Y^{\text{test}}$ including the 3D coordinates of samples are obtained according to line 3 of Algorithm 1. Finally, two data sets, denoted by $D^{\text{train}}$ and $D^{\text{test}}$ are made (line 31). The training set, $D^{\text{train}}$, is given to a CNN consisting of five convolutional layers followed a max pooling layer and a fully-connected layers with 256 neurons. The structure of the CNN employed in the proposed indoor localization algorithm is given in Table 3. The hyperparameters of the proposed model, i.e., the number of convolution and hidden (fully-connected) layers, are selected after testing several values since too many layers might lead to overfitting, while insufficient layers result in a low degree of freedom and a high positioning error. For the training process, we use a well-known technique, namely, error backpropagation. Moreover, the batch size and the number of epochs are set to 32 and 30, respectively. The R2 regularization is used to avoid overfitting in the proposed model. Fig. 3 demonstrates the learning curve of the proposed CNN model. As it can be seen from the figure, the loss function, i.e., mean square error, related to the training set and validation set, converges to a fixed and small value which shows the generalization of the CNN model.

Then, the mean 3D positioning error of the test set is calculated as follows:

$$MPE = \frac{1}{N^{\text{test}}} \sum_{k=1}^{N^{\text{test}}} \sqrt{\|y^{(k)} - \tilde{y}^{(k)}\|^2},$$ (41)
S. M. Sheikholeslami et al.: Sub-Decimeter VLC 3D Indoor Localization

FIGURE 4. a) Training points and b) test points, at the corner of the living room for a 1 m × 1 m area and height around $z = 1$ with $d_x = d_y = d_z = 20$ cm.

where $\hat{y}^{(k)}$ is the predicted location of $k^{th}$ test sample and $N_{\text{test}}$ represents the total number of test samples. The simulation results show that the algorithm reaches a mean positioning error of 4.31 centimeters over the test set. Moreover, more than 95.7% of test samples has a position error less than 10 cm for element size $d_x = d_y = d_z = 5$ cm.

To investigate the impact of the element sizes $(d_x, d_y, d_z)$ on the accuracy of the proposed indoor localization, Fig. 5 compares the CDF of position error for different sizes $d_x = d_y = d_z \in \{5, 10, 15, 20, 30\}$. The results show that the CDF curve gets a sharper ramp for a smaller element size meaning that the position error decreases as the element size is reduced. The mean positioning error on the test set for the aforementioned step sizes are 4.31, 11.15, 14.17, 18.21, and 24.50 centimeters, respectively.

In order to evaluate the proposed algorithm, we compare the CNN model with four common localization methods, i.e., MLP [22], ANN, WKNN [19], and KRRML [18]. For sake of fair comparison, we use the same training and test data sets $(d_x = d_y = d_z = 5)$ cm). As shown in Fig. 6, the proposed indoor localization algorithm outperforms the other four schemes in terms of position error with the mean positioning error of 4.31 centimeters. Among the other four methods, MLP has the lowest localization error with mean error of 12.81 centimeters, while forming an ANN with one hidden layer results in a 20.03 centimeter positioning error. Lastly, WKNN and KRRML provide the localization error of 52.39 and 80.23 centimeters which are not acceptable for 3D indoor localization.

Figure 7 illustrates the Probability Distribution Function (PDF) of the proposed VLC indoor localization algorithm and the alternative techniques. As shown in this figure, the proposed CNN architecture for 3D indoor localization reaches sub-decimeter positioning error for a majority of test samples. More precisely, 95.7% of test samples have 3D positioning error less than 10 centimeters with the proposed CNN-based indoor localization algorithm, while 50.0% of the test samples lie within 10 centimeter error with the MLP method. This percentage is 15.1% and 24.2% for ANN and WKNN schemes, respectively. Additionally, none of the test samples reach sub-decimeter accuracy with the KRRML algorithm. The mean and standard deviation of the 3D positioning error are reported in Table 4.

We also compare the complexity of the proposed indoor localization CNN algorithm presented in the block diagram of Figure 2 with MLP [22], ANN, KRRML [18], and WKNN [19] methods. The indoor localization algorithm contains offline and online stages. Similar to [22], we only
### TABLE 4. Mean and standard deviation of positioning error for different algorithms.

| Algorithm  | Mean error (cm) | Standard deviation of error (cm) |
|------------|----------------|-------------------------------|
| proposed CNN | 4.31           | 2.86                          |
| MLP [22]   | 12.81          | 9.61                          |
| ANN        | 20.03          | 12.80                         |
| WKNN [19]  | 52.39          | 61.63                         |
| KRRML [18] | 80.23          | 21.86                         |

### TABLE 5. Complexity comparison.

| Algorithm         | No. of Trainable Parameters | Positioning time (s) |
|-------------------|-----------------------------|----------------------|
| proposed CNN      | 190,371                     | $3.9456 \times 10^{-5}$|
| MLP [22]          | 97,583                      | $2.8347 \times 10^{-5}$|
| ANN               | 18,623                      | $2.3872 \times 10^{-5}$|
| WKNN [19]         | -                           | $51.6619 \times 10^{-5}$|
| KRRML [18]        | -                           | $9.0182 \times 10^{-5}$|

Consider the online stage for evaluation of the complexity because the offline stage is a one-time run and occurs in the CCM, while the online stage is what the users employ every time they want to be localized. Note that Algorithm 1 is not considered in our computations since it is a part of the offline stage. Moreover, Algorithm 1 is used only to generate the data required to train and evaluate the algorithms including the proposed algorithm and the alternative methods (i.e., MLP, ANN, WKNN, and KRRML). A decent criterion for comparing the complexity of deep learning and machine learning techniques is the computation time. Table 5 compares the computation time and the number of trainable parameters of the proposed indoor localization technique with the alternative MLP, ANN, WKNN and KRRML techniques. ANN provides the lowest positioning time with the minimum number of trainable parameters. On the other hand, the proposed CNN algorithm has the highest number of trainable parameters, and it is slightly slower than MLP and ANN in the online mode. Moreover, WKNN and KRRML have the highest positioning time.

### C. HANDOVER SIMULATIONS

In this part, we evaluate the handover process during the user’s movement according to the proposed indoor localization algorithm. In Fig. 8, we investigate the BER of the proposed CoMP-based handover versus the average SNR in the range [0, 30] dB for different schemes in Fig. 8. Besides, we apply the turbo code as a fixed-rate code and fountain code in our proposed scheme. Compared to conventional CoMP-based soft handover and hard handover, BER in the proposed dynamic handover is improved by at least 25% and 12%, respectively. This superiority comes from the fact that we can substantially prevent the ping-pong effect during the handover process by applying the proposed algorithm. Compared to CoMP-based soft handover algorithm, the SNR values vary considerably during the hard handover process, leading to an increase in BER [40]. In addition, the proposed algorithm outperforms the soft handover process because we dynamically set HOM and WT based on the status of VLC channels in successive time slots, while in the conventional CoMP-based soft handover, the user connects to two VLC transmitters [37]. Accordingly, we propose an algorithm to handle the
handover process without considering the users’ movement speed, and if the VLC channels are inaccessible, the user connects to the mmWave RF transmitter. It is shown that BER can be improved significantly by applying the fountain code compared to the other two cases: i) without applying coding, ii) with turbo code. To implement the fountain code, we first determine the degree distribution of each input signal. Then, the matrix $G$ is generated such that the length of input signals ($K = 1000000$), the length of output signals $N$, and the tolerable error probability ($\delta = 0.3$) are considered as the input for generating matrix $G$. Then, the input signals are multiplied by the matrix $G$ to encode the signal. Modulus-2 addition is applied during the matrix multiplication. At the receiver, the algorithm finds a received signal with degree 1, decodes it, and then, it is removed as a neighbor from all symbols that have it as a neighbor. This process is performed iteratively until all the received signals are decoded. If the size of the encoded signal is not sufficient, the decoding process leads to failure. As shown in Fig. 8, the proposed algorithm with fountain code outperforms the conventional fixed-rate codes, like turbo codes.

In Fig 9, the average effective throughput of the dynamic soft handover versus the average SNR varying from 0 dB to 30 dB is evaluated. As shown, the average effective throughput of the proposed algorithm decreases whenever the average SNR increases. In fact, enhancing the average SNR value of LEDs increases the received SNR at PDs. Thus, the packet successful rate probability improves, and hence, the average effective throughput improves substantially.

For completeness, we investigate HOM and WT before the handover process versus the velocity of the users, varying from 1 m/s to 5 m/s in Fig 10. As seen, the increase of user’ speed leads to enhancing HOM and decreasing the WT to trigger the handover process. In fact, whenever the users move rapidly, WT should be a low value; otherwise, the connection between the user and the VLC transmitter is dropped for a long time. In addition, HOM should be adjusted to a high value because the received SNR at the smart device is changed significantly because of the high speed, and the low value of HOM causes successive handovers. With a similar argument as in the case of high speed, we can perceive that HOM and WT should be adjusted to a low and high value, respectively, when the users move with low speed. To overcome this issue, we propose an algorithm to dynamically adjust HOM and WT based on the rate of changes in the SNR value to prevent the ping-pong effect or SNR reduction during the handover process. Hence, HOM and WT can be adjusted according to the velocity and status of VLC channels, expressed as:

$$HOM = a \times \frac{\left(\gamma_{T_s, u_m}(t) - \gamma_{T_s, u_m}(t - 1)\right)}{b},$$  \hspace{1cm} (43)$$

$$WT = \frac{\left(\gamma_{T_s, u_m}(t) - \gamma_{T_s, u_m}(t - 1)\right)}{b}.\hspace{1cm} (44)$$

The values of $a$ and $b$ are sensitive. Therefore, they should be selected such that BER is minimized during the handover process.

VI. CONCLUSION

This paper considered a smart indoor environment with multiple LEDs placed on the ceiling used as VLC transmitters, and a mmWave AP used as a complementary uplink technology. The users utilized the PDs on their smart phones as VLC receivers. We proposed a CNN-based indoor localization algorithm that uses the LoS channel gain and the ToA as well as the angular position obtained from the gyroscope sensor on the smart device to estimate the 3D location of the user. Moreover, a virtual soft handover based on CoMP transmission was presented, where HOM and WT were set
during the user’s movement according to changes in the SNR value. We derived a closed-form expression for the average effective throughput. Finally, simulation results showed an average positioning error of 4.31 centimeters, while more than 95.7% of the test points had an accuracy less than 10 centimeters. One possible future work is to address more realistic conditions such as shadowing and blind spots and their effects on the accuracy of indoor localization and the handover process in a hybrid RF/VLC system. Toward this goal, one heuristic solution is to employ the new technique called Reconfigurable Intelligent Surface (RIS) in the proposed system model.

REFERENCES

[1] T. S. Rappaport, Y. Xing, O. Kanhere, S. Ju, A. Madanayake, S. Mandal, A. Alkhateeb, and G. C. Chirpoupolou, “Wireless communications and applications above 100 GHz: Opportunities and challenges for 6G and beyond,” IEEE Access, vol. 7, pp. 78729–78757, 2019.

[2] Y. Li, Y.-C. Wang, and E. J. Powers, “A machine learning algorithm with sigmoid function data preprocessing for visible-light-positioning (VLP) system using kernel ridge regression machine learning algorithm with sigmoid function data preprocessing method,” IEEE Access, vol. 8, pp. 214269–214281, 2020.

[3] A. H. A. Bakar, T. Glass, H. Y. Tee, F. Alam, and M. Legg, “Accurate visible light positioning using multiple-photodiode receiver and machine learning,” IEEE Trans. Instrum. Meas., vol. 70, pp. 1–12, 2021.

[4] Y. Chen, W. Guan, J. Li, and H. Song, “Indoor real-time 3-D visible light positioning system using fingerprinting and extreme learning machine,” IEEE Access, vol. 8, pp. 13875–13886, 2020.

[5] S. M. Sheikholeslami, Y. Hou, S. Xiao, M. Bi, Y. Xue, W. Pan, and W. Hu, “‘Single LED beacon-based 3-D indoor positioning using off-the-shelf devices,” IEEE Photon. J., vol. 8, no. 6, pp. 1–11, Dec. 2016.

[6] N. S. M. Arcuri, P. F. Scala, and A. Khreishah, “Indoor localization using visible light via fusion of multiple classifiers,” IEEE Photon. J., vol. 9, no. 6, pp. 1–16, Dec. 2017.

[7] B. Munir and V. Dyo, “Passive localization through light flicker fingerprinting,” IEEE Sensors J., vol. 19, no. 24, pp. 12137–12144, Dec. 2019.

[8] X. Guo, F. Hu, N. R. Ellisplik, and L. Li, “Indoor localization using visible light via two-layer fusion network,” IEEE Access, vol. 7, pp. 16421–16430, 2019.

[9] H. Q. Tran and C. Ha, “High precision weighted optimum K-nearest neighbors algorithm for indoor visible light positioning applications,” IEEE Access, vol. 8, pp. 114597–114605, 2020.

[10] A. R. A. Al-Omran, A. Khreishah, and M. Legg, “Experimental demonstration of 3D visible light positioning using received signal strength with low-complexity stimulation assisted by deep learning technique,” IEEE Access, vol. 7, pp. 93986–93997, 2019.

[11] A. M. Khreishah, M. A. A. H. A. Bakar, T. Glass, H. Y. Tee, F. Alam, and M. Legg, “Accurate visible light positioning using machine learning,” IEEE Access, vol. 8, pp. 13875–13886, 2020.
[38] M. D. Soltani, A. A. Purwita, Z. Zeng, H. Haas, and M. Safari, “Modeling the random orientation of mobile devices: Measurement, analysis and LiFi use case,” IEEE Trans. Commun., vol. 67, no. 3, pp. 2157–2172, Mar. 2018.

[39] A. Géron, Hands-on Machine Learning With Scikit-Learn, Keras, and TensorFlow: Concepts, Tools, and Techniques to Build Intelligent Systems. Newton, MA, USA: O’Reilly Media, 2019.

[40] A. Gupta, P. Garg, and N. Sharma, “Hard switching-based hybrid RF/VLC system and its performance evaluation,” Trans. Emerg. Telecommun. Technol., vol. 30, no. 2, p. e3515, Feb. 2019.

SEYED MOHAMMAD SHEIKHOLESLAMI

received the B.Sc. degree in power engineering from Bu-Ali Sina University, Iran, and the M.Sc. degree in communication systems engineering from the University of Tehran, Iran, in 2017 and 2020, respectively. His research interests include indoor localization, visible light communication, signal processing, and smart grid.

FAHIMEH FAZEL

received the B.Sc. degree in electrical engineering from Amirkabir University of Technology, Tehran, Iran, in 2016, and the M.Sc. degree in communication systems engineering from the University of Tehran, Tehran, in 2019. Her research interests include wireless communication, cache-enabled UAV networks, visible light communication, coding theory, and their applications in visible light communications.

JAMSHID ABOUEI

(Senior Member, IEEE) received the B.Sc. degree (Hons.) in electronics engineering and the M.Sc. degree (Hons.) in communication systems engineering from Isfahan University of Technology (IUT), Iran, in 1993 and 1996, respectively, and the Ph.D. degree in electrical engineering from the University of Waterloo, Canada, in 2009. In 1996, he joined the Department of Electrical Engineering, Yazd University, Iran, as a Lecturer, and was promoted as an Assistant Professor, in 2010, and an Associate Professor, in 2015. From 1998 to 2004, he served as a Technical Advisor and a Design Engineer for the Research and Development Center, Cable Design Department, SGCC, Iran. From 2009 to 2010, he was a Postdoctoral Fellow with the Multimedia Laboratory, Department of Electrical and Computer Engineering, University of Toronto, Canada, and worked as a Research Fellow with Self-Powered Sensor Networks (ORF-SPSN) Consortium. During his sabbatical, he was an Associate Researcher with the Department of Electrical, Computer and Biomedical Engineering, Ryerson University, Toronto, Canada. He currently directs the research group at the Wireless Networking Laboratory (WINEL), Yazd University. His research interests include the next generation of wireless networks (5G) and wireless sensor networks (WSNs), with a particular emphasis on PHY/MAC layer designs, including the energy efficiency and optimal resource allocation in cognitive cell-free massive MIMO networks, multi-user information theory, mobile edge computing, and femtocaching. He is a member of the IEEE Information Theory Society. He received several awards and scholarships, including FOE and IGSA awards for excellence in research from the University of Waterloo; the MSRT Ph.D. Scholarship from the Ministry of Science, Research and Technology, Iran, in 2004; the Distinguished Researcher Award, Yazd, Iran, in 2011; and the Distinguished Researcher Award from the Electrical Engineering Department, Yazd University, in 2013. He was a recipient of the Best Paper Award for the IEEE Iranian Conference on Electrical Engineering (ICEEE 2018). He was the International Relations Chair of the 27th ICEEE2019 Conference, Iran, in 2019.

KONSTANTINOS (KOSTAS) N. PLATANIOTIS

(Fellow, IEEE) received the B.Eng. degree in computer engineering from the University of Patras, Greece, and the M.S. and Ph.D. degrees in electrical engineering from Florida Institute of Technology, Melbourne, FL, USA. He is currently a Professor with The Edward S. Rogers Sr. Department of Electrical and Computer Engineering, University of Toronto, Toronto, ON, Canada, where he directs the Multimedia Laboratory. He has been holding the Bell Canada Endowed Chair in multimedia, since 2014. His research interests include image/signal processing, machine learning and adaptive learning systems, visual data analysis, multimedia and knowledge media, and affective computing. He is a fellow of the Engineering Institute of Canada and Canadian Academy of Engineering, and a Registered Professional Engineer in Ontario. He was the Technical Co-Chair of the IEEE 2013 International Conference in Acoustics, Speech and Signal Processing, and served as the Inaugural IEEE Signal Processing Society Vice President for Membership (2014–2016) and the General Co-Chair for 2017 IEEE GLOBALSIP. He served as the General Co-Chair for the 2018 IEEE International Conference on Image Processing (ICIP 2018) and the IEEE International Acoustics, Speech and Signal Processing (ICASSP 2021). He is also the General Chair of the 2027 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP2027). Toronto. He served as the Editor-in-Chief for the IEEE SIGNAL PROCESSING LETTERS.