PERFORMANCE IMPROVEMENT APPROACHES FOR VISIBLE LIGHT COMMUNICATION

by

Muhammet Yaser YAĞAN

B.S., Electrical Electronics Engineering, Trakya University, 2019

Submitted to the Institute for Graduate Studies in
Science and Engineering in partial fulfillment of
the requirements for the degree of
Master of Science

Graduate Program in Electrical Electronics Engineering

Boğaziçi University

2021

ACKNOWLEDGEMENTS

I would first like to express my sincere gratitude to my supervisor, Assoc. Prof. Ali Emre Pusane for his endless support during my study. Apart from the high academic supervision, his valuable thoughts, positive energy, and patience have been a grand motivation for me. I feel very thankful for having him as my advisor.

I would like to thank Prof. Arda Deniz Yalçınkaya, the executive of 117E058 TÜBİTAK project, which covered a major part of this thesis. His contributions and valuable comments were always motivating the project's team and raising our productivity. I would also like to thank Prof. Serhat Erküçük for participating in my thesis jury.

I would like to thank my colleague at MNCL laboratory, Rifat Kısacık for his advice and practical experience, which have enriched the content of my thesis.

Finally, my deep gratitude to my mother and sisters for their faith in me and endless support during my life, this work would never have been possible without their motivation.

This thesis has been supported by the Scientific and Technological Research Council of Turkey (TÜBİTAK) under project number 117E058.

ABSTRACT

PERFORMANCE IMPROVEMENT APPROACHES FOR VISIBLE LIGHT COMMUNICATION

Visible light communication (VLC) is an emerging technology aiming to use lighting equipment to convey data via modulating light intensity. With the running out spectrum for radio frequency wireless communication, VLC is gaining more attention in the literature and its practical issues are investigated intensely. Two main challenges distinguish VLC from other communication systems: satisfying illumination requirements during data transmission, and the hard channel presented by the transmitters' (usually Light Emitting Diodes (LED)s) limited bandwidth and inter-symbol interference. IEEE 802.15.7 group has defined the required standards for VLC. The standard proposes the usage of run-length limited (RLL) codes to provide constant dimming levels while transmitting data. In this thesis, the VLC channel is investigated and experimental measurements for frequency modeling of LEDs are conducted first. Two different approaches for performance improvement of VLC links are then proposed. In the first approach, an RLL code that guarantees a maximum run-length of 3 is proposed and experimentally tested. By guaranteeing the run-length limit, the code prevents the vanishing of bits due to the high-pass effect of equalizers. Consequently, the bit error rate (BER) improvement is verified. In the second approach, a high-performance, robust deep learning-based end-to-end VLC system is proposed. The proposed model provides dimming support by modulating the data in zero-mean codewords that can be superimposed with arbitrary DC levels for dimming control. The model is then modified to compensate for the LEDs' nonlinearity. Numerical results show that the proposed model can operate in two different channels with the same high performance.

ÖZET

GÖRÜNÜR IŞIK HABERLEŞMESİNDE VERİM İYİLEŞTİRME YAKLAŞIMLARI

Haberleşme alanında yeni olarak ortaya çıkan görünür ışık haberleşmesi (VLC), aydınlatma ekipmanlarını kullanarak, ışığın şiddetini değiştirerek veri aktarmaya hedefleyen teknolojidir. Son zamanlarda kablosuz haberleşmede kullanılan radyo frekanslarının spektrumu tükendiği için, VLC araştırmacıların dikkatlerini çekmeye başlayıp, pratik sorunları yoğun bir şekilde incelenmiştir. VLC başka haberleşme tiplerinden iki ana alanda farklılık göstermektedir: veri aktarırken gerekli aydınlatma koşullarının sağlanması gerekmesi, ve göndericilerin (genelde Işık Yayan Diyotlar (LED)'ler) sınırlı bant genişliği ve simgelerarası karışımdan oluşan zor kanalıdır. IEEE 802.15.7 grubu VLC için gerekli standartları belirleyip, sabit aydınlatma seviyesini sağlamak için tekrarlanan bitlerin uzunluğu sınırlı olan (RLL) kodları kullanmayı önermiştir. Bu tez çalışmasında, ilk olarak VLC kanalı araştırılmış olup, deneysel olarak LED'lerin frekans tepkisi modellenmiştir. Daha sonra, VLC sistemlerin performansını artıran iki farklı yaklaşım önerilmiştir. İlk yaklaşımda, RLL kodu tasarlanmış olup deneysel olarak denetlenmiştir. Tasarlanan RLL kodu, bit serilerinde arka arkaya gelen aynı bitlerin sayısını 3 ile sınırlamaktadır. Dolayısıyla, yüksek geçiren karakteristiğe sahip kanal denkleştiricilerine karşı sinyalleri iyileştirip bit hata oranını (BER) düşürmektedir. İkinci yaklaşımda, derin öğrenmeyi kullanarak VLC sistemi önerilmiştir. Önerilen sistem, verileri ortalaması sıfır olan sözükler haline modüle edip farklı karartma seviyelerinde çalışabilmektedir. Önerilen model düzeltilip, LED'lerin doğrusalsızlıklarına karşı koyabilecek bir şekilde tasarlanmıştır. Sayısal sonuçlara göre, önerilen sistem, iki farklı kanalda aynı yüksek performansa sahiptir.

TABLE OF CONTENTS

ACKNOWLEDGEMENTS i
ABSTRACT ii
ÖZET iii
LIST OF FIGURES vi
LIST OF TABLES viii
LIST OF SYMBOLS ix
LIST OF ACRONYMS/ABBREVIATIONS xi
1. INTRODUCTION 1
1.1. The Motivation for Visible Light Communication 1
1.2. General Architecture of a VLC System 2
1.3. Challenges in VLC 4
1.4. Contribution of The Thesis 7
1.5. Organization of The Thesis 8
2. VLC CHANNEL 9
2.1. LED's Response 9
2.1.1. Single Pole Model 10
2.1.2. One-Zero One-Pole Model 11
2.1.3. Gaussian Filter Model 13
2.1.4. LEDs' Frequency Response Measurements 14
2.2. Physical Channel Response 17
3. RLL CODING FOR VLC 19
3.1. Literature Review 19
3.2. Proposed Coding Scheme 22
3.2.1. Encoding & Decoding Algorithms 23
3.2.2. Look-Up Table Design 27
3.2.3. Error Detection 27
3.2.3.1. Type I Error 27
3.2.3.2. Type II Error 28

3.3. Experimental Results 28
4. END TO END DEEP LEARNING-BASED DIMMING SUPPORTED VLC SYSTEM 31
4.1. Autoencoder 31
4.2. Literature Review 34
4.3. Proposed Model 35
4.4. Dimming Support With Nonlinearity 37
4.5. Numerical Results 42
5. CONCLUSION AND FUTURE WORK 47
5.1. Future Work 48
REFERENCES 49

LIST OF FIGURES

Figure 1.1. Components of a VLC system. 4

Figure 1.2. Average light intensity per MFTP for different data rates: (a) 64 kbps, (b) 1 Mbps, (c) 40 Mbps, (d) 100 Mbps. 6

Figure 2.1. The linear relation between driving current and optical power of LEDs. 10

Figure 2.2. Equivalent circuit of the LED model of Equation (2.2). 11

Figure 2.3. Equivalent circuit of the LED model of Equation (2.4). 11

Figure 2.4. Different frequency response models for a LED. 14

Figure 2.5. Experimental Setup. 15

Figure 2.6. Measured and fitted frequency responses for different commercial LEDs: (a) LUMINUS SST-90-B-F11-KG300, (b) LUMINUS SST-90-W, (c) BROADCOM ASMT-MT00, (d) CREE XLamp 7090XRE-16F. 16

Figure 2.7. Channel impulse response for (a) D1, (b) D2 in Scenario 3 of [34]. 18

Figure 3.1. Look-up tables, mapping, and header generation. 26

Figure 3.2. Example of received bit powers (a) without the proposed code (b) with the proposed code. 29

Figure 3.3. BER performance of the proposed code compared to other methods. 30

Figure 4.1. A simple autoencoder. 32

Figure 4.2. Dimming levels interpreted as autoencoder constraints. 38

Figure 4.3. A Simple iterative algorithm to generate dimming controlled signal. 39

Figure 4.4. BER performance of the zero-mean autoencoder for LOS channel. 42

Figure 4.5. BER performance of the zero-mean autoencoder for the channel of Figure 2.7(a). 43

Figure 4.6. BER performance of the zero-mean autoencoder tested in two channels. 44

Figure 4.7. BER performance of the modified zero-mean autoencoder with clipping tested in two channels. 45

Figure 4.8. FER comparison of the proposed model and other methods. 46

LIST OF TABLES

Table 1.1. Comparison between RF-based WC and VLC. 2

Table 1.2. A list of commercially available LEDs and their bandwidths. . . . 7

Table 3.1. PHY I operating modes according to IEEE 802.15.7. 21

Table 3.2. PHY II operating modes according to IEEE 802.15.7. 22

Table 4.1. The proposed autoencoder architecture. 36

Table 4.2. Iterations required to obtain different clipping error values. 40

Table 4.3. The modified autoencoder architecture for nonlinearity avoidance. 41

LIST OF SYMBOLS

C_d	Diffusion capacitance of LED
C_j	Junction capacitance of LED
D	Decoder function of an autoencoder
E	Encoder function of an autoencoder
e_m	one-hot representation of a binary vector
$\hat{e_m}$	Recovered one-hot representation at the output of an autoencoder
f_0	Cut-off frequency of a low-pass system
f_p	Pole frequency of a system
f_z	Zero frequency of a system
$H_{LED}(f)$	Transfer function of a LED
$H_{Eq}(f)$	Transfer function of an equalizer
I_s	The saturation current of a LED
P_i	The power of a light ray reaching a receiver
R_i	A voltage source resistance
r_d	The dynamic resistance of a LED
r_s	The series resistance inside a LED
u	A binary data vector
$\hat{u}$	A binary data vector recovered at the output of an autoencoder
V_D	Applied bias voltage to a LED
V_T	Thermal voltage of a LED
x	A representation vector generated by an autoencoder
α	Pre-clipping bias
δ	The impulse function
ℓ	A loss function for training an autoencoder
ℓ_{BCE}	Binary cross-entropy loss

ℓ_{CCE}	Categorical cross-entropy loss
γ_L	The lower bound of a LED's linear region
γ_U	The upper bound of a LED's linear region
τ_i	Delay of a light ray reaching a receiver
Θ_E	An encoder's trainable parameters in an autoencoder
Θ_D	A decoder's trainable parameters in an autoencoder

LIST OF ACRONYMS/ABBREVIATIONS

ASK	Amplitude shift keying
BER	Bit error rate
CAP	Carrierless amplitude-phase modulation
CIR	Channel impulse response
CSK	Color shift keying
CTLE	Continuous time linear equalizer
DL	Deep learning
D2D	Device to Device communication
FEC	Forward error correction
IoT	Internet of Things
IR	Infrared
ISI	Inter-symbol interference
LED	Light Emitting Diode
LOS	Line of sight
LSB	Less significant bit
MFTP	Maximum flickering time period
ML	Machine learning
MSB	Most significant bit
OFDM	Orthogonal frequency division multiplexing
OOK	On-off keying
OWC	Optical wireless Communication
PD	Photodiode
PPM	Pulse position modulation
PSK	Phase-shift keying
RF	Radio frequency
RL	Reinforcement learning
RLL	Run-length limited
SNR	Signal to noise ratio

TIA	Trans-impedance Amplifier
VLC	Visible light communication

1. INTRODUCTION

1.1. The Motivation for Visible Light Communication

The last decade has witnessed an unprecedented increase in mobile devices and the data traffic related to them. According to market analysis [1], global mobile data traffic reached 11.5 exabytes (one exabyte equals one billion gigabytes) per month at the end of 2017 and is expected to be 77 exabytes by 2022. Furthermore, current wireless communication systems start to become insufficient to meet users' needs and fail to adapt to new technologies that require growing access to the network like the internet of things (IoT), device to device (D2D) communication, vehicular communication, and many industrial applications.

Being used in mostly all daily life applications, from radio and cellphones to remote door openers, the main reason for the inabilities of current radio frequency (RF) systems is the overcrowded limited spectrum [2]. Although releasing a new spectrum would appear as a trivial solution to this problem it has some disadvantages like the high cost, atmospheric attenuation at high frequencies, and the fact that the released spectrum will also be finite. Furthermore, practical solutions to increase the capacity of RF communication can be stated under increasing the spectral efficiency of current systems, which also has an upper limit and may have already reached it. This has led researchers to seek a state-of-the-art solution to meet this huge demand and consider the possible growth soon. Mainly, the focus is on providing many access technologies for end-users to use with the current ones. Many solutions were recommended for this problem and optical wireless communication (OWC) was a successful candidate for many reasons. In general, OWC utilizes infrared (IR), visible light, and ultraviolet (UV) rays [3] introducing ultra-high bandwidth of unlicensed spectrum. Furthermore, OWC does not suffer from electromagnetic interference and due to spatial boundaries [4], it offers unlimited frequency reuses with physical security. Other advantages of OWC can be listed as energy efficiency and low cost of development and installation.

With the fact that almost 70% of the wireless voice and data traffic is conveyed in an indoor environment [5], OWC seems to be a perfect medium for such data transfers due to high security.

Visible light communication is a special kind of OWC, in which the already deployed lighting equipment and visible light waves (380-700nm) are utilized to transfer data in free space. With hundreds of terahertz of available spectrum, VLC has attracted attention and taken place between the hot research topics in past years, and in 2009 IEEE has started standardization process for VLC under the group 802.15.7 [6] and numerous system improvements and methods have been proposed in the literature. Most VLC usage areas are indoor communication [3, 7, 8], vehicle-to-vehicle communication, and underwater communication. Indoor positioning [9, 10] is an important application of VLC, where GPS is not available. Table 1.1 summarizes a comparison between current conventional RF communication and visible light communication.

Table 1.1. Comparison between RF-based WC and VLC.

	RF	**VLC**
Spectrum	hundreds of GHz	hundreds of THz
Licensing	Required	Free
Spectrum Reusage	Interference problem	Infinite in indoor applications
Environment	Indoor & Outdoor	Mostly indoor
Range	Large (km)	Short (<km)
Security	Less secure	Physical security
Complexity	More complex	Reduced complexity

1.2. General Architecture of a VLC System

Like in any other communication system, VLC systems can be presented by the three main blocks: transmitter, channel, and receiver. Light Emitting Diodes

used for lighting are usually the transmitters of VLC systems. The VLC channel is composed of line of sight (LOS), on which the transmission heavily relies, and an inter-symbol interference (ISI) part. Line of sight presents the light rays reaching the receiver directly from the source and ISI comes from reflections in the indoor environment. The receiver is a photodiode (PD), or in some applications, cameras are utilized as receivers. The transmitter generates signals by modulating the light intensity of the LED. Figure1.1 summarizes how binary data is conveyed in a VLC system. The signals are transmitted via modulating the light intensity of the LED by changing the driving current. Modulating signal has to be a real and positive signal. Different modulation schemes are adopted in VLC:

- Among digital baseband modulation techniques, on-off-keying (OOK) is the most used modulation type for simple implementation and detection. To drive the LED with the modulating signal a DC bias is required, Bias Tee is usually utilized to impose the varying OOK signal with the DC component. Other modulations like amplitude shift keying (ASK) and phase-shift keying (PSK) can also be utilized [11].
- Pulse Position Modulation (PPM) is also a common modulation type, in which information symbols are encoded in the position of the pulse according to a reference pulse signal. However, when bandwidth efficiency is concerned, PPM is usually avoided.
- Color shift keying (CSK) [12] is a unique modulation scheme for VLC, where bits are encoded by changing the color of the emitted light, this requires LEDs that can generate light in different colors and may not always be available. For example, most white LEDs used for lighting do not have this property.
- For bandwidth efficiency, multicarrier modulations are utilized like orthogonal frequency division multiplexing (OFDM) [13] and carrierless amplitude-phase (CAP) [14] modulation.

At the receiver side, PD senses the light variation and generates a current in line with it, to produce voltage, a trans-impedance amplifier (TIA) is usually placed at the

output of the PD. The voltage signal obtained from TIA is fed to an energy detector for the direct detection of binary symbols in OOK.

In most cases, equalization is required to compensate for the limited bandwidth of the LED and the ISI channel. Equalization can be achieved at the transmitter as a pre-equalizer or the receiver as a post-equalizer and can be implemented digitally or as an analog system.

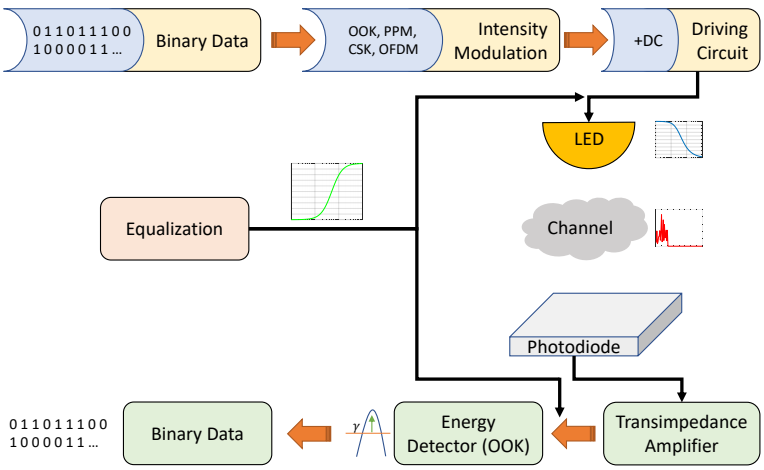


Figure 1.1. Components of a VLC system.

1.3. Challenges in VLC

Aside from its advantages, VLC still faces many challenges making it difficult to provide wide commercial spread and requiring more research and experiments. The main philosophy of VLC is to use the same hardware used for lighting, this means that the system must fulfill different lighting requirements while keeping high performance in terms of data rates and without increasing complexity. Lighting requirements can be stated in two points: flicker prevention and dimming support.

Since data is transmitted by changing the light intensity, these changes must be fast enough to not be caught by human eyes, flicker is the phenomenon in which light intensity changes slowly and the changes are perceived by human eyes. Besides being annoying, it has physiological effects on humans and must be avoided in any VLC system. A condition for flicker is defined as the maximum flickering time period (MFTP), which is the shortest time in which human eyes can perceive light changes and approximately equals 5 ms [15]. If light intensity changes in a period longer than MFTP, flicker happens. In other words, the average light intensity over each MFTP must be the same to avoid flicker. Flicker is not a problem in high data rates, for example, at a data rate of 20 Mbps, a single MFTP contains 10^5 bits, this number is large enough to sample the source distribution and hence the mean is always equal to the source mean. Flicker is usually measured by showing the average light intensity over MFTPs, neglecting nonlinear effects of LED, the intensity changes linearly with the current driving the LED. Figure 1.2 shows the average current over MFTPs for different data rates; as seen in the figure, without implementing any flicker mitigation technique, flicker decreases significantly with increasing data rates.

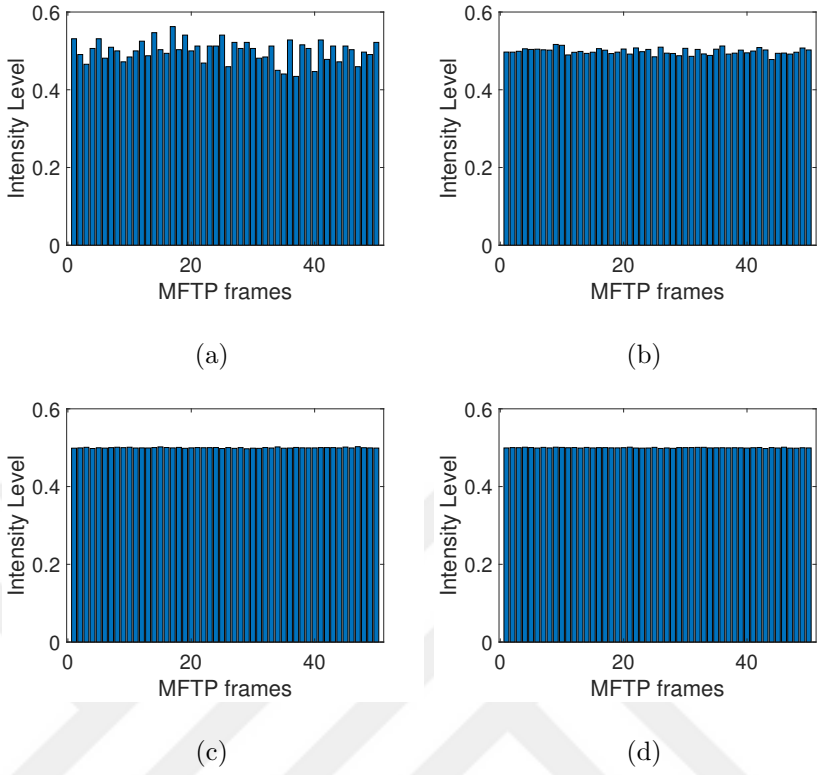


Figure 1.2. Average light intensity per MFTP for different data rates: (a) 64 kbps, (b) 1 Mbps, (c) 40 Mbps, (d) 100 Mbps.

Dimming support refers to the ability of the system to operate in the same performance under different dimming levels chosen by the users. The difference between flicker mitigation and dimming support is that the former focuses on keeping the same mean over MFTPs while the last is concerned with the mean itself. Dimming support can be achieved in analog form by changing the DC level of the OOK signal, or digitally, by controlling the number of zeros and ones in bitstreams via coding. IEEE 802.15.7 standard has defined different coding schemes for different data rates to provide a constant DC of 50% for codewords. The standard also proposes adding compensation symbols at the end of data frames to change the dimming levels [6] [16]. The other challenge related to the communication system itself is the LED and the channel. Light Emitting Diodes used for lighting usually have limited bandwidths, restricting data rates dramatically. Different equalization schemes have been proposed in

the literature [17–20] to extend the bandwidth of the LEDs, but they stay inadequate to meet the high data rate needs. Thus, techniques like OFDM are used to efficiently utilize the available bandwidth. Table 1.2 shows a list of commercially available LEDs used for illumination and their modulation bandwidths.

Table 1.2. A list of commercially available LEDs and their bandwidths.

LED	**Color**	**Bandwidth**
OSRAM LE UW S2W-PXQX-4P7R	Cool White	1.3 MHz
CREE XREWHT-L1-0000-00D01	Cool White	1.2 MHz
Luminus SST-90-W	Cool White	1.3 MHz
OSRAM GF CS8PM1.24	Red	2.3 MHz
CREE CLMVB-FKACFHEHLCBB7A363	Red	20 MHz

Taking the indoor environment into account, the channel in which light is spreading and data transmission is taking place is the ISI channel due to reflections that cause receiving replicas of the transmitted signal in different delays. More details about LED's frequency response and the ISI channel of VLC will be given in Chapter 2.

Another problem to be considered, especially when introducing OFDM as a solution for the limited bandwidth and the ISI channel is that LEDs have nonlinear characteristics [21]. Many methods are proposed in the literature to compensate for these effects, including adaptive Volterra filtering [22].

1.4. Contribution of The Thesis

Experiments reported in this thesis are a part of a TÜBİTAK (the Scientific and Technological Research Council of Turkey) project under the project number 117E058 carried out in the Department of Electrical and Electronics Engineering at Bogazici University. The contribution of this thesis can be grouped into two main topics:

(i) Proposing a novel RLL code for visible light communication that reduces ISI and provides BER improvement with experimental verification.

(ii) Proposing a robust deep learning (DL)-based end-to-end VLC system that provides dimming support with high performance and can be modified to compensate for LEDs' nonlinearity.

1.5. Organization of The Thesis

In Chapter 1, general information about VLC systems architecture, and practical issues are given, components, modulation and bandwidth extension techniques are briefly mentioned. In Chapter 2, the ISI channel of VLC is described in more detail with mathematical models, frequency modeling of LEDs and its impact on equalizing and BER performance are investigated, different LEDs responses are experimentally obtained and given. Run length limited codes and their usage according to IEEE 802.15.7 are examined in Chapter 3, after related literature review and comparisons, a novel RLL code is presented, encoding and decoding algorithms are described and experimental results are reported. Chapter 4 explores applications of DL in communications and related VLC issues that are discussed, a DL-based system for VLC channels is proposed and numerical results are provided. In Chapter 5, conclusions are made and future works are discussed.

2. VLC CHANNEL

Channel modeling is a critical step in developing any communication system, since realistic channel modeling allows reasonable simulations for performance evaluation and testing. In VLC systems, the LED's response and the spatial reflections form the channel response.

2.1. LED's Response

LEDs as transmitters in VLC systems are a bottleneck, limiting the performance significantly. As shown in Chapter 1, LEDs used for lighting typically have bandwidths of couple of MHz. To achieve data transmission of rates around 100 Mbps, LEDs response at high frequencies must be equalized. As the name suggests, LEDs emit light by applying forward bias to a p-n junction. Spatially, a small amount of the released photons travel in the desired direction of lighting, while most of the light spreads in other directions. This results in low efficiency in current-to-light conversion in the LED. The current-voltage relationship of the LED is known as

$$I = I_s(e^{\frac{V_D}{V_T}} - 1). \tag{2.1}$$

On the other hand, a linear characteristic is observed between the driving current and the radiated optical power [23], as shown in Figure 2.1. In frequency domain, there are many physical structures behind different LEDs' behaviours, proposed in the literature. These structures are usually described by equivalent circuits and corresponding transfer functions.

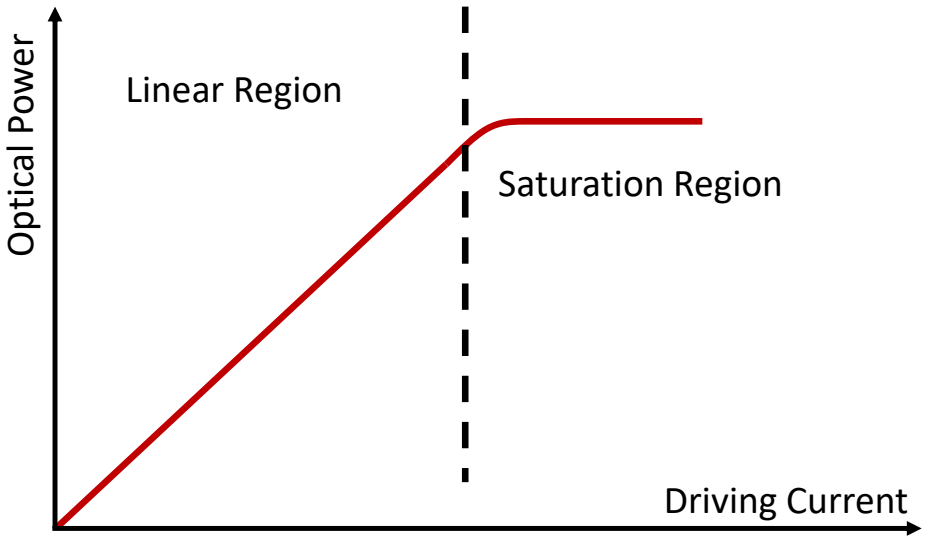


Figure 2.1. The linear relation between driving current and optical power of LEDs.

2.1.1. Single Pole Model

One of the most used models [23, 24] that fits a LED's frequency response is the first-order low-pass filter given as

$$H_{LED}(f) = \frac{1}{1 + j\frac{f}{f_0}}, \tag{2.2}$$

where f_0 in the transfer function is the pole frequency or the -3 dB point of the frequency response of the LED. This results in -20 $dB/decade$ decreasing magnitude of the response as shown in Figure 2.4. This model has a flat response in low frequencies up to the cut-off frequency, where it falls by 3 dB. It can be presented by the simple equivalent RC circuit shown in Figure 2.2, where R and C_j presents the resistance and junction capacitance of the LED, respectively. From the equivalent circuit, the cut-off frequency can be found as

$$f_0 = \frac{1}{2\pi RC_j}. \tag{2.3}$$

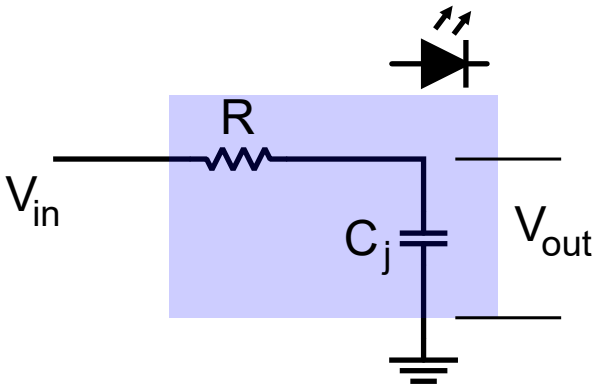


Figure 2.2. Equivalent circuit of the LED model of Equation (2.2).

2.1.2. One-Zero One-Pole Model

A more realistic physical model that considers different phenomena observed and measured in the p-n junction [25] is presented in the equivalent circuit shown in Figure 2.3.

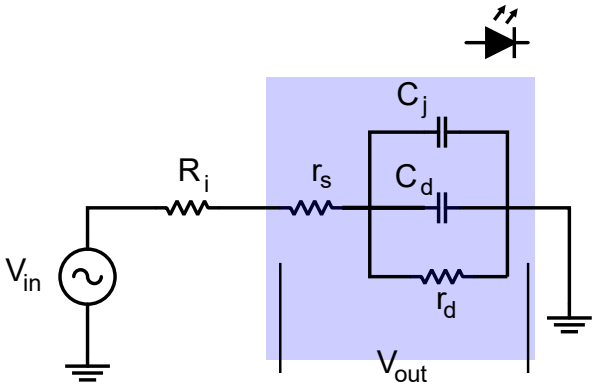


Figure 2.3. Equivalent circuit of the LED model of Equation (2.4).

In Figure 2.3; C_j, C_d, and r_d are called junction capacitance, diffusion capacitance, and dynamic resistance, respectively. The resistance r_s presents a series resistance

inside the LED. When applying a voltage using a source v_{in} having a resistance R_i, the transfer function of this circuit is given as

$$H_{LED}(s) = \frac{V_D(s)}{V_{in}(s)} = \frac{s + \frac{r_d + r_s}{(C_j + C_d) r_d r_s}}{s + \frac{R_i + r_d + r_s}{(C_j + C_d)(R_i + r_s) r_d}}. \tag{2.4}$$

In other words, the model has a one-zero one-pole characteristic that is described by

$$H_{LED}(f) = \frac{1 + j\frac{f}{f_z}}{1 + j\frac{f}{f_p}}. \tag{2.5}$$

Here, the zero and pole frequencies are

$$f_z = 2\pi \frac{r_d + r_s}{(C_j + C_d) r_d r_s}, f_p = \frac{R_i + r_d + r_s}{(C_j + C_d)(R_i + r_s) r_d}, \tag{2.6}$$

respectively. In this model, the response is flat at low frequencies and decreases to $-3\ dB$ at f_p to fall by $-20\ dB/decade$ until reaching f_z, where it remains a flat behaviour again, as shown in Figure 2.4.

It can be noticed that both models give the same curve for a wide range of frequencies if the zero of Equation (2.4) is located at a high frequency. However, the one-zero one-pole model proves very useful in equalization. Equalizing this response can be achieved by implementing a filter with the inverse transfer function (one-zero one-pole again), as described in

$$H_{Eq}(f) = \frac{1 + j\frac{f}{f_p}}{1 + j\frac{f}{f_z}}. \tag{2.7}$$

This model can be implemented using discrete components like in [26] or digitally by digital filter design techniques. The gain of the equalizer is limited by the pole (i.e, the zero of the LED). In fact, the pole may be chosen different than the LED's zero as it limits the gain, since infinite gain is not achievable. This, in turn, results in equalizing

the LED for a certain point. The equalized system's response is expected to be flat up to that point and then start to attenuate resulting in decreasing signal to noise ratio (SNR) with higher data rates at the receiver. As a result, BER increases.

2.1.3. Gaussian Filter Model

Another model proposed in the literature [27, 28] is the Gaussian filter model, which has a transfer function of the form

$$H_{LED}(f) = exp\left[-ln(2)\left(\frac{f}{f_0}\right)^2\right]. \tag{2.8}$$

Compared to the previous models, this model differs slightly before the $-3dB$ point but attenuates impressively after it, limiting the bandwidth and the ability to equalize. Figure 2.4 shows a magnitude plot of the three models for an example LED having cut-off frequency f_0. In the next section, measured frequency responses for different LEDs and the corresponding fittings of the aforementioned models are presented.

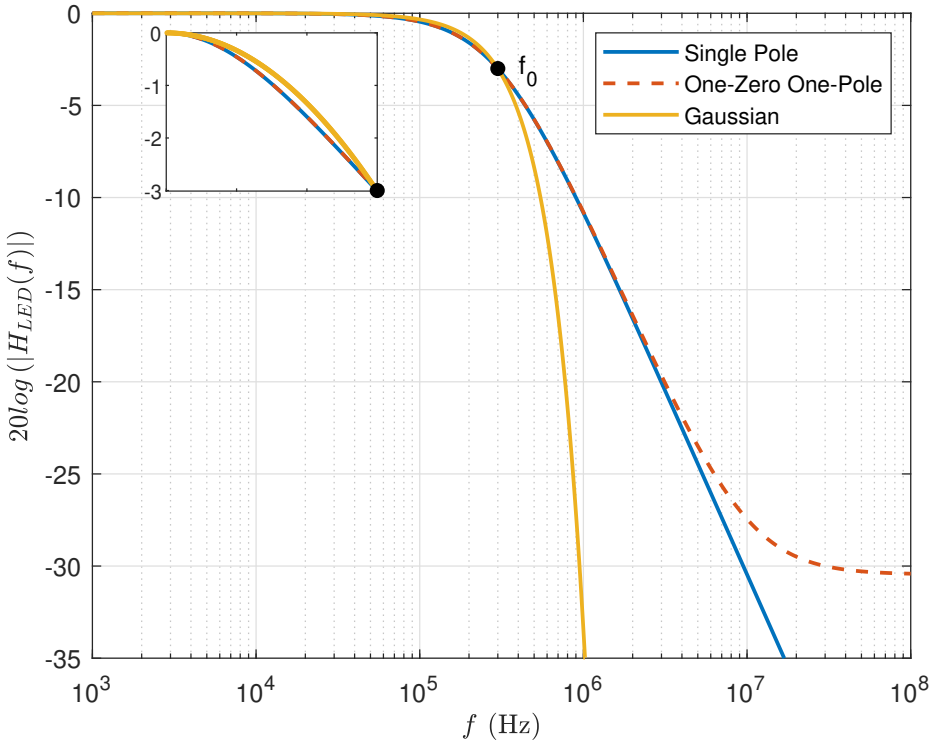


Figure 2.4. Different frequency response models for a LED.

2.1.4. LEDs' Frequency Response Measurements

Using the experimental setup shown in Figure 2.5, the frequency responses of the following commercial LEDs are measured:

- LUMINUS SST-90-B-F11-KG300 Blue LED [29]
- LUMINUS SST-90-W White LED [30]
- BROADCOM ASMT-MT00 Red LED [31]
- CREE XLamp 7090XRE-16F White LED [32]

The LEDs are biased at $2.5V$ during measurements and the same bias voltage is applied in further experiments in order to deal with the same response, since it changes with the junction capacitance that depends on the bias current. Sinusoidal signals at

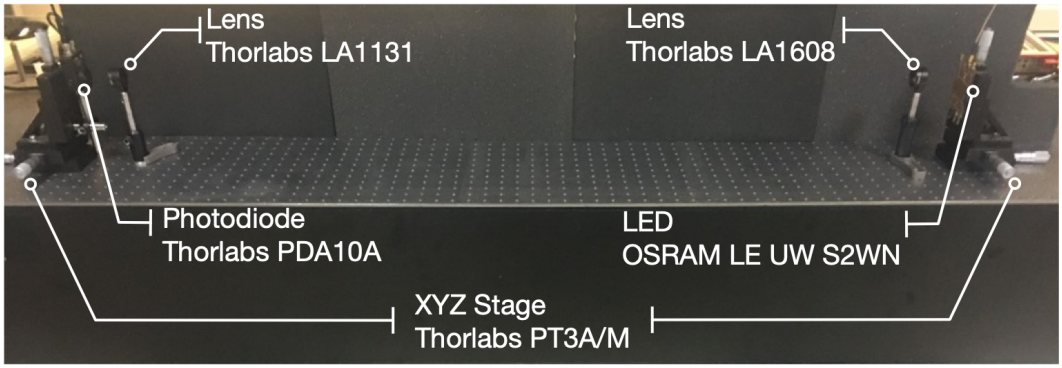


Figure 2.5. Experimental Setup.

different frequencies are generated, superimposed to the DC voltage, and fed to the LED. A photodiode (Thorlabs PDA10A, TIA contained) on the other side is used to observe the optical signal radiated by the LED. The output of the photodiode is then applied to a spectrum analyzer (Rhode&Schwarz FSV), where the electrical power is measured.

For generating DC superimposed sinusoids a signal generator (Textronix AFG-3101) is used. Lenses are used in front of both the transmitter and the receiver to focus the light, consequently, no reflections are observed and the channel is accepted to be LOS. The photodiode has a bandwidth of 150 MHz, thus it has a flat response over the observed frequencies and does not affect the received signals considerably. Efficiency of the LED in converting the current to optical power and the TIA efficiency of the PD attenuates the received signal's power significantly. Thus, to compensate for these effects and obtain a realistic transfer function, the measured signals' power at relatively low frequencies are accepted as a reference response for each LED and subtracted from all other measurements and then the response is fitted and plotted accordingly in Figure 2.6. As seen in the figure, for some LEDs, the models fail to fit the measured responses especially at high frequencies. This is a major factor for limiting the performance of designed equalizers and hence the VLC system.

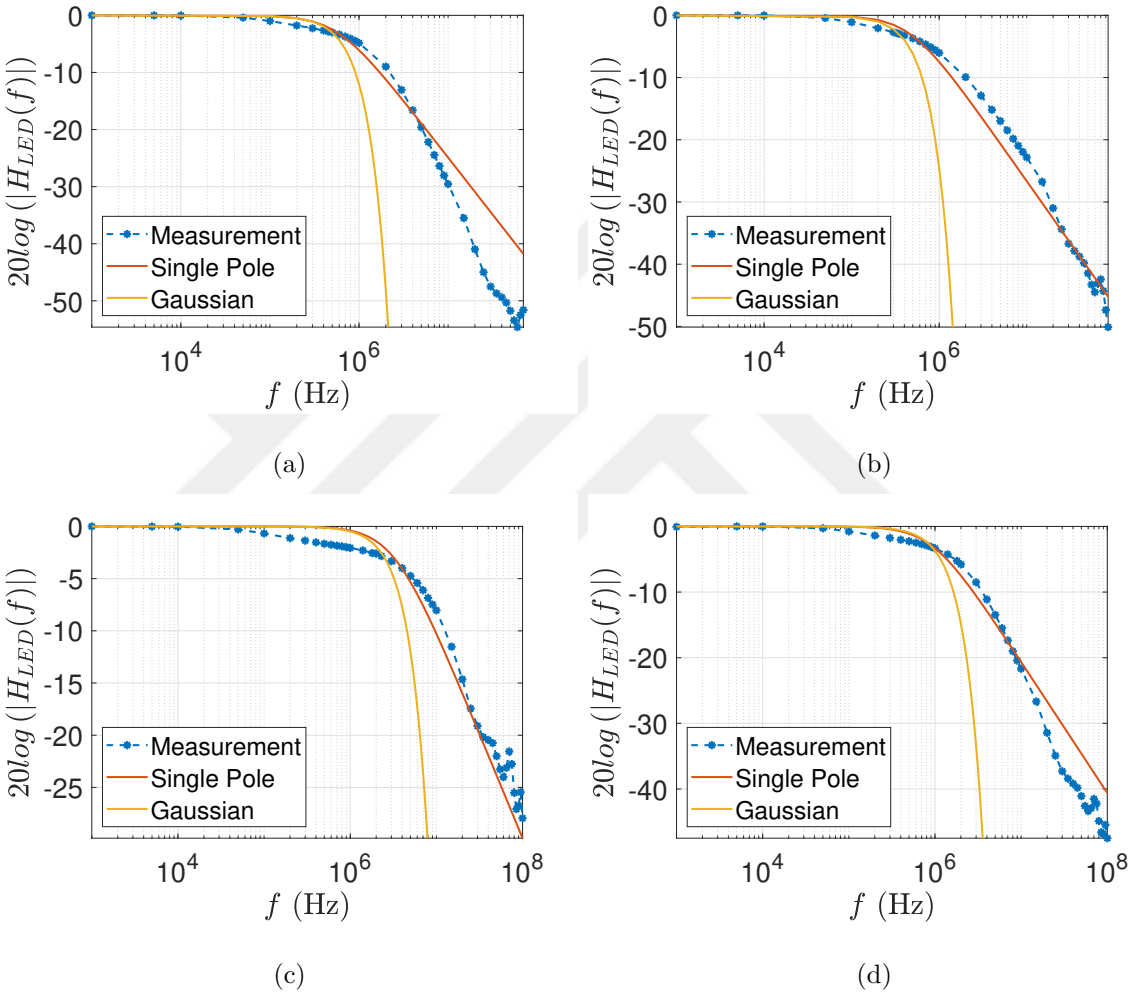


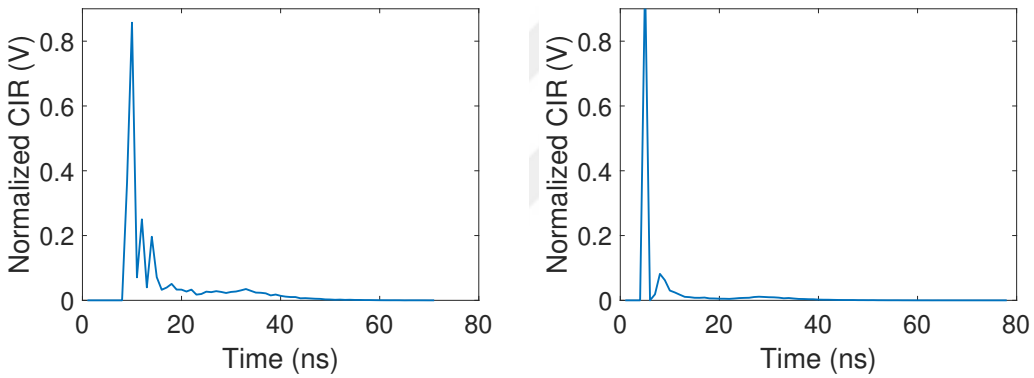
Figure 2.6. Measured and fitted frequency responses for different commercial LEDs: (a) LUMINUS SST-90-B-F11-KG300, (b) LUMINUS SST-90-W, (c) BROADCOM ASMT-MT00, (d) CREE XLamp 7090XRE-16F.

2.2. Physical Channel Response

The main difference between RF and VLC channels is based on the difference in wavelengths, which affects reflections significantly. Reflectance values in VLC vary with the varying wavelengths, this in turn makes it harder to model the channel response. The method adopted by IEEE 802.15.7 group is ray tracing [24, 33]. In ray tracing, as the name suggests, each light ray emitted from the source is tracked over time until it reaches a specific destination. This requires an accurate modeling of the 3D environment, the objects around the source, and the surface materials. A single ray (i) reaching the destination after τ_i delay with power P_i can be represented as an impulse. The channel impulse response can then be written as the summation of these impulses as

$$h(t) = \sum_{i=1}^{N_r} P_i \delta(t - \tau_i). \tag{2.9}$$

Different scenarios were defined by the IEEE 802.15.7 group and the corresponding 3D models were built. Ray tracing is realized using the Zeemax ® software and many light sources and many receivers operating simultaneously were considered. Scenario 3 from [34] is considered in this thesis. The scenario simulates a living room with home objects. The room contains 9 ceiling lightnings as transmitters distributed uniformly in a $6 \times 6m^2$ area, and 8 receivers. Receivers $D1$ and $D2$ are observed and their corresponding channel impulse responses, given in Figure 2.7, are used in simulations of Chapter 4. It is important to notice that the physical channel response in time domain is very short compared to the LED's impulse response. This means that the channel effect as an ISI channel appears at relatively very high data rates.



(a) (b)

Figure 2.7. Channel impulse response for (a) D1, (b) D2 in Scenario 3 of [34].

3. RLL CODING FOR VLC

Run length limited codes are desired in different communication schemes as well as VLC. By preventing long runs of ones and zeros flicker mitigation, better clock synchronization, and as we prove experimentally, BER improvement can be achieved in VLC. The need for RLL codes in BER improvement comes mainly from the equalizers. In VLC, using zero-forcing equalizers at the receiver to compensate for the LED's limited bandwidth results in a high-pass effect. On the other hand, long runs of zeros or ones represent a low-pass signal. As a result, these long runs get distorted by the equalizer raising the BER of the system. For this reason, using an RLL code can improve the performance significantly.

3.1. Literature Review

Like any engineering solution, there are trade-offs in the different RLL coding schemes between bandwidth efficiency, performance, and implementation complexity. In the standard; Manchester, 4B6B, and 8B10B codes are proposed for different modulation schemes and data rates as seen in Tables 3.1 and 3.2 to provide constant DC over codewords. The three coding schemes provide an equal number of zeros and ones in a single codeword. Manchester coding is the most guaranteed method for DC balance and does not suffer from the worst sequence problem. However, bandwidth efficiency is very low due to the coding rate (1/2) and especially when Forward Error Correction (FEC) is used like in the first three lines of Table 3.1. The 4B6B and 8B10B codes provide better bandwidth efficiency but the worst sequence case sequence rises. When a codeword ends with a short run of zeros or ones and the next codeword starts with the same bits a relatively long run is present. While this does not affect the DC balance and hence the flicker mitigation or dimming support, it indirectly causes bit errors at the equalizer's output. Backing to the bandwidth efficiency of proposed codes, the best performance is achieved when 8B10B is used without FEC, this means a flicker-free

communication but no BER improvement is provided.

In [35], a decoding algorithm for RLL codes based on soft-input soft-output decoder is proposed, RLL codes presented in the standard are considered and better decoding performance is achieved with the decoding algorithm. The algorithm was evaluated with different RLL and FEC coding schemes as well as other decoding methods. Compared to previous methods, the highest gain in SNR was reported as 3.67 dB at a BER of 10^{-5}. In [36], a study on the dimming control and flicker mitigation of three RLL codes is conducted. The evaluated codes are Manchester, FM0/1, and Miller Code. Furthermore, an algorithm for decoding FEC codes concatenated with RLL codes in VLC systems is proposed and performance improvement is reported. In [37], an RLL code that maps 3 bits input to 6 bits output was designed. The code designed with trellis representation provides error correction as well as DC balance. Compared to the FEC methods in the standard, a gain of 2 to 6 dB in SNR is achieved at a BER of 10^{-5}. In [38], a generalized algorithm for generating high rate RLL codes with low encoding and decoding complexity is proposed. For performance analysis, different criteria were evaluated like BER and flicker mitigation. The BER performance of a 4B5B code was examined. In [39], a class of rate (n−1)/n RLL coding scheme is presented for OOK-modulated VLC systems and the performance of 2/3 RLL code is investigated. A low complexity split phase code is proposed in [40]. The proposed code is concatenated with a convolutional FEC code and evaluated numerically. Bit error rate improvement at different dimming levels is provided. A finite-state machine-based approach for designing RLL codes to mitigate flicker and reduce BER is presented in [41], the trade-off between states number and flicker mitigation is investigated. Numerical results have shown a 4 dB gain at a codeword error rate of 10^{-5}.

In this thesis, a look-up table-based RLL coding scheme is proposed. The proposed code guarantees a maximum run length of 3 in a bitstream considering worst case sequences. The code aims to provide BER improvement at high data rates, where the long runs of zeros or ones may be attenuated by the equalizers while retaining a high bandwidth efficiency. We furthermore prove this concept experimentally by exhibiting

Table 3.1. PHY I operating modes according to IEEE 802.15.7.

Modulation	**RLL**	**Optical Clock Rate**	**FEC**		**Data Rate**
			Outer Code (RS)	Inner Code (CC)	
OOK	Manchester	200 kHz	(15,7)	1/4	11.67 kbps
			(15,11)	1/3	24.44 kbps
			(15,11)	2/3	48.89 kbps
			(15,11)	None	73.3 kbps
			None	None	100kbps
VPPM	4B6B	400 kHz	(15,2)	None	35.56 kbps
			(15,4)	None	71.11 kbps
			(15,7)	None	124.4 kbps
			None	None	266.6 kbps

Table 3.2. PHY II operating modes according to IEEE 802.15.7.

Modulation	**RLL**	**Optical Clock Rate**	**FEC**	**Data Rate**
VPPM	4B6B	3.75 MHz	RS(64,32)	1.25 Mbps
			RS(160,128)	2 Mbps
		7.5 MHz	RS(64,32)	2.5 Mbps
			RS(160,128)	4 Mbps
			None	5 Mbps
OOK	8B10B	15 MHz	RS(64,32)	6 Mbps
			RS(160,128)	9.6 Mbps
		30 MHz	RS(64,32)	12 Mbps
			RS(160,128)	19.2 Mbps
		60 MHz	RS(64,32)	24 Mbps
			RS(160,128)	38.4 Mbps
		120 MHz	RS(64,32)	48 Mbps
			RS(160,128)	76.8 Mbps
			None	96 Mbps

BER improvement in comparison with non-coded transmission and 8B10B coding.

3.2. Proposed Coding Scheme

A look-up table-based RLL code with a maximum run length of 3 is proposed in this section. The idea is to define some RLL criteria and select codewords satisfying the criteria among all possible bit sequences of a specific length. Then the other (invalid) bit sequences will be mapped to these (valid) codewords. First, it is important to provide the run length limit inside and between codewords to solve the worst case sequence problem. If only codewords with a specific maximum run length are used without considering the effect of adjacent codewords the run length may be doubled. Furthermore, the run length of 3 in the proposed model is guaranteed in the worst

case sequence. The encoder takes a 16-bit sequence as an input and produces a 19-bit sequence as an output. The 19-bit sequence is composed of a 16-bit mapped codeword according to the look-up table and a 3-bit header that defines the mapping and ensures the inter-codewords run length.

3.2.1. Encoding & Decoding Algorithms

The 16-bit codewords are assumed to have a maximum run length of three inside them and provide the same run length limit when combined with their headers and the next codewords' header. The first condition can be applied directly to the inner portion of 16-bit codewords, i.e. to accept only codewords that have a maximum run length of 3 inside them. However, to satisfy the second condition, the edges of the 16-bit codewords are allowed to have a run length of only 2 and the previous or next headers are assumed to extend this run-length limit to 3 or keep it. Consequently, all 16-bit sequences that have a maximum run length of 3 in the inner portion (from 3rd bit to 14th bit) and a maximum run length of 2 in the edges (bits 1,2 and 15,16) are accepted to be valid transmission codewords, other sequences that do not satisfy these conditions will be mapped to the transmission codewords.

Out of 2^{16}, $14,946$ 16-bit sequences satisfy the conditions leaving $50,590$ invalid sequences to be mapped. Accordingly, each one of the transmission codewords will be related to three or four of the $50,590$ sequences, in other words, the $50,590$ sequences will be split into four groups and each group will be mapped to the valid transmission codewords. At this stage, two main cases need to be distinguished by the header:

(i) The original 16 bits form a valid transmission codeword and will be transmitted directly,
(ii) The original 16 bits are an invalid sequence from one of the four groups and have been mapped.

This results in five different states to be presented by the header, thus the header

needs to contain 3 bits at least. It can be thought as one bit will describe whether mapping happened or not (the two previous cases) and two bits will explain how mapping happened if it did. Consequently, for the i'th codeword in a bitstream, the header for the first case (no mapping) is defined as:

$$[MSB_i \quad MSB_i' \quad MSB_i]$$

and for the second case:

$$[X \quad Y \quad MSB_i'],$$

where (MSB) is the most significant bit of the 16-bit transmission codeword, ($'$) is the logical complement, and (X, Y) are the two bits to identify the mapping as will be explained further.

Since any transmission codeword can have at most two repeated bits at its beginning, the proposed header for the first case results in repeating these bits once and no more, thus it guarantees the run length limit of three. Furthermore, the same situation is retained for a previous codeword, if it ends with $[MSB_i \quad MSB_i]$ then the header similarly guarantees the run length limit. In summary, if the first case is applicable for codeword $[i]$ the worst sequence between $[i]$ and $[i-1]$ is

$$[\ \underbrace{MSB_i \quad MSB_i}_{\text{right edge of } (i-1)\text{'th codeword}} \quad \underbrace{MSB_i \quad MSB_i' \quad MSB_i}_{i\text{'th header}} \quad \underbrace{MSB_i \quad MSB_i}_{\text{left edge of } i\text{'th codeword}}\ ]$$

and guarantees the run length limit of three.

On the other hand, when mapping happens, the proposed header implies that the run length limit at the beginning of the transmission codeword will not increase since it adds the complement of the first bit before it. This in turn gives the chance to extend the group of valid transmission codewords and accept the ones with three repeated bits at their beginnings. Searching for this extended criteria gives $17,808$ valid

codewords. As a result, the $50,590$ invalid data sequence needs to be mapped to $17,808$ codewords. Some of the extended valid codewords are also included in the $50,590$ and will be transmitted as they are, but this process has to be considered as 'mapping' for header generation. Under these conditions, each codeword from the extended set will be related to two or three sequences from the invalid ones. In other words, the $50,590$ sequences will be grouped in three sets and each one will be mapped to the extended valid codewords. Accordingly, the 3-bit header for the i'th codeword will be generated as:

(i) $[0 \quad 1 \quad MSB_i']$ for the first set mapping,
(ii) $[1 \quad 0 \quad MSB_i']$ for the second set mapping,
(iii) $[LSB_{i-1}' \quad LSB_{i-1}' \quad MSB_i']$ for the third set mapping.

Adding the complement of the first bit of the codeword before means stopping any run of zeros or ones. Since the previous codeword can have at most two repeated bits at its end, adding $[0 \quad 1]$ or $[1 \quad 0]$ can increase the run length limit to three only while the header in the third case does not increase it. The worst case sequences can be listed as

$$[\underbrace{0 \quad 0}_{\text{right edge of } (i-1)\text{'th codeword}} \quad \underbrace{0 \quad 1 \quad MSB_i'}_{i\text{'th header}} \quad \underbrace{MSB_i \quad MSB_i}_{\text{left edge of } i\text{'th codeword}}]$$

$$[\underbrace{1 \quad 1}_{\text{right edge of } (i-1)\text{'th codeword}} \quad \underbrace{1 \quad 0 \quad MSB_i'}_{i\text{'th header}} \quad \underbrace{MSB_i \quad MSB_i}_{\text{left edge of } i\text{'th codeword}}]$$

$$[\underbrace{MSB_i \quad MSB_i}_{\text{right edge of } (i-1)\text{'th codeword}} \quad \underbrace{MSB_i' \quad MSB_i' \quad MSB_i'}_{i\text{'th header}} \quad \underbrace{MSB_i \quad MSB_i}_{\text{left edge of } i\text{'th codeword}}]$$

and the run length limit is preserved. As a result, three look-up tables are generated as shown in Figure 3.1 and the header is generated accordingly for each codeword to be transmitted.

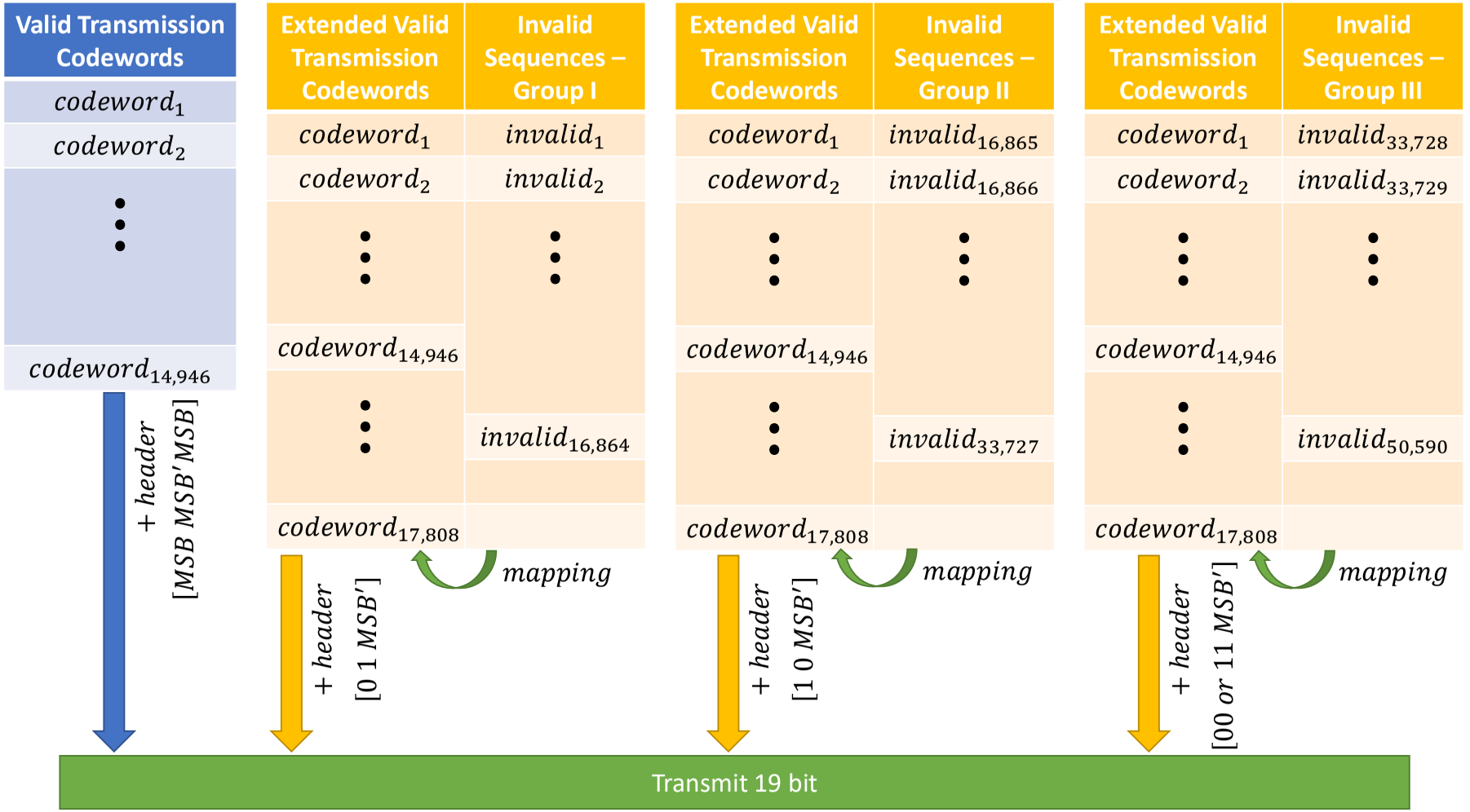


Figure 3.1. Look-up tables, mapping, and header generation.

At the receiver side, the decoder controls the third and fourth bits of the received 19-bit sequence (the third bit of the header and the first bit of the codeword). If the bits are identical then it simply ignores the header and accepts the 16-bit codeword as the decoded sequence. If the bits are different, the look-up table to be checked is defined by observing the first two bits of the header and then a remapping process takes place.

3.2.2. Look-Up Table Design

The look-up table has to be designed such that bit errors happening during the transmission do not result in more bit errors after remapping. For example, a one-bit error inside the 16-bit codeword should result in a one-bit error after decoding. This implies that neighbor 16-bit sequences in Hamming distance should be mapped to neighbor valid codewords too. Errors in the headers should also be considered as well, an error in the first two bits of the header results in remapping the codeword from a wrong look-up table, thus, 16-bit sequences assigned to the same codeword should be close to each other in Hamming distance.

3.2.3. Error Detection

In this section, error detection techniques included in the decoding algorithm are explained. Some errors in the 16-bit codeword or the header can be detected and are grouped into two types:

3.2.3.1. Type I Error. In this type of error, the criteria of valid transmission codewords and the extended valid transmission codewords are examined, detection of this type of error can be achieved by:

- Continuously checking for the run length limit in the bitstream,
- Checking if there are more than 3 consecutive repeated bits inside the 16-bit codeword,

- Given that no mapping has taken place, checking the edge run-length condition for both edges of the codeword,
- Given that mapping has happened, checking the edge run-length condition for the right edge of the codeword.

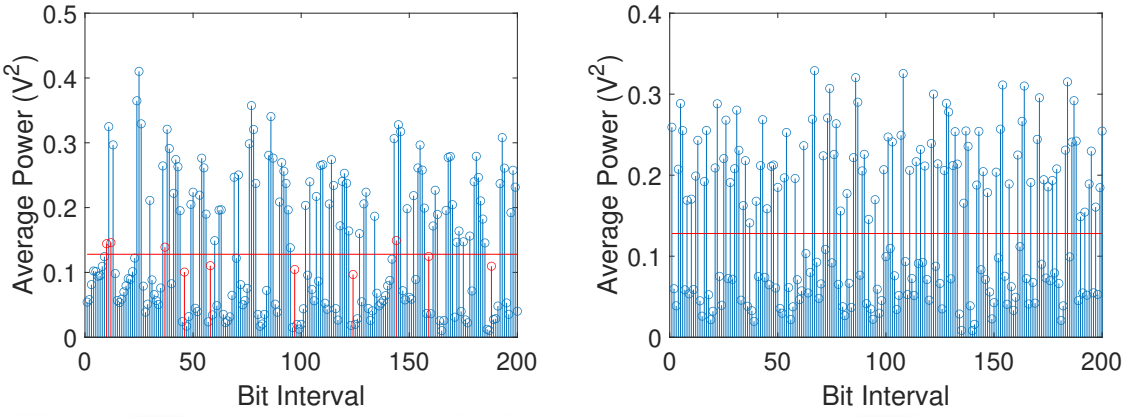
3.2.3.2. Type II Error. This type of error is detected during the remapping operation when the received codeword is not related to a 16-bit sequence, the error may have been in the header or the codeword without disturbing the run-length conditions. These errors are detectable since some codewords are related to two 16-bit sequences instead of three.

3.3. Experimental Results

The proposed code is tested experimentally in the setup shown in 2.5 with an optoelectronic receiver [26], that compensates for the LED's attenuation up to approximately 40 MHz. It includes a TIA, a continuous-time linear equalizer (CTLE), and an amplifier. A random bit pattern of 1008 bits is generated and encoded with the proposed code in Matlab. The coded data is then uploaded to the signal generator to feed it repeatedly to a white LED (OSRAM LE UW S2W). The signal generator drives the LED with an OOK signal of $1.5V_{PP}$ superimposed on $3V$ DC bias.

The output of the receiver is connected to an oscilloscope (Rhode&Schwarz RTE-1034), where the received waveform is observed. An interval containing 10^5 bits is captured from the oscilloscope for threshold detection. In Matlab, the received power over bit slots is calculated and bit detection is realized. For comparison, the same experiment is conducted with non-coded and 8B10B coded data at different data rates. Increasing the data rate means lowering the SNR as the LED impairs the transmitted signal.

Figure 3.2 shows the received powers over bit intervals for non-coded and coded data with the proposed code. The coding effect on ISI mitigation can be seen clearly



(a) (b)

Figure 3.2. Example of received bit powers (a) without the proposed code (b) with the proposed code.

in the figure. When RLL is not used, ones and zeros levels are distributed almost continuously and mixed, making it hard to define an appropriate energy threshold for symbol detection. As a result, the bits marked in red color shown in Figure 3.2(a) are incorrectly detected. The proposed RLL code provides better clearance between ones and zeros as seen in Figure 3.2(b).

The BER results for the proposed code, 8B10B, and non-coded data are presented in Figure 3.3. The data rates in the figure are the original bit rates. The clock frequency for 8B10B and the proposed code are 10/8 and 19/16 times the data rates, respectively. The BER raises remarkably after a certain frequency, and the gain provided by the proposed code at high rates comes from the difference in bandwidth efficiency as well as the constant run-length limit.

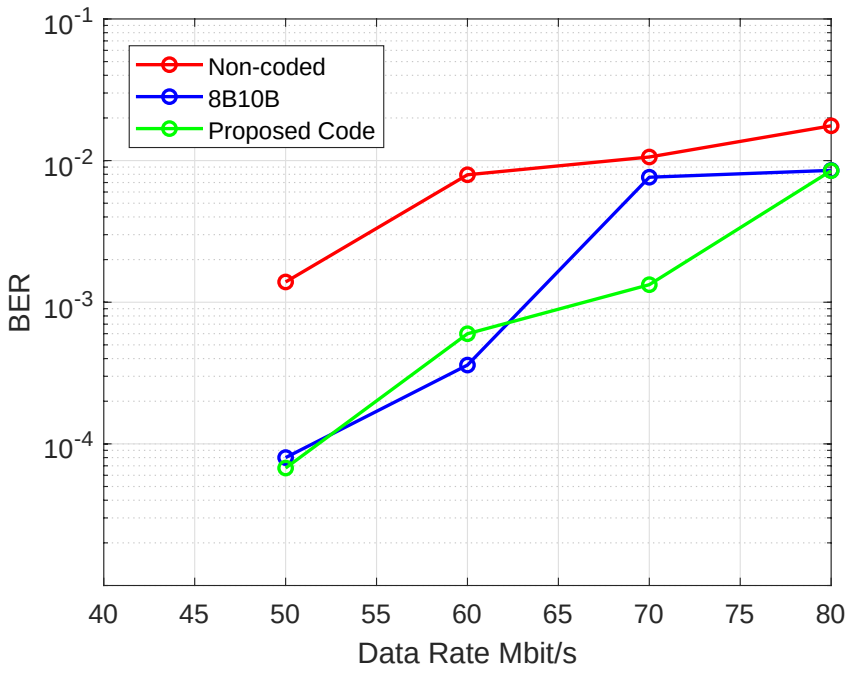


Figure 3.3. BER performance of the proposed code compared to other methods.

4. END TO END DEEP LEARNING-BASED DIMMING SUPPORTED VLC SYSTEM

With the rapid growth and development in machine learning (ML), it started to become a vital part of many software and hardware applications as well as communication systems. Machine learning proved useful in many applications in this field like channel estimation, anomaly detection, modulation classification, and resource allocation. In the recent years, DL and neural networks have also started to engage many daily life applications and provided impressive results. An important application of DL in communication is modeling the end-to-end communication system as a DL structure known as Autoencoders.

4.1. Autoencoder

Normally, an autoencoder is a neural network utilized for representation learning, where the machine learns how to form a representation of data vectors under certain regularization, and how to reconstruct the original vector from that representation [42]. While autoencoders usually learn to find shorter representations (compress the data), longer representation of data is desired in communication applications, which in fact, simulates a channel code.

Mathematically, an autoencoder takes an input vector u and generates a hidden representation

$$x = E(u, \Theta_E), \tag{4.1}$$

where Θ_E are the encoder parameters learned during training. The decoder part takes the representation to recover

$$\hat{u} = D(x, \Theta_D), \tag{4.2}$$

where Θ_D are the decoder parameters. The aim is to obtain $\hat{u}$ close to u with least errors. Consequently, a loss function

$$\ell\left(u, D\left(E(u, \Theta_E), \Theta_D\right)\right)$$

needs to be defined such that minimizing it improves the performance of the system. As in any communication system, maximizing mutual information between the transmitted u and received $\hat{u}$ data, means raising the system's performance. Thus, the loss function should have cross-entropy form. An example autoencoder structure is shown in Figure 4.1.

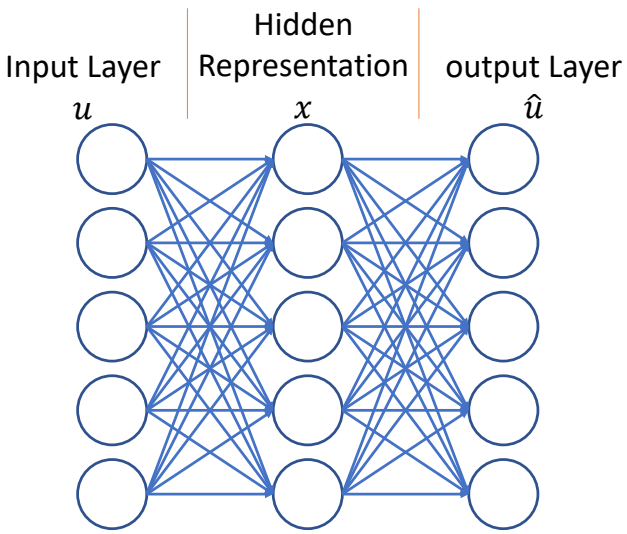


Figure 4.1. A simple autoencoder.

When modeling communication systems as an autoencoder, the input messages are first represented as one-hot vectors. Considering a binary vector v of length m, it is encoded as a vector of length 2^m, where all elements are zeros except for one element having the value one. The binary representation of the index of this one element is equal to the vector v. This one-hot representation is fed to the autoencoder and the

output is expected to be the same. This can be treated as a classification task and the loss function used in this case is the categorical cross-entropy, which can be expressed as

$$\ell_{CCE} = -\sum_{j=1}^{2^m} e_m[j] \log \hat{e_m}[j], \tag{4.3}$$

where e_m is the one-hot encoded message vector, $\hat{e_m}$ is the predicted output and the summation is over their components. The expression in Equation (4.4) presents the categorical cross-entropy for a single message (bit sequence). Since the data is grouped in batches during training, the optimized loss function is a summation of the term presesnted in Equation (4.4) for individual messages. Although optimizing the autoencoder over this loss function is efficient, the memory requirement for the one-hot encoder is large and grows exponentially 2^m, this also raises the number of the autoencoder's trainable parameters resulting in longer training procedure.

Another approach for memory efficiency is to apply the binary vector directly to the autoencoder and observe a binary vector at the output as well. The loss function in this case can be the binary cross-entropy as

$$\ell_{BCE} = -\sum_{j=1}^{m} \left(u[j] \log \hat{u}[j] + (1 - u[j]) \log(1 - \hat{u}[j]) \right), \tag{4.4}$$

where u and $\hat{u}$ are the original binary message and the corresponding prediction, respectively.

Historically, learning-based approaches have been utilized in communication, an example can be given from the 1990 [43], where neural networks were exploited for decoding improvement. Compared to conventional communication systems, the autoencoder has the advantage of optimizing the transmitter (the encoder part) and the receiver (the decoder part) simultaneously, while this is not feasible in conventional designs.

Optimizing or training the autoencoder is realized by the backpropagation, where the derivative of the loss function with respect to the autoencoder parameters is calculated at each training epoch and the parameters are updated accordingly. This information is assumed to be conveyed from the receiver to transmitter via another reliable channel. This assumption is reasonable for many VLC scenarios, where the receiver is a mobile device and the transmitter is the static illumination equipment.

4.2. Literature Review

Many works in the literature considered autoencoders for communication systems in general, for example, [44] provided a theoretical basis for this concept by proving that any communication system can be modeled as a regularized autoencoder, and by optimum training, Shannon capacity can be achieved. The study also proposed a sub-optimum training algorithm and reported numerical results. In [45] a reinforcement learning (RL)-based approach to train autoencoders without channel information is proposed. A well-known disadvantage of RL is the long training process, however, learning without channel information is a significant advantage.

An important issue in autoencoders is the data size. For training period and memory requirements, the training set is preferred to be relatively small. Thus, if all possible data combinations are large and a small part of them is used in training, the model fails to generalize to the rest of the data. Numerically, training an autoencoder that learns to encode 20-bit sequences with 2^{16} different bit sequences means that the autoencoder is seeing less than 7% of the possible sequences. Circular convolution was proposed in [46] as a solution to this problem. Numerical results have shown a BER improvement besides lowering the training data size.

Apart from these works, machine learning application sin VLC, like nonlinearity mitigation, jitter compensation, modulation format identification, and phase estimation were investigated in [47]. However, a few works considered designing special autoencoders for VLC. In [48] the VLC system is modeled as an autoencoder and the

vanishing gradient problem in training is considered. A penalization term for dimming is added to the loss function. However, this does not guarantee the system to operate always at the specific dimming level. Another work [49] has considered using autoencoder for OWC, since VLC is not aimed, dimming is not considered. A deep learning system model that produces dimming control codes for OOK is investigated in [50], dimming levels are fed as a part of the input for the encoder and the decoder of the system. The work also focused on the binarization of codewords to generate OOK signals. In a recent work [51], a novel autoencoder that models the VLC system is proposed. By defining a special activation function, the model provides run-length limit as well as dimming support. Due to the special activation function, the dimming level is guaranteed to be in specified range but without guaranteeing a certain value.

In this work, an autoencoder that provides flicker-free fully dimming supported VLC is designed. By producing zero-mean output, the transmitted signals will always have a constant DC. The model is then modified to consider the LEDs' nonlinearity to provide arbitrary dimming levels.

4.3. Proposed Model

An autoencoder design for dimming supported VLC is proposed in this section. The encoder part of the autoencoder learns to represent a 12-bit data in modulated 20-bit codewords that have zero-mean, the zero-mean codewords can be superimposed on different DC values to provide different dimming levels. The receiver removes the DC part and feeds the representation to the decoder part of the autoencoder, which extracts the original bit sequence.

The zero-mean condition can be provided by adding a penalty to the representation's mean in the cost function. However, this does not ensure the mean to be zero in all cases. Instead, a deterministic layer subtracting the mean of the representation is added at the output of the encoder part. Consequently, the decoder learns to decode zero-mean codewords, and the encoder is optimized for this. The proposed autoencoder

architecture is presented in Table 4.1.

Table 4.1. The proposed autoencoder architecture.

Layer	**Type**	**Output Shape**
E1	Dense + ReLU	256
E2	Dense + ReLU	128
E3	Dense + ReLU	64
E4	Dense	20
E5	Mean Subtraction	20
Channel: Convolution + Noise		
D1	1D Convolution + ReLU (kernel size:5)	32×30
D2	Flatten	960
D3	Dense + ReLU	512
D4	Dense + ReLU	256
D5	Dense + ReLU	128
D6	Dense + Sigmoid	12

The convolutional layer at the receiver can be considered as a 5-tab FIR equalizer that learns 32 different impulse responses to equalize the ISI channel's effect. To ensure this concept the model was first trained without noise and it was able to uniquely encode and decode any possible bit sequence of length 12. Two main techniques were adopted during training:

- Training with a fixed SNR level and evaluating at different SNR levels.
- Training with arbitrary changing SNR levels (between 5 and 12 dB).

For each case, the model is:

(i) trained and tested for LOS channel,

(ii) trained and tested for the ISI channel of Figure 2.7(a),

(iii) trained with arbitrary changing channels of Figure 2.7(a) and (b) and tested with them.

The CIRs presented in Figure 2.7 were sampled at a sampling period of 6 ns, and then applied to the autoencoder as a deterministic convolutional layer in the channel part. Numerical results are proposed and discussed in Section 5 of this chapter. In the following, an important issue is considered and the autoencoder design is modified accordingly.

4.4. Dimming Support With Nonlinearity

As mentioned in Chapter 2, the LED operates in a linear region up to a certain point, after which it saturates. This may affect the DC-controlled dimming level as well as the system's performance. To explain this concept, let us assume a LED with a linear operating range between γ_L and γ_U. At a dimming level of 50% and a normalized signal for this range, the system operates normally. However, at a different dimming level, if only the DC of the signal is changed without its range, the signal will be clipped by the LED and its mean will be different than the desired dimming level. Considering this effect in the autoencoder results in applying a dimming-related constraint as shown in Figure 4.2.

The constraint should be applied while keeping the mean zero. This can be achieved by adding a specific DC bias α to the signal before clipping such that the mean becomes zero after clipping. Mathematically, for a signal x and clipping function C, an α value is desired such that

$$mean(C(x - \alpha)) = 0. \tag{4.5}$$

Solving Equation (4.5) may not be achievable in closed form. However, it can be solved iteratively with a low number of iterations. Basically, the signal is first zero-meaned,

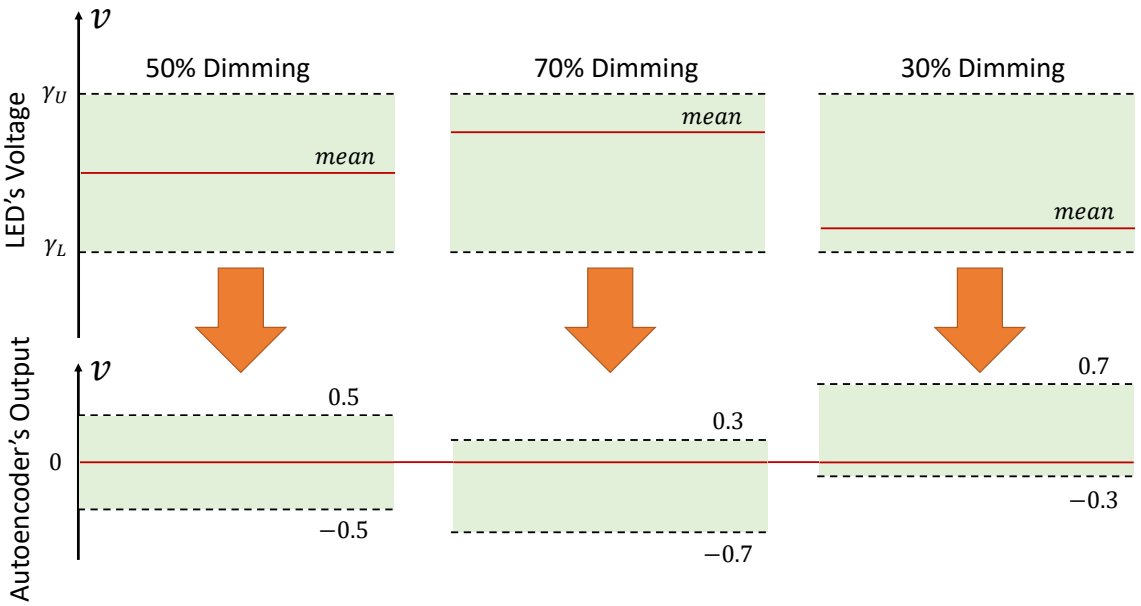


Figure 4.2. Dimming levels interpreted as autoencoder constraints.

and then the error that will occur in the mean after clipping can be calculated. The calculated error is then subtracted from the signal's mean and calculated again. The process repeats until reaching a small ϵ value of error. A pseudocode of such an iterative algorithm is proposed in Figure 4.3.

Input: x signal vector, γ_U upper bound, γ_L lower bound.

Output: y zero-mean bounded vector.

$m = mean(x)$

$y = x - m$

$clip_err = 1$ the mean error resulting at each iteration.

$tot_err = 0$ the total error to be subtracted as the bias.

while $clip_err > \epsilon$ **do**

$z = clip(y, \gamma_U, \gamma_L)$

$clip_err = mean(z)$

$tot_err = tot_err + clip_err$

end while

return: $y = x - m - tot_err$

Figure 4.3. A Simple iterative algorithm to generate dimming controlled signal.

Backing to the autoencoder design, iterative operations are not a suitable structure for neural networks, instead, we applied the algorithm as consecutive deterministic layers, each layer calculates the clipping error and subtracts it from the input. We called these layers pre-clipping bias layers. The number of them can be a hyperparameter of the design controlling the accuracy in the dimming levels.

To obtain a rough approximation of the number of these layers, the iterative algorithm has been simulated for large amount of arbitrary outputs. Histograms of iteration numbers are obtained for different ϵ values and the results are summarized in Table 4.2.

Table 4.2. Iterations required to obtain different clipping error values.

ϵ	**Average Iterations**	**Maximum Iterations**
10^{-5}	6	13
10^{-6}	8	16
10^{-7}	9	19
10^{-8}	11	23

It is important to notice that the complexity of these layers is not of considerable importance for the backpropagation algorithm, as they are only adding bias. The modified autoencoder architecture is presented in Table 4.3.

Table 4.3. The modified autoencoder architecture for nonlinearity avoidance.

Layer	**Type**	**Output Shape**
E1	Dense + ReLU	512
E2	Dense + ReLU	256
E3	Dense + ReLU	128
E4	Dense + ReLU	64
E5	Dense	20
E6	Mean Subtraction	20
E7	**Pre-Clipping Bias layers**	20
E8	**Clip** (γ_L, γ_U)	20
Channel: Convolution + Noise		
D1	1D Convolution + ReLU (kernel size:5)	32×30
D2	Flatten	960
D3	Dense + ReLU	1024
D4	Dense + ReLU	512
D5	Dense + ReLU	256
D6	Dense + ReLU	128
D7	Dense + Sigmoid	12

Compared to Table 4.1, apart from adding the pre-clipping bias and clipping layers; the depth and width of the neural network are increased to provide more complexity so that the model can be optimized under the clipping constraint. In order to provide a differentiable structure for the backpropagation process, the clipping function is defined as

$$C(x) = (tanh(x) + 1) \times \left(\frac{\gamma_U - \gamma_L}{2} \right) + \gamma_L. \tag{4.6}$$

Consequently, an output bounded by γ_L and γ_U is generated via a soft differentiable function.

4.5. Numerical Results

Figure 4.4 shows the BER performance of different training techniques of the zero-mean autoencoder in LOS channel. For this case, the coding rate is choosen to be (20/16) since the channel is not high challenging. However, in all other simulations, coding rate is chosen to be (20/12) for comparison with other work and providing dimming ability. The training data set is of size $2^{16} \times 10$. The model is trained for 1000 epochs with early stopping when the loss changes are relatively small. The batch size equal to 512. The optimization algorithm is Adam with learning rate of 10^{-4}.

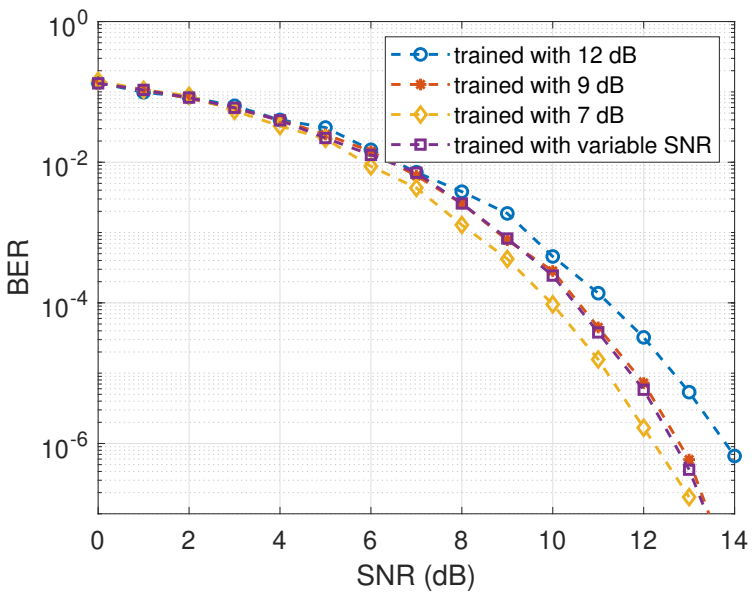


Figure 4.4. BER performance of the zero-mean autoencoder for LOS channel.

As seen in the figure, the four approaches gives close generalized results. Being able to train the model at arbitrary SNR and obtain the same performance is a major advantage of the proposed autoencoder. Figure 4.5 shows the BER performance of different training techniques of the zero-mean autoencoder in the channel of Figure 2.7(a). The training data set is of size 2^{16}. Other parameters are the same as in the previous simulation.

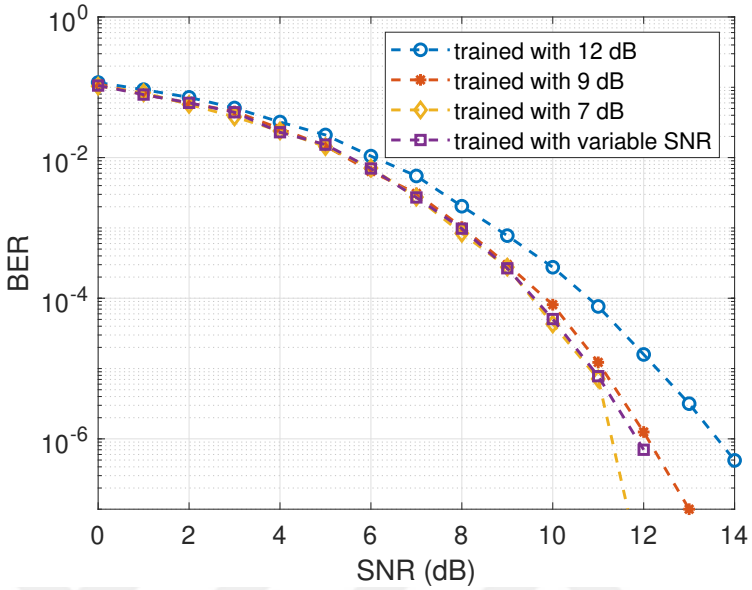


Figure 4.5. BER performance of the zero-mean autoencoder for the channel of Figure 2.7(a).

For robustness, the channel have been randomly changed from the one in Figure 2.7(a) to the one in Figure 2.7(b) during training. The resulted model is then tested in both channels separately and the performance is presented in Figure 4.6.

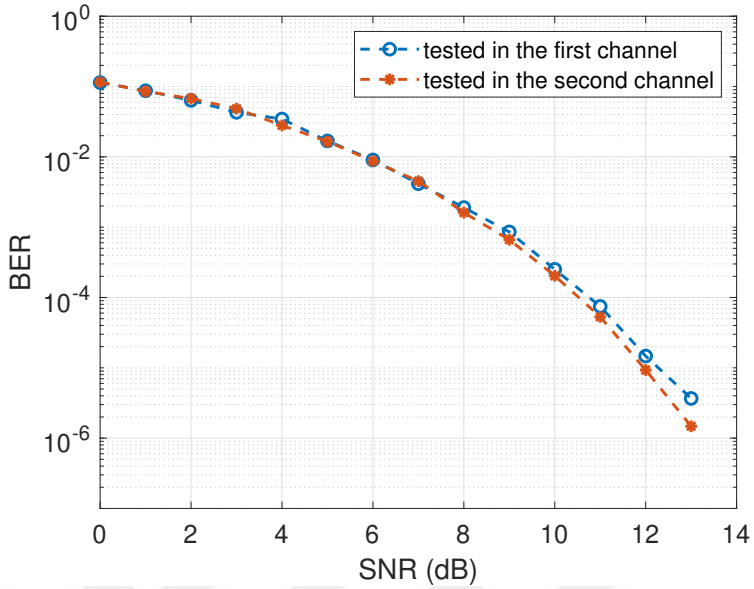


Figure 4.6. BER performance of the zero-mean autoencoder tested in two channels.

The modified zero-mean autoencoder has been trained with randomly changing SNR values and channels for different dimming levels and their corresponding clipping. The models are then tested in both channels and the results are reported in Figure 4.7. The number of pre-clipping bias layers used in the models is 10. A Slight drop in the performance is observed between the channels. However, the model operates in almost the same performance for different dimming levels.

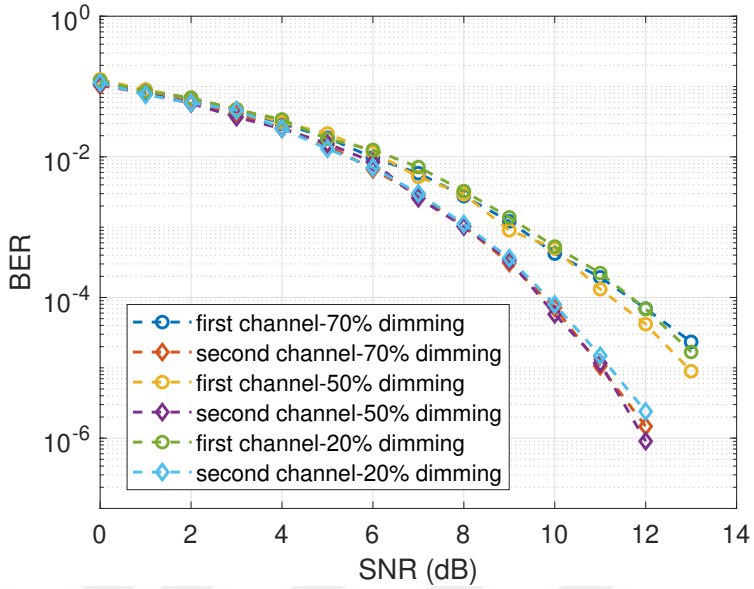


Figure 4.7. BER performance of the modified zero-mean autoencoder with clipping tested in two channels.

Finally, Figure shows a comparison between the proposed modified autoencoder and VLCnet [51] and other conventional coding schemes. In VLCnet, three parameters control the dimming range λ, c, and l. For $\lambda = 0.8, c = 3, l = 5$ the dimming is restricted in the range (48-60)%. Accordingly, the proposed modified autoencoder with clipping is trained on the same channel with 50% dimming level and the results are compared. Comparisons are based on the frame error rate, that is, 12-bit frames.

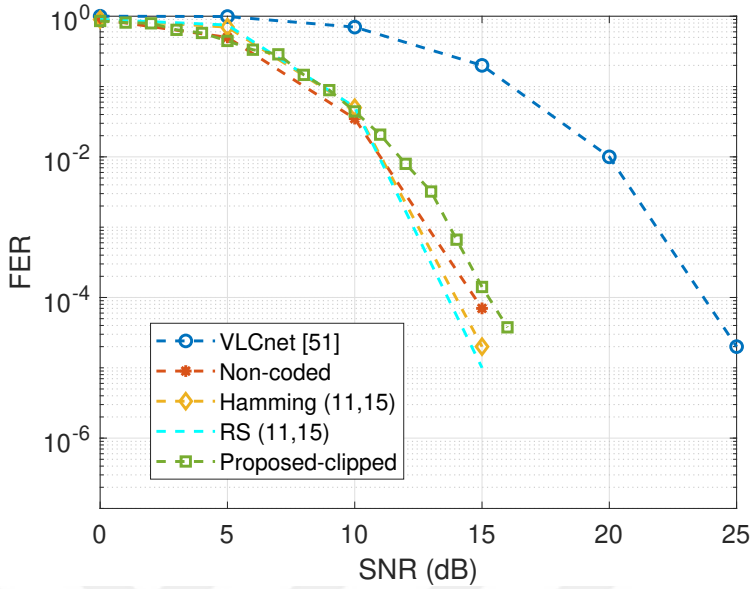


Figure 4.8. FER comparison of the proposed model and other methods.

Besides providing accurate dimming levels, channel and nonlinearity robustness, the proposed model overcomes the other method and exhibit a performance close to FEC codes as seen in the figure.

5. CONCLUSION AND FUTURE WORK

Visible light communication is a promising technology to provide high-rate data transmission. There are many difficulties related to the VLC channel and lighting requirements making it a hot research topic. In this thesis, VLC channel is first investigated. Frequency responses of different commercial LEDs is measured experimentally and the corresponding models in the literature are evaluated.

A new RLL coding scheme that reduces BER is then proposed. The code is look-up table based and has simple encoding and decoding algorithms. The code is experimentally implemented and BER improvement is verified. Compared to 8B10B and non-coded transmission, the proposed code has a better performance allowing data transmission at higher rates.

In the second part of the thesis, an autoencoder design for VLC is proposed. Implementing dimming related constraint as deterministic layer forces the autoencoder to operate with arbitrary dimming levels. Nonlinearity of LEDs' is then considered. To avoid operating in the saturation region of LEDs, the autoencoder is modified to generate clipped output that guarantees the dimming level. This is achieved implementing an iterative algorithm as deterministic layers to calculate the required bias and subtract it from the signal prior to clipping. A differentiable function is implemented as clipping layer so that it does not affect the backpropagation during training. By increasing the number of the pre-clipping bias layers, the accuracy of the dimming support can be increased. Compared to other works, the proposed model provides best accuracy in dimming levels and robustness to channel mismatch.

5.1. Future Work

Implementation of the proposed RLL code can be investigated, different implementation scheme can be designed as a future work. Deep learning is a very fast growing field and different structures can be included to the proposed autoencoder model for further functionality improvement. Another topic that can be researched is the implementation of the bias subtracting layers for probable less complexity. Orthogonal frequency division multiplexing with dimming support is an important problem that can be addressed by autoencoders. Finally, with the approaching DL chips, more efficient ways of implementing the neural networks can be investigated.

REFERENCES

1. *Cisco Visual Networking Index: Global Mobile Data Traffic Forecast Update, 2017–2022*, Cisco White Paper, February 2019.

2. Ghassemlooy, Z., L. Alves, S. Zvanovec and M.-A. Khalighi, *Visible Light Communications, Theory and Applications*, June 2017.

3. Elgala, H., R. Mesleh and H. Haas, "Indoor Optical Wireless Communication: Potential and State-of-the-Art", *IEEE Communications Magazine*, Vol. 49, No. 9, pp. 56–62, 2011.

4. Arnon, S., J. Barry, G. Karagiannidis, R. Schober and M. Uysal, *Advanced Optical Wireless Communication Systems*, Cambridge university press, 2012.

5. Uysal, M., "Visible Light Communications: From Theory to Industrial Standardization", *Optical Fiber Communication Conference*, pp. Th3I–4, Optical Society of America, 2019.

6. "IEEE Standard for Local and Metropolitan Area Networks–Part 15.7: Short-Range Wireless Optical Communication Using Visible Light", *IEEE Std 802.15.7*, pp. 1–309, 2011.

7. Song, J., W. Ding, F. Yang, H. Yang, B. Yu and H. Zhang, "An Indoor Broadband Broadcasting System Based on PLC and VLC", *IEEE Transactions on Broadcasting*, Vol. 61, No. 2, pp. 299–308, 2015.

8. Elgala, H., R. Mesleh and H. Haas, "Indoor Broadcasting via White LEDs and OFDM", *IEEE Transactions on consumer electronics*, Vol. 55, No. 3, pp. 1127–1134, 2009.

9. Lin, B., X. Tang, Z. Ghassemlooy, C. Lin and Y. Li, "Experimental Demonstra-

tion of an Indoor VLC Positioning System Based on OFDMA", *IEEE Photonics Journal*, Vol. 9, No. 2, pp. 1–9, 2017.

10. Zhang, W. and M. Kavehrad, "Comparison of VLC-Based Indoor Positioning Techniques", *Broadband access communication technologies VII*, Vol. 8645, p. 86450M, International Society for Optics and Photonics, 2013.

11. Rodríguez, J., D. G. Lamar, P. F. Miaja and J. Sebastián, "Reproducing Single-Carrier Digital Modulation Schemes for VLC by Controlling the First Switching Harmonic of the DC–DC Power Converter Output Voltage Ripple", *IEEE Transactions on Power Electronics*, Vol. 33, No. 9, pp. 7994–8010, 2018.

12. Monteiro, E. and S. Hranilovic, "Design and Implementation of Color-Shift Keying for Visible Light Communications", *Journal of Lightwave Technology*, Vol. 32, No. 10, pp. 2053–2060, 2014.

13. Wang, Z., T. Mao and Q. Wang, "Optical OFDM for Visible Light Communications", *2017 13th International Wireless Communications and Mobile Computing Conference (IWCMC)*, pp. 1190–1194, 2017.

14. Chvojka, P., K. Werfli, P. A. Haigh, S. Zvanovec, Z. Ghassemlooy and M. R. Bhatnagar, "Multi-Band Carrier-less Amplitude and Phase Modulation for VLC: An Overview", *2017 First South American Colloquium on Visible Light Communications (SACVLC)*, pp. 1–6, 2017.

15. "IEEE Recommended Practices for Modulating Current in High-Brightness LEDs for Mitigating Health Risks to Viewers", *IEEE Std 1789-2015*, pp. 1–80, 2015.

16. Rajagopal, S., R. D. Roberts and S.-K. Lim, "IEEE 802.15.7 Visible Light Communication: Modulation Schemes and Dimming Support", *IEEE Communications Magazine*, Vol. 50, No. 3, pp. 72–82, 2012.

17. Jin, J., J. Wang, D. Chen and H. Lu, "Experimental Implementation of Digital

Equalizer for Multilevel Signal in Visible Light Communication", *Optical Engineering*, Vol. 59, No. 1, p. 016110, 2020.

18. Yeh, C., C. W. Chow, Y. Liu and P. Huang, "Simple Digital FIR Equalizer Design for Improving the Phosphor LED Modulation Bandwidth in Visible Light Communication", *Optical and Quantum Electronics*, Vol. 45, No. 8, pp. 901–905, 2013.

19. Haigh, P., Z. Ghassemlooy, S. Rajbhandari and E. Leitgeb, "A 100 Mb/s Visible Light Communications System Using a Linear Adaptive Equalizer", *2014 19th European Conference on Networks and Optical Communications-(NOC)*, pp. 136–139, 2014.

20. Liu, Y.-F., C.-H. Yeh, C.-W. Chow, P.-Y. Huang and Y. Liu, "Demonstration of Using Digital FIR Filter and Matched Filter to Increase Data Rate in Visible Light Communication", *Broadband Access Communication Technologies VII*, Vol. 8645, p. 86450N, International Society for Optics and Photonics, 2013.

21. Elgala, H., R. Mesleh and H. Haas, "A Study of LED Nonlinearity Effects on Optical Wireless Transmission Using OFDM", *2009 IFIP International Conference on Wireless and Optical Communications Networks*, pp. 1–5, 2009.

22. Hsu, C.-W., G.-H. Chen, L.-Y. Wei, C.-W. Chow, I.-C. Lu, Y.-L. Liu, H.-Y. Chen, C.-H. Yeh and Y. Liu, "Adaptive Filtering for White-Light LED Visible Light Communication", *Optical Engineering*, Vol. 56, No. 1, p. 016115, 2017.

23. Ghassemlooy, Z., W. Popoola and S. Rajbhandari, *Optical Wireless Communications: System and Channel Modelling with Matlab®*, CRC press, 2019.

24. Uysal, M., F. Miramirkhani, O. Narmanlioglu, T. Baykas and E. Panayirci, "IEEE 802.15. 7r1 Reference Channel Models for Visible Light Communications", *IEEE Communications Magazine*, Vol. 55, No. 1, pp. 212–217, 2017.

25. Kisacik, R., A. D. Yalcinkaya, A. E. Pusane, T. Baykas and M. Uysal, "Character-

ization of LEDs for Visible-Light Communications", *Optical Engineering*, Vol. 60, No. 2, p. 024102, 2021.

26. Erkinaci, T., *Integrated CMOS Receiver System For High Speed Visible Light Communication*, MSc. Thesis, Bogazici University, 2020.

27. Wolf, M., S. A. Cheema, M. Haardt and L. Grobe, "On the Performance of Block Transmission Schemes in Optical Channels with a Gaussian Profile", *2014 16th International Conference on Transparent Optical Networks (ICTON)*, pp. 1–8, 2014.

28. Azim, A. W., Y. Le Guennec and G. Maury, "Performance Analysis of Precoded Layered ACO-OFDM for Visible Light Communication Systems", *Optics Communications*, Vol. 440, pp. 49–60, 2019.

29. *Luminus SST-90 RGB*, `https://www.digchip.com/datasheets/parts/datasheet/1984/SST-90-B-F11-KG300-pdf.php`, accessed in August 2021.

30. *Luminus SST-90-W LEDs*, `https://download.luminus.com/datasheets/Luminus\_SST-90-W\_Datasheet.pdf`, accessed in August 2021.

31. *Broadcom ASMT-MT00*, `https://docs.broadcom.com/doc/AV02-1693EN`, accessed in August 2021.

32. *Cree® XLamp® XR-E LED*, `https://cree-led.com/media/documents/XLamp7090XRE-16F.pdf`, accessed in August 2021.

33. Miramirkhani, F. and M. Uysal, "Channel Modeling and Characterization for Visible Light Communications", *IEEE Photonics Journal*, Vol. 7, No. 6, pp. 1–16, 2015.

34. Uysal, M., T. Baykas, F. Miramirkhani, N. Serafimovski and V. Jungnickel, "TG7r1 Channel Model Document for High-Rate PD Communications. doc: IEEE 802.15-15/0685r0, September 2015", .

35. Wang, H. and S. Kim, "Soft-Input Soft-Output Run-Length Limited Decoding for Visible Light Communication", *IEEE Photonics Technology Letters*, Vol. 28, No. 3, pp. 225–228, 2016.

36. Lu, X. and T. J. Li, "Exploiting RLL Codes in Visible Light Communication", *2016 IEEE/CIC International Conference on Communications in China (ICCC)*, pp. 1–6, 2016.

37. Dhannoon, Z. and R. F. H. Fischer, "Performance of Run-Length-Limited Codes in Visible-Light Communications", *SCC 2019; 12th International ITG Conference on Systems, Communications and Coding*, pp. 1–6, 2019.

38. Thummaluri, u., A. Kumar and L. Natarajan, "Flicker Mitigating High Rate RLL Codes for VLC with Low Complexity Encoding and Decoding", *2018 IEEE International Symposium on Smart Electronic Systems (iSES) (Formerly iNiS)*, pp. 209–214, 2018.

39. Li, Z., H. Yu, B. Shan, D. Zou and S. Li, "New Run-Length Limited Codes in On–Off Keying Visible Light Communication Systems", *IEEE Wireless Communications Letters*, Vol. 9, No. 2, pp. 148–151, 2020.

40. Jain, N. and A. Banerjee, "On Performance Analysis and Code Design for Visible Light Communication", *2019 57th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pp. 536–543, 2019.

41. Mejia, C. E., C. N. Georghiades, M. M. Abdallah and Y. H. Al-Badarneh, "Code Design for Flicker Mitigation in Visible Light Communications Using Finite State Machines", *IEEE Transactions on Communications*, Vol. 65, No. 5, pp. 2091–2100, 2017.

42. Goodfellow, I., Y. Bengio and A. Courville, *Deep Learning*, MIT Press, 2016.

43. Hamalainen, A. and J. Henriksson, "A Recurrent Neural Decoder for Convolu-

tional Codes", *1999 IEEE International Conference on Communications (Cat. No. 99CH36311)*, Vol. 2, pp. 1305–1309, 1999.

44. Balevi, E. and J. G. Andrews, "Autoencoder-Based Error Correction Coding for One-Bit Quantization", *IEEE Transactions on Communications*, Vol. 68, No. 6, pp. 3440–3451, 2020.

45. Aoudia, F. A. and J. Hoydis, "End-to-End Learning of Communications Systems Without a Channel Model", *2018 52nd Asilomar Conference on Signals, Systems, and Computers*, pp. 298–303, 2018.

46. Ye, H., L. Liang and G. Y. Li, "Circular Convolutional Auto-encoder for Channel Coding", *2019 IEEE 20th International Workshop on Signal Processing Advances in Wireless Communications (SPAWC)*, pp. 1–5, 2019.

47. Chi, N., J. Jia, F. Hu, Y. Zhao and P. Zou, "Challenges and Prospects of Machine Learning in Visible Light Communication", *Journal of Communications and Information Networks*, Vol. 5, No. 3, pp. 302–309, 2020.

48. Lee, H., I. Lee, T. Q. Quek and S. H. Lee, "Binary Signaling Design for Visible Light Communication: A Deep Learning Framework", *Optics express*, Vol. 26, No. 14, pp. 18131–18142, 2018.

49. Soltani, M., W. Fatnassi, A. Aboutaleb, Z. Rezki, A. Bhuyan and P. Titus, "Autoencoder-Based Optical Wireless Communications Systems", *IEEE Globecom Workshops (GC Wkshps)*, pp. 1–6, 2018.

50. Lee, H., T. Q. Quek and S. H. Lee, "A Deep Learning Approach to Universal Binary Visible Light Communication Transceiver", *IEEE Transactions on Wireless Communications*, Vol. 19, No. 2, pp. 956–969, 2019.

51. Ulkar, M. G., T. Baykas and A. E. Pusane, "VLCnet: Deep Learning Based End-to-End Visible Light Communication System", *Journal of Lightwave Technology*,

Vol. 38, No. 21, pp. 5937–5948, 2020.