2013 International Workshop on Nonlinear Circuits,
Communications and Signal Processing
NCSP'13, Israrnd of Hawaii, Hawaii, USA. March 4-7, 2013

Development of Simple Simulator for Visible Light Communication Using LED and Camera

Takao Fukumoto†, Shintaro Arai†, Tomohiro Yendo‡,
Takaya Yamazato*, Hiraku Okada* and Toshiaki Fujii*

†Kagawa National College of Technology
551 Kohda, Takuma-cho, Mitoyo, Kagawa, JAPAN
Phone:+81-875-83-8617
Email: a12513@sr.kagawa-nct.ac.jp,
arai@cn.kagawa-nct.ac.jp

‡Nagaoka University of Technology
1603-1 Kamitomioka, Nagaoka, JAPAN

* Nagoya University
Furo-cho, Chikusa-ku, Nagoya, JAPAN

Abstract

This paper develops a simple simulator for the visible light communication (VLC) system using a LED traffic light and a camera for intelligent transport systems (ITS). We use image processing techniques for simulating effects of optical spatial channel which is a unique channel in VLC. We describe a system model of the VLC simulator and perform the simulation with various parameters of the image processing.

1. Introduction

Light emitting diode (LED) has the advantage of long life, low power consumption, low heat generation and good visibility as compared with conventional incandescent lights. In recent years, lighting equipments and traffic lights using LED have become more common. Since LED is semiconductor devices, we can control its intensity electrically at a fast rate. Namely, LED can be used not only as lighting devices but also as communications devices. The communication using the light, which is visible to the naked eye, is called "visible light communications (VLC)" [1]–[4].

In this study, we consider the VLC system using a LED traffic light (transmitter) and a camera (receiver) for intelligent transport systems (ITS) [2]–[4], as shown in Fig.1. By using the LED traffic light, it is possible to use as not only the primary task as the traffic light but also the communication system for vehicles and pedestrians. The camera can individually recognize a transmitter from plural light sources. Further, the parallel data transmission is possible by individually modulating LEDs of the transmitter due to the camera can recognize each LED luminance from the received image.

In the system using the camera, the image processing block is very important to recover data correctly. Especially, the quality of the received image degrades with increasing the communication distance. For example, the camera captures a blurred image due to long communication distance, as shown in Fig. 1. In addition, the size of the LED transmitter in the received image depends on the communication distance. Furthermore, the camera is affected by a shot noise from an ambient light such as sunlight, i.e., the ambient light noise. These phenomena influence the recognition of LED luminance in the receiver. Thus, the error rate of data increases due to the phenomena.

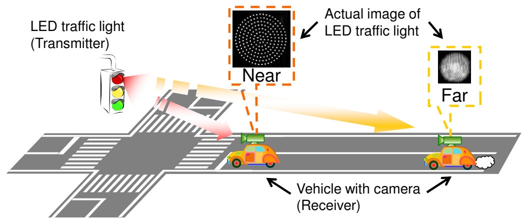


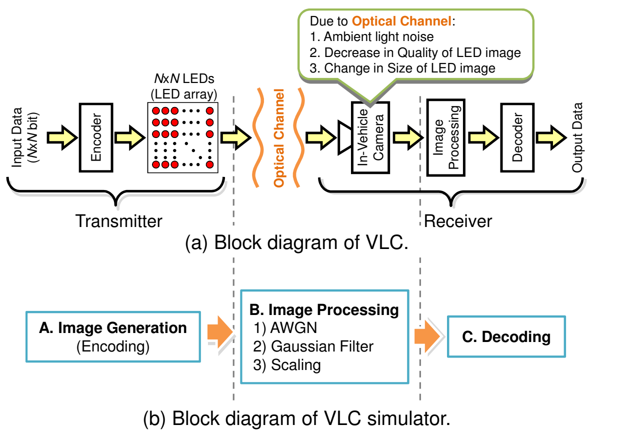
Figure 1: Example image of visible light communication (VLC) using LED and camera for ITS.

As a solution for the problems, we need to carry out computer simulations with various communication environments and to analyze the influence of the transmission channel. Therefore, it is important to develop a simulator of VLC which simulates its specific channel characteristics. In this study, we develop the simple VLC simulator using the image processing techniques as the specific channel. Specifically, the blurred image is simulated by using the Gaussian filter. In addition, the size of the traffic light depending on the distance is simulated by the image scaling processing. Furthermore, we assume the additive white Gaussian noise (AWGN) as the ambient light noise. We describe a system model of the VLC simulator and perform the simulation with various parameters of image processing techniques.

2. System Model of VLC Using LED and Camera

In this section, we introduce the system model of the VLC using LED and the camera in [2]. Figure 2(a) shows a block diagram of the system model.

The transmitter consists of LEDs arranged in $N \times N$ square matrix (i.e., LED array) and a encoder. The transmitter can generate nonnegative pulse and can change a width of pulse. Thus, we can control a LED lighting pattern (i.e., luminance of LED) by changing the width. The transmitter modulates



(a) Block diagram of VLC.

(b) Block diagram of VLC simulator.

Figure 2: System model: (a) Block diagram of VLC using LED and camera, (b) Block diagram of VLC simulator.

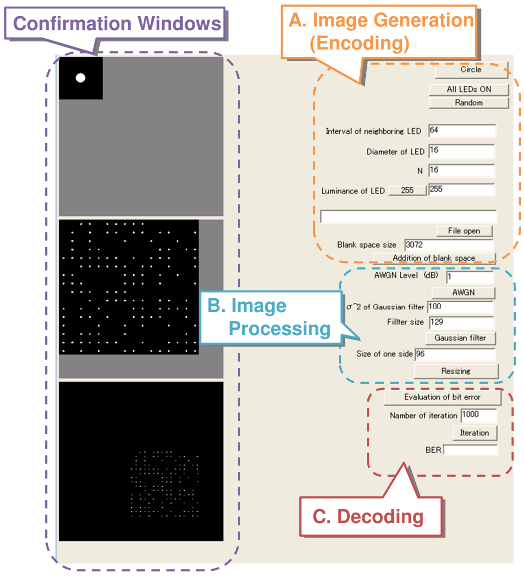


Figure 3: Dialog box of VLC simulator.

data by controlling the luminance. In the case of On-Off Keying (OOK) modulation, the transmitter represents data as On/Off control of the luminance (On='1', Off='0').

The receiver consists of an in-vehicle camera, an image processing unit and a decoder. First, the camera captures the transmitted LED pattern which passes through the optical channel. Second, the image processing unit detects the position of each LED and extracts the luminance. Finally, the decoder demodulates data using the extracted luminance. In the case of OOK, the decoder determines data by the threshold processing of the extracted luminance.

3. VLC Simulator

In this section, we explain our VLC simulator and its operation. Figures 2(b) and 3 show a block diagram of the VLC simulator based on the system model in [2] and a dialog box of the VLC simulator in this study, respectively. The simulator consists of three blocks: Image generation (Encoding), Image processing and Decoding. The correspondence relationship between the actual system and the simulator is shown in Fig. 2.

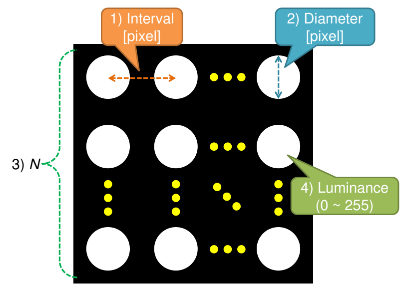


Figure 4: Parameters of LED array for image generation.

3.1 Image Generation (Encoding)

This block generates the image of the LED array. We set four parameters of the LED array: 1) Interval of the neighboring LED [pixel], 2) Diameter of LED [pixel], 3) N, 4) Luminance of LED (Max=255). The detail of each parameter is shown in Fig. 4. According to the parameters, the block can generates two kinds of the lighting pattern ("All LEDs On" and "Random"). We can make an either/or choice between these two patterns and can get the image of the LED array.

Let us focus on the random pattern of LED array. In this case, it is considered that the random pattern is equivalent to the pattern which is encoded by OOK. Namely, we can use the random pattern for the VLC simulation with OOK modulation.

3.2 Image Processing

This block performs the image processing as the optical channel of VLC. We use three kinds of image processing techniques: 1) AWGN, 2) Gaussian filter, 3) Image scaling.

3.2.1 AWGN

The camera is affected by the shot noise from the ambient light (i.e., the ambient light noise). In this study, we assume AWGN as the ambient light noise. AWGN is the wide-band noise with the Gaussian distribution of the amplitude and the constant spectral density. When the ambient light has high-intensity, the shot noise from the ambient light can be modeled as white, Gaussian, and signal/pixel independent [5]. Our simulator adds AWGN with the power spectral density $N_0/2$ to each pixel of the original image.

3.2.2 Gaussian Filter

The Gaussian filter is used for simulating the blurred image. The filter function is expressed by the following equation;

$$h_g(x, y) = \frac{1}{2\pi\sigma^2} \exp\left(-\frac{x^2 + y^2}{2\sigma^2}\right), \tag{1}$$

where σ^2 is the variance of the Gaussian distribution, x and y is the distance from the origin in the horizontal axis and

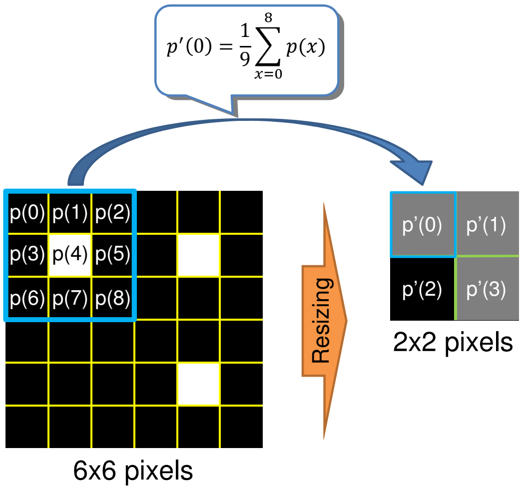


Figure 5: Resizing of image using area averaging.

the vertical axis, respectively. By using the Gaussian function, we perform the filtering for each pixel. In this process, the average of pixel value is calculated by using the focused pixel of the image with noise and its neighborhood pixels. This process allocates the weight of the smoothing depending on the distance from the focused pixel in advance. This weight decreases with increasing the distance from the focused pixel. Based on the condition, the process calculate the weighted mean according to the allocated weight and perform the smoothing for the image. Here, the number of neighborhood pixels and the blur intensity depend on the filter size and the variance σ^2, respectively. We assume this filtered image as the blurred image due to long communication distance.

3.2.3 Image Scaling

The size of LED array depending on the distance is simulated by the image scaling processing. This study uses the area averaging [6] which is one of the image scaling methods. Here, we explain the reduction method using the area averaging. First, we divide the original image (the blurred image) into equally intended image size. Next, the average pixel value is calculated for each divided pixel group. This average value is used as the pixel value of the reduced image. As an example, let us consider the scaling from 6×6 pixels to 2×2 pixels, as shown in Fig. 5. The reduced image value is calculated as

$$p(0) = \frac{1}{9}\sum_{k=0}^{8} p(x), \tag{2}$$

where $p(0)$–$p(8)$ denote the divided pixel group of the original image and $p(0)$ denotes its average value, i.e., the pixel value of the reduced image.

3.3 Decoding

This block performs a demodulation of data from the processed image and a calculation of the bit error rate (BER).

Table 1: Simulation parameters.

Number of LEDs ($N \times N$)	16×16
Interval of neighboring [pixel]	64
Diameter of LED [pixel]	16
Modulation method	OOK
Size of original LED array image [pixel]	1024^2
σ^2 of Gaussian filter	10, 100, 1000
Filter size	129^2
Scaling size [pixel]	64^2, 32^2,16^2 ,8^2

Table 2: BER performance for different scaling sizes (w/o AWGN).

Scaling size	w/o filter	Gaussian filter	
	–	$\sigma^2 = 10$ or 100	$\sigma^2 = 1000$
64^2 pixels	0.0	0.0	7.2×10^{-2}
32^2 pixels	0.0	0.0	6.6×10^{-2}
16^2 pixels	0.0	0.0	7.8×10^{-2}
8^2 pixels	5.0×10^{-1}	5.0×10^{-1}	5.0×10^{-1}

Here, we explain an operation of decoding. First, the block finds positions (i.e., coordinates) of LEDs from the processed image using its size and the number of LEDs. Second, the luminance of each LED is extracted from the coordinates. Third, the block calculates an average of the extracted luminance values and uses the average value as a threshold for decoding. Fourth, the data is demodulated by a threshold-based decision. The block decides the decoded data as 1 (or 0) if the extracted luminance is larger (or smaller) than the threshold. Finally, BER is calculated by comparing the demodulated data and the data pattern on the transmitting side.

4. Operation Result of VLC Simulator

In this section, we perform the VLC simulator and show its performance. This paper shows output images and BER performance as the simulation results.

4.1 Simulation conditions

Table 1 shows simulation parameters. In this study, the number of LEDs, the interval and the diameter are determined based on [2]–[4]. Also, the filter size is set to the double of the interval. This paper observes two kinds of BER performances. First is BER for scaling sizes without AWGN. Second is BER for various E_b/N_0 [dB] values, where E_b denotes the average energy per bit. In each simulation, we iterate the simulation 10^3 times with the different σ^2 of the Gaussian filter and record BERs.

4.2 Simulation results

Figures 6, 7 and 8 show output images with AWGN, images using the Gaussian filter and resized output images, respectively. From these figures, we can observe that the original image changes according to the simulation parameters. Therefor, our simulator can output various images depending on the VLC environment.

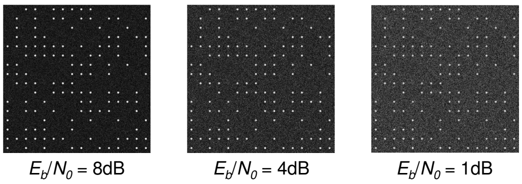


Figure 6: Output images with AWGN.

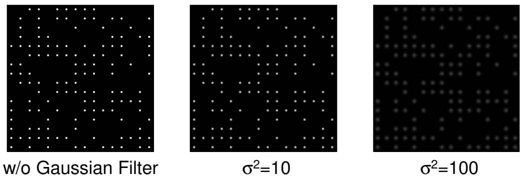


Figure 7: Output images using Gaussian filter.

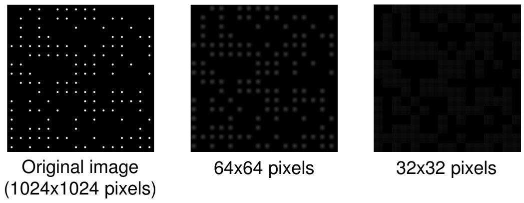


Figure 8: Resized output images using area averaging.

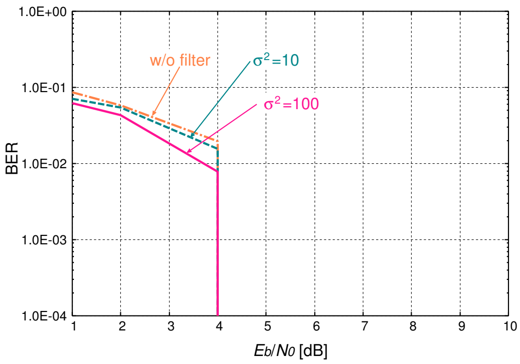


Figure 9: BER vs. E_b/N_0 for different σ^2 (scaling size $= 16^2$ pixels).

Table 2 shows BER performance of our simulator for the different size of the resized image. From this table, we can confirm the error-free transmission when the scaling size is 16^2 pixels or more, and σ^2 is 100 or less. However, we can also confirm that the error occurs in the case $\sigma^2 = 1000$, i.e., the strong blur intensity. This result has been reported in previous researches [2]–[3]. We can say that the phenomenon similar to the previous researches is also obtained in our simulator. Next, let us focus on BERs when the scaling size$= 8^2$ pixels. In this scaling size, Fig. 2 indicates BER$= 5.0 \times 10^{-1}$. This is because that the image resolution of the LED array area is lower than the number of LEDs (N^2). The luminance value of the image decreases due to the area averaging, as shown in Fig. 8. We consider that it is difficult to demodulate data because of the very low resolution and luminance. In the actual VLC system, the luminance does not decrease significantly because the sensitivity of camera sensor is high. Thus, it is our future work that the performance improvement of our simulator with the sensitivity.

Figure 9 shows BERs versus E_b/N_0 for the different filter size σ^2 when the scaling size $= 16^2$ pixels. We plot three performances: 1) w/o filter, 2) $\sigma^2 = 10$, 3) $\sigma^2 = 100$. From this figure, we can confirm that each BER performance decreases with increasing noise. In other words, our simulator can perform the VLC environment depending on the influence of the ambient light noise. However, we can also confirm BER performance improves with increasing σ^2 in low E_b/N_0 regions. As the reason for this, we focus on the capability of the Gaussian filter. In general, the Gaussian filter cuts high-frequency components contained in the image [6]. Namely, the Gaussian filter operates as the low-pass filter. Our simulator adds AWGN to each pixel of the original image. When E_b/N_0 is low, high-frequency components contained in the image increase. Therefore, we consider that noise components contained the image decrease because the filter of $\sigma^2 = 100$ greatly cuts its high-frequency components as compared with others.

5. Conclusions

This paper has developed the simple VLC simulator using image processing techniques as the specific channel of VLC. We have achieved that our simulator outputs images and records BER performances depending on various VLC environments by using the image processing.

Acknowledgment

This work is supported in part by "JSPS KAKENHI Grant Number 24760307", "Project of Education Research in Cooperation with KOSEN by Toyohashi University of Technology" and "Collaborative Research Grant by Nagaoka University of Technology".

References

[1] M. Akanegawa, Y. Tanaka and M. Nakagawa, "Basic study on traffic information system using LED traffic lights," *IEEE Trans. Intelligent Transportation Systems*, Vol. 2, No. 4, pp. 197-203, Dec. 2001.

[2] S. Arai, S. Mase, T. Yamazato, T. Yendo, T. Fujii, M. Tanimoto and Y. Kimura, "Feasible Study of Road-to-Vehicle Communication System Using LED Array and High-Speed Camera," *Proc. 15th World Congress on ITS*, Nov. 2008.

[3] Y. Shiraki, T. Nagura, T. Yamazato, S. Arai, T. Yendo, T. Fujii and H. Okada, "Robust Receiver Design for Road-to-Vehicle Communication System Using LED Array and High-Speed Camera," *Proc. 18th World Congress on ITS*, Oct. 2011.

[4] S. Nishimoto, T. Yamazato, H. Okada, T. Fujii, T. Yendo and S. Arai "High-speed Transmission of Overlay Coding for Road-to-Vehicle Visible Light Communication Using LED Array and High-Speed Camera," *Proc. OWC'12*, pp. 1234-1238, Dec. 2012.

[5] J. M. Kahn and J. R. Barry, "Wireless Infrared Communications," *Proc. IEEE*, vol. 85, pp. 265-298, Feb. 1997.

[6] R.C. Gonzalez and R.E. Woods, *Digital image processing (3rd edition)*, Prentice Hall, Aug. 2007.