Deep learning based detection of anti-reflective obstacles in VLC systems

Marina E. Plissiti
Computer Science & Engineering Dept
University of Ioannina
Ioannina, Greece
marina@uoi.gr

Christoforos Papaioannou
Computer Science & Engineering Dept
University of Ioannina
Ioannina, Greece
chpapaioannou@cs.uoi.gr

Yiorgos Sfikas
Computer Science & Engineering Dept
University of Ioannina
Ioannina, Greece
gsfikas@uoi.gr

Georgios Papatheodorou
Computer Science & Engineering Dept
University of Ioannina
Ioannina, Greece
g.papatheodorou@uoi.gr

Simon-Ilias Poulis
Computer Science & Engineering Dept
University of Ioannina
Ioannina, Greece
s.poulis@uoi.gr

Aristides Efthymiou
Computer Science & Engineering Dept
University of Ioannina
Ioannina, Greece
efthym@uoi.gr

Yiorgos Tsiatouhas
Computer Science & Engineering Dept
University of Ioannina
Ioannina, Greece
tsiatouhas@uoi.gr

***Abstract*—In Visible Light Communication (VLC) systems, the existence of non-reflective obstacles in the field of view of the receiver may cause deterioration or complete blockage of the optical signal which leads to an error-prone decoding process. We observe that image processing techniques, in combination with deep learning, provide effective ways to deal with this problem. In this work, we propose an efficient technique for the decoding of the signal in frames with partially occluded areas. First, a deep learning detector is used for the extraction of the obstacle, resulting in the determination of the regions of interest in each frame. Then, in a second step the decoding of the signal is employed, based on image processing techniques. The proposed method is evaluated in terms of the size of the obstacle and it is compared with other decoding schemes, providing very promising results.**

***Index Terms*—Visible Light Communication (VLC), Non-Line-of-Sight (NLOS), Wireless Communication, Rolling Shutter, Signal Decoding, Deep learning, YOLO.**

I. Introduction

In the domain of wireless communication, the Visible Light Communication (VLC) technology is constantly developing, providing new systems of high performance. In this context, the integrated LED lighting systems are used to provide both illumination and data communication, which achieves data transfer through the modulation of the signal (light) emitted by LEDs. This inexpensive and reliable technology gains on an ongoing basis the scientific and industrial interest, leading in a new era in communications.

Nowadays the LED-to-Camera systems, which are composed of an LED source as a transmitter and a camera of a smart devise as a receiver, are widely spread and used for many purposes. This is mainly due to the exponential growth of mobile devices with embedded cameras, which renders an extensive field of applications, such as intelligent lighting systems (smart homes, smart cities), indoor positioning, automotive communication, location-based services, healthcare applications, underwater communications [1]. These systems provide privacy-protection and data transmission is achieved without harmful radiation.

The signal decoding procedure is based on the rolling shutter effect of the CMOS sensor of the camera. In each captured frame, dark and bride stripes are depicted, and the width of each stripe is depended on the modulation scheme that is used. With the processing of each frame and the determination of every stripe width, the decoding of the signal is achieved.

It is very important that in the captured frames the rolling shutter effect must be depicted clearly and without any restrictions. However, this is not always possible in the non-line-of-sight (NLOS) systems, where there is no direct path between the LED and the camera. In these cases, the light signal is reflected on homogeneous surfaces and then it is captured by the receiver. Nevertheless, if the surface is not reflective enough or if there is an anti-reflective obstacle in the field of view of the camera, then the received signal is disturbed, since there is severe fluctuation in intensity in the captured frames. The same effect is also present when temporary blockage or shadowing in dynamic environments (i.e. mobile people and obstacles) is observed [2], where a part of the rolling shutter stripes in each frame are vanished.

In order to achieve high accuracy in signal decoding procedure, several techniques have been proposed for the definition of the part of the captured frame that contains useful information. These techniques may concern the processing of the rolling shutter effect in the whole image ([3], [4]) or the extraction of the region of interest (ROI) in the frame, in which the decoding process is focused [5].

In this work, an improved version of our previous algorithm for ROI extraction [5] is presented. More specifically, instead of using image processing techniques and mathematical morphology operators, we define the region of the obstacle that invalidates the rolling shutter effect in each frame, using the deep learning model YOLOv5. This is a fast and efficient way for the accurate definition of the obstacle area, which results in the correct delimitation of the rolling shutter area that can be used in the decoding process. The model is trained in several images offline, and the final model is installed in the receiver (smart phone). Furthermore, we have developed an application that is able to decode the signal in the VLC system in real time. In addition, an extensive search for the determination of the optimal values for the parameters of the camera has been conducted. Experimental results validate the efficiency of the proposed solutions.

II. System Architecture

In our system, the transmitted light signals are generated by a downlight luminaire, which is composed of an LED strip mounted on the inner perimeter of the metallic frame of the luminaire. A semi-transparent plastic diffuser is incorporated in the frame. The smartphone we use as a receiver, is a Xiaomi Redmi Note 9 smartphone and the application we have implemented was built using the Camera2 API as a template. More details are provided in [6].

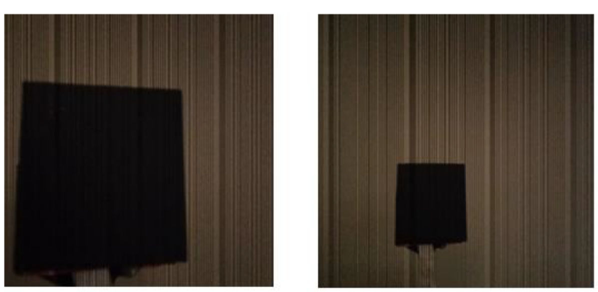

Fig. 1. Representative images in the training set. The rolling shutter effect is covered by a non reflective obstacle (dark area).

III. Methods

A. Obstacle Detection

As it is already mentioned, the existence of non reflective objects in the filed of view of the camera, covers the rolling shutter effect and thus interfere in the decoding process. For this reason, it is necessary to define and reject the region of the object.

Object detectors based on deep learning are extensively used in image processing. More specifically, the later versions of You Only Look Once (YOLO) model, aim at the improvement of the original deep learning model [7], in both complexity and accuracy. Our approach is based on YOLOv5, which results in a fast and accurate detection of the obstacles. The network takes as input the frames recorded by the mobile, and predicts the area of the obstacle within the image. This area is then excluded by the decoding process so that we avoid the potential interference.

A set of 300 frames was created from images obtained by placing an obstacle in front of the reflective surface. The frames were captured by placing the mobile phone at various distances and viewing angles from the obstacle. In this way, the shape and the size of the covered area varies (Fig. 1). For the annotation of these images, the free-source LabelImg [8] was used to label these images (i.e., to determine the region of the obstacle in the image). Thus, for each image a file containing the obstacle coordinates was created.

We have also included data augmentation to increase the number of samples, and the final set is consisted of 767 images, resulting from horizontal and vertical rotation of the images. This image dataset is randomly splitted into a training set of 665 images, a validation set of 68 images and a test set of 34 images.

Before the training of the network, all sample images are uniformly resized to 416×416 pixels. The batch size and epochs were set to 16 and 100 respectively and Stochastic Gradient Descent (SGD) is selected as the optimizer. The model was trained in GoogleColab [9]. It must be noted that the model converges fast as the images are relatively simple and no more additional epochs are needed (Fig. 2)

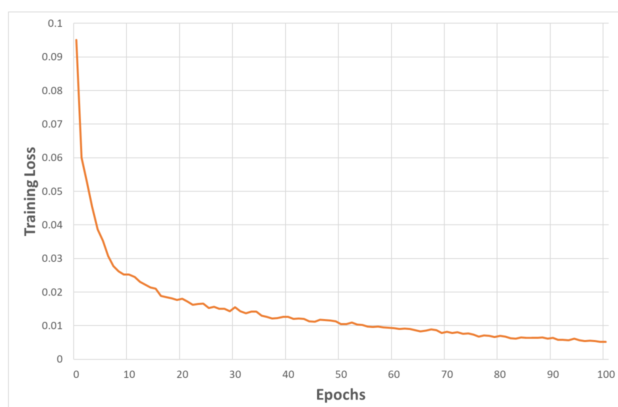

Fig. 2. The training loss reaches the value of $5 \times 10^{-}3$ at the final epoch.

The results of object detection with YOLOv5 are impressive. As it is observed in Fig. 3 the object bounding box is correctly identified in the captured frames. Integrating the trained YOLOv5 model into the mobile phone application required complex mechanisms, but its operation is very fast, efficient, and extracts the obstacle area in real time, as it will be explained in Section IV.

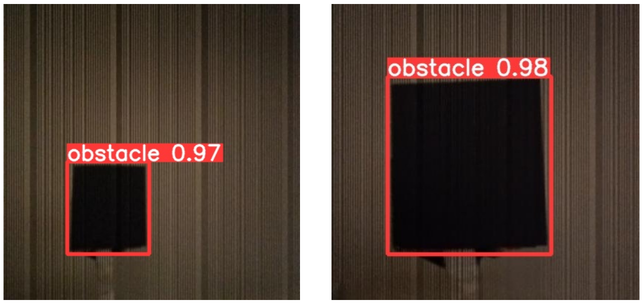

Fig. 3. Obstacle detection with YOLOv5.

B. Signal Decoding

We use the same configuration of VLC signal transmission as it is presented in [6]. Each packet in VLC signal is composed of an 8-bit preamble and a 24-bit payload. Manchester encoding is used for the avoidance of flickering and a packet splitter is used between consecutive packets.

We follow the decoding process that is described in [5]. The region of interest in every frame is determined by the extraction of the anti-reflective object area from each frame. In the remaining part of the image, the adaptive thresholding scheme is used [6] and we obtain a single dimension waveform, from which the signal packets are extracted.

Fig. 4. Application Front-End. At the upper window the rolling shutter effect and the identified obstacle are depicted. At the bottom window, the decoded signal (text) is presented.

IV. Android Application

The Android application that we have developed, (Fig. 4), is flexible to different light conditions and the user sets the camera's parameters in order to capture the rolling shutter effect clearly. Several experiments have been performed in order to investigate the influence of the exposure time and ISO parameters in the decoding process, and the optimal values were selected, as it is explained in section V-A .

The Android app uses a multi-threaded approach to optimize its performance when recording from the camera. Each frame generated during the recording process is assigned to a dedicated thread for processing. This systematic approach ensures efficient use of device resources. In this way, multiple threads are used to simultaneously handle the decoding of the received information.

For the obstacle detection, the Android app uses the trained YOLOv5 model. The advantage of the algorithm is that the detection of the obstacle is obtained in real time. In order to incorporate YOLOv5 in the adroid app, it is necessary to convert it to TensorFlow Lite (.tflite) format. This is an optimized version for mobile platforms, '.tflite' models integrate seamlessly into Android apps. Having prepared a valid '.tflite' file, an asset folder is created inside the android app and the model file is added to this folder. In the Android application code, access to the model was implemented using the AssetManager to read the file from the asset folder, and the received camera frame is inserted as input to the model. Thus, each thread can use this '.tflite' model and extract the positions of the obstacles present on each frame.

As the camera records, each individual frame is assigned to its own dedicated thread. This thread is responsible for processing and decoding the information contained in the frame. This distribution of threads allows for parallel processing, greatly improving the overall performance and responsiveness of the application. Each dedicated thread is responsible for generating a portion of the data required for the final output.

These partial outputs serve as building blocks for the final output. After experiments we choose that a mobile application needs five threads, one of which is responsible for collecting all production data from the other threads (main thread). This main thread acts as a coordinator, responsible for orchestrating the entire process. The main thread collects and aggregates the production data generated by all other threads. This ensures that the information processed by each thread is combined to form a coherent output. To maintain the integrity of the recorded content, the main thread checks the packet ID of each data packet. This packet ID is used to determine the correct sequence of decoded packets.

V. Experimental Results

A. Selection of camera parameters

The quality of the recorded images is not only depended on the brightness (lux value) of the reflecting surface, but also on the camera parameters. As it was observed, the values for ISO and exposure time affected the images captured by the mobile phone. Thus, small values of ISO

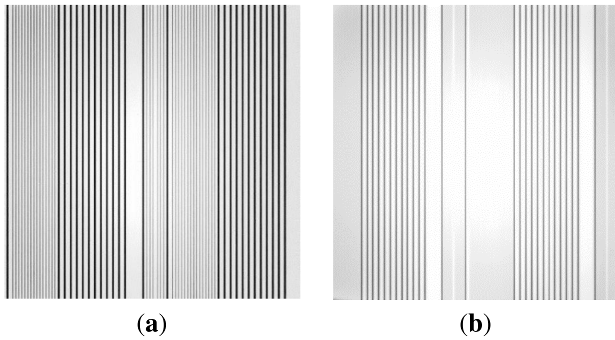

Fig. 5. Effect of different values of exposure time on image quality. (**a**) exposure time=8000^{-1} sec , (**b**) exposure time=4000^{-1} sec.

produce images with low average brightness, while higher values results in images with high level of brightness. In addition, exposure time also affects the brightness of images. More specifically, the longer the exposure time, the brighter the resulted image (Fig. 5).

It is clear that in order to correctly decode the signal from the images, there should be a distinction between light and dark stripes of the rolling shutter effect. Thus, suitable values for the camera parameters must be determined, which will result in images with the appropriate brightness and contrast. These values should be effective for a wide range of conditions (external lighting, source intensity, reflection surface brightness).

In order to define the optimal values for ISO and exposure time parameters of the camera, experiments were performed with different values, taking into account the brightness on the reflection surface. More specifically, 4 different values were tested for the ISO, and 3 different values for the exposure time and the brightness on the reflecting surface respectively. Thus, experiments were performed with all possible combinations of the values shown in Table I. Finally, the effectiveness of the decoding method based on the combination of these values was examined and the number of signal transmissions required to recover the entire message was calculated.

From the diagrams in Fig. 6, it is observed that for all lux values, the results were optimal for the values 3200 for ISO, and 1/12000 for the exposure time. With these values our method obtains very good results for a wide range of luminance of the reflecting surface, which is the desirable condition for the application, i.e. to produce accurate results regardless the level of the luminance.

B. Method Performance

The proposed method was compared with both the decoding procedure that does not include the obstacle detection step [6], and with the decoding procedure which includes the obstacle detection algorithm based on mathematical morphology [5], for several sizes of obstacles.

TABLE I
VALUES OF PARAMETERS

ISO	Exposure Time	LUX
2000	8000^{-1}	200
3200	12000^{-1}	350
6400	16000^{-1}	500
8000	-	-

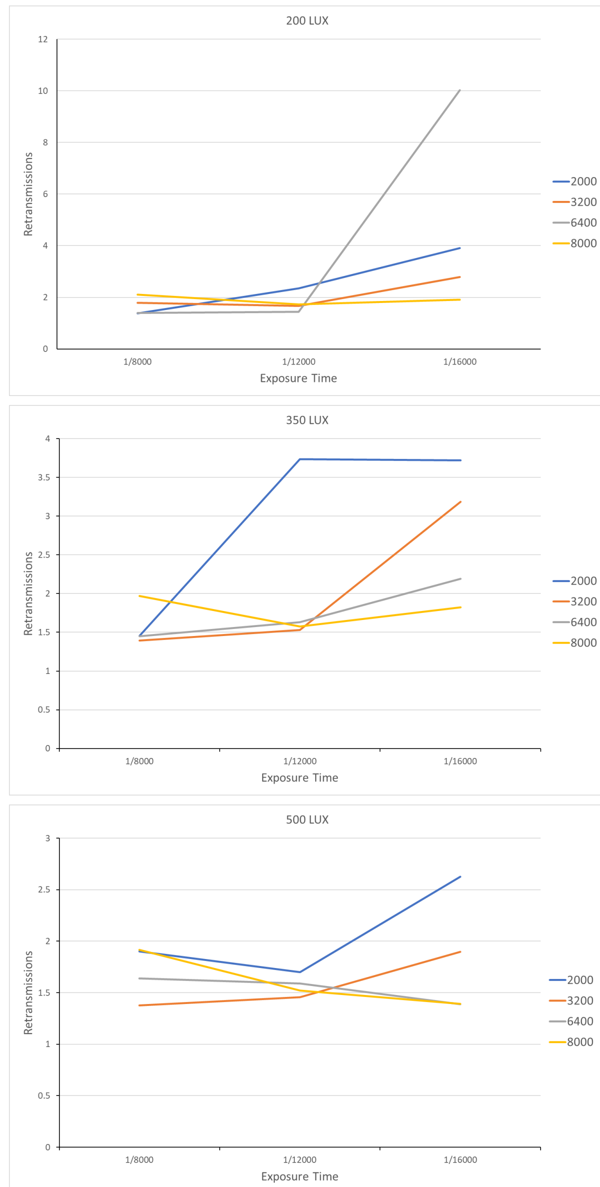

Fig. 6. Influence of different values of the camera parameters in the method performance (total number of retransmissions). Three experiments with different levels of the brightness on the reflection surface of 200, 350 and 500 Lux were performed. The curves in each diagram refer to different values of ISO. As we can observe, the optimal values of exposure time is 12000^{-1} sec and for ISO is 3200.

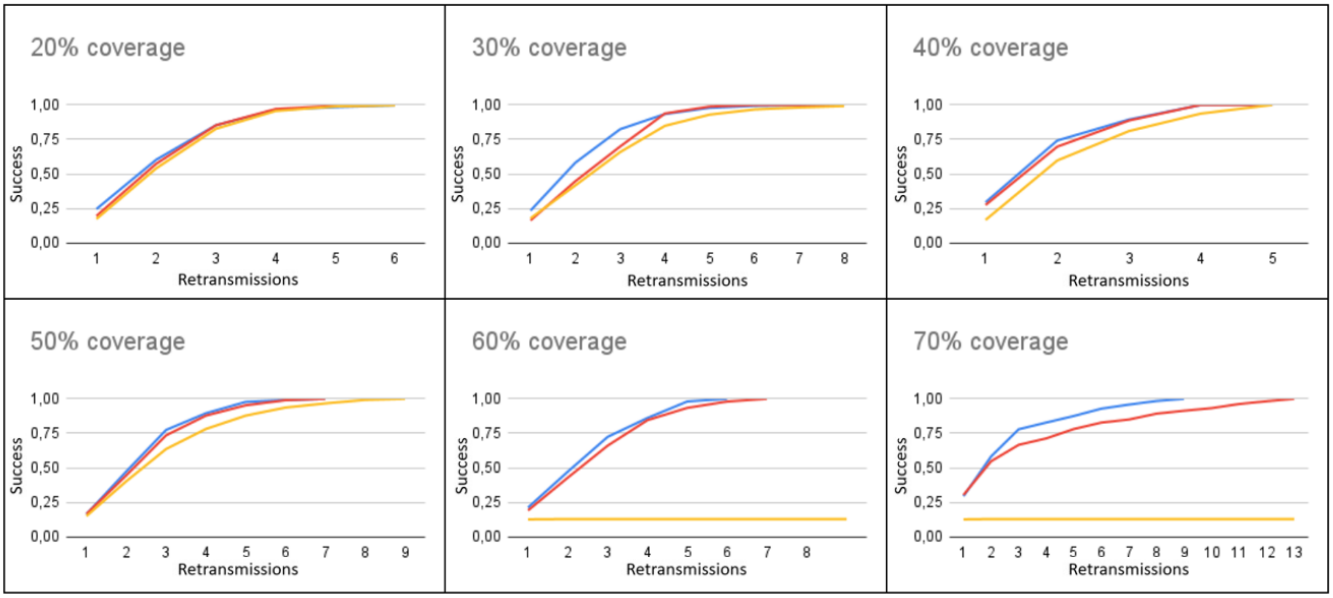

Fig. 7. Comparison of the performance of our method and the methods proposed in [6] and [5]. The results of our method are depicted with the blue line. The results of the methods presented in [6] (without obstacle detection) and in [5] (obstacle detection based on mathematical morphology) are depicted with the yellow and the red line respectively.

The LED source was placed at a distance of 2 meters from the wall used as a reflection surface, with the luminance varied between 210 and 350 lux, for several sizes of obstacles (percentage coverage). The camera was placed at a distance of 1.5 meters. Both the LED and the camera were perpendicular to the wall. We have also placed a mat black paper with several sizes in the field of view of the camera, which produce a a partial overlap of the rolling shutter effect, represented in Fig. 7 as a percentage coverage of the total area of the frame, which varies between 20% and 70%.

The frames are recorder through the application, which inserts each frame as input to the trained YOLOv5 model. The delimitation of the area of the obstacle is obtained in real time and then the signal decoding procedure is performed. The estimation of the effectiveness of our method is based on the number of the retransmissions needed to recover the entire information of the signal, in the presence of various obstacles. It must be noted that the message is consisted of 64 packets of 32 bits each.

The signal is continuously transmitter to cope with the inherent losses due to the interframe gap. The number of correctly decoded data packets is calculated in each transmission. Each experiment is repeated thirty times to ensure the methods' robustness, and an average number of decoded packets is calculated.

The comparison of our method with different signal decoding algorithms is shown in Fig 7. We observe that when the obstacle coverage is 20%, the algorithms show similar performance, while for higher coverage values, the decoding process is improved when using the trained neural networks as obstacle detector, since the number of signal retransmissions is smaller. More specifically, our method can detect the entire signal in 6 and 9 retransmissions for 60% and 70% frame coverage, respectively. Furthermore, for these overlap rates, using the entire image to decode the signal leads to unacceptable results (yellow line).

VI. Conclusion

An automatic obstacle detection algorithm is necessary to correctly decode the signal in VLC systems. We have developed a method that uses the deep learning model YOLOv5 as the object detector for the delimitation of the obstacle area of in a frame of a camera, when the rolling shutter effect is exploited to receive the signal. As it was verified by the experimental results, the method presents high performance, even in cases of high coverage in the rolling shutter effect, captured by the receiver camera. Our method is suitable for real time signal decoding in LED-to-camera NLOS VLC systems that follow Manchester encoding. As a future work, we intent to test our method in in various signal transmission scenarios and extent it, in order to deal with frames that contain two or more obstacles.

References

[1] L. E. M. Matheus, A. B. Vieira, L. F. M. Vieira, M. A. M. Vieira and O. Gnawali: Visible Light Communication: Concepts, Applications and Challenges. IEEE Communications Surveys & Tutorials, vol. 21, no. 4, pp. 3204-3237, Fourthquarter 2019, doi: 10.1109/COMST.2019.2913348IEEE.

[2] H. Du, C. Zhang and Z. Wu: Robust Beamforming-aided Signal Recovery for MIMO VLC System with Incomplete Channel. IEEE Access, vol. 7, pp. 128162-128170, 2019, doi: 10.1109/ACCESS.2019.2939847.

[3] W.-C. Wang, C.-W. Chow, L.-Y. Wei, Y. Liu, and C.-H. Yeh: Long distance non-line-of-sight (NLOS) visible light signal detection based on rolling-shutter-patterning of mobile-phone camera. Opt. Express, 25, 10103-10108 (2017)

[4] Z. Zhang, T. Zhang, J. Zhou, Y. Lu and Y. Qiao: Mobile phone Camera Based Visible Light Communication Using Non-Line-of-Sight (NLOS) Link. In: 2018 International Conference on Network Infrastructure and Digital Content (IC-NIDC), Guiyang, China, 2018, pp. 35-39, doi: 10.1109/ICNIDC.2018.8525559.

[5] C. Papaioannou, M. E. Plissiti, Y. Sfikas, G. Papatheodorou, S.-I. Poulis, A. Efthymiou and Y. Tsiatouhas: Signal decoding in an NLOS VLC system with the presence of anti-reflective obstacles. In: 2022 IEEE International Black Sea Conference on Communications and Networking (BlackSeaCom), Sofia, Bulgaria, 2022, pp. 303-309, doi: 10.1109/BlackSeaCom54372.2022.9858232.

[6] M. E. Plissiti, C. Papaioannou, Y. Sfikas, G. Papatheodorou, S.-I. Poulis, A. Efthymiou and Y. Tsiatouhas: An efficient adaptive thresholding scheme for signal decoding in NLOS VLC systems. In: 2021 IEEE International Mediterranean Conference on Communications and Networking (MeditCom), Athens, Greece, 2021, pp. 378-382, doi: 10.1109/MeditCom49071.2021.9647560.

[7] J. Redmon, S. Divvala, R. Girshick and A. Farhadi: You Only Look Once: Unified, Real-Time Object Detection. arXiv:1506.02640, 2016

[8] LabelImg: Graphical Image Annotation Tool. https://github.com/tzutalin/labelImg

[9] E. Bisong: Google Colaboratory. In: Building Machine Learning and Deep Learning Models on Google Cloud Platform. Apress, Berkeley, CA, 2019. https://doi.org/10.1007/978-1-4842-4470-8_7