Research Article | Vol. 30, No. 12 / 6 Jun 2022 / Optics Express 20261
Optics EXPRESS

Deep learning for signal clock and exposure estimation in rolling shutter optical camera communication

CRISTO JURADO-VERDU,[1,*] VICTOR GUERRA,[2] JOSE RABADAN,[1] AND RAFAEL PEREZ-JIMENEZ[1]

[1] *Photonics and Communications Technology Division, Institute for Technological Development and Innovation in Communications (IDeTIC), Universidad de Las Palmas de Gran Canaria (ULPGC), Polivalente II, Planta 2, Las Palmas de Gran Canaria, PC: 35017, Spain*
[2] *Pi Lighting Sarl, Sion, Switzerland*
[*] *cjurado@idetic.eu*

Abstract: In rolling shutter (RS)-based optical camera communication (OCC) links, selecting the appropriate camera's exposure time is critical, as it limits the reception bandwidth. In long exposures, the pixels accumulate over time the incoming irradiance of several consecutive symbols. As a result, a harmful intersymbol interference corrupts the received signal. Consequently, reducing the exposure time is required to increase the reception bandwidth at the cost of producing dark images with impracticable light conditions for human or machine-supervised applications. Alternatively, deep learning (DL) equalizers can be trained to mitigate the exposure-related ISI. These equalizers must be trained considering the transmitter clock and the camera's exposure, which can be exceptionally challenging if those parameters are unknown in advance (e.g., if the camera does not reveal its internal settings). In those cases, the receiver must estimate those parameters directly from the images, which are severely distorted by the exposure time. This work proposes a DL estimator for this purpose, which is trained using synthetic images generated for thousands of representative cases. This estimator enables the receiver operation under multiple possible configurations, regardless of the camera used. The results obtained during the validation, using more than 7000 real images, registered relative errors lower than 1% and 2% when estimating the transmitter clock and the exposure time, respectively. The obtained errors guarantee the optimal performance of the following equalization and decoding receiver stages, keeping bit error rates below the forward error correction limit. This estimator is a central component of any OCC receiver that operates over moderate exposure conditions. It decouples the reception routines from the cameras used, ultimately enabling cloud-based receiver architectures.

1. Introduction

Optical camera communication (OCC) is a branch of visible light communication (VLC) [1], in which the optical receivers are the pixels of an image sensor. The interest in this technology lies in the reuse of embedded cameras in a wide range of end-user devices (e.g., smartphones, vehicle dashcams, laptops). In this way, it is intended to break the market's entry barriers imposed on VLC due to the need of using dedicated reception hardware (i.e., photodiode-based receivers) [2].

Notwithstanding, the actual reuse of rolling-shutter (RS) cameras for simultaneous data acquisition and scene visualization is challenging. Using RS-cameras as receivers requires the camera to be optimally configured to achieve the highest link throughput [3]. Its exposure time should be minimized, otherwise, it will restrict the available reception bandwidth (acting as a low pass filter [4,5]). However, under short exposure conditions, the camera delivers dark images

#458538 https://doi.org/10.1364/OE.458538
Journal © 2022 Received 15 Mar 2022; revised 13 May 2022; accepted 16 May 2022; published 23 May 2022

Optics EXPRESS

with impracticable light conditions for human or machine-supervised applications (i.e., users cannot perceive objects in the scene) [4].

Consequently, a tradeoff appears in selecting the suitable exposure time for a particular application. To obtain an in-depth understanding of how the exposure time affects communications is necessary to examine the RS-cameras' image acquisition mechanism [6]. In contrast to global shutter (GS) cameras that expose the whole image sensor during acquisition, RS cameras expose it sequentially row-by-row of pixels from top to bottom as shown in Fig. 1. In this acquisition, each row is activated after transcurred a fixed interval, known as the row sampling time, T_S. After the activation, the row's pixels remain exposed to light during a configurable amount of time, known as the exposure time, t_{exp}. As a result, each row samples the light at different sampling instants, generating an image composed of different intensity bands depending on the illumination state of the transmitter [6–9].

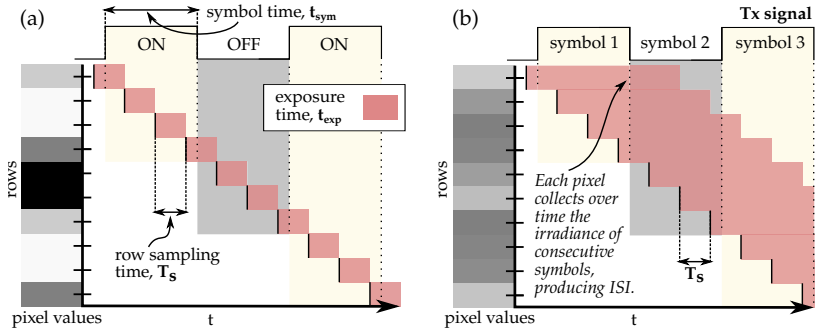

Fig. 1. Rolling shutter image acquisition mechanism.

Figure 2(a) shows an example of the generation of the symbol bands (or stripes) for binary transmission. From this figure, it can be extracted that the row sampling time (which generally ranges from tens of nanos to microseconds) constitutes, essentially, the receiver's sampling period. In contrast, in GS cameras, the receiver's sampling period corresponds to the time elapsed between two consecutive frames [7]. Thus, RS cameras outperform GS cameras in terms of achievable data rate [10].

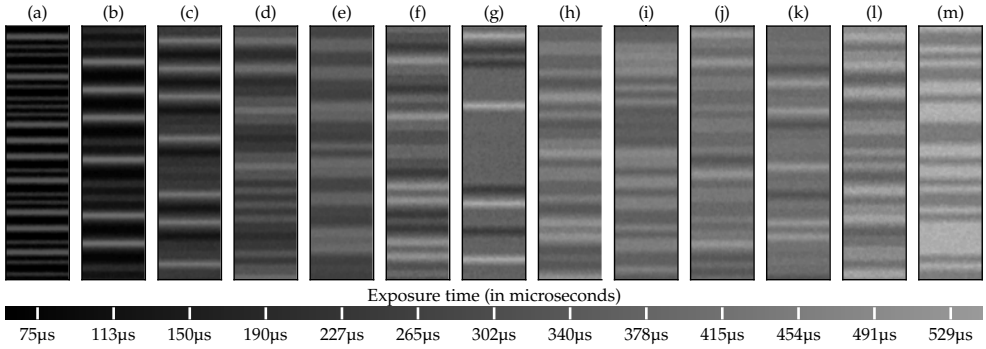

Fig. 2. Captures of the same transmission with increasing exposure time.

Fig. 1(a) illustrates an ideal sampling, in which the exposure time equals the row sampling time ($t_{exp} \leq t_{row}$). In this case, there is no overlap between the exposure of two consecutive rows. On the other hand, Fig. 1(b) illustrates a case where the exposure time is roughly 2.5 times greater than the symbol time. In this case, the pixels accumulate, over time, the incoming light irradiance

Optics EXPRESS

of several consecutive symbols, corrupting the transmitted signal with a devastating intersymbol interference (ISI). As seen in this figure, the received signals are comparatively different. Their brightness increases with the exposure as the pixels accumulate light over a longer time, while their peak-to-peak amplitude decreases and varies more slowly. Figure 2 shows real captures under increasing exposure times for the same transmission (i.e., same frequency and data). It can be appreciated that these examples significantly differ either in shape or brightness, despite capturing the same transmitted signal, which reveals the complexity faced by the receiver at the decoding stage.

In conclusion, the exposure time has devastating effects, which are more detrimental depending on the ratio between the exposure and the symbol time, called the exposure to symbol ratio (ESR) [4]. Experimentally it can be demonstrated that the exposure effects are neglectable when the exposure time is less or equal to half the symbol time ($t_{\text{exp}} \leq t_{\text{sym}}/2$ or ESR $\leq 1/2$). This criterion remarkably restricts the number of different camera devices that can be used as receivers for a given transmitter. First, not all cameras can achieve the required short exposure. Second, they might rely on automatic algorithms for setting the exposure based on ambient light conditions. Finally, they might not allow access to their internal settings. Therefore, it is important to propose equalization stages to mitigate the inevitable exposure-related ISI.

To improve the reception's bandwidth, Liqiong et al. [11] propose a CNN-based neural architecture for demodulating on-off keying (OOK) signals captured with an RS camera. In addition, Yun-Shen et al. [12,13] propose two different AI-assisted approaches for demodulating 4-level pulse amplitude modulation (PAM4) signals: a pixel-per-symbol labelling (PPSL) and a long short-term memory neural networks (LSTM). Despite these architectures notably improve data decoding, their performance has been evaluated in relatively good exposure conditions. Based on the configured exposure time and the attained baud rates, in these works, the exposure time does not exceed the symbol time (ESR ≤ 1).

On the other hand, Younus et al. [14–16] propose the use of a one-dimensional artificial neural network (ANN) equalizer. This network can equalize OOK (or 4PAM) signals affected by exposure times that are up to 4 times the symbol time (ESR ≤ 4), improving the receiver's bandwidth by nine times compared to non-equalized receivers. However, this approach did not take advantage of the spatial redundancy of the data replicated in nearby columns [17], which can be exploited to increase the signal-to-noise ratio (SNR). Hence, the proposed equalizer is very sensitive to noise. In addition, the experimental setup used to train the system was very complex to conduct. It requires placing the receiver camera alongside a photodiode to obtain the training signals. In other words, the authors propose an ad-hoc solution that involved a highly time-consuming training phase for equalizing the images captured by a specific camera.

Alternatively, a two-dimensional convolutional autoencoder (CAE) equalizer trained using exclusively synthetic images was proposed in a previous work [4]. The validation (with real images) demonstrated that this network records bandwidth improvements of up to 14 times for OOK signals (compared to non-equalized receivers) under moderate SNR conditions (12 to 18dBs). This equalizer can decode signals with exposure times that are up to seven times longer than the symbol time (ESR ≤ 7), attaining bit error rates (BER) below the forward error correction (FEC) limit (i.e., 3.8×10^{-3}). Furthermore, since the training is conducted using synthetic images, it can be done offline and on-demand for a wide range of RS cameras with different characteristics.

For all the works above mentioned there is a common requirement. The receiver must be tuned, or adjusted, to the transmitter's clock frequency and the camera's exposure time. Meeting this requisite is exceptionally challenging when these parameters are unknown in advance or cannot be established. For example, when the transmitter dynamically adapts its transmission frequency, or, when the camera does not provide access to its internal settings [18], or dynamically adjusts its configuration based on ambient light estimations. In those cases, the receiver must recover

Optics EXPRESS

those parameters directly from the received images, which is an exceptionally complex task because the images are severely distorted by the exposure.

In this work, a novel estimator block based on convolutional neural networks (CNN) is proposed to address this challenge. This estimator, pretrained with large synthetic datasets, ingests real images containing data packets and estimates the required signal parameters delivering them to the following equalization and the decoding stages: the signal clock and the camera's exposure time. Furthermore, the training dataset contains thousands of representative cases and noise conditions, enabling the estimator to operate effectively in many configurations, regardless of the camera used in the final deployment. The role of the proposed estimator becomes indispensable in any RS-OCC link that operates over moderate exposure conditions. It is responsible for recovering, under harsh conditions, the exposure time used for adjusting the equalizers and the signal clock required for data decoding. Consequently, it allows decoupling of the receiver's equalization and decoding routines from the hardware used (i.e. cameras), enabling cloud architectures that can handle many different image streams.

In addition, in this work, a dataset containing more than 7000 real-captured images for different exposure times and transmission frequencies was generated and released to the scientific community (Dataset 1 [19]).

The remainder of this paper is organized as follows. Section 1 introduces the proposed architecture and details the role of the estimator, which inputs it takes and which parameters deliver to the equalizer and the decoder in the reception chain. Section 2 describes the methods, procedures, metrics and the experimental setup used to evaluate the estimator's performance, including the synthetic network training and the validation using real captured images. Section 3 presents the results. Finally, the conclusions of this work are summarized in section 4.

2. Communications scheme

Figure 3 shows the proposed system architecture and its functional blocks. At the transmitter side, a uniform illuminated flat-panel LED sends non-return to zero (NRZ) Manchester encoded on-off keying (OOK) pulses. Pseudo-random data bits are grouped into packets with a fixed length and enclosed with a header consisting of five consecutive ones and a zero-bit trailer. In addition, after three consecutive bits, a stuffed bit is inserted, preventing the header sequence from appearing within the payload. This redundant bit is set to one if the preceding bit is zero or zero otherwise. For instance, considering the following payload bits, '11011', the transmitted data packet is '11111-110-1-11-0'. On the receiver side, two independent subsystems are interconnected through a shared interface: the image and data acquisition subsystems. The former consist of a generic RS-camera that continuously delivers a stream of compressed JPEG images. This camera is configured, if possible, to select its lowest exposure time required for human or machine-supervised applications. The data acquisition subsystem includes functional software components that can be either embedded in a hardware platform physically interconnected to the camera or deployed in a cloud infrastructure. The first block, the stream manager, controls different image streams and constitutes the input interface to the reception chain. Through this interface, it receives images and stores them in buffers until decoding resources are available.

The blocks that follow the stream controller are responsible for recovering the data embedded in the image: the equalizer, the decoder and the estimator blocks. In the proposed architecture, the equalizer consists of a bank of pre-trained CAEs (detailed in [4]) trained for different exposure times and signal clocks. Their function is to mitigate the exposure-related ISI and reduce the noise. On the other hand, the decoder uses 2D correlation techniques for packet detection, synchronization, and data decoding (as detailed in [20]). Figure 3 shows that these subsystems receive as inputs two important parameters: the number of pixels per symbol (NPPS) and the exposure to symbol ratio (ESR). The former, as introduced in [16], represents the theoretical number of samples per symbol. It is the ratio between the symbol time, t_{sym} and the RS-camera's

Optics EXPRESS

row sampling period, T_s, (NPPS = t_{sym}/T_s). Hence, this parameter is directly related to the signal clock. As an example, in Fig. 1 the signal's NPPS is 3, i.e., theoretically, 3 pixels (or samples) is the expected height for each symbol. Emphasize that the NPPS is not necessarily an integer value. The latter is the ratio between the exposure, t_{exp} and the symbol times, t_{sym}. It indicates how deteriorated is the signal due to the exposure time. As an example, in Fig. 1(a) the ESR is lower than 0.5 (i.e., ≈ 0.33) and, hence, the receiver is operating in optimal sampling conditions ($t_{exp} \leq t_{sym}/2$). However, in Fig. 1(b), the ESR is 2.5, and the ISI is significant. In conclusion, these dimensionless parameters (i.e., NPPS and ESR) characterize the signal received and allow the equalizer to be adjusted for its optimal performance and the decoder to recover the signal clock and proceed with data acquisition.

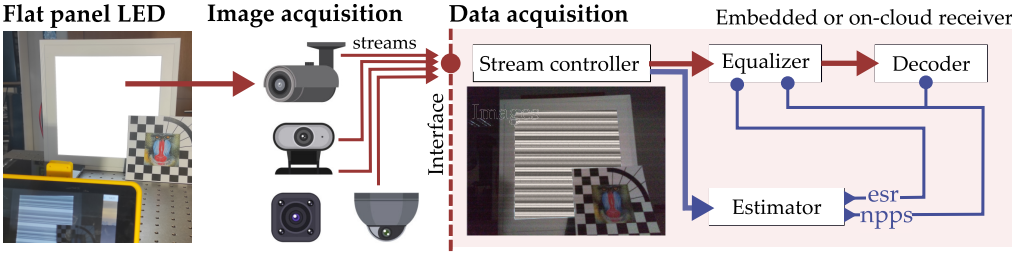

Fig. 3. Embedded or on-cloud receiver scheme.

Finally, the estimator block is responsible for predicting those parameters when they cannot be set or known in advance. To accomplish this, it ingests real images containing data packets and delivers the NPPS and ESR estimates to the corresponding blocks in the reception chain. In this way, the receiver communication algorithms are decoupled from the hardware used to stream the images, even from the camera's configuration.

It should be highlighted the complexity of estimating those parameters from the received signal when the exposure comes into play. Under ideal exposure conditions, the NPPS can be easily estimated by counting the number of sequences of ones and zeroes samples after thresholding the image with a fixed value, as detailed in [21,22]. In those works, the NPPS is known as the stripe width. Moreover, it is unnecessary to estimate the ESR as there is no need to perform the exposure-related equalization. This strategy for estimating the NPPS is accurate only under ideal sampling conditions. In any other case, the ISI severely distorts the signal, altering its shape, intensity and temporal evolution. This estimator is based on a convolutional neural network (CNN) (Fig. 4). The topology of this network consists of a sequence of feature extraction blocks (Fig. 4(b)) that obtains a set of features from the input images (Fig. 4(a)), and a regression artificial neural network (ANN) (Fig. 4(c)), that estimates the parameters as a function of those latent features. The feature extraction blocks are composed of depthwise separable convolutional layers (DSC) [23], replacing the 2D convolutional neural layers (Conv2D) used in traditional CNNs. Compared to the latter which performs the spatial and channel-wise convolution of the inputs using three-dimensional kernels (Fig. 4(e)), DSC layers split the computation into two steps (Fig. 4(f)): a depthwise convolution, followed by a pointwise convolution. The advantage of using DSC layers compared to Conv2D layers is that the number of training parameters is considerably reduced. Figure 4(e) compares the Conv2D and the DSC layers. The former uses a unique kernel, K, with the same number of channels (depth) as the input, I. The result of the convolution operation, Z, is a two-dimensional tensor, which is expressed in Eq. (1).

$$Z[i,j] = (I \circledast K)[i,j] = \sum_{m=0}^{k_w-1} \sum_{n=0}^{k_h-1} \sum_{l=0}^{k_d-1} I[m,n,l] \cdot K[i-m, j-n, l] \tag{1}$$

Optics EXPRESS

where i, j are the output's indexes, m, n, l, the horizontal, vertical and channel kernel's indexes, and k_w, k_h, k_d, the kernel's width, height and depth. On the other hand, a DSC separates this computation into two steps. First a depthwise convolution is applied to each independent channel of the image, I, using a two-dimensional kernel, $K_{\text{depthwise}}$. The outputs for each iteration are then stacked together, obtaining and output tensor that has the same depth as the input, expresed in Eq. (2). In this step, a kernel vector, $K_{\text{pointwise}}$ with a length, k_d equals to the number of input's channels is convolved with every spatial point of the stack, resulting in a two dimensional tensor, Z, which is expressed in Eq. (3).

$$Z'[i,j,k] = \sum_{n=0}^{k_w-1} \sum_{m=0}^{k_h} I[m,n,k] \cdot K_{\text{depthwise}}[i-m, j-n] \tag{2}$$

$$Z[i,j] = \sum_{l=0}^{k_d} Z'[i,j,l] \cdot K_{\text{pointwise}}[l] \tag{3}$$

Following the convolution, the outputs are biased ($B[i,j]$) and transformed using a non-linear activation function, ϕ, which generates a feature map, $F[i,j]$ (Eq. (4)) for the next stage.

$$F[i,j] = \phi(Z[i,j] + B[i,j]) \tag{4}$$

The non-linear activation functions used in this work are the Leaky Rectified Linear Unit (Leaky ReLU) and the Sigmoid functions [24,25]. Afterwards, a sequence of pooling layers replaces the outputs in specific locations with a statistical summary of the outputs in the vicinity. This contributes to increase the non-linearity of the outputs and reduces the total number of network parameters. In this model, max-pooling and average pooling layers are used. The former returns the maximum value and the second the average value of a rectangular patch. This pooling stage starts with an average pooling layer (Fig. 4(g)) with kernel's dimensions 1xN where N is the number of columns considered. Then it follows a max-pooling layer (Fig. 4(h)) Mx1 where M is the number of rows.

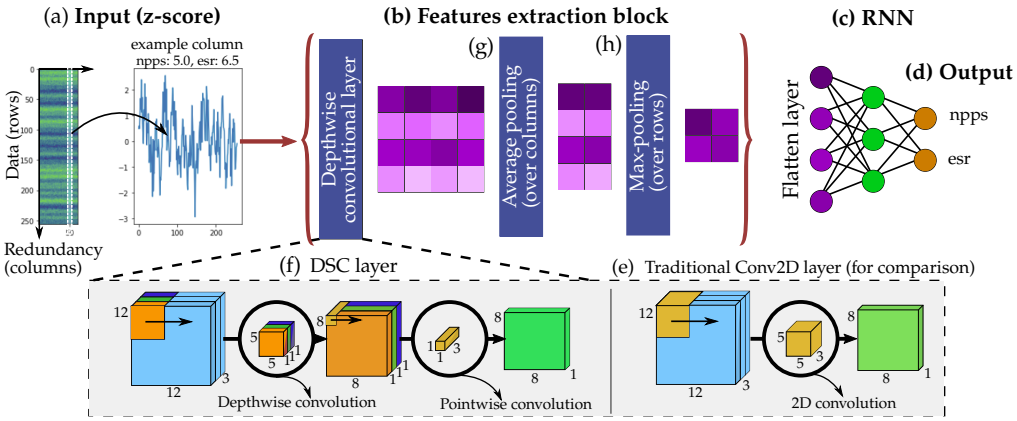

Fig. 4. Deep learning estimator model.

Finally, the RNN consists of a dense neural network with a single hidden layer and an output layer of 2 neurons, one per estimated parameter (i.e., NPPS and ESR). Also, as the outputs are normalized between zero and one, the activation function used in this case is the softmax.

Optics EXPRESS

3. Methodology

The evaluation of the proposed estimator's performance takes place in the following phases. In the first phase, both the synthetic and real datasets were generated. The algorithm used to create the synthetic images is detailed in [4]. For generating the real dataset (Dataset 1 [19]), the images were captured using the experimental setup shown in Fig. 5. It consists of a flat-panel LED pointing towards an RS-camera separated at a distance of 20 cm. At this distance, the transmitter occupies the image entirely. The signal is generated using an arbitrary wave generator, and a power supply is used to feed the light source. Table 1 summarizes the key parameters of the experiment necessary for its replicability.

The real dataset (Dataset 1 [19]) contains more than 7000 image samples for different NPPS and ESR. The NPPS values range from 4.0 to 7.0 in steps of 0.5 units and the ESR from 1.0 to 7.0 in steps of 0.5 units. After the generation of both datasets (synthetic and real), a rigorous analysis is carried out to verify that they are comparable, at least from the perspective of the neural network.

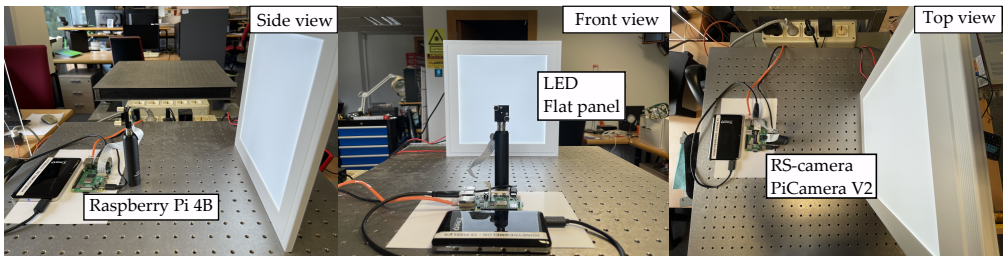

Fig. 5. Experimental setup.

Table 1. Experiment's key parameters.

Parameter	*Value*	Parameter	*Value*
Camera		*Transmitter*	
Hardware	Sony IMX219 [26]	LED color	Cold white
Aperture lens	f/2 |Focal length (3 mm)	Source voltage (V)	30 to 35
Image resolution	1920x1080 pixels	Symbol time, t_{sym} | NPPS	75 μs to 135 μs | 4 to 8
Exposure times, t_{exp}	20 μs to 1500 μs (step 19 μs)	Header, payload, stuffed, and trailer bits	5, 35, 12, 1
Sampling period, T_S	18.9035 μs	Random seed	31415

The following phase is the actual training of the network using the synthetic dataset. The training is conducted offline using a computer with local access to the datasets stored in memory. For training, the input images are z-score standardized, and the outputs are normalized between zero and one using a fixed value. The loss function used for training the network is the Mean Squared Error (MSE). Additionally, to improve the training results (i.e., minimizing the MSE), a network hyperparameter optimization is conducted, following the hyperband algorithm detailed in [27]. The layers of the optimized network model are summarized in Table 2. This table presents from top to bottom the subsequent layers, starting from the input layer, which ingests images, and finalizing with the output layer consisting of two neurons (one per estimate). In each row, it is detailed the layer type (such as DSC, average pooling, max pooling, dense,. . .), the total number of trainable parameters (such as the layer's weights and biases), specific parameters (such as the number of filters, the kernel size,..), and, finally, the layer's output shape. The latter is the

Optics EXPRESS

shape of the output tensor delivered from a particular layer. For instance, the input layer returns a tensor with a shape of 32 images (i.e., corresponding to the training batch size) with a height and width of 256 and 64 pixels, respectively, and 1 colour channel (i.e., the image is in grayscale).

Table 2. Model summary.

Layer	Train params.	Description	Output shape
Input	0	-	(32, 256, 64, 1)
DSC	47	Filters=16, Kernel=(3,5)	(32, 256, 64, 16)
AveragePool	0	Kernel=(1,4), Strides=(1,2)	(32, 256, 32, 16)
MaxPooling	0	Kernel=(2,2), Strides=(1,1)	(32, 128, 16, 16)
DSC	688	Filters=32, Kernel=(3,3)	(32, 128, 16, 32)
AveragePool	0	Kernel=(1,2), Strides=(1,1)	(32, 128, 8, 32)
MaxPool	0	Kernel=(2,1), Strides=(1,1)	(32, 64, 8, 32)
DSC	2400	Filters=64, Kernel=(3,3)	(32, 64, 8, 64)
AveragePool	0	Kernel=(1,2), Strides=(1,1)	(32, 64, 4, 64)
MaxPool	0	Kernel=(2,1), Strides=(1,1)	(32, 32, 4, 64)
DSC	8768	Filters=128, Kernel=(3,3)	(32, 32, 4, 128)
AveragePool	0	Kernel=(1,2), Strides=(1,1)	(32, 32, 2, 128)
MaxPool	0	Kernel=(2,1), Strides=(1,1)	(32, 16, 2, 128)
DSC	17536	Filters=128, Kernel=(3,3)	(32, 16, 2, 128)
AveragePool	0	Kernel=(1,2), Strides(1,1)	(32, 16, 1, 128)
Flatten	0	-	(32, 2048)
Dense	524544	Units=256, bias=True	(32, 256)
Dense	16448	Units=64, bias=True	(32, 64)
Dense	128	Units=2, bias=False	(32, 2)
Total trainable params:		570,559	

The next phase consists of validating the model training using the real dataset (that has not been used during training). The validation metric is the root MSE (RMSE) of the estimates. In addition, based on the results obtained, the training is improved iteratively by refining the synthetic dataset to make it more similar to the real dataset. In this way, it is possible to progressively enhance the network's performance, reducing the validation error. In the last phase, the validation estimates are analyzed and dissected in the NPPS and ESR dimensions to examine possible influences between them. For instance, the ESR conditions may influence the estimation of NPPS and vice versa. Finally, the relative errors (RE) at estimating the transmitter clock and the camera's exposure time is computed.

4. Results

This section starts by introducing the preliminary examination and comparison between the synthetic and the real datasets. Then, the iterative process accomplished to refine the results obtained in the training and validation of the model is presented. At the end of this section, the final results obtained for the best training configuration are dissected and discussed, analyzing the model's precision and accuracy in estimating the NPPS and the ESR.

Figure 6 displays two example sets (left and right) with three different images to provide a visual comparison between the real and synthetic datasets. The first image (in those sets) belongs to the real dataset. It is a grayscale image that takes values between 0 and 255. The second image is a 2D tensor obtained from the z-score standardization of the previous image. It has values

Optics EXPRESS

between -3 and 3 so that its mean and standard deviation are close to 0 and 1, respectively. The third image is the synthetic version (z-score standardized) created with the algorithm detailed in [4]. The left and right sets represent signals with NPPS of 4 and ESRs of 1.1 and 5.6, respectively. These sets are affected by different SNRs: 20 dB (left) and 7 dB (right). It should be remarked that only the z-score standardized tensors will be used as inputs in the training and validation. Therefore, they should be comparatively similar, at least from the perspective of the neural network. At first glance, they might look quite similar in this visualization. However, to give rigour to this analysis, the cross-correlation between both datasets was carried out. The resulting cross-correlation matrix is shown in Fig. 7. Each point of the correlation matrix represents the maximum correlation value obtained for a pair set (synthetic, real). Highlight that for this experiment, the transmitter is configured to send packets with the same payload, and the synthetic versions of these images also contain the same data. The results are distributed in the matrix as follows. The left axis represents the synthetic images, and the upper axis, the real images used for the correlation. The axis labels are sorted based on the pair sets (NPPS, ESR). From left to right and top to bottom, the NPPS value increases. Also, for each NPPS (e.g., 4.0), the ESR is increased until it reaches its maximum value, obtaining the corresponding pairs sets (e.g. ,(4.0, 1.0), (4.0, 1.5),. . . , (4.0, 7.0)).

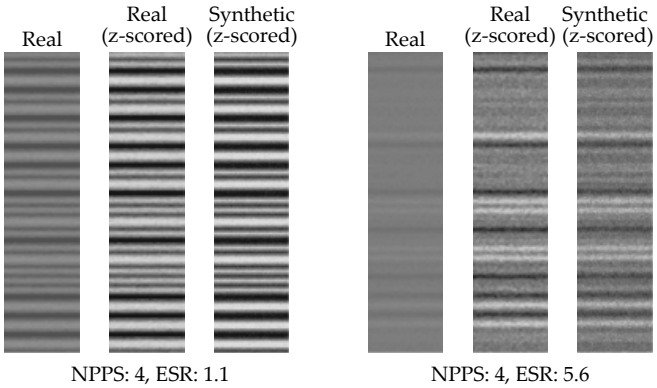

Fig. 6. Comparison between real and synthetic segments after applying the z-score standardization.

This matrix shows that the maximum values are in the diagonal, indicating that the synthetic generated versions match the camera images successfully. On the other hand, it can be appreciated that the correlation values decay softly from the diagonal towards the edges (i.e., the correlation values decrease softly as the ESR slightly increases or decreases). There is an explanation for this, and it is related to the inherent behaviour of the genuine acquisition regarding the exposure time. As the camera's exposure duration is increased (with respect to the original exposure conditions), the obtained images progressively lose their similarity with the initial image. The point at which increasing the exposure time makes two images significantly different can be determined from this matrix. This is the point where the correlation value decreases from 0.95 (high correlation) to 0.65 (low correlation). At this point, the ESR has increased (or decreased) by roughly one unit, as can be observed in Fig. 7(b). Finally, this matrix shows that the correlation output values are low for different NPPS (i.e., the images with different NPPS can be successfully distinguished). From these results, it can be preliminarily concluded that the model would perform worst at estimating the ESR compared to the NPPS.

Once the similarity of the images has been verified, it is important to highlight the benefits of using a synthetic dataset for training rather than the real dataset. The first advantage is reducing the generation time and memory resources. Capturing real images requires building and adapting

Optics EXPRESS

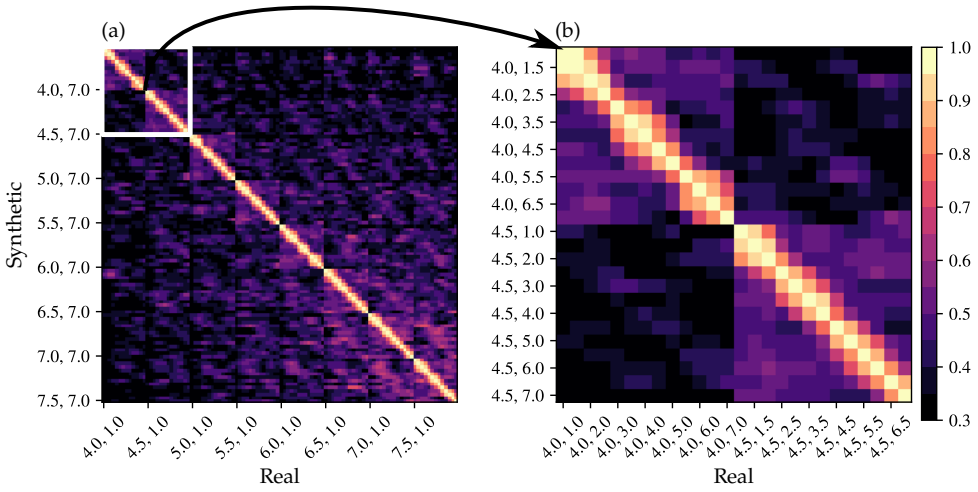

Fig. 7. Cross-correlation matrix of the real and synthetic datasets.

an experimental setup which consumes lots of resources and time. On the other hand, there is no need to rely on cameras to generate synthetic images; hence, they can be made in parallel at high speed. Moreover, specific parameters such as SNR and additional image compression effects are more controllable and rigorous in synthetic images. For example, in real images, to vary the SNR, it is required to precisely adjust the light power of the source, as it is considerably complex to control the noise contributions. In addition, the estimation of the SNR accurately in real images is a remarkably elaborate task. In contrast, different controlled noise contributions can be easily added to synthetic images. Finally, the use of synthetic images allows decoupling the network's training from the camera that will be used in the final deployment. This enables the training for a wide range of generic RS cameras.

Regarding the network's training, Fig. 8(a) shows the training and validation losses per training epoch. The solid blue line represents the validation loss achieved using the best training configuration. The blue dashed line represents, on the other hand, the training loss. Alternatively, the pink lines (solid and dashed) represent the losses (obtained in validation and training) for previous training configurations. These pink lines are used as references only to provide a visual comparison of the improvements obtained in each iteration. As shown in this graph, there is always an irreducible gap between validation and training losses. This is because the synthetic dataset (used for training) and the real dataset (used for validation) are not perfectly similar. However, despite this gap, the validation loss precisely follows the training loss, indicating that the model can optimally generalize the features of the training images without overfitting.

Highlight that the effectiveness of the training depends not only on the network's architecture but also on the design and selection of the appropriate training and validation datasets. Therefore, although the network hyperparameters were optimized using the original datasets, it is possible to further reduce the validation losses by adjusting the training dataset. This iterative dataset refinement procedure is analyzed in Fig. 8(b-f) based on the MSE obtained in the validation. Remark that this metric follows the same evolution as the loss, as the latter is derived from the MSE. However, the loss values are higher because of the regularizing parameters. In these graphs, the validation (solid line) and training (dashed line) MSE curves are coloured in red for the latest best training configuration (reference) and blue after introducing a new change. The pink lines are also maintained as references to other iterations. In the first iteration, the original synthetic generation algorithm [4] is modified by adding a binary quantization of the

Optics EXPRESS

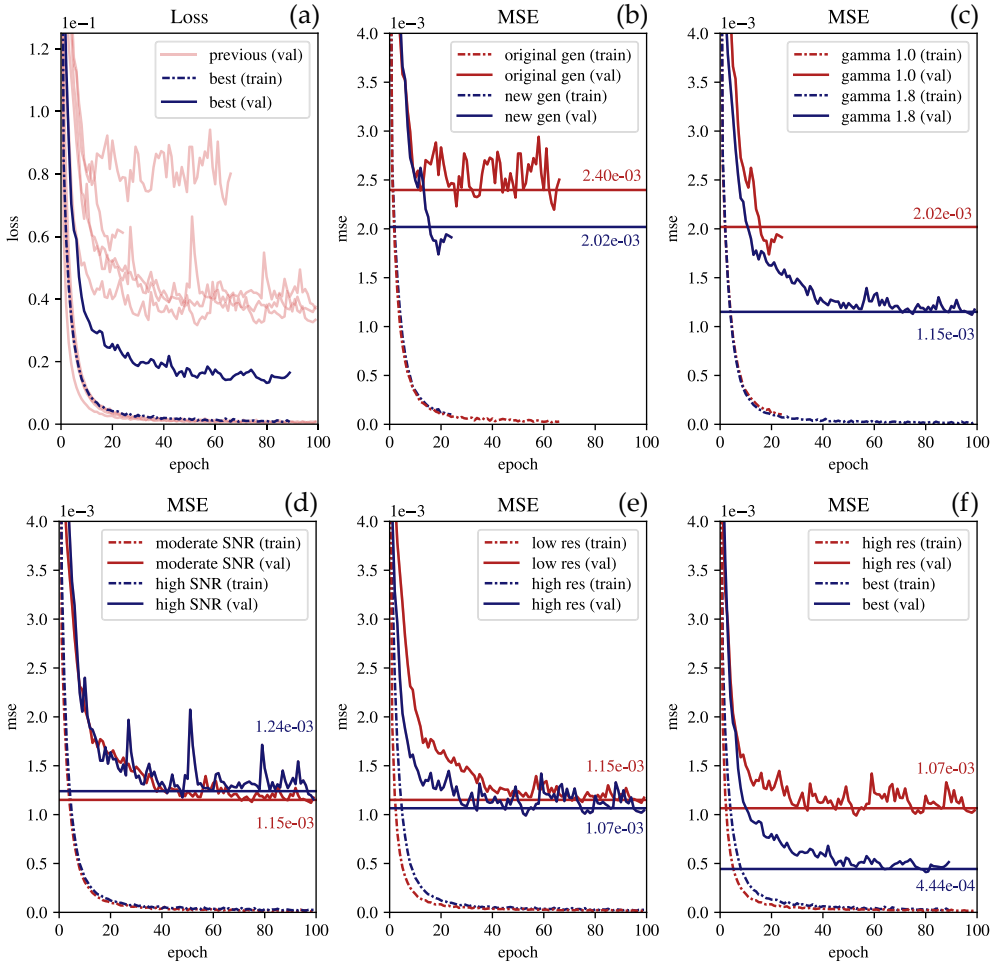

Fig. 8. Training and validation losses and MSE over epochs.

synthetic images, restringing its values to integers between 0 and 255. This step introduces a quantization error that the network can exploit efficiently. Furthermore, it makes synthetic images more representative by removing non-discrete values that can not happen in real images. In addition, a JPEG compression stage follows this binarization with different qualities (from 75 to 90). This introduces some artificial effects observed in real images. As shown in Fig. 8(b), these modifications turn out to be adequate and further reduce the gap between the validation and training MSE. In the second iteration (Fig. 8(c)), a gamma transformation of 1.8 is applied to the samples of the synthetic dataset, reducing the validation MSE considerably. In the third iteration (Fig. 8(d)), the SNR range used to generate the training images is increased. Instead of using an interval of 5 to 30 dB, it was extended from -2 to 40 dB. It can be seen that this approach does not reduce the validation MSE significantly. Furthermore, it has a drawback as it makes the validation unpredictable and unstable. For this reason, the original SNR range was preserved in the following iterations. In the fourth iteration (Fig. 8(e)), the number of images used from training is augmented. The NPPS and ESR's resolution step of 0.5 units is reduced to 0.25 units. The result is a slight improvement in the validation MSE. Based on the previous result, a combined strategy is accomplished in the last iteration (Fig. 8(f)). First, the gamma

Optics EXPRESS

is increased to 2.2 (matching the gamma commonly used in JPEG-encoded images). Second, the training space was generated using random uniform distributed values for the NPPS and the ESR instead of picking them from a grid with a fixed resolution step. This procedure provides a significant improvement in the final validation MSE.

The model and the weights trained in this last iteration will be used in the final evaluation that is presented in the remainder of this section.

Regarding the estimator performance, Fig. 9(a) shows the estimates obtained from the validation dataset. The x-axis represents the NPPS, and the y-axis represents the ESR. The black star markers denote the target points, i.e., the ground truth. The black dots denote the mean of the estimates. Finally, the coloured dots represent the estimates obtained for different images. Dots with the same colour belong to the same target value. The colour does not hold any special meaning. It is only used to ease the identification of different clusters. At first glance, it can be seen that the dots in the lower part of the graph form clusters that are smaller than in the upper part (i.e., where the dots are more spread). This indicates that the model estimations are more precise under short to moderate exposure conditions (i.e., ESR from 1 to 3) than under high conditions (i.e., ESR from 5 to 7). In other words, the estimations under short to moderate exposure conditions have a lower error. Consequently, different errors are obtained depending on the location of the target values in the space domain.

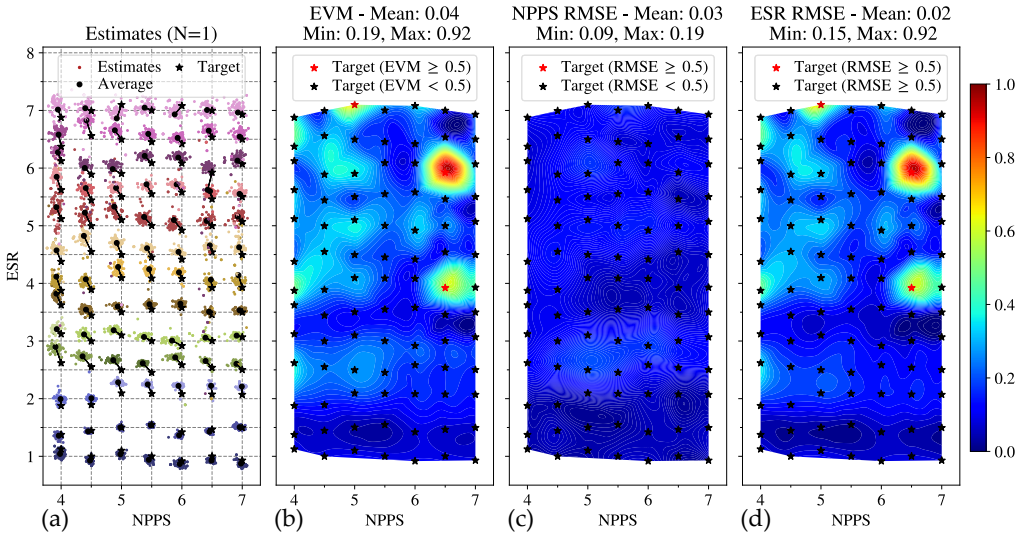

Fig. 9. NPPS and ESR estimates for the validation dataset.

The error vector magnitude (EVM) is used to quantify the errors in the estimations. Figure 9(b) shows the EVM obtained at different points in the space domain. The areas coloured in red represent areas where the estimates have the greatest observed error considering their target values (i.e., EVM equals 1). On the other hand, the areas coloured in blue represent the areas with the lowest estimation errors. As it can be seen, the point (NPPS=6.5, ESR=6) has the highest estimation error, while the point (NPPS=5, ESR=1.5) has the lowest.

However, although this metric quantifies well the estimation error, it does not provide an in-depth analysis of the nature of the error made. Ultimately the EVM metric combines the error obtained in the NPPS and ESR dimensions. Therefore, it is necessary to analyze the NPPS and the ESR errors independently. For this purpose, Fig. 9(c) shows RMSE obtained for the NPPS and Fig. 9(d) the RMSE for the ESR, respectively. In both cases, the RMSE values vary from 0

Optics EXPRESS

(blue) to 1 (red). These graphs reveal that the model can estimate the NPPS more accurately than the ESR. The ESR estimation errors contribute highly to the EVM.

Examining the EVM map in detail, it is observed that the errors are greater as the ESR increases, which is in line with the expected behaviour of the network. The longer the exposure time, the more severe distortion affects the received signal.

On the other hand, the errors also increase with the NPPS. This seems to contradict the expected behaviour, since increasing the redundancy per symbol (higher NPPS) should help the network at estimating the output parameters. However, the opposite seems to be happening. The reason behind this lies in the fact that the dimensions of the network's input image are fixed to 256x64x1 pixels. Hence, the samples of the transmitted signal correspond just to 256 pixel rows. Consequently, increasing the redundancy of symbol samples comes at the cost of reducing the number of different symbols that fit inside the image. In other words, increasing the redundancy reduces the variability of the signal samples. For instance, for NPPS equal to 7, the number of symbols within the image is approximately 36 (36 = 256 / 7), while for NPPS equal to 4, this number is 64. As a consequence, for higher NPPS, the network might deliver estimates that greatly deviate from their target value, estimation outliers. For example, in Fig. 9(a) can be seen a few purple-coloured dots around the coordinates (NPPS=6.5, ESR=3.5) when they should be located near their target point (NPPS=6.5, ESR=6). Nevertheless, these outliers represent a rare case. For this particular target point (NPPS=6.5, ESR=6), only four estimates of 85 fall in a region far distant from their target point. Furthermore, it is experimentally validated that these outliers are generated systematically and predictably. They appear when the transmitted payload bitstream has a considerable number of '01' or '10' bit sequences chained together (e.g. '01010101$\cdots$'). This causes the final transmitted signal, which is Manchester encoded, to be confused with another one generated with half the actual clock frequency. This is a classical problem in traditional clock recovery systems based on Manchester encoded systems, which is solved by inserting packet preambles or by sending pilot clock signals. Definitely, the estimator might be confused when facing those rare cases, which are more likely to appear when the NPPS is higher, due to the fixed size of the input image. These outliers can be avoided by changing the proposed bit stuffing technique to prevent the presence of '01' or '10' bit sequences or by inserting packet preambles.

Alternatively, outlier-resistant methods can be applied to a set of different estimates to effectively eliminate the presence of these outliers, revealing the actual trend of the errors that the network makes in the estimation of the output parameters.

The first strategy consists of computing the mean of a set of N estimates. This set is generated by collecting the estimates delivered from N random input images. The final ESR and NPPS estimates are obtained by computing the mean of the previous set.

Figure 10 shows the estimates obtained for N equal 5. In this figure, it can be observed that the impact of outliers is partially mitigated. The RMSE errors for the NPPS and the ESR are considerably reduced. However, Fig. 10(a) shows that there are still some estimates that spread over the ESR dimension. For example, the dots belonging to the target points (NPPS=6.5, ESR=4.0) and (NPPS=7, ESR=4) are spread within the range of ESR from 4.0 to 4.5. Consequently, with this approach, the outliers still significantly impact the ESR estimation.

On the other hand, given the low probability of the appearance of the outliers, the delivered estimations can be further improved by using the median instead of the mean. Figure 11 shows the estimates obtained for N equal 5. In contrast with the mean, the median produces much better results. The dispersion of the estimates in the ESR domain using the median is significantly reduced. This graph verifies that after eliminating the effect of the outliers, the behaviour of the network corresponds to what is expected. The errors are greater for low NPPS and high ESR.

Finally, Fig. 12 displays the relative error (RE) obtained for the NPPS and the ESR separately using the median approach, with N equals 10. To improve the visualization, the colour scale

Optics EXPRESS

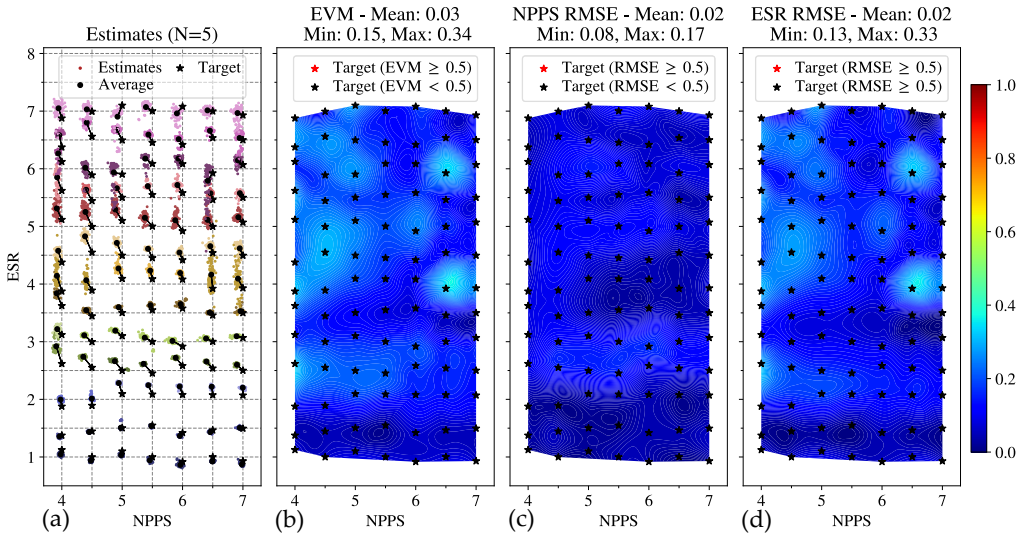

Fig. 10. NPPS and ESR estimates using the average of the outputs of 5 random images.

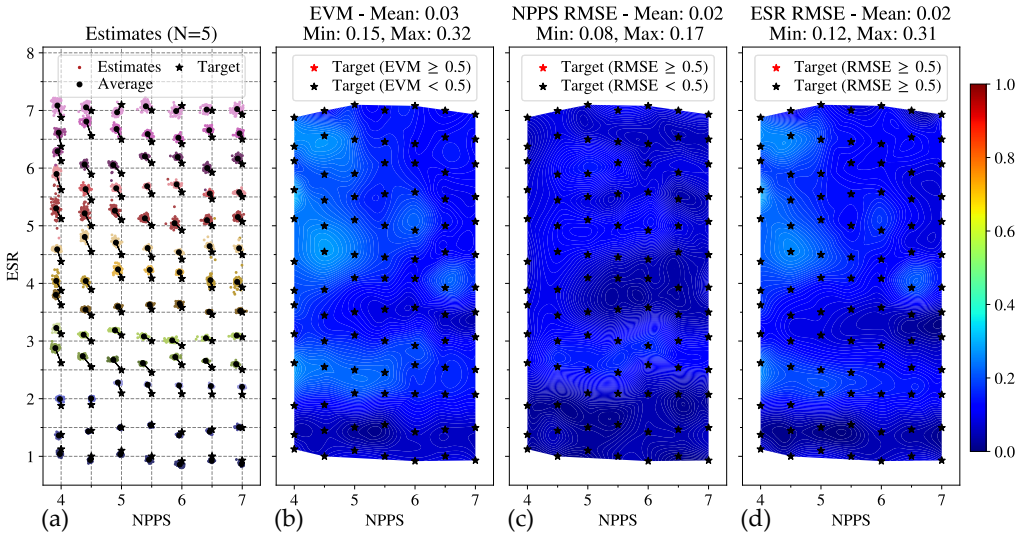

Fig. 11. NPPS and ESR estimates using the median of the outputs of 5 random images.

Optics EXPRESS

ranges from 0 to 0.14. These plots show that the maximum and minimum RE observed for the NPPS are approximately 3% and 0.2%, respectively. In the case of the ESR, they are 9% and 0.6%, respectively. On average, the error for both estimations is approximately 2%. From the communications perspective, these results imply, on the one hand, that the estimation of the signal clock, related to the NPPS, has a minimum and a maximum RE of 0.2% and 3% respectively. Furthermore, if the transmission clock frequency is known, the camera's row sampling time can be characterized by using the relation (NPPS = t_{sym}/T_S). The error made in the estimation of the row sampling time can be computed using the error propagation theory.

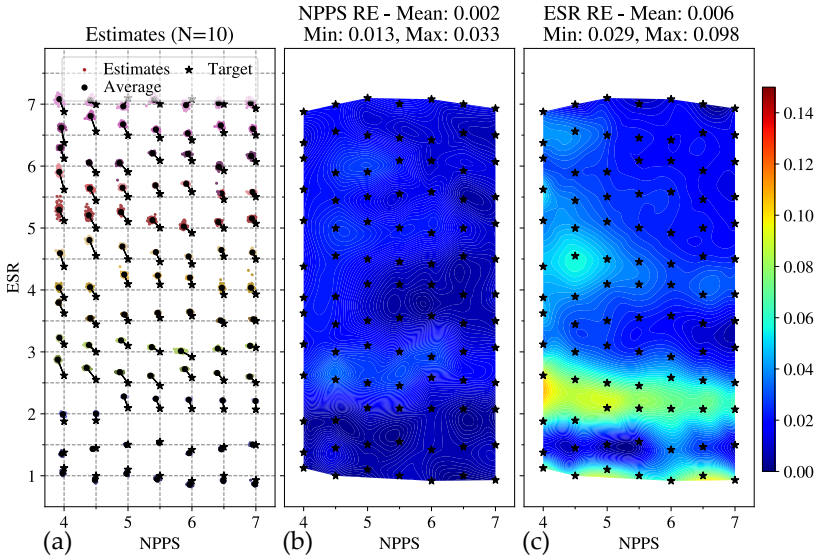

Fig. 12. NPPS and ESR RE using the median of the outputs of 10 random images.

On the other hand, the RE made at estimating the camera's exposure time is upper bounded to 9%. As discussed in the previous work [4], an exposure-equalizer trained for a given exposure time can successfully equalize images exposed with slightly higher or lower exposure times. Those equalizers allow a maximum deviation of up to 11% for the target exposure time while ensuring a Bit Error Rate (BER) lower than the Forward Error Correction (FEC) limit ($3.8x10^{-3}$). Therefore, as the estimator's ESR estimates have an average RE of 2% (upper bounded to 9%), it is ensured that the receiver will select the optimal equalizer based on the delivered ESR.

5. Conclusions

This work presents and evaluates a deep learning-based approach to accurately estimate two fundamental parameters of optical signals acquired with RS-cameras: the NPPS and the ESR. These dimensionless parameters relate the transmitted symbol duration with the camera's sampling frequency and exposure time, respectively. Hence, the NPPS is directly related to the signal clock and the ESR with the camera's exposure settings. These parameters are required during the equalization and decoding stages at reception. Its precise estimation will allow the receiver to select the optimal equalizer to mitigate the exposure-related ISI and recover the clock for synchronization and decoding, even when the transmission frequency and the camera's internal settings are unknown. Therefore it becomes an essential part of RS receivers operating over moderate exposure conditions. In addition, this estimator decouples the reception algorithms from the image stream providers, enabling cloud architectures that can practically handle multiple camera devices (or another type of image streamers). On the other hand, it can be used to

Optics EXPRESS

characterize cameras in the time domain if needed. For example, if the camera does not reveal its internal settings due to operative system constraints. In addition, the network's training using synthetic images covers a vast training space, with thousands of representative cases considering multiple configurations for the NPPS and the ESR and different SNR conditions. Furthermore, the network's validation using a real dataset favoured the introduction and evaluation of incremental training improvements by refining the original datasets. The final evaluation of the model shows that the minimum, mean, and maximum relative errors for the NPPS estimates are 0.2%, 1.3%, and 3%, respectively. This implies an average RE of 1% at determining the transmission frequency of the source. Furthermore, these errors are significantly low when the exposure time is shorter. On the other hand, the minimum, mean and maximum relative errors for the ESR estimates are 0.6%, 3% and 9%, respectively. In this estimation, the neural network produced worse estimates, and still, the REs obtained are consistently lower (i.e. RE < 9%) than the 11% ESR deviation supported by pretrained equalizers. Therefore, it is ensured that the BERs after the equalization will remain below the FEC limit (3.8×10^{-3}) under the stated conditions. It should be highlighted that the RE does not exceed 3% in most cases, which indicates that the system is significantly robust in this estimation, favouring a better performance of the equalizers. In conclusion, this estimator is essential in all RS-based OCC links, in which both the signal clock and the exposure time must be retrieved from the images. Consequently, it enables the design of novel generic OCC links that do not require setting rigid requirements for the transmitter and the camera settings. Instead, those links will cover many different RS camera devices.

Funding. European Cooperation in Science and Technology (NEWFOCUS COST action (Ref: CA19111)); Agencia Estatal de Investigación (Project OCCAM, Ref. PID2020-114561RB-I00).

Disclosures. The authors declare no conflicts of interest.

Data availability. Data underlying the results presented in this paper are available in Dataset 1, [19].

Supplemental document. See Supplement 1 for supporting content.

References

1. "IEEE Standard for Local and metropolitan area networks–Part 15.7: Short-Range Optical Wireless Communications, *IEEE Std 802.15.7-2018 (Revision of IEEE Std 802.15.7-2011)* pp. 1–407 (2019).
2. S. A. H. Mohsan, "Optical camera communications: practical constraints, applications, potential challenges, and future directions," J. Opt. Technol. **88**(12), 729–741 (2021).
3. G. Cossu, A. Sturniolo, and E. Ciaramella, "Modelization and characterization of a cmos camera as an optical real-time oscilloscope," IEEE Photonics J. **12**(6), 1–13 (2020).
4. C. Jurado-Verdu, V. Guerra, V. Matus, J. Rabadan, and R. Perez-Jimenez, "Convolutional autoencoder for exposure effects equalization and noise mitigation in optical camera communication," Opt. Express **29**(15), 22973–22991 (2021).
5. X. Li, N. B. Hassan, A. Burton, Z. Ghassemlooy, S. Zvanovec, and R. Perez-Jimenez, "A simplified model for the rolling shutter based camera in optical camera communications," in *2019 15th International Conference on Telecommunications (ConTEL)*, (IEEE, 2019), pp. 1–5.
6. H. Aoyama and M. Oshima, "Line scan sampling for visible light communication: Theory and practice," in *2015 IEEE International Conference on Communications (ICC)*, (2015), pp. 5060–5065.
7. W. A. Cahyadi, Y. H. Chung, Z. Ghassemlooy, and N. B. Hassan, "Optical camera communications: Principles, modulations, potential and challenges," Electronics **9**(9), 1339 (2020).
8. N. Saeed, S. Guo, K.-H. Park, T. Y. Al-Naffouri, and M.-S. Alouini, "Optical camera communications: Survey, use cases, challenges, and future trends," Phys. Commun. **37**, 100900 (2019).
9. N. T. Le, M. Hossain, and Y. M. Jang, "A survey of design and implementation for optical camera communication," Signal Process. Image Commun. **53**, 95–109 (2017).
10. C.-W. Chow, C.-Y. Chen, and S.-H. Chen, "Visible light communication using mobile-phone camera with data rate higher than frame rate," Opt. Express **23**(20), 26080–26085 (2015).
11. L. Liu, R. Deng, and L.-K. Chen, "47-kbit/s rgb-led-based optical camera communication based on 2d-cnn and xor-based data loss compensation," Opt. Express **27**(23), 33840–33846 (2019).
12. Y.-S. Lin, C.-W. Chow, Y. Liu, Y.-H. Chang, K.-H. Lin, Y.-C. Wang, and Y.-Y. Chen, "Pam4 rolling-shutter demodulation using a pixel-per-symbol labeling neural network for optical camera communications," Opt. Express **29**(20), 31680–31688 (2021).

Optics EXPRESS

13. C.-W. Peng, D.-C. Tsai, Y.-S. Lin, C.-W. Chow, Y. Liu, and C.-H. Yeh, "Long short-term memory neural network to enhance the data rate and performance for rolling shutter camera based visible light communication (vlc)," in *2022 Optical Fiber Communications Conference and Exhibition (OFC)*, (2022), pp. 1–3.
14. O. I. Younus, N. B. Hassan, Z. Ghassemlooy, S. Zvanovec, L. N. Alves, P. A. Haigh, and H. Le Minh, "An artificial neural network equalizer for constant power 4-PAM in optical camera communications," in *2020 12th International Symposium on Communication Systems, Networks and Digital Signal Processing (CSNDSP)*, (IEEE, 2020), pp. 1–6.
15. O. I. Younus, N. B. Hassan, Z. Ghassemlooy, P. A. Haigh, S. Zvanovec, L. N. Alves, and H. Le Minh, "Data rate enhancement in optical camera communications using an artificial neural network equaliser," IEEE Access **8**, 42656–42665 (2020).
16. O. I. Younus, N. B. Hassan, Z. Ghassemlooy, S. Zvanovec, L. N. Alves, and H. Le-Minh, "The utilization of artificial neural network equalizer in optical camera communications," Sensors **21**(8), 2826 (2021).
17. P. Zhang, Q. Wang, Y. Yang, Y. Wang, Y. Sun, W. Xu, J. Luo, and L. Chen, "Enhancing the performance of optical camera communication via accumulative sampling," Opt. Express **29**(12), 19015–19023 (2021).
18. P. Nguyen, N. T. Le, and Y. M. Jang, "Challenges issues for occ based android camera 2 api," in *2017 Ninth International Conference on Ubiquitous and Future Networks (ICUFN)*, (2017), pp. 669–673.
19. C. Jurado-Verdu, V. Guerra, J. Rabadan, and R. Perez-Jimenez, "Effects of the camera's exposure time on rolling shutter based optical camera communication links," figshare (2022), Accessed on: Feb. 10, 2022. [Online], doi: https://doi.org/10.6084/m9.figshare.19153166 (2022).
20. C. Jurado-Verdu, V. Matus, J. Rabadan, V. Guerra, and R. Perez-Jimenez, "Correlation-based receiver for optical camera communications," Opt. Express **27**(14), 19150–19155 (2019).
21. Z. Huang, J. He, K. Yu, and W. Li, "Efficient demodulation scheme based on adaptive clock extraction andmapping-sampling for a mobile occ system," Appl. Opt. **60**(12), 3308–3313 (2021).
22. J. He, K. Yu, Z. Huang, and Z. Chen, "Multi-column matrices selection combined with k-means scheme for mobile occ system with multi-leds," IEEE Photonics Technol. Lett. **33**(12), 623–626 (2021).
23. F. Chollet, "Xception: Deep learning with depthwise separable convolutions," in *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, (2017).
24. A. Dubey and V. Jain, *Comparative Study of Convolution Neural Network's Relu and Leaky-Relu Activation Functions* (2019) pp. 873–880.
25. M. Khalid, J. Baber, M. K. Kasi, M. Bakhtyar, V. Devi, and N. Sheikh, "Empirical evaluation of activation functions in deep convolution neural network for facial expression recognition," in *2020 43rd International Conference on Telecommunications and Signal Processing (TSP)*, (2020), pp. 204–207.
26. Sony Corporation, "IMX219PQH5-C Datasheet," Accessed on: Feb. 10, 2022. [Online]. Available: https://datasheetspdf.com/pdf/1404029/Sony/IMX219PQH5-C/1 (2014).
27. L. Li, K. Jamieson, G. DeSalvo, A. Rostamizadeh, and A. Talwalkar, "Hyperband: A novel bandit-based approach to hyperparameter optimization," The J. Mach. Learn. Res. **18**, 6765–6816 (2017).