



Sub-frame timestamping of a camera network using a coded light signal

Yunhyeok Han ^{a,*}, Stefania Lo Feudo ^a, Gwendal Cumunel ^b, Franck Renaud ^a

^a Quartz Laboratory (EA7393), ISAE - Supméca, 3 rue Fernand Hainaut, Saint-Ouen, 93400, France

^b Lab Navier, Gustave Eiffel University, ENPC, CNRS, F-77455, Marne-la-Vallée, 77455, France

ARTICLE INFO

Keywords:

Camera network
Computer vision
Image processing
Frame timestamp reconstruction
Uncertainty characterization

ABSTRACT

To extract the 3D displacements of structures using a camera network, it is essential to know the frame timestamps. However, technical constraints in outdoor conditions can make synchronous frame acquisition impossible. This study aims to reconstruct the timestamps of frames during post-processing to employ cameras without special camera feature. To achieve this, a LED panel that emits a coded light signal is placed in the field of view common to all cameras. The pixel intensity of the LED region in the sequential images is processed to reconstruct each frame's timestamp. The uncertainty of the timestamp reconstruction is numerically estimated to encode the light signal. Validation was conducted through indoor experiments and outdoor applications to monitor a tower crane. The uncertainty of the obtained timestamp is 1.65% of the frame interval under uncontrolled conditions. The proposed method can be used with cameras, even when synchronization is unattainable.

1. Introduction

A camera network is obtained by positioning several cameras at different locations in a given environment [1] and allows to measure the same event from various angles or in a wider field of view (FOV) [2,3]. This technique has received a lot of attention from the industrial and academic communities [4]. For example, it can be employed to perform multi-view tracking, 3D displacement and deformation measurements by stereo-vision, modal analysis, and 3D reconstruction [5–7]. Also, it is a valuable structural health monitoring method for the safety and security of large civil structures such as highways and bridges [8,9].

Recently, increasing interest is being placed in using low-cost cameras in outdoor environments to study the dynamics of large-scale structures. Indeed, combining commercial cameras, such as smartphones or action cameras, rather than industrial ones can be a reasonable way to get a camera network at a relatively low cost. In order to investigate the dynamic scene by a camera network, synchronization is always mandatory. However, these cameras do not provide specific synchronization features, so synchronous image acquisition can be complex. Nevertheless, the obtained image data must be aligned in time by the process to acquire the timestamp of frames for further analysis, such as 3D displacement measurement and Operational Modal Analysis [10].

Existing methods to know the timestamp of frames and their limitations will be reviewed here. The first approach is synchronous image acquisition. In hardware-based synchronization, the images are acquired synchronously from more cameras through a trigger signal provided by a hardware source [11–13]. As a drawback, additional

hardware is required. This hardware has to be connected to each camera with a line, so all cameras in the network must support the corresponding feature [14]. As another strategy, network-based synchronization can be used. If all cameras support the related feature, synchronous image acquisition can be executed after synchronizing the clocks of each camera via signal exchanges through a network. In other words, the camera clock can be synchronized through protocols such as the Network Time Protocol (NTP) [15,16] and the Precision Time Protocol (PTP) [16,17]. This approach requires every camera to be connected through Ethernet. Therefore there is a limitation not only on the bandwidth [5] but also on the remote or wireless application [18]. More importantly, this approach cannot be used when cameras do not provide the corresponding NTP or PTP feature.

Another solution is to reconstruct the timestamps of the frames in a post-processing stage, in which the times of the asynchronous signals obtained from a set of independent sensors are reconstructed after the data acquisition [19]. Frame matching methods by correlating specific fingerprints in videos have been proposed in [2,20]. However, these methods only match frames and do not consider asynchronous frame acquisition and acquisition time in sub-frames. In addition, several studies have proposed methods for reconstructing timestamps that take into account the asynchronous frame acquisition in the sub-frame. Examples of this include methods based on the motion of silhouettes [21], trajectory matching [22], and the human pose estimation [23]. In these studies, reconstruction of timestamps is performed through the movement of the subject in each scene. The reconstruction of timestamps of

* Corresponding author.

E-mail address: yunhyeok.han@isae-supmeca.fr (Y. Han).

moving cameras is done through baseline matching and tracking [24]. As a drawback, all these studies require recording a subject with large movement in the FOV common to all cameras.

A sub-frame timestamp reconstruction method that can be used wirelessly and without additional camera features consists of placing an independent event in the FOV common to all cameras. In this framework, Zhao and Chen [18] measured the offset time between video records by adding a light source providing a random On/Off light signal. The sub-frame offset between cameras is obtained by using the binary light signal and its filmed binary signal. An average error of 8% of the frame interval is obtained at a frame rate of 200 Hz. Ding and Tao. [25] extracted the timestamps of frames by using LEDs in the form of a matrix that displays different combinations of arrays. However, this method required a LED matrix that covers a large part of the FOV with a theoretical accuracy reported as less than 1 ms. In the study of Kim and Ichikawa [26], the timestamp of frames is achieved by knowing the position of a linearly oscillating lighting spot filmed by each camera. The accuracy obtained is around 2% of the frame interval when the frame rate is set to 1000 Hz. The feasibility of their timestamp reconstruction method has been proven in a large-scale camera network. However, the implementation in an uncontrolled outdoor environment or with different camera parameters is not detailed.

In this paper, we propose a sub-frame timestamp reconstruction method of a camera network using a coded light signal that is applicable to a large-scale scene. The reconstructed timestamp with an identical clock can be used for interpolation followed by 3D displacement triangulation in the case of a camera network filming a large-scale structure. The method is suitable for dynamic and vibration analysis of structures. This is conducted by post-processing the light signal in the sequential frames already acquired. The protocol works with a camera network consisting of a variety of commercial-grade cameras, without any special camera features, and consists in placing a light source with a coded signal in the filmed scene. The fundamental concept underlying the reconstruction of the time stamp with a coded light signal is the utilization of slightly different frequencies for the light signal and the camera frame rate. Furthermore, if the camera exposure is partially overlapped with the light-on time, it helps to achieve higher accuracy. A suitable coded light signal was thus determined by numerical simulation. The accuracy of the proposed method is quantified in a laboratory environment with different camera parameters and in an outdoor environment on a large-scale scene.

The paper is organized as follows. In Section 2, the proposed system and the timestamp reconstruction protocol are presented. Numerical simulations are carried out in Section 3 to determine the parameters of the coded light signal and to investigate the influence of the frame rate and exposure time of the camera. Finally, in Section 4, the validation and accuracy of the proposed method will be discussed through experiments in two environments. First, the process and results of the experiment in a laboratory environment will be presented for different camera settings (frame rate and exposure time) and second, for an outdoor application when the camera network is employed to measure the 3D displacements of a tower crane.

2. Principle of the method for reconstructing the timestamp of a camera network from a coded light signal

A representation of tools required for the method proposed in this study is shown in Fig. 1. The light signal generator is placed in the common FOV of a camera network, where a structure of interest is also observable. As a light signal source, a system composed of a control box and a LED panel is manufactured and is represented in Fig. 1. Inside the control box, a microprocessor NI (National Instruments) MyRio is installed and programmed to send a specific sequence of On/Off signals to the LED panel. A lithium iron phosphate battery (12.8 V–12 000 mAh) is installed in the control box to serve as a power source for the

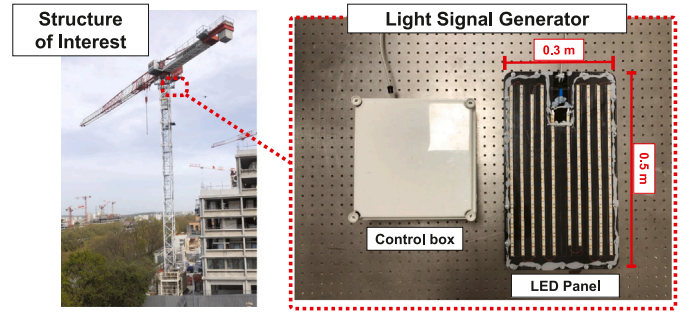


Fig. 1. The structure of interest and light signal generator as a timestamp reconstruction tool. The light signal generator and LED panel are placed in the common FOV of a camera network. In this study, they are located on the top of a tower crane.

LED panel and microprocessor. The LED panel size is $500 \times 300 \text{ mm}^2$ in order to have sufficient brightness to be detected by the cameras and the light signal to be recognized, even in an outdoor environment. The maximum distance can be calculated for this size of LED panel using the thin lens equation, assuming that one side of the LED panel should be at least three pixels. For example, if a camera Basler acA1300-200um with a lens with a focal length of 8 mm is used, the maximum distance can be calculated as 277 m. The maximum distance can be also calculated as 1060 m when a camera Vieworks VC-12MX with 35 mm lens is used. The features of the coded signal will be described in the next sections.

The general concept behind the sub-frame timestamp reconstruction to find the time when the frames were acquired is shown in Fig. 2. The light source emits a light signal as programmed (Fig. 2(a)). Asynchronous video recording of each camera is performed for a defined duration (Fig. 2(b)). A region of interest (ROI) is selected around the LED panel and the mean pixel intensity value is calculated for each frame (Fig. 2(c)). This mean pixel intensity depends on the relationship between the time instants of the acquisition of the frames and those of the light signal flashes. By relating these two pieces of information, the determination of the timestamp of each frame can be conducted (Fig. 2(d)), i.e. the clock of each camera can be synchronized with the clock of the light signal. The details are presented in Section 2.2, and all the parameters that are used in this paper is summarized in Table 1.

2.1. Camera signal

During the acquisition of a frame, the image sensor receives photons for a defined period of time, called the exposure time, and converts the quantity of photons received into gray levels (for monochrome camera) to form an image. Assuming that the camera sensor exposure starts and ends instantaneously and that the acquisition frame rate of sequential images is constant, which is a common assumption when timestamp is reconstructed with events in the FOV [18,21,22], the exposure state of the camera sensor can be expressed as a rectangular pulse signal, as shown in Fig. 3. The start (C_k^{start}) and end (C_k^{end}) states of this rectangular pulse signal can be expressed in the following manner:

$$\begin{aligned} C_k^{\text{start}} &= \{C_1^{\text{start}}, \dots, C_k^{\text{start}}, \dots, C_N^{\text{start}}\} \text{ with } C_k^{\text{start}} = t_0 + (k-1)dt \\ C_k^{\text{end}} &= \{C_1^{\text{end}}, \dots, C_k^{\text{end}}, \dots, C_N^{\text{end}}\} \text{ with } C_k^{\text{end}} = t_0 + t_e + (k-1)dt \end{aligned} \quad (1)$$

where t_0 is the offset time with respect to a reference time, dt is the frame interval which is the inverse of the frame rate (f_{ps}), t_e is the exposure time, and N is the number of sequential images. The reference time is the time instant with respect to which frames of each camera are reconstructed. The k th image is an image obtained by exposing the image sensor from time instants C_k^{start} to C_k^{end} .

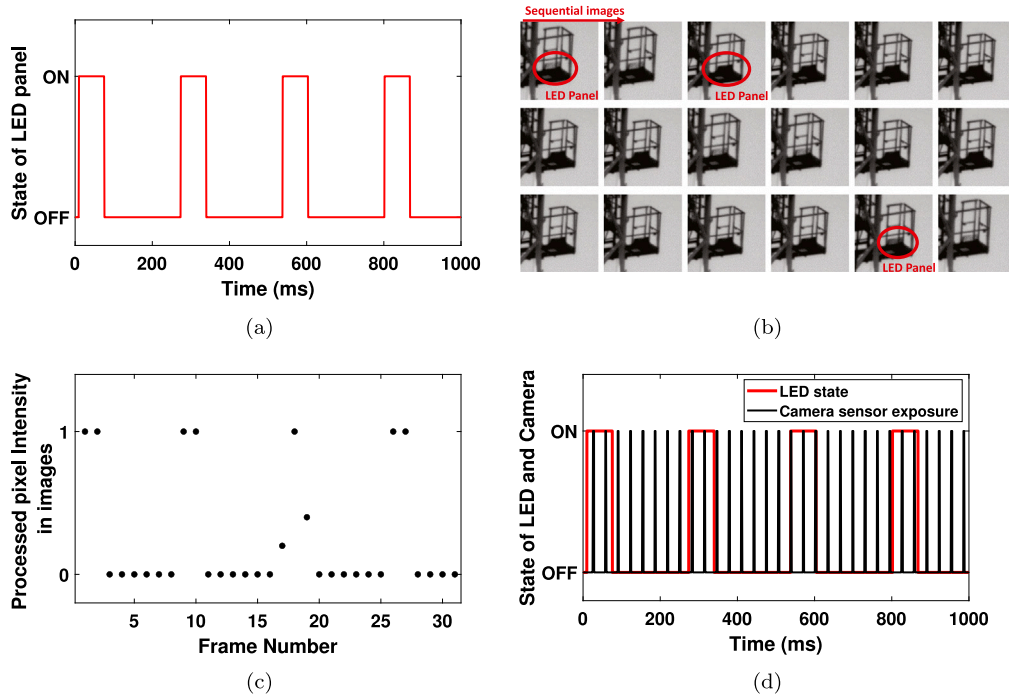


Fig. 2. Steps of timestamp reconstruction of a camera network using a coded light signal. (a) MyRio microprocessor On/Off sequence for the light signal, (b) recorded frames, (c) mean pixel intensity processed in the ROI, (d) reconstructed timestamp with respect to the coded light signal.

Table 1

Description of parameters used.

Parameters	Description
C^{start}	Vector of time when the camera exposure starts. k th component is C_k^{start}
C^{end}	Vector of time when the camera exposure starts. k th component is C_k^{start}
t_0	Offset time with respect to a reference time
dt	Frame interval (inverse of the frame rate)
t_e	Exposure time
L^{start}	Vector of time when flash turns on. l th component is l_i^{start}
L^{end}	Vector of time when flash turns off. l th component is l_i^{end}
S	Vector of the overlapping ratio of the light-on state with respect to the camera sensor exposure time
t_k^{ov}	Overlapping time during which the camera sensor is on and the LED is turned on
I	Vector of mean pixel intensity of the ROI of the LED region
I^{max}	Vector of maximum value of the mean pixel intensity of the LED region
I^{min}	Vector of minimum value of the mean pixel intensity of the LED region
S^{ob}	Vector of observed overlapping ratio S
T_{on}	Duration when the light is on
T_{off}	Duration when the light is off
T_{cycle}	Duration of a cycle of light signal
N_{slot}	Number of slots flashes that can appear in each pattern at most
T_h	Duration of <i>hour pattern</i>
T_m	Duration of <i>minute pattern</i>
T_b	Duration of <i>normal pattern</i>
T_{stop}	Duration when off state maintains after each pattern
H	Current hour
M	Current minute
N_h	Number of flash occurrences corresponding to the current hour
N_m	Number of flash occurrences corresponding to the current minute
C	Time base unit (a coded light signal is coded as a multiple of C)

2.2. Determination of the timestamp of frames

In this section, we will explain how the timestamp of the recorded video can be reconstructed from the measurement of the ROI light intensity obtained through a video acquisition. By assuming that the physical transition of the LED state is instantaneous, the transition times of the light during video acquisition can be expressed as:

$$\begin{aligned} L^{start} &= \{l_1^{start}, l_2^{start}, l_3^{start}, \dots\} \\ L^{end} &= \{l_1^{end}, l_2^{end}, l_3^{end}, \dots\} \end{aligned} \quad (2)$$

which indicates that the i th flash turns on at time instant l_i^{start} and turns off at time instant l_i^{end} . The overlapping ratio of the light-on state with respect to the camera sensor exposure time for the k th frame can be calculated as in Eq. (3).

$$S_k = \frac{t_k^{ov}}{C_k^{end} - C_k^{start}} = \frac{t_k^{ov}}{t_e}, \quad \text{with } S = \{S_k\} \quad (3)$$

where the overlapping time t_k^{ov} indicates the duration for which both the camera sensor state signal, for the k th image, and the LED state signal, for the i th flash, are On. This is graphically represented in

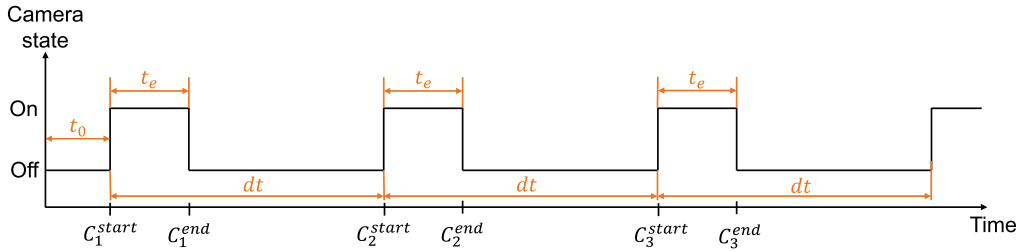


Fig. 3. Camera image acquisition signal represented as a rectangular pulse signal. When the camera state is On, the camera sensor exposure starts and the sensor receives photons. When the camera state is Off, the image sensor does not receive photons.

Table 2

Cases of overlapping ratio between the camera and the light state signals for the k th frame. The mean pixel intensity in the ROI around the LED panel is maximum for fully overlapping On-state signals, and minimum when they do not overlap.

Signal	Fully overlapping case	Partially overlapping case	Non-overlapping case
S_k	$S_k = 1$	$0 < S_k < 1$	$S_k = 0$
Signal visualization			
Image example			

Table 2. This overlapping time t_k^{ov} can be calculated with the time instants C_k^{start} and C_k^{end} and the corresponding light-on time instants l_i^{start} and l_i^{end} (Eq. (4)).

Three cases of overlapping occur, as presented in Table 2. First, there is the fully overlapping case, where the light is always on during the camera frame acquisition. The overlapping time t_k^{ov} can be calculated with Eq. (4a). Second, there is a partially overlapping case in which the state of light changes while the acquisition of an image is in progress. In this case, Eq. (4b) and (4c) can be used to obtain the overlapping time. Last, the non-overlapping case occurs when the light is always off during the camera frame acquisition. Since there is no overlapping between the two On-state signals, the overlapping time is equal to zero as written in Eq. (4d).

$$t_k^{ov} = t_e \quad \text{if} \quad l_i^{start} < C_k^{start} < C_k^{end} < l_i^{end} \quad (4a)$$

$$t_k^{ov} = l_i^{end} - C_k^{start} \quad \text{if} \quad l_i^{start} < C_k^{start} < l_i^{end} < C_k^{end} \quad (4b)$$

$$t_k^{ov} = C_k^{end} - l_i^{start} \quad \text{if} \quad C_k^{start} < l_i^{start} < C_k^{end} < l_i^{end} \quad (4c)$$

$$t_k^{ov} = 0 \quad \text{if} \quad C_k^{start} < C_k^{end} < l_i^{start} \quad \text{or} \quad l_i^{end} < C_k^{start} < C_k^{end} \quad (4d)$$

As a result, given \mathbf{L}^{start} and \mathbf{L}^{end} , \mathbf{S} is the output of a function f , which inputs are the parameters needed to determine C^{start} and C^{end} :

$$f: \mathbf{X} \mapsto \mathbf{S} \quad (5)$$

where $\mathbf{X} = \{t_0, t_e, dt\}$, $\mathbf{S} \in \mathbb{R}^N$, and $0 \leq S_k \leq 1$.

Fig. 4 shows the overlapping ratio for a given sequence of camera and light state signals.

In addition to the assumption of instantaneous physical change of camera sensor and light states, we assumed that the relationship between exposure time and pixel intensity is linear before the pixel saturation occurs. This assumption can be reliable when γ -correction

is switched off [27]. Thanks to this assumption, the ratio S_k can be calculated from the mean pixel intensities measured in the ROI placed around the LED panel. The observed overlapping ratio S_k^{ob} can then be calculated by:

$$S_k^{ob} = \frac{I_k - \min_{[k-n, k+n]}(I_k)}{\max_{[k-n, k+n]}(I_k) - \min_{[k-n, k+n]}(I_k)} = \frac{I_k - I_k^{\min}}{I_k^{\max} - I_k^{\min}} \quad (6)$$

with $\mathbf{S}^{ob} = \{S_k^{ob}\}$, $\mathbf{I} = \{I_k\}$, $\mathbf{I}^{\max} = \{I_k^{\max}\}$, and $\mathbf{I}^{\min} = \{I_k^{\min}\}$. I_k is the mean pixel intensity of the ROI in the k th frame. I_k^{\max} is the maximum value of the mean pixel intensity in the neighborhood of the k th image. The number of neighborhood frames considered, n , has to be large enough to enclose the fully and non-overlapping cases. Likewise, I_k^{\min} is the minimum value of the mean pixel intensity in the neighborhood of the k th image. Indeed, during a video sequence, the extreme values of the mean pixel intensity may be time-dependent. For example, variations may occur outdoors due to changes in lighting, light reflections, and vibration of structures.

Based on the assumptions of the linear relationship between the exposure time and pixel intensity, and the instantaneous physical transition of camera sensor and LED states, we can consider that \mathbf{S} is equal to \mathbf{S}^{ob} . As a consequence, the timestamp of each frame can be obtained by finding the optimal \mathbf{X} that minimizes the root-mean-square error (RMSE) between \mathbf{S} and \mathbf{S}^{ob} according to:

$$\arg \min_{\mathbf{X}=(t_0, t_e, dt)} \sqrt{\sum_{k=1}^N \frac{(S_k(\mathbf{X}) - S_k^{ob})^2}{N}} \quad (7)$$

As a result, reconstructed timestamps of a video sequence (C^{start} and C^{end}) with respect to the reference time can be obtained by the determination of optimal \mathbf{X} . In this way, we can find t_0 , as well as t_e and

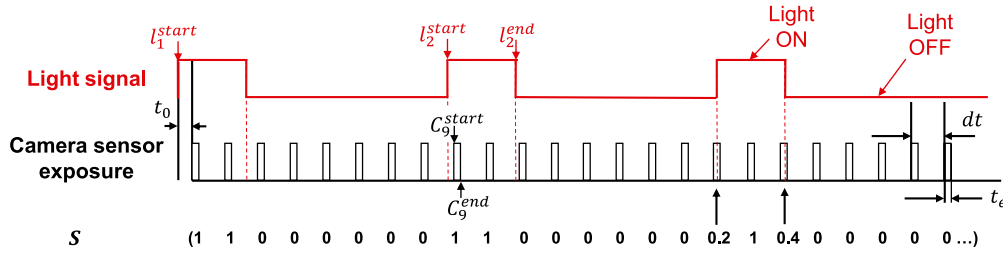


Fig. 4. Overlapping ratio S between camera and light state signals determined by t_0 , t_e , and dt . In this case, t_0 is the offset time with respect to the reference time l_1^{start} .

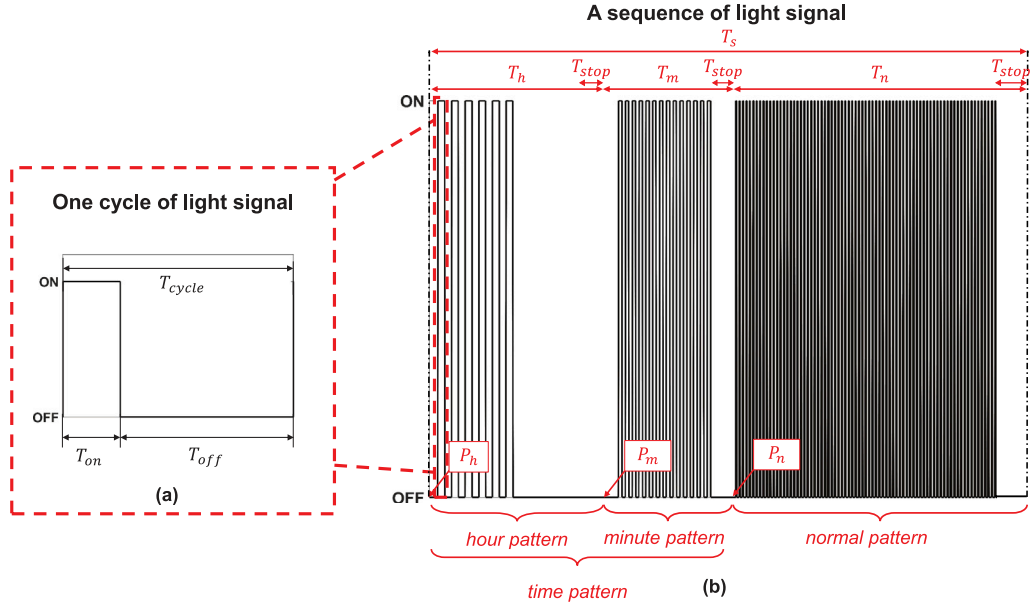


Fig. 5. One period of light state signal (a) and a sequence of the coded light state signal (b). The sequence repeats periodically. At the start of a sequence, the current time is indicated through the *time pattern* and then the *normal pattern* is executed until the next sequence starts.

dt , which are close to the camera settings, but with an uncertainty that will depend on different parameters that will be studied in Section 3.

2.3. Structure of the coded light state signal

In order to reconstruct the timestamps of the On-state signal of the camera sensor with the aforementioned method, the light-state signal has to be encoded in such a way as to ensure a certain level of accuracy and robustness with respect to the camera settings, in particular frame rate and exposure time. Therefore, we propose to encode the light state signal as the repetition of a sequence that can be tunable depending on the camera settings.

A flash of light can be represented as a cycle of a rectangular pulse signal, where T_{on} and T_{off} are the durations of the On and Off states, respectively (see Fig. 5(a)). In one sequence, three pulse patterns are coded to be distinguished in the videos by specific cycle duration and number of occurrences, as shown in Fig. 5(b): one pattern to give the hour information by the number of occurrences of flashes as a 12-hour clock, another to give the minute information by the number of occurrences as the ordinal number of the current sequence in 60 min, and the other for the determination of timestamp with sub-frame resolution. Each pattern ends by maintaining the Off state for a duration T_{stop} .

The first pattern of the programmed LED state signal is the *hour pattern*, starting from the point P_h and having a duration T_h . The second one is the *minute pattern*, starting from the point P_m and having a duration T_m . The *time pattern* consists of these first two patterns, starting from the point P_h and having a duration $T_h + T_m$. The last pattern,

called here the *normal pattern*, starts from point P_n and has a duration T_n . The whole sequence repeats periodically with the period T_s . At the beginning of each sequence, the *time pattern* is updated and indicates the current time. The following steps are used to determine the coded light state signal enabling the determination of the timestamp of the images obtained by the cameras.

1. Select the camera parameters (dt from the camera frame rate, exposure time t_e , and video acquisition time) depending on the objective of the camera network.
2. Decide the duration of a sequence (in minutes) which is shorter than the video acquisition time and is a divisor of 60 min.
3. Select T_{on} and T_{cycle} in the *normal pattern*, based on the numerical simulations presented in Section 3.
4. Decide on the specific duration of each pattern.

Since the purpose of our camera network is to study the dynamics of large structures, for which the frequency range of interest is below 2 Hz, dt is set to 32 ms (i.e. $f_{ps} = 31.25$ Hz) and the video acquisition time to more than 4 min. Therefore, the duration of a sequence T_s is set to 4 min to always indicate the current time in the acquired videos. The next step is to find the specific duration of the *normal pattern* based on the numerical simulations presented in Section 3 for the selected camera parameters.

3. Determination of the parameters of the coded light state signal

In order to illustrate the proposed method and to determine the parameters of the coded light state signal, we checked whether X

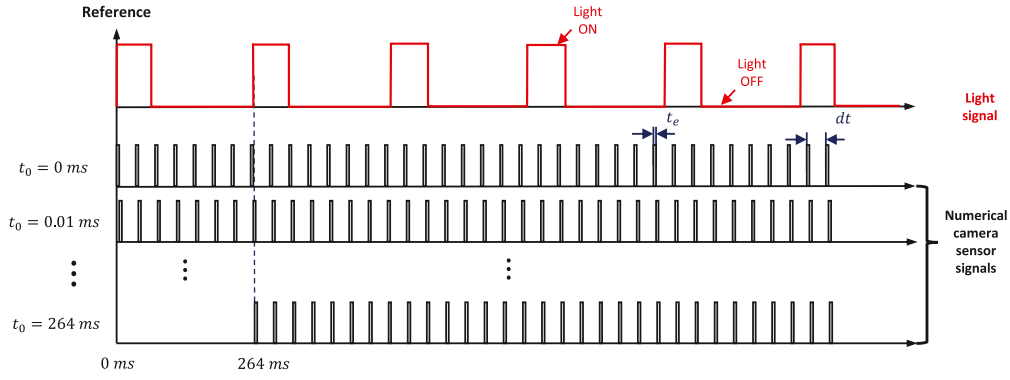


Fig. 6. Simulation of numerical camera sensor signals to calculate the overlapping ratio vector \mathbf{S} for each signal. With fixed exposure time t_e and frame acquisition interval dt , the vector \mathbf{S} is calculated for t_0 from 0 to 264 ms with a step of 0.01 ms.

(= $\{t_0, t_e, dt\}$) can be specified by \mathbf{S} calculated for a synthetic *normal pattern* ($\mathbf{L}^{\text{start}}$ and \mathbf{L}^{end}). For these simulations, the exposure time t_e and frame acquisition interval dt are fixed so that the unknown vector \mathbf{X} is reduced to the offset time t_0 . For this case, the situation where \mathbf{X} and \mathbf{S} have a one-to-one correspondence is investigated. Otherwise, a range of \mathbf{X} (equivalent to t_0 here) that yields the same vector \mathbf{S} is identified and this range is defined as the offset time uncertainty.

3.1. Quantification of the offset time uncertainty

For these simulations, the exposure time t_e is set to 0.5 ms and the frame acquisition interval dt to 32 ms. The *normal pattern* is arbitrarily defined by the duration of the light-on and light-off states: $T_{\text{on}} = 66$ ms and $T_{\text{off}} = 198$ ms (or $T_{\text{cycle}} = 264$ ms). These durations are chosen as multiples of a time base unit C , defined in Section 3.4, needed to encode the light state signal used by the MyRio microprocessor. The offset time uncertainty is here calculated and studied only with the *normal pattern*. To quantify the offset time uncertainty, the overlapping ratio vector \mathbf{S} is calculated and investigated for t_0 varying from 0 to 264 ms with a step of 0.01 ms. The reference time for the offset time t_0 corresponds to the first light-on time instant (t_1^{start}), as shown in Fig. 6.

Each overlapping ratio S_i is taken as an idealized observed overlapping ratio S_i^{ob} and called S_i^{real} , where i corresponds to the index of the step of the offset time t_0 (see Fig. 6), called t_0^{real} . Then, a RMSE vector whose components are the RMSE between the overlapping ratios, S_j , and the one taken as a reference, S_i^{real} , is calculated for each step (t_0^{real}) of the offset time t_0 . The RMSE vectors obtained for $t_0^{\text{real}} = 89.6, 105, \text{ and } 132$ ms (corresponding respectively to $i = 8961, 10501, \text{ and } 13201$) are depicted in Fig. 7. Assuming that t_0^{real} is the real offset time and S_i^{real} is the idealized overlapping ratio S_i^{ob} , the uncertainty of the offset time t_0 is defined as the range of t_0 where the RMSE between S_j and S_i^{real} is equal to 0, i.e. S_j is equal to S_i^{real} (see Eq (7)). In this way, the uncertainty of the estimation of the offset time t_0 is obtained. Fig. 7 shows the uncertainties obtained for $t_0^{\text{real}} = 89.6, 105, \text{ and } 132$ ms.

In the case where t_0^{real} is equal to 132 ms, for example, the offset time uncertainty obtained is 5.5 ms. The evolution of the offset time uncertainty determined for all t_0^{real} is shown in Fig. 8.

The evolution of the offset time uncertainty presented in Fig. 8 is periodic. At worst, the sub-frame accuracy is therefore equal to 5.5 ms when the frame acquisition interval is 32 ms. Only with the *normal pattern*, the proposed method enables an image timestamp 5.8 times more accurate than the frame acquisition interval. Better timestamp accuracy can be achieved by using also the time pattern, as discussed in Section 4.

3.2. Influence of the durations T_{on} and T_{cycle} on the offset time uncertainty

In order to determine the parameters T_{on} and T_{cycle} of the *normal pattern* of the light state signal, their influence on the uncertainty of

the offset time, defined in Section 3.1, is investigated in this section and their values for the coded light signal are decided accordingly. The range studied for these parameters is [33; 100] ms for T_{on} and [100; 300] ms for T_{cycle} .

In Fig. 9(a), the maximum uncertainty of the offset time t_0 is plotted with respect to the durations of the light state signal T_{on} and T_{cycle} , where the x-axis is the duration of the light state cycle T_{cycle} and the y-axis is the light-on state duration T_{on} . For this calculation, the camera acquisition parameters are set to $dt = 32$ ms for the frame acquisition interval, $t_e = 0.5$ ms for the exposure time, and $N = 1000$ images for the number of frames. In order to be able to observe the change of state of the light signal through the images acquired, it is necessary to have at least one non-overlapping case when the state of the light signal is Off and one fully overlapping case when the state of the light signal is On (see Table 2).

It can be seen from Fig. 9(b) that when T_{cycle} is a multiple of dt , this corresponds to the peaks of the maximum uncertainty curve. In Fig. 9(c), the values for which T_{on} is a multiple of dt also yield a high level of the maximum uncertainty, even though the influence of T_{on} is smaller than that of T_{cycle} . These results indicate that the situation where T_{cycle} and T_{on} are multiples of dt should be avoided. As a time base unit has to be defined to encode the light state signal used by the MyRio microprocessor and multiples of dt should be avoided, the time base unit C is arbitrarily set to 33 ms since dt is 32 ms for the camera ($f_{\text{ps}} = 31.25$ Hz). Thus, based on the numerical calculations and since the durations have to be multiples of the time base unit C , we decided to set T_{on} to 66 ms ($= 2C$) and T_{cycle} to 264 ms ($= 8C$) for the *normal pattern*. For these values, the maximum uncertainty of the offset time is 5.5 ms (see Fig. 9(c)).

3.3. Sensitivity of the offset time uncertainty to camera frame rate and exposure time

The influence of the camera frame rate and exposure time on the offset time uncertainty is investigated here to check the adaptability of the proposed method to determine a coded light signal for other values of the camera settings and to study the effect of a change of the camera settings, especially the exposure time, on the offset time uncertainty. The range studied for these parameters is [30; 125] Hz for f_{ps} and [0; 5] ms for t_e . For this calculation, the following parameters are set: the number of frames $N = 1000$ images, the light-on state duration $T_{\text{on}} = 66$ ms, and the duration of the light state cycle $T_{\text{cycle}} = 264$ ms. In Fig. 10(a), the maximum uncertainty of the offset time t_0 is plotted with respect to the camera frame rate (x-axis) and the exposure time (y-axis). The effect of the camera frame rate is presented in Fig. 10(b) for $t_e = 0.5$ ms and Fig. 10(c) investigates how the exposure time affects the maximum uncertainty of t_0 for fixed camera frame rates ($f_{\text{ps}} = 30, 31.25, 60, \text{ and } 62.5$ Hz).

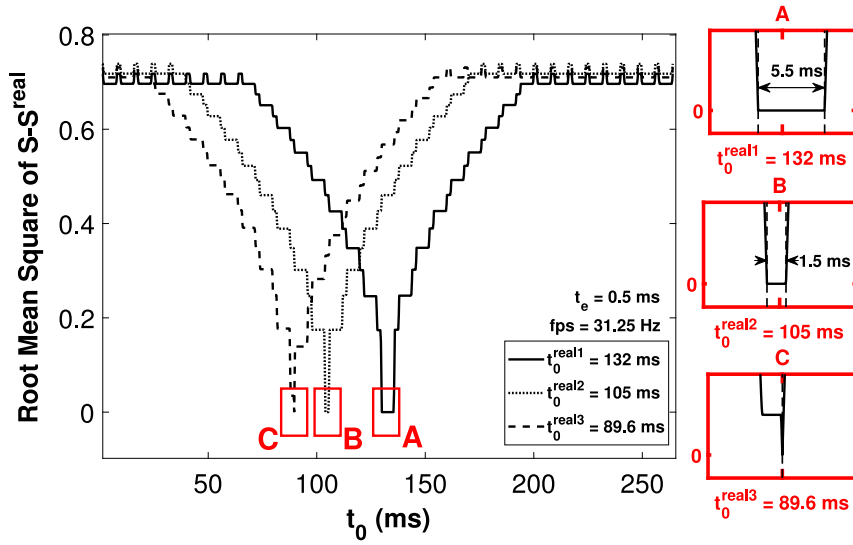


Fig. 7. Quantification of the offset time uncertainty (with $t_e = 0.5$ ms and $dt = 32$ ms) as the range of t_0 for which S_j is equal to S_j^{real} , i.e. the RMSE between S_j and S_j^{real} is equal to zero. In the red rectangles, the offset time uncertainty regions are zoomed and their ranges are visualized.

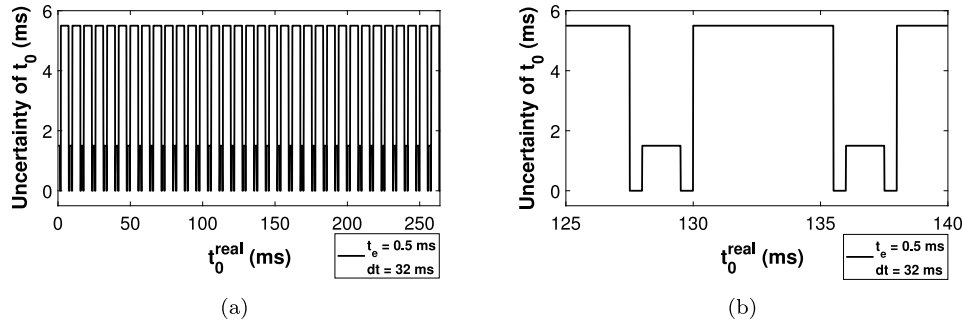


Fig. 8. Evolution of the offset time uncertainty with respect to t_0^{real} when $t_e = 0.5$ ms and $dt = 32$ ms (or $fps = 31.25$ Hz). In (a), the uncertainty of the offset time t_0 is calculated for every t_0^{real} from 0 to 264 ms. In (b), the uncertainty is depicted for t_0^{real} from 125 to 140 ms.

The graph of Fig. 10(b) presents several peaks. This occurs when the least common multiple of T_{cycle} and dt ($= 1/fps$) is too small. For example, when $fps = 30.3$ Hz, dt is approximately 33 ms, which is a divisor of $T_{\text{cycle}} = 264$ ms. As a result, the coded light state signal can be used with different frame acquisition intervals if those are selected based on the numerical calculations. In Fig. 10(c), the maximum uncertainty linearly decreases when t_e increases. This can be explained by the fact that, when t_e increases, the number of occurrences of the fully or non-overlapping cases is reduced and that of the partially overlapping case increases leading to the one-to-one correspondence between X and S .

3.4. Characteristics of the coded light state signal

Since the duration of a sequence of the coded light state signal is fixed at 4 min, the *time pattern* shown in Fig. 5 is coded as follows. During each pattern, at most N_{slot} flashes can appear. For example, during the *hour pattern*, at most 12 flashes can appear, where each corresponding to an hour. For the *minute pattern*, at most 15 flashes can appear, one every 4 min. Each pattern ends by maintaining the Off state for a duration T_{stop} . Thus, the total duration T_i of each pattern, where $i = h, m, n$ respectively for hour, minute, and normal, can be expressed by the Eq. (8).

$$T_i = N_{\text{slot}} T_{\text{cycle}} + T_{\text{stop}} \quad (8)$$

where $T_{\text{cycle}} = T_{\text{on}} + T_{\text{off}}$.

The current time is indicated in the *time pattern* through the number of flash occurrences. In the *hour pattern*, the number of flashes is set to be equal to the current hour (H) according to Eq. (9). For example, a On/Off LED state cycle occurs nine times at 9 am, whereas it occurs three times at 3 pm. Likewise, in the *minute pattern*, the number of flashes is set to be determined by the current minute (M) based on Eq. (10) and is updated every 4 min, according to the sequence period.

$$N_h = \begin{cases} H & \text{if am} \\ H - 12 & \text{if pm} \end{cases} \quad (9)$$

$$N_m = \begin{cases} E\left(\frac{M}{4}\right) + 1 & \text{for } M < 60 \\ 1 & \text{for } M = 60 \end{cases} \quad (10)$$

where N_h and N_m are the number of flash occurrences corresponding to the current hour H and minute M , respectively. $E()$ is the integer part of division. As an example, Fig. 5 indicates that the current time is 6:52.

The specific duration of each pattern for the experimental study presented in Section 4 is detailed in Table 3. The duration of the light signal during each pattern is programmed here to be proportional to the time base unit C . For the specific durations T_{on} and T_{off} of the *hour pattern* and *minute pattern*, we set them arbitrarily to 16 and 8 times the time base unit C to make it easier to distinguish each pattern by the duration of the light state signal.

Thanks to this coded light state signal, the frames recorded from different cameras can be matched using the unique *time pattern* that appears every T_s . As detailed in Section 3.1, a timestamp with a sub-frame accuracy can be obtained with the *normal pattern*.

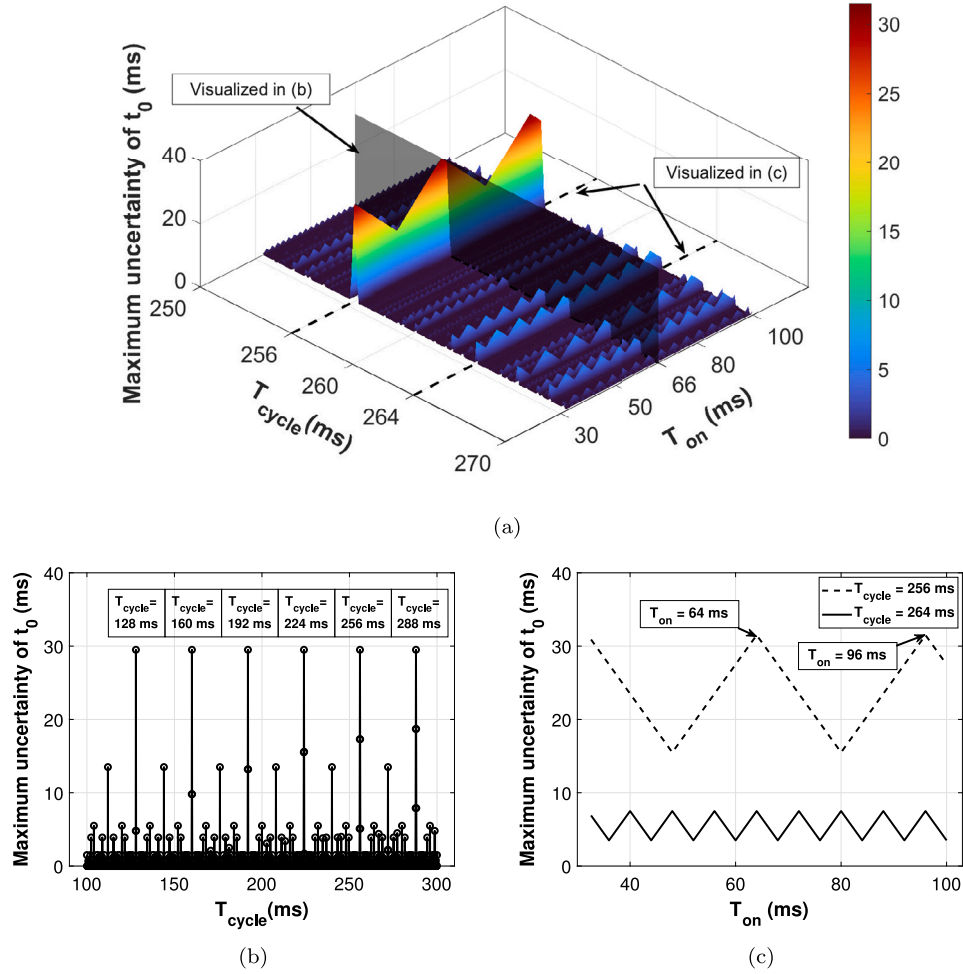


Fig. 9. Influence of the coded light signal parameters on the offset time uncertainty: (a) the maximum uncertainty of t_0 is investigated for different light state signals when camera parameters are $dt = 32$ ms, $t_e = 0.5$ ms, and $N = 1000$ images. The maximum uncertainty is plotted with respect to the duration of the light state cycle T_{cycle} (x-axis) and the duration of the light-on state T_{on} (y-axis). The maximum uncertainty is calculated every 0.1 ms along x- and y-axis. In (b), the influence of T_{cycle} on the maximum uncertainty is shown when $T_{on} = 66$ ms. In (c), the relationship between the maximum uncertainty and T_{on} is shown when T_{cycle} is set to 256 and 264 ms.

Table 3

Detailed duration of each pattern in one sequence of the coded light state signal. $C = 33$ ms is a time base unit of the light state signal. The sequence period is $T_s = 4$ min.

Pattern	T_{on}	T_{off}	T_{cycle}	N_{slot}	T_{stop}	T_i
Hour	16 C	16 C	32 C	12	40 C	12 $T_{cycle} + 40 C = 424 C = T_h$
Minute	8 C	8 C	16 C	15	35 C	15 $T_{cycle} + 35 C = 275 C = T_m$
Normal	2 C	6 C	8 C	818	30 $C - 9$	818 $T_{cycle} + 30 C - 9 = 6574 C - 9 = T_n$

4. Experimental validation

4.1. Test case 1. Sub-frame timestamp reconstruction of a camera network for a laboratory experiment

In order to confirm the applicability of the proposed method and to estimate the accuracy of the timestamp reconstruction with the coded light signal previously defined, an experiment is conducted in a controlled indoor environment.

The system is configured as shown in Fig. 11. The LED panel is placed in an environment without ambient light and filmed by two cameras Vieworks (Model VC-12MX). Both cameras are synchronized through a hardware trigger to enable the validation of the timestamps reconstructed by the proposed method. A parametric study is performed for the frame rate and exposure time of the cameras. The frame rate of the cameras is successively set to 31.25, 62.5, 125, and 250 Hz, and the exposure time is set to 0.5, 1, and 3 ms for each frame rate. In this way,

a total of 12 experiments are first conducted. Videos with a duration of 256 s are acquired synchronously by the two cameras.

To summarize, the timestamp reconstruction of the cameras by the proposed method is performed according to the following steps:

1. For each frame, the mean pixel intensity I in the ROI containing the LED panel is extracted;
2. I^{\max} and I^{\min} are evaluated as the upper and lower envelopes of the intensity at each frame;
3. S^{ob} is obtained from Eq. (6);
4. $X = (t_0, t_e, dt)$ is obtained by minimizing the RMSE between S^{ob} and S . The minimization of Eq. (7) is performed in MATLAB[®] using the *fmincon* function;
5. The timestamp of each camera, C^{start} , is reconstructed with the optimized vector X_{opti} from Eq. (1).

The signal processing and optimization process of Camera 1 are shown in Figs. 12(a) and 12(b) for $t_e = 1$ ms and $f_{ps} = 125$ Hz, and in

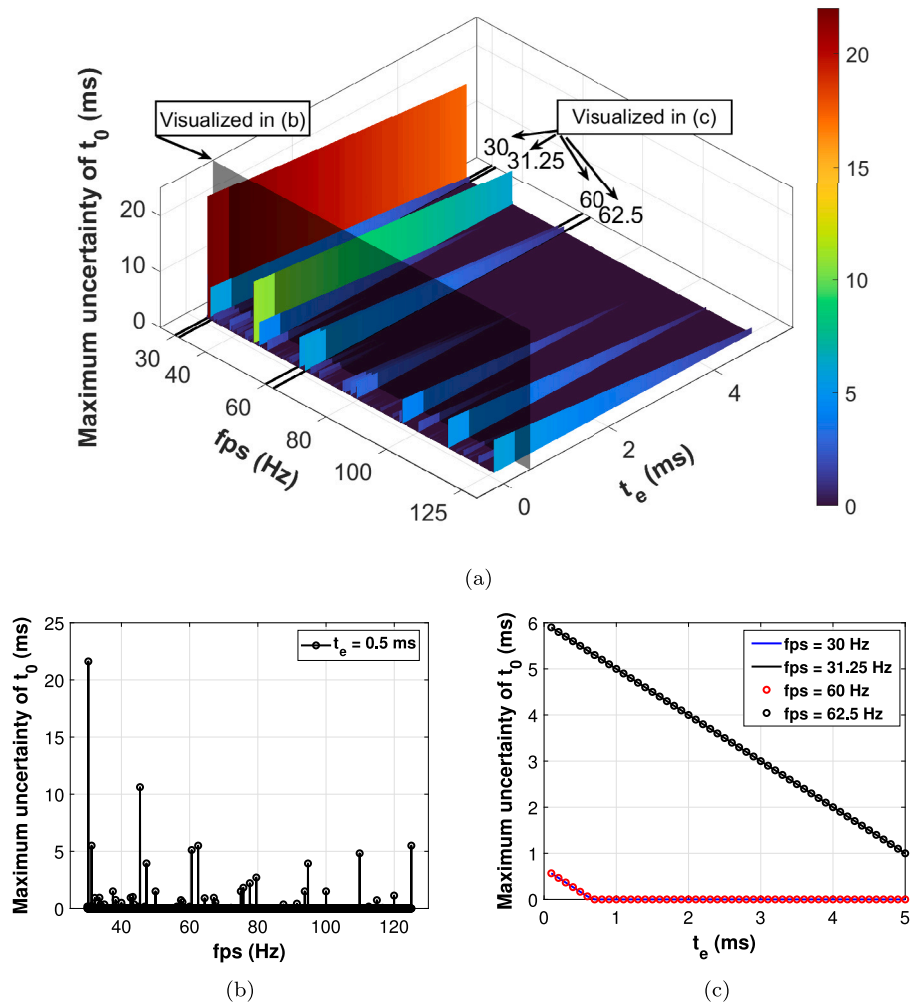


Fig. 10. (a) The maximum uncertainty of the offset time t_0 is plotted with respect to the camera frame rate (x-axis) and exposure time (y-axis). The maximum uncertainty is calculated every 0.05 Hz along x-axis and 0.05 ms along y-axis. (b) Influence of the camera frame rate on the maximum uncertainty when the exposure time is fixed to 0.5 ms. (c) Maximum uncertainty with respect to the camera exposure time when the camera frame rate is set to 30, 31.25, 60, and 62.5 Hz.

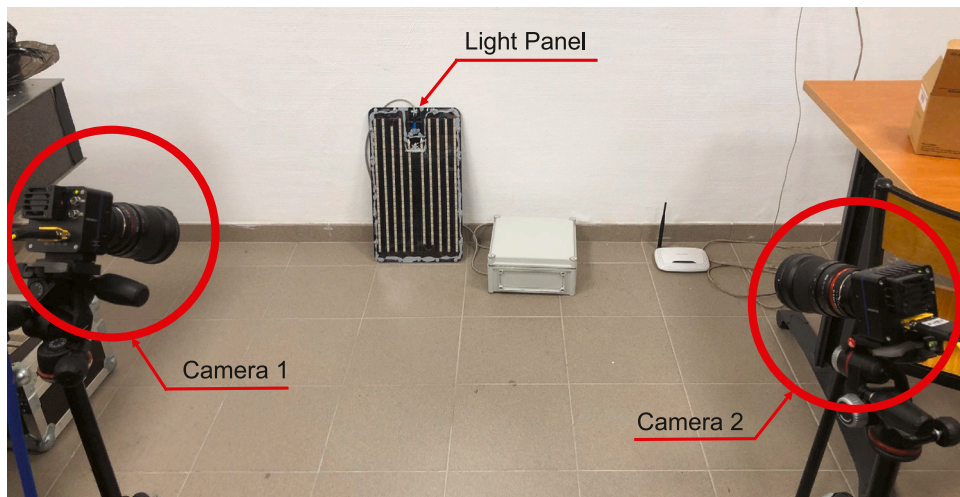


Fig. 11. Experimental setup for the timestamp reconstruction of two cameras in the controlled indoor environment.

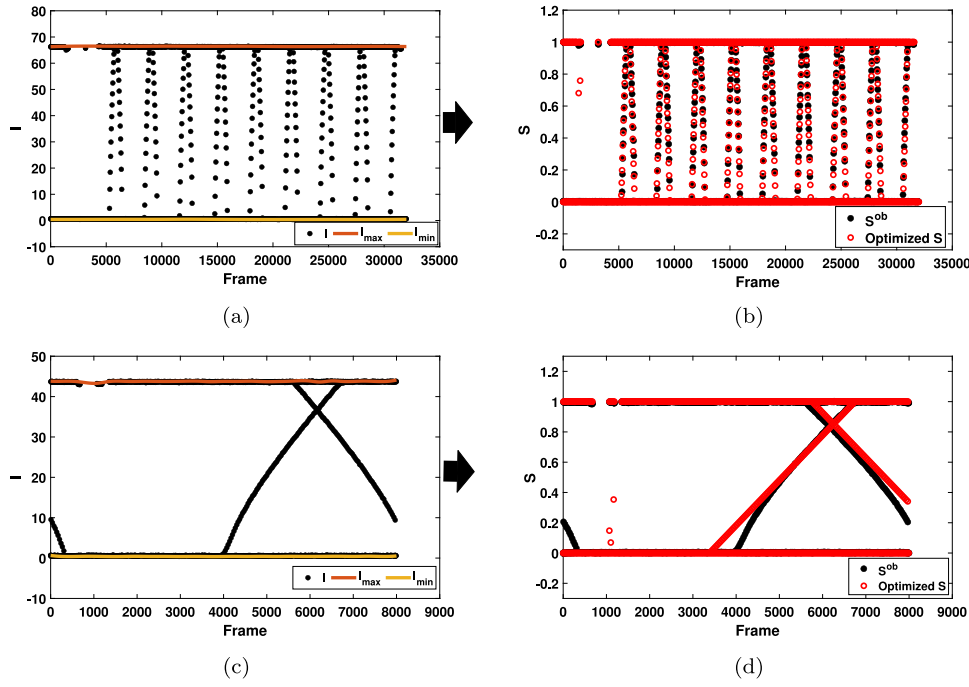


Fig. 12. Process to reconstruct the timestamps in a laboratory test.(a,b): $t_e = 1$ ms, $f_{ps} = 125$ Hz, (c,d): $t_e = 3$ ms, $f_{ps} = 31.25$ Hz. (a)(c) Raw signal: mean pixel intensity calculated for the ROI around the LED panel for each frame, and the corresponding upper and lower envelopes. (b)(d) Optimized results after normalization of the envelopes: comparison between the observed overlapping ratio S^{ob} and the overlapping ratio S optimized from the reconstructed timestamps.

Table 4

Ratio of the uncertainties δt_0 and δt_0^{RMSE} by the frame acquisition interval dt for all exposure times and frame rates of the parametric study. The uncertainty δt_0 is the absolute value of the difference of the offset times t_0 estimated for each camera $|t_{0,cam_1} - t_{0,cam_2}|$. The uncertainty δt_0^{RMSE} is computed as the RMSE between the reconstructed timestamps obtained for each camera ($C_{cam_1}^{start}$ and $C_{cam_2}^{start}$).

Ratio (in %)	f_{ps}			
$= \left(\frac{\delta t_0}{dt}, \frac{\delta t_0^{RMSE}}{dt} \right)$	31.25 Hz	62.5 Hz	125 Hz	250 Hz
t_e				
0.5 ms	(0.43, 0.24)	(0.29, 0.26)	(6.89, 3.78)	(7.51, 3.98)
1 ms	(0.10, 0.08)	(0.03, 0.04)	(1.19, 1.31)	(19.68, 11.20)
3 ms	(0.46, 0.32)	(1.69, 0.86)	(4.09, 2.90)	(7.28, 3.71)

Figs. 12(c) and 12(d) for $t_e = 3$ ms and $f_{ps} = 31.25$ Hz. The overlapping ratio S calculated using X_{opt} is represented by red circles in **Figs. 12(b) and 12(d)**.

Since it is assumed that these two cameras are perfectly synchronized through the hardware trigger, the uncertainty of the timestamp determined by the present method is estimated in two ways: as the absolute value of the difference of the offset times t_0 estimated for the two cameras ($\delta t_0 = |t_{0,cam_1} - t_{0,cam_2}|$) and the RMSE between the reconstructed timestamps obtained for each camera ($\delta t_0^{RMSE} = \sqrt{\frac{1}{N} \sum_{k=1}^N (C_{k,cam_1}^{start} - C_{k,cam_2}^{start})^2}$). **Table 4** presents the ratio of these uncertainties by the frame acquisition interval dt for all exposure times and frame rates of the parametric study. The results confirm that the uncertainty is significantly lower than the frame acquisition interval dt for all tests.

As the frame rate of commercial cameras is fixed or of very limited choice, for example the GoPro camera HERO9 only has two possible frame rates (30 or 60 Hz), another experiment is conducted at these frame rates, with $t_e = 0.5$ ms, to evaluate the accuracy of the timestamp reconstruction. The ratio of the absolute value of the difference of the offset times t_0 estimated for each camera, δt_0 , by the frame acquisition interval, dt , is equal to 0.39% for the frame rate of 30 Hz and to 1.71% for 60 Hz. For the ratio of the RMSE between the reconstructed timestamps obtained for each camera, δt_0^{RMSE} , by the frame acquisition

interval, it is equal to 0.20% for the frame rate of 30 Hz and 0.86% for 60 Hz. At these frame rates, a sub-millisecond accuracy is also achieved. Thus, it is possible to affirm that the proposed method is applicable with accuracy in the microsecond range when using a commercial camera, the frame rate of which cannot be modified and which does not provide other synchronization features.

4.2. Test case 2. Timestamp reconstruction of a camera network in uncontrolled environment

One purpose of using a camera network is to measure the 3D displacements of a structure, for example for structural health monitoring applications. In this section, it is studied whether the proposed method can be used and what is its accuracy for real outdoor conditions and the reconstruction of the displacements of a tower crane. The setup of the experiment in an outdoor environment is shown in **Fig. 13**. Five cameras are placed on the third floor of a building in order to record the motion of a tower crane located near the building. The LED panel and the control box are placed on the upper part of the tower crane (see the upper right corner of **Fig. 13**), so that the light signal from the LED panel could be filmed by each camera. As presented in **Table 5**, different models of camera are used. The frame rate is set to 31.25 Hz and different exposure times are set to suit the characteristics of each camera. The Cameras 4 and 5 are synchronized by a hardware trigger and the other three start video acquisition asynchronously. The timestamp is then reconstructed using the sequence of light state signal defined in Section 3.4 and used by the LED panel filmed by all cameras.

The process of timestamp reconstruction is shown in **Fig. 14**. In the first step, the mean pixel intensity I is calculated for the ROI with the LED panel. Next, the maximum and minimum values of the mean pixel intensity in the neighborhood of each image, I^{max} and I^{min} , are obtained. I^{max} and I^{min} represent the lower and upper envelopes of the mean pixel intensity. The number of neighborhood frames to determine I^{max} and I^{min} is 5 when the light state signal is in the *normal pattern* part and 401 when the light state signal is in the *time pattern* part. In this outdoor experiment, it can be seen that the local minima and maxima change due to the varying lighting conditions. Moreover, the lower and

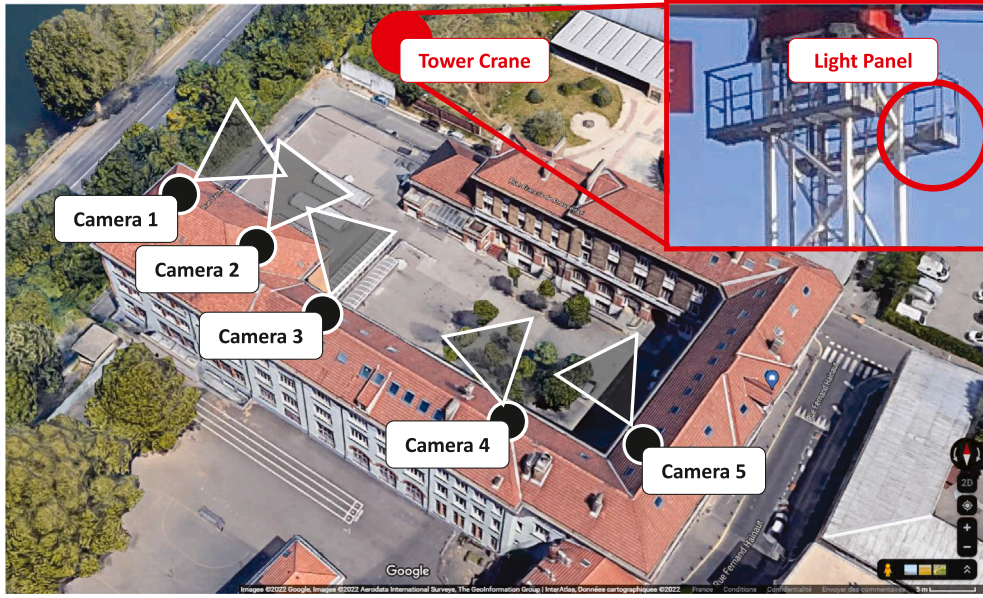


Fig. 13. Position of the cameras of the network in real environment for the monitoring of a tower crane (reprinted from Google Maps).

Table 5

Properties of the cameras used for the outdoor experiment.

	Camera 1	Camera 2	Camera 3	Camera 4	Camera 5
Camera model	Basler acA1300-200um	Basler acA1300-200uc	Vieworks VC-12MX	JAI SP-12000	JAI SP-12000
f_{ps}	31.25 Hz	31.25 Hz	31.25 Hz	31.25 Hz	31.25 Hz
t_e	1 ms	2 ms	3 ms	2.5 ms	2.5 ms
Synchronization with hardware trigger	NO	NO	NO	YES	YES

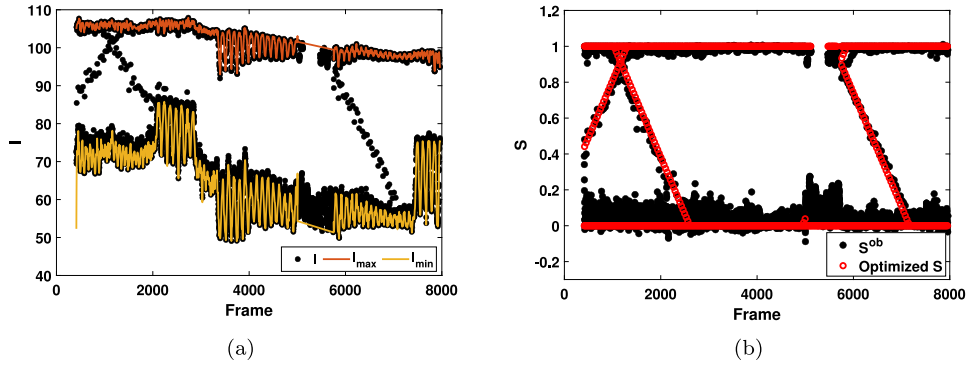


Fig. 14. Reconstruction process of the timestamps for the outdoor test (Camera 4). (a) Raw signal: mean pixel intensity calculated for the ROI around the LED panel for each frame, and the corresponding upper and lower envelopes. (b) Optimized result after normalization of the envelopes: comparison between the observed overlapping ratio S^{ob} and the overlapping ratio S optimized from the reconstructed timestamps.

upper envelopes oscillate due to the vibration of the tower crane. The resulting oscillations in the envelope signals are related to the number of neighborhood frames used to compute the local maxima and minima. These changes and the important noise level in pixel intensity due to the application of the proposed method in outdoor conditions can be clearly observed when comparing the raw pixel intensity (I) with that obtained in the laboratory, as illustrated in Fig. 12(c). Fig. 14(b) presents the normalized intensity signal S^{ob} and the signal S obtained after the optimization process, from the reconstructed timestamps. The influence of illumination changes and interferences due to vibration of the structure was mitigated in the normalization process in S^{ob} , however the noise level is larger than that observed in the laboratory tests.

The vector of optimized parameters \mathbf{X}_{opti} is reported in Table 6 for each camera. Thus, the timestamps of each camera can be reconstructed

based on Eq. (1). The camera image acquisition signal of each camera is shown in Fig. 15 for the first three frames when the acquisition start time of the 450th frame of Camera 3 (C_{450, cam_3}^{start}) is considered as a reference. In this way, the timestamp reconstruction of different cameras is carried out in an outdoor environment by reconstructing timestamps of frames acquired by each camera.

In this experiment, the external lighting, the vibration of the tower crane, and the large distance between the cameras and the LED panel may affect the accuracy of the timestamp reconstruction. Since Cameras 4 and 5 are synchronized by a hardware trigger, the uncertainty of the proposed method for this outdoor test is estimated by calculating the RMSE between the reconstructed timestamps obtained for these cameras, δt_0^{RMSE} , as previously done for the laboratory test. The uncertainty obtained is equal to 0.527 ms, i.e. 1.65% of the set frame acquisition interval $dt = 32$ ms. It is then possible to confirm that

Table 6

Camera settings and optimized parameters from the timestamp reconstruction process based on the coded light signal for the outdoor test.

		Camera 1	Camera 2	Camera 3	Camera 4	Camera 5
Camera settings	t_e (ms)	1	2	3	2.5	2.5
	dt (ms)	32	32	32	32	32
Optimized values	t_0 (ms)	152.740	156.394	140.167	167.816	168.261
	t_e (ms)	1.00	1.80	2.91	2.45	2.58
	dt (ms)	31.9993	31.9994	31.9991	31.9984	31.9984

Table 7

Comparison of the proposed method with the existing methods.

	Error as percentage of frame interval	Test Condition	Test scale of camera network
Zhao and Chen [18]	8%	Controlled + Uncontrolled	Lab-scale
Ding and Tao [25]	Not measured	Controlled	Lab-scale
Kim and Ichikawa [26]	1.62%, 2.9%	Controlled	Lab-scale
Proposed method	1.65%	Controlled + Uncontrolled	Large-scale (100 m)

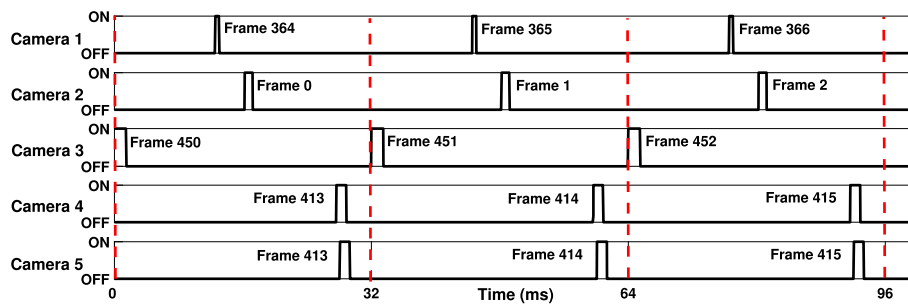


Fig. 15. Reconstructed timestamps of the camera sensor signals with sub-frame accuracy when the acquisition start time of the 450th frame of Camera 3 is considered as a reference.

the uncertainty of the proposed timestamp reconstruction method in an outdoor environment is not only in the sub-millisecond range but also 60 times smaller than the frame acquisition interval. Additionally, our error is compared with the similar methods that were presented by Zhao and Chen [18], Ding and Tao [25], and Kim and Ichikawa [26] in Table 7.

5. Conclusions

A method of sub-frame timestamping for camera networks through a coded light signal is presented in this paper. It reconstructs the timestamp of asynchronous frames by post processing the light signal in sequential images. The influence of camera exposure time and frame rate on the uncertainty of timestamp identification is investigated by numerical simulations. It helps to decide the parameter values of the coded light signal. The proposed timestamp reconstruction procedure is evaluated through two experiments, one indoor and the other outdoor for monitoring a tower crane. Then, sub-millisecond uncertainty is obtained in the outdoor experiment although this latter is affected by poorly controlled lighting conditions, structural motion, and reduced camera resolution. This corresponds to an uncertainty of 1.65% of the time interval between two consecutive frames.

The proposed method can be carried out with a simple and low-cost system. Additionally, since the timestamping of sequential images is performed through a light signal, the camera network can be configured with cameras that do not support functions such as hardware-triggered image acquisition and network-based synchronization (NTP or PTP). Therefore, commercial-grade cameras without synchronization feature can be used simply for infrastructure and environment monitoring. Finally, it does not require any line connection between cameras, so some restriction in camera position can be overcome.

However, there is a limitation in that the proposed method does not allow cameras of a camera network to acquire every image simultaneously as our method is applicable when this is not possible. This

is because our method is used to determine the instant time when the sensor has operated and the image is acquired. In the case of accurate 3D measurements, the extracted timestamps of each frame can be used for further analysis. Moreover, since the LED panel should be located in the common FOV of the camera network, our method is adequate when one structure is continuously monitored by several cameras.

CRedit authorship contribution statement

Yunhyeok Han: Writing – review & editing, Writing – original draft, Methodology, Investigation, Formal analysis. **Stefania Lo Feudo:** Writing – review & editing, Supervision, Methodology, Funding acquisition. **Gwendal Cumunel:** Writing – review & editing, Writing – original draft, Supervision, Funding acquisition. **Franck Renaud:** Writing – review & editing, Supervision, Methodology, Investigation, Funding acquisition, Formal analysis, Conceptualization.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

Data will be made available on request.

Acknowledgments

The authors want to thank Simon Boucher and his colleagues from Eiffage for helping them setting the light panel on the tower crane. The authors acknowledge the support of the French National Research Agency (ANR), under grant ANR-21-CE22-0027 (project DYNATIM-BEREYES).

References

- [1] D. Li, X. Wang, C.C. Menassa, V.R. Kamat, Understanding the impact of building thermal environments on occupants' comfort and mental workload demand through human physiological sensing, in: F. Pacheco-Torgal, E. Rasmussen, C.-G. Granqvist, V. Ivanov, A. Kaklauskas, S. Makonin (Eds.), *Start-Up Creation*, second ed., in: Woodhead Publishing Series in Civil and Structural Engineering, Woodhead Publishing, 2020, pp. 291–341, <http://dx.doi.org/10.1016/b978-0-12-819946-6.00012-6>.
- [2] P. Shrestha, M. Barbieri, H. Weda, Synchronization of multi-camera video recordings based on audio, in: *Proceedings of the ACM International Multimedia Conference and Exhibition*, 2007, pp. 545–548, <http://dx.doi.org/10.1145/1291233.1291367>.
- [3] A. Llagostera Casanovas, A. Cavallaro, Audio-visual events for multi-camera synchronization, *Multimedia Tools Appl.* 74 (4) (2015) 1317–1340, <http://dx.doi.org/10.1007/s11042-014-1872-y>.
- [4] A. Dore, M. Pinasco, C.S. Regazzoni, Multi-modal data fusion techniques and applications, in: *Multi-Camera Networks: Principles and Applications*, 2009, pp. 213–237, <http://dx.doi.org/10.1016/B978-0-12-374633-7.00011-2>.
- [5] Q. Zhang, A.B. Chan, Single-frame-based deep view synchronization for unsynchronized multicamera surveillance, *IEEE Trans. Neural Netw. Learn. Syst.* (2022) 1–15, <http://dx.doi.org/10.1109/TNNLS.2022.3170642>, [arXiv:2007.03891](https://arxiv.org/abs/2007.03891).
- [6] W. Qiu, H.M. Xie, X.L. Gong, Special collection on new methods and applications of optical and photo-mechanics, *Exp. Mech.* 61 (9) (2021) 1507–1509, <http://dx.doi.org/10.1007/s11340-021-00799-2>.
- [7] N.V. Doan, V.T. Le, H.C. Park, N.S. Goo, Modal analysis using a virtual speckle pattern based digital image correlation method: An application for an artificial flapping wing, *Exp. Mech.* 62 (2) (2022) 253–270, <http://dx.doi.org/10.1007/s11340-021-00775-w>.
- [8] R.J. Radke, Multi-view geometry for camera networks, in: *Multi-Camera Networks: Principles and Applications*, 2009, pp. 3–27, <http://dx.doi.org/10.1016/B978-0-12-374633-7.00003-3>, (September 2008).
- [9] M. Kalaitzakis, N. Vitzilaos, D.C. Rizos, M.A. Sutton, Drone-based stereodic: System development, experimental validation and infrastructure application, *Exp. Mech.* 61 (6) (2021) 981–996, <http://dx.doi.org/10.1007/s11340-021-00710-z>.
- [10] Y. Chi, L. Yu, B. Pan, Low-cost, portable, robust and high-resolution single-camera stereo-DIC system and its application in high-temperature deformation measurements, *Opt. Lasers Eng.* 104 (2018) 141–148, <http://dx.doi.org/10.1016/j.optlaseng.2017.09.020>, *Optical Tools for Metrology, Imaging and Diagnostics*.
- [11] A.C. Silva, J. Lambros, D.M. Garner, E.A. Patterson, Dynamic response of a thermally stressed plate with reinforced edges, *Exp. Mech.* 60 (1) (2020) 81–92, <http://dx.doi.org/10.1007/s11340-019-00536-w>.
- [12] F. Chen, X. Chen, X. Xie, X. Feng, L. Yang, Full-field 3D measurement using multi-camera digital image correlation system, *Opt. Lasers Eng.* 51 (9) (2013) 1044–1052, <http://dx.doi.org/10.1016/j.optlaseng.2013.03.001>.
- [13] H. Hu, J. Liang, Z. zhong Xiao, Z. zong Tang, A.K. Asundi, Y. xin Wang, A four-camera videogrammetric system for 3-D motion measurement of deformable object, *Opt. Lasers Eng.* 50 (5) (2012) 800–811, <http://dx.doi.org/10.1016/j.optlaseng.2011.12.011>.
- [14] P. Hyla, Multi camera triggering and synchronization issue : case study, *J. KONES Powertrain Transp.* 23 (3) (2016) 193–200.
- [15] G. Litos, X. Zabulis, G. Triantafyllidis, Synchronous image acquisition based on network synchronization, in: *2006 Conference on Computer Vision and Pattern Recognition Workshop, CVPRW'06*, 2006, p. 167, <http://dx.doi.org/10.1109/CVPRW.2006.200>.
- [16] V. Subramanyam, J. Kumar, S.N. Singh, Temporal synchronization framework of machine-vision cameras for high-speed steel surface inspection systems, *J. Real-Time Image Process.* 19 (2) (2022) 445–461, <http://dx.doi.org/10.1007/s11554-022-01198-z>.
- [17] IEEE standard for a precision clock synchronization protocol for networked measurement and control systems, in: *IEEE Std 1588-2019 (Revision of IEEE Std 1588-2008)*, 2020, pp. 1–499, <http://dx.doi.org/10.1109/IEEESTD.2020.9120376>.
- [18] Q. Zhao, Y.Q. Chen, High-precision synchronization of video cameras using a single binary light source, *J. Electron. Imaging* 18 (4) (2009) 040501, <http://dx.doi.org/10.1117/1.3247860>.
- [19] E. Olson, A passive solution to the sensor synchronization problem, in: *IEEE/RSJ 2010 International Conference on Intelligent Robots and Systems, IROS 2010 - Conference Proceedings*, 2010, pp. 1059–1064, <http://dx.doi.org/10.1109/IROS.2010.5650579>.
- [20] P. Shrestha, H. Weda, M. Barbieri, D. Sekulovski, Synchronization of multiple video recordings based on still camera flashes, in: *Proceedings of the 14th Annual ACM International Conference on Multimedia, MM 2006, Vol. 2*, 2006, pp. 137–140, <http://dx.doi.org/10.1145/1180639.1180679>, (i).
- [21] S.N. Sinha, M. Pollefeys, Synchronization and calibration of camera networks from silhouettes, in: *Proceedings - International Conference on Pattern Recognition*, Vol. 1, 2004, pp. 116–119, <http://dx.doi.org/10.1109/ICPR.2004.1334021>.
- [22] A. Elhayek, C. Stoll, K.I. Kim, H.P. Seidel, C. Theobalt, Feature-based multi-video synchronization with subframe accuracy, in: *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, in: LNCS, vol. 7476, 2012, pp. 266–275, http://dx.doi.org/10.1007/978-3-642-32717-9_27.
- [23] Z. Zhang, C. Wang, W. Qin, Semantically synchronizing multiple-camera systems with human pose estimation, *Sensors* 21 (7) (2021) 1–12, <http://dx.doi.org/10.3390/s21072464>.
- [24] T. Tuytelaars, L. Van Gool, Synchronizing video sequences, in: *Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition*, Vol. 1, 2004, pp. 1–7, <http://dx.doi.org/10.1109/cvpr.2004.1315108>.
- [25] X. Ding, P. Tao, Synchronization detection of multicamera system based on LED matrix, in: *2016 13th International Conference on Embedded Software and Systems, ICESS*, 2016, pp. 18–23, <http://dx.doi.org/10.1109/ICISS.2016.13>.
- [26] H. Kim, M. Ishikawa, Sub-frame evaluation of frame synchronization for camera network using linearly oscillating light spot, *Sensors* 21 (18) (2021) <http://dx.doi.org/10.3390/s21186148>.
- [27] Y. Cao, X. Pan, X. Zhao, H. Wu, An analog gamma correction scheme for high dynamic range CMOS logarithmic image sensors, *Sensors (Switzerland)* 14 (12) (2014) 24132–24145, <http://dx.doi.org/10.3390/s141224132>.