Design of Active Visual Marker for High Speed Object Tracking by Frame Subsampling

Adnan Kreho, Jasmin Hadzajlic and Emir Sokic
Department of Automatic Control and Electronics
Faculty of Electrical Engineering, University of Sarajevo
Sarajevo, Bosnia and Herzegovina
{akreho1, jhadzajlic1, esokic}@etf.unsa.ba

***Abstract*—The trajectory of a moving object may be extracted from video using image processing algorithms. However, the quality of the extracted information largely depends on the frame rate and exposure time of the camera, thus it is difficult to capture fast movement using slower and less expensive cameras. To this end, we propose to use an active modulated light source for object tracking, interacting with exposure times and subsampling existing frames. A prototype of a multi-functional active visual marker is presented in this paper. The system is based on the ESP-WROOM-32 microcontroller, which is configured to use various communication protocols, namely WiFi 802.11, RF 2.4 GHz GFSK and RS485. The microcontroller controls the RGB LED, which is used as light source. In addition, the system can be synchronized with the external real-time clock. The experimental results have illustrated the advantages and disadvantages of the designed active markers and pointed out the directions for future work and development.**

***Index Terms*—Object tracking, active marker, frame subsampling, embedded systems.**

I. Introduction

Motion analysis is frequently encountered in many fields, from filmography [1], stress and vibration analysis of mechanical systems [2], to estimation of position of UAV in closed area [3]. Occasionally, it is possible to install obtrusive measurement sensors (e.g. in industrial automatic control), such as incremental encoders or measuring plates [4]. On the other hand, there are cases when it is important that the sensor does not affect the measuring process due to the fact that it disturbs the movement of the object and has its own dynamics. Hence, the non-obtrusive sensors, such as cameras, lasers, radar or lidars need to be used [5]. These sensors usually require significant amount of signal preprocessing and postprocessing.

Using video cameras for motion analysis is convenient for a number of reasons. Cameras are present in everyday modern world, they are relatively inexpensive compared to other similar sensing methods and the results may be easily visualized. Cameras may record a series of photos from which the relative position of the observed object can be determined. There are many available algorithms which are developed for tracking objects on videos (e.g. [6]–[10]). Some of them are based on passive methods (tracking shape, color or texture [6]), or on active methods (active markers) [7]. Some of them have difficulties in achieving high accuracy [8], perform in real time [9] or require expensive equipment [10].

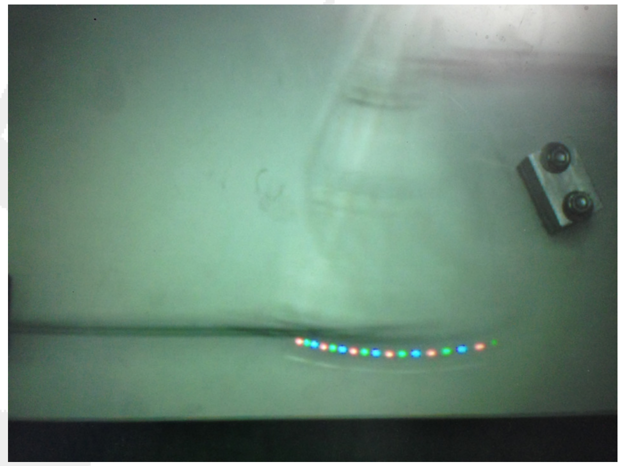

Fig. 1. Frame taken while moving the proposed modulated active marker - 2FPS, 500ms exposure time

When tracking fast objects, the number of frames per second of the camera, as well as the exposure time needs to be adjusted accordingly. Cameras that are used in everyday life, typically have frame rate only up to 60 FPS for video recording. Therefore, for tracking high speed moving objects (for example, when capturing objects that move more than 1-2 pixels during 1/FPS seconds) this FPS is not high enough, and the ”jumps” between frames are relatively high. If the exposure time is equal to 1/FPS then the image may even be blurred. Moreover, if the exposure time of the camera is significantly reduced, then the scene needs additional light. Thus, we proposed to use the camera in combination with a specifically designed active marker - a light source, which can be adequately modulated throughout frame acquisition, which would enable to:

- facilitate image processing (much easier extraction of points of interest)
- eliminate the need for additional light,
- sub-sample the frame by modulating the marker source, hence increasing temporal resolution of the extracted trajectory.

The main idea is presented in Fig.1. It can be seen that while the fast object was moving (1100 mm/s, 10 pixel/ms), the active marker was leaving a trail which may be exploited to analyze the movement of the mechanical system. The final goal is to design a standalone active marker which may be connected to or synchronized with a standard (low FPS) PC-based or mobile phone-based camera, which would allow to simply acquire the trajectory of the recorded objects. In this paper, only the design of the prototype marker and some illustrative experiments will be discussed, while the complete hardware and software implementation will be described in our future papers.

The paper is organized as follows. The motivation for the proposed design is given in section II. The third section discusses the hardware structure of the proposed prototype, while the fourth section gives some insight in software implementation. Experimental results are given in section V and conclusion is given in the last section.

II. Motivation

There are two important parameters of every camera: frame rate or the number of frames per second (FPS) and Exposure time. FPS represents the number of photos (frames) a camera can record in one second. Fast cameras are characterized with higher FPS, which means they can record more images in a time unit.

Exposure time represents a time period during which the optical sensor is exposed to the light, or simply said, the time period during which a frame is being taken. Reciprocal value of exposure time is called shutter speed. Photographers tend to use lower exposure time when they are trying to achieve sharp shots, while higher exposure time usually brings smoother shots, which may create blurriness. While the exposure time is usually user-defined, the frame rate is predefined by the camera manufacturer.

Motion blurriness which appear on frames that capture moving objects is usually considered as an additional difficulty for image processing algorithms which attempt to extract information about current position of an object on a frame. An example is given in Fig.2. The blurriness of the image is affected by exposure time and the speed of the object through the composition. Higher exposure time means that the optical sensor will be longer exposed to the light. Therefore, setting higher exposure time is appropriate when the photo is recorded in darker environment. However, higher exposure time results in motion blurriness if the objects in photo's composition are moving fast. Reducing exposure time of the sensor will result in less blur on images, but will require additional illumination of the scene, and will reduce the temporal resolution of the extracted trajectory from the sequence of video frames.

Another example, so called ”running light”, which represents series of eight LED diodes which are consequently turned on and off is illustrated in Fig. 3. Although all LEDs are turned on the same amount of time, with the same amplitude, and only one LED is active at the time, it seems that many LEDs are active simultaneously and that some of them are brighter than others. The explanation of this effect is presented in Fig.4. Three consecutive frames are represented by three dashed lines, while the exposure time is represented with pink

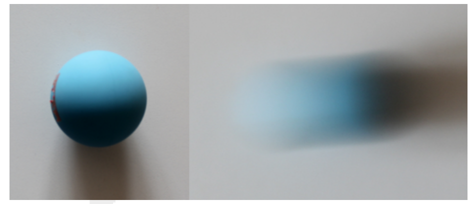

Fig. 2. Photos of a moving ball with different exposure times

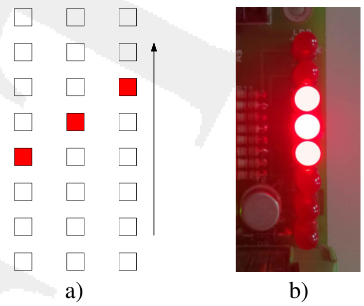

Fig. 3. Illustration of the ”running light” effect a) principle of LED activation, b) actual frame acquired with a longer exposure time.

colored areas. During frame acquisition the shutter will open, the sensor will be exposed to the light and capture some of the LEDs turning on and off. For the rest of the frame period, the shutter will be closed and the sensor will stop capturing the frame. The shutter will be again opened starting the following frame period.

Some of the principles explained in the previous example can be used to design a usable active marker object tracker. Namely, instead of sequentially turning on and off multiple LEDs, one LED may be modulated on a moving object resulting in similar images. Additionally, the motion blurriness of points of interest will be reduced if the attached light source can emit light impulses in short time periods.

The design of the prototype for the proposed control system of the specific light source will be presented in the following section.

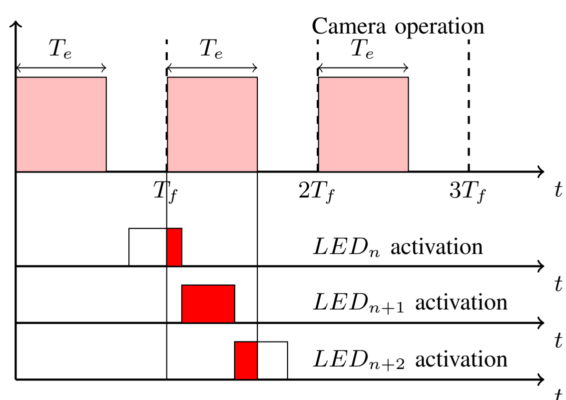

Fig. 4. Explanation of the ”running light” capture effect - during one frame exposure, several LEDs are active and therefore create effects of simultaneous activation.

III. Hardware structure of the proposed system

The main goal was to design and build a prototype system to control a miniature light source. The light source should be developed in two versions - wireless (standalone, battery powered, controllable by mobile-phone or tablet) and wired (powered by a power supply, synchronized with PC and/or camera). The light source should be able to be attached to an object whose trajectory is of interest. Finally, the system should be tested to see if it is possible to extract information about object's trajectory from series of frames recorded with a camera.

The prototype system consists of a light source, control system (microcontroller), communication modules and synchronization module.

A. Light source

The most adequate light source is a RGB LED. It is suitable due to the fact that it is extremely simple to control, has relatively quick response, the color and intensity of the emitted light can be set, it provides a significant amount of light considering its dimensions and it is inexpensive. The chosen RGB LED is Lilypad Tri-Color LED DSE-153-008 which is a common-anode RGB.

B. Control system

The most important part of the prototype is the microprocessor-based system which is used to control the RGB LED. Microprocessor-based system ESP-WROOM-32 is chosen for this purpose. ESP32 is based on chip ESP32-D0WDQ6, which consists of two Xtensa 32-bit LX6 microprocessors. It has 448 kB of internal memory for programming and basic functions and 520 kB SRAM on chip for data and instructions. [11] One of the more important characteristics of ESP32 is that it supports Wi-Fi, Bluetooth and BLE communication protocols, which makes it suitable for this prototype.

C. Communication modules

Three communication protocols are considered and tested for communication and synchronization with external computer: Wi-Fi 802.11, RF 2.4 GHz GFSK and RS485.

The ESP-WROOM-32 is out-of-the box capable to communicate using Wi-Fi protocol. In order to transfer needed parameters, socket programming is used. MCU is chosen as socket server, while the PC is set up to be socket client.

For implementation of the RF 2.4 GHz GFSK communication protocol, two nRF24L01 transceivers are used. One nRF24L01 board interfaced with MCU via SPI protocol serves as data receiver, while the other one interfaced with PC over an auxiliary Arduino UNO serves as data transmitter.

Finally, to implement RS485 protocol, transceiver SP3485 is used. The transceiver is interfaced with MCU using UART2 port and it serves as data receiver. On the transmission side, the PC is connected to a communication module via USB to RS485 converter MAX385.

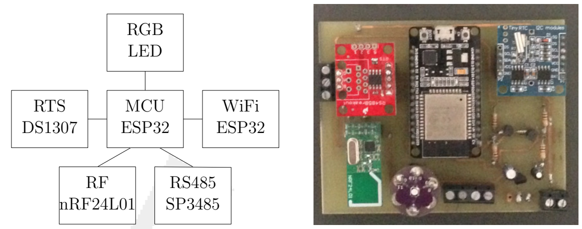

Fig. 5. a) Proposed system structure, b) actual prototype

The first two communication approaches are wireless thus they are convenient for building standalone wireless modules. WiFi requires a compatible communication device, which is not a significant constraint taking into account that many mobile phones, tablets and PCs are WiFi enabled and that more and more space is covered with WiFi signal every day in the modern world. RF 2.4GHz modules require additional electronic components to connect with the PC. RS485 is the communication of choice when active markers are supposed to be attached by wire.

D. Synchronization modules

It is important that the light source has the ability to switch on and off in precise time moments. The relative synchronization could be achieved internally using the clock of the MCU itself (which is quartz based). However, absolute synchronization with the specific camera and/or PC can be done either with interactive "ticks" from the PC to the marker (which, for example, correspond to specific points such as start of frames or frame blocks), or by setting the same absolute time on both sides. One way to do it is to use internal or external real-time clock (RTC) subsampled to microsecond precision. To this end, a DS1307 RTC is attached to the prototype module.

E. Assembled system

The complete system structure, as well as the fully developed system, is illustrated in Fig.5. In the current state, the LED is attached to the system with wires. Different communication approaches are available, however due to the system higher operating speed, RS485 is preferred to be used for in-frame synchronization, while the other two methods are more convenient to be used for pre-acquisition setup.

IV. Software structure of the proposed system

A. Light source control

In order to effectively control the light source, the control parameters have to be initialized properly. The software commands are organized in a way that each of the three diodes can either emit one short light impulse and be turned off afterwards ("pulse" mode), or emit light impulses periodically ("burst" mode). Due to the fact that single LED may be controlled separately, there are literally infinite combinations of visual signals that may be generated on images. Typical control signals for the LED diodes are presented in Fig.6.

Accordingly, the needed control parameters are:

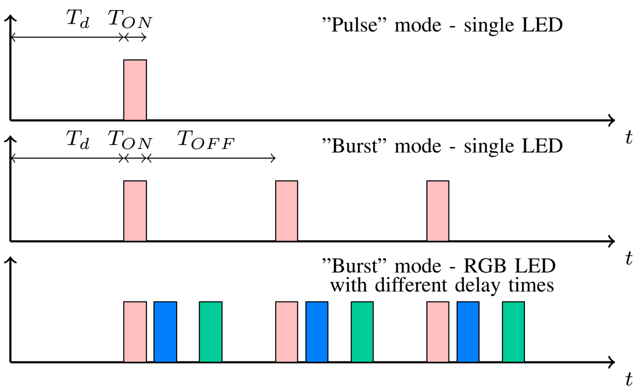

Fig. 6. Light source operation modes.

- flag to determine the diode operating mode,
- time delay before first impulse emission,
- diode's ON-time,
- diode's OFF-time.

The control parameters are sent in the following form:
#A*XYZ$XYZ%XYZ&T#B*XYZ$XYZ%XYZ&T#C*XYZ$XYZ%XYZ&T
The string contains 48 characters divided in three substrings, each 16 characters long. The substrings, starting with #A, #B and #C consist of data used for setting up control parameters for red, blue and green LED, respectively. In case that any of these characters are not transferred in its original form, re-transmission of the message is requested. X, Y and Z are digits from 0 to 9. In every substring, three-digit number *XYZ* after character * represents time delay before the emission of light impulse(s) in multiples of 100 microseconds. The three-digit number following the character *&* represents diode's ON-time, while the three-digit number following the character *%* represents diode's OFF-time, both in multiples of 100 microseconds.

The digit T can be 0 or 1. If it is 0, the LED should emit just one light impulse. Otherwise, the LED should emit periodical series of light impulses.

B. System triggering

There are two ways to trigger the system:

- externally, using communication modules,
- internally, using real-time clock.

Each approach requires the control parameters to be set prior to the trigger signal.

In case of external triggering, after all control parameters are set up, the system is waiting until it receives string #z. The cyclic interrupt routine begins operating when the microcontroller receives the command, through which the microcontroller generates the impulses to the LED.

If the triggering is done using real-time clock, the system waits until the time preset by user. The system may operate on two real time clocks:

- internal RTC on MCU,
- external RTC.

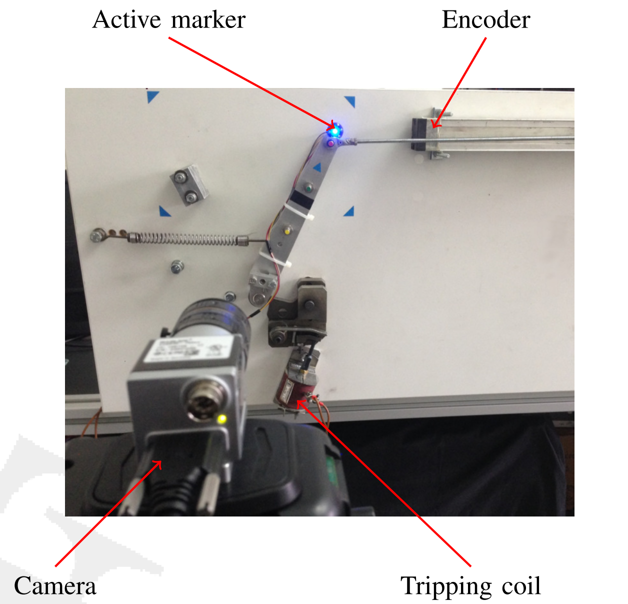

Fig. 7. Model of the motion mechanism of 420kV HVCB

a) b)

Fig. 8. a) Model od the motion mechanism of 420kV HVCB, b) actual moving mechanism of 420kV HVCB

Internal RTC is located inside the MCU, while DS1307 RTC is used as external RTC.

When the MCU connects to the local Wi-Fi network, it synchronizes both RTCs with the NTP server *de.pool.ntp.org*. If the user chose to trigger the system by real-time clock, the moment when the trigger signal should occur need to be defined. When the RTC signalizes that the predefined time has elapsed, the interrupt routine starts executing, as if external triggering was used.

V. Experimental Results

In order to test and verify the operation of the prototype, an experimental testbed was developed (Fig.7 and Fig.8a). The testbed is actually a replica of the motion mechanism of the 420kV High Voltage Circuit Breaker on which the developed system will be installed (Fig. 8b).

The model illustrated in Fig.8 is mechanically simple. The lever is attached to the spring, and manually strained. When the coil is energized, it pushes the anchor, releases the hatch and the lever starts rotating from right to left. When the lever hits

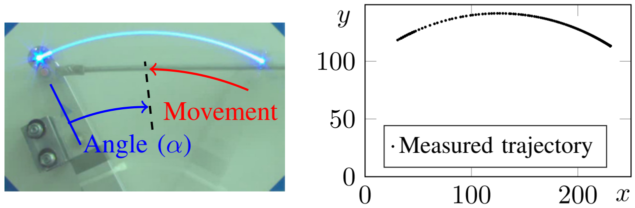

Fig. 9. Experiment nr.1 - left: single frame, 2 FPS, 500[ms] exposure time, blue LED constantly on., right: measured trajectory

the limiter, it slightly bounces, until it finally stops moving. In order to be able to synchronize the motion with the start of the video and the active marker, a simple control and acquisition system is used. This system can trip the coil and it is able to measure the actual movement of the sensor (indirectly) by using linear encoder. The encoder is sampled every 1[ms]. The actual position of the sensor is computed from the encoder readings, using analytical geometry. Prior to semi-automatic extraction of the measurement data, the camera need to be calibrated in order to minimize lens distortion effects. The blue stickers are used to eliminate the projective transformation.

The following experiments conducted with different camera setups (FPS, exposure time) and active marker setups will be discussed in this paper:

- Exp. 1 - 2 FPS, 500ms expos. - B LED, constantly on,
- Exp. 2 - 2 FPS, 500ms expos. - B LED, "burst" mode,
- Exp. 3 - 2 FPS, 500ms expos. - RGB LED, "burst" mode,
- Exp. 4 - 20 FPS, 45ms expos. - RGB LED, "burst" mode,
- Exp. 5 - 100 FPS, 9.5ms expos. - RGB LED, "burst" mode,

In the first experiment (Fig.9), the LED is constantly on and the frame rate is very low (2FPS). The whole movement may be recorded on a single image. Although the movement path may be extracted from the image - the trajectory (movement in time) may not.

To this end, the marker is put into "burst" mode (1[ms] ON, 2[ms] OFF) in Experiment nr.2 (Fig.10). The whole movement is also captured on one frame. In contrast, now it is possible to extract the exact position of the marker in time. The assumption used in order to extract the part of the trajectory is that the marker does not perform sudden changes of movement direction. There are two issues which are immediately identified - first the synchronization is rather difficult (all "dots" are the same color) and in case of slow movement the marker trail on the image is overlapped (blurred). Therefore the exact location of the marker is indeterminable. Indeterminable regions of the trajectory need to be discarded.

Experiment nr.3 has the same setup, only the LED color is now altered (Fig. 11). Alteration of the marker color can improve the performance of the markers - one of the color may be used e.g. for synchronization. Moreover, if the color is altering in a specific manner, the points incorrectly identified by the processing algorithm may easily be eliminated and/or predicted/interpolated.

The first three experiments suggest that better results might

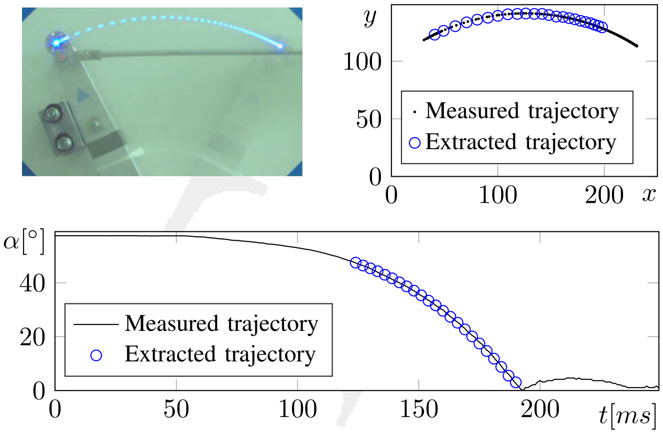

Fig. 10. Experiment nr.2 - up left: single frame, 2 FPS, 500[ms] exposure time, blue LED "burst" mode (1[ms] ON, 2[ms] OFF), up right: measured and extracted trajectory, below: measured and extracted trajectory (movement angle α).

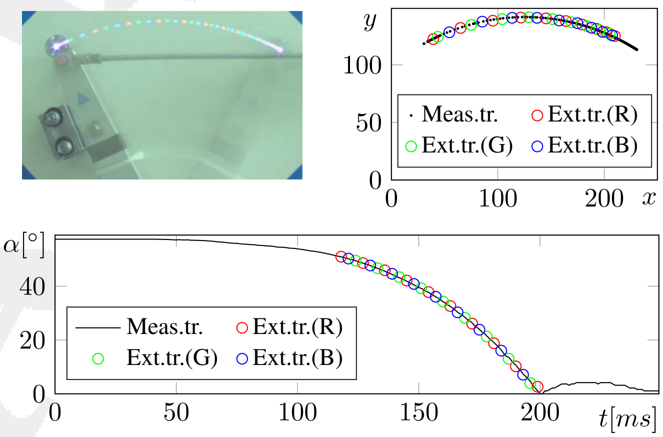

Fig. 11. Experiment nr.3 - up left: single frame, 2 FPS, 500[ms] exposure time, RGB LED "burst" mode (1[ms] ON, 2[ms] OFF), up right: measured and extracted trajectory, below: measured and extracted trajectory (movement angle α).

be achieved if the frame rate is increased. Therefore, the Experiment nr.4 is conducted using 20FPS (50[ms] per frame), and 45ms exposure (Fig.12). It can be seen that not all frames are usable for trajectory extraction, but now the synchronization is much simpler, and the localization of the marker is easier. It is important to note that the marker is visible only if it is captured during exposition time (which may be shorter than the period of the frame). It is interesting that the images are relatively blurred, but the extraction is accurate due to the fact that the marker is adequately modulated.

In the last presented Experiment nr.5 (Fig.13), 100FPS frame rate is used, and 10[ms] exposure time (which correspond to the 10[ms] sampling period). Note that in this setup, blur-free images require exposure time of 0.4[ms] at most. However, this would require additional light source, and still, the resulting sampling rate would not be high enough to accurately capture the movement. Using the proposed active marker it is possible to effectively "subsample" the frame. In addition, synchronizing the frame rate and exposure time with

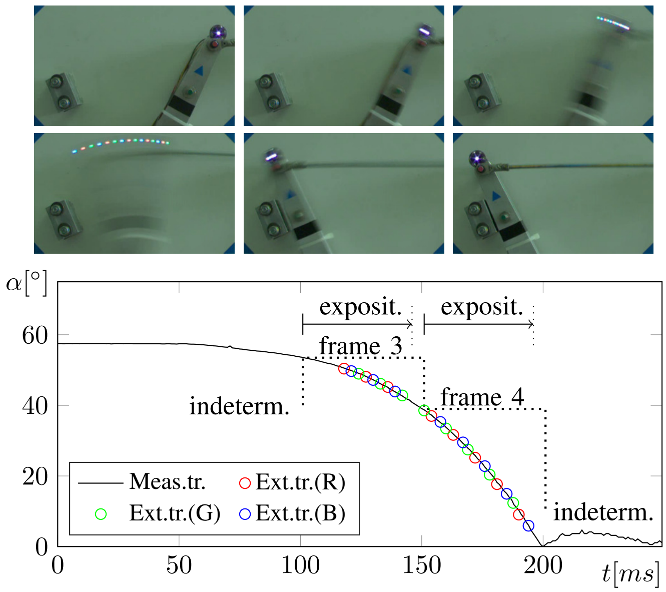

Fig. 12. Experiment nr.4 - 20 FPS, 45 [ms] exposure time, RGB LED ”burst” mode (1[ms] ON, 2[ms] OFF).

the active marker increases the robustness of the detection (e.g. if the R,G and B LED are indistinguishable then the chances are that the object is not moving, therefore the detection error does not affect the measurements).

VI. Conclusion and Future Work

The paper presented the design and verification of the concept of the actively modulated visual marker for high speed object tracking. There are multiple advantages of using this system - it does not require expensive high speed camera or additional lighting fixtures, while the color and shape-based extraction of the marker dotted trail from the image is almost trivial. The marker showed promising results, and pointed out the future direction of the work. Namely, the plan is to develop a wireless standalone battery-powered active marker. Moreover, the marker needs to have the possibility to (adaptively) trigger itself to every frame and not just the start of the measurement. The fully automated image-processing application for extracting the trajectory of the marker(s), based on the observed principles will also be developed.

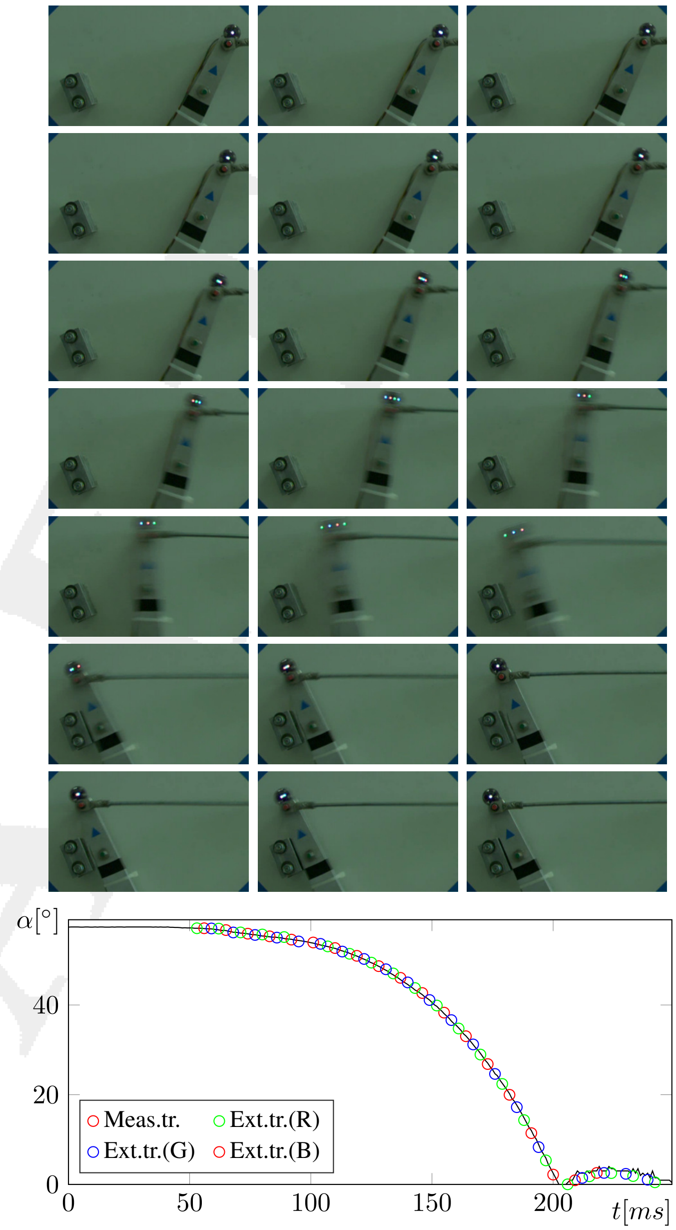

Fig. 13. Experiment nr.5 - 100 FPS, 9.5[ms] exposure time, RGB LED ”burst” mode (0.2[ms] ON, 2.8[ms] OFF).

References

[1] K. Kurzhals, M. John, F. Heimerl, P. Kuznecov and D. Weiskopf, *”Visual Movie Analytics”*, *IEEE Transactions on Multimedia*,Vol. 18, pp. 2149-2160, Sep. 2016

[2] O. Buyukozturk, J. G. Chen, N. Wadhwa, A. Davis, F. Durand and W. T. Freeman, *”Smaller Than the Eye Can See: Vibration Analysis with Video Cameras”*, *World Conference on Non-Destructive Testing 2016*, 2016

[3] A. Mashood, A. Dirir, M. Hussein, H. Noura and F. Awwad, *”Quadrotor Object Tracking using Real-Time Motion Sensing”*, *5TH International Conference on Electronic Devices, Systems and Applications (ICEDSA)*, 2016

[4] A. Anuchin, V. Astakhova, D. Shpak, A. Zharkov and F. Briz, *”Optimized method for speed estimation using incremental encoder”*, *International Symposium on Power Electronics*, Oct. 2017

[5] C. Weitkamp, *Lidar : Range-Resolved Optical Remote Sensing of the Atmosphere*, New York, Springer, 2005

[6] D. Jansari, S. Parmar and G. Saha, *”Real-time object tracking using color-based probability matching”* , *IEEE International Conference on Signal Processing, Computing and Control (ISPCC)*, Sep. 2013

[7] M. T. K. Tsun, B. T. Lau, H. S. Jo and S. L. Lau, *”A human orientation tracking system using Template Matching and active Infrared marker”*, *International Conference on Smart Sensors and Application (ICSSA)*, May 2015

[8] S. Prasad and S. Sinha, *”Real-time object detection and tracking in an unknown environment”*, *World Congress on Information and Communication Technologie*, Dec. 2011

[9] D. Zhou and D. Hu, *”A robust object tracking algorithm based on SURF”*, *International Conference on Wireless Communications and Signal Processing*, Oct. 2013

[10] O. Javed, Z. Rasheeed, K. Shafique, M. Shah, *”Tracking Across Multiple Cameras With Disjoint Views”*, *Proceedings Ninth IEEE International Conference on Computer Vision*, Oct. 2003

[11] Espressif Systems, *”ESP32-WROOM-32”*, ESP32-WROOM-32 Datasheet, Jan. 2019