

Frequency-selective trajectory analysis for pulsating visual marker detection under one-dimensional image motion

Piotr Miś^{1*} , Przemysław Szulim² 

¹ Division of Automotive Engineering, Mechatronics and Mechanics, Faculty of Automotive and Construction Machinery Engineering, Warsaw University of Technology, Narbutta 84, 02-524 Warsaw, Poland

² Division of Numerical Methods and Intelligent Structures, Faculty of Automotive and Construction Machinery Engineering, Warsaw University of Technology, Narbutta 84, 02-524 Warsaw, Poland

* Corresponding author's e-mail: piotr.mis.dokt@pw.edu.pl

ABSTRACT

Visual markers emitting pulsating light can act as reference points in camera-based localization systems. Their identification is possible based on the frequency of changes in the intensity of the detected light. In the localization system concept developed by the authors, frequency information is intended to facilitate the detection and differentiation of markers whose spatial position is known. However, a significant problem that arises during camera movement is the movement of the marker projection between successive frames. In such a situation, analysis of the signal from a single stationary pixel utilizes only a portion of the pulsation information and can lead to degraded detection. This paper proposes a frequency-selective analysis of image-time trajectories, designated FSTRT. The method samples the image intensity along candidate trajectories, and for each of them determines the amplitude of the component corresponding to the known marker pulsation frequency. This allows for simultaneous detection of pulsation and estimation of the initial coordinate and slope of the apparent movement of the marker projection within the analyzed window. A search across the entire adopted parameter range and a local variant using trajectory prediction were considered. Simulation studies showed that for rotational and complex motion, FSTRT recovers approximately 98–100% of the amplitude available for the reference trajectory, while analysis of stationary pixels retains approximately 23–25%. Under high noise conditions, using the local variant of FSTRT increased the detection probability from 0.218 (for the baseline STATIC/STF method) to 0.950 with a comparable false alarm probability. The effects of exposure time, frequency mistuning, and trajectory prediction errors were also analyzed. The presented results concern a single marker, one-dimensional motion of its projection, and block analysis of a single image window. They do not include continuous tracking or estimation of the camera position. Instead, they represent a step towards developing a localization system based on pulsating markers capable of operating during camera motion.

Keywords: spectrum analysis, direction estimation, Radon transform, localization, image processing.

INTRODUCTION

Vision-based localization systems can use markers with known geometry or position as objects that facilitate scene element identification, camera position estimation, or the integration of semantic information with a map. Fiducial markers support the construction of maps and situational graphs, while mobile solutions use them to link the user's location with augmented reality

content [1,2]. However, three functions must be distinguished in this case: detecting the marker in the collected image data, determining its image coordinates, and determining the camera position based on the geometric model and known reference points. The presence of a marker does not necessarily mean solving the full localization task.

Active visual markers extend the above concept because the emitted signal can carry an identifying feature in the time domain. Cameras used

as receivers in optical camera communication (OCC) and visible light positioning (VLP) systems record LED modulation between captured frames or as stripes generated by a rolling shutter mode. Depending on the system, various LED signal modulation techniques are used, including ON-OFF Keying (OOK), Manchester coding, and variations of pulse position modulation (PPM) [3–5]. Once a transmitter is recognized, its image coordinates can be combined with a projection model, visual odometry or additional sensors to determine its position [5,6]. Therefore, modulation information increases the discrimination of the active tag, but its use still requires maintaining a sufficiently consistent temporal observation.

In the earlier spatial–time–frequency (STF) method proposed by the authors, a sequence of N frames is transformed into a set of temporal signals assigned to fixed image coordinates. For each pixel, the intensity change is analyzed, and the amplitude of the component corresponding to the pulse frequency creates an amplitude image in which a high response indicates the possible presence of a marker [7]. The authors' further work analyzed the influence of recording parameters, in particular exposure time and the use of such a representation in the concept of a positioning system. [8,9]. The basic STF construction assumes that the marker projection remains approximately on the same group of pixels within the window under consideration. This assumption becomes weak when significant relative motion of the camera and marker occurs during acquisition. If the marker projection moves between frames, a single stationary pixel records its pulsation only through a portion of the window. The energy of the marker frequency component is then distributed among signals corresponding to subsequent image positions and the response of the classical STF decreases. This phenomenon should not be associated solely with motion blur in a single frame. Blurring depends on motion during the exposure time, while loss of STF coherence also occurs for sharp frames if the marker projection passes through different pixels within the N images analyzed. From the STF perspective, what is needed is not so much to restore the sharpness of each frame, but to create a temporal signal along the correct trajectory in the image-time space.

In the literature on dynamic OCC and active beacons, several methods for mitigating the effects of motion are used. X. Sun et al. [10] combined LED array detection with learned

deblurring in vehicular OCC. Niarchou et al. [11] used motion information to limit the searched region of interest (ROI) for a wearable transmitter. Song et al. [12] presented a system for mobile IR LEDs in which OCC identification, marker detection, optical flow determination, and positioning were organized into separate stages. After identification, the LEDs remained continuously on during tracking. Optical flow estimation was implemented using the Lucas-Kanade algorithm, while positioning was based on the Perspective-point (PnP) algorithm.

Event cameras enable asynchronous processing of rapid intensity changes and are often used for OCC with moving markers. Wang et al. [13] demonstrated high-frequency beacon communication under the assumption of low motion. Von Arnim et al. [14] presented a combination of sparse optical flow, clustering, Kalman filtering, and parallel decoding of a pulsating beacon. All this is used to dynamically track a marker and estimate its motion trajectory. In [15], the goal is similar, but H. Su et al. applied periodicity detection, fast feature tracking of a dynamic marker and discrete Fourier transform (DFT) for the event signal in the ROI. Ebmer et al. [16] used frequency-sensitive active LEDs for fast 6-degrees-of-freedom (DoF) pose estimation using an event camera. The cited works confirm the validity of the marker motion trajectory estimation problem, but they use a different data representation and separate motion estimation from detection and decoding.

Motion compensation and optical flow cover much broader areas than OCC. Classical methods estimate a displacement field and then use it for prediction, warping or temporal filtering along motion paths. Their effectiveness depends on the quality of the derived vectors [17]. Modern models can estimate flow from multiple frames and propagate motion information over a longer time scale [18]. Frequency information has also been used to model the periodic evolution of the optical flow field throughout the sequence [19]. For event data, a continuous motion trajectory can also be embedded in a Gaussian process model and optimized for pattern consistency [20]. However, these solutions estimate the full or local motion field, a continuous pattern transformation or a separate tracker state. The problem considered in this paper is narrower: it concerns a single source with a known frequency, a short window, and a two-parameter model of the image trajectory. The trace of a point moving with constant velocity has

the form of a line in the space-time representation. The authors use the Radon transform in their work because it has already been used to detect such trajectories and estimate velocity [21]. In the proposed method, the Radon transform is used to parameterize motion hypotheses and the element that distinguishes it from the presented solutions is the use of the amplitude of the component with a known modulation frequency as a criterion for evaluating each trajectory.

However, the above groups of methods leave a specific gap in terms of STF for a pulsating marker captured with a frame-by-frame camera. OCC solutions typically detect or decode the transmitter in the ROI, after which the motion is handled by a separate tracker, visual odometer, or event sensor. General motion compensation methods first estimate the flow or transformation and then filter the compensated data. Classic Radon applications select the trajectory based on the sum of intensity, energy or shape consistency rather than the response to a known pulsation frequency. Based on the literature reviewed, no method for a classic frame-by-frame sequence has been identified that selects the parameters of the linear trajectory of a pulsating marker projection directly based on the amplitude of its known modulation frequency in a single window.

This paper proposes a frequency-selective trajectory radon transform (FSTRT) method, interpreted as a frequency-selective trajectory response function based on Radon parameterization. It extends the STF from a stationary pixel to a family of linear trajectories in the image-time space. In a simplified one-dimensional version, the hypothesis has the form:

$$x(k) = x_0 + ak \quad (1)$$

where: x_0 – the horizontal initial position of the projection, a – apparent displacement of the projection per frame.

For each pair (x_0, a) the intensity is interpolated and sampled in subsequent frames along the appropriate line, and the response to the hypothesis is the amplitude of the component with the marker pulse frequency. For $a = 0$ the construction reduces to the analysis of a stationary pixel used in STF. The Radon parameterization describes a family of linear trajectories, while the frequency criterion is derived from the earlier STF detector. The method does not replace general optical flow, but uses strong a priori information about one known source and a simple model of its motion in a short window.

The aim of this work is to verify whether sampling along a hypothesized trajectory allows for recovering the frequency response of a pulsating marker attenuated by its apparent motion in the image. The proposed response function for the parameters (x_0, a) is compared with classical fixed-pixel analysis, full and local hypothesis grid search in the ROI, and a reference variant using a real trajectory. The effects of exposure time, frequency mistuning, prediction errors, and deviations from the 1D motion model are assessed. The study concerns a single synthetic marker and a single window of N frames; it does not include continuous tracking, recursive state estimation, or camera pose determination.

Spatial-time-frequency analysis

A STF algorithm is used to detect pulsating visual markers in images captured by a camera. The visual markers should emit pulsed light with a known pulsation frequency. The camera captures a sequence of images, and the STF algorithm identifies pixels whose intensity level changes correspond to the pulsation frequency of the visual marker.

The main idea of the discussed processing method is to change the sequence of a two-dimensional signal (which is a set of recorded frames) into a set of one-dimensional signals of changes in the intensity level of pixels over time and find those pixels on which the pulsation frequency of one of the markers was captured. The algorithm reads the pixel's intensity level value from one of the three RGB recording layers for each frame. Then a signal of the change in intensity level over time for a single pixel is generated. The procedure is repeated for each pixel of the image sequence. Given an m -by- n image sequence consisting of k consecutive frames, the algorithm translates the generation of $m*n$ intensity level change signals over time. The number of signals in the set must match the number of pixels in a single image from the sequence.

In the next step, the algorithm performs a spectral analysis for each generated signal to find pixels with recorded traces of marker pulsations. The amplitude value for the pulsation frequency in each acquired spectrum is checked. This value is saved in a new image. The location of the stored value corresponds to the position of the pixel whose spectrum is currently being analyzed. By repeating the operation for each pixel, the

algorithm generates an image in which high pixel values indicate the detection of the desired marker pulsation while low values indicate background.

It is assumed that the visual markers are stationary. The localization problem concerns the camera position. The essence of the STF algorithm is to appropriately process the image sequence to produce an image of characteristic frequency amplitudes, which is then subjected to known image pattern search and localization techniques.

Marker blur problem in images

The time at which the marker was actually at a given location in the image is crucial for STF analysis. Particularly in the context of using the vision-based localization system with moving objects. In the basic approach, the temporal signal of light intensity for a specific point in the matrix is generated by combining the intensity values of the same pixel in subsequent image frames. This approach is valid when the relative motion between the camera and the visual marker is insignificant within the analyzed time window. The marker image is projected onto approximately the same group of pixels and its pulsation can be detected through spectral analysis of the corresponding temporal signals.

Figure 1 shows a pixel matrix representing a sequence of four consecutively recorded frames. This matrix is located in the image-time space

where the image is associated with the (x, y) coordinates of a given pixel and time is correlated with the frame number. The green dots represent pixels where the marker was captured, while the small black cuboids represent the remaining image pixels. The marker's position changes and its pulsations are captured on different pixels with each subsequent frame. The STF algorithm must process intensity level changes for several pixels to detect the marker in the image in the standard configuration. This is indicated by blue arrows. The red dashed arrow represents the marker's motion trajectory in the image which could be used for STF analysis and the detection of the marker's position in motion. The task is to correctly recognize this trajectory.

The position of the marker in the camera system changes due to both camera translation and rotation. Consequently, the marker's projection position in the image changes in subsequent frames and is captured over a larger pixel area than in the stationary case. The marker appears to move within the image and its position information becomes blurred. As a result, a single image pixel may contain information about the marker's pulsation only for a short portion of the analyzed sequence. This reduces the amplitude of the component corresponding to the marker's pulsation frequency and can lead to unstable or erroneous detection. Therefore, the analyzed signal should not generally be generated from the intensity

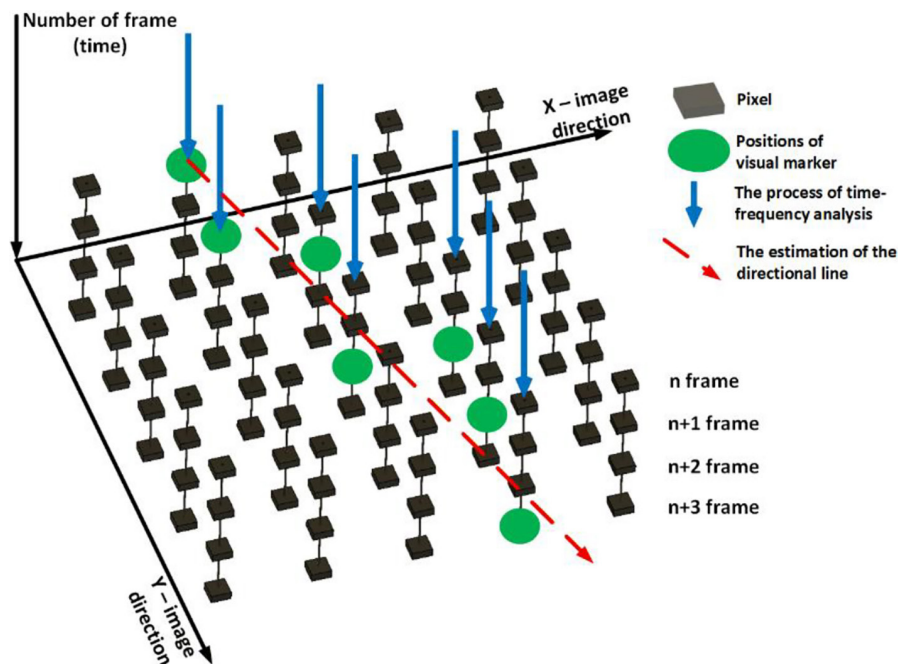


Figure 1. The concept of sampling along a trajectory in image-time space: STF signals for stationary pixels are shown in blue, and the marker trajectory used by FSTRT is shown in red dashed line

values of a single fixed image point (blue lines in Figure 1), but from values lying on the appropriate trajectory in the image-time space (red dashed line in Figure 1).

A simplifying assumption was made in this work. The marker's movement trajectory in subsequent frames will be simplified to a straight line. The Radon transform, combined with the STF algorithm, will be used to assess whether the intensity signal sampled along a given candidate trajectory contains a component corresponding to the known pulse frequency of the visual marker. It is necessary to develop a geometric-kinematic model to use the presented tools. This model takes into account camera motion and propose equations for projecting the desired motion trajectory onto the image.

GEOMETRIC-KINEMATIC MODEL OF THE APPARENT MOTION OF THE MARKER PROJECTION IN THE IMAGE

This chapter presents a simplified model describing the change in the position of a stationary visual marker's projection in the image of a camera moving relative to that marker. The goal of the model is to relate the physical position of the marker and the camera's motion to the trajectory observed in the image sequence. This model is necessary because the position of the marker's projection is not constant across successive image frames in the case of relative motion between the camera and the marker.

A visual marker with known coordinates in the world frame is considered. The following simplifying assumptions were made:

- the marker is stationary in the world frame,
- the camera is rigidly attached to the moving object (e.g. a mobile robot),
- the relative motion of the camera-marker system is planar,
- the center of the marker and the optical center of the camera are at the same height,
- the marker is treated as a point,
- the camera is described by a pinhole model [22],
- lens distortion is ignored or a distortion-corrected image is assumed,
- in a short time window, the marker's projection trajectory can be linearly approximated.

Assumptions that the center of the marker and the optical center of the camera are approximately at the same height, and that the relative movement of the camera-marker system occurs in the horizontal plane allows to ignore the vertical component of the marker position in the camera system. However, this is not a necessary assumption for the general method, but a simplification introduced to clearly derive the basic relationships and facilitate analysis of the results obtained later. In the general case, when the marker's center and the camera's optical center are not at the same height or when the camera performs a complex motion involving roll or tilt, the vertical image coordinate, and the full three-dimensional projection model must also be considered.

Let $\{W\}$ denote the world coordinate system and $\{C\}$ the camera coordinate system. The

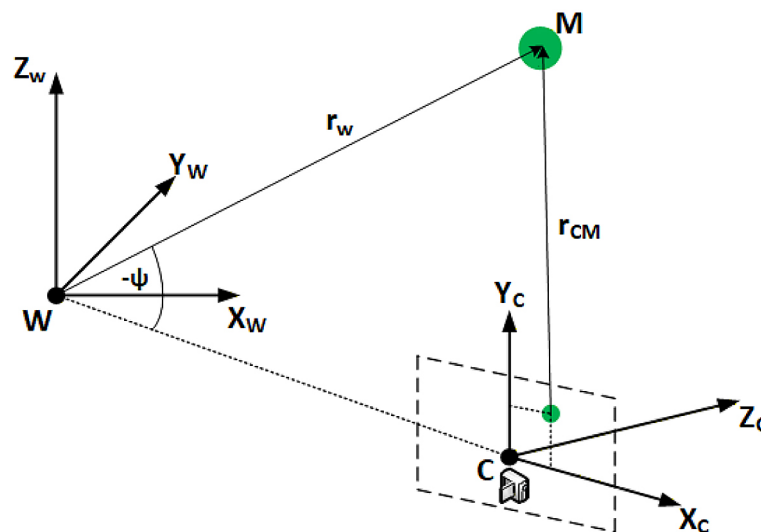


Figure 2. Schematic drawing of the world coordinate system, camera coordinate system and marker position

camera frame is defined so that the Z_C axis is aligned with the camera's optical axis, the X_C axis corresponds to the horizontal image direction, and the Y_C axis corresponds to the vertical image direction. The coordinates of a point in the image are denoted as (u, v) , where u is the horizontal and v is the vertical coordinate. The described coordinate systems are shown in Figure 2.

Robot motion

Motion in the horizontal plane is analyzed in the simplified planar model. It is convenient to assume that the X_W and Y_W describe the plane of motion while Z_W describes the height. The position of the center of the stationary visual marker M in the world frame is written as:

$$p_M^W = \begin{bmatrix} X_M^W \\ Y_M^W \\ Z_M^W \end{bmatrix}, \quad (2)$$

while the position of the camera's optical center is written as:

$$p_C^W = \begin{bmatrix} X_C^W \\ Y_C^W \\ Z_C^W \end{bmatrix}, \quad (3)$$

In the considered model, it is assumed that the center of the marker and the optical center of the camera are at the same height: $Z_M^W - Z_C^W \approx 0$. This allows further analysis to be limited to the motion in the $X_W - Y_W$ plane. First, the marker position vector relative to the camera is analyzed, limited to the horizontal plane:

$$r_W = \begin{bmatrix} X_M^W - X_C^W \\ Y_M^W - Y_C^W \end{bmatrix}. \quad (4)$$

Let ψ denote the camera's rotation angle relative to the world frame in the plane of motion (the deviation angle of the camera's optical axis). The marker's position vector relative to the camera expressed in the camera frame is obtained by applying the following planar coordinate transformation:

$$r_{CM}^C = \begin{bmatrix} X_M^C \\ Y_M^C \end{bmatrix} = T_{C \leftarrow W}(\psi) \begin{bmatrix} X_M^W - X_C^W \\ Y_M^W - Y_C^W \end{bmatrix}, \quad (5)$$

where:

$$T_{C \leftarrow W}(\psi) = \begin{bmatrix} -\sin \psi & \cos \psi \\ \cos \psi & \sin \psi \end{bmatrix}. \quad (6)$$

After expansion:

$$\begin{aligned} X_M^C &= -\sin \psi (X_M^W - X_C^W) + \cos \psi (Y_M^W - Y_C^W), \\ Y_M^C &= \cos \psi (X_M^W - X_C^W) + \sin \psi (Y_M^W - Y_C^W). \end{aligned} \quad (7)$$

The X_M^C describes the lateral position of the marker relative to the camera's optical axis, while Z_M^C describes the marker's depth (its distance along the optical axis). The marker can be registered by the camera only when it is in front of the camera, when and when its projection is inside the image area. The horizontal coordinate of the marker's projection on the image for a camera described by the pinhole model is given by the relationship:

$$u = f_x \frac{X_M^C}{Z_M^C} + c_x, \quad (8)$$

where: f_x – focal length expressed in pixels, c_x – horizontal coordinate of the image's main point.

The above model describes the instantaneous position of the marker projection in the image. The movement of the camera relative to the stationary marker must be considered in order to describe the change in this position in an image sequence. Let the camera move in the horizontal plane with a translational velocity expressed in the camera frame and with an angular velocity:

$$v_C^C = \begin{bmatrix} v_x^C \\ 0 \\ v_z^C \end{bmatrix}, \omega_C = \begin{bmatrix} 0 \\ \omega \\ 0 \end{bmatrix}. \quad (9)$$

where: ω_C – angular velocity of the camera around the vertical axis of the camera system.

The velocity vector of the marker relative to the camera $\dot{r}_{CM}^C$ is:

$$\dot{r}_{CM}^C = - \begin{bmatrix} v_x^C \\ 0 \\ v_z^C \end{bmatrix} - \begin{bmatrix} \omega Z_M^C \\ 0 \\ -\omega X_M^C \end{bmatrix}, \quad (10)$$

what can be represented as:

$$\begin{aligned} \dot{X}_M^C &= -v_x^C - \omega Z_M^C, \\ \dot{Z}_M^C &= -v_z^C + \omega X_M^C. \end{aligned} \quad (11)$$

These relations show that the position of the marker in the camera system changes due to both camera translation and camera rotation. In the case of translational motion alone, the terms dependent on ω disappear. In the case of camera rotation alone, the projection of the marker also

changes position in the image, even though the camera's optical center does not perform any translational motion. The apparent velocity of the marker projection in the image can be obtained by differentiating the projection Equation 8:

$$\dot{u} = f_x \frac{\dot{X}_M^C Z_M^C - X_M^C \dot{Z}_M^C}{(Z_M^C)^2}. \quad (12)$$

After substituting Equations 11 and arranging the components, Equation 12 can be reduced to the form:

$$\dot{u} = f_x \left(-\frac{v_x^C}{Z_M^C} + \frac{X_M^C v_z^C}{(Z_M^C)^2} - \omega \left(1 + \frac{(X_M^C)^2}{(Z_M^C)^2} \right) \right). \quad (13)$$

The resulting relationship has an important interpretation. The apparent velocity of the marker projection in the image is not equal to the physical velocity of the camera. It depends on the lateral component of the camera velocity v_x^C , the component along the optical axis v_z^C , the angular velocity ω , the marker depth Z_M^C , the marker's lateral position X_M^C , and the camera's focal length f_x . The same camera movement can therefore cause different apparent velocities in the image for markers located at different distances and at different angles to the optical axis.

Linear trajectory of the marker movement

The marker projection position can be approximated linearly in a short time window. For the horizontal coordinate, this can be written as:

$$u(t) \approx u_0 + \dot{u}(t - t_0), \quad (14)$$

where: $u_0 = u(t_0)$ – position of the marker projection at the initial moment of the analyzed time window.

The resulting local trajectory of the marker projection in the image can be written as:

$$x(t) = x_0 + a_t(t - t_0), \quad (15)$$

where: $x_0 = u_0$ – starting point of the local trajectory, $a_t \approx \dot{u}$ – marker movement speed along the local trajectory.

In a practical implementation, the image sequence is discrete. If the camera records images at a frequency f_s , then the time interval Δt between successive frames is:

$$\Delta t = \frac{1}{f_s}. \quad (16)$$

For a window covering N consecutive frames, the recording moments can be written as:

$$t_n = t_0 + n\Delta t, \quad (17)$$

where: $n = 0, 1, \dots, N-1$ – the number of the next frame of the sequence of recorded images.

Substituting Equations 16 and 17 into 15 gives the discrete form of the linear trajectory model:

$$x_n = x_0 + a_t n\Delta t = x_0 + \frac{a_t}{f_s} n. \quad (18)$$

This means that the offset of the marker projection between two consecutive frames is:

$$\Delta x = x_{n+1} - x_n = \frac{a_t}{f_s} = a. \quad (19)$$

Based on (19), the parameter a_t can be written as:

$$a_t = f_s \Delta x. \quad a = \frac{a_t}{f_s} \quad (20)$$

For the entire analyzed window, the total shift of the marker projection D_x between the first and last frame is:

$$D_x = x_{N-1} - x_0 = (N - 1)\Delta x = \frac{a_t}{f_s} (N - 1). \quad (21)$$

This relationship shows that the parameter a_t can be interpreted as the apparent velocity of the marker projection in the image, expressed in pixels per second while a can be interpreted as the corresponding inter-frame displacement between successive frames, expressed in pixels per frame.

The presented model shows that the marker projection trajectory in an image sequence results from the physical movement of the camera relative to the stationary marker, but is not identical to this movement. The trajectory parameters in the image depend on camera translation, camera rotation, marker distance, its position relative to the optical axis, and internal camera parameters. In a short time window, this trajectory can be approximated by a line in the image-time space. This means that the intensity signal used to detect marker pulsations should not be generated solely from the values of a single fixed pixel, but from values sampled along the trajectory corresponding to the apparent movement of the marker projection.

Let the image sequence be represented by a continuous intensity function:

$$I(x, y, t) \quad (22)$$

where: (x, y) – image coordinates, t – time.

If the relative motion between the camera and the marker can be approximated by uniform motion over a short analysis window, then the marker's projection in (x, y, t) space approximates a linear trajectory. Instead of analyzing intensity changes in fixed image coordinates, a temporal signal can be created by sampling a sequence of images along a candidate straight line in the spatial-time space. Given Equation 15 and the assumption that the marker's projection motion can initially be analyzed primarily along the horizontal image coordinate, the candidate marker trajectory can be described by the following relations:

$$x(t) = x_0 + a_t(t - t_0) \quad (23)$$

where: x_0 – initial position of the marker projection in the image for the moment t_0 , a_t – slope of the trajectory in the x axis direction with respect to time.

The above parameters should be interpreted as parameters of the apparent motion of the marker projection in the image sequence. In the simplified one-dimensional case, when the analysis is mainly focused on the motion along the horizontal coordinate of the image, the description can be reduced to a vector:

$$p_{RF} = [x_0 \quad a_t]^T. \quad (24)$$

The mathematical considerations presented here show that camera motion cannot be directly translated into marker motion in the image. In a short time window, the marker position model can be reduced to a sequence of images with the marker's position linearly changing in subsequent frames. The marker's motion in the image can be approximated by a straight-line trajectory, which in the simplified planar case reduces to two parameters: x_0 and a_t . The FSTRT was used to find these parameters for a trajectory containing a component corresponding to the known pulsation frequency of the visual marker.

FSTRT FORMULATION IN IMAGE–TIME SPACE

The Radon transform is used to analyze spatial structures by integrating functions along lines.

In the two-dimensional case, the Radon transform of the function can be written as [23]:

$$R_f(\rho, \theta) = \iint f(x, y) \delta(x \cos \theta + y \sin \theta - \rho) dx dy \quad (25)$$

where: δ – Dirac delta, (ρ, θ) – parameters of the line along which integration takes place.

The Dirac delta component selects points lying on a line defined by the parameters (ρ, θ) and the integral sums the values of the analyzed function along this line.

Using the Fourier kernel and the candidate trajectory Equation 23, the proposed frequency-selective projection based on the Radon transform can be written as:

$$A(x_0, a_t, \omega_0) = \left| \iint I(x, t) \delta(x - x_0 - a_t(t - t_0)) e^{-j\omega_0 t} dx dt \right| \quad (26)$$

where: ω_0 – known angular pulsation of the visual marker.

The Dirac delta selects points satisfying condition (23). In the two-dimensional case, a second Dirac delta would need to be added to check the linear approximation condition for the y direction. The Fourier kernel $e^{-j\omega_0 t}$ allows to check whether the intensity signal sampled along this line contains a component corresponding to the expected marker pulsation frequency. The modulus of the integral determines the response amplitude for a given candidate trajectory.

The above notation can be interpreted as a one-dimensional Fourier analysis performed along the candidate trajectory:

$$A(x_0, a_t, \omega_0) = \left| \int I(x_0 + a_t(t - t_0), t) e^{-j\omega_0 t} dt \right| \quad (27)$$

The proposed transform does not simply search for a spatial line with a high intensity value. Its goal is to find a trajectory in the image-time space along which the intensity signal contains a frequency component corresponding to the known marker pulsation frequency. Optimal trajectory parameters can be determined by maximizing the response amplitude:

$$(\hat{x}_0, \hat{a}_t) = \arg \max_{x_0, a_t} A(x_0, a_t, \omega_0) \quad (28)$$

The maximum value $A(x_0, a_t, \omega_0)$ indicates the trajectory for which the marker pulsation is most strongly represented in the analyzed image sequence. The proposed transform allows for

simultaneous detection of marker pulsation and estimation of its apparent motion direction in the image. In the proposed approach, the FSTRT should be interpreted as a response function defined in the parameter space of the candidate trajectory.

Simulation research

The goal of the simulation study was to verify whether sampling the signal along the estimated trajectory of the apparent marker motion allows for the recovery of the pulse-frequency component, which in classical analysis of a stationary pixel is attenuated by image motion. The scope of the study was limited to five issues directly related to the method's essence: demonstration of the compensation mechanism, evaluation of the recovered amplitude relative to the ORACLE reference trajectory, the influence of acquisition time parameters, statistical evaluation of detection, and the robustness of local search to odometric errors.

Three repeated camera motion scenarios were considered: translation, rotation, and complex motion. The camera recorded sequences at a frequency of 60 Hz and the marker pulse frequency was 20 Hz. The one-dimensional horizontal motion of the marker projection was analyzed. A full 2D model, multiple markers, and real-world data remain the subject of further research.

The camera is rigidly attached to the robot moving in the horizontal plane. The robot's state at time k can be described as:

$$x_k = [X_k \ Y_k \ \psi_k]^T, \quad (29)$$

where: (X_k, Y_k) – the position of the robot or the center of the camera in the world coordinate system, ψ_k – camera orientation angle relative to this system.

If the robot has odometry in the form of path increment Δs_k and angle increment $\Delta \psi_k$, a simple prediction model can be written as:

$$\begin{aligned} X_{k+1} &= X_k + \Delta s_k \cos \psi_k, \\ Y_{k+1} &= Y_k + \Delta s_k \sin \psi_k, \\ \psi_{k+1} &= \psi_k + \Delta \psi_k. \end{aligned} \quad (30)$$

In compact notation this model can be represented as:

$$\begin{aligned} x_{k+1} &= f(x_k, u_k) + w_k, \\ u_k &= [\Delta s_k \ \Delta \psi_k]^T \end{aligned} \quad (31)$$

where: u_k – odometric input vector, w_k – motion model disturbances and odometry uncertainty (assumed $w_k = 0$, uncertainty introduced solely by odometry and prediction errors).

After making the prediction, the estimate and the covariance matrix are equal:

$$\begin{aligned} \hat{x}_{k|k-1} \\ P_{k|k-1}. \end{aligned} \quad (32)$$

The matrix $P_{k|k-1}$ contains information about the uncertainty of the robot's predicted position and orientation. This uncertainty should influence the range of parameters for which the FSTRT response is calculated in future. The greater the uncertainty in the odometry or motion model, the larger the range of candidate trajectories should be. Otherwise, there is a risk that the actual trajectory of the marker projection will fall outside the analyzed parameter space.

For a marker with a known position in the P_M world frame, the geometric-kinematic model and the camera projection model allow to predict the parameters of its projection trajectory in the image. In general, this can be written as:

$$\hat{p}_{RF,k} = h(\hat{x}_{k|k-1}, P_M), \quad (33)$$

where: $h(\cdot)$ – a transition function from the predicted state of the robot and the known position of the marker to the parameters of the apparent trajectory of the marker projection in the image.

In the full variant of movement in the vertical and horizontal directions, it can be assumed that:

$$\hat{p}_{RF,k} = [\hat{x}_0 \ \hat{y}_0 \ \hat{a} \ \hat{b}]^T \quad (34)$$

while in the one-dimensional variant:

$$\hat{p}_{RF,k} = [\hat{x}_0 \ \hat{a}]^T. \quad (35)$$

Robot state estimation was not the subject of the presented research. Each sequence of N frames was analyzed independently without recursive propagation or correction of the robot's position. In the FSTRT-ODOM variant, the external prediction of trajectory parameters (x_0, a) was used solely to determine the center of the local search area. No filter estimating the robot's position and orientation was implemented, and the covariance matrix was not updated based on the FSTRT response. The most important simulation parameters and their assumed values are presented in Table 1.

The robot's motion is generated based on the translational velocity V and the angular velocity ψ . The simulator generates a ground-truth trajectory, the marker's projection onto the image, a sequence of grayscale images, and a noisy odometry. The marker is modeled as a Gaussian spot

with sinusoidal intensity varying at a frequency of $f_0 = 20$ Hz. In the current simulation variant, the image model with the marker takes into account the basic perspective effect. The simulator results are presented in Figures 3 and 4.

Figure 3 shows an example robot trajectory in the world frame and the successive positions of the marker's center in the camera coordinate system. Figure 4 shows sample frames from generated grayscale images with the marker in motion. The marker's position in the world coordinate system was constant at $P_M = [5.0; 0.75; 1.0]$.

In a short window of N frames, the horizontal trajectory of the marker projection was approximated by a linear model:

$$u_k = x_0 + ak, k = 0, \dots, N - 1 \quad (36)$$

where: x_0 – the initial position of the projection, a – apparent velocity of the projection expressed in pixels per frame.

The signal sampled along the trajectory has the form after image interpolation:

$$s_k(x_0, a) = I_k(x_0 + a_k) \quad (37)$$

Table 1. Key camera, marker, motion, noise, and evaluation parameters used in the simulations

Group	Parameter	Value used in the simulations	Notes
Image and camera	Image format and sampling	320×240 px; $f_s = 60$ Hz; $T_s = 16.667$ ms	Double-precision intensities normalized to $[0, 1]$. Nominal quantization was disabled; the realistic-noise variant used 8-bit quantization.
Image and camera	Intrinsic parameters	$K = [260, 0, 160.5; 0, 260, 120.5; 0, 0, 1]$	$f_x = f_y = 260$ px, $c_x = 160.5$ px, $c_y = 120.5$ px, zero skew. Lens distortion was not modelled.
Image and camera	Shutter and exposure	Ideal global shutter; $T_{exp} = 16.667$ ms ($\rho = 1$)	The principal Monte Carlo tests used $\rho = 0.5$ (8.333 ms). The exposure sweep used $\rho = \{0, 0.1, 0.25, 0.5, 0.75, 1\}$ and 15 temporal samples per exposure.
Sequence	Analysis-window length	$N = 30, 60, 90$, and 120 frames	Window durations were 0.5, 1.0, 1.5, and 2.0 s. The principal Monte Carlo experiment used $N = 60$ and a rectangular temporal window.
Marker	Position and spatial profile	$P_M = [5.0, 0.75, 1.0]^T$ m; stationary 2-D Gaussian marker	The 1-D analysis used the horizontal cross-section. $r_M = 0.03$ m and $\sigma_{px} = f_x r_M / Z$, constrained to 0.8–12 px.
Marker	Effective image size	$FWHM$ approximately 3.65–3.79 px	For $N = 60$: 3.729 px (translation), 3.658 px (rotation), and 3.712 px (combined motion).
Marker	Temporal modulation	$f_0 = 20$ Hz; $\varphi_0 = 0$; sinusoidal waveform	$I(t) = 0.45[1 + 0.70 \sin(2\pi f_0 t + \varphi_0)]$. The modulation amplitude was 0.315 and the marker-intensity range was 0.135–0.765 before background, noise, and clipping.
Background	Background model	Linear gradient; $I_{bg,0} = 0.08$; $g_x = 0.03$; $g_y = -0.01$	The full 2-D background ranged from 0.06 to 0.10. The 1-D simulations used the horizontal gradient only, giving 0.065–0.095.
Noise	Nominal image noise	Independent Gaussian read noise $N(0, 0.015^2)$	Shot noise and quantization were disabled in the nominal variant; samples were clipped to $[0, 1]$.
Noise	Sensitivity and realistic-noise variants	$\sigma_{read} = \{0.005, 0.015, 0.25, 0.50\}$; realistic: $\sigma_{read} = 0.02$, $c_{shot} = 0.002$	The realistic variant used $\sqrt{I c_{shot}} N(0, 1)$, clipping, and 8-bit quantization. This was a Gaussian approximation, not an exact Poisson model.
Motion	Motion scenarios	Translation: $V = 0.15$ m/s; rotation: $\omega = 4$ deg/s; combined: both	The translation-only and rotation-only cases used $\omega = 0$ and $V = 0$, respectively.
Odometry	Increment errors and biases	$\sigma_{\Delta s} = 0.0015$ m/frame; $\sigma_{\Delta \psi} = 0.05$ deg/frame; $b_s = 0.002$; $b_\psi = -0.001$	$R = \text{diag}(2.25e-6, 7.615e-7)$, with diagonal units m^2 and rad^2 . The ground-truth trajectory was deterministic; no separate process noise w_k or covariance Q was generated.
Trajectory prediction	Prediction-error distribution	$[x_0, a]^T$ follows a zero-mean bivariate Gaussian distribution	$\sigma_{x_0} = 3$ px, $\sigma_a = 0.08$ px/frame, $\text{corr}(x_0, a) = -0.25$, and $\sigma_{x_0, a} = [9, -0.06; -0.06, 0.0064]$.
Trajectory search	Search range and grid	x_0 : truth ± 10 px; a : truth ± 0.2 px/frame	Tested $\Delta x_0 = \{0.25, 0.5, 1, 2\}$ px and endpoint displacement steps $\Delta D = \{0.25, 0.5, 1, 2\}$ px, where $\Delta a = \Delta D / (N-1)$. This produced 132 - 7695 candidate trajectories for $N = 60$.
Trajectory sampling	Interpolation and boundaries	Linear interpolation; reject boundary policy	Candidate trajectories containing samples outside the image domain were rejected.
Robustness tests	Frequency, phase and exposure	True frequencies 20.0 and 19.7 Hz; phase varied separately	These tests addressed frequency mismatch, spectral leakage, phase sensitivity, aliasing considerations, and exposure-time averaging.
Monte Carlo	Principal and auxiliary experiments	Principal: 500 threshold H_0 + 1000 test H_0 + 1000 H_1 ; auxiliary: 100 + 200 + 200	Independent H_0 threshold and test sets were used. The principal counts applied to nominal combined motion and the high-noise condition.

The amplitude of the response for frequency f_0 was determined as:

$$A(x_0, a; f_0) = \frac{|\sum_{k=0}^{N-1} w_k (s_k - \bar{s}) e^{-j2\pi f_0 t_k}|}{\sum_{k=0}^{N-1} w_k} \quad (38)$$

where: w_k – the time window coefficient, t_k – the timestamp of the k -frame, $\bar{s}$ – the weighted average of the signal.

In the basic research, a rectangular window was used. Four processing variants were compared, listed in Table 2.

The quotient of the amplitude of a given variant and the STATIC amplitude describes the gain compared to the classic analysis of a stationary pixel:

$$G_{m/STATIC} = \frac{A_m}{A_{STATIC}} \quad (39)$$

However, this value depends on the specific motion, window length, marker image size, exposure, and visibility within the camera field. Therefore, it is not considered a fixed FSTRT characteristic. The basic measure of compensation quality is the relative efficiency of amplitude recovery:

$$\eta_m = \frac{A_m}{A_{ORACLE}} \quad (40)$$

For a trajectory perfectly matching the actual trajectory, $\eta_m \approx 1$ is expected. However, ORACLE is not a strict maximum in a single noisy realization: maximizing over multiple trajectories may, due to noise, return a value slightly larger than the amplitude for the actual trajectory. Therefore, a small excess of unity does not necessarily mean a physically better result than ORACLE. It is useful to consider the total displacement of the projection in the window relative to the width of the marker image in order to describe the magnitude of the movement:

$$D_{norm} = \frac{|a|(N-1)}{FWHM_u} \quad (41)$$

For $D_{norm} \ll 1$, the signal remains approximately within the same pixel region and the advantage of FSTRT over STATIC is small. Increasing D_{norm} increases the importance of compensation, but does not lead to a universal saturation gain because the result also depends on field-of-view constraints, exposure, and the trajectory model.

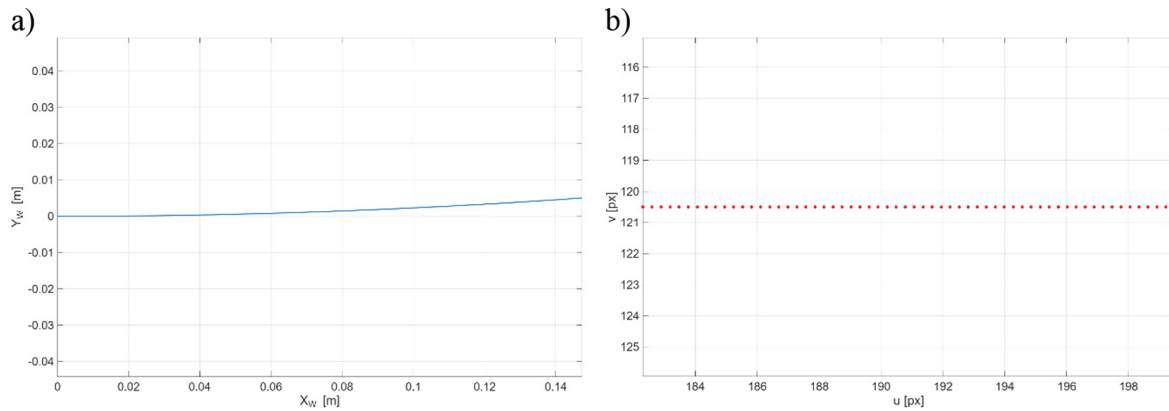


Figure 3. a) Simulated trajectory of a robot with a camera moving in complex motion; b) consecutive positions of the marker center in the camera system generated for the complex motion

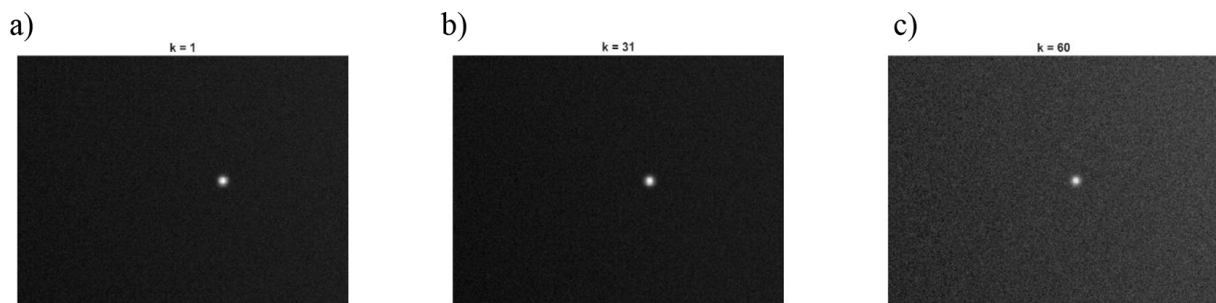


Figure 4. Example frames with a recorded marker: a) beginning, b) middle, c) end of the image sequence. The apparently increased graininess in panel (c) results from independent automatic scaling of the intensity range used to display each frame and not from a change in the simulated noise

Table 2. Definitions of the compared processing variants

Variant	Trajectory	Search area	Role in the study
STATIC/STF	$a = 0$	Maximum after x_0	Classic reference point
ORACLE	real u_k	One known trajectory	Reference of recoverable amplitude
FSTRT-GLOBAL	Estimated	Global grid (x_0, a)	Variant without motion prior
FSTRT-ODOM	Estimated	Local grid around odometry	Practical local variant

Linear interpolation was used to determine intensity values for non-integer trajectory coordinates: $x(k) = x_0 + ak$. This interpolation can cause a slight smoothing of the sampled marker profile and associated amplitude attenuation. However, it was applied consistently across all variants requiring subpixel sampling, and did not alter the qualitative conclusions from the comparisons. The trajectory moving outside the image was not a separate subject of research. Therefore, the motion scenarios and search ranges were selected so that the analyzed trajectories remained in the range $1 \leq x \leq 320$ px. The implementation included an additional safeguard: a candidate for which at least one sample was outside the image was rejected, given a value of $-\infty$ and could not be selected as a solution.

In the main Monte Carlo experiments, the FSTRT-GLOBAL variant searched $x_0 = x_{0,c} + [-20, 20]$ px with a step size of $\Delta x_0 = 1$ px and $a \in [-0.70, 0.70]$ px/frame with a step size $\Delta a = 0.05$ px/frame, corresponding to $N_{traj} = 41 \cdot 29 = 1189$ trajectories. The symbol $x_{0,c}$ denotes the nominal starting position defined for a given scenario. The term GLOBAL denotes a search of the entire assumed region of interest without using odometric prediction, rather than all image columns. The FSTRT-ODOM variant used a local grid around the prediction ($x_{0,odom}, a_{odom}$): $x_0 = x_{0,odom} + [-8, 8]$ px with a step size $\Delta x_0 = 0.5$ px and $a = a_{odom} + [-0.25, 0.25]$ px/frame with the step size $\Delta a = 0.025$ px/frame. This grid encompassed $N_{traj} = 33 \cdot 21 = 693$ trajectories. In both variants, a rectangular time window was used and the amplitude of the Fourier component was calculated for $f_0 = 20$ Hz.

Time measurements were performed for complex motion and windows of $N \in \{30, 60, 120\}$ frames. For FSTRT-GLOBAL, meshes covering $N_{traj} \in \{231, 861, 3321\}$ trajectories were used, while for FSTRT-ODOM $N_{traj} \in \{25, 50, 200\}$ trajectories. Each measurement was preceded by two warm-up realizations, after which 12 iterations were performed. The calculations were

performed in MATLAB R2025b Update 2, under Windows 11 Pro, on a computer with an Intel Core i7-8700 3.20 GHz processor, 15.8 GB of RAM and Intel UHD Graphics 630. The code did not use explicit GPU acceleration.

Marker projection linearity test

The first test examined whether the marker projection in a short time window could be described by a linear model. This test validates the basic assumption of the single-dimensional FSTRT variant. For the translation scenario, a very small fit error was obtained because the marker projection moved slowly. In the rotation and complex motion scenarios, the fit error was larger, but the linear model remained usable in the short time window. The results of the linear approximation check are presented in Table 3 along with an example in Figure 5.

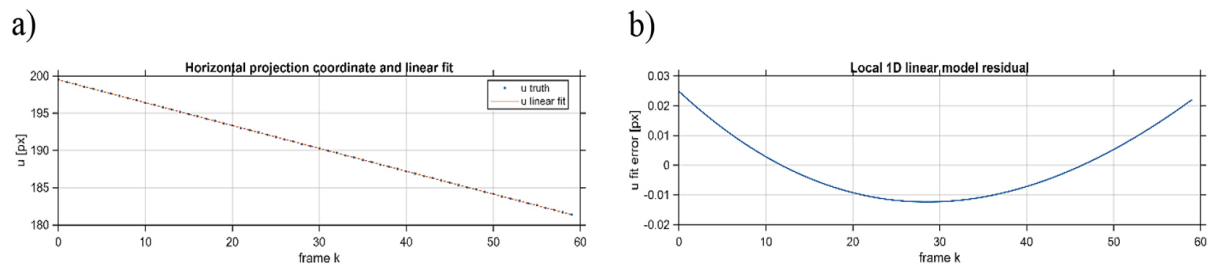
The actual trajectory was approximated using the least squares method using the linear model $\hat{u}_k = x_0 + a \cdot k$. the residual was defined as $r_k = u_k - \hat{u}_k$. The RMSE, the maximum absolute value of the residual, and the coefficient of determination R^2 were used as measures of the quality of fit. Errors were also related to the average FWHM of the marker image within the analyzed window.

For all scenarios, the R^2 exceeded 0.99974. The largest error occurred for complex motion and a window of 120 frames: the RMSE was 0.048 px (1.28% of the marker's FWHM), while the maximum residual was 0.117 px (3.11% of the marker's FWHM). The results confirm that under the conditions considered, the linear approximation is very accurate, even for the longest analyzed window.

The interpretation of this test is important. If the linear approximation error increases, it means that the analysis window is too long or the camera motion is too nonlinear. In this case, the time window should be shortened or the trajectory model should be extended with a curvature component.

Table 3. Evaluation of the accuracy of linear approximation of the marker projection trajectory

Motion scenario	N [frames]	FWHM [px]	RMSE [px]	Max. residual [px]	RMSE/FWHM [%]	Max. residual /FWHM [%]	R ²
Translation	30	3.700	0.00067	0.00135	0.0180	0.0366	0.9999848
Translation	60	3.729	0.00273	0.00587	0.0733	0.1573	0.9999383
Translation	90	3.758	0.00630	0.01381	0.1676	0.3674	0.9998588
Translation	120	3.787	0.01147	0.02546	0.3027	0.6723	0.9997450
Rotation	30	3.665	0.00318	0.00660	0.0869	0.1801	0.9999986
Rotation	60	3.658	0.01103	0.02483	0.3015	0.6789	0.9999957
Rotation	90	3.652	0.02102	0.05012	0.5755	1.3725	0.9999930
Rotation	120	3.647	0.03077	0.07797	0.8437	2.1377	0.9999915
Complex movement	30	3.692	0.00166	0.00350	0.0448	0.0948	0.9999996
Complex movement	60	3.712	0.00836	0.01915	0.2253	0.5158	0.9999973
Complex movement	90	3.735	0.02284	0.05427	0.6116	1.4531	0.9999910
Complex movement	120	3.759	0.04801	0.11678	1.2772	3.1068	0.9999778


Figure 5. a) The reference trajectory of the marker's horizontal projection coordinate for the complex motion scenario. b) The linear least-squares approximation for the complex motion scenario

Fixed pixel and trajectory sampling comparison test

In the next test, the intensity change signal of a single stationary pixel, the signal sampled along the correct trajectory and the signal corresponding to the mismatched trajectory were compared. The time signals and spectra are shown in Figure 6.

The results showed that the benefit of FSTRT increases with increasing the apparent marker velocity in the image. In the case of translation, the marker projection moved slowly, so classical analysis of a fixed pixel did not lose a significant portion of the pulsation energy. For rotation and complex motion, the marker moved faster across the image so the fixed pixel contained pulsation information only for a portion of the time window. Trajectory sampling allowed for the recovery of the strong frequency component.

Amplitude recovery efficiency in relation to ORACLE

Table 4 presents the results of individual, deterministic executions of three traffic scenarios. The gains relative to STATIC serve only to illustrate the scale of the effect under these conditions. Conclusions regarding the quality of the compensation itself are based on the η values relative to ORACLE.

In the translation scenario, STATIC retained 94.2% of the ORACLE response. This means that in this particular implementation the displacement of the projection was too small for the motion compensation to result in a significant increase in amplitude. For rotational motion, STATIC's efficiency dropped to 23.3%, while FSTRT-GLOBAL and FSTRT-ODOM recovered 98.5% and 99.3% of the ORACLE amplitude, respectively. For complex motion, STATIC achieved 24.8%, and both FSTRT variants achieved a

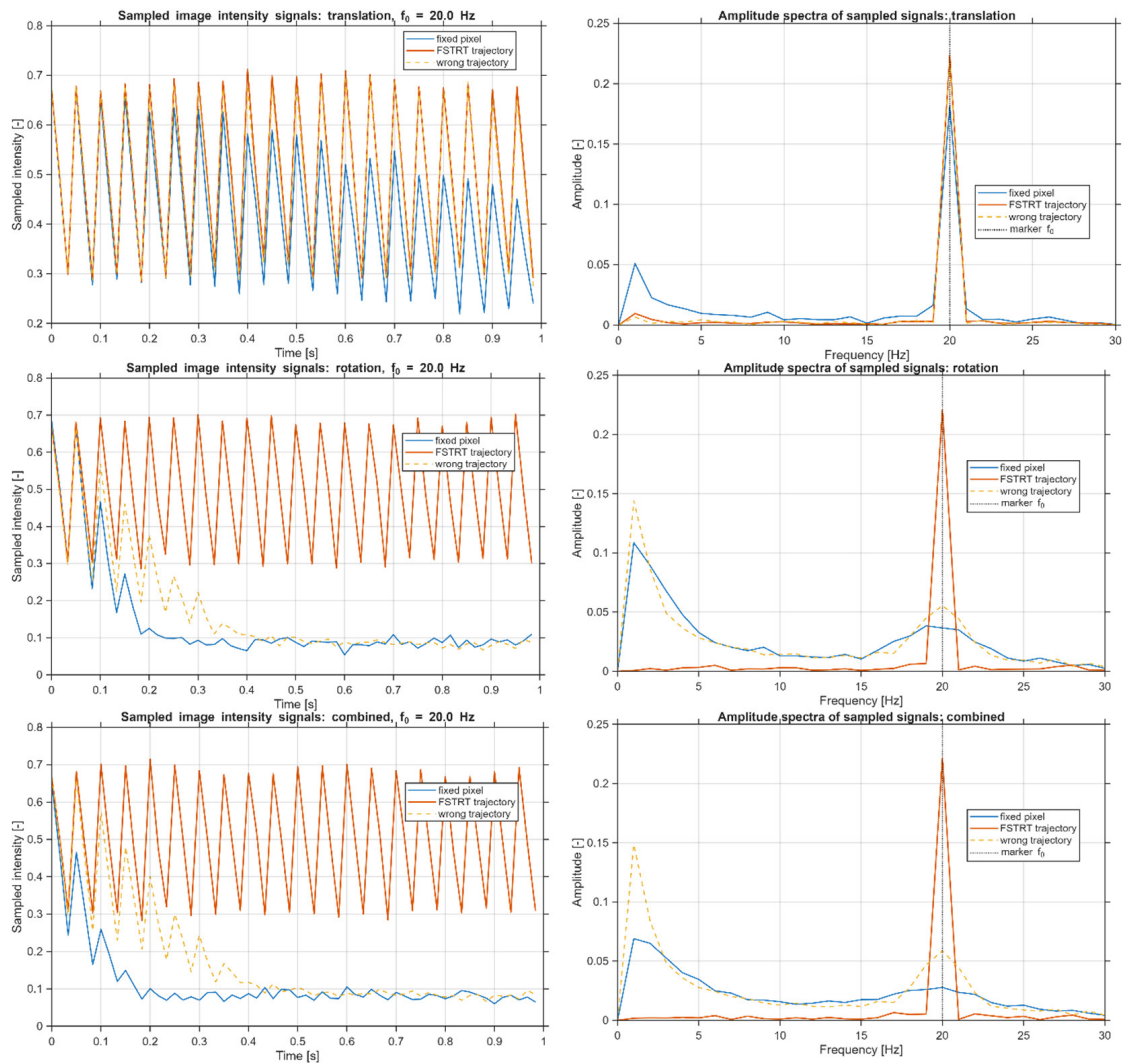


Figure 6. Comparison of intensity signals sampled at a fixed pixel, along the correct FSTRT trajectory and along the incorrect trajectory for three motion scenarios. The signal waveforms are on the left and the amplitude spectra are on the right

response practically equal to the reference. Gains of approximately 4.0–4.3 for rotation and complex motion are not universal parameters of the method. These are point scores for the assumed apparent velocity, window length, marker image size, and exposure time. The most important conclusion is that under conditions of significant motion, the practical FSTRT variants retained about 98–100% of the amplitude available for the actual trajectory, while STATIC lost about three-quarters of this response.

Motion-compensated amplitude map test

The next simulation compared amplitude images as the apparent marker's motion trajectory moved further and further away from the reference trajectory. For this purpose, an amplitude

image was generated for the reference trajectory (x_{0ref}, a_{ref}) , and two amplitude images were generated for the cases $a = 0$ and $a = 0.5 \cdot a_{ref}$. The first variant corresponds to the classic fixed-pixel analysis, while the second variant represents partial motion compensation. For translation, the amplitude images are shown in Figure 7, and the corresponding horizontal cross-section for the u coordinate is shown in Figure 8. For rotation, similar results are presented in Figures 9 and 10.

Amplitude maps demonstrate the mechanism of motion compensation. In the case of $a = 0$ corresponding to classical STF analysis performed for stationary pixels, the marker response is spread along the direction of its apparent motion. This is due to the fact that in subsequent frames the energy of the pulse signal reaches different image pixels. Applying partial compensation, $a = 0.5a_{ref}$

Table 4. Amplitudes, efficiency relative to Oracle and sample gain relative to Static for representative scenarios

Scenario	A_{ST}	A_{OR}	A_{GL}	A_{OD}	η_{ST}	η_G	η_O	G_G/ST	G_O/ST
Translation	0.118	0.126	0.119	0.126	0.942	0.950	1.006	1.008	1.068
Rotation	0.029	0.127	0.125	0.126	0.233	0.985	0.993	4.227	4.261
Complex Movement	0.031	0.126	0.126	0.126	0.248	1.005	1.001	4.050	4.035

Note: ST – Static/STF, OR – Oracle, OD – ODOM, GL – Global, G – FSTRT-Global, O – FSTRT-ODOM.

reduces the spatial blurring of the response, while sampling the intensity along a trajectory consistent with the reference motion, $a = a_{ref}$, results in coherent summation of the frequency component and a concentration of the peak near the marker's actual position.

The cross-section of the amplitude maps shown in Figure 10 confirms the gradual increase and narrowing of the response maximum with improved fit of parameter a . For $a = 0$, the tracer pulse information is distributed across pixels with coordinates u ranging from approximately 180 to 200. In the classical STF method, this

would require separate analysis of signals from multiple stationary pixels. The FSTRT method accumulates information from successive tracer positions along a single hypothetical trajectory. Therefore, compensation does not increase the tracer energy, but limits its spatial dispersion caused by image motion.

For the rotation scenario, the ratio of the maximum map obtained for $a = a_{ref}$ to the maximum for $a = 0$ was $G_{map} = 4.176$, whereas for the complex motion $G_{map} = 3.921$. In the case of translation, the gain remained close to unity because the displacement of the marker projection was small.

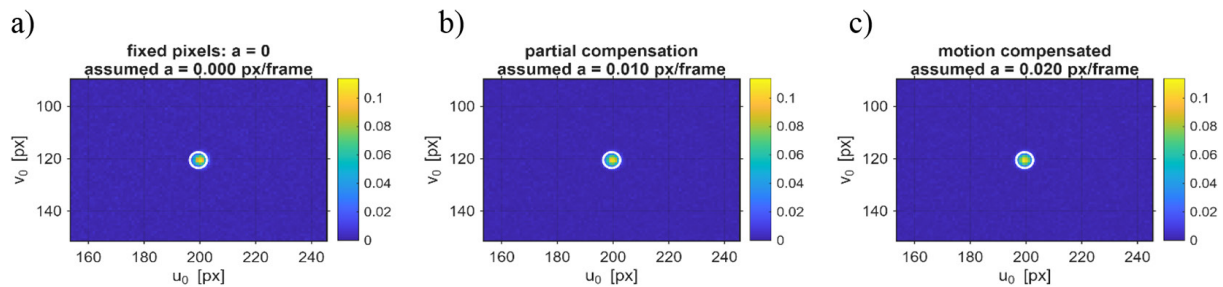


Figure 7. Amplitude images for the translation scenario: a) classic fixed-pixel analysis, b) partial motion compensation, c) compensation consistent with the FSTRT trajectory

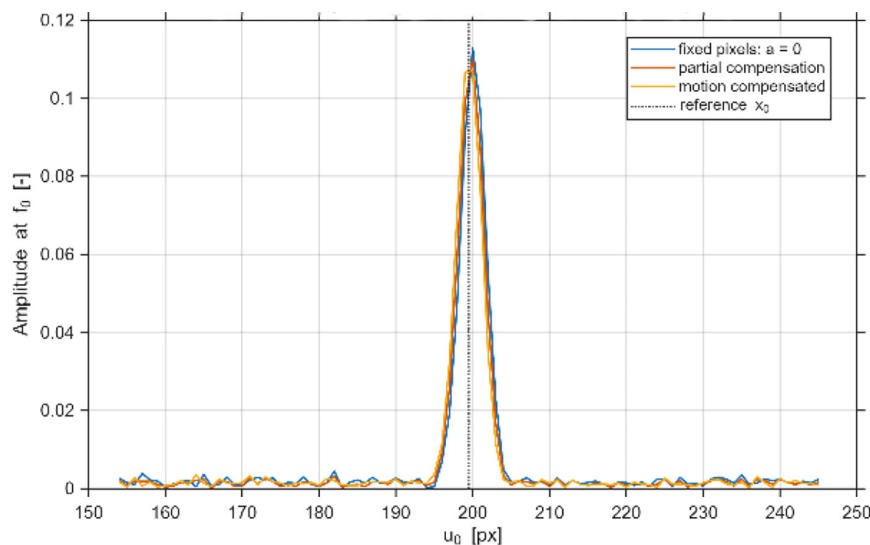


Figure 8. Horizontal amplitude profiles corresponding to the variants cases $a = 0$, $a = 0.5 \cdot a_{ref}$ and $a = a_{ref}$ for the translation scenario

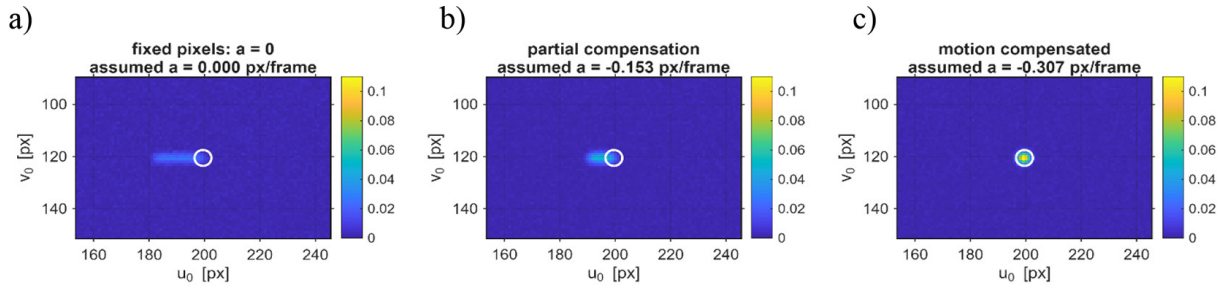


Figure 9. Amplitude images for the rotation scenario: a) classic fixed-pixel analysis, b) partial motion compensation, c) compensation consistent with the FSTRT trajectory

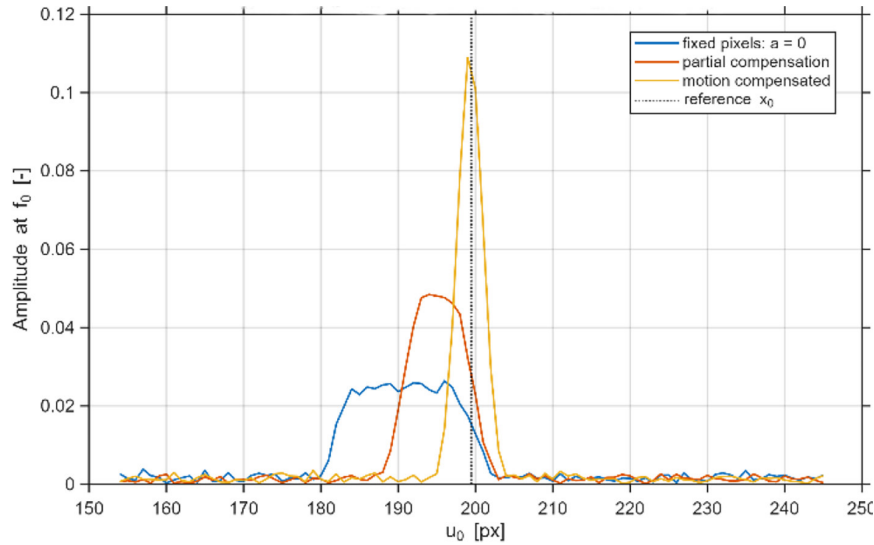


Figure 10. Horizontal amplitude profiles corresponding to the variants cases $a = 0$, $a = 0.5 \cdot a_{\text{ref}}$ and $a = a_{\text{ref}}$ for the rotation scenario

These values refer to individual amplitude maps and the definition of maximum adopted in this study. For the comparison of the STATIC, ORACLE, FSTRT-GLOBAL, and FSTRT-ODOM variants, the raw amplitudes and relative amplitude recovery efficiency presented in Table 4 were used in further analysis. The presented maps illustrate the compensation mechanism, while the statistical evaluation of the effectiveness was performed in further Monte Carlo studies.

Maximum response shape analysis

In this study, the shape of the FSTRT peak for translation, rotation and complex motion was compared using statistical parameters. The test results along with a description of each parameter are presented in Table 5.

In the Table 5, denotes the coordinate covariance matrix (Δx_0 , Δa) determined from the weighted local response map. The weights w_i were obtained after normalizing the map values.

The values λ_{max} and λ_{min} are the largest and smallest eigenvalues of the matrix, respectively. The parameter M denotes the number of map samples. The values of the presented parameters for the translation, rotation and complex motion scenarios are presented in Table 6.

For translation, the values were similar, although the ridge anisotropy was the highest. Therefore, the map shape metrics alone do not allow for a clear distinction of the type of camera movement in the current variant. However, the response map is highly elongated, and the Δx_0 and Δa errors are correlated. This is important for the subsequent use of the FSTRT response in the recursive filter as the image observation should not be treated as an independent measurement of x_0 and a . An example response map in relative axes is shown in Figure 11. Additionally, the normalized FSTRT response maps were compared between the motion scenarios. Figure 12 presents the normalized amplitude maps for translation and rotation, as well as their difference map.

Table 5. Definition of parameters used for research

Parameter	Equation	Definition
Δx_0	$\Delta x_0 = \hat{x}_0 - x_{0pred}$	Error of the map peak position in the x-axis
Δa	$\Delta a = \hat{a} - a_{pred}$	Error of the trajectory slope in the a-axis
Peak	$\frac{A_{FSTRTmax}}{\text{median}(A_{FSTRT})}$	Contrast of the peak relative to the background of the response map
$\sigma_{\Delta x_0}$	$\sqrt{\Sigma_{A11}}$	Width of the peak in the Δx_0 axis
$\sigma_{\Delta a}$	$\sqrt{\Sigma_{A22}}$	Width of the peak in the Δa axis
$\text{corr}_{\Delta x_0 \Delta a}$	$\frac{\Sigma_{A12}}{\sigma_{\Delta x_0} * \sigma_{\Delta a}}$	Correlation of the Δx_0 and Δa errors
Ridge anisotropy	$\sqrt{\frac{\lambda_{max}}{\lambda_{min}}}$	Response ridge extension
Entropy norm	$-\frac{\sum_i w_i \log(w_i)}{\log(M)}, w_i = \frac{A_{FSTRTi}}{\sum_i A_{FSTRTi}}$	Normalized entropy of the response map

The differences between the maps for translation and rotation were found to be small. This indicates that in the single-dimensional variant under consideration, the response map itself primarily describes the correspondence between the image trajectory and the pulsation. Based on the maps alone, it is impossible to clearly determine the type of camera movement – whether it is translation, rotation or a combination of movements.

The influence of incorrect assumption of parameter a

In the next test, for one camera movement sequence, the assumed parameter $a_{assumed}$ was varied relative to the reference value a_{ref} . The maximum amplitude was observed for the case $a_{assumed} = a_{ref}$. For classic STF processing, the normalized A_{STATIC} amplitude was only 0.144. However, for the correct assumption of the coefficient $a_{assumed} = a_{ref}$, the amplitude was equal to 1 on the normalized scale. The curve of the dependence of the amplitude on the error in the assumption of the parameter a is shown in Figure 13.

This result shows that the FSTRT response is strongly dependent on the correct choice of trajectory slope. At the same time, the response does not immediately disappear for values close to a_{ref} , indicating that there is a certain tolerance for prediction error.

Effect of marker apparent velocity and time window length

In a separate test, the camera's angular velocity was varied to obtain different values of $|a|$. For each case, the FSTRT gain was calculated relative to a fixed pixel. This relationship is shown in Figure 14. The gain increased monotonically with the marker's apparent velocity.

This is one of the most important simulation results. It directly demonstrates that FSTRT is not merely an alternative method for recording frequency analysis, but provides a real advantage in conditions of image motion. The greater the marker displacement within the analyzed window, the more classical fixed-pixel analysis loses pulsation information.

Table 6. Parameter values for simulation studies of the maximum response shape

Parameter	Translation	Rotation	Complex
Peak	26.510	26.302	26.832
$\sigma_{\Delta x_0}$ [px]	2.367	2.320	2.320
$\sigma_{\Delta a}$ [px/frame]	0.076	0.072	0.073
$\text{corr}_{\Delta x_0 \Delta a}$	-0.947	-0.915	-0.915
Ridge anisotropy	96.528	79.731	79.217
Entropy norm	0.229	0.256	0.257

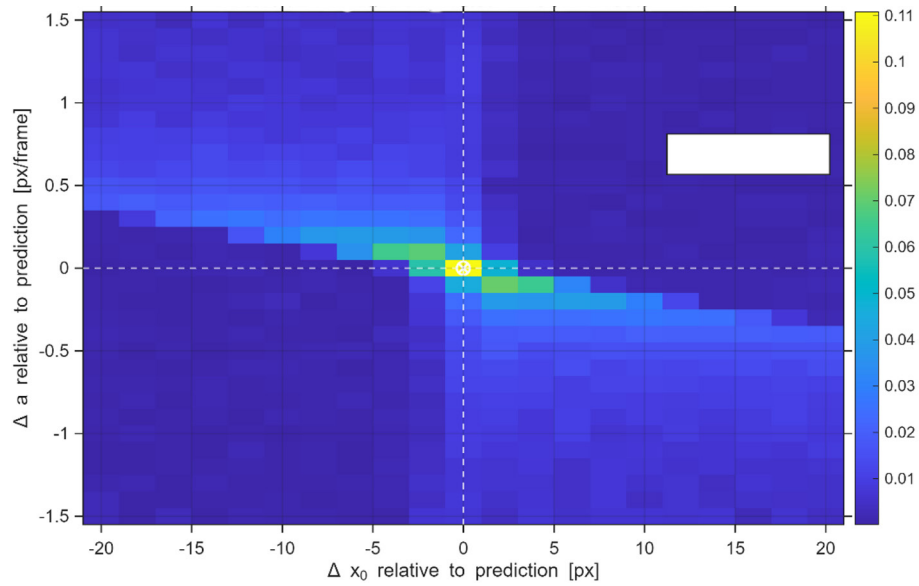


Figure 11. Example of the FSTRT response map in the relative coordinate system ($\Delta x_0, \Delta a$) for the complex motion scenario

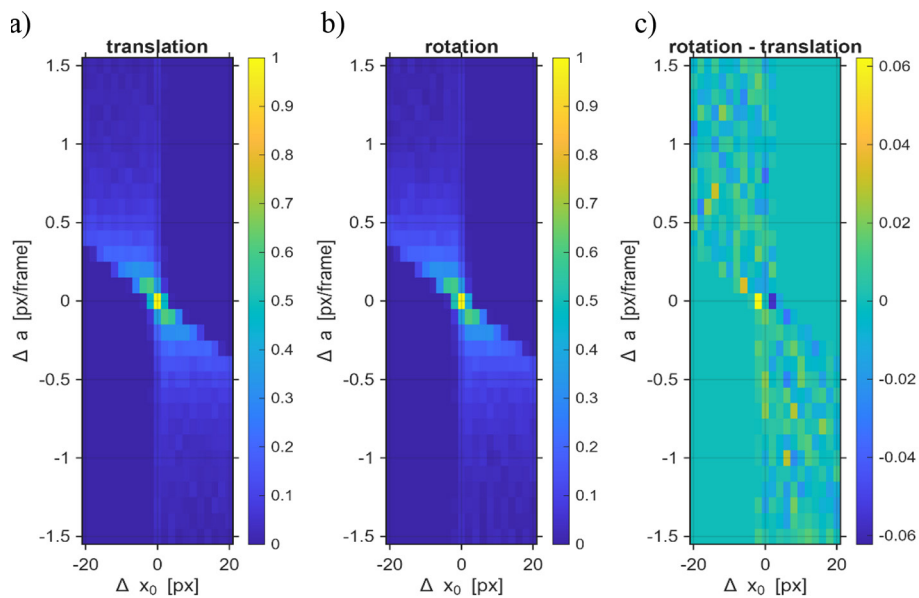


Figure 12. Comparison of normalized FSTRT response maps for: a) translation, b) rotation, c) the difference map

For the window length, values of 30, 60, 90 and 120 frames were tested which for a 60 Hz camera corresponds to times of 0.5 s, 1.0 s, 1.5 s and 2.0 s. The results of this test are shown in Figure 15. The FSTRT gain relative to a fixed pixel increased with the window length.

A longer window increases the number of signal samples and allows for a more pronounced processing advantage along the trajectory. At the same time, the width of the response peak in the Δx_0 axis increases for longer windows. This means

that a longer window does not always improve all aspects of the estimation. In the case of nonlinear motions, a trade-off between frequency selectivity and the accuracy of the linear trajectory model may arise.

The influence of recording time parameters

The amplitude recovered by FSTRT is influenced by three distinct temporal mechanisms: signal averaging during exposure, loss of

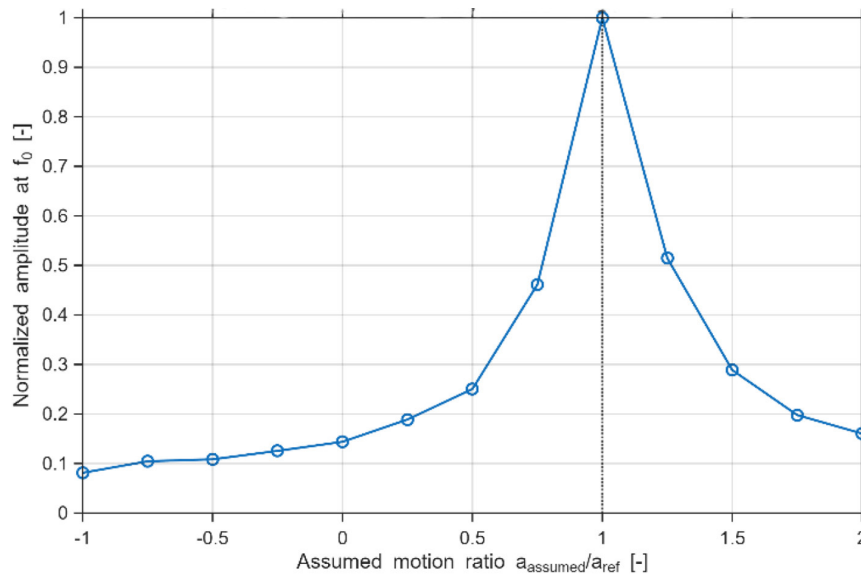


Figure 13. The effect of an incorrect assumption of apparent velocity a on the amplitude of the FSTRT response

coherence due to analysis frequency mistuning, and trajectory fitting error. Separating these factors is important because only the last of these can be constrained by the search for the x_0 and a parameters.

For a sinusoidal signal with frequency f_0 , the expected amplitude attenuation due to rectangular time integration is described by the relationship:

$$\eta_{exp} = \left| \frac{\sin(\pi f_0 T_{exp})}{\pi f_0 T_{exp}} \right| \quad (43)$$

Exposure time

Exposure time integrates the signal within each frame. A dimensionless exposure factor is introduced:

$$\rho = \frac{T_{exp}}{T_s}, T_s = \frac{1}{f_s} \quad (42)$$

The simulation results are presented in the Table 7 and in the Figure 16.

For $\rho = 0.5$ the measured relative ORACLE amplitude was approximately 0.955, consistent with the theoretical value of 0.955. For an exposure equal to the sampling period ($\rho = 1$, $T_{exp} = 16.67$ ms) the obtained values were 0.827

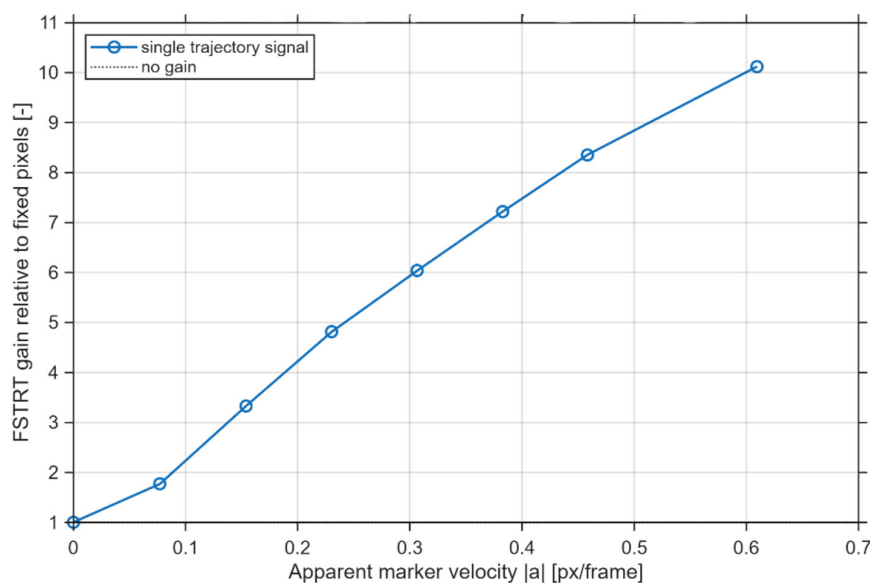


Figure 14. Dependence of the FSTRT gain relative to the classical fixed pixel analysis on the apparent velocity of the marker in the image

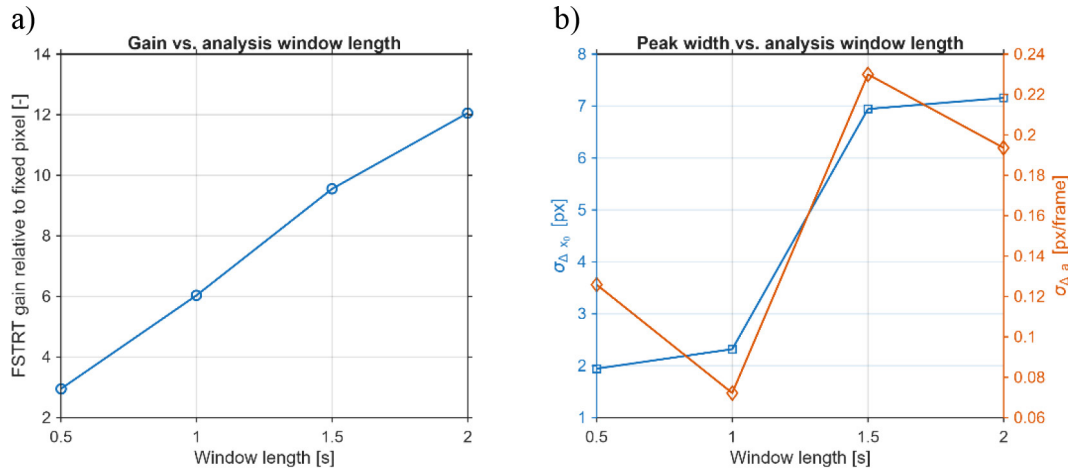


Figure 15. The effect of window length N on: a) gain relative to STATIC/STF, b) width of the maximum response in the Δx_0 direction

for rotation and 0.824 for complex motion, while the Sinc model predicted 0.827. Simultaneously, the FSTRT-GLOBAL/ORACLE ratio remained close to unity. This indicates that the observed decrease was due to temporal integration within the camera, not trajectory retrieval error.

Spatial motion compensation cannot recover the frequency component suppressed during exposure. Therefore, exposure time is a limitation independent of the accuracy of the x_0 and a estimates. This result was obtained for the global shutter model. In a rolling shutter camera, the sampling time also depends on the image row number, which requires extending the FSTRT kernel to include the readout time of a given row.

Frequency misalignment and phase sensitivity

The previous test assumed a known pulse frequency. In practice, the LED generator frequency, frame rate, and time stamps may be inaccurate. To a first approximation, the relative amplitude measured relative to the ORACLE response without time losses can be written as the product:

$$\frac{A_{meas}}{A_{ORACLE,0}} \approx \eta_{exp} \eta_f \eta_{traj} \quad (44)$$

where: η_{exp} – exposure attenuation, η_f – frequency coherence loss, $\eta_{traj} = A_{FSTRT}/A_{ORACLE}$ – loss resulting from trajectory mismatch.

For N samples and a rectangular window, the dependence of η_f on the error $\Delta f = f_{analysis} - f_{true}$ is described by the Dirichlet kernel modulus:

$$\eta_f = \left| \frac{\sin\left(\frac{\pi N \Delta f}{f_s}\right)}{N \sin\left(\frac{\pi \Delta f}{f_s}\right)} \right| \quad (45)$$

In the controlled study, $N = 60$ frames were used at $f_s = 60$ Hz. The DFT fringe width was $\Delta f_{DFT} = f_s/N = 1$ Hz. Table 8 and Figure 17 summarize the amplitude and its value normalized to the case $\Delta f = 0$.

For a rectangular window, an error of half a DFT fringe limited the amplitude to 0.629–0.652 of the maximum value, while a single fringe shift almost zeroed the response. The Hann window retained amplitudes of 0.850–0.863 and

Table 7. Relative Oracle amplitude for selected exposure times and additional trajectory search loss

ρ	T_{exp} [ms]	Sinc - theoretical	ORACLE - rotation	ORACLE – complex motion	GLOBAL/ORACLE – complex motion
0.00	0.00	1.000	1.000	1.000	1.000
0.25	4.17	0.989	0.990	0.989	1.000
0.50	8.33	0.955	0.957	0.954	1.000
0.75	12.50	0.900	0.902	0.899	1.000
1.00	16.67	0.827	0.827	0.824	1.001

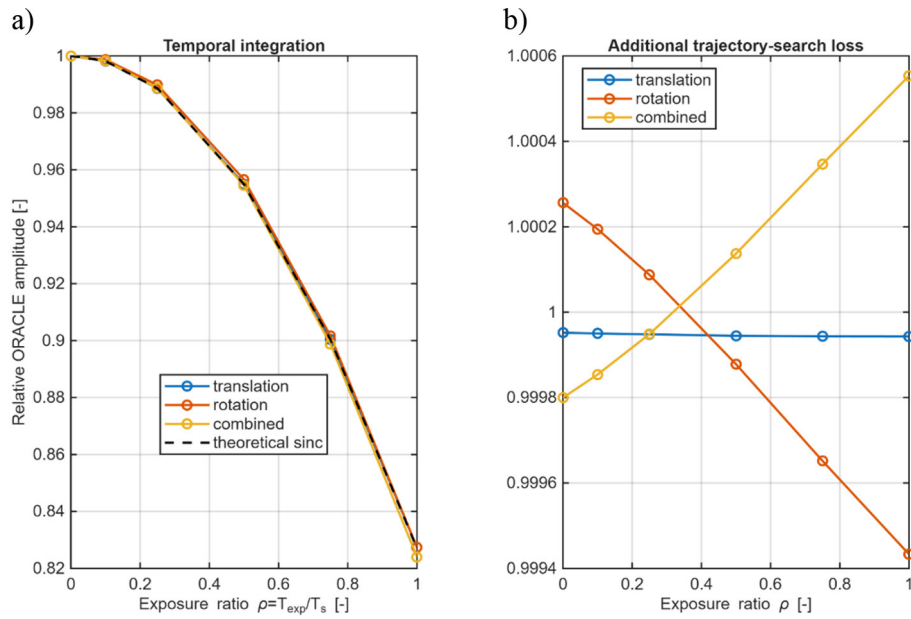


Figure 16. Effect of normalized exposure time: a) relative ORACLE amplitude compared to theoretical sinc attenuation, b) quotient of FSTRT-GLOBAL and ORACLE amplitude

0.510–0.535, respectively. It provides greater robustness to mistuning at the expense of a wider fringe and lower frequency resolution. Sensitivity to initial phase was secondary: the amplitude range was approximately 1.2% at a fine tuning of 20 Hz and approximately 3.7% at 19.7 Hz. This result indicates that with a finite window, frequency estimation or search is required, whereas a separate phase search is not essential under the analyzed conditions.

Exposure and frequency losses occur before trajectory consistency assessment. To a first approximation, their influence is multiplicative, so a high value of η_{traj} does not guarantee a large absolute amplitude if η_{exp} or η_f are small.

For a sampling frequency of $f_s = 60$ Hz, the applied marker frequency of $f_0 = 20$ Hz does not lead to aliasing in an ideal sampling model. However, this does not eliminate the mistuning resulting from the difference between the

actual and nominal frequency of the generator or camera clock.

Perturbed prediction correction test

A test of local correction of the prediction of x_0 and a was also performed. The prediction was shifted relative to the reference value and it was checked whether the local maximum of FSTRT returned to the correct trajectory. The correction vectors are shown in Figure 18.

This test should be considered an analysis of the local search capture range, not a full test of robot state correction. The results showed that the correction effectiveness depends on the relationship between the prediction error and the size of the local search window. If the correct maximum is located within the analyzed region, FSTRT can correct the prediction. However, if the prediction error shifts the

Table 8. Effect of frequency mistuning on the FSTRT amplitude for $N = 60$

$\Delta f/\Delta f_{DFT}$	A_{rect}	η_{rect}	A_{Hann}	η_{Hann}
-1.0	0.002	0.016	0.082	0.535
-0.5	0.095	0.629	0.132	0.863
0	0.151	1.000	0.153	1.000
+0.5	0.099	0.652	0.130	0.850
+1.0	0.001	0.007	0.078	0.510

Note: rect – rectangular window, Hann – Hann window.

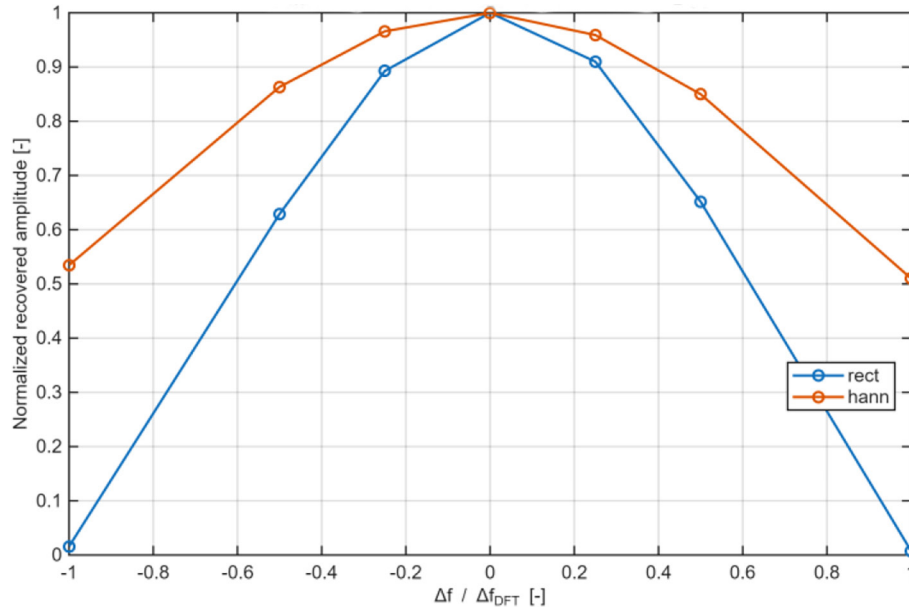


Figure 17. Normalized amplitude recovered by FSTRT as a function of frequency mistuning, expressed in DFT fringe widths, for rectangular and Hann windows ($N = 60$)

correct maximum outside the local region, the method selects a local or random maximum.

Detection with controlled false alarm probability

Amplitude alone does not determine detection efficiency. Therefore, a Monte Carlo study was performed for the hypothesis H_0 , meaning the absence of a pulsating tracer, and H_1 , meaning

its presence. For the detection statistic Λ and threshold γ , the detection and false alarm probability were defined as:

$$P_D = P(\Lambda > \gamma | H_1), P_{FA} = P(\Lambda > \gamma | H_0) \quad (46)$$

Their estimates were determined as shares of realizations exceeding the threshold:

$$\hat{P}_D = \frac{N_{det,H1}}{N_{H1}}, \hat{P}_{FA} = \frac{N_{det,H0}}{N_{H0}} \quad (47)$$

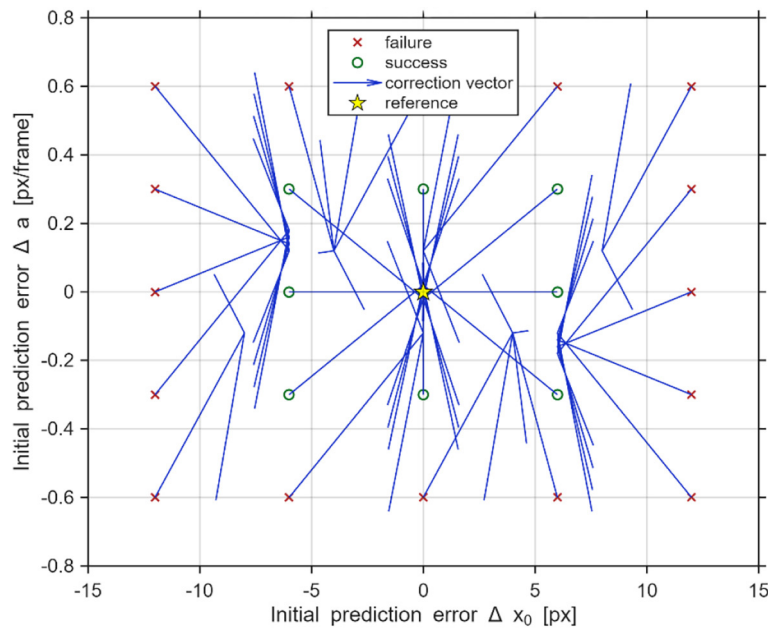


Figure 18. FSTRT correction vectors in the space $(\Delta x_0, \Delta a)$ for disturbed initial predictions in the complex motion scenario

The threshold for each variant was independently set from 500 realizations of H_0 for a target P_{FA} of 0.05 and then evaluated on independent sets of 1000 realizations of H_0 and 1000 realizations of H_1 . For each realization of H_0 , the statistic was the maximum in the entire search area of the given variant, which accounts for the effect of multiple comparisons. Additional measures included the area under the ROC curve (AUC) and the trajectory error:

$$\hat{P}_D = \frac{N_{det,H1}}{N_{H1}}, \hat{P}_{FA} = \frac{N_{det,H0}}{N_{H0}} \quad (48)$$

Table 9 shows the complex motion condition with high read noise level $\sigma_{read} = 0.5$. This corresponds to an amplitude index $SNR_A = 0.63$ (this is not a signal-to-noise ratio). ROC curves are presented in Figure 19.

With an empirical P_{FA} close to 0.05, the STATIC method achieved $P_D = 0.218$. FSTRT-GLOBAL increased P_D to 0.817, while the FSTRT-ODOM variant increased P_D to 0.950, very close to the ORACLE value of 0.959. A similar

order of results was obtained for AUC: 0.736 for STATIC, 0.955 for FSTRT-GLOBAL, 0.988 for FSTRT-ODOM, and 0.991 for ORACLE. The results confirm that amplitude recovery through motion compensation translates into an increase in detectability and limiting the search area using odometry reduces the statistical cost of maximization over multiple trajectories.

The influence of odometry errors and search range

The FSTRT-ODOM variant limits the search range based on the predicted odometry trajectory. This reduces computational cost and the number of noise candidates, but is dependent on the quality of the motion model. A distinction must be made between the search range covering the actual trajectory, the correct maximum detection and the actual marker detection:

$$C_{cov} = P\left((x_{0,true}, a_{true}) \in \Omega_{search}\right) \quad (49)$$

Table 9. Trajectory detection and estimation results for complex motion and high noise conditions. Wilson's 95% confidence intervals are given in parentheses

Method	P_{FA}	P_D	AUC	F_1	RMSE _u [px]	Test size
STATIC/STF	0.053 (0.041–0.069)	0.218 (0.194–0.245)	0.736	0.343	9.550	1000 H_0 + 1000 H_1
ORACLE	0.043 (0.032–0.057)	0.959 (0.945–0.970)	0.991	0.958	0.000	1000 H_0 + 1000 H_1
FSTRT-GLOBAL	0.052 (0.040–0.068)	0.817 (0.792–0.840)	0.955	0.874	2.193	1000 H_0 + 1000 H_1
FSTRT-ODOM	0.062 (0.049–0.079)	0.950 (0.935–0.962)	0.988	0.944	1.168	1000 H_0 + 1000 H_1

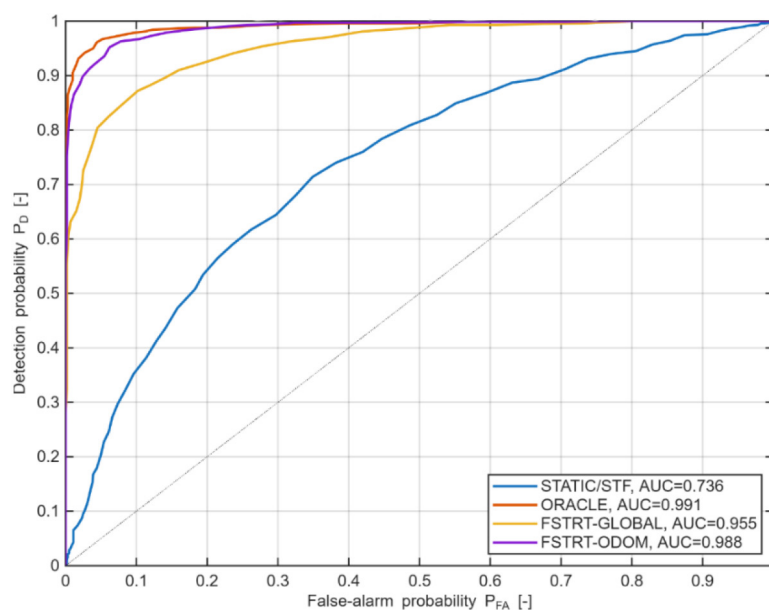


Figure 19. ROC curves for STATIC/STF, ORACLE, FSTRT-GLOBAL, and FSTRT-ODOM in the complex traffic and high noise scenario

$$P_{corr} = P\left(\left(|\hat{x}_0 - x_{0,true}| \leq \varepsilon_x\right) \wedge \left(|\hat{a} - a_{true}| \leq \varepsilon_a\right)\right) \quad (50)$$

The C_{cov} measure determines the probability that the true pair of parameters is located in the Ω_{search} region, while P_{corr} additionally requires that the estimates $\hat{x}_0$ and $\hat{a}$ fall within the assumed tolerances ε_x and ε_a . In the Monte Carlo study, 200 realizations were performed for both the fixed and adaptive search ranges. The case presented in Table 10 corresponds to an odometry error scale of 8. Additionally, the impact of trajectory and odometry model errors is presented in Figure 20.

The fixed search range covered the true trajectory in only 7.5% of the executions, which reduced P_D to 0.695 and increased $RMSE_u$ to 21.42 px. The adaptive range, expanded based on odometry uncertainty, provided $C_{cov} = 1.000$, $P_D = 1.000$ i $RMSE_u = 1.38$ px with an increase in the average computation time from 1.75 to 2.30 ms. Simultaneously P_{corr} was only 0.220. This means

that covering the true trajectory is a necessary, but not sufficient condition for its unambiguous estimation. A larger area increases the number of competing peaks and the effect of multiple comparisons. The results support selecting the range based on odometry covariance and multiresolution or coarse-to-fine search. Time values for individual frame series are presented in Table 11.

The influence of violating the one-dimensional motion assumption

The 1D variant assumes no significant vertical motion of the marker projection. In an additional test, this assumption was violated by introducing a vertical shift D_y , normalized by the $FWHM_v$ width of the marker image. A 3 dB amplitude decrease occurred at $D_y/FWHM_v = 1.312$. For a shift of 4 $FWHM_v$, $PD = 0.605$ was obtained with $PFA = 0.049$. This result does not validate the 1D method for vertical motion, but quantitatively defines its applicability range and justifies extending the parameter space to (x_0, y_0, a_x, a_y) .

Table 10. The influence of search range on the robustness of FSTRT-ODOM (200 realizations, odometry error scale 8)

Range	C_{cov}	P_{corr}	P_D	P_{FA}	$RMSE_u$ [px]	Time [ms]
Fixed	0.075	0.100	0.695	0.055	21.42	1.75
Adaptive	1.000	0.220	1.000	0.025	1.38	2.30

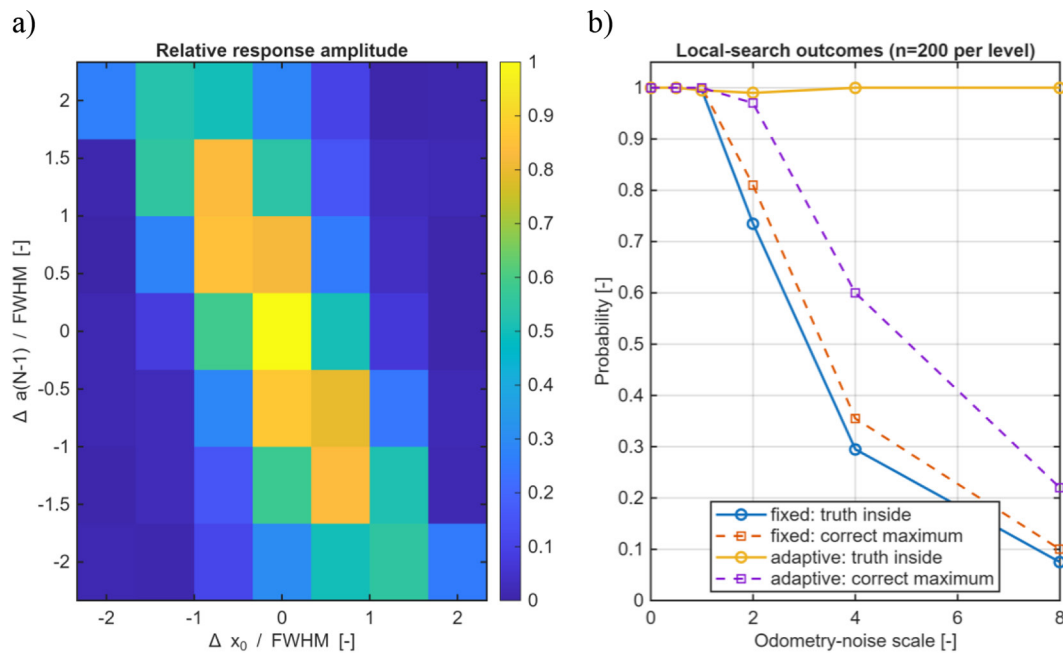


Figure 20. Influence of trajectory and odometry model errors: a) relative response amplitude for systematic errors x_0 and a , b) probability of covering the actual trajectory and indicating the correct maximum for fixed and adaptive search ranges

Table 11. Average computation time for a given number of frames N

N	STATIC/STF	FSTRT-ODOM	FSTRT-GLOBAL
30	0.53 ms	0.75 ms	3.85 ms
60	1.00 ms	1.35 ms	6.35 ms
120	1.95 ms	2.34 ms	11.36 ms

CONCLUSIONS

This paper addresses the problem of diminished marker pulsation information that occurs during camera movement. In classical STF analysis, the signal is generated based on the intensity values of the same pixel in successive frames. If the marker projection moves within the image, its pulsation information is distributed across multiple pixels. The proposed FSTRT method mitigates this effect by sampling the intensity along candidate trajectories in the image-time space and assessing the content of the known pulsation frequency.

Simulation results confirmed that, under the studied conditions, the importance of compensation increases with marker projection displacement within the analyzed window. In rotational and complex motion scenarios, the FSTRT variants recovered approximately 98–100% of the amplitude obtained for the reference trajectory, while the STATIC/STF analysis retained approximately 23–25%. In a Monte Carlo study for complex motion and high noise, the local variant of FSTRT increased detection probability from 0.218 to 0.950 with a similar false alarm probability. This confirms that motion compensation improves not only the response amplitude but also the practical detectability of the marker.

The conducted studies also showed that the effectiveness of the method depends on the acquisition parameters and the quality of the trajectory prediction. Exposure time causes amplitude attenuation that cannot be removed by spatial compensation, while frequency mistuning leads to a loss of coherence dependent on the length and type of the time window. Increasing the prediction uncertainty showed that a fixed local search area quickly loses the true trajectory, while an adaptive range can maintain its coverage and high detection probability.

However, this does not guarantee accurate trajectory parameter prediction, because expanding the range with a fixed number of candidates reduces the grid resolution and increases the number of competing maxima. The results indicate

the need for propagating the full covariance of the parameters (x_0 , a), using multiresolution or coarse-to-fine search and subsequently integrating FSTRT with a recursive state estimator. The presented test applies to independent windows and does not include the time-dependent odometry error growth.

The presented research involves block-based analysis of a single image window, a single pulsating marker, and a one-dimensional model of its image trajectory. The method does not currently support continuous tracking or estimation of camera or robot position. However, the obtained results represent an important step in the development of a pulsating marker-based localization system, as they address the problem of preserving frequency information during camera motion. Further work should address two-dimensional motion, multiple markers, subsequent time windows, integration with a position estimator, and validation using a real camera.

REFERENCES

1. Tourani A, Bayle H, Avşar DI, Sanchez-Lopez JL, Munoz-Salinas R, Voos H. Vision-based situational graphs exploiting fiducial markers for the integration of semantic entities. *Robotics* 2024;13. <https://doi.org/10.3390/robotics13070106>
2. Jiang J-R, Subakti H. An indoor location-based augmented reality framework. *Sensors* 2023;23. <https://doi.org/10.3390/s23031370>
3. Luo J, Fan L, Li H. Indoor positioning systems based on visible light communication: State of the Art. *IEEE Commun Surv Tutor* 2017;19:2871–93. <https://doi.org/10.1109/COMST.2017.2743228>
4. Rego M, Fonseca P. OCC Based Indoor Positioning System Using a Smartphone Camera. In: Santos V., Lau N., Neto P., Lopes A.C. (Eds.) *IEEE Int. Conf. Autonomous Robot Syst. Competitions, ICARSC*, Institute of Electrical and Electronics Engineers Inc.; 2021, p. 31–6. <https://doi.org/10.1109/ICARSC52212.2021.9429782>
5. Nguyen D.Q., Nguyen H.N. Accurate, extended-range indoor visible light positioning via high-efficiency MPPM modulation with smartphone

- multi-sensor fusion. *Photonics* 2025;12. <https://doi.org/10.3390/photonics12090859>
6. Zhu Z, Yang Y, Chen M, Guo C, Hao J, Cui S. Visible light positioning with visual odometry: a single luminaire based positioning algorithm. *IEEE Trans Commun* 2024;72:4978–91. <https://doi.org/10.1109/TCOMM.2024.3375817>
7. Miś P, Szulim P. Analysis of the possibility of using markers emitting pulsating light in the task of localization. *Applied Computer Science* 2021;Vol. 17, no 1:26–39. <https://doi.org/10.23743/acs-2021-03>
8. Miś P, Szulim P. The Effect of Camera Exposure on the Results of Spatial-Frequency Processing and the Quality of the Obtained Amplitude Images. In: Szewczyk R, Zieliński C, Kaliczyńska M. (Eds.) *Automation 2022: New solutions and technologies for automation, robotics and measurement techniques*. Cham: Springer International Publishing; 2022, 306–16.
9. Miś P, Szulim P, Mączak J. A concept of a vision system for position determination using spatial-time-frequency analysis. *Eng Trans* 2025;73:367–86. <https://doi.org/10.24423/engtrans.3499.2025>
10. Sun X., Shi W., Cheng Q., Liu W., Wang Z., Zhang J. An LED detection and recognition method based on deep learning in vehicle optical camera communication. *IEEE Access* 2021;9:80897–905. <https://doi.org/10.1109/ACCESS.2021.3085117>
11. Niarchou E, Matus V, Rabadan J, Guerra V, Perez-Jimenez R. Optical camera communications in healthcare: A wearable LED transmitter evaluation during indoor physical exercise. *Sensors* 2024;24:2766. <https://doi.org/10.3390/s24092766>
12. Song S, Wu P, Liu Y, Zhao L, Wu T, Liu X, et al. Implementation of an indoor optical camera communication and localization fusion system using infrared LED markers. *Opt Express* 2024;32:41361–75. <https://doi.org/10.1364/OE.538088>
13. Wang Z, Ng Y, Henderson J, Mahony R. Smart Visual Beacons with Asynchronous Optical Communications using Event Cameras. *arxiv*, 2022. <https://doi.org/10.48550/arXiv.2208.01710>
14. von Arnim A, Lecomte J, Borrás NE, Woźniak S, Pantazi A. Dynamic event-based optical identification and communication. *Frontiers in Neurorobotics* 2024;18–2024.
15. Su H., Gao L., Liu T., Kneip L. Motion-aware optical camera communication with event cameras. *IEEE Robotics and Automation Letters* 2025;10:1385–92. <https://doi.org/10.1109/LRA.2024.3517292>
16. Ebmer G, Loch A, Vu MN, Mecca R, Haessig G, Hartl-Nesic C, et al. Real-time 6-DoF pose estimation by an event-based camera using active LED markers. *Proc. IEEE Winter Conf. Appl. Comput. Vis., WACV*, Institute of Electrical and Electronics Engineers Inc.; 2024, p. 8122–31. <https://doi.org/10.1109/WACV57701.2024.00795>
17. Benmoussat N, Faouzi Belbachir M, Benamar B. Motion estimation and compensation from noisy image sequences: A new filtering scheme. *Image Vision Comput* 2007;25:686–94. <https://doi.org/10.1016/j.imavis.2006.05.010>
18. Shi X, Huang Z, Bian W, Li D, Zhang M, Cheung KC, et al. VideoFlow: Exploiting temporal cues for multi-frame optical flow estimation. *Proc IEEE Int Conf Comput Vision*, Institute of Electrical and Electronics Engineers Inc.; 2023, p. 12435–46. <https://doi.org/10.1109/ICCV51070.2023.01146>
19. Li L, Yang Y. Optical flow estimation for a periodic image sequence. *IEEE Trans Image Process* 2010; 19: 1–10. <https://doi.org/10.1109/TIP.2009.2032341>
20. Gentil CL, Alzugaray I, Vidal-Calleja T. Continuous-time Gaussian process motion-compensation for event-vision pattern tracking with distance fields. *Proc IEEE Int Conf Rob Autom*, vol. 2023- May, Institute of Electrical and Electronics Engineers Inc.; 2023, p. 804–12. <https://doi.org/10.1109/ICRA48891.2023.10160768>
21. Yamamoto S, Nakajima Y, Tamura S, Sato Y, Harino S. Extraction of fluorescent dot traces from a scanning laser ophthalmoscope image sequence by spatio-temporal image analysis: Gabor filter and radon transform filtering. *IEEE Trans Biomed Eng* 1999;46:1357–63. <https://doi.org/10.1109/10.797996>
22. Szeliski R. *Computer Vision: Algorithms and Applications*. Springer London; 2010.
23. Li R, Jiang Q. A photogrammetric method for the measurement of three-dimensional cervical range of motion. *IEEE J Biomedical Health Inform* 2022;26:685–95. <https://doi.org/10.1109/JBHI.2021.3089254>