1) Names and surnames of authors of the solution

Mirjana Filipovic,

Mihailo Pupin Institute, University of Belgrade, Volgina 15, 11060 Belgrade, Serbia. mirjana.filipovic@pupin.rs

Ljubinko Kevac,

[a]School of Electrical Engineering, University of Belgrade, [b]Innovation center of School of Electrical Engineering, University of Belgrade, Bulevar kralja Aleksandra 73, 11000, Belgrade, Serbia
ljubinko.kevac@ic.etf.rs

2) Name of the technical solution

Program package **CPRTS**, (**C**able-suspended **P**arallel **R**obots **T**rajectory **S**olver)

3) Key Words

CPR system, real time, trajectory planning, object monitoring.

4) For whom the solution was done (legal entity or industries)

Scientific Research for the purpose of validation of theoretical assumptions

5) The year when the solution was completed

2016.

6) Year when it began to be implemented and by whom

The program package **CPRTS** was implemented in published paper [1], in the journal Robotics and Autonomous Systems, 2017, appropriate category M22.
We present the confirmation of theoretical contributions in the subtitle 3. "The Program Package CPRTS - Simulation Experiments" using software package **CPRTS** in paper [1]. On this basis the software package **CPRTS** acquires the right to be ranked in the category M82.

7) The area and the scientific field, which the technical solution refers to

Object tracking for the CPR system is highly multidisciplinary area and it integrates several areas: robotics (kinematics and dynamics), different types of actuations systems, measurement and sensors, electronics, computers, control algorithms and their software implementation, visual systems (image acquisition and processing, visual navigation, etc.), procedure analysis and synthesis, pattern recognition, colours, artificial intelligence.

8) Problem solved by technical solution

A new procedure for creating the reference trajectory of the CPR system's camera intended for object tracking and monitoring in real time is presented. For the purpose of analyzing and using the defined algorithm, we have synthesized a program package **CPRTS**. By using this program package, the simulation experiments of the camera's trajectory generation for the purpose of the object monitoring and tracking are presented. By using the CPR Trajectory Solver, motion autonomy of CPR system's camera is increased. The camera has a task to follow and monitor the chaotically moving object.

9) State of the problem solution in the world

One strong class of mechanisms used for different camera motions of CPR type has been developed strongly in last three decades. For three decades, researchers have dealt with the mechanisms that carry a camera for observation of the space or moving objects in space. Implementation of this system uses the latest technical knowledge. This is achieved using the latest computer, network and new types of motors, combined with confirmed high quality camera and video components.

Specially trained camera operator is responsible for process and quality recording. It is clear that for the full functioning of such a complex system must be responsible a team of experts of various profiles.

In its initial phase the system was used in limited circumstances that the technology's development allowed. Because of that its development has been slow, due to the support of computers technique. From that time until now the use of this system has been constantly expanding. During the transfer of sporting or other public events (football, basketball, hockey or other games, tennis or athletic competitions, Eurovision, concerts of various musical genres etc.) the presence of **CPR** system used for recording can be noticed. But in many areas this system has not yet appeared.

It should be noted here that the position of the camera is controlled via joystick by the operators, who are located to monitor the whole workspace with moving subjects. In this case the operator via the joystick moves the camera in all directions left, right, up, down etc. This system in its functioning depends on the concentration and the responsibility of the operators. Their skill and experience play a major role. Since the implementation of task observation of an event can last for hours, it's a pretty big responsibility for operators. Lack of concentration and their fatigue during the period indicates a significant dependence of this system on the human factor, which as a result can have a number of unavoidable inaccuracies during the operation.

The patent of Richard Thomas Hale, Object-oriented Cable Camera System, Patent US20120300079 A1, 29 Nov. 2012, deals with an object-oriented cable camera system. This system processes the sports

Volgina 15, 11060 Belgrade, Serbia
Phone: (+381 11) 2771-398, 2772-876
Fax: (+381 11) 2776-583
WWW.INSTITUTEPUPIN.COM
email: info@institutepupin.com

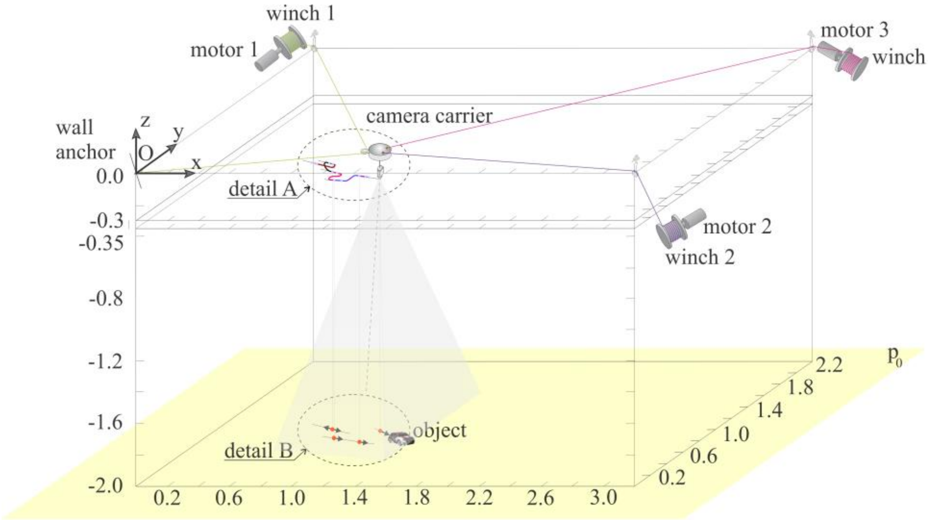
ball's position and automatically follows the ball in play. The inventors use tracking chips which are placed on both the camera and the ball and in that way they establish camera’s and object’s positions. The authors of the patent guarantee that the ball is always in camera’s sight. They have not defined the procedure for generation of the camera’s reference trajectory during ball’s tracking, which opened a space for our research in this area: definition of the algorithm for CPR system’s camera reference trajectory creation during the object real time tracking, which we have named the CPR Trajectory Solver. We assume that CPR system’s camera needs to follow the object which is, in camera’s sight, during its motion defined by Richard Thomas Hale. The tracking chip is a small device that is attached to the camera and relays its position back to a computer control system. This tracking device could include GPS, magnetic, infrared, RFID (Radio-frequency identification), reflector, sonar or other type of tracking technology. But, position and velocity orientation of the object are known only at certain moments. Also, it is needed that the object is always in camera’s sight. With these assumptions, the CPR Trajectory Solver ensures a smooth and continuous reference trajectory of the camera’s motion in time and space, while the object’s motion trajectory can be chaotic.

The program package **CPRTS** is the basic result and it opens up the possibilities for further development the completely autonomous CPR system is developed and that is why we have created the novel procedure CPR Trajectory Solver for CPR systems with camera to track and monitor objects in their workspace.

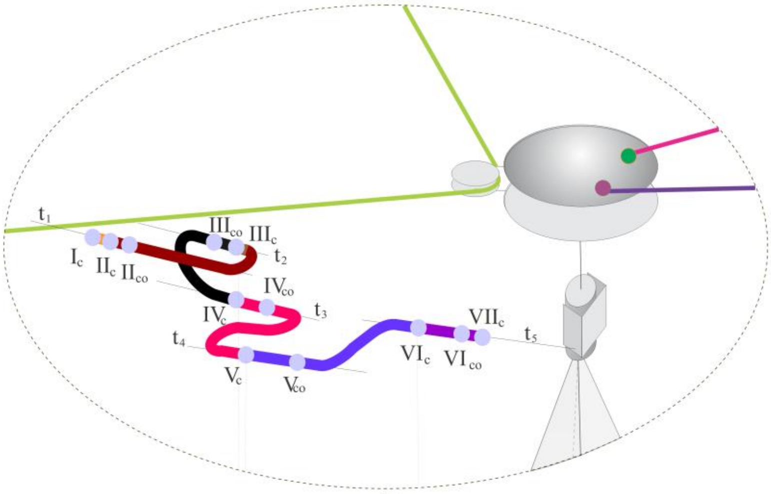
10) Description of the technical solution

This program system **CPRTS** was generated in MATLAB. The CPR Trajectory Solver is a procedure defined which is used to generate a smooth reference trajectory of CPR system’s camera which has a task to monitor and track the object in real time. For trajectory generation, the CPR Trajectory Solver uses the knowledge about the current positions and velocity orientations of the camera and object and then it defines the goal position and velocity orientation of the camera. The CPR Trajectory Solver chooses one of the generated primitives (from data base of four primitive trajectories) for interconnecting the current and goal positions of the camera. After completing the chosen primitive, the CPR Trajectory Solver establishes the new positions and velocity orientations of the object. This process is repeated cyclically until the real time object monitoring and tracking task is completed. For the purpose of analyzing and using the defined algorithm, we have synthesized a program package: CPRTS (CPR Trajectory Solver). By using the CPR Trajectory Solver, motion autonomy of CPR system’s camera is increased. The camera has a task to follow and monitor the chaotically moving object.

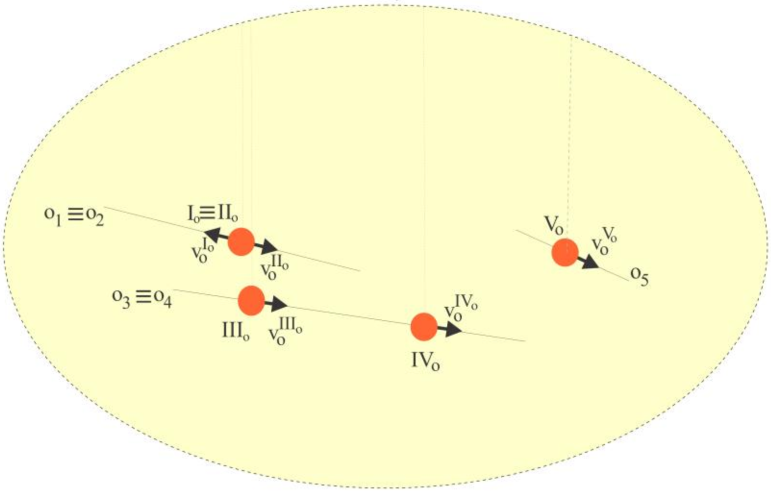
Volgina 15, 11060 Belgrade, Serbia
Phone: (+381 11) 2771-398, 2772-876
Fax: (+381 11) 2776-583
WWW.INSTITUTEPUPIN.COM
email: info@institutepupin.com



a)



b)



c)

Figure1. The RSCPR system's camera tracking and monitoring a chaotically moving object: a) 3D view, b) detail A: detailed view of the camera trajectory, c) detail B: detailed view of object's positions.

For the purpose of implementation of the task of object following, we have presumed that the CPR

Volgina 15, 11060 Belgrade, Serbia
Phone: (+381 11) 2771-398, 2772-876
Fax: (+381 11) 2776-583
WWW.INSTITUTEPUPIN.COM
email: info@institutepupin.com

system had sufficient energy resources so it could follow the object's highest velocity v_{omax}. Also, during the implementation of the task, the CPR system has a possibility of accelerating and decelerating the camera depending on the object's velocity change during the camera's task execution. This means that the camera's velocity needs to follow the object's motion at any moment according to the conditions of the CPR Trajectory Solver's functioning. We have presumed that the object moves only at the lowest plane of the camera's workspace. This plane is labelled as p_0 in Figs. 1 and 5. This plane is on level $z_{p_0} = -2[m]$. Starting position of the camera is defined by the operator so that the camera is above the current position of the object. At that moment, the operator chooses motion direction of the camera.

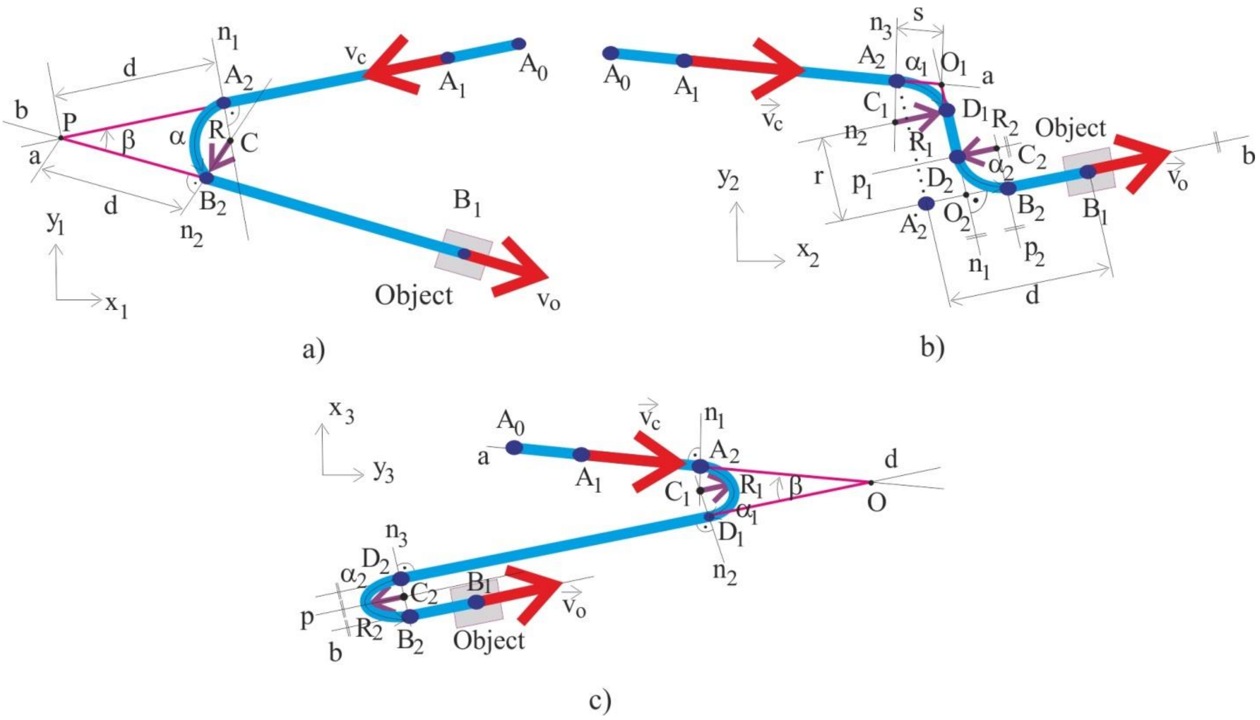


Figure 2. The primitives: a) the first primitive **(P1)**, b) the second primitive **(P2)** and c) the third primitive **(P3).**

Afterwards, the operator starts the program which, by definition, starts with the acceleration process (**ACC** process) from zero to the starting reference velocity of the CPR system's camera. The starting reference velocity of the camera is a presumed average velocity of the object. During the implementation of the task, camera's reference velocity can change depending on the object's motion and velocity variations. After the starting moment, the CPR Trajectory Solver takes control of the camera's motion. The monitoring and tracking system of the object is completely autonomous.

During the object tracking process, based on the current positions and velocity orientations of the

Volgina 15, 11060 Belgrade, Serbia
Phone: (+381 11) 2771-398, 2772-876
Fax: (+381 11) 2776-583
WWW.INSTITUTEPUPIN.COM
email: info@institutepupin.com

camera and object, the CPR Trajectory Solver chooses one of the primitives (see Fig. 2) from the data base and in that fashion the camera follows the object. This process is repeated until the operator decides to stop the object tracking task. At that moment, the CPR Trajectory Solver starts the camera deceleration process (**DEC** process) and it slows down the camera from the reference to zero velocity.

The CPR Trajectory Solver can use one of four predefined primitive trajectories (primitives) which are stored in data base. In Fig. 2, we show three primitives which can be used by the CPR Trajectory Solver during the object tracking based on the current positions and velocity orientations of the object and camera. The fourth primitive is a straight line defined as **P4.**

These primitives are defined in the local coordinate system, but the CPR Trajectory Solver can use the transformation matrix and map them at any plane of the global 3D coordinate system of the camera's workspace. The **ACC** and **DEC** processes can be used by the CPR Trajectory Solver not only by the operator but also at the moments when the object suddenly changes its velocity and/or when the object is too close to the border of the camera's sight. It is presumed that **ACC** and **DEC** processes can only be executed via **P4** primitive.
Primitives were chosen so that they cover all the presumed positions and velocity orientations of the camera and object during the task execution. For the sake of clarity of presentation, primitives **P1-P3** will be explained through following two Tables. Table 1 describes a general flow of execution of the CPR Trajectory Solver for any of three mentioned primitives. Primitive is generally composed of elements given in Table 1.

Point A_0	Position of the camera at the moment of last primitive's completion.
Motion $A_0 - A_1$	Camera keeps on moving along straight line and the CPR Trajectory Solver establishes the new position and orientation of the object. Based on that and current position of the camera, it defines camera's goal point and chooses the primitive.
Motion $A_1 - A_2$	Motion along the straight line.
Motion $A_2 - B_2$	Connecting the direction of camera's motion with direction of object's motion.
Motion $B_2 - B_1$	Camera keeps on moving along straight line until it reaches a position above the last known position of the object.
Point B_1	Last point for execution of current primitive and the first point for the next primitive.

Table 1. General description of primitives **P1-P3**.
As it can be seen in Fig. 2, primitives P**1-P3** are defined with certain number of straight lines and arcs.

Table 2 gives the conditions when and how the CPR Trajectory Solver chooses on of primitives **P1, P2** and **P3** from its data base. Primitives are defined in Fig. 2 in local coordinate system.

Volgina 15, 11060 Belgrade, Serbia
Phone: (+381 11) 2771-398, 2772-876
Fax: (+381 11) 2776-583
WWW.INSTITUTEPUPIN.COM
email: info@institutepupin.com

Primitive	Figure	Camera and object orientation	Object's position in relation to the camera
P1	2a)	Opposite direction of object's and camera's velocity in x_1 and y_1 directions of local coordinate system	Not important
P2	2b)	Same direction of object's and camera's velocity in x_2 direction and opposite direction of these velocities in y_2 direction	$x_{2B_1} > x_{2A_1}$
P3	2c)	Same direction of object's and camera's velocity in x_3 direction and opposite direction of these velocities in y_3 direction	$x_{3B_1} < x_{3A_1}$

Table 2. Conditions of choosing primitives **P1-P3**.

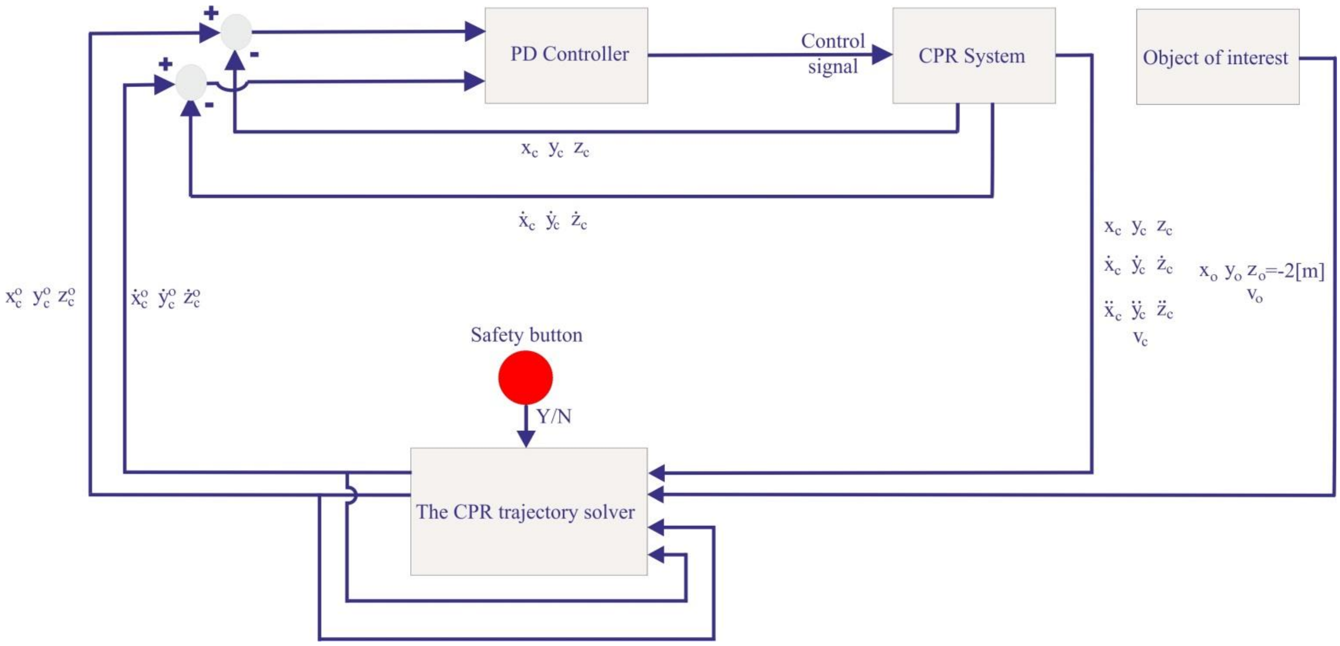


Figure 3. Block diagram of the trajectory choosing process.

Global structure of the system for detecting and following the object which is in camera's sight is shown in Fig. 3. Detailed presentation of the CPR Trajectory Solver is given with the flow chart in Fig. 4. At any moment, the system checks if the operator has activated the CPR Trajectory Solver. If the operator has not activated the CPR Trajectory Solver, the whole system is in idle state. When the operator activates the logic by pressing the **START** button, the CPR Trajectory Solver used for object tracking and reference trajectory generation is started. The process of reference trajectory generation repeats itself and the camera is able of monitoring and tracking the object. However, the structure of the system is much more complex. The system consists of the following blocks: **CPR System**, **PD Controller** and **the CPR Trajectory Solver**.

Volgina 15, 11060 Belgrade, Serbia
Phone: (+381 11) 2771-398, 2772-876
Fax: (+381 11) 2776-583
WWW.INSTITUTEPUPIN.COM
email: info@institutepupin.com

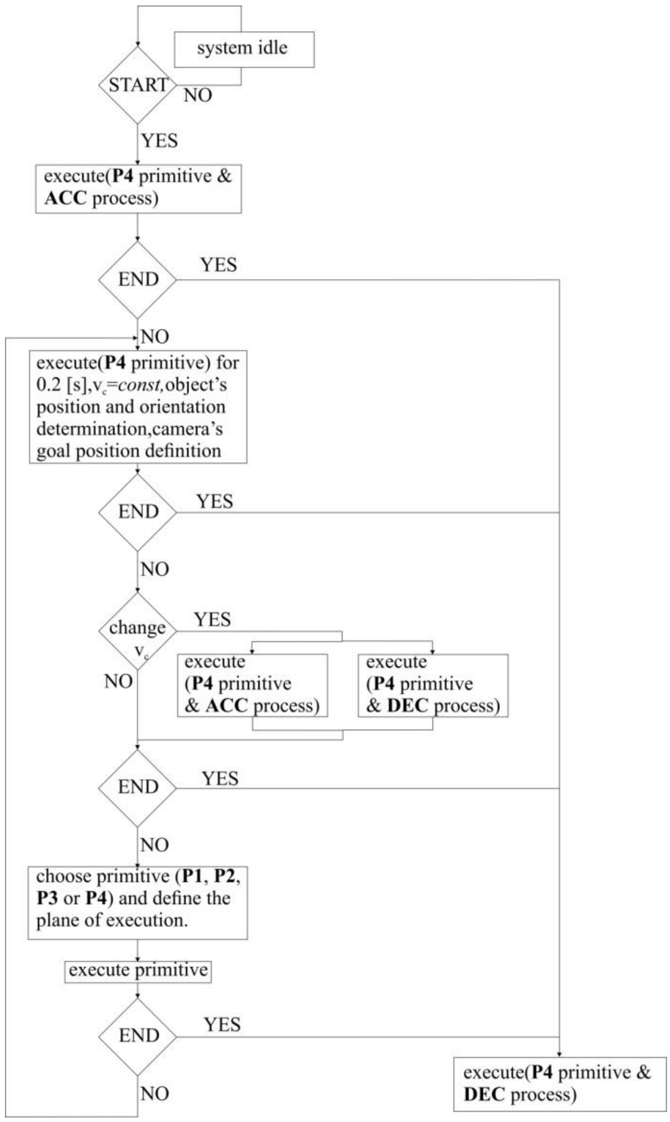


Figure 4. Flow chart of the CPR Trajectory Solver.

Also, the system contains a **Safety button** which can be used by the operator in case of an emergency.

Reference trajectory of camera in 3D space is defined in real time. Based on these variables and the real trajectories of the external coordinates x_c, y_c, z_c, by using the **PD Controller** the control signals are determined and the **CPR System** with the camera performs the real trajectory by following the generated reference trajectory. The topic of this paper is definition of the CPR Trajectory Solver and it will be explained in detail:

a) **Starting the program.** This procedure is performed by the operator based on his/her visual estimation of the current position and velocity orientation of the object. Before starting the object monitoring and tracking process, the operator places the camera at the position so that the object is close to the central point of the basis of the camera's sight.

b) **Acceleration process.** The operator starts this process by pressing button **START**. With this command, the CPR Trajectory Solver starts the **ACC** process and **P4** primitive. Camera starts to move in direction of current velocity orientation of the object based on the operator's estimation. Afterwards, the CPR Trajectory Solver takes control of the system and camera's motion is completely autonomous. The process of camera's acceleration towards its starting reference velocity is executed.

c) **The procedure and conditions of choosing the primitive**. Next 0.2 [s], the camera keeps

Volgina 15, 11060 Belgrade, Serbia
Phone: (+381 11) 2771-398, 2772-876
Fax: (+381 11) 2776-583
WWW.INSTITUTEPUPIN.COM
email: info@institutepupin.com

moving (**P4** primitive) in the same direction as during the acceleration and in that period it establishes the current position and velocity orientation of the object – **Object of interest**. At that moment, the CPR Trajectory Solver also knows the current position and velocity orientation of the camera. Based on these data, the CPR Trajectory Solver defines the next goal point of the camera in its workspace. In the next step, the CPR Trajectory Solver decides if it is needed for the camera to change its velocity magnitude. The change of camera's velocity magnitude is performed by using the **ACC** and **DEC** processes and primitive **P4**. Afterwards, the CPR Trajectory Solver chooses the next primitive from its data base. Then, it chooses the plane in which the primitive should be executed and transforms the primitive to the coordinates of the global Cartesian coordinate system. By using the chosen primitive (**P1** – **P4**), the CPR Trajectory Solver guides the camera towards its goal position.

d) Execution of the new primitive. The chosen primitive executes over certain number of sampling intervals. During the execution of the primitive, the CPR Trajectory Solver constantly reviews the position and velocity orientation of the camera relative to the goal point of the primitive. The chosen primitive is executed until the camera reaches its goal point. The CPR Trajectory Solver establishes this by constantly checking if camera has finished the previously defined primitive, i.e. reference trajectory. In the period of execution of any primitive, the position and velocity orientation of the object are unknown.

The procedures c) and d) of the CPR Trajectory Solver are executed cyclically during the tracking and monitoring the object in real time.

e) Stopping the program. This routine is started by the operator by pressing **END** button. In this way, the CPR Trajectory Solver runs the **DEC** process along **P4** primitive which is in the direction of camera's motion, as defined at the beginning of the **DEC** process. The camera slows down from the reference to zero velocity and the tracking and monitoring process is over. As it can be seen in Fig. 4, after execution of each primitive or choosing the **ACC** or **DEC** processes, the CPR Trajectory Solver checks if the operator has activated the **END** button and in that case the whole process of object tracking and reference trajectory generation is stopped. When the operator activates the **END** button, the CPR Trajectory Solver waits for the current primitive to finish and after its execution it starts the process of stopping the object monitoring and following.

At any moment, execution of the primitive can be stopped when the operator activates the **Safety**

Volgina 15, 11060 Belgrade, Serbia
Phone: (+381 11) 2771-398, 2772-876
Fax: (+381 11) 2776-583
WWW.INSTITUTEPUPIN.COM
email: info@institutepupin.com

button.

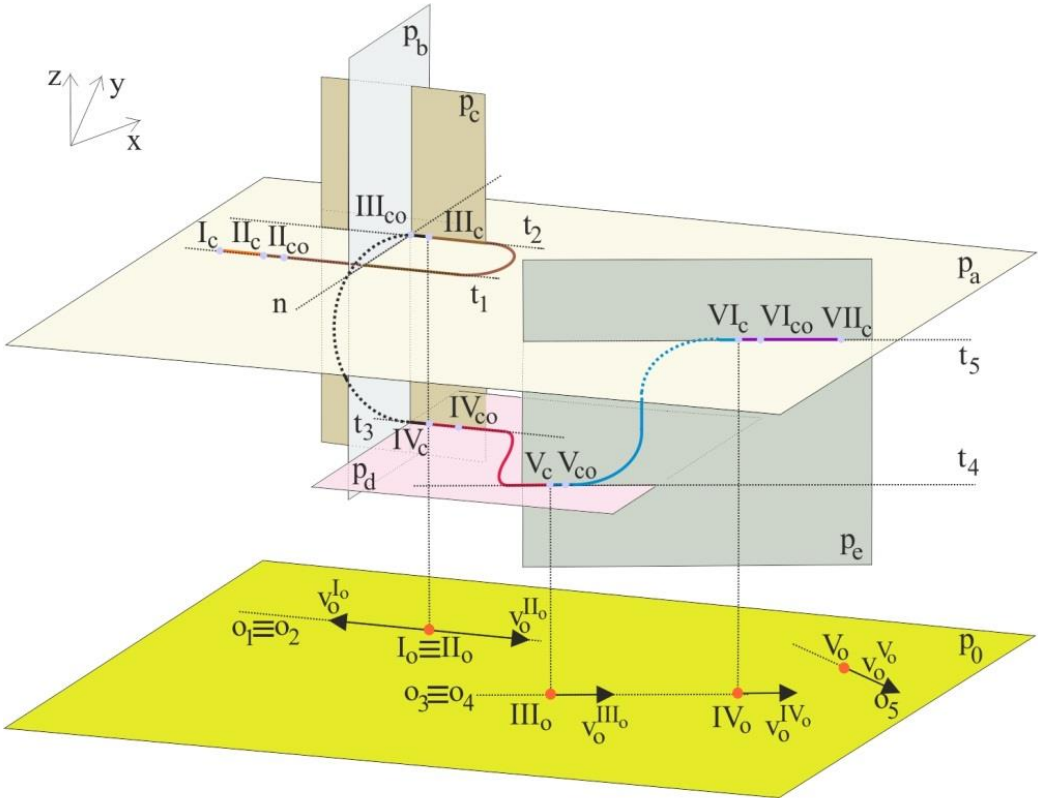


Figure 5. 3D trajectory defined in real time.

Certain primitive requires a certain amount of time. Time of execution of the primitive is conditioned by object's velocity. It is presumed that maximal velocity of object is $v_{omax} \leq v_{cmax}$, where v_{cmax} is camera's maximal velocity. Regardless of the chaotic motion of the object, for definition of v_{omax} the most unfavourable situation was assumed: the object moves between previously and currently detected object's position via straight line. In this fashion, camera's tracking of the object is ensured.

Results of object tracking are smoothly interconnected aforementioned primitives, which can be seen in Figs. 1, 5, and 6, which form a complex reference trajectory of the camera's motion. The complex reference trajectory of the camera is determined sequentially, based on the positions and velocity orientations of the camera and object at certain moments. We will once more emphasize that all the labels of positions and velocities are the same in Figs. 1, 5 and 6 and in Table 3 as well. The first research which was based on principles set in program system CPRTS was published in [1]. The software package **CPRTS** is used to verify the validity of the generated mathematical model.

Volgina 15, 11060 Belgrade, Serbia
Phone: (+381 11) 2771-398, 2772-876
Fax: (+381 11) 2776-583
WWW.INSTITUTEPUPIN.COM
email: info@institutepupin.com

This research was supported by the Ministry of Education, Science and Technological Development, Government of the Republic of Serbia through the following two projects: Grant TR-35003, "Ambientally intelligent service robots of anthropomorphic characteristics", by Mihajlo Pupin Institute, University of Belgrade, Serbia, and Grant OI-174001, "The dynamics of hybrid systems of complex structure", by SANU Institute Belgrade and Faculty of Mechanical Engineering University of Nis, Serbia.

References

[1] Ljubino Kevac, **Mirjana Filipovic**, Aleksandar Rakic: The trajectory generation algorithm for the cable-suspended parallel robot—The CPR Trajectory Solver, ROBOT AUTON SYST 94C, 25-33, DOI: http://dx.doi.org/10.1016/j.robot.2017.04.018, 2017.

Volgina 15, 11060 Belgrade, Serbia
Phone: (+381 11) 2771-398, 2772-876
Fax: (+381 11) 2776-583
WWW.INSTITUTEPUPIN.COM
email: info@institutepupin.com