

ROBOTIC TESTING PLATFORM FOR AUTONOMOUS RENDEZVOUS AND DOCKING OF FLOATING VEHICLES

*Atheel Redah⁽¹⁾, Harish Rao Ramavaram⁽¹⁾, Sergio Montenegro⁽¹⁾

⁽¹⁾Chair of Aerospace Information Technology, University of Würzburg,

Josef-Martin-Weg 52/2, 97074 Würzburg, Germany, E-Mail: redah@informatik.uni-wuerzburg.de

ABSTRACT

In order to test and validate autonomous space systems, testbeds for on the ground experimental validation of autonomous proximity navigation and docking maneuvers of robotic vehicles in a frictionless space-like environment are needed. In this paper, a robotic testbed that can auto track two floating vehicles' movements on a testing platform at all times is presented. In this testbed, an uninterrupted air and power supply can be provided for two experimental air bearing vehicles in order to ease the process of testing different control algorithms and strategies for space rendezvous, docking, and formation flying. At the same time, critical information about the statuses of the monitored vehicles will be provided by the tracking system to be used for the purpose of optimizing the control algorithms.

Keywords: Robotic Testbed, Floating Vehicles, Optical Tracking, Mathematical Modeling, PID Algorithm.

1. INTRODUCTION

The ability to use space agents (i.e., satellites, robots, modules, and spacecraft) that can perform autonomous rendezvous and docking maneuvers with other orbiting elements for different purposes, such as refueling, on-orbit servicing, debris and retired elements removal or on-orbit self-assembly to build larger and more efficient systems are becoming increasingly attractive for the development and operation of future space systems.

In order to demonstrate and validate such space applications, the chair of Aerospace Information Technology at the University of Würzburg has developed air bearing vehicles able to navigate in an almost frictionless environment with three degrees of freedom consisting of two components of translation and one angle of rotation. The vehicle uses pressure release nozzles as steering thruster actuators and one reaction wheel to control the orientation of the vehicle. Flat circular air bearing pads which produce frictionless motion on a flat

and smooth surface of glass plates for microgravity environment are used. The propulsion air supply subsystem of the developed floating vehicle consists mainly of compressed high-pressure air tanks, inline pressure reducers, proportional valves and pressure sensors to control the flow rate to the air bearing pads and to the thrusters [1].

Testing different control algorithms and strategies in such systems is not a comfortable process as it has to be interrupted very often because of the limitation of the air supply inside the mounted high-pressure air tanks. Therefore, a robotic testing platform with a mechanism to track two vehicles at a time using two cameras mounted on two movable supplying arms has been developed as shown in Fig. 1.

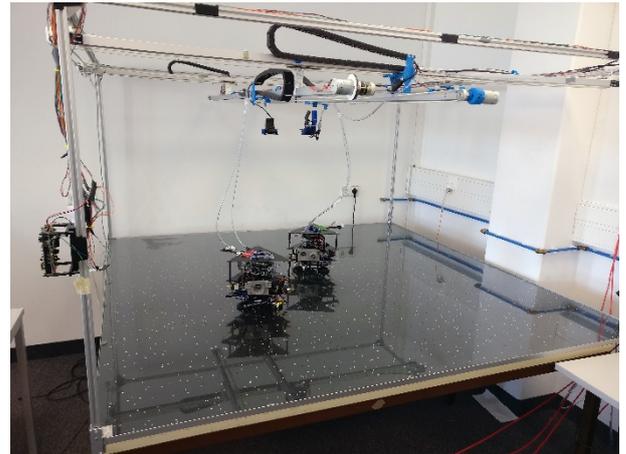


Figure 1: Robotic Testing Platform

The air is supplied to the tracked vehicles through very flexible air tubes coming from the supplying arms which are always centered above the vehicles to minimize the effect of the disturbance forces that the vehicles may encounter during their movement. The air is supplied from a nearby rotary screw air compressor which allows the air supply to be continuous and without any interruption. A color-coded object detection technique is used to determine the vehicle position and velocity on the platform. These information are used to control four motors that drive the robotic platform. In addition, a Graphical User Interface (GUI) has been developed

to provide full control over both the object detection and the tracking control aspects, along with the ability to monitor the process in real time.

2. SYSTEM DESCRIPTION

In the developed testbed, a 2x2x1 meter mechanical structure of two movable guide rails with an attached chassis is placed above a 2x2 meter testing platform covered with a 10[mm] thick glass plate. The attached chassis are used to hold the cameras for tracking the vehicles' movement on the platform as well as for supplying the air to the vehicles through mounted tubes. A starscape poster based on the Hipparcos catalog is placed beneath the glass surface to help the vehicles navigate on the platform through onboard star tracker sensor system. Each of these guide rails are attached at two joints to another moving belt mechanisms that can move in the axis perpendicular to the motion of the guide rails which enable the supply arms to move along the horizontal plane above the platform. As shown in Fig. 2, motors 3 and 4 are connected to their respective connecting rods and controlling the movement of the belt mechanisms perpendicular to the motors axis through pulleys which enable controlling the movement of the supply arms along the X-axis. At the same time, motors 1 and 2 are connected to their respective rails which enable controlling the movement of the supply arms along the Y-axis. This gives the freedom for each of the supplying arms to move along the entire Y-axis two-meter range and to the location of the other guide rail along the X-axis on the platform.

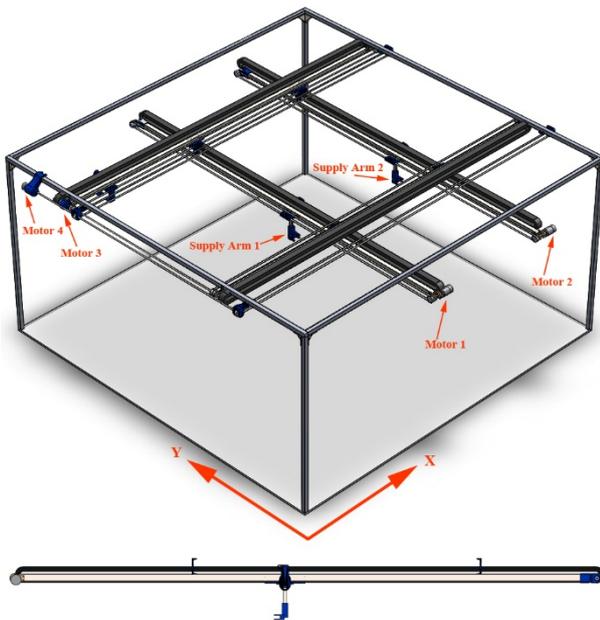


Figure 2: Robotic Testing Platform Mechanical Structure

DC motors that are able to supply sufficient torque to move the 5 Kg guide rails with the attached supplying arms are used. Brushed DC motors have been selected due to their cost effectiveness and the fact that there can easily be controlled with Pulse Width Modulation (PWM) signals through simple H-bridge power drivers.

A Camera is mounted to each of the two supply arms and connected to the Control and Monitoring Station (CMS) using Universal Serial Bus (USB) cables. The cameras support up to video stream resolution of 1280x720 pixels. However, only 640x480 pixels at the center of the frame are used for image processing in order to reduce the field of view of the camera.

The control and monitoring station acquires the video stream from these two cameras and performs the necessary image processing operations onboard in order to isolate the tracked objects and calculate their position and velocity information [2].

The calculated tracked information are then transmitted to a local Robotic Control Unit (RCU) through USB and Bluetooth/Wifi interfaces. The robotic control unit is consisting of a STM32F4 discovery microcontroller board featuring 32-bit ARM Cortex M4 with FPU core and an extension development board. The extension development board contains 4 H-bridges with the necessary power circuits to drive the motors with onboard Bluetooth and Wifi modules that can be used to transfer information from/to the control unit. The microcontroller generates four PWM signals that represents the magnitude of the voltage need to be applied to the motors along with the polarity signals depending on the control signals calculated by the controllers. The Real-time Onboard Dependable Operating System (RODOS) have been used at the robotic control unit. The Operating System was designed using object-oriented C++ interfaces [9]. RODOS is designed for applications demanding high dependability with simplicity as its main strategy. Information exchange takes place among the components of the software through the RODOS middleware which uses the publisher-subscriber mechanism [10]. A predefined protocol for communication inside RODOS called the Gateway is being used to transfer information between the Robotic Control Unit and the Control and Monitoring Station.

The control and monitoring platform is developed using C++ on a QT environment. Image processing operations are done using OpenCV library which has been integrated into the QT Creator. The

control and monitoring GUI can be shown in Fig. 3.



Figure 3: Control and Monitoring GUI

3. MATHEMATICAL MODELING AND CONTROL

The robotic testing platform can be treated as four distinct linear belt drive models. Each of these models consist of a DC motor, reduction gear, two pulleys and a toothed belt that drives the mass. The drive pulley is connected to the drive shaft of the reduction gear which is in turn connected to the motor shaft and the driven pulley rotates freely. The mathematical modeling can be broken down into two unique parts namely electrical modeling and mechanical modeling. The electrical part primarily deals with the relationship between the voltage, current and torque produced by the DC motor, while the mechanical part relates the torque, forces and the motion associated with the linear belt drive [2].

The mathematical model of the DC motor can be shown in Eq. 1.

$$L \frac{di}{dt} + Ri = V - K_t \dot{\phi} \quad (1)$$

In Eq. 1, L is the motor electric inductance, R is the motor electric resistance, K_t is the motor torque constant, ϕ is the motor shaft angular position, i is the motor armature current and V is the applied voltage to the motor.

The forces that are acting on the linear belt drive during operation are shown in Fig. 4 [2]. The pulley P1 is attached to the drive shaft of the reduction gear, which is in turn attached to the motor shaft. The pulley P2 is free to rotate about its axis. The belt is treated as a spring with its spring constant proportional to its length. This causes the segments of the belt attached to the Mass to have different spring constants as the mass moves in either direction. In order to simplify the mathematical model, the following assumptions have been made: pulleys P1 and P2 have identical inertial properties

(J_p) and size (R_p); the belt can be treated as a massless spring; the belt does not slide over the pulley; the motor has high dynamic torque response.

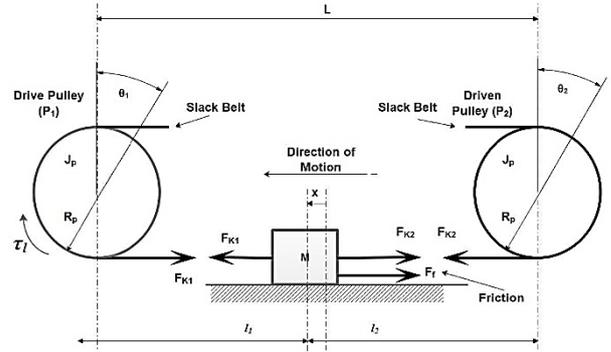


Figure 4: Forces on the Spring-Mass-Pulley model

The overall force equation of the belt drive model is shown in Eq. 2.

$$(R_p \theta_1 - x)K_1 - (x - R_p \theta_2)K_2 - F_f = M \ddot{x} \quad (2)$$

In Eq. 2, R_p is the radius of the pulley, θ_1 is the Angle of the drive pulley P1, θ_2 is the Angle of the drive pulley P2, x is distance traveled by the belt drive block, M is the Mass of the belt drive block, K_1 and K_2 are the spring constants of the belt.

Basic reduction gear setup is shown in Fig. 5 [2]. The larger gear is attached to the pulley P1 and therefore has the same angular motion.

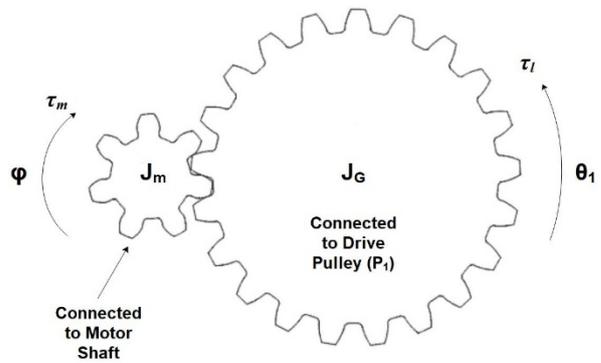


Figure 5: Reduction Gear and Motor Shaft

The overall torque equation of the belt drive model is shown in Eq. 3.

$$(J_m + GJ_G + GJ_p) \ddot{\phi} = \tau_m - \tau_f - (R_p G \phi - \frac{J_p}{K_2 R_p} \ddot{\theta}_2 - R_p \theta_2) K_1 R_p \quad (3)$$

In Eq. 2, J_m is moment of inertia of the motor shaft, J_G is moment of inertia of the gear, J_p is the moment of inertia of the pulley, G is the gear ratio, τ_m is the motor torque and τ_f is the friction torque.

The mathematical model shown in Eq. 3 is highly nonlinear and can be simplified with few reasonable assumptions: assuming that x is small, the spring constants K_1 and K_2 would be equal and can be replaced by K ; assuming the inertia of the pulleys J_p to be negligible compared to the one of the motor shaft and the reduction gear allows the driven pulley to perfectly follow the mass. The resulting mathematical equations of the linearized belt drive system are shown in Eq. 4 and 5.

$$(J_m + GJ_G)\ddot{\theta} = \tau_m - \tau_f - (R_p G \theta - x)K R_p G \quad (4)$$

$$M\ddot{x} = (R_p G \theta - x)K - F_f \quad (5)$$

There are two major sources of friction in the belt drive model. One of which acts as a resisting force between the sliding arm and the guide rail. The other acts as a resisting torque at motor-gear-pulley interfaces. The frictional forces can be classified mainly into three: Coulomb friction, Static friction and Viscous friction. Coulomb friction acts between two sliding surfaces with its magnitude directly proportional to the normal force acting between those two surfaces. In the current application, the normal force between the sliding arm and guide rail is expected to be constant as there are no inclined surface or major intermittent loads. Static friction acts between two surfaces when a force is applied to them as they are not in relative motion. The magnitude of static friction depends on the normal force between those surfaces and is marginally higher than the Coulomb frictional force. Viscous friction is directly proportional to the relative velocity between two sliding surfaces.

In order to measure the friction coefficients, a simple approach was adopted. The underlying principle behind this approach is that when the sliding arm moves at a constant speed (steady state), the torque generated by the motor is equal to the opposing frictional forces in the system. In addition, we know that the torque generated by the brushed DC motor is directly proportional to the current. The resulting mathematical equation that relate the friction forces with motor torque at steady state can be shown in Eq. 6.

$$(f_c + \mu_v \dot{x})R_p = GK_t i \quad (6)$$

In Eq. 6, f_c is the Coulomb friction force and μ_v is the Viscous friction coefficient.

It can be noticed from Eq. 6 that by performing a linear fit between current drawn by the motor at different voltages and measuring the corresponding speeds we can obtain both f_c and μ_v . The velocity of the sliding arm was measured using the tracking

system. The camera setup is calibrated to provide reasonably accurate position and velocity of the object that it tracks. Steady state current values of all the motors that are used to drive the two arms have been measured along with the no load current of the DC Motor. In addition, the corresponding velocities of the robotic arms in relationship with the current drawn by the motors are also measured as shown in Fig. 6 [2].

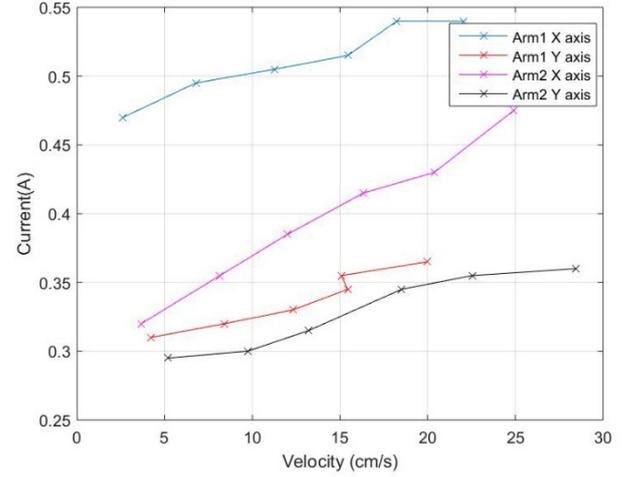


Figure 6: Arm Velocity vs. Motor Current Plot

It can be noticed from Fig. 7 that the current draw is significantly higher on the X-axis motors of both the arms. This is because the X-axis arms carry the entire rail on which the Y-axis arms move causing them to be significantly heavier and consequently experiencing more frictional force.

Proportional-Integral-Derivative (PID) controllers are used to calculate the appropriate voltages that are to be applied to the motors of the supplying arms. A PID controller is the most common feedback loop component in industrial control systems. This controller is proven to be robust and extremely beneficial in the control of many applications [3]. PID expression refers to the three controller correcting calculations, whose sum constitutes the output of the PID controller. The Proportional part is used to eliminate the immediate error, where the error signal is multiplied by the proportional constant. The Integral part is used to eliminate the steady state error (past error), where the error signal is integrated (added up) over a period of time, and then multiplied by the integral constant. The Derivative part is used to compensate for future error, where the first derivative of the error signal (the slope of the error) is calculated over time, and multiplied by the derivative constant.

One main problem with the PID controller is that

whenever the reference input is changed in a step form, the error signal will have a corresponding step change as well. The sudden step change of the error signal can cause the derivative of the error signal to be infinite. This is known as the derivative kick. In order to eliminate the derivative kick and avoid any unwanted rapid changes in the control signal due to any change in the desired process value, the derivative term can be based on the system output only. At the same time, the proportional term of the controller can suffer from a proportional kick, where the proportional term can respond strongly to a step change in the reference input. Although, the proportional kick is much less severe than the derivative kick, it may be too aggressive for some systems. In addition, the use of a summation to calculate the contribution of the integral term can lead to a phenomenon known as integral windup. As a larger disturbance than the system is capable of handling effects the controlled process, the error between the measured process value and the reference input value cannot be eliminated in a short time. This situation can be problematic due to the integral term of the controller. The integral term will sum up as long as this situation last causing long periods of saturation in the control signal. The extended velocity form of the PID controller shown in Fig. 7 [3] has been utilize as it has the ability to eliminate a possible proportional or derivative kick and ensure that the integral windup problem will not occur. The only drawback is that the controller will respond to the reference input's step changes through the integral term only, which makes the controller respond slower to the reference input changes.

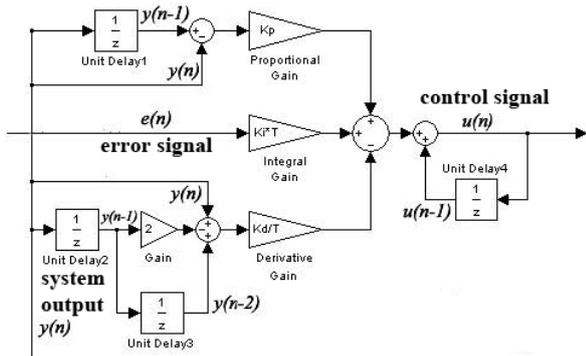


Figure 7: The Simulink Block Diagrams of the Discrete Extended Velocity PID Controller Algorithm

The simulated and measured response for the movement of the supplying arm in the X-Axis is shown in Fig. 8 [2].

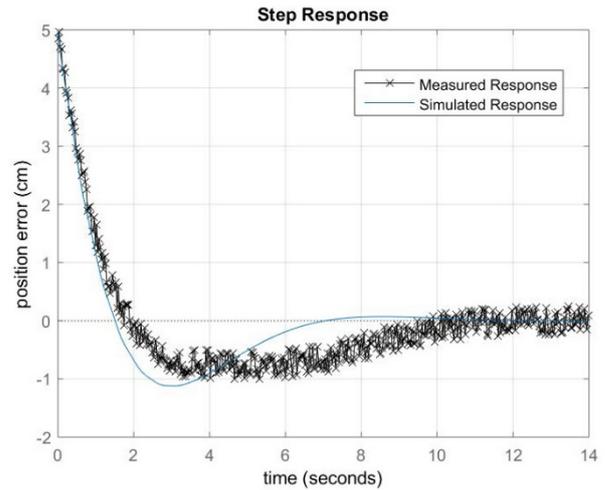


Figure 8: Simulated and Measured Response for the Movement of the Supplying Arm in the X-Axis

4. THE TRACKING SYSTEM

The determination of position and velocity of the floating vehicles that the air supply arm needs to track is done optically with the help of cameras [2]. The three widely used methods for optically tracking an object are colour coded object detection, frame differential method and pattern recognition. Colour coded object detection is the simplest and the least computationally intensive method [4]. It works on the principle of filtering out the video stream in order to extract a specific predefined colour. This method is fast and works very efficiently in a controlled environment where the colour of the object that needed to be tracked is unique with respect to the background and if there are no unwanted objects with the same colour. However, this method cannot be reliably used to track objects in an uncontrolled environment such as tracking objects on a street. The frame difference method works on the principle of comparing two consecutive frames of the video stream in order to determine the object that is moves relative to a stationary background. Multiple moving objects can be detected using this method [5]. However, isolating the intended target can become difficult if there is more than one moving object. This method fails if the entire background moves along with object that is to be tracked or if the object to be tracked is stationary [6]. Pattern recognition works on the principle of recognizing a predefined pattern on the object. The pattern can be a simple geometric share or a complex pattern such as a 2D barcode [7]. The cameras need to have high resolution if complex patterns are to be processed. This is by far the most effective method but is also

very computationally intense. This method fails to determine the target if part of the pattern is obstructed partially. In the current application, the frame difference method cannot be used because the camera is moving with the air supply arm which will cause the entire background of the frame to change. Pattern recognition method was also ruled out because there is a high likelihood of the pattern being partially obstructed by the supplying tubes that supply high pressure air to the floating vehicles. Therefore, the only logical solution was to use the much simpler colour coded method for object tracking. The test bench on which this system is used behaves as a reasonably controlled environment with a maximum of two floating vehicles moving at any given time. The two vehicles can be colour coded with two separate colours enabling them to be tracked by two different cameras. The other advantage of using this method is that even if the colour code is partially obstructed, this method works very well. The image processing operations that have been performed are shown in Fig. 9.

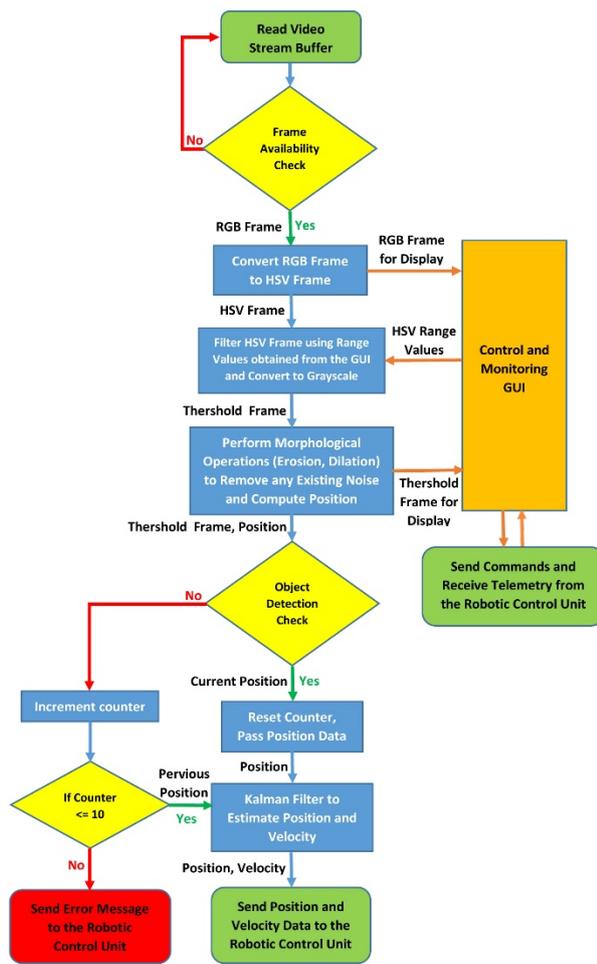


Figure 9: Tracking System Operation Flowchart

Kalman filter is a widely used algorithm for estimating the states of dynamical systems with noisy measurements. Once the object has been detected using the colour coded method, the x and y pixel location of the object is obtained. The pixel location is then transformed into distance in centimeters after careful calibration. The obtained data is noisy and the velocity information is not available. Calculating velocity from discrete noisy position data is not ideal. Kalman filter is quite useful in these kind of scenarios. The data flow of the Kalman filter with the equations that are used to estimate the states [8] are shown in Fig. 10.

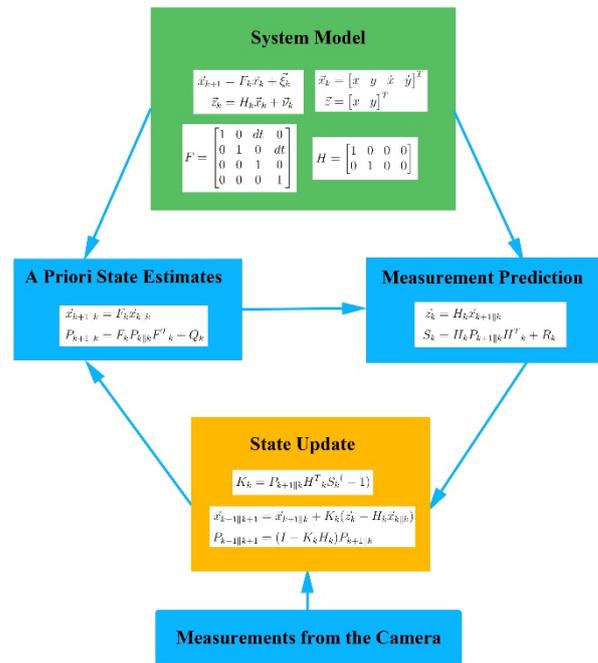


Figure 10: Kalman Filter Operation

A Comparison of velocity estimated with and without the usage of Kalman filter is shown in Fig. 11 [2].

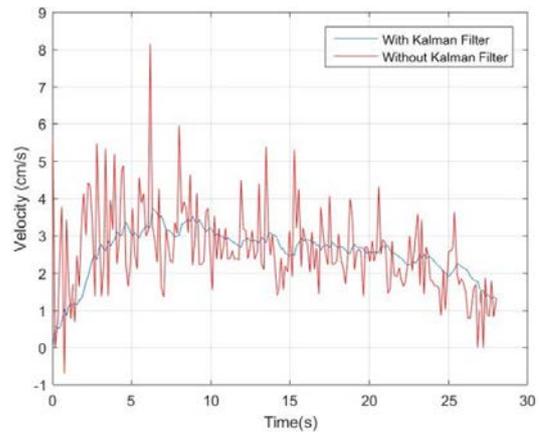


Figure 11: Velocity Estimation using Kalman Filter

5. CONCLUSION

This paper has presented the developed robotic testing platform for autonomous rendezvous and docking of two floating vehicles. In this platform, an uninterrupted air supply can be provided using two moveable supplying arms. The mechanical structure of the platform and the system operation information are explained and detailed. The mathematical model of the system is presented. Each air supply arm is mathematically modeled as a separate linear belt drive that controls the X and Y directions independently. The belt drive model consists of a drive pulley, driven pulley and a mass connected by an elastic spring. A simple approach was adopted to measure the friction coefficients. An improved discrete form of the designed PID controller is presented. A comparison between the simulation model time response and the real time measured data is shown. In this platform, two cameras are used to track the movements of the floating vehicles. The image processing method for tracking the vehicles and determining their position and velocity is explained. In addition, a Kalman filter has been used to improve the result. An overview of the used filter and the results obtained is presented.

Acknowledgement

The Robotic Testing Platform is part of the YETE2/SIW project, which is funded by the Space Agency of the German Aerospace Center (DLR) with federal funds of the German Federal Ministry of Economics and Technology (BMWi) under 50RA1734.

References

- [1] Redah, A., Mikschl, T., Montenegro, S., "Physically Distributed Control and Swarm Intelligence for Space Applications", i-SAIRAS 2018, Madrid, 2018.
- [2] Ramavaram, H. R., "Software and Control Design for 2-D Floating Satellite Simulator", Master Thesis, University of Würzburg, Germany, 2018.
- [3] Redah, A., "Modelling and Development of a Magnetic Levitation System for Fault-Tolerance Demonstration of a Hard Real-Time Avionics", ISBN-13: 978-3639461954, 2010.
- [4] Shahbaz Khan, F., Anwer, R., van de Weijer, J., Bagdanov, A., Vanrell, M. and Lopez, A., "Color Attributes for Object Detection", IEEE Conference on Computer Vision and Pattern Recognition, 2012.
- [5] Zhan, C., Duan, X., Xu, S., Song, Z. and Luo, M., "An Improved Moving Object Detection Algorithm Based on Frame Difference and Edge Detection", Fourth International Conference on Image and Graphics (ICIG), 2007.
- [6] "Detecting Moving Objects, Ghosts, and Shadows in Video Streams", IEEE Transactions on Pattern Analysis and Machine Intelligence, 25(10), pp.1337-1342, 2003.
- [7] Bishop, C., "Pattern Recognition and Machine Learning", New York: Springer.
- [8] Gelb, A., "Applied Optimal Estimation", Cambridge, Mass.: MIT Press, pp.102-142, 2006.
- [9] Montenegro, S., and Dannemann, F., "RODOS: Real Time Kernel Design for Dependability", DASIA 2009 DATA Systems In Aerospace, Istanbul, 2009.
- [10] Faisal, M., Redah, A., Montenegro, S., "Design of a Remote-Cockpit for small Aerospace Vehicles", 11th IAA Symposium on Small Satellites for Earth Observation, 2017.