ECE490: Virtual Servoing Project Proposal

Matthew Ebert
V00884117
dept. of Electrical and Computer Engineering
University of Victoria
ebertmx@gmail.com

Abstract—

A device is needed to test virtual servoing methods. A ball and beam device provides a visual and easily developed control system for testing and development of virtual servo methods. Additionally, the abundant literature and examples of ball and beam devices throughout control theory allow for easy comparison. The requirements of the device include simplicity, control stability, real-time operation, and embedded design. The proposed design will use an ESP32-CAM microcontroller which, with its dual core processor and integrated camera, can read and process frames while controlling a dc motor. Research and development of the device is required in the areas of image and position virtual servoing, geometric camera calibrations, and developing a concurrent FreeRTOS program to implement on the ESP32-CAM.

Keywords— virtual servoing, control systems, ball and beam, ESP32-CAM

1. Introduction

Virtual Servoing (VS) is the application of computer vision (CV) in control processes. Using cameras and CV image analysis, a feedback signal is constructed for use in a control loop. When processing power is limited, these types of system can be difficult to implement. To test the application of VS, a realized physical control system is needed. The apparatus must be simple and dependent on CV. Additionally, this device must test the limits of these methods in a real-time embedded system.

This proposal outlines the scope, requirements, basic design, evaluation methods, and research requirements of system for the purpose of VS study.

2. Project Scope

The objective of the design is to create a realized system which demonstrates the basic techniques of virtual servoing. As such, the project will remain simple to limit scope.

2.1 The Task

The goal statement of the project follows,

Create a simple control system which, by using a camera and CV techniques, demonstrates the possibilities and limitations of virtual servoing. The design should mimic a fundamental system from the field of control theory.

This statement allows for a broad range of solutions, but it is limited to simple, standard control systems which allow for easy comparison between virtual servoing and classical control techniques.

2.2 Requirements

The functional requirements of the design are outlined in *Table 1*.

Table 1: Functional Design Requirements of the Control Project

R No.	Title	Description
1	Stable Control	The system should respond to input to control or drive a system to a steady, stable, and final state which corresponds to the input.
2	1 DoF	The system should have 1 degree of freedom such that only a single control loop is necessary.
3	Embedded Design	The system should be self-contained and use only integrated hardware (microcontroller, camera, motor ect.) to accomplish the task.
4	Portable	The system will be used to demonstrate virtual servoing; thus, it must be mobile and possible to deploy in many situations.
5	Maximum Vision	The system will use camera(s) as the primary feedback sensor. Although other sensors might be necessary, they should be kept to a minimum.
6	Multiple Configurations	The system should allow for several configurations to demonstrate different methods of virtual servo including hand-in-eye and hand-to-eye setups.
7	Real Time	The system should operate smoothly in real time
8	Accuracy	The system should control the device to within 10% of the accuracy of the least accurate component (Camera or motor resolution).
9	Speed	The system should complete the task as fast as possible with the hardware.
10	User Interface	The system should have an integrate UI which allows for easy setup and control.
11	Data Storage	The controller should store and save state, performance, and configuration data for later analysis.

2.3 Control Requirements

During the design phase, the target criteria for settling time and overshoot will be added to requirements. Currently, reasonable values depend on the implemented system. The final report on the implemented design will include detailed requirements for the device including these criteria.

2.4 Operation Enviroment Considerations

In the field of computer vision, inconsistent environments (backgrounds, lighting, changing objects) can cause problems and added computation in systems. Since this design is embedded, limited computer resources are available. As such, a consistent and easily differentiable environment should be used to focus the project more on the use of VS and less on the image analysis.

3. Proposed Design

The proposed design consists of a ball and beam system. A CAD model of the design is shown below.

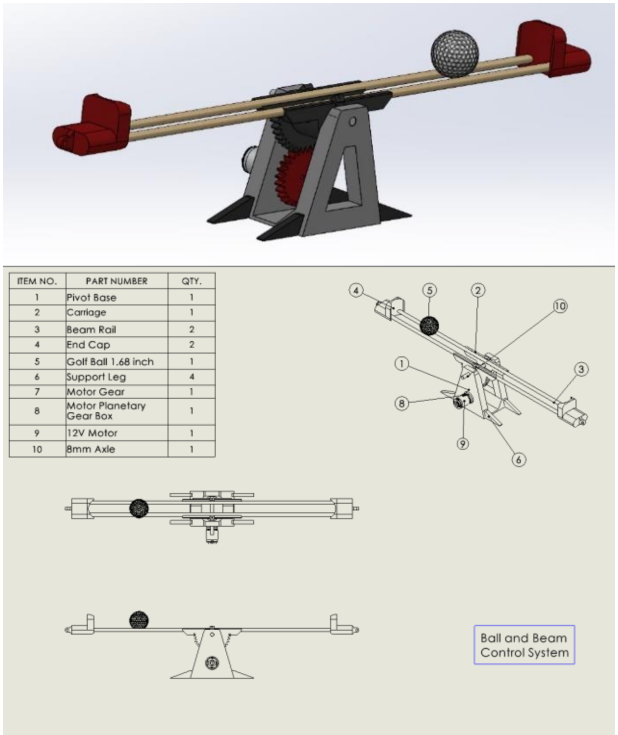

Figure 1: Ball and Beam control system for implementation with visual servo.

The apparatus mainly consists of a beam, pivot base, drive gear, and DC motor. The beam rotates on the pivot base creating a variable incline. A ball on the beam changes velocity and position in response to the angle.

3.1 Camera Positioning

A camera can be integrated in two possible locations: externally with a 2D view of the apparatus or attached to 1 end of the beam. These two mounting locations will demonstrate eye-to-hand and eye-in-hand virtual servo methods.

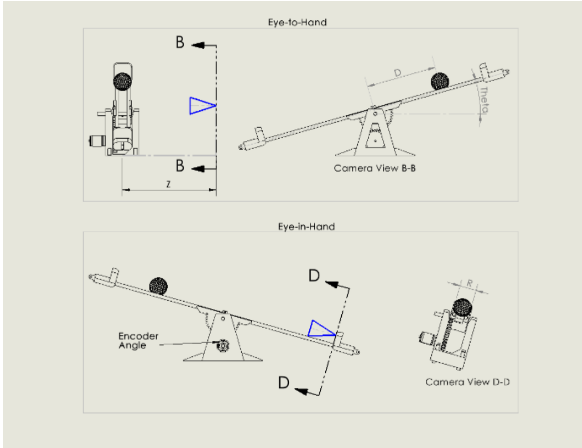

Figure 2: Possible camera mounting positions

For the Eye-to-Hand configuration, the camera will be mounted a known distance away from the apparatus. The camera views the ball (with a slight occlusion) and the ramp angle. The camera can detect the dynamic properties of the ball (speed, position, acceleration) by comparing the center of mass of the ball in different frames. The ramp angle can be found by comparing the two end points of the beam.

For the Eye-in-hand configuration, the camera will view a full profile of the ball along the two beam rails. Based on its distance from the camera, the ball will appear as a particular radius R. Since the true radius of the ball will be known, the system can determine the balls distance and speed based on the camera parameters, relative radius, and previous frames of the ball. For this implementation, the camera will have no sense of the angle of the beam. Consequently, the system requires a calibration step and the addition of an encoder to keep track of the angle.

3.2 Control Implementation

To demonstrate the visual control system, the ball will be positioned in a random location on the beam. One can then enter a desired location (such as the center of the beam) into the device. The system will respond by changing the angle of the beam with the DC motor to move the ball to the desire location. A control response can be implemented to change how the ball approaches and settles at the target location.

Two control response methods have been considered for this design: PID and Fuzzy Logic.

3.2.1 Classical PID Control

The figure below shows a classical control loop of the proposed device

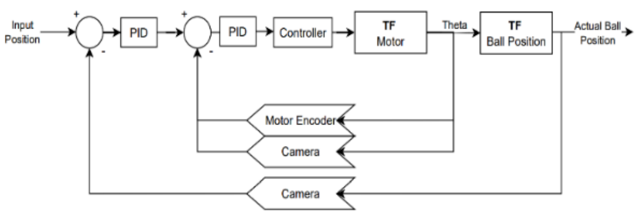

Figure 3: Control loop for proposed ball and beam control system. Uses PID control

This proposed design consists of two combined control system: Control of the motor position and control of the moving ball. The motor transfer function relates the output PWM voltage signal from the controller to the motor angle. The ball position transfer function relates the angle of the beam to the position of the center of the ball on the beam. The TF for motor position control is well covered in literature [1].

$$\frac{\alpha(s)}{V(s)} = \frac{K}{s\big((Js+b)(Ls+R)\big)+K^2}$$

Symbol	Description
α	Motor angle
V	Input Voltage (PWM controlled in this case)
K	Motor torque and back emf constant
J	Moment of inertia of the rotor
b	Viscous friction constant
R	Motor Resistance
L	Motor Inductance

For the ball and beam system, we can derive the TF from the equation of motion of the ball and the parameters of the system as shown below.

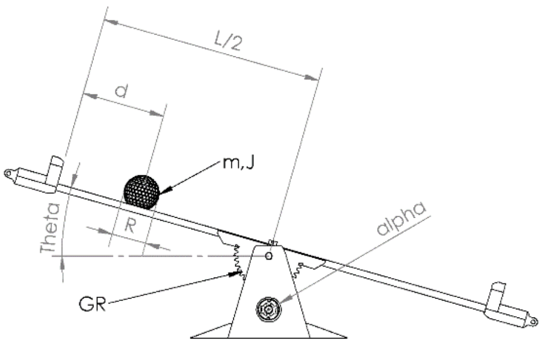

Figure 4: Parameter definitions for proposed ball and beam system

Symbol	Description
L	Length of beam
d	Ball distance from beam end
theta	Beam angle
R	Ball Radius
m	Ball mass
J	Ball moment of inertia
GR	Gear ratio
alpha	Motor angle

This system is linearized with the *small angle approximation* to produce the following transfer function

$$\frac{D(s)}{\alpha(s)} = \frac{mg(GR)}{s^2(\frac{J}{R^2}+m)} \quad [2]$$

3.2.2 Fuzzy Logic Control

An alternative to PID control is fuzzy logic control. This type of controller uses observations rather than dynamics to control the process.

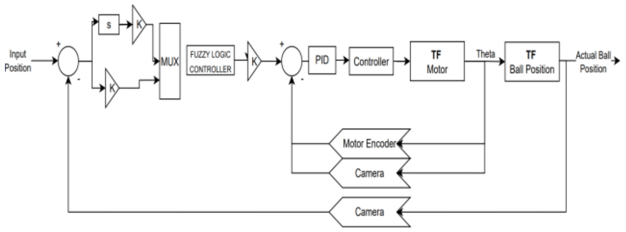

Figure 5: Control loop system for fuzzy logic control

This type of controller has proven effective with non-linear systems [3]**.** More research must be conduct on fuzzy logic design but based on a survey of publications it seems viable [3] [4].

3.3 Hardware and Software

The primary components of this assembly include an ESP32-Cam microcontroller, 12V Brushless motor with encoder and PWM supported driver, an 8mm axle and bearing set, 3D printed gears and components, and 3/8in wooden dowels.

The ESP32-Cam controller is the most important device. It has the following specifications

- Dual Core 160MHz processor
- 9 GPIO ports (Supports PWM)
- Wi-Fi and Bluetooth
- 2 Megapixel Camera
- Max 60 fps image transfer rate

The main advantage of this micro controller is the concurrent programming capabilities. For this application, the microcontroller can process frames and actuate the motor simultaneously. The 60-fps camera will most likely exceed the processing power of the ESP32 and likely not limit the system.

The ESP32 is compatible with the FreeRTOS C++ system. Thus, the CV portion of the control system can be implemented with the OPENCV library and multi cored C++ programming.

4. EVALUATION AND TESTING

Since a camera is required to detect the ball, the system requires visual servoing for function. Evaluating the success of the design becomes simple: Can the system control the location of the ball?

More extensive testing will test the requirement satisfaction and the limitations of the design.

Table 2 shows the initial fundamental tests for the system

Table 2: Test for evaluation of the proposed design.

Test No.	Title	Description	Expected Result
1	Ball Control	Place the ball in a random location on the beam except the target position. Input the location of the middle of the beam into the system. Repeat for several different starting locations	The system should cause the ball to move in response to the control signal and stop the ball on the desired location within a certain margin.
2	Stability	Repeat the steps for test 1. Change the response parameter of the system incrementally starting at a mild response and rising to a point where overshoot and response time exceed the limits of the mechanical system such as maximum beam angle and length.	The system should respond in accordance with the change of parameters. Within the mechanical limits the ball should always reach a steady state point within error margins of the target location.
3	Step Response	Repeat the steps for test 1. Save the position data to a file for later analysis. Plot the position of the ball over time until steady state is reached for several seconds.	The ball should reach its steady state position within the overshoot and settling time criteria.
4	Ramp Response	Place the ball on 1 end of the beam. Input a ramp input to the system.	The ball should track the current position with a margin of error until it reaches the end of the beam.
5	Max FPS	Disable the motor. Place the ball on the ramp and manually move on end of the beam up and down. Have the system operate as normal to a step input at the center of the beam. Record the FPS processed by the ESP32.	The ESP32 should process frames at a minimum rate of 15 FPS.

5. Research and Development

Research on visual servo implementation will be required. Several topics have been identified as necessary including

1. **Image based VS (IBVS):** A type of VS which uses image space at a known distance from the target to create a control signal
2. **Position based VS (PBVS):** A type of VS which finds relative cartesian coordinates from image data to control a system.
3. **Geometric Video Calibration:** Calibration with the camera lens to account for distortion created by camera optics.
4. **Fuzzy Logic Control Systems:** A type of control process which uses human observation to tune a controller

Research of these areas will influence the control software of virtual servo system and will be the focus at the beginning of the project.

The control software will be the largest area of development. It will involve concurrent programming, image processing, and control loop implementation. The real-time requirement of this system puts a heavy emphasis on efficient code. Due to the computationally expensive nature of CV, a fast detection and tracking technique must be developed (or employed). Since the background will be fixed and easy to identify, thresholding may be used to reduce video frames to binary and improve speed.

The other major systems involve OTS components or can be 3D printed.

6. Conclusion

The objective of the design project is to create a control system to demonstrate virtual servoing. The proposed design consists of a ball and beam system which positions a ball using a 12V DC motor and an ESP32-CAM microcontroller device. This design has two camera mounting configurations suitable for eye-in-hand and eye-to-hand control. PID or Fuzzy logic will be used to tune the response of the control system. Primary success will be determined by the completion of the task at any level; in this case, success mean moving the ball to steady state at the desired location. Evaluation of the design performance will involve detail testing. Research must be conducted into several VS techniques including IBVS and PBVS. The major area of development for this project will be implementing the control loop with image analysis on the ESP32 microcontroller.

7. References

[1] The University of Michagan, Carnegie Mellon University, "Example: Modeling DC Motor Position," [Online]. Available: https://www3.diism.unisi.it/~control/ctm/examples/motor2/motor.html. [Accessed 20 06 2022].

[2] University of Michigan, Carnegie Mellon University, University of Detriot Mercy, "Control Tutorials for Matlab and Simulink," [Online]. Available: https://ctms.engin.umich.edu/CTMS/index.php?example=BallBeam&section=SystemModeling. [Accessed 20 June 2022].

[3] M. M. Gouda, S. Danaher and C. P. Underwood, "FUZZY LOGIC CONTROL VERSUS CONVENTIONAL PID CONTROL FOR CONTROLLING," in *IFAC Computer Aided Control Systems Design*, Salford, 2000.

[4] M. Abdullah and Z. Shareef, "Fuzzy Logic Control of Ball and Beam System," in *2nd International Conforence on Education Technology and Computer (ICETC)* , 2010.