DRIVER ASSISTED STEERING SYSTEM FOR REVERSING AN ARTICULATED VEHICLE

R.N. RANAWEERA

M.Sc.Eng. **2019**

DRIVER ASSISTED STEERING SYSTEM FOR REVERSING AN ARTICULATED VEHICLE

R.N. RANAWEERA

M.Sc.Eng.

Department of Electrical and Electronic Engineering,
Faculty of Engineering,
University of Peradeniya,
Peradeniya, Sri Lanka 2019

This thesis has been submitted to the University of Peradeniya as a partial fulfillment of the requirements for the degree of Master of the Science of Engineering.

Declaration

I declare that this thesis has been composed solely by myself and that it has not been submitted, in whole or in part, in any previous application for a degree. Except where states otherwise by reference or acknowledgement, the work presented is entirely my own.

R.N. Ranaweera
(PG/EE/17/MSc/37)

Acknowledgements

I would like to thank my supervisors Dr. D.H.S. Maithripala, Senior Lecturer, Dept. of Mechanical Engineering, Faculty of Engineering, University of Peradeniya and Dr. J.V. Wijayakulasooriya, Senior Lecturer, Dept. of Electrical & Electronic Engineering University of Peradeniya, for the valuable guidance they provided me during my MSc study and research.

Besides them I would like to thank Prof. A.D. Polpitiya, Vice President for Academic Affairs, Sri Lanka Technological Campus and Mr. Senevirathne, Director (Engineering), ART Logistics (Pvt) Ltd for the enormous support and encouragements they provided me during this research.

Also, I would like to thank Sri Lanka Technological Campus (Pvt) Ltd., the National Science Foundation and ART Logistics (Pvt) Ltd for the financial support they provided for this research.

Abstract

Truck overloading has become a major issue in Sri Lanka. It causes roads and bridges to break due to the heavy loads. As a solution to this, relevant authorities will impose axle load limits to the trucks in the near future. With that, truck owners will be able to transport a limited load with a single truck which will ultimately decrease their profits. However, as a solution for this issue, a truck can be used with a trailer attached to transport the goods while keeping the axle load limits.

However heavy transportation trailers need wheels at the front of the trailer as well. Hence after attaching the trailer to the truck, there will be two pivot joints. Reversing such a system is an extremely difficult task. When reversing, the trailer moves in an unpredictable direction with respect to the truck's movement. Most of the time, vehicle will fold into a position it cannot come out without a forward motion which is called the jackknife effect. Hence the reversing moment is very complex. Even the professional drivers with a high skill level will fail when it comes to reversing. For such a vehicle, reversing is almost impossible without any guidance.

As a solution to this problem this thesis presents the development and implementation of a controller for fully automated and semi-automated reversing of a truck and trailer system with two pivot joints. The controller is verified through simulations. Experimental validation

of the results for the controller also done using a fully autonomous small scale prototype, and a semi-autonomous (driver assisted) larger scale prototype.

The developed driver assisted system with the help of the controller, could assist the driver by providing the steering guidelines for stable reversing. Therefore the driver can reverse the vehicle without being worrying about the trailer behavior.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

Currently in Sri Lanka considerable amount of trucks are in operation in various industries. However to gain the maximum profit, the owners are used to overload the trucks than the rated values which ultimately increase the axle load. This truck overloading has becoming a major issue in Sri Lanka. It causes roads and bridges to damage due to the heavy loads.

Hence, as a solution to this problem the authorities are considering the strike imposition of axle load limits. With such restrictions, truck owners will be able to transport a limited load with a single truck which will ultimately decrease their profits. However, one possibility of overcoming this limitation is the use of a truck with a trailer attached to transport the goods while keeping the rated axle load limits. Thus the use of truck and trailer systems, also known as articulated vehicles, will become a vital part of the industry in Sri Lanka. For similar reasons articulated vehicles have already become very popular in major industrial countries of the world.

Choosing articulated vehicles over rigid vehicles has other major advantages as well. These vehicles can take much sharper turns than similar sized rigid vehicles and it is possible for the wheels to keep in contact with the ground even when the surface is uneven. Since these

vehicles can carry more load, cost of transportation can be reduced with a reduced fuel consumption [8],[5].

Heavy transportation trailers need wheels at the front of the trailer as well. Hence after attaching the trailer to the truck, there will be two pivot joints. That is needed especially to ensure lane keeping in narrow roads. Training a driver to control such a vehicle takes a lot of effort and time [6]. Especially when reversing, the trailer moves in an unpredictable direction with respect to the truck movement and this unpredictability is very high with two pivot points. Hence the reversing movement is very complex [3]. In certain cases when reversing, there is more tendency for the vehicle to fold into a position it cannot come out without a forward motion. This is called the jackknife effect [1]. Even the professional drivers with a high skill level can fail when it comes to reversing a double pivot articulated vehicle. It is almost impossible to reverse the vehicle without any guidance [7].

Hence it will be beneficial if there is some way to assist the driver by providing the necessary guidelines for accurate reversing. Then the driver can reverse the vehicle along the desired path without worrying about the trailer behavior. Developing such a driver assisted steering system is the ultimate objective of this research.

Published results for multi trailer systems are significantly less compared to single trailer systems. Though driver assisted systems are available for single body vehicles, for multibody vehicles, the technology it is still under development [7].

Derivation of the kinematic equations of motion for an N-trailer configuration can be found in [2][7][4], while single trailer systems are considered in [3][9][5]. The paper [1] proposes a new hybrid feedback control scheme for two trailer system, considering low level controls for backward driving along a line and a curve while [6] presents an approach for stabilization and path tracking for a two trailer configuration using an LQ-controller that stabilizes the internal angles and a pure pursuit controller is used for path tracking. The work presented in [2] solves the motion planning problem for an N-trailer system by converting the system into chained form. A driver assistance system for backward docking of an N-trailer vehicle is discussed in [7][4], using a cascaded VFO control law as a control assistance function while [8] proposes a novel approach to control multiple trailers connected to the front bumper of a truck.

The controller used in the fully autonomous prototype in this research uses an onboard vision based localization system with the help of computer vision. The developed manual driver assisted system also incorporates computer vision to produce the guidance instructions.

To the best of my knowledge, a compact on-board vision based solution for a two trailer reversing problem is not available. Most of the controllers depend on external vision based localization techniques [7][4] while some processing is done outside the system [6].

First part of this thesis presents a development of controller for such a system with simulation verification. Then then it presents the experimental validation results of the controller for two cases: fully autonomous small scale prototype, and semi-autonomous (driver assisted) larger scale prototype. Finally it presents the improvements to the current system and the computer vision techniques used.

Chapter 2

Mathematical model of the system

2.1 Notation

A truck and trailer system with pivots can be represented as in *Figure 2.1.* The system is represented as a single pivoted two trailers attached to a truck, which gives a clearer impression. However, a single trailer with two pivots, attached to a truck will be equivalent to the system represented in the *Figure 2 .1.*

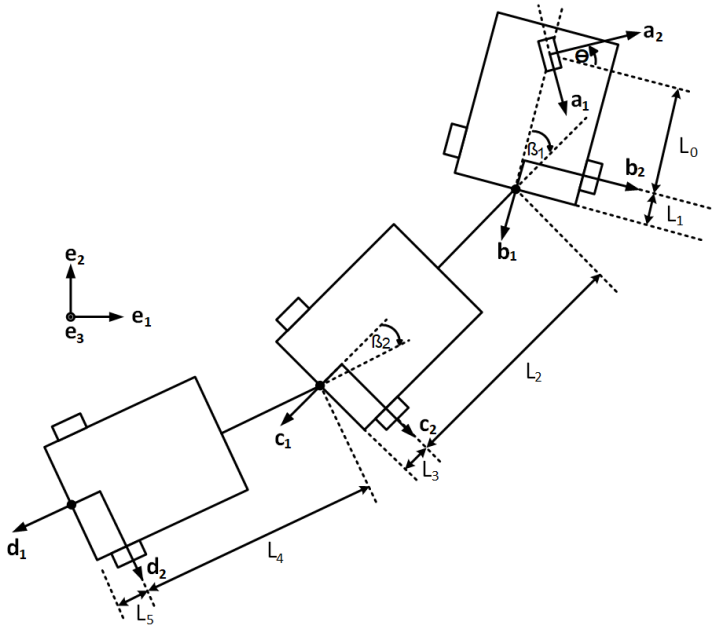

Figure 2.1: Reference frames for the trailer system analysis

To analyze the truck and trailer system, reference frames are placed on each moving body and the steering axle. When deriving the controller, initially it is assumed that vehicle has a

one imaginary front wheel. The calculated steering angle can be mapped to get the real front wheel steering angles using the Ackermann formula as explained in section 2.3. The earth frame is represented by frame ***e***, as marked in *Figure 2.1*.

The rotational matrix R_z is defined as the rotation around axis 3 in any reference frame.

$$\mathbf{e} = \begin{bmatrix} e_1 \\ e_2 \\ e_3 \end{bmatrix} \quad R_z(\theta) = \begin{bmatrix} \cos(\theta) & -\sin(\theta) & 0 \\ \sin(\theta) & \cos(\theta) & 0 \\ 0 & 0 & 1 \end{bmatrix}$$

$X(L)$ represents a translation L along the axis 1 for any reference frame.

$$X(L) = \begin{bmatrix} L \\ 0 \\ 0 \end{bmatrix}$$

2.2 Coordinate system

To find a relationship between each moving body, initially the equations can be derived with respect to the earth frame. Coordinates of each reference frame on the system with respect to the earth frame can be written as,

$$\begin{aligned} o_b &= o_a + R_z(\alpha - \theta)X(L_0) \\ o_c &= o_a + R_z(\alpha - \theta)X(L_0 + L_1) + R_z(\alpha - \theta - \beta_1)X(L_2) \\ o_d &= o_a + R_z(\alpha - \theta)X(L_0 + L_1) + R_z(\alpha - \theta - \beta_1)X(L_2 + L_3) \\ &\quad + R_z(\alpha - \theta - \beta_1 - \beta_2)X(L_4) \end{aligned}$$

Where O_a is the coordinates of "***a***" reference frame with respect to the earth frame "***e***".

Then the velocities of the origins of each reference frame with respect to the earth frame can be written as,

$$
\begin{aligned}
\dot{o_a} &= R_z(\alpha)\begin{bmatrix} V_{a_1} \\ 0 \\ 0 \end{bmatrix} \\
\dot{o_b} &= \dot{o_a} + R_z(\alpha-\theta)\Omega_z(\dot{\alpha}-\dot{\theta})X(L_0) \\
\dot{o_c} &= \dot{o_a} + R_z(\alpha-\theta)\Omega_z(\dot{\alpha}-\dot{\theta})X(L_0+L_1) \\
&\quad + R_z(\alpha-\theta-\beta_1)\Omega_z(\dot{\alpha}-\dot{\theta}-\dot{\beta}_1)X(L_2) \\
\dot{o_d} &= \dot{o_a} + R_z(\alpha-\theta)\Omega_z(\dot{\alpha}-\dot{\theta})X(L_0+L_1) \\
&\quad + R_z(\alpha-\theta-\beta_1)\Omega_z(\dot{\alpha}-\dot{\theta}-\dot{\beta}_1)X(L_2+L_3) \\
&\quad + R_z(\alpha-\theta-\beta_1-\beta_2)\Omega_z(\dot{\alpha}-\dot{\theta}-\dot{\beta}_1-\dot{\beta}_2)X(L_4)
\end{aligned}
$$

where V_{a1} is the vehicle moving velocity along "$\boldsymbol{a_1}$" direction. For a reversing task this will be a very small value compared to normal vehicle driving velocity. "$\boldsymbol{\Omega_z(\theta)}$" is defined as,

$$\Omega_z(\theta) = R_z(\theta)^T \dot{R}_z(\theta)$$

Velocities of the origins of each reference frame in their 1,2,3 directions with respect to the earth frame can be written as,

$$
\begin{aligned}
V_b &= R_z(\alpha-\theta)^T \dot{o_b} \\
V_c &= R_z(\alpha-\theta-\beta_1)^T \dot{o_c} \\
V_d &= R_z(\alpha-\theta-\beta_1-\beta_2)^T \dot{o_d}
\end{aligned}
$$

2.3 Ackermann steering angle calculation

The Ackermann steering mechanism is used to reduce the tire slippage when taking curvatures. **w** is the lateral wheel separation and $\mathbf{L_0}$ is the longitudinal wheel separation. $\mathbf{\Phi_1}$ is the relative steering angle of the outer wheel and $\mathbf{\Phi_2}$ is the relative steering angle of the inner wheel and **R** is the distance between **ICC** (instantaneous center of curvature) and the vehicle center. With the above notation the Ackermann steering equation can be derived as following,

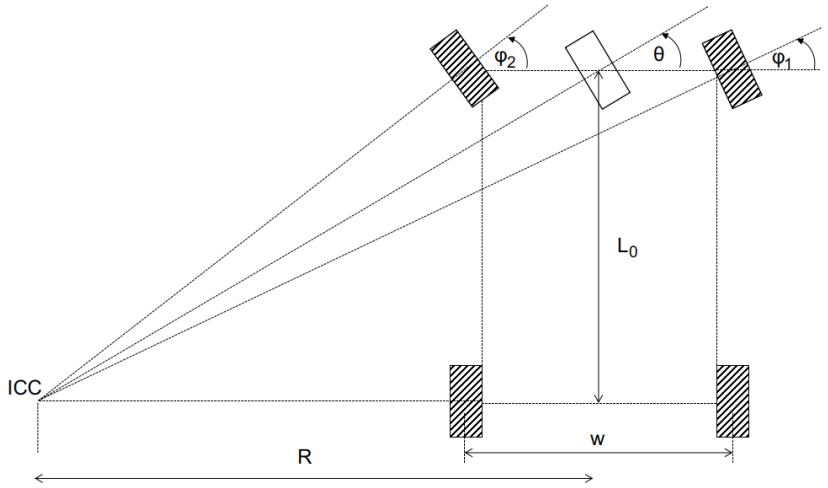

Figure 2.2: Ackermann steering mechanism

$$\tan(\theta) = \frac{L_0}{R}$$

$$\tan(\phi_1) = \frac{L_0}{R + w/2}$$

$$\tan(\phi_2) = \frac{L_0}{R - w/2}$$

By combining above equations, following equations can be derived to find the actual steering angles of the wheels when **θ** is known.

$$\frac{L_0}{tan(\theta)} = \frac{L_0}{tan(\phi_1)} - \frac{w}{2} = \frac{L_0}{tan(\phi_2)} + \frac{w}{2}$$

$$\phi_1 = \tan^{-1}\left(\frac{2L_0\tan(\theta)}{2L_0 + w\tan(\theta)}\right)$$

$$\phi_2 = \tan^{-1}\left(\frac{2L_0\tan(\theta)}{2L_0 - w\tan(\theta)}\right)$$

2.4 Differential drive velocity and steering angle mapping

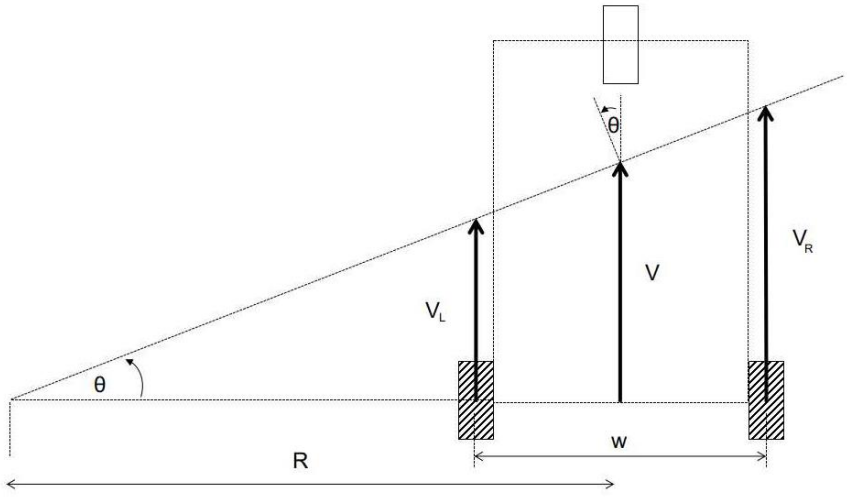

Figure 2.3: Differential drive velocities

When using a differential drive mechanism steering angle (**θ**) can be mapped to the wheel speeds as explained below. **V** is the center velocity and $\mathbf{V_L}$ and $\mathbf{V_R}$ are the left and right wheel velocities accordingly.

$$tan(\theta) = \frac{V}{R} = \frac{V_L}{(R - w/2)} = \frac{V_R}{(R + w/2)}$$

By using above equations, left and right wheel velocities can be expressed in terms of steering angle and center velocity.

$$V_L = \frac{2V - w\tan(\theta)}{2}$$
$$V_R = \frac{2V + w\tan(\theta)}{2}$$

2.5 Stabilizing the system with the controller

Considering the no slip condition for the truck and trailer system, following constraints can be made,

$$V_{b2} = 0 \quad (1)$$
$$V_{c2} = 0 \quad (2)$$
$$V_{d2} = 0 \quad (3)$$

Where V_{b2}, V_{c2} and V_{d2} are the velocities of relevant reference frames along direction 2. From those constraints following relationships can be found.

From 1,

$$\dot{\alpha} = \frac{L_0\dot{\theta} - V_{a_1} sin(\theta)}{L_0}$$

From 2 and 3,

$$\dot{\beta_1} = -\frac{\mathrm{L_2\,V_{a_1}\,\sin\theta - L_0\,V_{a_1}\,\sin\beta_1, \cos\theta + L_1\,V_{a_1}\,\cos\beta_1\sin\theta}}{\mathrm{L_0\,L_2}}$$

$$\dot{\beta_2} = -\frac{F}{(L_0 L_4)}$$

Where F is given by,

$$\begin{aligned} F = (&L_0 L_4 \dot{\beta}_1 + L_4 V_{a_1} sin(\theta) \\ &+ L_0 L_2 \dot{\beta}_1 cos(\beta_2) + L_0 L_3 \dot{\beta}_1 cos(\beta_2) \\ &+ L_2 V_{a_1} cos(\beta_2) sin(\theta) + L_3 V_{a_1} cos(\beta_2) sin(\theta) \\ &- L_0 V_{a_1} sin(\beta_1 + \beta_2) cos(\theta) + L_1 V_{a_1} cos(\beta_1 + \beta_2) sin(\theta)) \end{aligned}$$

Using small angle approximation, the linearized version of the system can be represented in the state space form using the above equations,

$$\dot{x} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}$$

$$\begin{bmatrix} \dot{\beta}_1 \\ \dot{\beta}_2 \end{bmatrix} = V_{a_1} \begin{bmatrix} \frac{1}{L_2} & 0 \\ \frac{-(L_3+L_4)}{L_2 L_4} & \frac{1}{L_4} \end{bmatrix} \begin{bmatrix} \beta_1 \\ \beta_2 \end{bmatrix} + V_{a_1} sin(\theta) \begin{bmatrix} \frac{-(L_1+L_2)}{L_0 L_2} \\ \frac{L_1(L_3+L_4)}{L_0 L_2 L_4} \end{bmatrix}$$

$$A = V_{a_1} \begin{bmatrix} \frac{1}{L_2} & 0 \\ \frac{-(L_3+L_4)}{L_2 L_4} & \frac{1}{L_4} \end{bmatrix}$$

$$B = V_{a_1} \begin{bmatrix} \frac{-(L_1+L_2)}{L_0 L_2} \\ \frac{L_1(L_3+L_4)}{L_0 L_2 L_4} \end{bmatrix}$$

$$u = sin(\theta)$$

Y=Cx

There were two prototypes built for the validation of the algorithms. One is small scale and other one is industrial scale. For the industrial scale prototype lengths in centimeters were,

$L_0 = 122$, $L_1 = 32$, $L_2 = 74$, $L_3 = 0$, $L_4 = 106$.

Then it will give A and B matrices as,

$$A = \begin{bmatrix} 0.0135 & 0 \\ -0.0135 & 0.0094 \end{bmatrix} \quad B = \begin{bmatrix} -0.0117 \\ 0.0035 \end{bmatrix}$$

Further it can be showed that the V_{a1} doesn't affect the eigenvalues of the system.

$$A = V_{a_1} \begin{bmatrix} 0.0135 & 0 \\ -0.0135 & 0.0094 \end{bmatrix}$$

$$|A - \lambda I| = V_{a_1} | \begin{bmatrix} 0.0135 - \lambda & 0 \\ -0.0135 & 0.0094 - \lambda \end{bmatrix} |$$

$$\lambda^2 - 0.0229\lambda + 0.0001269 = 0$$

Since V_{a1} cancels out, theoretically the vehicle speed doesn't affect the system stability considering the small angle approximation.

Eigenvalues of matrix A are $\lambda_1 = 0.0094$ and $\lambda_2 = 0.0135$. Since eigenvalues are in the right half plane, the system is unstable.

2.6 The controller

Since the system is unstable in general without any controller, it can be stabilized by using a state feedback controller in the following form.

$$\theta = -K \begin{bmatrix} \beta_1 \\ \beta_2 \end{bmatrix}$$

Suitable value for gain K can be found using the root locus analysis of the system.

Figure 2.4 shows the root locus diagram of the system considering the dimensions of the industrial scale prototype.

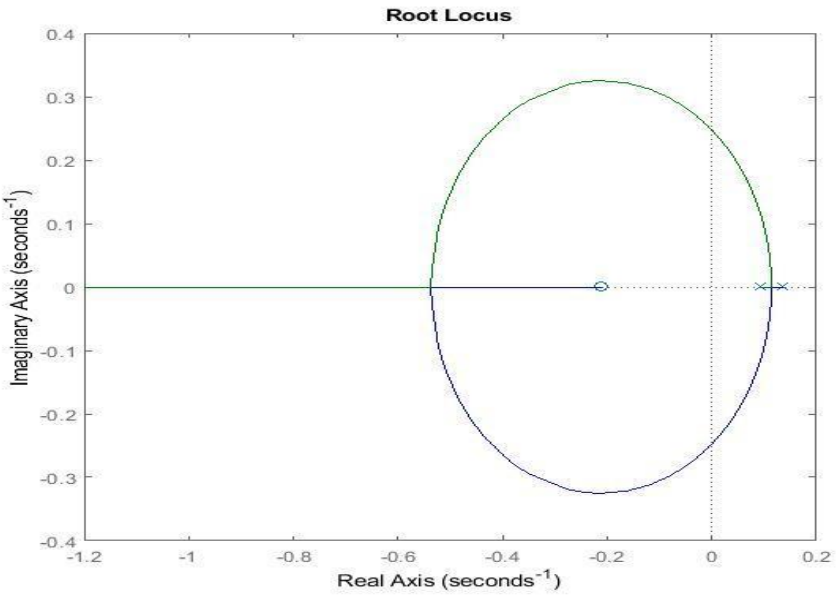

Figure 2.4: Root locus diagram for the system

By picking up values in the left half plain, K will be

$$K = [-6.7730\ 6.3263]$$

Which places the poles at

$$P_1 = -0.001 \text{ and } P_2 = -0.078$$

Even though for the linearized system $u = \sin(\theta)$. For the easiness of implementation $u = \theta$ is taken. Since θ is the steering angle it will go usually less than 45 degrees (0.78 rad). Therefore, the produced error will be small and we can assume that $\sin(\theta) = \theta$. The relevant error for this assumption is shown in the *Figure 2.5*. The simulations also gave the expected results when using the assumption $u = \theta$.

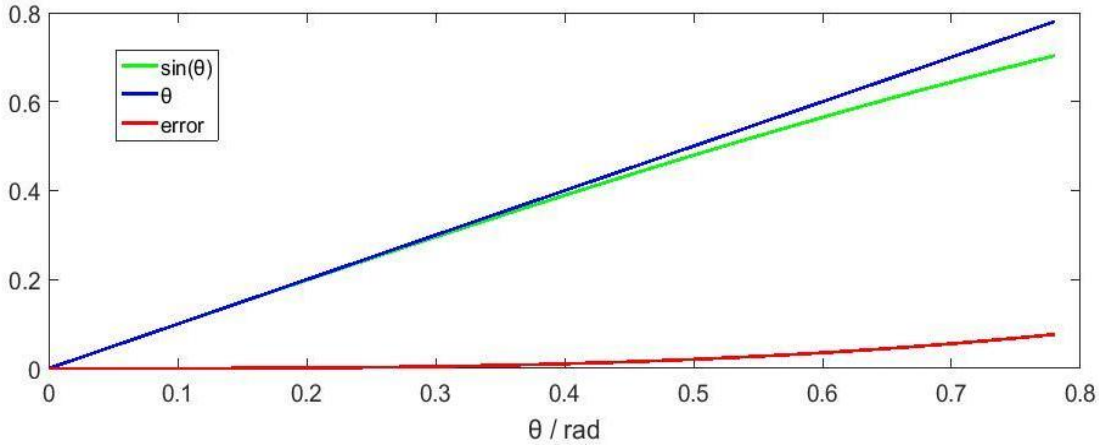

Figure 2.5: $sin\ (\theta)$ and θ values

Chapter 3

Simulation results

3.1 Stabilizing the system from a given orientation

The linearized system is in the form

$$\dot{x} = \mathbf{A}\mathbf{x} + \mathbf{B}\mathbf{u}$$

$$\begin{bmatrix}\dot{\beta}_1 \\ \dot{\beta}_2\end{bmatrix} = V_{a_1}\begin{bmatrix}\frac{1}{L_2} & 0 \\ \frac{-(L_3+L_4)}{L_2L_4} & \frac{1}{L_4}\end{bmatrix}\begin{bmatrix}\beta_1 \\ \beta_2\end{bmatrix} + V_{a_1}sin(\theta)\begin{bmatrix}\frac{-(L_1+L_2)}{L_0L_2} \\ \frac{L_1(L_3+L_4)}{L_0L_2L_4}\end{bmatrix}$$

The system was initially tried to stabilize (ie: to make β_1 and β_2 values go to zero) using the state feedback controller.

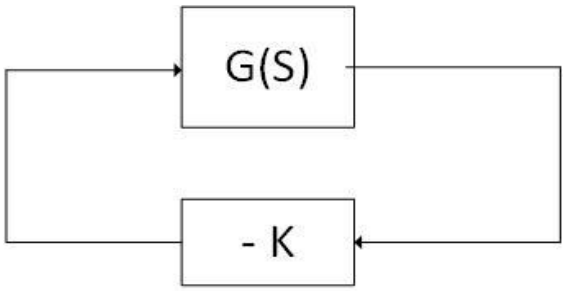

Figure 3.1: Block diagram of the state feedback controlled system

The simulations were generated correspond to an industrial scale prototype dimensions which is used to verify the control algorithms. The goal is to make sure that the system is stabilizing from an initial position.

Initial conditions for the prototype were taken as in *Table 3.1*. Also the reversing speed taken as $V_{a1} = 100\ mm/s$.

Table 3.1: Initial conditions

$x_{(0)} = 0$	$y_{(0)} = 0$	$\alpha_{(0)} = 1.2$	$\dot{\theta}_{(0)} = 0$	$\beta_{1(0)} = 0.2$
$\beta_{2(0)} = -0.2$	$\dot{\alpha}_{(0)} = 0$	$\dot{\beta}_{1(0)} = 0$	$\beta_{2(0)} = -0.2$	$\theta_{(0)} = -K \begin{bmatrix} \beta_{1(0)} \\ \beta_{2(0)} \end{bmatrix} = 0$

Figure 3.1 demonstrates that the block diagram of the controller which makes the pivot angles to go to zero. Then the trailer system is able to reverses along a straight line while stabilizing the system. *Figure 3.2* shows how the pivot angles behave with the steering angle when reversing. *Figure 3.3* shows the relevant vehicle travelling path.

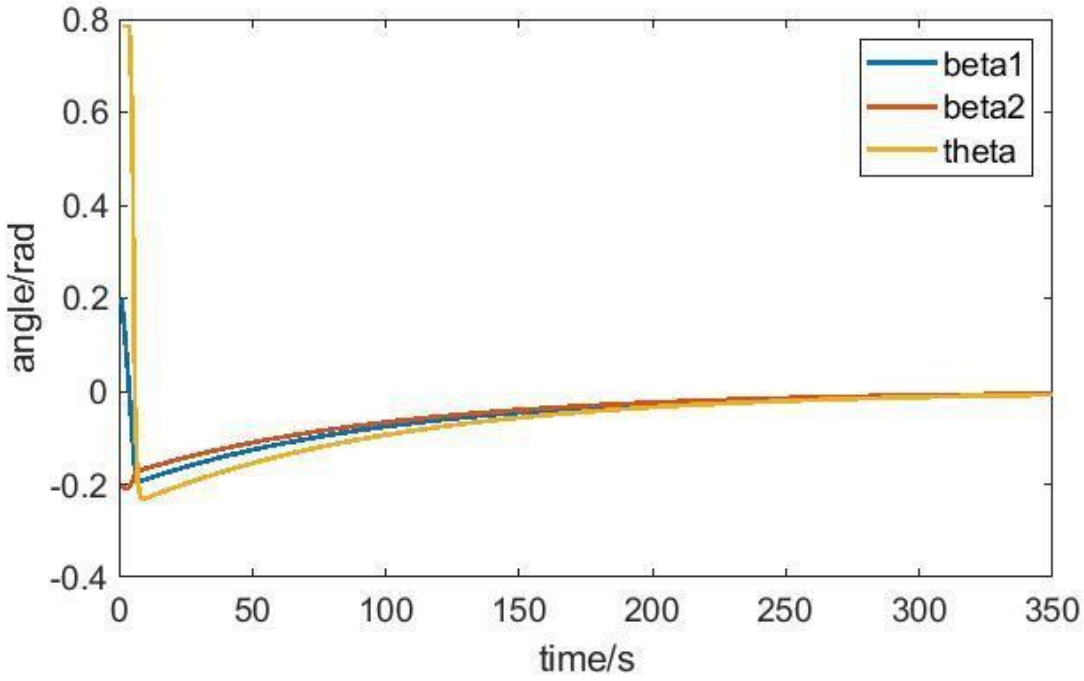

Figure 3.2: Variation of pivot angles with steering angle when stabilizing

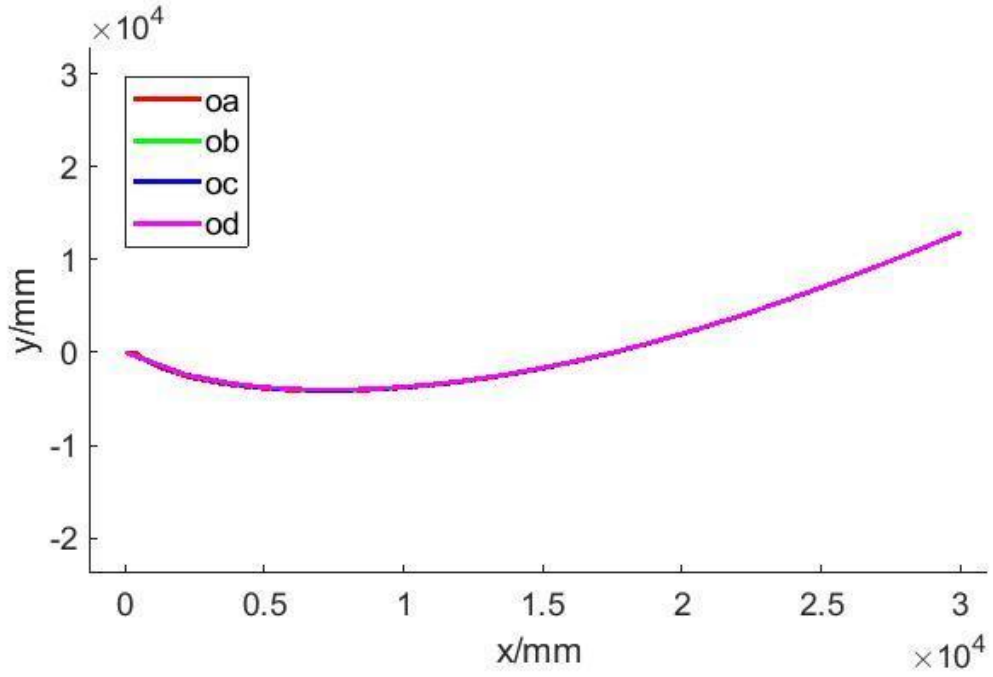

Figure 3.3: Vehicle travelling path when stabilizing

3.2 Taking curved paths

To stabilize the system while introducing small curvatures, control law of u = −kx can be changed to u = k($x^* - x$) where x^* is the reference which need to be tracked. With the reference pivot angles. New control law will be,

$$\theta = K \begin{bmatrix} (\beta_1^* - \beta_1) \\ (\beta_2^* - \beta_2) \end{bmatrix}$$

Above control law can be rearranged as following,

$$\begin{aligned} \theta &= [k_1\ k_2] \begin{bmatrix} (\beta_1^* - \beta_1) \\ (\beta_2^* - \beta_2) \end{bmatrix} \\ &= k_1(\beta_1^* - \beta_1) + k_2(\beta_2^* - \beta_2) \\ &= -(k_1\beta_1 + k_2\beta_2) + k_1\beta_1^* + k_2\beta_2^* \\ &= -[k_1\ k_2] \begin{bmatrix} \beta_1 \\ \beta_2 \end{bmatrix} + (k_1\beta_1^* + k_2\beta_2^*) \\ &= -[k_1\ k_2] \begin{bmatrix} \beta_1 \\ \beta_2 \end{bmatrix} + d \end{aligned}$$

When *d* is set to a certain value, pivot angle references will be automatically set. It will cause system to stabilize while introducing a curvature. The size of the curvature will depend on the value of *d* which can also be positive or negative for taking clockwise or counter clockwise curvatures. Hence, the control law for taking curvatures can be represented as following.

$$\theta = -K \begin{bmatrix} \beta_1 \\ \beta_2 \end{bmatrix} + d$$

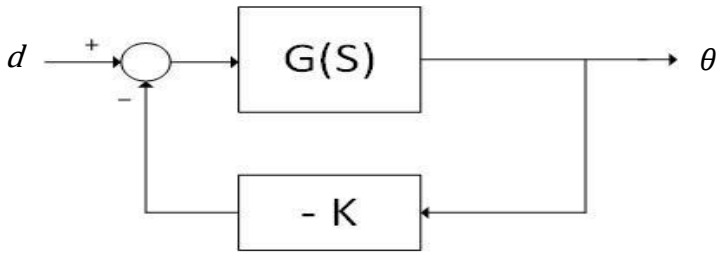

Figure 3.4: Block diagram of the state feedback controlled system for taking curvatures

Figure 3.5 shows the simulation results for taking a curved path starting from the same initial position as in the reversing along straight line case while having bias of *d=0.5*.

According to the results, it can be seen that the system stabilizes while maintaining the pivot angles at a constant value which makes trailer system to move in a curved path. The relevant curved path of the vehicle can be observed in *Figure.3.6.* It can be seen that after sometime vehicle is starting to follow a curved path.

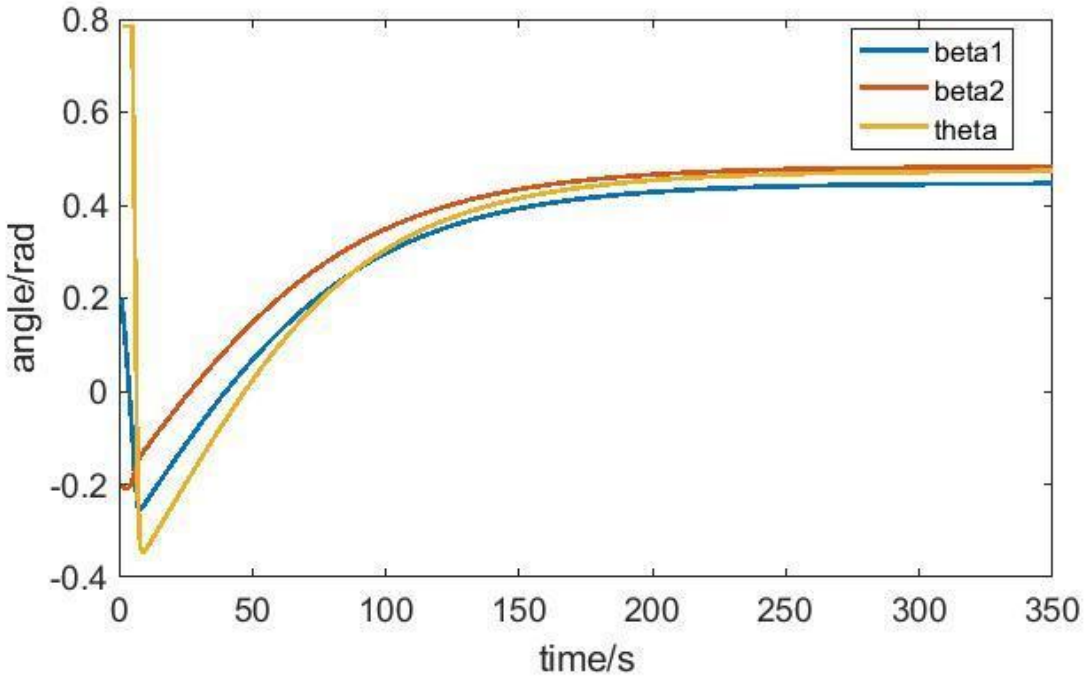

Figure 3.5: Variation of pivot angles with steering angle when taking a curvature

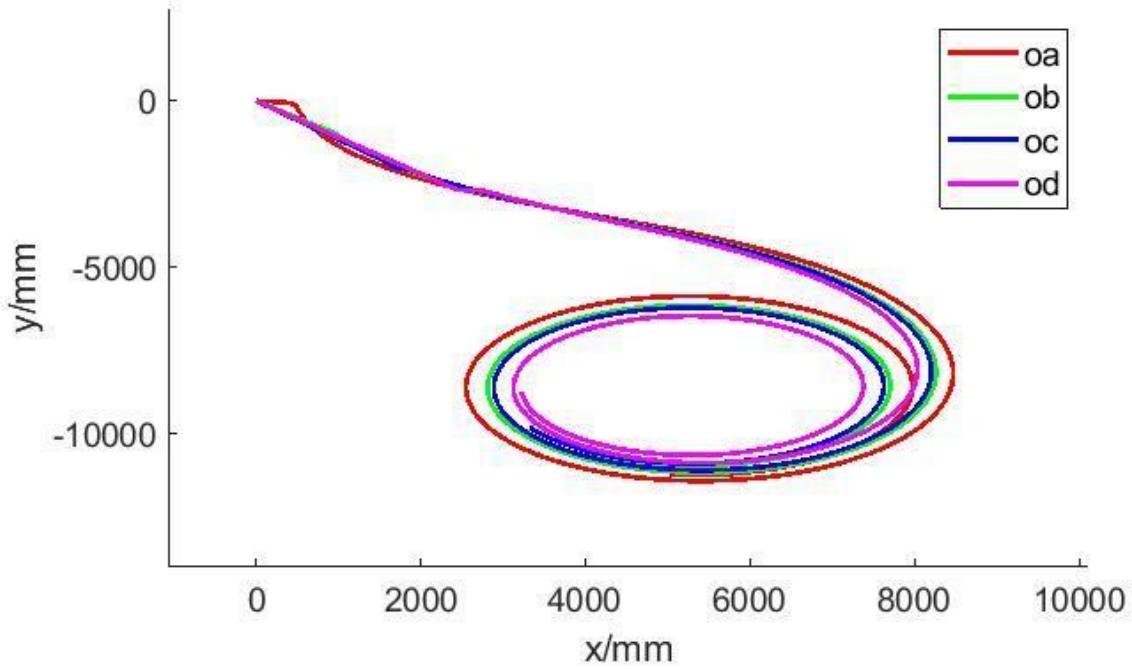

Figure 3.6: Vehicle travelling path when taking a curvature

3.3 Simulation results for the real vehicle

Simulations were done for the real vehicle as well to make sure the developed controller is working fine with the real vehicle.

The dimensions of the real vehicle in millimeters are,

L_0=5595, L_1=2265, L_2=2867, L_3=0, L_4=3796

Which gives

$$A = V_{a_1} \begin{bmatrix} 0.0349 & 0 \\ -0.0349 & 0.0263 \end{bmatrix}$$

$$B = V_{a_1} \begin{bmatrix} -0.0320 \\ 0.0141 \end{bmatrix}$$

By pole placement for the state feedback controller, suitable gains can be obtained as,

$$K = [-1.4 \quad 14]$$

Figure 3.7 shows the variation of pivot angles with steering angle when real vehicle is trying to stabilize from an initial position. The initial conditions were taken as in *Table 3.2* with the reversing speed of V_{a1}=100 mm/s.

Table 3.2: Initial conditions for real vehicle simulation

$x_{(0)} = 0$	$y_{(0)} = 0$	$\alpha_{(0)} = 1.2$	$\dot{\theta}_{(0)} = 0$	$\beta_{1(0)} = 0.3$
$\beta_{2(0)} = -0.3$	$\dot{\alpha}_{(0)} = 0$	$\dot{\beta}_{1(0)} = 0$	$\beta_{2(0)} = -0.2$	$\theta_{(0)} = -K \begin{bmatrix} \beta_{1(0)} \\ \beta_{2(0)} \end{bmatrix} = 0$

Relevant vehicle travelling path can be seen in *Figure 3.8*. It can be seen that after sometime vehicle is travelling in an almost straight line. Hence, it can be concluded that developed controller is suitable for the real truck and trailer system.

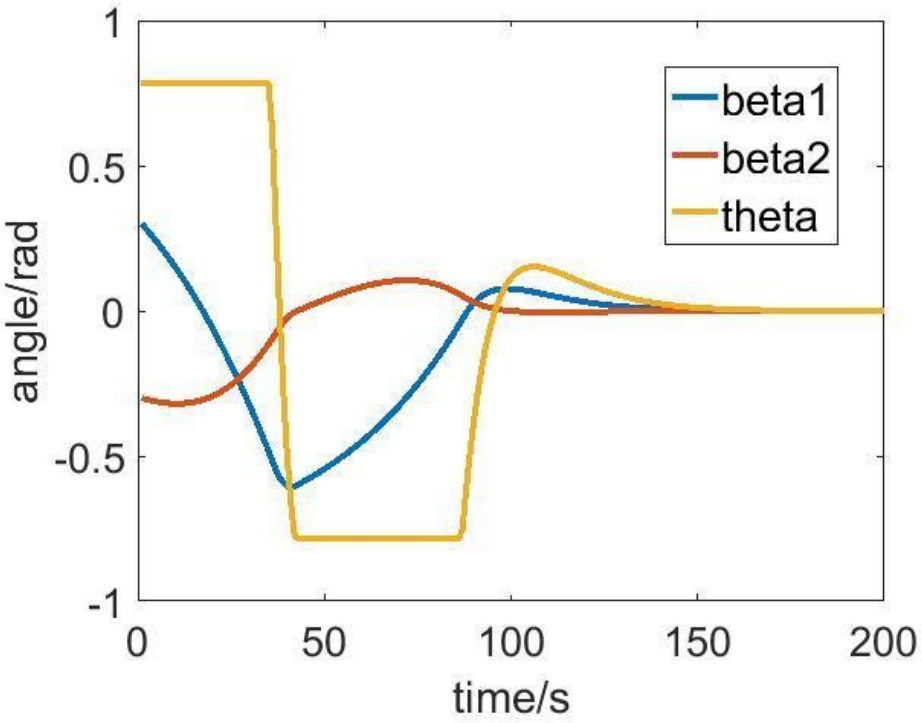

Figure 3.7: Variation of pivot angles with steering angle when stabilizing the real vehicle

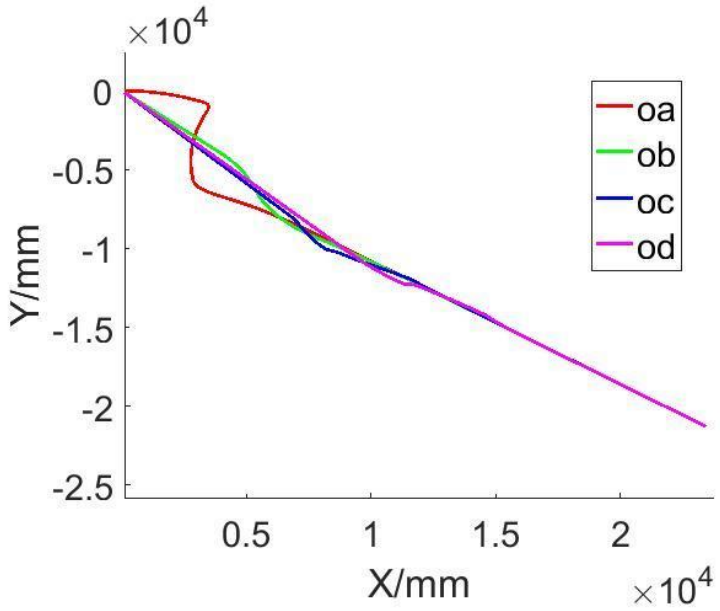

Figure 3.8: Vehicle travelling path when stabilizing the real vehicle

The simulations also done for the real vehicle for the case of reversing in curvatures. *Figure 3.9* shows how the pivot angles will change with the steering angle when taking a curvature. Relevant vehicle travelling path can be seen in *Figure 3.10*. When considering the real vehicle dimensions, the simulation results are realistic and practically achievable.

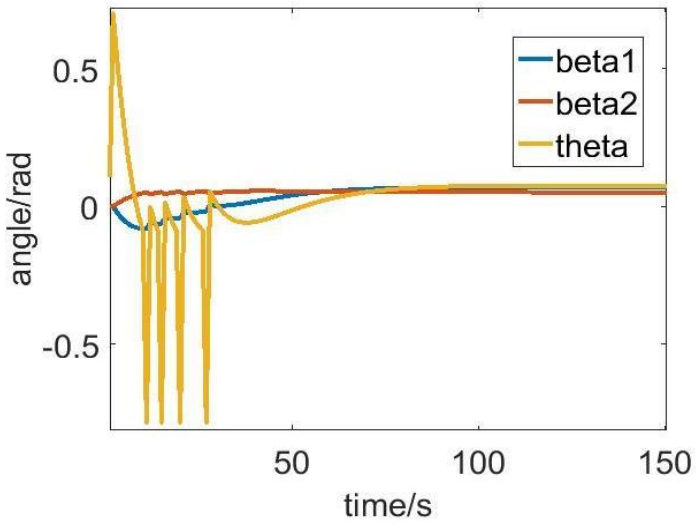

Figure 3.9: Variation of pivot angles with steering angle when taking a curvature in the real vehicle

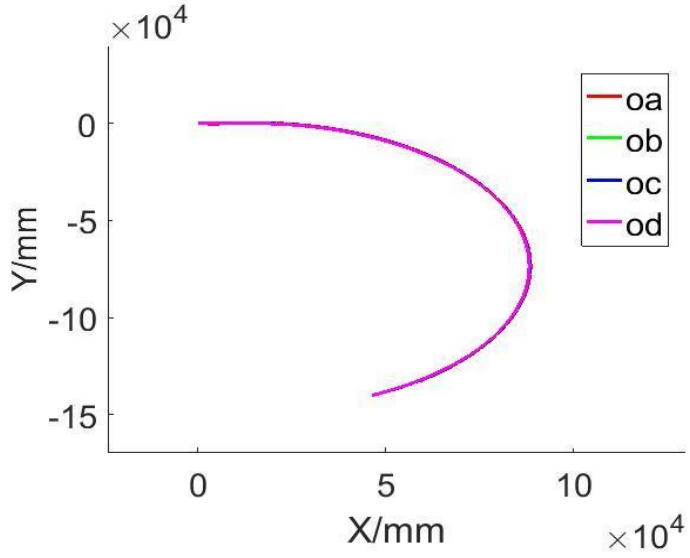

Figure: 3.10
Vehicle travelling path when taking curvature in the real vehicle

Furthermore, simulations were done for stabilizing the system considering the various initial poses. *Figure 3.11* shows variation of angles for various initial β_1 values while keeping initial β_2 value at zero. Similarly *Figure 3.12* shows variation of angles for various initial β_2 values while keeping initial β_1 value at zero.

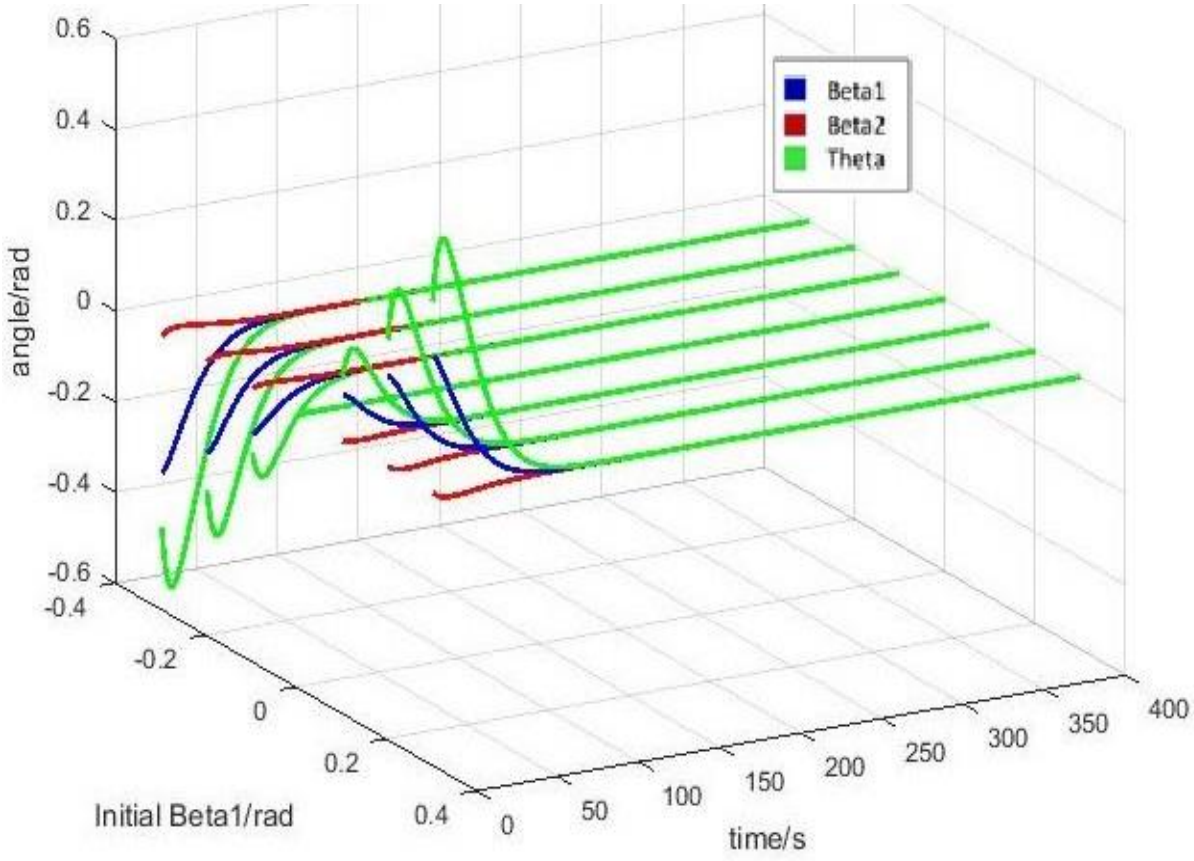

Figure: 3.11 Variation of angles for various initial $\beta_1$values while keeping initial $\beta_2$value at zero

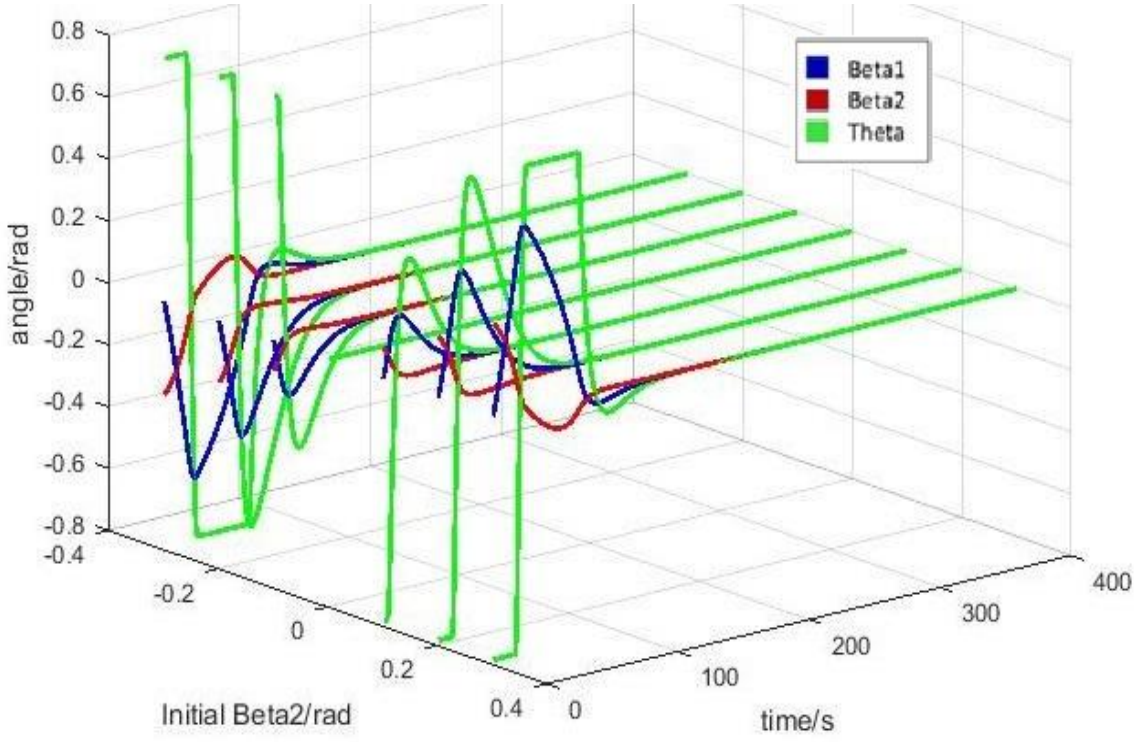

Figure: 3.12 Variation of angles for various initial $\beta_2$values while keeping initial β_1 value at zero

Those simulation results theoretically verify that the truck and trailer system could be reversed while stabilizing from various initial positions under the constraint of small angle approximation.

Chapter 4

Hardware and software implementation

4.1 Fully automated prototype

A hardware prototype was implemented with two passive trailers and a differential driven engine part that could be actuated using continuous servo motors. To measure the pivot angles potentiometers were used and the controller was implemented on an arduino board. The prototype can be seen in *Figure 4.1*. The ultimate objective of making this fully automated prototype is to verify the control algorithms.

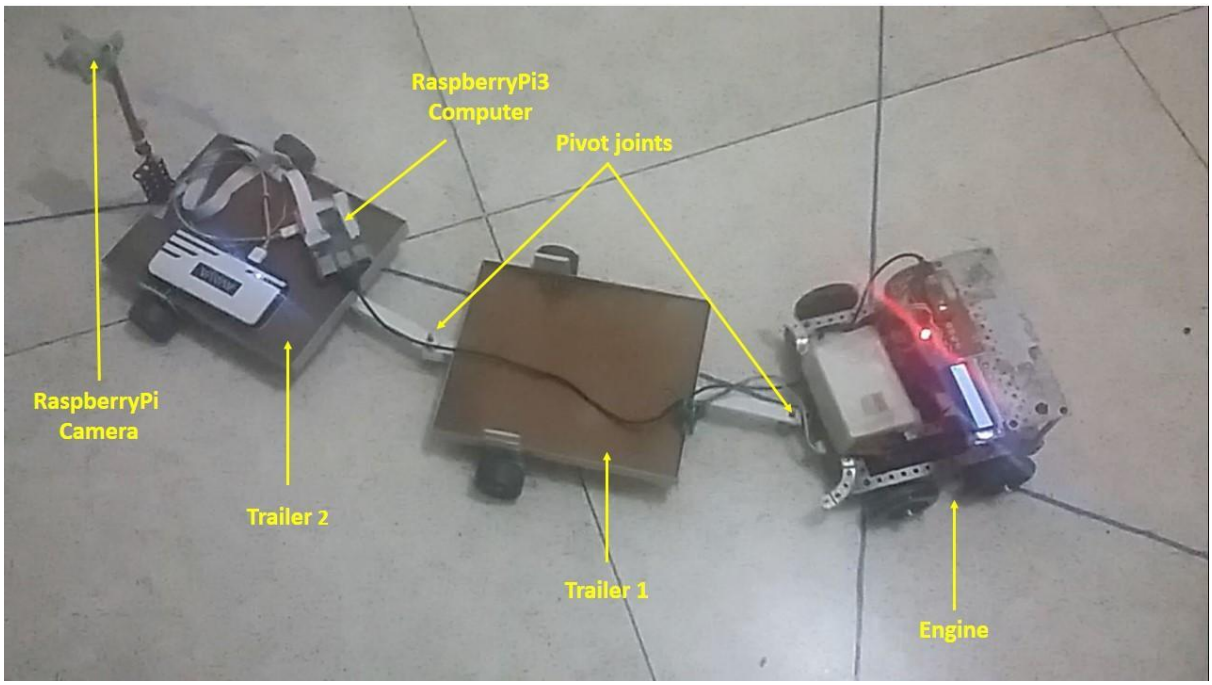

Figure 4.1: Fully automated prototype

The first task was to stabilize the system. To achieve this, relevant gains were obtained as described in the mathematical model. Then the system was able to reverse on a straight line while maintaining the pivot angles at zero.

Next task was to reverse in a curved path. For this a small bias was introduced as described in Chapter 3. With the bias, the system was able to successfully reverse in a curved path while stabilizing the pivot angles.

After successfully stabilizing the system and reversing in curved paths, reversing the system along a given path was tested. For that visual sensing was used with a camera. The desired path was marked using a coloured strip and the system should reverse along that path. To identify the path using the camera, another path detection algorithm was implemented using OpenCV.

Initially this path detection algorithm was created to detect the lane markers on a typical road. A block diagram representation of the lane marker detection algorithm can be seen in *Figure 4.2.*

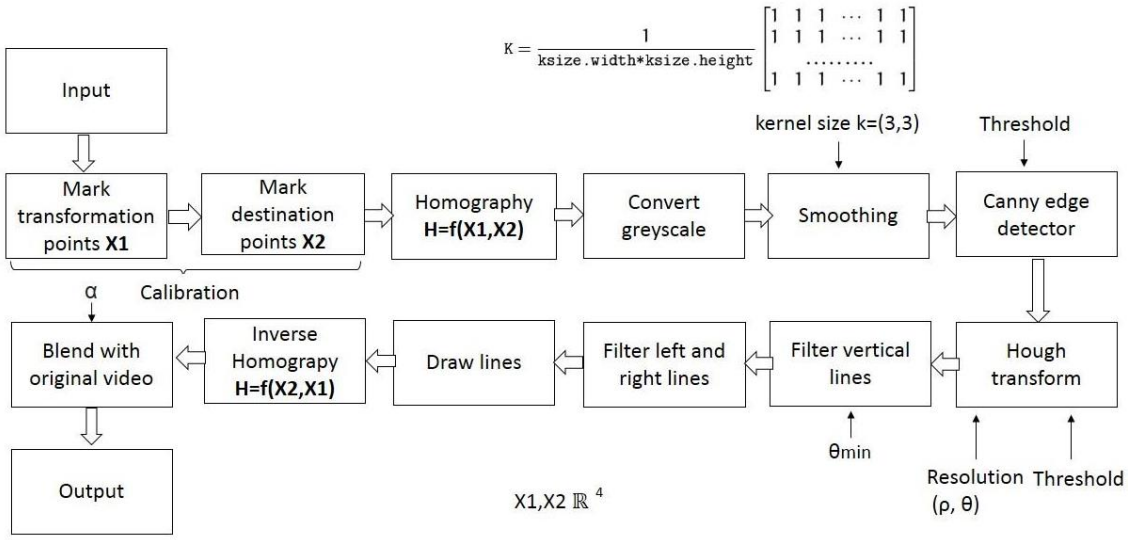

Figure 4.2: Lane marker detection algorithm

It is assumed that camera feed is taken from a camera which is mounted at the back of the trailer of the vehicle. *Figure 4.3* shows outcome of the lane marker detection algorithm.

Figure 4.3: Detected lane markers

The lane marker detection algorithm was modified to identify a path on a 2d plane by eliminating the homography transformations. The block diagram representation of the modified algorithm can be seen in *Figure 4.4.*

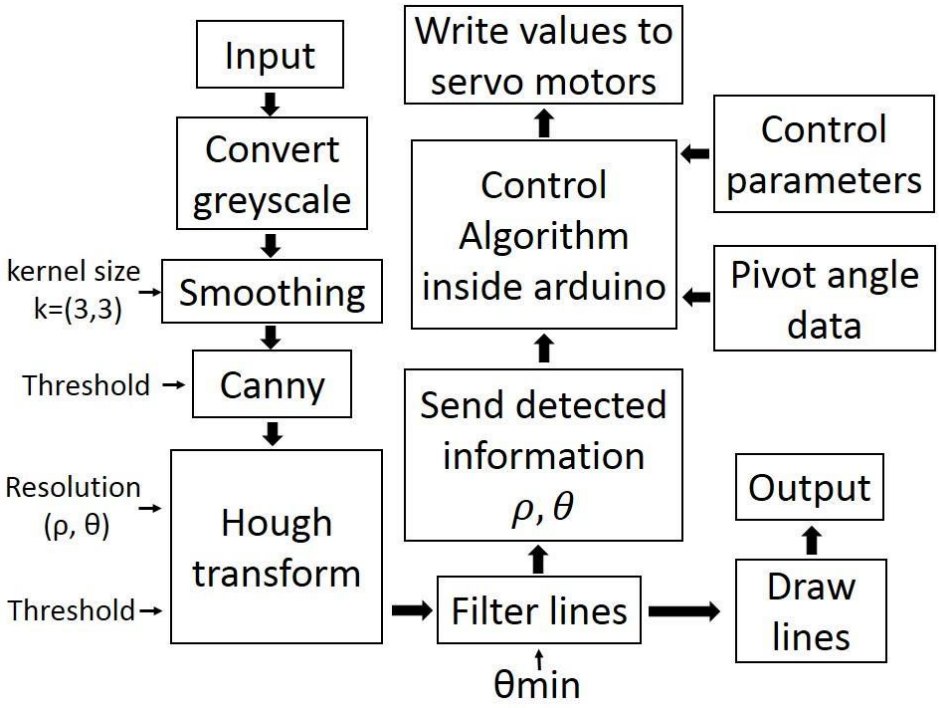

Figure 4.4: Path detection algorithm.

The image processing algorithm was implemented inside a Raspberry Pi computer. Extracted image information was passed into the controller implemented inside an arduino board via a serial connection. Block diagram of the internal hardware architecture is represented as in *Figure 4.5.*

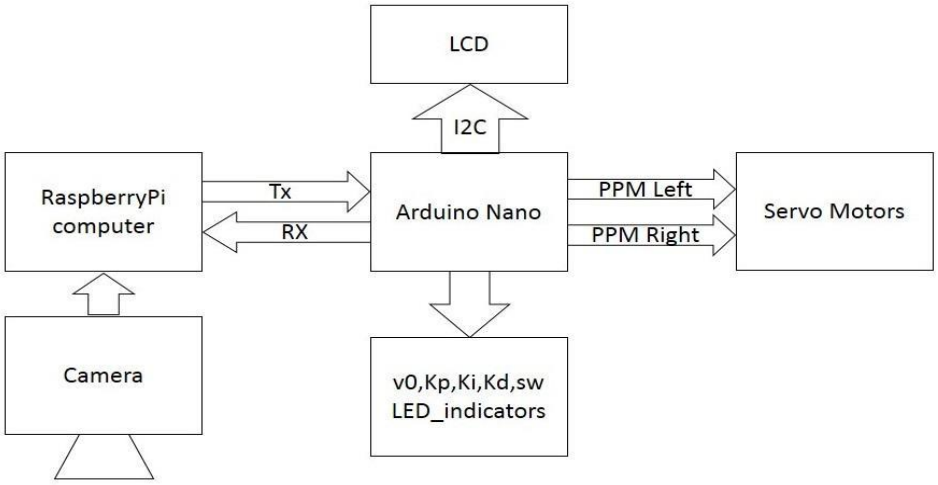

Figure 4.5: Internal hardware architecture

A video recording of the system performance is available at

https://youtu.be/Px5YawUA4k8.

4.2 Industrial scale prototype

With successful results with the fully automated prototype, an industrial scale prototype also developed as shown in *Figure 4.6.*

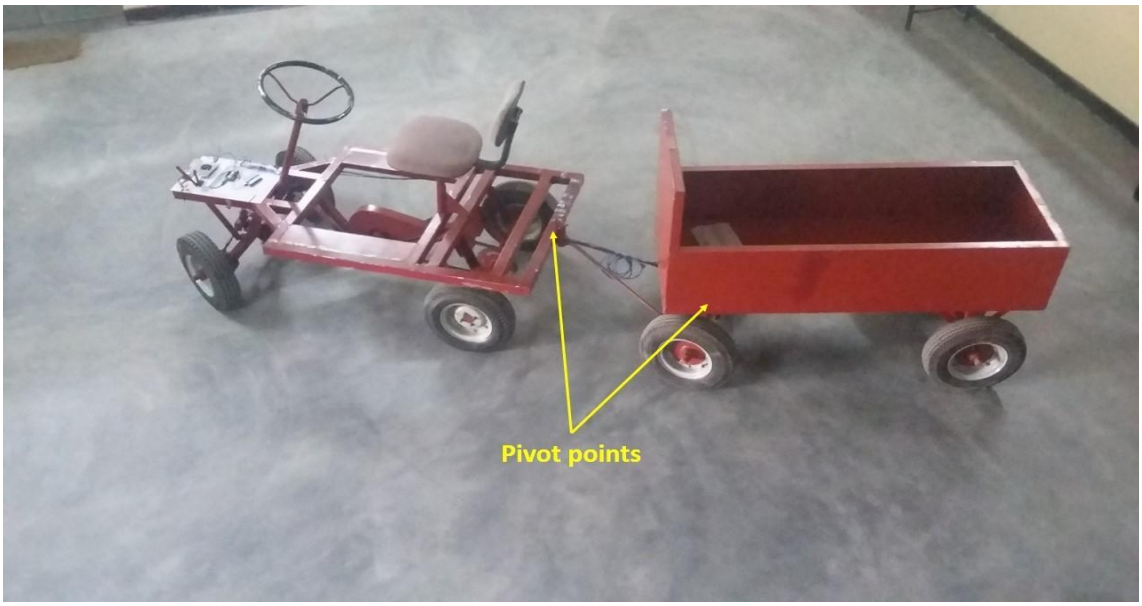

Figure 4.6: Industrial scale prototype

This prototype has one trailer and two pivot joints. Hence it is equivalent to the system described in the mathematical model in chapter 2.

This prototype is a manual system where a human driver has to do the steering. Also, there is a peddling mechanism to maneuver the vehicle. There are no actuators available in this vehicle at the present moment.

According to the pivot angles, desired steering angle is calculated inside the controller. Pivot angles were measured using potentiometers installed at the pivot joints. Both desired and actual steering angles are displayed to the driver using simple user interface as shown in *Figure 4.7.*

Figure 4.7: Display showing the desired and actual steering positions

The driver has to track the desired steering angle with the actual steering angle manually while reversing. This is possible by trying to move the square shape relevant to the actual steering position on the screen to coincide with the square shape relevant to the desired steering position. The square shape position relevant to the actual steering will change when driver is rotating the steering wheel.

When it is required to take curved paths driver can introduce the *d* bias term to the controller by rotating a knob. However the prototype was heavier than expected, initial testing was done manually, pushing the vehicle while steering. Video recording of the prototype is available at https://youtu.be/n7IWPR3_g-E.

4.3 Improving the system to measure angles with computer vision.

When it comes to a real vehicle, mounting sensors to measure pivot angles become an expensive and complicated task. Because those sensors should withstand all the vibrations and impacts when the vehicle is in move. As a solution to that fiducial marker based angle measurement method is introduced. For that ArUco library[10][11] is adopted to estimate the fiducial marker pose.

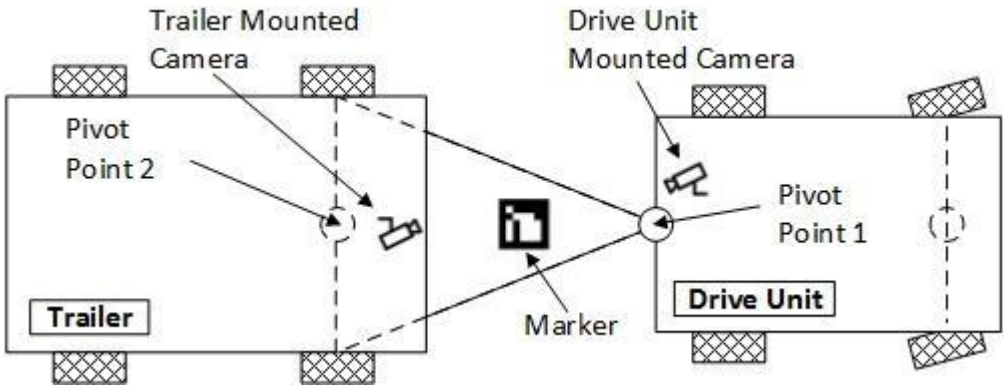

Figure 4.8: Drive unit and trailer arrangement with cameras

Two cameras were mounted on the drive unit and the trailer, focused at a fiducial marker mounted on the connection link as in *Figure 4.8.* This method allows to estimate the two pivot angles β_1 and β_2 using cameras as sensors without using any complicated mechanical mounts.

The marker pose is given with respect to the calibrated camera in axis angle representation.

$$\vartheta \boldsymbol{V} = (\omega_1, \omega_2, \omega_3)$$

Where

$$\boldsymbol{V} = (V_1, V_2, V_3)$$
$$\left|\left|\boldsymbol{V}\right|\right| = 1$$

and ϑ is the rotation angle around vector $\boldsymbol{V}$. This can be converted into a rotation matrix R using Rodrigues formula.

$$R = \begin{bmatrix} r_{11} & r_{12} & r_{13} \\ r_{21} & r_{22} & r_{23} \\ r_{31} & r_{32} & r_{33} \end{bmatrix}$$

Rotation matrix R can be decomposed as rotations around axis 1,2 and 3 [12].

$$\mu_1 = \tan^{-1}(r_{32}/r_{33})$$
$$\mu_2 = \tan^{-1}(-r_{31}/\sqrt{{r_{32}}^2 + {r_{33}}^2})$$
$$\mu_3 = \tan^{-1}(r_{21}/r_{11})\}$$

Then the pivot angle measurement can be obtained using μ_3 after an angle calibration.

4.4 Improvements to the steering angle measurement method

In the small scale prototype and the initial stage of the industrial scale prototype testing, carbon type linear potentiometers were used with the internal 10 bit ADC in the arduino. It is assumed that voltage will change proportional to the rotated angle. This combination gave a considerably good angle measurement than expected.

However carbon potentiometers could easily tend to wear when used for a longer periods and accuracy of the measured angles will become considerably less. Hence, angle measurement was improved using wire a wound potentiometer. Since, both pivot angles could be measured using the vision based method, only one potentiometer was used to measure the steering angle.

In modern vehicles with electronic steering, the steering angle is available through the vehicle CAN bus. In such scenarios, usage of separate potentiometer can be completely eliminated. However the developing driver assisted system is proposed fix into the older vehicles which don't have built-in steering angle sensors. Hence for such systems, only available option is to use an external potentiometer to measure the steering angle.

The used wire wound potentiometer parameters can be identified as,

- 10 K ohm resistance
- Power consumption max 2W
- Max working voltage 160V
- Mechanical travel 3600^0+10^0

The steering wheel not rotate less than 10 turns. Therefore an amplification circuit was added to increase the resolution of the measurement while reducing the number of turns of the potentiometer. The schematic diagram of the complete signal conditioning circuit is shown in *Figure 4.9.*

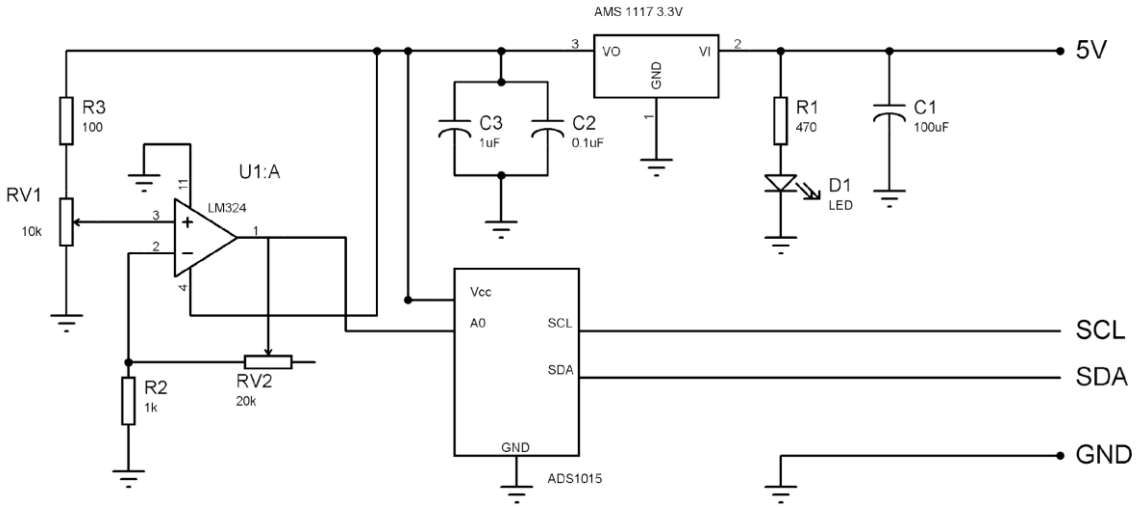

Figure 4.9: Schematic diagram of the signal conditioning circuit

The circuit is mounted near the potentiometer and it is connected to the main unit through the I2C interface. Converting potentiometer value to digital format and transmitting them in digital format will avoid any erroneous readings due to voltage drops.

The resolution of the angle measurement unit can be adjusted using the RV2 preset. The angle value can be calculated as in following equation.

$$Theta = V\_measured\ x\ (10100\ Ohm/3.3V)\ x\ (3600\ deg\ /10000\ Ohm)\ x\ (1/gain)$$

Where gain is given by,

$$gain\ =\ (1 + RV2/R2)$$

Variation of measured voltage with potentiometer angle for measuring up to 3600 degrees and 360 degrees can be seen in Figure 4.10. Similarly depending on the maximum angle, gain can be adjusted to get the maximum resolution. The completed hardware unit can be seen in *Figure 4.11*.

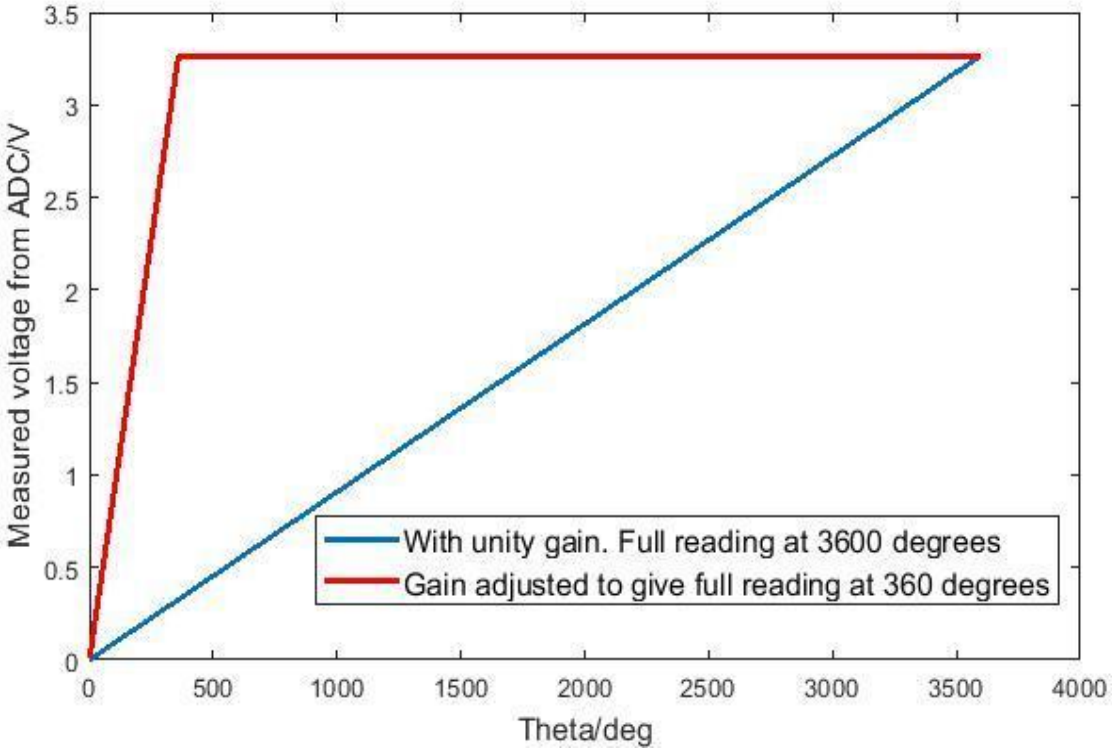

Figure 4.10: Variation of measured voltage with rotated angle

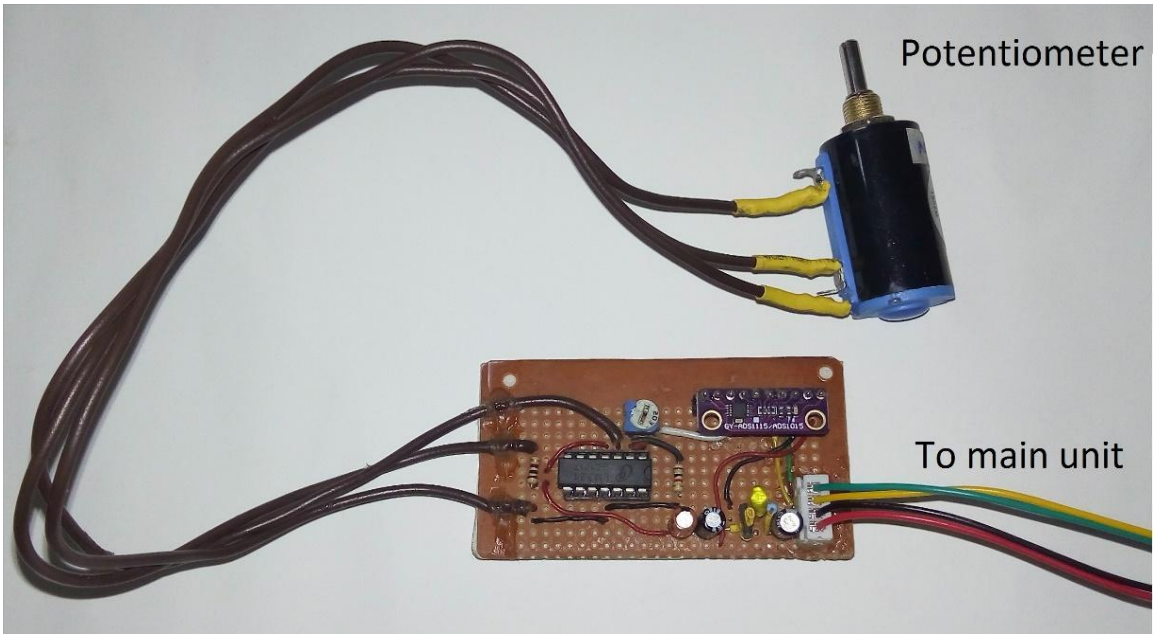

Figure 4.11: Angle measurement unit

4.5 Improvements to the user interface

Image processing algorithms and control algorithms were implemented inside a raspberry pi computer making a single compact device. The steering guidelines including the desired and actual steering angles were displayed to the driver using a touch sensitive screen where driver can change the vehicle curvature as required. Also the vehicle pose is displayed to the driver in real time. System user interface is described in *Figure 4.12.*

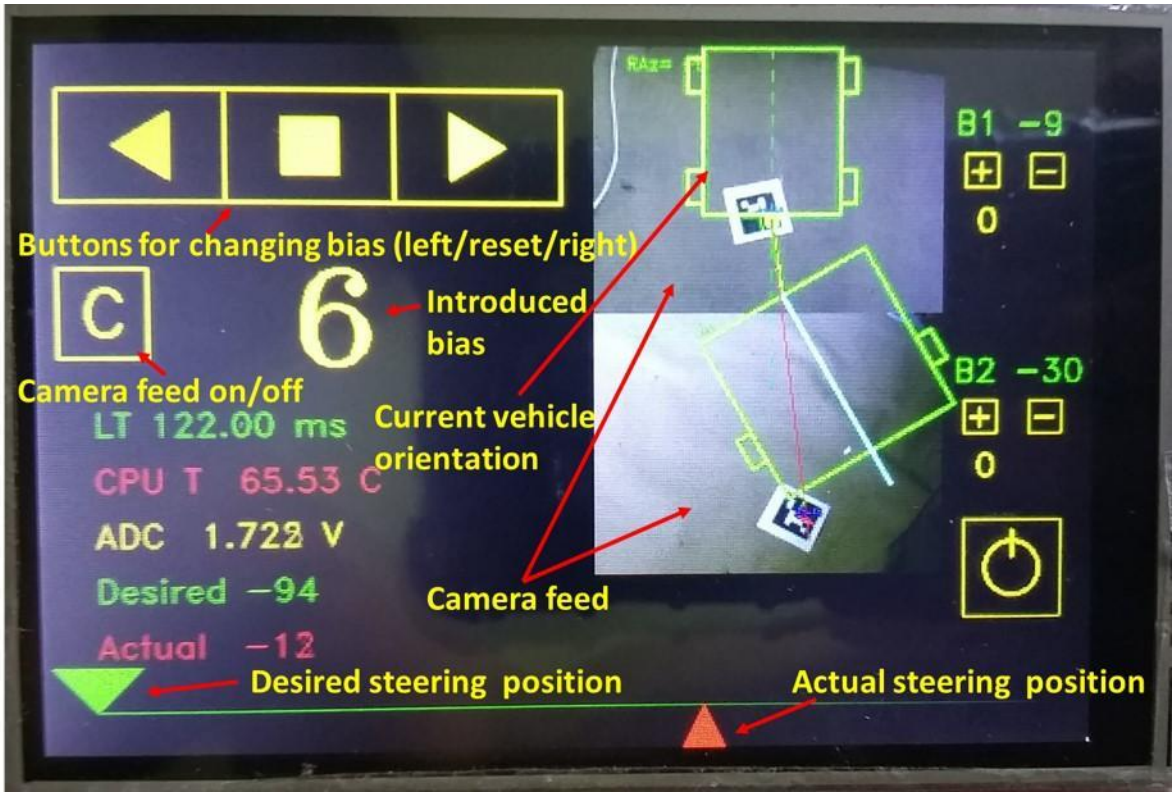

Figure 4.12: Improved user interface

4.6 Real time path prediction

As an additional feature, real-time path prediction is incorporated with the current system. It is capable of predicting the truck and trailer system movement path using the pivot angle and introduced bias data. The 2-D version of the path prediction feature can be observed as in *Figure 4.13.* By observing the predicted path driver can easily introduce the correct amount of bias to the system to reverse the vehicle to a desired location.

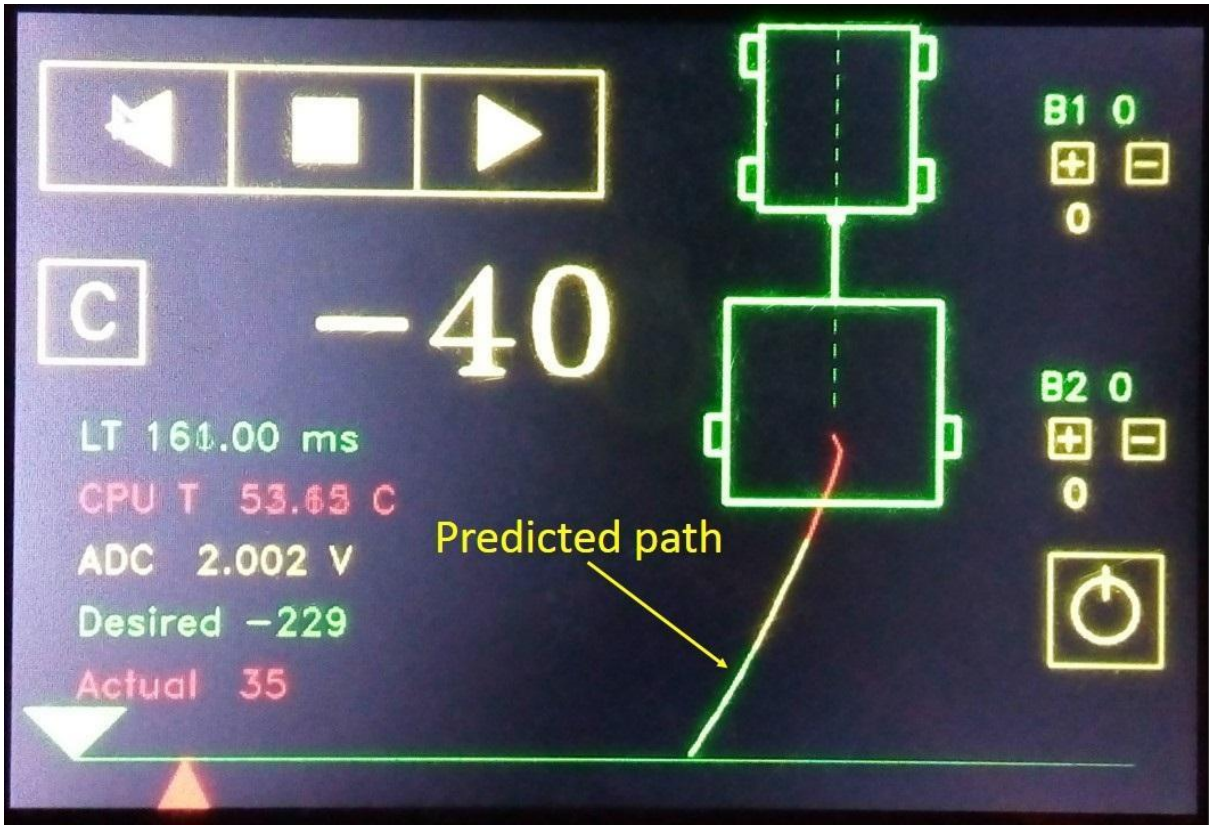

Figure 4.13: Real-time path prediction with UI

Also this path prediction was extended for the 3D case. In this scenario a camera feed from a camera mounted at the end of the trailer will be projected on the screen. On top of the camera feed, predicted path will be drawn as a virtual path.

Currently drawing of the path was done as shown in *Figure 4.14*. Exactly placing this virtual lines on road could be done finding homography between projection planes. This calibration is planned to perform after mounting cameras in a real scale vehicle.

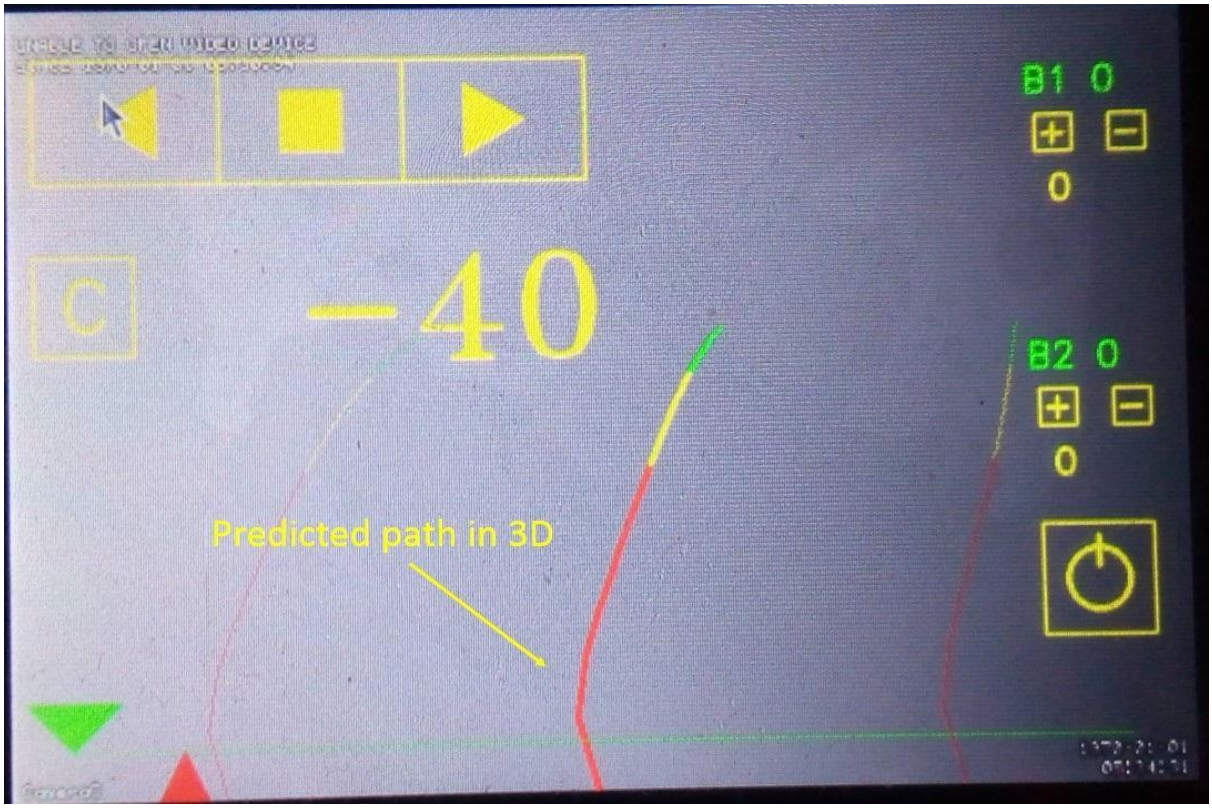

Figure 4.14: Real-time path prediction 3D view

4.7 Final prototype

Final prototype was obtained by compacting all the embedded electronics into a single main unit. This can be seen in *Figure 4.15 - B.* Even though USB cameras used initially to test the system later the system was modified to be compatible with the IP cameras.

Figure 4.15 – A shows the currently used camera system for testing purposes. These are USB web cameras but converted to work as IP cameras with the help of a RaspberryPi computer.

However this camera system will be replaced with industrial IP cameras when fixing this system to a real scale vehicle.

To measure the steering angle, potentiometer in the steering angle measurement unit needs to be mechanically coupled with the vehicle steering wheel. This unit is shown in *Figure 4.15 - C.*

Also, as the marker an illuminated marker was made with LED backlights to allowing the system to work at night and low light conditions. The illuminated marker can be seen in *Figure 4.15 – D.*

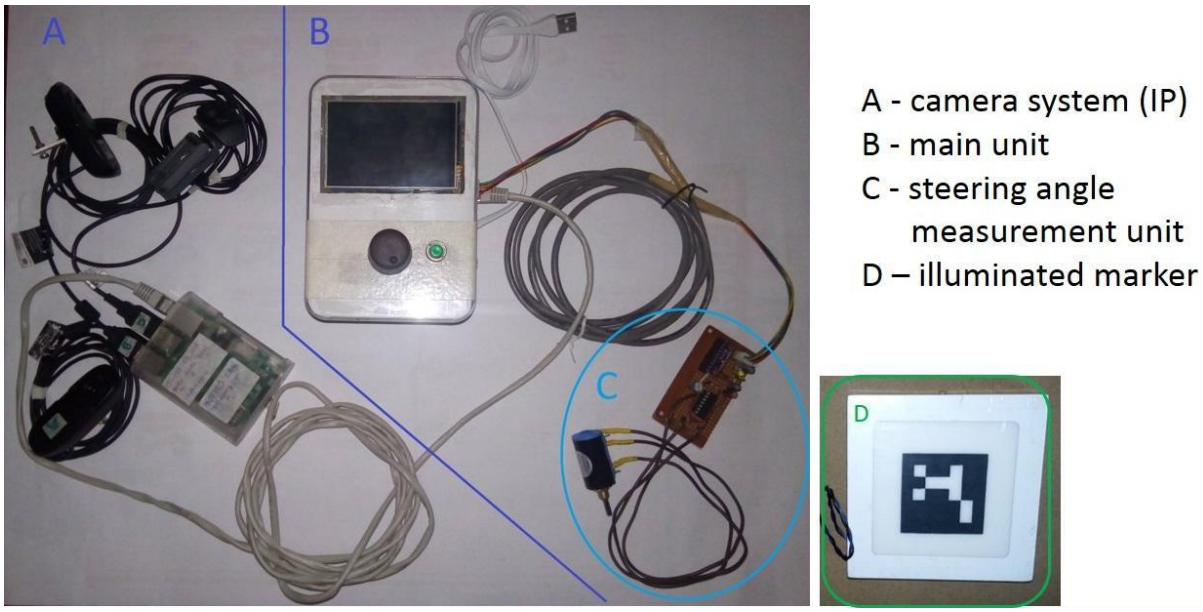

Figure 4.15: Completed prototype of the device

Chapter 5

Other outcomes

5.1 Publications

R.N. Ranaweera, D.H.S. Maithripala, A.D. Polpitiya, S.K. Senevirathne, Driver assisted steering system for reversing an articulated vehicle, IESL 112th Annual Sessions, pp.551-558, 2018.

5.2 Pending patents

A patent is pending for the developed driver assist system under the title “Trailer reverse assist system with a human interface”

Chapter 6

Conclusions and future work

6.1 Conclusions

This work presents a controller for reversing a truck and trailer mechanism. The controller is verified using simulations as well as using the hardware prototypes. These prototypes could reverse along a straight line or a curve. The small scale fully autonomous prototype can reverse along a given path while visually tracking the path. The developed manual system for industrial scale prototype is capable of helping the truck and trailer system to reverse straight as well as in curvatures while stabilizing the vehicle.

Also to provide more effortless reversing, real time prediction of the vehicle movement is incorporated to the system. This predicted path will be merged with another camera image, where the camera is mounted at back of the trailer which will make this into a complete driver assisted steering system.

The developed driver assist system is currently available as a single unit that is waiting to be fixed into a real vehicle for testing.

6.2 Future work

There are some future upgrades that can be done to the existing system,

- Equip the industrial scale prototype with industrial IP cameras to make it a more robust system.
- Automating the industrial scale prototype to operate along given paths which can be used in warehouse operations.
- Update the system UI with aesthetic graphics using Qt framework to enhance the user experience.
- Manufacturing the system as a plug and play device that can be fixed into any truck and trailer system.

Bibliography

[1] C. Altafini, A. Speranzon, and K. H. Johansson, Hybrid control of a truck and trailer vehicle, in Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics), 2002.

[2] R. M. Murray, Trajectory Generation for the N-Trailer Problem Using Goursat Normal Form, IEEE Trans. Automat. Contr., vol. 40, no. 5, pp. 802819, 1995.

[3] Z. Dieter, D. Polock, and P. Wojke, Steering Assistance for Backing Up Articulated Vehicles, Syst. Cybern. Informatics, 2003.

[4] M. M. Michalek and M. Kielczewski, The Concept of Passive Control Assistance for Docking Maneuvers with N-Trailer Vehicles, IEEE/ASME Trans. Mechatronics, 2015.

[5] P. Ritzen, E. Roebroek, N. Van De Wouw, and Z. Jiang, Trailer Steering Control of a Tractor Trailer Robot, IEEE Trans. Control Syst. Technol., 2016.

[6] N. Evestedt, O. Ljungqvist, and D. Axehill, Path tracking and stabilization for a reversing general 2-trailer configuration using a cascaded control approach, in IEEE Intelligent Vehicles Symposium, Proceedings, 2016.

[7] M. Michalek and M. Kielczewski, Helping a driver in backward docking with N-trailer vehicles by the passive control-assistance system, in IEEE Conference on Intelligent Transportation Systems, Proceedings, ITSC, 2013.

[8] J. Roh and W. Chung, Reversing control of a car with a trailer using a driver assistance system, in Proceedings of IEEE Workshop on Advanced Robotics and its Social Impacts, ARSO, 2010.

[9] P. Svestka and J. Vleugels, Exact motion planning for tractor-trailer robots, in Proceedings - IEEE International Conference on Robotics and Automation, 1995.

[10] Garrido-Jurado, Muñoz-Salinas, Madrid-Cuevas, & Marín-Jiménez, Automatic generation and detection of highly reliable fiducial markers under occlusion, 2014

[11] Garrido-Jurado, Muñoz-Salinas, Madrid-Cuevas, & Medina-Carnicer, Generation of fiducial marker dictionaries using Mixed Integer Linear Programming, 2016.

[12] Computing Euler angles from a rotation matrix, Slabaugh, 1999.

[13] "Speeded up detection of squared fiducial markers", Francisco J.Romero-Ramirez, Rafael Muñoz-Salinas, Rafael Medina-Carnicer, Image and Vision Computing, vol 76, pages 38-47, year 2018.

[14] Agarwal, Anubhav & V Jawahar, C & J Narayanan, P. (2008). A Survey of Planar Homography Estimation Techniques.