Case Study Report:
Development of a human-eye rotation inspired two-axis camera mounting system

Course code and title:	B51RO – Robotic Mechanical Systems
Group:	Team 6
Student Names and Heriot-Watt Numbers:	Baudouin BELPAIRE – H00445613 João COELHO – H00342606 Hari GOVIND – H00449213 Pranav SAJIKUMAR – H00436046 Sam WATSON – H00344148
Date:	17/04/2024

Table of Contents

Table of Figures

Table of Equations

1. Introduction (Hari and Baudouin)

The development of humanoid robots that mimic human behaviour holds a great promise across diverse applications. By reproducing natural gaze behaviours, such as maintaining eye contact, following objects of interest, and expressing emotions through eye movements, these robots can enhance human-robot interactions in various fields. In social robotics, humanoid robots with a realistic gaze behaviour can foster deeper connections with users. As research progresses in this area, the integration of sophisticated gaze behaviour into humanoid robots has the potential to revolutionise the human interaction with robots and benefit from robotic technology in our daily lives.

The human eye tracking system functions as an intriguing feedback control loop, managing a seamless interaction between sensory input, neural processing, and motor commands as represented in Figure 1. Throughout this process, continuous feedback mechanisms ensure accurate eye positioning and stable fixation, maintaining clear and focused vision across diverse environmental conditions. Essentially, the human eye tracking system exemplifies the cooperation between sensory perception and motor control, allowing to navigate through the surroundings with remarkable precision and adaptability.

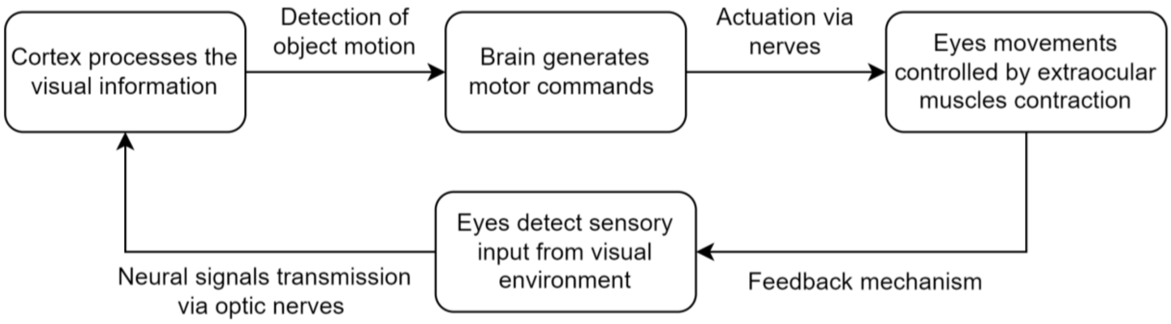


Figure 1: Human eye feedback control loop

Thus, the human eye like mechanism developed was defined based on the previous characteristics. The system must comprise 2 eyes rotating along the horizontal and vertical axes which are controlled by 2 servo motors. In addition, the system includes a camera to detect visual information from the world. Contrary to the human vision system, the visual perception will be fixed such that the detected objects will be with respect to the fixed frame. The feedback loop control system is then shown in Figure 2.

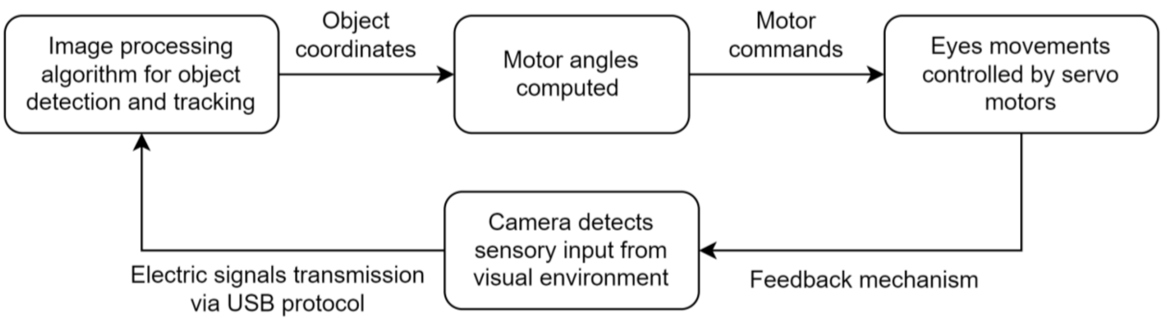


Figure 2: Human eye inspired mechanism feedback control loop

2. Robotic System Design

2.1. Mechanical Design

2.1.a. Design concepts (Sam)

The initial design for the eyes was based on the tracking camera such as the Minrray UV100 Tracking camera, as this would allow for the two degrees of freedom that are required for the eyes to be able to rotate around the horizontally and vertical axis.

Figure 3: Tracking camera and initial basic design for eye

It was then decided for the design to have the eye mounted in a cradle-like link to stop any deviation that would occur due to the eye only being fixed at one for rotation around the vertical axis and as per the design constraints, it was investigated different servo locations that would allow for only two to be used.

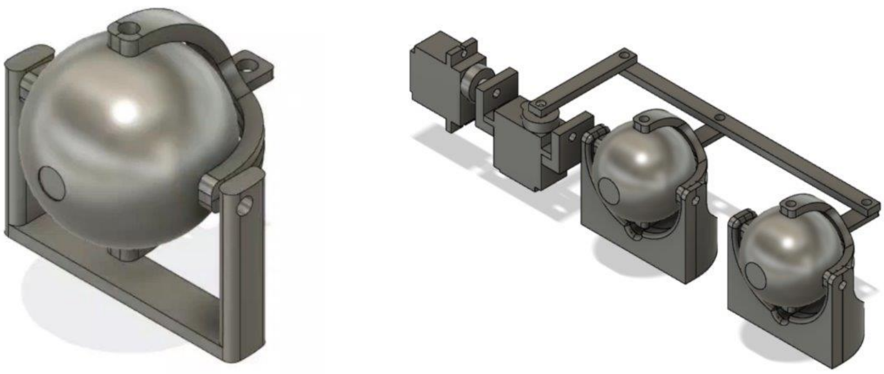


Figure 4: Eye with cradle link and the initial concept for the location of two servos

The mechanical workspace that was utilised for the eyes allowed for a total of 120 degrees of rotation around the vertical axis (60 degrees left and 60 degrees right), and there was a total of 90 degrees rotation around the horizontal axis (45 degrees up and 45 degrees down) this allowed the eyes to follow any object in the field of view of the camera used while ensuring that no links collided with any other part of the structure.

An initial prototype was created to test the workspace and allow for easy visualisation on what was required to be changed leading to the final design which will be discussed in section 3.1

2.1.b. System conception (Baudouin)

As previously mentioned, the eye rotating system has been designed in a cradle like system to enable the rotation in both directions as shown in the kinematic diagram in Figure 5.

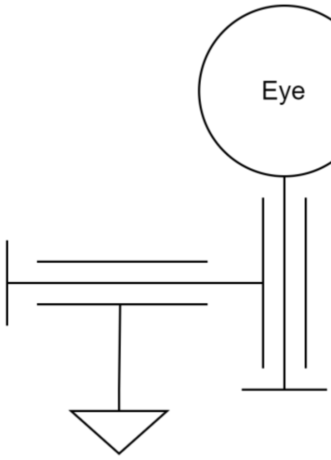


Figure 5: Eye system kinematic diagram

To enable these rotations, 2 succinct rotations must be performed in series with the use of an intermediary part. Thus, the intermediary part enables a vertical rotation between the eye base and itself, but also a horizontal rotation between itself and the eye. This intermediary part is called the eye link and can be referred in the orthographics appendix. Furthermore, the cross-section analyses are shown in Figure 6 to demonstrate the moving joints obtained.

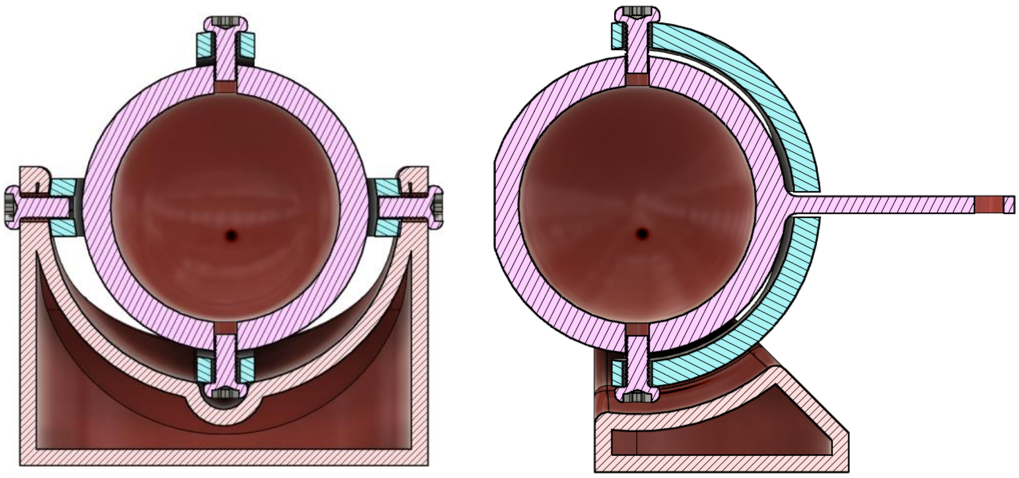


Figure 6: Eye assembly cross-section views

The revolute joints are obtained by tightening 2 parts together using a loose and tighten fit. The loose fit hole is placed in the squeezed part and is used for the screw to freely move in acting as a rotating axis. While the loose fit hole is at the opposite of the screw head and is used to create a connection with the screw. An example is provided to further explain the connection of the revolute joints in Figure 7.

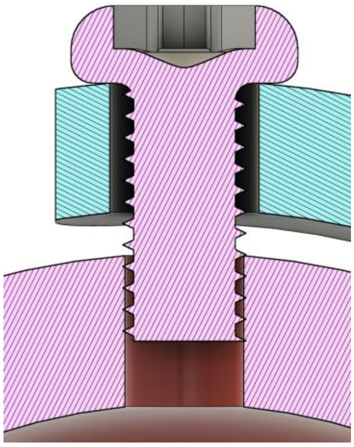


Figure 7: Zoom on the revolute joint

The servo motor controlling the eyes horizontal angle must be placed on a rotating axis to accordingly follow the eyes orientation according to the horizontal plane. Thus, it has been placed on a cradle like system rotating along the same axis as the 2 eyes. However, as the vertical motor generates the vertical angular motion, this one is grounded with its rotating axle aligned to the eyes axis to ensure a proper functioning.

Furthermore, to ensure the transmission of the motors motion, a crank has been used to move both eyes using a single motor for the horizontal rotation. Similarly, a linking part is mounted between the eye support parts and the vertical motors to ensure the vertical articulation of both eyes again using a single motor. This part could have been avoided as the crank is already linking the two eyes, but it was implemented to reduce the moment acting on it as it is placed away from the rotating axis.

2.1.c. Mechanical parameters calculations (Baudouin)

Based on the previous assumptions defining the mechanical workspace, and the constrained system workspace, two important distances must be computed to avoid collisions while the eyes are controlled. The affected parameters are the lengths of the eye links and the eyes height position to avoid collision between the connecting crank and the eye support as well as the links with the base ground. These two parameters are illustrated in the Figure 8.

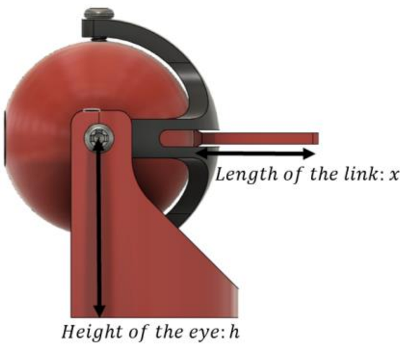


Figure 8: Illustration of the eye link and height parameters

To define the eye link length, it is crucial to determine first the necessary length between the eye centre and the minimal position the link can achieve. This length is defined by the eye radius, the collision free space between the eye and the support, the eye support thickness, the connecting crank width, and a safety distance. The result is obtained in Equation 1.

$$L_{minimal} = r_{eye} + d_{eye,support} + t_{support} + w_{crank} + d_{safety} = 17.5 + 1 + 3 + 6 + 1 = 28.5mm$$

Equation 1: Minimal distance achievable by the eye link at full extension

From there, the minimal length for the eye link can be computed for a theta tilt angle. It is important to mention that the eye link comprises the eye radius and the eye link length itself as shown in Equation 2.

$$l_{eye\ link,norm} = r_{eye} + l_{eye\ link} = 17.5 + x\ mm$$

$$l_{eye\ link,x-component} = (17.5\ +\ x) * \cos\theta_1 =\ 17.5 * \cos\theta_1\ +\ x * \cos\theta_1\ mm$$

Equation 2: Eye link length based on the horizontal angle

Once the maximal theta angle is achieved, the eye link must be at least equal to the previous length (L) computed. Thus, the eye link calculation is expressed by the Equation 3.

$$l_{eye\ link,x-component} = L_{minimal}$$

$$\rightarrow 17.5 * \cos\theta_1 + x * \cos\theta_1 = 28.5\ mm$$

$$\rightarrow x = \frac{28.5 - 17.5 * \cos\theta_1}{\cos\theta_1} = \frac{28.5}{\cos\theta_1} - 17.5\ mm$$

Equation 3: Eye link length formula

Similarly, the height position is affected by the vertical rotation angle and the eye link previously determined. Thus, the formula defining the ideal height position is demonstrated in Equation 4.

$$h = (17.5 + x) * \sin\theta_2 = 28.5 * \frac{\sin\theta_2}{\cos\theta_1}\ mm$$

Equation 4: Eye height position formula

Note: The height position is dependent on the maximum horizontal and vertical angles whereas the eye link parameter only depends on the maximum horizontal angle.

Overall, the final lengths used in the design have been gathered in the Table 1 based on the two maximal angles for both rotations.

Table 1: Results of the eye link length and height based on the maximal rotation angles

$\boldsymbol{\theta_1}$ (°)	$\boldsymbol{\theta_2}$ (°)	$\boldsymbol{x}$ ($\boldsymbol{mm}$)	$\boldsymbol{h}$ ($\boldsymbol{mm}$)
50	45	26.8	31.4

2.2. Electronics System

2.2.a. Motor selection (João)

Choosing the appropriate motors for the robot is an important decision that profoundly influences the performance and capabilities of the system. In this section, we explore the rationale behind the selection of the SG90 micro servo for our project, highlighting its key features, advantages, and limitations.

The SG90 micro servo, shown in Figure 9, is a compact and versatile motor renowned for its reliability, precision, and ease of integration. With its reduced size and impressive performance characteristics, the SG90 stands as an optimal choice for applications demanding precise control and manoeuvrability.

Figure 9: Illustration of the SG90 micro servo motor

One of the standout features of most servos, including the SG90, is its built-in feedback control mechanism, which uses Pulse Width Modulation (PWM) for precise positioning. PWM is a modulation technique that uses a series of pulses to control the position and speed of the servo motor. This feedback control system enables the SG90 servo to accurately track and maintain its position, ensuring precise control over the robot’s movements. By utilising PWM for feedback control, the SG90 servo achieves remarkable accuracy and repeatability in its operations. The PWM signal, with its varying pulse widths, allows for fine adjustments to the servo's position, enabling it to reach and maintain the desired angle with high precision. This level of accuracy is essential for our task.

Furthermore, the integration of PWM feedback control enhances the overall performance and reliability of the robotic arm. The servo's ability to receive and interpret PWM signals facilitates communication between the controller and the motor, ensuring smooth and consistent operation in various applications. Due to its reduced size, the SG90 servo is remarkably easy to integrate into our robotic system. Its uniform dimensions and design, and straightforward interface simplify the integration process, allowing for an easy incorporation into our design. This ease of integration accelerates the development cycle and facilitates rapid prototyping and experimentation.

One limitation of the SG90 servo is its relatively low torque output of 1.6 kg-cm. While sufficient for our application, which utilises PLA links, heavier loads may necessitate a motor with higher torque capabilities. It is essential to consider the specific torque requirements of the application when selecting an optimal motor. Another issue with the SG90 servo is that it rotates at a rate of 60 degrees per 0.12 seconds, which may be considered slow compared to other high-speed motors. This also means that the acceleration is fixed and cannot be adjusted to allow smoother rotational movements. While the motor specifications are enough for our application, in case a rapid motion is necessary, implementing a motor with higher rotational speed and more control such as a stepper motor may be more beneficial.

Globally, the selection of the SG90 micro servo for our project is a testament to its exceptional performance, reliability, and versatility. With its feedback control, accuracy, ease of integration, and compact design, the SG90 servo proves to be an asset to achieve the objectives of our project. While it has certain limitations, such as low torque, moderate rotational speed, and no acceleration control, these drawbacks are outweighed by its many advantages in our specific application context.

By taking advantage of the capabilities of the SG90 servo, we are guaranteed to develop a highly functional and efficient robot capable of executing the supposed task accurately.

2.2.b. Image feed from the camera (Baudouin)

The camera ordered for the project is a Sony IMX179 with a high-resolution of 8M pixels and a USB connectivity facilitating direct connection to a laptop. The camera supports 30 frames per second (FPS) which is sufficient for our real-time application and can detect the environment with a field of view (FOV) of 75°. The FOV is the major characteristic impacting the system final workspace. Indeed, the defined FOV indicates the camera can detect in both X and Y directions within a range of [-37.5; +37.5]°. In comparison to the previous mechanical workspace defined, this one is smaller and will mark the limits of the system.

In addition, the detected object position in the image will be used to move the servo motors accordingly using forward kinematics. This can be done by remapping the pixels coordinates into the camera FOV range as the servo motors will have the same angles as the camera ones.

$$angle_i = \frac{(pixel_i - pixel_{min}) * (angle_{max} - angle_{min})}{pixel_{max} - pixel_{min}} + angle_{min}$$

Equation 5: Pixel to angle mapping formula

Additionally, the angle and pixel boundaries for both X and Y directions have been summarised in the Table 2.

Table 2: Pixel and angle boundaries for remapping

Parameters	X-axis	Y-axis
pixel_min	0	0
pixel_max	width	height
angle_min	-37.5	-37.5
angle_max	37.5	37.5

Although, the angles calculation must be carried out carefully as the camera and motor angles coordinates frames are different. Indeed, the motor angles are expressed in the original coordinate frame expressed by the X-axis along the horizontal line and the Y-axis along the vertical one with the origin in the bottom-left corner. However, the image coordinate frame is different with the axes inverted and the origin in the top-left corner as shown in the Figure 10.

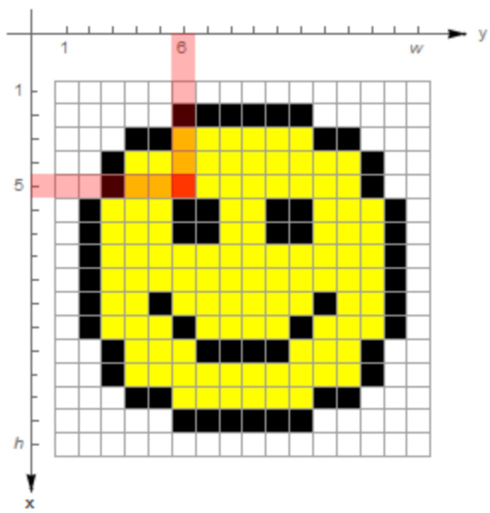


Figure 10: Illustration of the image coordinate system

Thus, the point coordinates must be translated into the world coordinate system using the Equation 6.

$$X_{world} = Y_{image}$$

$$Y_{world} = height_{image} - X_{image}$$

Equation 6: Image to world coordinate frame transformation

2.3. Robotics Programming (Baudouin and Pranav)

The real-time object tracking system uses powerful image processing algorithms to analyse camera feeds and extract useful information. Visual data is methodically analysed using a series of computational algorithms to identify items within the frame based on various properties such as shape, colour, and texture. Object localisation is subsequently accomplished by determining the spatial coordinates of these items within the frame, allowing exact tracking of their movements. This complex image processing pipeline shown in Figure 11 serves as the system's backbone, allowing flawless object detection and tracking in real time.

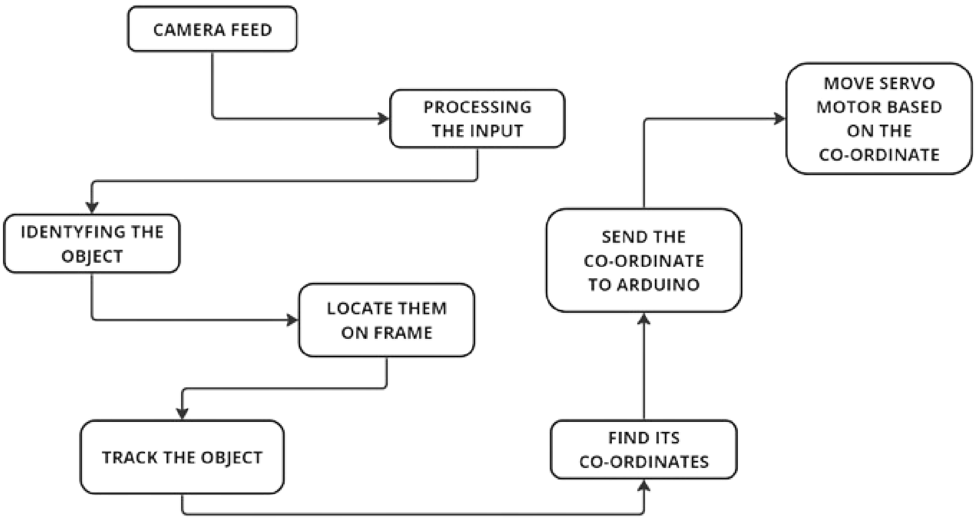


Figure 11: Object tracking process pipeline

2.3.a. Object detection and tracking

To recognise and track objects, an object detector combined to an OpenCV's KCF (Kernelized Correlation Filters) tracker have been implemented. The object detector aims to detect the object while the tracker aims at tracking its motion. Indeed, the tracker is not autonomous as it requires a bounding box at initialisation but is also not performant when the object depth varies as the bounding box has a fixed size. Thus, an object detector has been developed to automate the two previous tasks. Inversely, the object detector cannot work by itself to detect and track object motion as its performances are bad in tracking efficiently and smoothly moving objects without lagging. This is why the tracker is optimal by its capacities to track the object motion efficiently and accurately with a high level of confidence. Indeed, its main characteristics are further detailed:

- Efficient: It computes the filter and response map using circulant matrices and Fast Fourier Transforms (FFTs), resulting in computational efficiency.

- Accurate: It can handle a variety of difficult conditions, including occlusions, rotations, and scale changes, with excellent accuracy.
- Robust: It is less sensitive to variations in light, background clutter, and target object deformations.

The image processing algorithm starts by initially taking a frame from the video capture device and converts it from RGB to HSV colour space. Then, it generates a mask, resulting in a binary image, to separate green parts depending on the predefined HSV range. To improve the object detection, a Canny edge detector is combined for better result. From here, it detects potential objects, by discovering the contours in the mask and filters them depending on their shape, which is assumed to be a cube.

Once a closed surface is found with an area exceeding a specific threshold, a rectangle is estimated from it and returns the origin position and the size of the sides. These parameters can be now used to form the bounding box and feed the tracker.

From that, the tracker tracks the object motion at each new frame based on the initial bounding box given. In case a new bounding box is found, the tracker is reinitialised with this new box. Thus, the tracker outputs the new position of the bounding box tracked in the next frame. The object centre coordinates are retrieved using the formula in Equation 7.

$$X_{center} = X_{0,box} + \frac{width_{box}}{2}$$

$$Y_{center} = Y_{0,box} + \frac{height_{box}}{2}$$

Equation 7: Object centre coordinates formula

These coordinates are then remapped to angle values as shown in section 2.2.b and gathered to generate the data packet to send through serial communication as explained in the section 2.3.b. The global process for detecting and tracking objects previously discussed is summarised in the Figure 12.

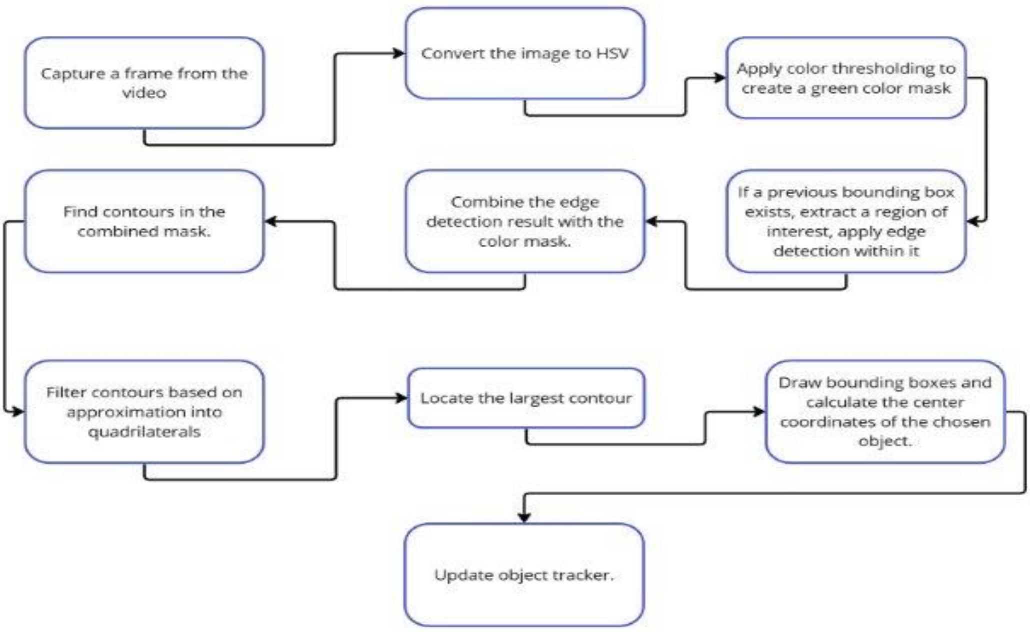


Figure 12: Object detection and tracking flowchart process

The implementation of a colour calibration and an FPS counter were also important to improve the system and to assess its performance. Indeed, the calibration function was implemented to adjust the detector colour range to the object tracked which can vary depending on the environment and illumination, or even to change the type of object colour to track. While the FPS counter aims at defining the image frequency rate assessing the performances of the different trackers. The framerate is computed using a running average based on 10 frame samples as demonstrated in Equation 8.

$$Framerate = \frac{N_{frames}}{t_{ellapsed}}\ FPS$$

Equation 8: Framerate calculation

2.3.b. Data transmission

After computing the angle motor commands to actuate accordingly the eye positions, the data must be encrypted and sent through serial communication to the Arduino board. The data encryption is a crucial step in the data transmission process as it facilitates the reading of the angles at reception. As the data submitted consists of a pair of two integer numbers, their values must be thoroughly retrieved. This consists of successfully reading the corresponding values for the X and Y motors without mixing them or reading undesired data. Thus, the structure shown in the Figure 13 has been used to avoid the issues previously stated.

$$data_{TX} = < angle_X, angle_Y >$$

Figure 13: Structure of the data transmitted

The opening chevron indicates the beginning of a new pair of angles and the next character represents the angle value for the X-direction motor. This value is followed by a comma indicating the change to the angle value for the Y-direction motor at the next character. Finally, the data transmission ending is marked by the presence of a closing chevron.

The data transmission is ensured on Python thanks to the 'serial' library which the object is initialised using the COM port, baud rate, and byte size. Before transmitting the data, it must be encoded into bytes using the ASCII format.

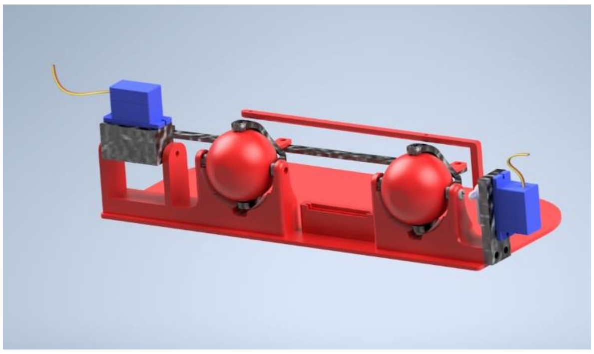
Upon data reception on the Arduino board, the entire string is read using the 'readString' Serial built-in method. This one allows to read the entire string at once instead of receiving byte per byte. The X and Y angles are retrieved using the substring method combined to the index of the chevrons, and the comma. The substrings obtained are converted to integers and trimmed in case the angles out of the constrained workspace which cause damages to the system. Finally, the trimmed values are fed into the servo motors using the 'servo' library.

3. Robot Design and Simulation

3.1. Final Design (Sam)

3.1.a. Assembly design

The final design consisted of all stationary parts being attached to a base to prevent any unnecessary movement. The camera was fitted in the centre of the two eyes using a secure and straight forwards slot that it fits into. The servo controlling rotation around the horizontal axis was placed to the right of the base and the servo controlling rotation around the vertical axis, via links at the back of the eyes was mounted in a way that allowed it to rotate with the eyes. This allowed the eyes to rotate both horizontally and vertically simultaneously.



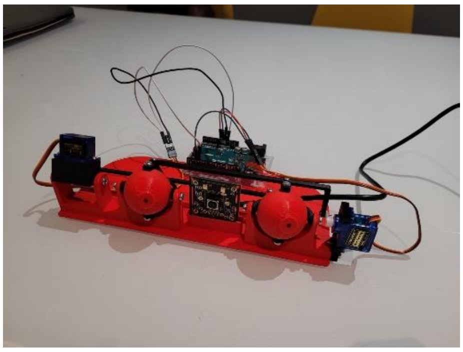


Figure 14: Final design

For the parts list and orthographic please refer to the Appendix section.

3.1.b. Fabrication

When fabricating the components of the design for both the final design and initial prototyping, it was decided that the Bambu Lab P1P 3D printer would be used due to its high accuracy and speed. This allowed for parts such as the eyes to be printed with a smooth finish and for larger parts such as the base, the Bambu was able to print the base alongside both eyes and multiple links in close to 3.5hr whereas a printer such as the Creality Ender 5 , which can produce prints of similar quality, the base alone would have taken close to 8 hours.

3.2. Real-time application (Baudouin)

In order to obtain a responsive system actuating nearly in real time, the object tracker and data transmission had to be improved to reduce latency.

3.2.a. Object detection and tracking

For the object detection and tracking algorithm, the combination of the object detector and the KCF tracker is optimal as it efficiently tracked the object with a very good accuracy while achieving reasonable FPS. Indeed, the framerate obtained for the KCF tracker was around 25-32FPS which enabled a smooth running of the application. In comparison to other tracker, this was the most efficient in object tracking with the highest framerate as some had rates around 5FPS resulting in lags and bad object tracking performance. Furthermore, it was first tried the object detector as the tracker, but the motion tracking was not fluid as the illumination varied depending on the object orientation.

In the Figure 15 is shown the output from the image processing algorithm detecting the object and tracking it. The thin closed loop in green represents the area detected by the object detector. The green rectangle refers to the bounding box detected used to feed the tracker. Finally, the blue point defines the object centre coordinates to be followed by the eyes.

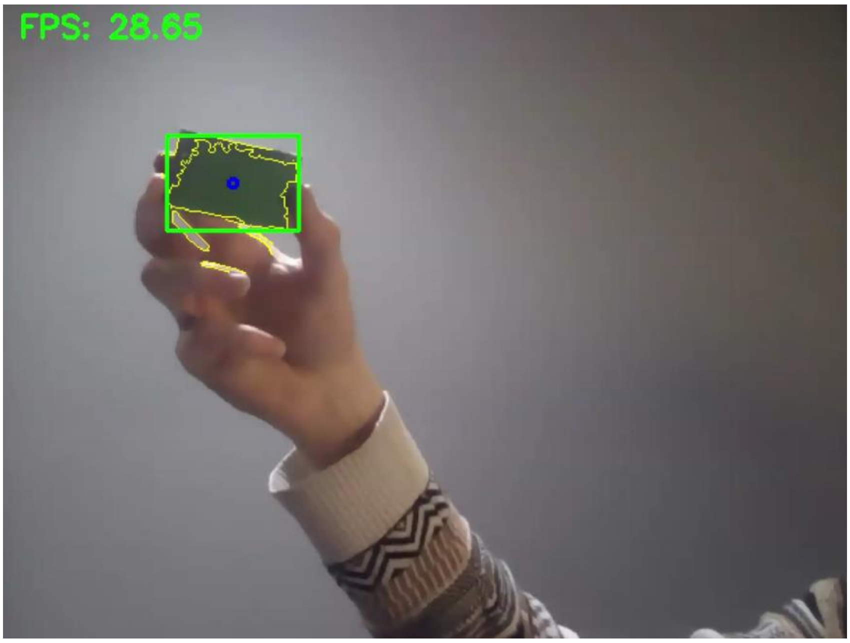


Figure 15: Object detection and tracking result

3.2.b. Data transmission

Concerning the data transmission, the timeout value for the ‘readString’ method had to be tuned as the default value of 1s caused high delay in the system response. Indeed, the ‘readString’ method is a blocking function that waits for the set period to ensure the reception of the entire data packet. Nonetheless, the required time to send the motor angles is much less at a baudrate of 115200 bits per seconds. The message to transmit contains at most 9 bytes as each angle value can be up to 3 bytes long and each other characters are 1 byte long. Thus, the transmission time to send the data packet has been calculated in the Equation 9.

$$n_{maximum\ number\ of\ bits} = bytes * bits\ per\ byte = 9 * 8 = 72\ bits$$

$$t = \frac{n_{maximum\ number\ of\ bits}}{baudrate} = \frac{72}{115200} = 625\ \mu s$$

Equation 9: Timeout value calculation based on the data transmission time

Thus, it only takes 625µs to transmit the data without counting the start and end bits. The final timeout value implemented is then 3ms accounting for error and safety margins.

3.3. Forward Kinematics (João)

Forward kinematics is a fundamental concept that deals with determining the position and orientation of the end-effectors of a robotic system based on the joint angles. In this section, we will explore the forward kinematics of our robotic system composed of multiple links and joints, aiming to understand how the configuration of the system evolves with changes in joint angles ‘ϑ1’ (along the x-axis) and ‘ϑ2’ (along the z-axis).

A trigonometrical approach was used to calculate the positions of the end effectors based on the angles and relative position to other structural points. Figure 16 shows the calculations, done on MATLAB, to obtain the positions of the end effectors. And Figure 17 displays a labelled sketch of the system.

```
% Define lengths of the links
L1 = 24;
L2 = 129;
L3 = 17;
L4 = 164;
L5 = 15;
L6 = 15;


% Input joint angle theta1
theta1 = 60; % Value between 45 and 135
theta2 = -60; % Values between -150 and -30

%Forward Kinematics Calculation

%Point 1 (Joint 1)
x1 = 0;
y1 = 0;
z1 = 0;

% Point 2
x2 = 0;
y2 = L1 * cosd(theta1);
z2 = L1 * sind(theta1);

% End-effector 2
x_end1 = 29;
y_end1 = y2;
z_end1 = z2;

% End-effector 1
x_end2 = L2;
y_end2 = y2;
z_end2 = z2;

% Point 3 (Joint 2)
x3 = 193;
y3 = 0;
z3 = 0;

% Point 4
x4 = x3 + L3 * cosd(theta2);
y4 = y3 + L3 * sind(theta2);
z4 = 0;

%Point 5
x5 = x4 - 64;
y5 = y4;
z5 = 0;

%Point 6
x6 = x4 - L4;
y6 = y4;
z6 = z4;

%End effector 3
x_end3 = L2;
y_end3 = y5 + L5;
z_end3 = z5;

%End effector 4
x_end4 = 29;
y_end4 = y6 + L6;
z_end4 = z6;
```

Figure 16: Forward kinematics calculation

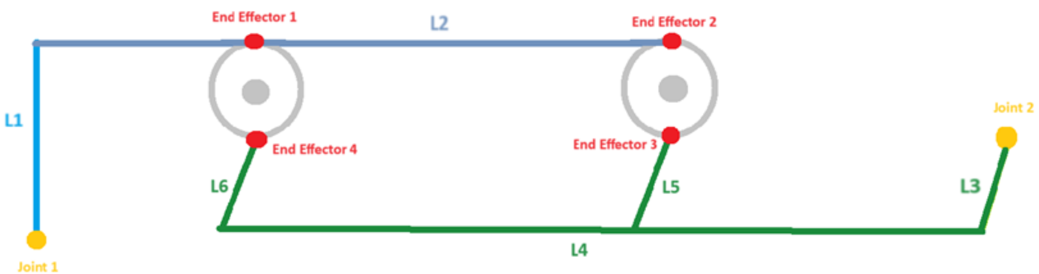


Figure 17: System sketch

Figures 18 shows two examples of this calculation. It gives the ‘x’,’y’ and ‘z’ positions of the end-effectors given ‘ϑ1’ and ‘ϑ2’.

```
theta1 = 60
theta2 = -60

xend1 = 29
yend1 = 12
zend1 = 20.7846

xend2 = 129
yend2 = 12
zend2 = 20.7846

xend3 = 129
yend3 = 0.27757
zend3 = 0

xend4 = 29
yend4 = 0.27757
zend4 = 0
```

```
theta1 = 45
theta2 = -150

xend1 = 29
yend1 = 16.9706
zend1 = 16.9706

xend2 = 129
yend2 = 16.9706
zend2 = 16.9706

xend3 = 129
yend3 = 6.5
zend3 = 0

xend4 = 29
yend4 = 6.5
zend4 = 0
```

Figure 18: Calculation example 1 (left) and 2 (right)

4. Future Work and Conclusion

4.1. Future Work

4.1.a. Design (Sam)

Future work that could be carried out to improve the design would be to align the axis the motor balance rotates around with the axis that the eyes rotate around. Due to the design being relatively compact, with the full design being less than 220mm in width, the decision to have the motor balance slightly out of alignment allowed ease of assembly as without doing this attaching the screws became challenging. The same was true about the axis that the servo that controlled vertical movement, see figure 19.

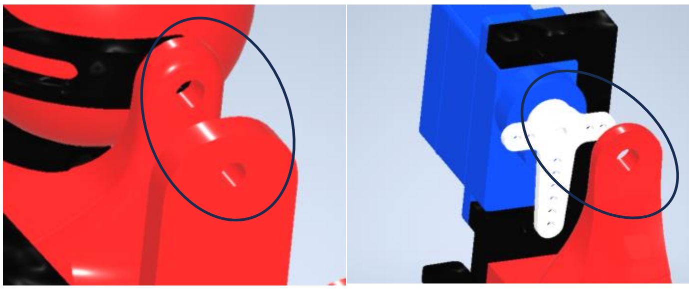


Figure 19: Areas of misalignment

The decision for both of these to be misaligned was made for ease of assembly and a solution to this would be to break the base into different parts so that parts could be assembled in an order that meant that there was no need to attach screws that are on the same axis of alignment as one and other.

Other design options that could potentially have been investigated include encasing the full design so that it had a more human like aesthetic. Another area that could have been investigated would be using more servos to allow the eyes to move independently

4.1.b. Electronics (Hari)

Switching from a servomotor to a stepper motor can significantly improve the performance of the designed system, particularly in terms of speed and acceleration control. Stepper motors provide precise step-by-step motion, allowing finer control of speed and acceleration profiles during tracking. Unlike the SG90 servomotors used which cannot control the speed nor acceleration, they lack precision in motion control, while stepper motors perform great at executing predefined speed and acceleration profiles with high accuracy. This capability ensures smoother, more predictable tracking movements, even in dynamic environments where objects can rapidly change speed or direction. By exploiting the speed and acceleration control in stepper motors, the tracking system becomes more responsive and agile, resulting in a more accurate and efficient object tracking system.

4.1.c. Robotics (Hari)

Integrating a CNN tracker into the existing system, instead of the object detection algorithm and the KCF tracker, offers several improvements. At first, CNNs improve accuracy by learning complex visual features, adapting to object variations, better handling occlusions and can be trained end-to-end for optimal tracking methods. Thus, CNNs object tracker could be more efficient on detecting object based on the trained features instead of searching for colour in the environment. This gives best possibilities as complex objects can be detected such as plastic bottles that would be impossible with the current algorithm. In addition, the CNN algorithms perform best for real-time applications as YOLO object tracker can reach framerate up to 45FPS which would improve the system latency. Overall, this integration enhances the system's tracking capabilities, providing more accurate, adaptive and robust object tracking in a variety of scenarios.

4.2. Conclusion (Hari)

In conclusion, the development of a human-eye rotation-inspired two-axis camera mounting system holds significant importance and offers a wide range of practical applications across various industries. By mimicking the natural movement of the human eye, this innovative system enhances the versatility, flexibility, and efficiency of camera positioning and surveillance tasks.

One of the major benefits of this system is its ability to provide dynamic and intuitive camera control, allowing users to capture a comprehensive view of the surroundings with enhanced situational awareness. This feature is particularly valuable in surveillance, security monitoring, and reconnaissance applications, where real-time tracking of objects or individuals is crucial for threat detection, crime prevention, and emergency response.

The two-axis camera mounting system can be utilised in fields such as cinematography, aerial photography, and virtual reality, where precise control over camera orientation and perspective is essential for capturing captivating visual content and immersive experiences. By enabling smooth and fluid camera movements, this system enhances creativity and storytelling in multimedia production and content creation. To further enhance its usefulness and effectiveness, ongoing system improvements can be implemented to address various challenges and limitations. For instance, advancements in motor technology, control algorithms, and sensor integration can optimise the system's performance, accuracy, and reliability. Additionally, the development of user-friendly interfaces, remote control capabilities, and automation features can streamline operation and increase user satisfaction.

Overall, the human-eye rotation-inspired two-axis camera mounting system represents a groundbreaking innovation with vast potential for improving surveillance, cinematography, and other imaging applications. Its ability to replicate natural eye movements, coupled with continuous advancements in technology, makes it a valuable tool for enhancing visual perception, capturing compelling imagery, and shaping the future of imaging technology.

5. Work Distribution

<table>
<tr><th>Names</th><th>Tasks contribution</th></tr>
<tr><td>Baudouin BELPAIRE</td><td>Robotics:
– Programming of the object detector algorithm and implementation of the KCF tracker
– Programming of the data transmission
– Programming of the Arduino code for motor controls
Electronics:
– Transformation of the image coordinate frame
– Remapping of object position into motor angles
– Calculation of the data transmission speed
Mechanical:
– Design of the eye motion system
– Realisation of the 1st prototype
– Calculation of the mechanical parameters</td></tr>
<tr><td>João COELHO</td><td>Electronics:
– Motor analysis and selection
Mechanical:
– Forward Kinematics analysis</td></tr>
<tr><td>Hari GOVIND Achanveettil Puthukkad</td><td>Introduction:
– Overview about the project
Conclusion and future work
References</td></tr>
<tr><td>Pranav SAJIKUMAR</td><td>Programming:
– Object Tracking
• Worked On KCF Algorithm
• Tested CSRT Tracker
• Tested DeepSORT Algorithm
– Worked On Optimising Code
– Worked On Optimising Bounding Box
– Worked On Coordinate Extraction</td></tr>
<tr><td>Sam WATSON</td><td>Mechanical:
– Initial designs for eye mechanism
– Development of final design of robot
– Fabrication and construction of final design including redesign of multiple components
– Generation of all Orthographic and all CAD images</td></tr>
</table>

References (Hari)

1. S. S. Ge, C. C. Hang, and T. H. Lee, "Design and control of a two-degree-of-freedom camera platform for tracking moving objects," *IEEE Transactions on Robotics and Automation*, vol. 17, no. 6, pp. 922-927, Dec. 2001.

2. M. Liu, Z. Wang, and Q. Zhang, "Development of a human-eye inspired camera mount with adaptive zoom control," *Journal of Mechanical Engineering Research*, vol. 8, no. 1, pp. 1-12, Jan. 2016.

3. R. M. Haralick and L. G. Shapiro, "Human eye models for pattern recognition," *Computer Vision, Graphics, and Image Processing*, vol. 44, no. 1, pp. 12-38, Apr. 1988.

4. Y. C. Tung and R. M. Brown, "A novel approach to two-axis camera stabilization using an adaptive fuzzy-logic controller," *IEEE Transactions on Industrial Electronics*, vol. 49, no. 6, pp. 1294-1306, Dec. 2002.

5. A. El-Sayed, S. Shehata, and A. Abdel-Dayem, "Design and implementation of a human-eye-inspired camera mounting system for surveillance applications," *International Journal of Electrical and Computer Engineering*, vol. 7, no. 6, pp. 3329-3337, Dec. 2017.

6. J. S. Son, K. J. Lee, and J. H. Kim, "Development of a two-axis camera gimbal system for unmanned aerial vehicles," *IEEE/ASME Transactions on Mechatronics*, vol. 22, no. 2, pp. 726-735, Apr. 2017.

7. N. A. B. Adzman, M. N. Taib, and N. M. Din, "Development of a two-axis gimbal system for cinematography applications," *Journal of Automation, Control and Intelligent Systems*, vol. 1, no. 1, pp. 18-26, Jan. 2017.

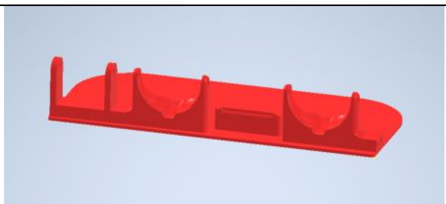
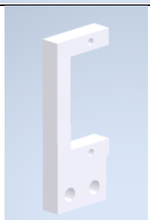
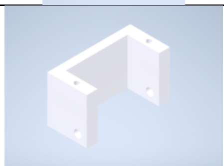
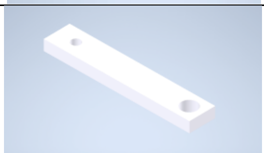
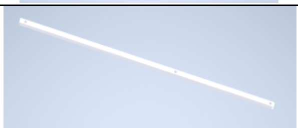
8. D. Kim, J. Park, and K. J. Lee, "Bio-inspired design of a two-axis camera mount using saccadic eye movements," *Bioinspiration & Biomimetics*, vol. 14, no. 4, pp. 046013, Jul. 2019.

9. H. J. Kim, S. W. Yoon, and J. Y. Lee, "Development of a human-eye inspired camera mount with real-time image stabilization," *International Journal of Precision Engineering and Manufacturing-Green Technology*, vol. 6, no. 2, pp. 367-375, Mar. 2019.

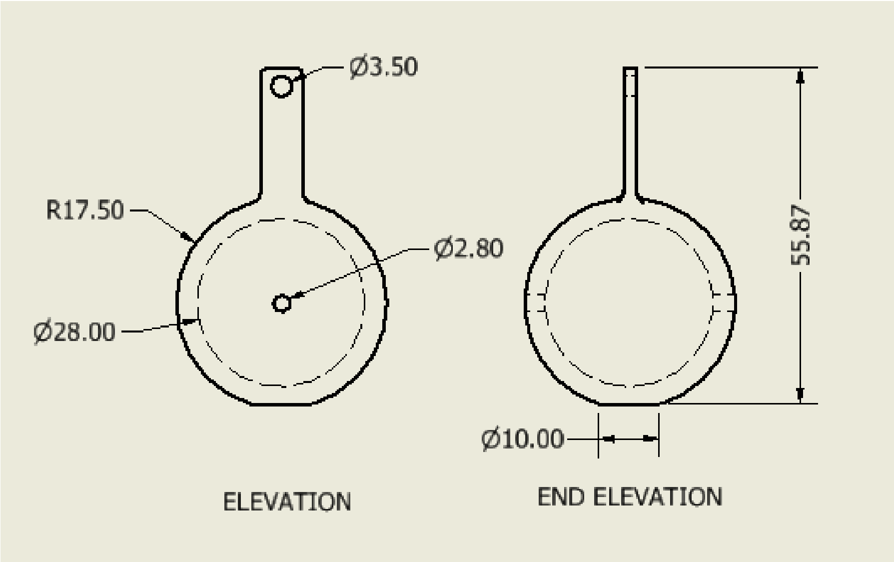
10. W. J. Lee, J. Y. Choi, and J. S. Hong, "Design and implementation of a two-axis camera gimbal system for drone-based aerial photography," *Journal of Intelligent & Robotic Systems*, vol. 95, no. 3, pp. 621-635, Sep. 2019.

Appendix (Sam)

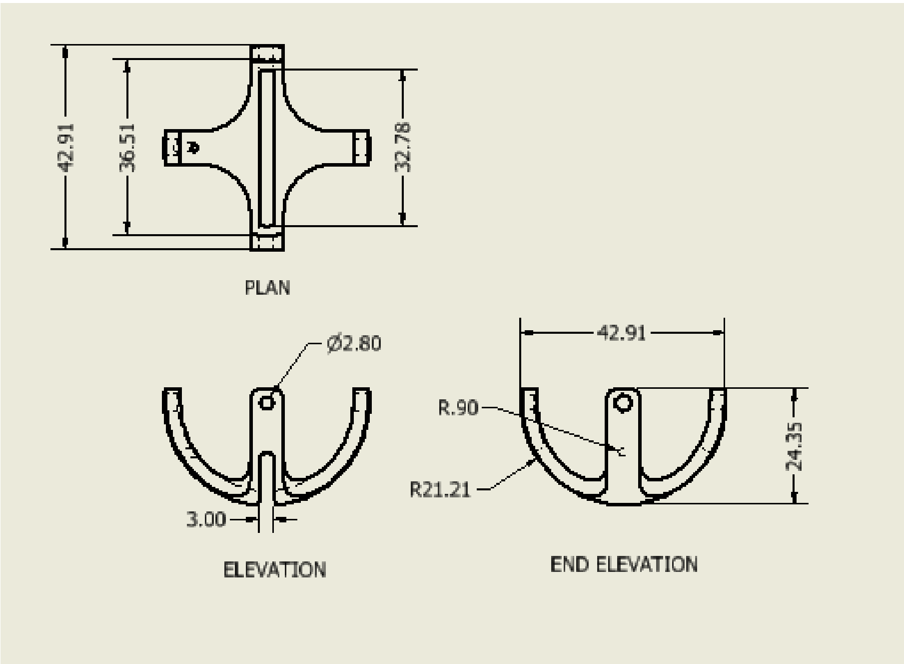
Final Design Parts List

Part		Number of Parts
Base		1
Eye		2
Eye Link		2
Vertical servo mount		1
Horizontal servo balance		1
Vertical movement link		1
Horizontal link 1		1
Horizontal link 2		1

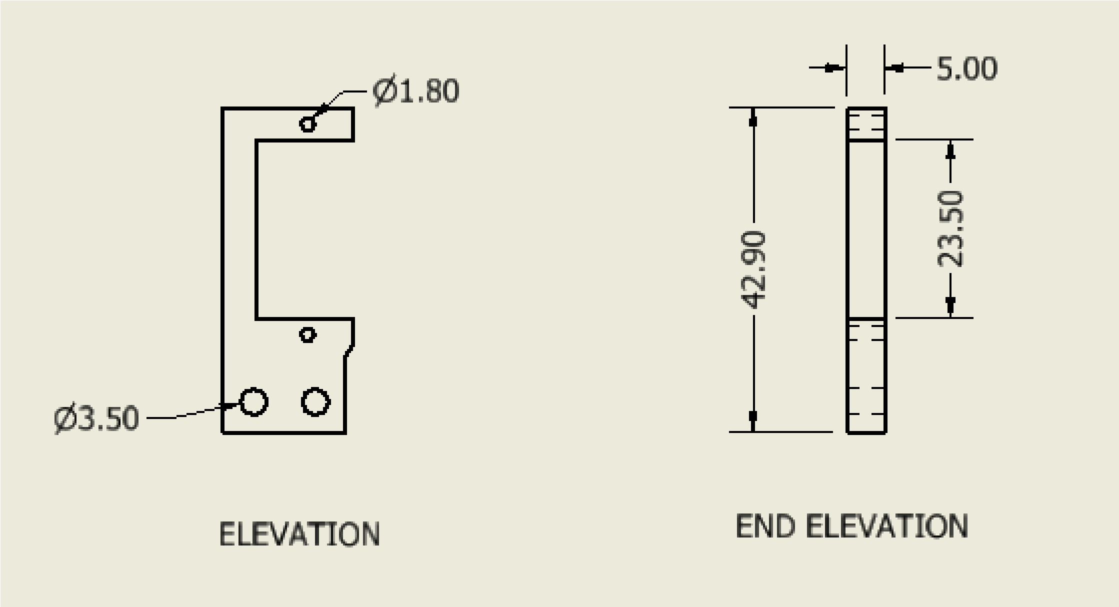
Eye Orthographic



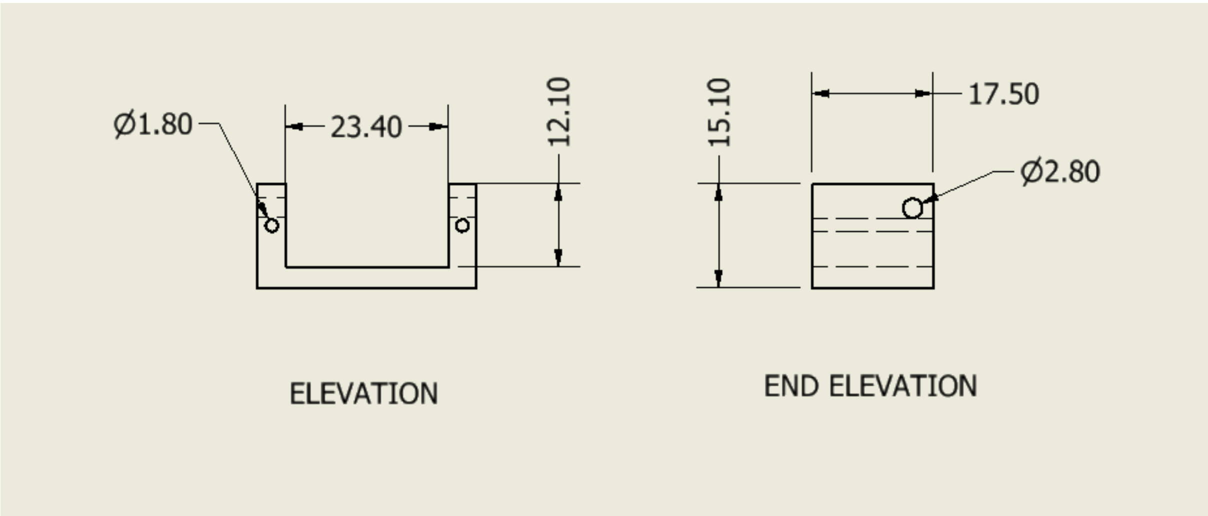
Eye link



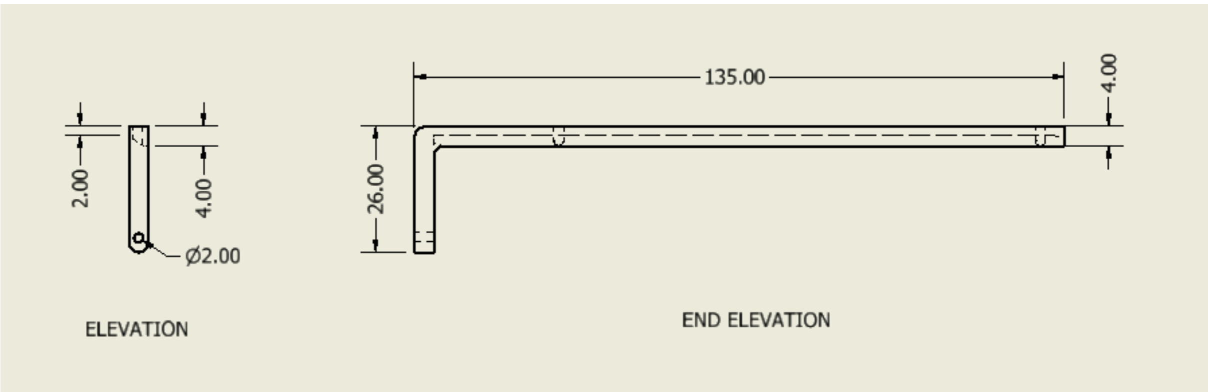
Vertical Servo Mount



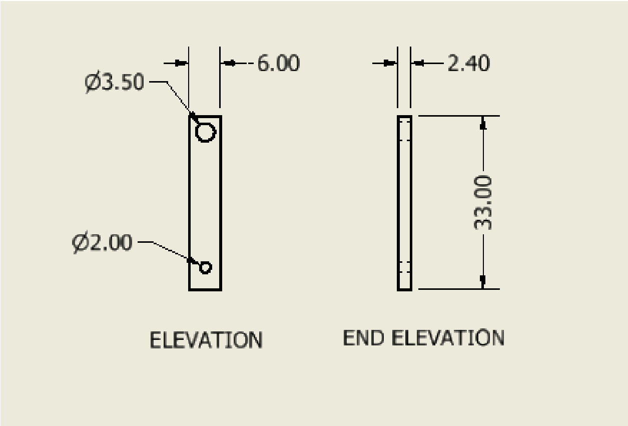
Horizontal Servo Balance



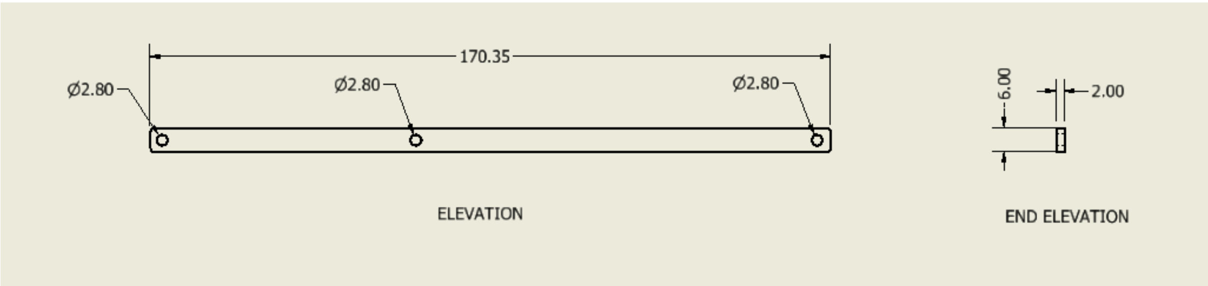
Vertical Movement Link



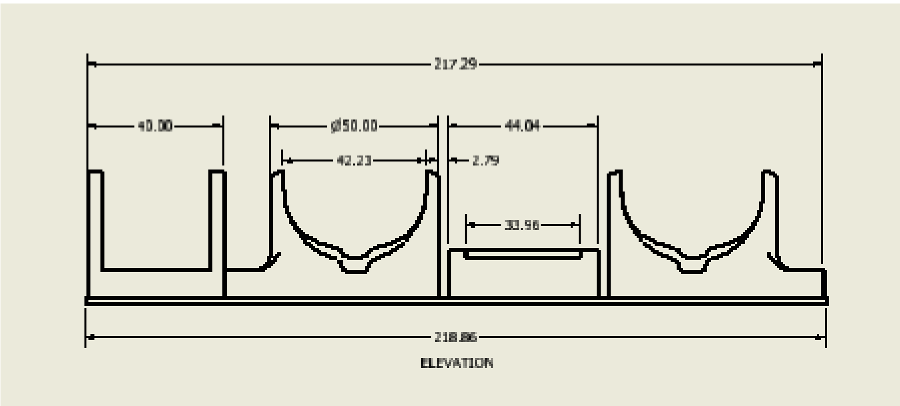
Horizontal Movement Link 1



Horizontal Movement Link 2



Base



Course code and name:	B51RO – Robotics Mechanical Systems
Type of assessment:	Group
Coursework Title:	Case Study
Student Name:	Baudouin BELPAIRE
Student ID Number:	H00445613

Declaration of authorship. By signing this form:

- **I declare** that the work I have submitted for individual assessment OR the work I have contributed to a group assessment, is entirely my own. I have NOT taken the ideas, writings or inventions of another person and used these as if they were my own. My submission or my contribution to a group submission is expressed in my own words. Any uses made within this work of the ideas, writings or inventions of others, or of any existing sources of information (books, journals, websites, etc.) are properly acknowledged and listed in the references and/or acknowledgements section.

- I confirm that I have read, understood and followed the University's Regulations on plagiarism as published on the University's website, and that I am aware of the penalties that I will face should I not adhere to the University Regulations.

- I confirm that I have read, understood and avoided the different types of plagiarism explained in the University guidance on Academic Integrity and Plagiarism

Student Signature: *Baudouin BELPAIRE*

Date: *20/04/2024*

Copy this page and insert it into your coursework file in front of your title page.
For group assessment each group member must sign a separate form and all forms must be included with the group submission.

Your work will not be marked if a signed copy of this form is not included with your submission.

Course code and name:	B51RO – Robotics Mechanical Systems
Type of assessment:	Group
Coursework Title:	Case Study
Student Name:	João COELHO
Student ID Number:	H00342606

Declaration of authorship. By signing this form:

- **I declare** that the work I have submitted for individual assessment OR the work I have contributed to a group assessment, is entirely my own. I have NOT taken the ideas, writings or inventions of another person and used these as if they were my own. My submission or my contribution to a group submission is expressed in my own words. Any uses made within this work of the ideas, writings or inventions of others, or of any existing sources of information (books, journals, websites, etc.) are properly acknowledged and listed in the references and/or acknowledgements section.

- I confirm that I have read, understood and followed the University's Regulations on plagiarism as published on the University's website, and that I am aware of the penalties that I will face should I not adhere to the University Regulations.

- I confirm that I have read, understood and avoided the different types of plagiarism explained in the University guidance on Academic Integrity and Plagiarism

Student Signature: *João Lopes*

Date: *20/04/2024*

Copy this page and insert it into your coursework file in front of your title page.
For group assessment each group member must sign a separate form and all forms must be included with the group submission.

Your work will not be marked if a signed copy of this form is not included with your submission.

Course code and name:	B51RO – Robotics Mechanical Systems
Type of assessment:	Group
Coursework Title:	Case Study
Student Name:	Hari GOVIND
Student ID Number:	H00449213

Declaration of authorship. By signing this form:

- **I declare** that the work I have submitted for individual assessment OR the work I have contributed to a group assessment, is entirely my own. I have NOT taken the ideas, writings or inventions of another person and used these as if they were my own. My submission or my contribution to a group submission is expressed in my own words. Any uses made within this work of the ideas, writings or inventions of others, or of any existing sources of information (books, journals, websites, etc.) are properly acknowledged and listed in the references and/or acknowledgements section.

- I confirm that I have read, understood and followed the University's Regulations on plagiarism as published on the University's website, and that I am aware of the penalties that I will face should I not adhere to the University Regulations.

- I confirm that I have read, understood and avoided the different types of plagiarism explained in the University guidance on Academic Integrity and Plagiarism

Student Signature: *HARIGOVIND ACHANVEETTIL PUTHUKKAD*

Date: *20/04/2024*

Copy this page and insert it into your coursework file in front of your title page.
For group assessment each group member must sign a separate form and all forms must be included with the group submission.

Your work will not be marked if a signed copy of this form is not included with your submission.

Course code and name:	B51RO – Robotics Mechanical Systems
Type of assessment:	Group
Coursework Title:	Case Study
Student Name:	Pranav SAJIKUMAR NAIR
Student ID Number:	H00436046

Declaration of authorship. By signing this form:

- **I declare** that the work I have submitted for individual assessment OR the work I have contributed to a group assessment, is entirely my own. I have NOT taken the ideas, writings or inventions of another person and used these as if they were my own. My submission or my contribution to a group submission is expressed in my own words. Any uses made within this work of the ideas, writings or inventions of others, or of any existing sources of information (books, journals, websites, etc.) are properly acknowledged and listed in the references and/or acknowledgements section.

- I confirm that I have read, understood and followed the University's Regulations on plagiarism as published on the University's website, and that I am aware of the penalties that I will face should I not adhere to the University Regulations.

- I confirm that I have read, understood and avoided the different types of plagiarism explained in the University guidance on Academic Integrity and Plagiarism

Student Signature: *Pranav Sajikumar Nair*

Date: *20/04/2024*

Copy this page and insert it into your coursework file in front of your title page. For group assessment each group member must sign a separate form and all forms must be included with the group submission.

Your work will not be marked if a signed copy of this form is not included with your submission.

Course code and name:	B51RO – Robotics Mechanical Systems
Type of assessment:	Group
Coursework Title:	Case Study
Student Name:	Sam WATSON
Student ID Number:	H00344148

Declaration of authorship. By signing this form:

- **I declare** that the work I have submitted for individual assessment OR the work I have contributed to a group assessment, is entirely my own. I have NOT taken the ideas, writings or inventions of another person and used these as if they were my own. My submission or my contribution to a group submission is expressed in my own words. Any uses made within this work of the ideas, writings or inventions of others, or of any existing sources of information (books, journals, websites, etc.) are properly acknowledged and listed in the references and/or acknowledgements section.

- I confirm that I have read, understood and followed the University's Regulations on plagiarism as published on the University's website, and that I am aware of the penalties that I will face should I not adhere to the University Regulations.

- I confirm that I have read, understood and avoided the different types of plagiarism explained in the University guidance on Academic Integrity and Plagiarism

Student Signature: *Sam Watson*

Date: *20/04/2024*

Copy this page and insert it into your coursework file in front of your title page. For group assessment each group member must sign a separate form and all forms must be included with the group submission.

Your work will not be marked if a signed copy of this form is not included with your submission.