

Ball-catching Robot

Project Plan Group B

Olle Flitig¹ Sara Rask² Björn Duktig³ Amanda P. Solver⁴

Project Advisor: Mycke K. Unnig

¹olle.flitig@student.lu.se

²sara.rask@student.lu.se

³bjorn.duktig@student.lu.se

⁴amanda.solver@student.lu.se

1 Project Purpose

The goal of this project is to make a robot arm that can catch balls based on visual feedback. The work will be based on the PhD-thesis (TFRT-1098) made by Magnus Linderöth.

2 Equipments and material

In this section we propose a list containing the materials needed for the project. We have appended the approximate costs of each item on the list — based on the research we have been doing online.

2.1 Robotics

- Robotic arm metal frame with six degrees of freedom. (approx. 1000 SEK)
- MG996R Servo x 4 for movement off the robotic arm. (approx. 80 SEK/each)
- MG90S Servo x 2 for movement off the robotic arm. (approx. 80 SEK/each)
- Arduino Uno R3 to generate pwm signals to the servos. (approx. 200 SEK)
- 5V DC power supply for servos. (approx. 100 SEK)
- 12V DC power supply for arduino. (approx. 100 SEK)

2.2 Ball detection

- High speed cameras (basler ACA800-200GC) x 2. (approx. 1000 SEK/each)
- PoE gigabit switch. (approx. 1000 SEK)
- Gigabit ethernet cable x 3. (approx. 100 SEK/each)

3 Modelling and System Design

We are proposing the code structure seen in Figure 1 for our project. Considering a structure using four different packages the project can be split up and developed independently from one another. Furthermore, the packages represent the division of labour proposed in section 4
The hardware setup will include:

- The robot arm.
- Two cameras mounted behind the robot.
- An arduino to distribute signals to the servos in the robot arm.
- A PC to run the software on.

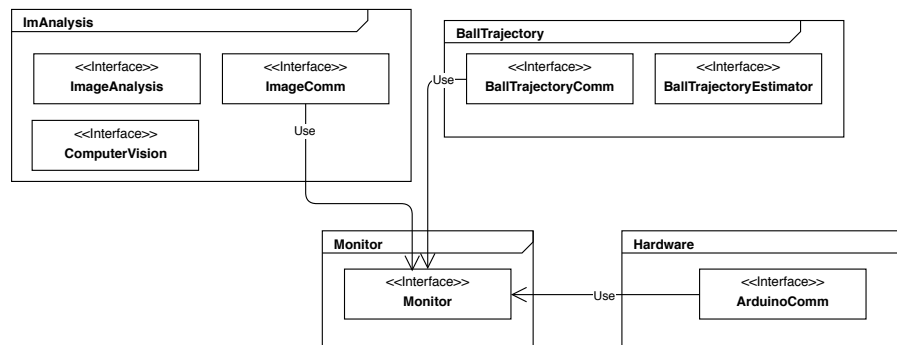


Figure 1: The structure of the software.

The overall software design can be found in Figure 1. We have aimed for a design with low coupling between the different modules, the monitor package will be the central connection point of the program. This package will communicate information to the remaining packages. Since most packages will be able to run independently, they will run on different threads. Most software will be written in java, the exception being the software that will run on the arduino, this will be written in the "arduino language".

4 Division of labour

We have decided to split our project into three parts: image analysis, hardware and servo position implementation, and finding ball trajectory and general code structure.

4.1 Image analysis

The image analysis is a major part of our project. Without it working, we will not have a reference for the ball position. Therefore, if the image analysis is not working, we will not have a working project. We will be using two cameras (see Section 2) to determine the position of the ball.

Olle Flitig and Sara Rask will be in charge of this section. We decided that it was a two person job since it is a quite daunting task.

4.2 Hardware and servo position

We will try to control the position of the servos based on a 3D point (which tells us where the ball is supposed to land). Most of the hardware implementation and the servo control will be handled by this package. Björn Duktig is in charge of this section.

4.3 Ball trajectory and general code structure

We will try to estimate the final position of the ball based on a few Cartesian points (identified by the image analysis section). Furthermore, the general structure of the code will be handled by this task (e.g. the observer-observable implementation and the semaphores in the monitor). Amanda P. Solver is in charge of this section. It is also planned that she will aid in the "optimal" trajectory estimation for the robot arm (Hardware section).

5 Time Plan

5.1 Subtasks

- Create a Monitor package including classes used by the image analysis and the hardware systems. Implement Observer-Observable and real time behaviour. **Estimated deadline:** 8/11
- Assemble the robot arm. **Estimated deadline:** 12/11
- Set up a display in such a way that the cameras and the robot arm have stationary positions. **Estimated deadline:** 12/11
- Implement communication between arduino and PC. **Estimated deadline:** 12/11
- Find mathematical model for how the ball trajectory should be estimated. Implement this into the ballTrajectory package. **Estimated deadline:** 15/11
- Find a model for the arm trajectory. **Estimated deadline:** 19/11
- Create a package for how the servos should respond to the given arm trajectory. **Estimated deadline:** 19/11
- Perform image segmentation to find the ball in the two cameras. **Estimated deadline:** 3/12
- Make a computer vision system to map the ball segments in the images to a 3D-coordinate. Including calibration and on-line calculations. **Estimated deadline:** 3/12
- (Optional); Implement feedback control of robot arm position **Estimated deadline:** N/A

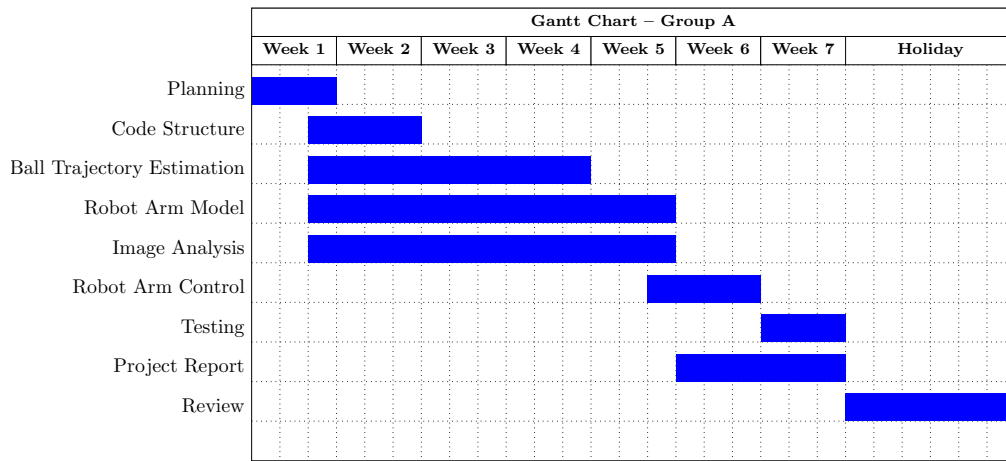


Figure 2: The Gantt chart describing the work flow of our project.

5.2 Important dates

- 10/11 - Hand in project plan.
- 19/11 - Feedback seminar 1 on the modeling and design.
- x/11 - Report should be pushed to git to allow peer review by other groups.
- 10/12 - Feedback seminar 2 on the design and implementation.
- 17/12 - The whole process should be finished and ready for demonstration.
- 20/12 - A complete project report should be pushed to git and the supervisor notified.
- x/1 - A final revised report should be pushed to git and the supervisor notified.
- x/1 - Project presentation and demonstration.

5.3 Gantt Chart

We have formalized the time plan as a Gantt chart. The major tasks can be seen plotted in Figure 2.