fhwedel
UNIVERSITY OF APPLIED SCIENCES

IT Project

KUKA & JENGA

Name : **Chien-Hsun Chao**

Field of Study : **IT Engineering**

Matriculation number : **ITE104218**

Semester of study : **WS2022**

1. The developing progress of project

1.1 KUKA Robot

1.1.1 Understanding initialization set-up

- **Task description**

Understanding how to work KUKA with the WorkVisual Software Program, for example., how to create a new project, and how to connect the real KUKA robot with the WorkVisual software. After realizing the basic operation of the WorkVisual software, and using WorkVisual software to manipulate the KUKA robot moving to the first random position.

1.1.2 Moving between four corners' coordinates

- **Task description**

To get much more familiar with the operation of the KUKA robot, the four corners' coordinates are recorded by moving the KUKA robot manually. Once those four coordinates have been collected, we use the point-to-point method to move the KUKA to these four points.

- **Problem analysis**

If only using the point-to-point method with these four coordinates, and may cause the robot cannot move to each point as our expectation. when the KUKA reaches the first position, the status of the KUKA is nearly close to the ground surface. Once the KUKA robot is moving to the next position, it will rub on the ground and may cause unexpected damage to itself or the ground.

- **Resolution analysis**

To solve the possibility of damaging the working surface issue, adding a few linear movements to smooth the whole process. After reaching the first corner points, the KUKA robots need to raise its arm on the z-axis by 20 millimetres before getting to the next position. This linear movement avoid the gripper from rubbing on the ground.

1.1.3 Customized a gripper

- **Task description**

Due to the KUKA do not have an appropriate gripper for the specific project, a customized gripper is necessarily built for clamping up one piece of JENGA's block. As Figure 1 shows below, this task utilized the FreeCAD software to construct a gripper by measuring the shape size of one block. After a model of the gripper is built, we used a 3D printer to create the gripper, which is shown in Figure 2 below.

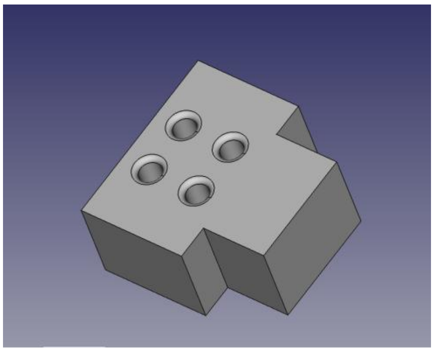

Figure 1 The customized gripper - FreeCAD model

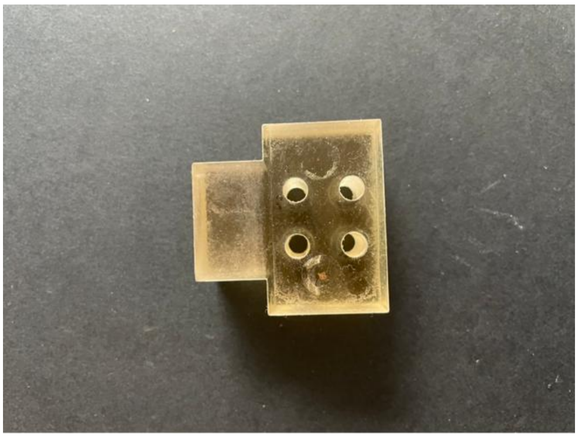

Figure 2 The customized gripper - 3D prints

1.1.4 Activate gripper

- **Task description**

Enabling the gripper to open and close by controlling from the work visual Software. Among the work visual software, an $OUT array has been used to formulate the switch of the air valve. The task will be by assigning a boolean value to the array relating to the on-switch to enable the gripper to open.

The schematic of the gripper on and off are shown on Figure 3 and Figure 4:

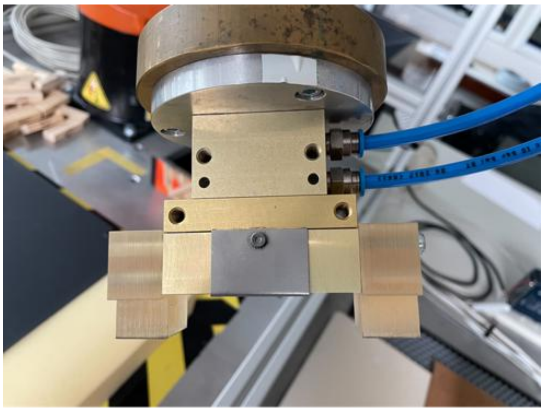

Figure 3 The gripper On-Switch

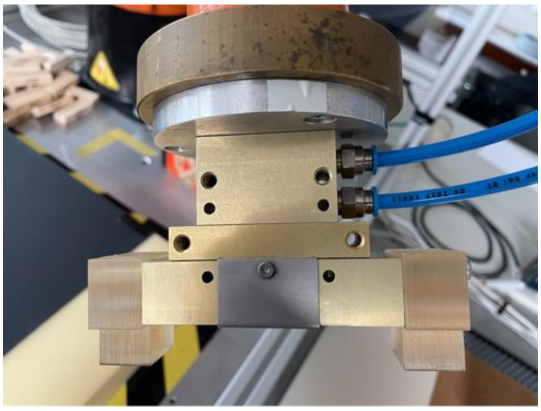

Figure 4 The gripper Off-Switch

1.1.5 Construct the first layer

- **Task description**

Regarding picking up a block and placing it in the working area centre, firstly we need three specific positions where can let the operator continuously place a block on the target coordinates.

The three specific positions have been marked as shown in the Figure 5 below.

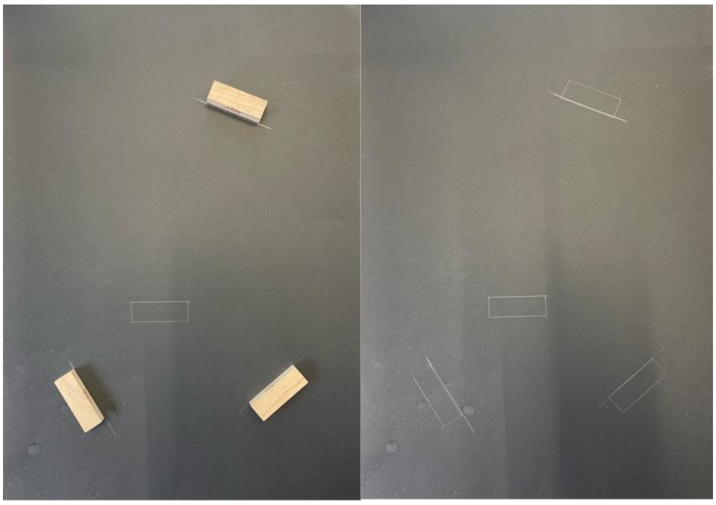

Figure 5 The three marked positions

We created a for-loop to iterate these three picking areas, which are outside the central area. And the gripper will start building a first layer which the long side along with the robot's X axis, and the long side of the second layer is along with the robot's Y axis. Additionally, the first block of the first layer coordinate in the central area also has been recorded. When the gripper picks up the second block and moves to the centre to place down the block, the coordinate of the dropping location will increase the y-axis value with 27 mm, which is the width of one block and keep iterating the increment until the first layer is completed. The expectation of JENGA's first layer is shown as Figure 6 below.

Figure 6 The first layer

- **Problem analysis**

While the gripper is on the coordinate of the block which should be picked, it closes and clamps a target. However, if the gripper is too close to the ground, the protecting sponge will be moved along as the gripper clamps. And also, due to the way of code executing, the time of clamping a target and reaching the target position simultaneously happened. This not only impacts the accuracy of the following coordinate and other target coordinates but also alters all fixed values of the target position.

- **Resolution analysis**

Additionally, from the Figure 7 shown below, we decrease the z-axis value of the KUKA when it approached the target position. It will not thoroughly contact the ground.

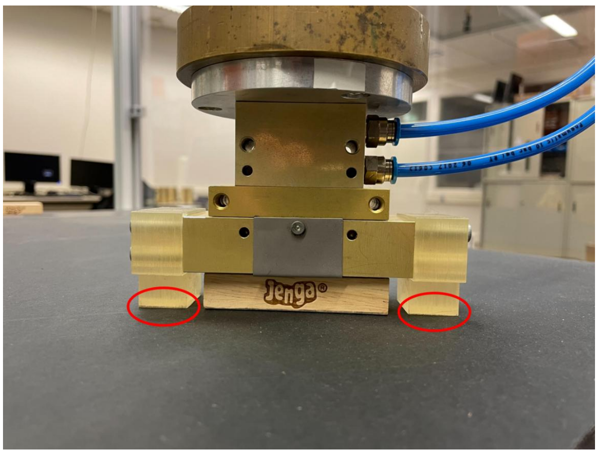

Figure 7 The demonstration of the proper z-axis value

1.1.6 Placing block on second layer

- **Task description**

After completing the first layer of the JENGA, we considered rotating the last joint of the KUKA called A6 to 90 degrees. To do so, we assign a flag in a program to identify whether the layer which long side along with the robot's X axis is completed. Once this layer is completed and the KUKA has clamped the fourth block from the first picking area, the KUKA robot rotates its last joint to right 90 degrees to the centre position. And it is ready to straightly place down the block in the first position of the layer which long side along with the robots' Y axis.

- **Problem analysis**

Compared to the orientation along with the robots' Y-axis layer, the orientation along with the robots' X-axis layers' intermediate point has a different measured value. The coordinate of the intermediary point is only satisfied with the robots X-axis layer condition as opposed to the robots Y-axis layer when the KUKA has clamped the fourth block and moved to the intermediary point prepared to straight down to place.The incorrect intermediary point affects the entire placement of building JENGA.

- **Resolution analysis**

The easiest way to resolve the different intermediary points is that manually measure the different intermediary points for the robots Y-axis's layer. The algorithm changes the coordinate value of the intermediary point to one that is appropriate for the robots' X orientation when the KUKA robot begins creating the layer for the robots' Y orientation. The schematic of the different orientations of the intermediary points is shown as Figure 8 below.

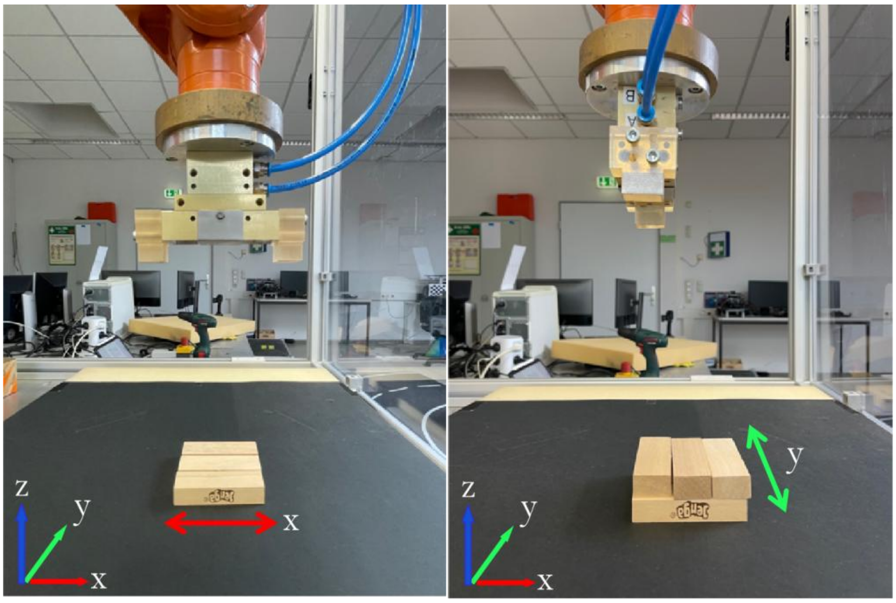

Figure 8 The tool in robots' X-axis and robots' Y-axis

1.1.7 Build JENGA with giving three specific areas

- **Task description**

After building two layers of the JENGA, it should consider about to construct an entire JENGA. The design of the temporary testing code for the KUKA robot allows it to pick up blocks in three designated locations, therefore the three designated locations are where the movement of the KUKA robot is expected to be presented. The operator should keep putting those blocks in the designated location until all of them have been utilized.

- **Problem analysis**

The orientation of the block's placement could affect the consequence of how the gripper picks up a block. If a block is not placed correctly, the gripper could clamp it in a weird position and when it reaches the dropping area, the block could be placed at an incorrect coordinate. This could affect the next pick-drop movement from subsequent blocks.

The following Figure 9 shows the correct and incorrect clamping positions.

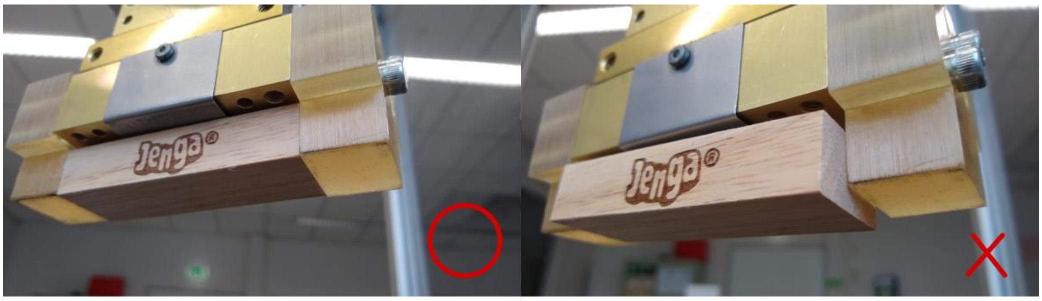

Figure 9 the correct clamping and the incorrect clamping

- **Realization analysis**

Due to the current task could be a temporary assignment, eventually, blocks will be picked up with the recognition from the Cognex camera. The Cognex recognition may increase the precision of the scattered blocks and transfer those block records to the KUKA robots.

Nevertheless, the temporary solution for this issue is that we can calibrate either every coordinate of the target or the arrangement of placing block by the operator.

1.1.8 Knocked JENGA Tower Down

- **Task description**

The last action of the KUKA Robots will be collapsing the JENGA Tower after a complete JENGA Tower has been constructed with three points. The approach we do is by having the KUKA move immediately to the spot that is prepared for demolishing the tower after it has finished its last placing movement. And the position will be on the left side of the JENGA tower. It will travel directly to the centre area once it has reached the ready position.

- **Problem analysis**

Due to the point-to-point simplicity of the movement instruction, the KUKA will go directly to the knocking place without avoiding any nearby barriers. Based on this, there was a problem with the KUKA destroying the JENGA tower in an inappropriate manner, as depicted in Figure 10 below. Additionally, a block itself or even many blocks might be damaged by this damage operation.

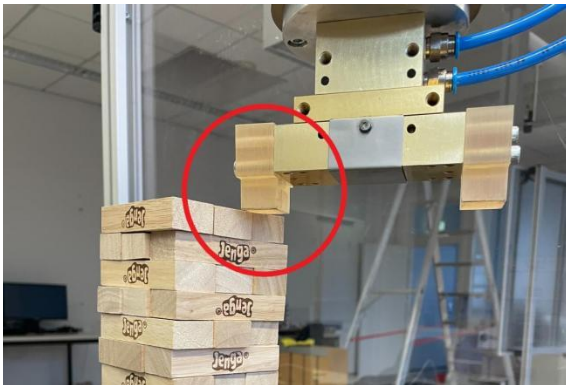

Figure 10 The incorrect knocking movement

- **Realization analysis**

The knocking movement's track is an issue that exists, as previously mentioned. A remedy to this would be to use a suitable and reasonable starting location while demolishing the Tower.

1.2 Cognex Camera

1.2.1 Blobs functions in Cognex

- **Task description**

The first task of manipulating the Cognex camera is that getting familiar with the functions in the Cognex software. The blobs function is able to create a region that identifies with items coordinate which is under the camera.

In our task, based on the part of the KUKA robot', we have already created a series of movements depending on three specific positions. Therefore, we used the three blobs function to recognise these three positions in the EasyView.

Moreover, the detailed information on these three blobs functions can be seen from the spreadsheet interface, which is represented in Excel format.

1.2.2 Scaling from pixel coordinate to world coordinate

- **Task description**

For the KUKA robot, they have various orientations and scaling factors depending on the coordinates they collected from Cognex. Thus, the transformation from Cognex to KUKA is essential in order to complete this project. According to the Figure 11 below, the Cognex camera features pixel coordinate orientation that along with downward the x-axis increase and the value of the y-axis increase as right.

However, the world coordinate of the KUKA robot has thoroughly opposite, which can be seen in the schematic below. The increment of the x-value is moving upward and the y-axis is moving leftward. As a result, we must synchronize the coordinates of Cognex and KUKA.

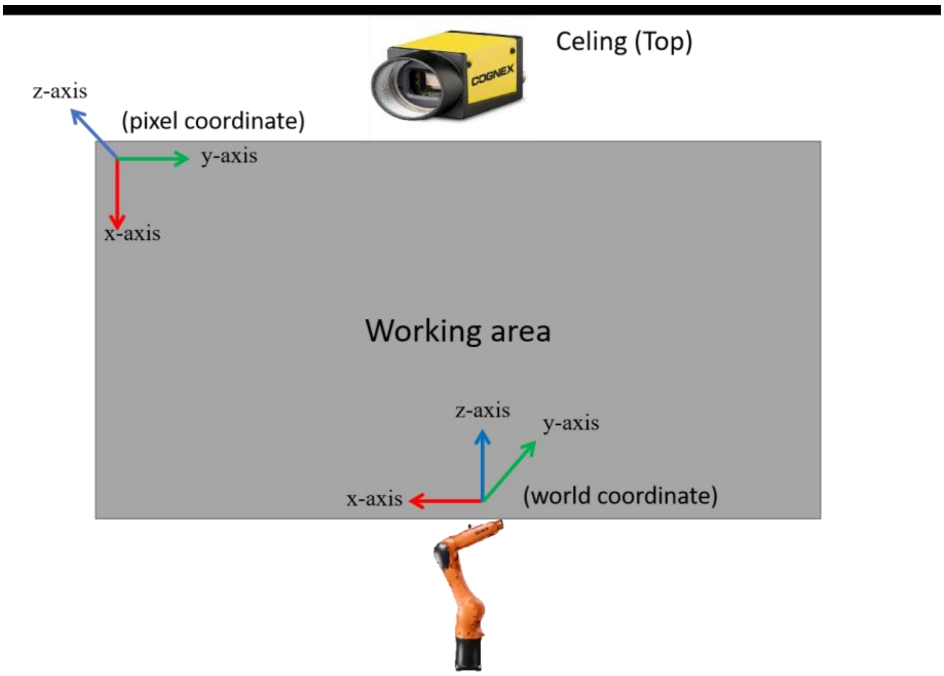

Figure 11 The Cognex and KUKA coordinate orientation

In the practical experiment, we decide to separate into two sections to complete the conversion.

1. X-Axis

From the above coordinates' schematic, we measure the total length's size of the camera scene. We use the maximal x value to minus the x coordinate of three blocks, respectively. After acquiring remain x value of the pixel coordinate, those x values need to multiply with a scaling factor, which can scale the pixel coordinate value to the world coordinate. However, as the Figure 12 below, because the KUKA robot is located approximately 360 millimetres away from the working area, thus the converted x coordinate value needs to add to the distance from the KUKA base to the working area.

The transformation equation is shown as below:

$$(2045 - \text{pixel\_x}) \times a + 360\text{mm} = \text{world\_x}$$

The scaling factor a : 0.2473

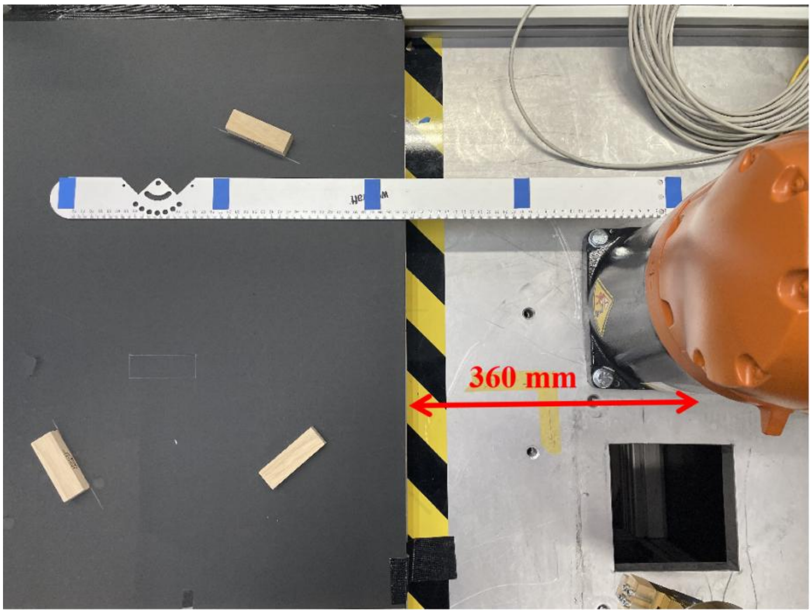

Figure 12 The distance 360 millimetre away from KUKA base

2. Y-Axis

According to the Figure 13 below, the relative position of Kuka is in the middle of the workspace and this may affect the difference in the situation of the Y-axis that the left direction is positive but the right direction is negative. Therefore, we need to measure the middle line value of the camera scene's width minus the Y-axis value of the three blobs' positions. After getting those positions y-coordinate, one thing that needs to be noticed is that the scaling factor of the Y-axis should have a negative sign for the right.

$$(1287 - \text{pixel\_y}) \times b = \text{world\_y}$$

The scaling factor b for left : 0.2513

The scaling factor b for right : -0.2513

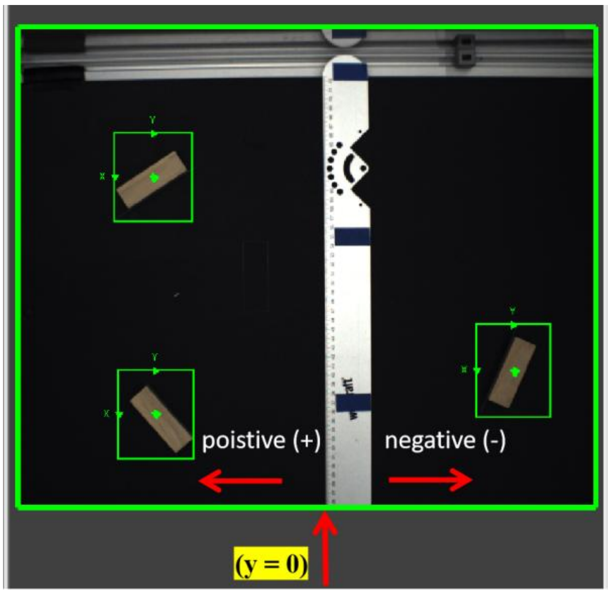

Figure 13 The schematic of actual position under camera

- **Problem analysis**

After transferring the blocks' pixel coordinate to the world coordinates, those fixed coordinate does not match with the conversion result that have some slight errors on both axis. The x coordinate in the second position of the block has a much worse biased result compared to the others two coordinates. However, the y coordinate of the first position yet has an unexpected value.

- **Realization analysis**

When the real-world distance between KUKA and the working area is measured using a ruler, it is inevitable to have a biased value of the measuring result. Therefore, optimisation and calibration always need to be implemented frequently. To solve these issues, we can either redraw the marked position on the working area or recalculate a new scaling factor for the coordinates conversion.

1.2.3 Move blocks from Cognex coordinate data

- **Task description**

After acquiring accurate coordinate of the blocks from the recognition from the Cognex camera, we expect KUKA robot now can easily pick up blocks from the three fixed positions. Hence, the triggering image setting of Cognex camera will be set up as continuous mode, which mean the camera will be updated the image continuously.

Due to the condition that we gave to restrict the KUKA movement, if the first position block is detected and it is always priority. If the first block is not detected, and the second block will be considered as picking target.

And so, once the KUKA has received a coordinate data from camera, and it will move to the position where the coordinate data is and raise down the height to clamp a block. After completing an entire pick and place action, the KUKA arm will continuously move to the next target position.

- **Problem analysis**

One problem we've run across is that the camera constantly updating the view of the working area and the robot arm is exposed under the view while the KUKA moves around to pick up a target. This might result in the target block being obscured when KUKA is at an angle. The prohibition on robot motions is also no longer valid in this case.

- **Realization analysis**

It is necessary to alter the picture mode from continuous trigger to manual trigger. To save a current image and select which target should be applied first, the operator must click the trigger's button each time. However, this is the temporary solution for current project. In the future, we will need an external components as a trigger button to capture an current image for running.

1.2.4 Detecting and converting blocks' angle

- **Task description**

The Cognex camera should be able to discern each block's angle and convert it to an appropriate angle that the KUKA robot can accept because the preceding solution had the blocks' angles set at a predetermined value, which is in line with our testing.

We require a scaling factor for the converting block angle, same as for the x and y axes. In our method, we spin the orientation of the blocks to observe and record variations in angle.

- **Problem analysis**

However, there was an unanticipated issue with the Cognex side's angle variation having unpredictable value changes. The block orientation displays a value that is irrational. It is not possible to find a simple and reasonable solution for the conversion from camera to KUKA.

- **Realization analysis**

We discovered after doing a few tests that the orientation values of different blob tools can interact. We, therefore, tried using a different blobs technique that can recognize many objects at once. It leads to the logical conclusion that spinning blobs in various orientations have unambiguous differentiation values.

1.2.5 Re-calibrating all block orientations

- **Task description**

Because the robot gripper cannot precisely clamp up blocks, the orientation of the blocks must be revised. A new measurement will be implemented, for example, by distributing numerous blocks across the entire area with varying orientations.

These calibration measurements are depicted as Figure 14 below.

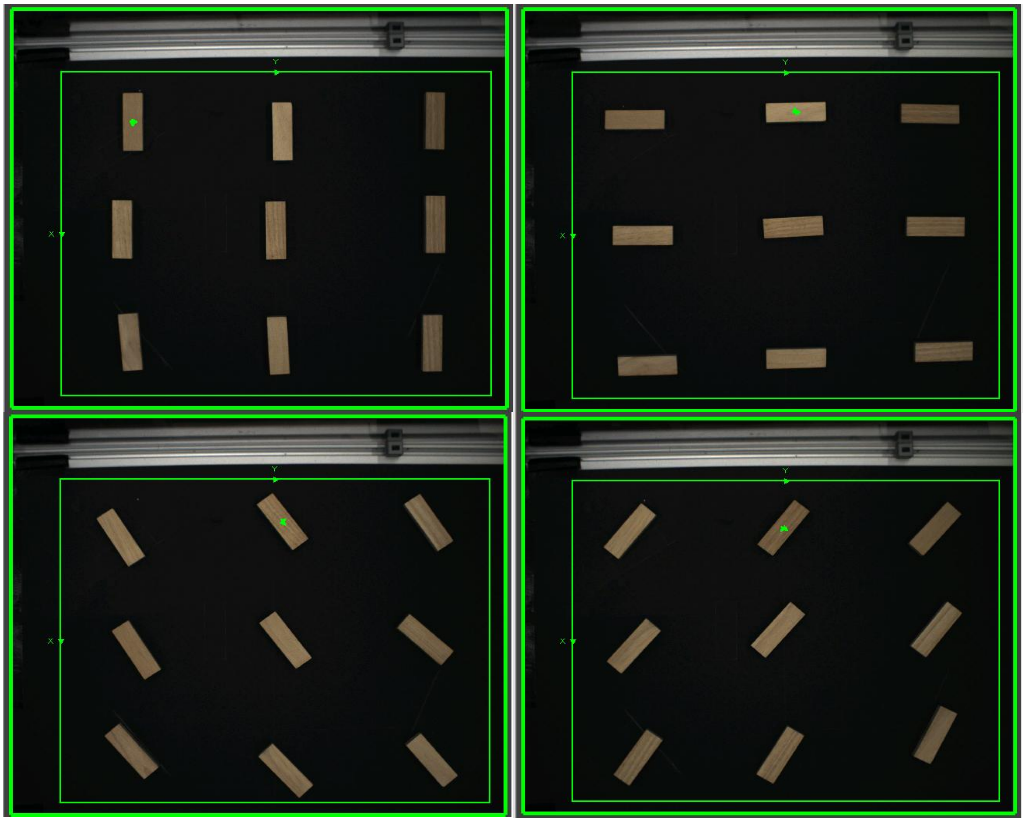

Figure 14 Four different measurement of re-calibrating blocks

1.2.6 Creating an alignment tool model

- **Task description**

Despite the improved coordinate measurement allowing the gripper to clamp up a block as precisely as possible, there is still a small amount of deviation that persists, preventing the gripper from constructing the tower without incident. Therefore, we decide to use FreeCAD software to build up a model took to align every blocking in order to align every clamping procedure.

- **Problem analysis**

When we are creating an aligned tool, and considering how to mount this aligning model on the framework of the cage which protects the KUKA robot, there are two orientations being considered.

Here are two different directions, the long side of the blocks placement area along with the cage's framework as Figure 15 and a 90-degree placement orientation with the cage's structure formed as a cross sign as Figure 16 .

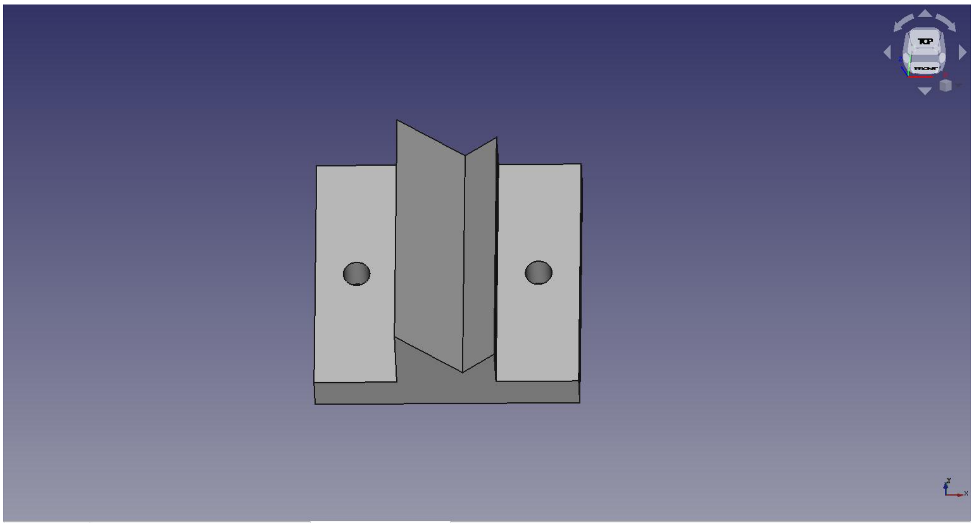

Figure 15 A cross-aligning orientation with the structure of the cage

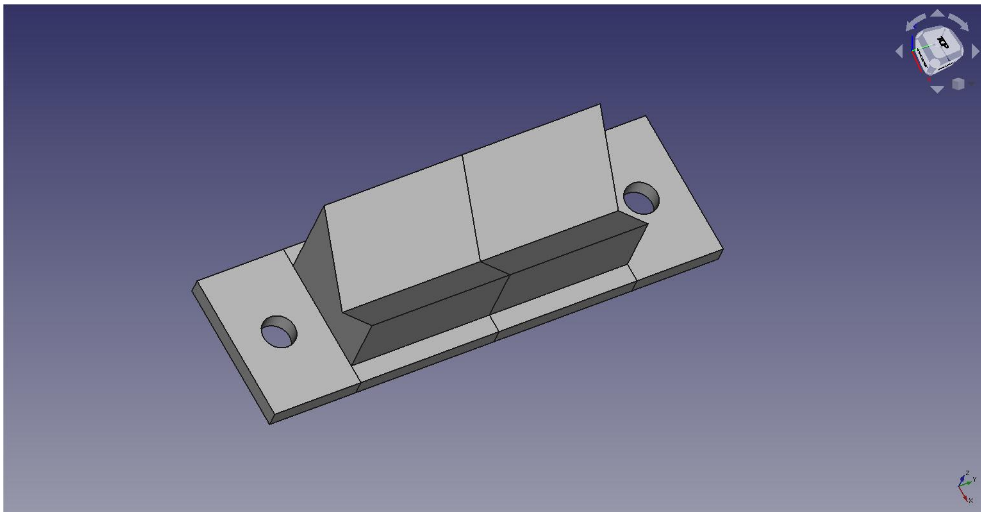

Figure 16 A same alignment with the cage's structure

● Realisation analysis

To reduce material consumption and the stability of the installation, we would like to construct the aligning tool in the same orientation as the cage framework as you can seen the Figure 17 below.

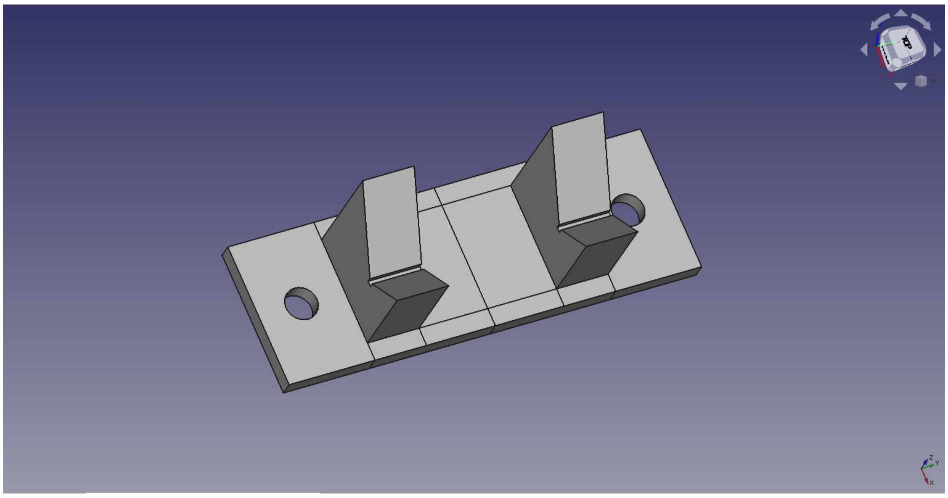

Figure 17 A final version - alignment tool

In Figure 18, it depicts the true model of the alignment tool, which includes a block that represents how the block is aligned.

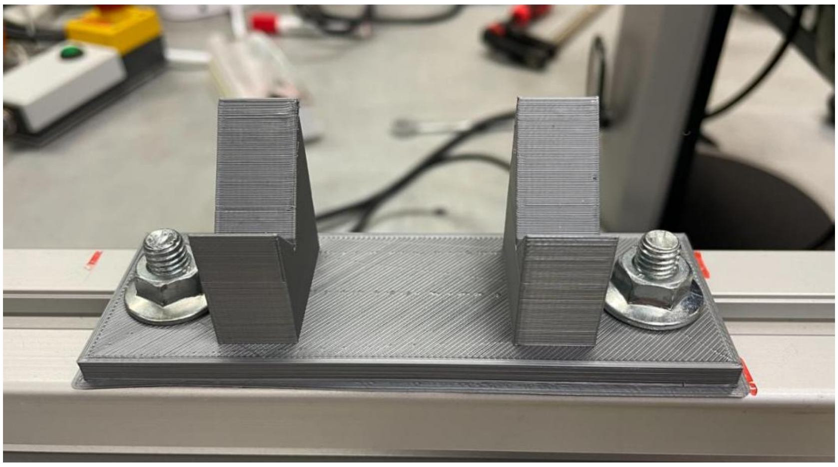

Figure 18 The real model of alignment tool

Once the actual alignment tool is completed and mounted on the protection cage, we can add a command in which the robot arm can clamp the block from the working area to the alignment tool and place it on this alignment tool in order to clamp the block correctly. The schematic of this movement can be seen in Figure 19 below.

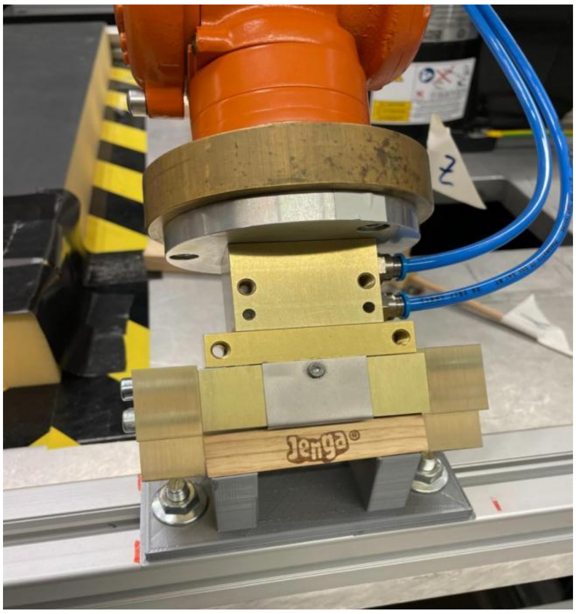

Figure 19 Gripper moves block to the alignment tool

1.2.7 Configure collision detect and boundary stop

- **Task description**

To enable the whole program to run automatically in a thoroughly safe condition, we need to set up some configuration of the KUKA robot's movement. When the robot gripper moves to some unexpected positions which may cause damage, the robot program should able to stop the KUKA movement immediately. In those unexpected positions, we assume them as the outsider of the cage.

- **Problem analysis**

The gripper length is not taken into account in the preceding set-up in the robot's software. As a result, the robot arm is unable to reach the needed position, which may be far higher than our expectations. This may cause the robot movement becomes out of control.

- **Realisation analysis**

The resolution of this problem is remeasuring the size of the gripper and input into the controller machine. And also, some of the fixed values of the coordinates are needed to add on the new value which is the size of the gripper, for example, the coordinate of the tower area, and the coordinate of the alignment tool.