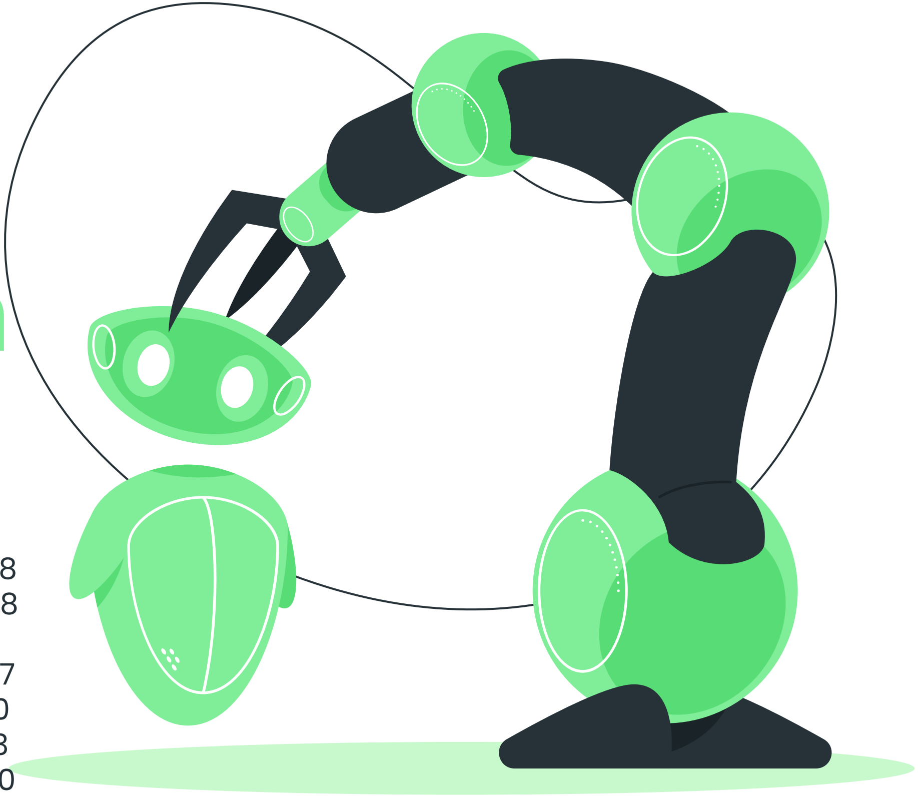
Quality Inspection Cell:

Burrs detection



Mechatronic Design MR3009

P4.

José Angel Soto Hernández A01039978
Nathalie Vilchis Lagunes A01364838
Hector Everardo Martínez Cisneros A01282300
Teclo Moreno Rodriguez A01252067
Estefany Morales Valdes A01281880
Diego A. Santisteban Pozas A01154423
Jose Antonio Arrambide Garza A00817790

24/11/2021

1
Original Problem/opportunity definition

Opportunity definition

Original Problem

During the drilling in aluminum process the presence of burrs is likely to happen.

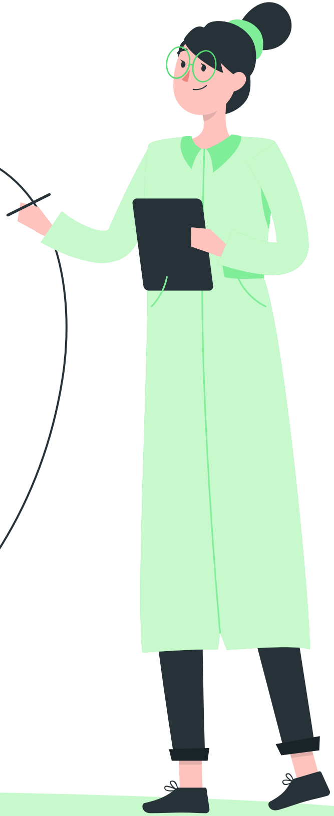
Burrs might represent a high risk in the subsequent process of the product

Suggested solution

Automate the inspection of drilled aluminum profiles. Integrate machine learning, computer vision and cobot usage to sort profiles (accepted and rejected parts). Eliminate human interaction.

2

Value proposition conclusion



Original Value Proposition

Quality Inspection Cell: Burrs detection

Increase

- Adaptability
- Detection speed
- Accuracy
- Higher quality

Create

- User-friendly interface
- Database
- Alarms
- Machine learning

Reduce

- Price
- Workspace
- Hardware/Software requirements

Eliminate

- Overhead
- Human error

Value Proposition Conclusion

Achieved

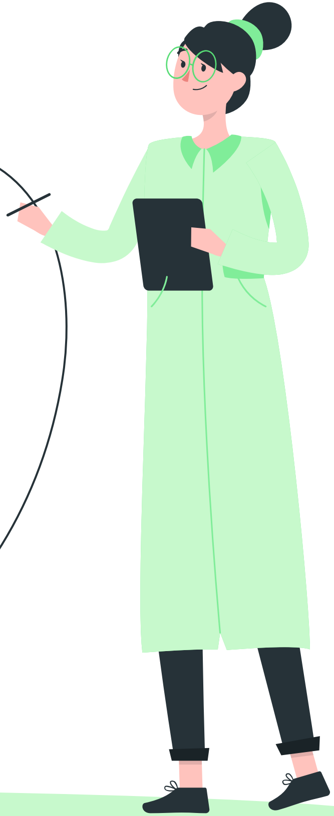
- High detection accuracy
- Higher quality products,
- High adaptability
- Machine Learning
- Result Database

NOT Achieved

- Eliminate all human interaction
- Real-time results
- User friendly interface
- Reduction of workspace
- Reduction of hardware and software requirements

3

Final Product Requirements



Final Product Requirements

Achieved

- Non invasive operation
- Cobot-Python-Orange communication
- Burr classification (92.2 %)
- Creating knowledge (databases)
- Machine learning model (Orange)

Partially Achieved

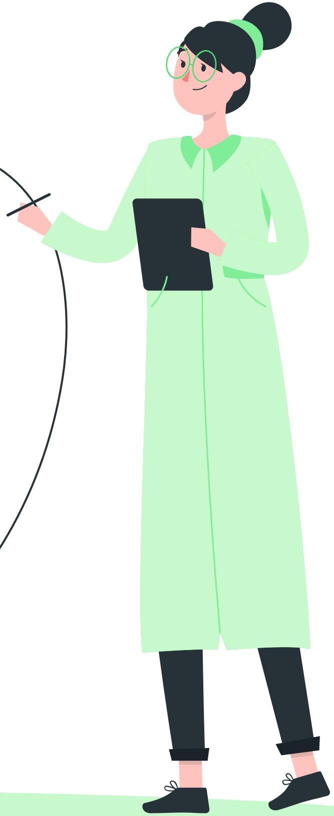
- Reduce workstation accidents
- Working station integrated in single working table (Simulated)
- Reduce workforce (simulated - 1 person)

NOT Achieved

- Near zero human interaction
- Response time less than 5 seconds
- Avoid product damage
- Avoid infrastructure damage
- Real time CV

4

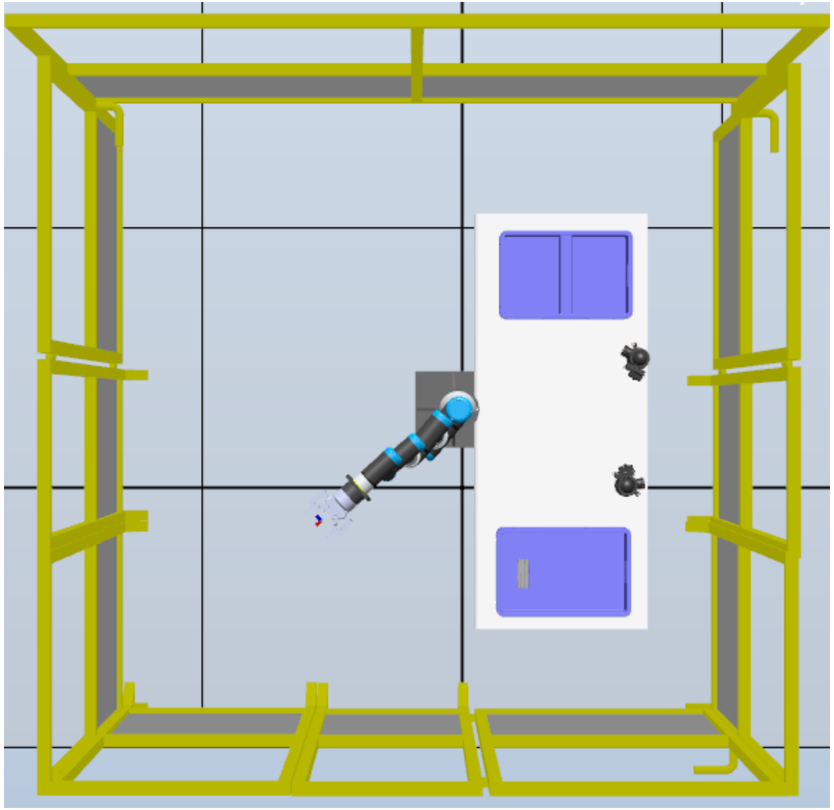
Final Design Details

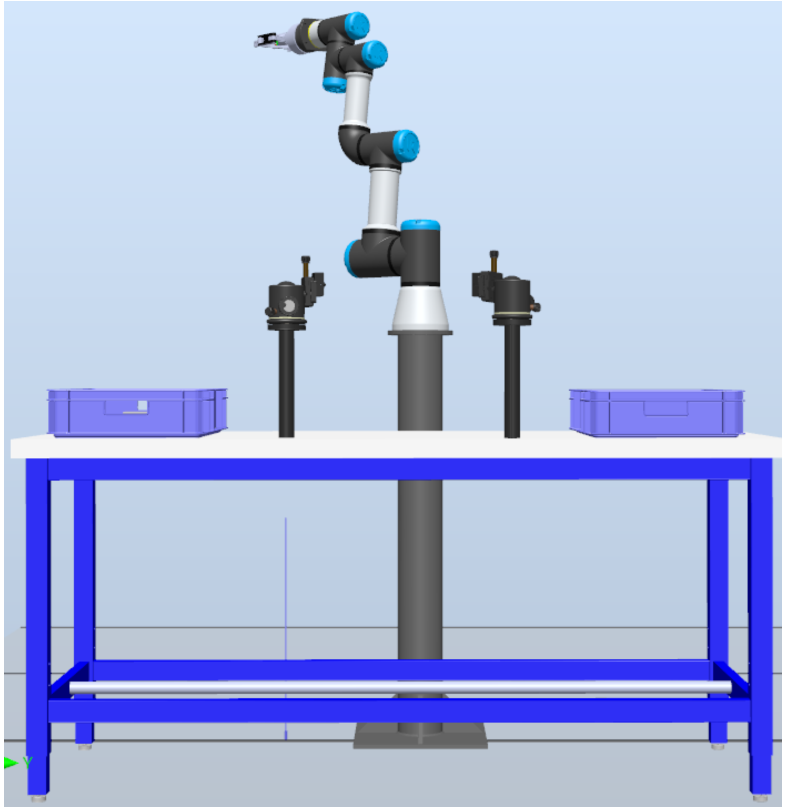


Final Design details

- Final model is fully simulated in RobotStudio
  - UR3 CAD model imported to RobotStudio
  - Custom controller designed for six axis robot with an ABB 140 driver and Robot Flex 125 toploader mechanical template.
- Communication done through python
  - Robotstudio -> Python -> Orange -> Python -> Robotstudio
- Classification and predictions done in orange

Cobot station





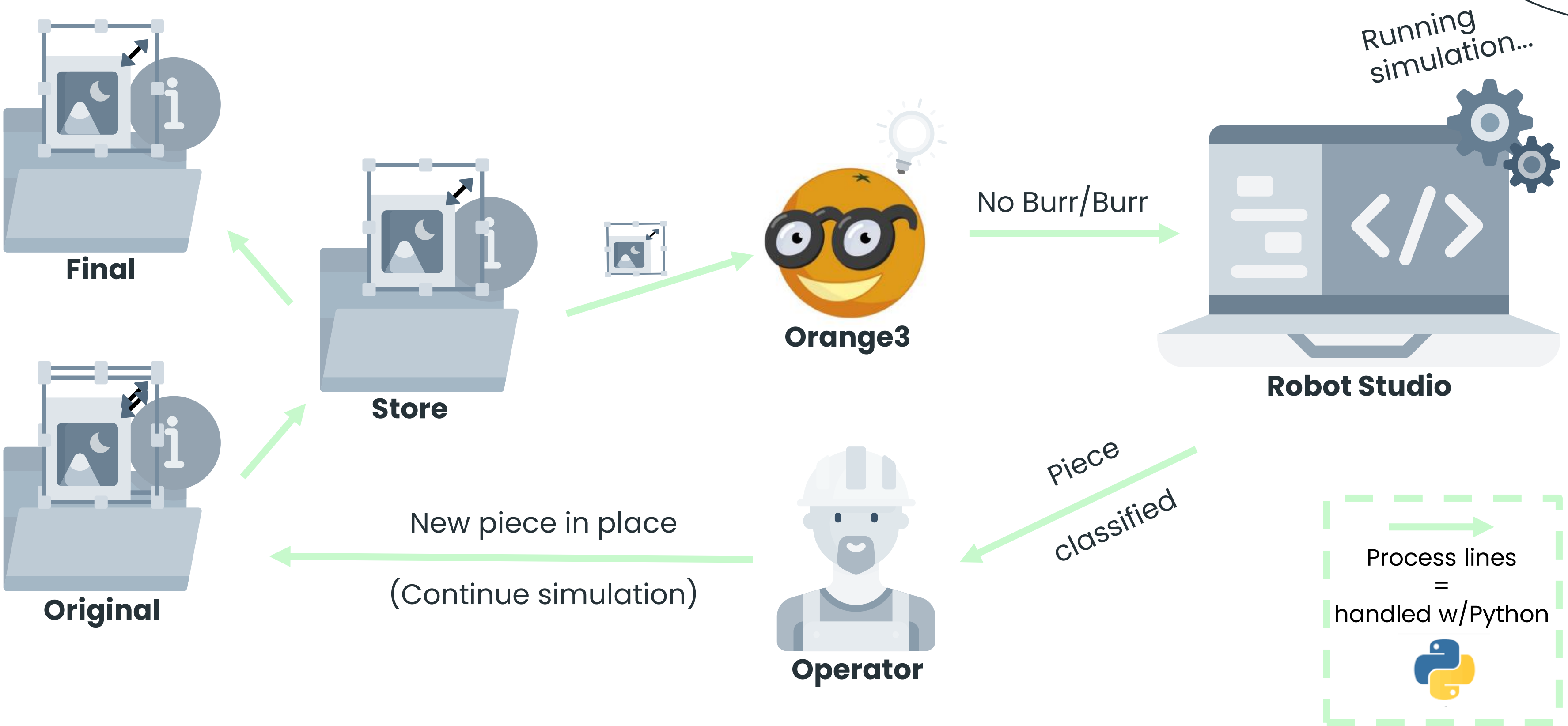
5

Verification results

Verification results

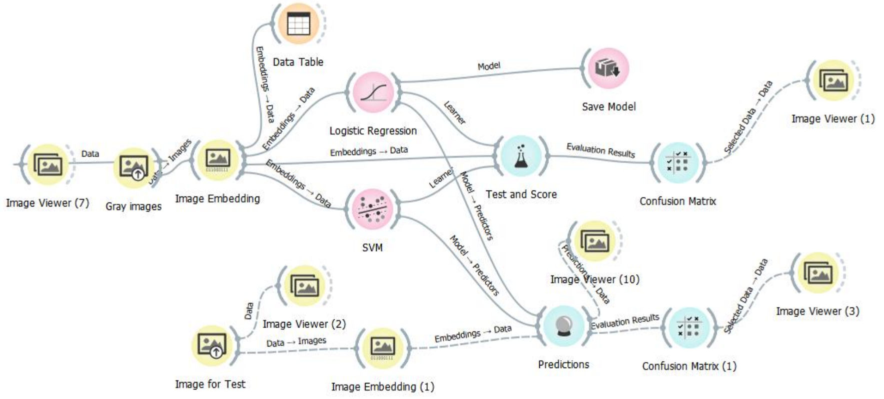
1. **Testing plan**
    1. Classification
        - We ran several tests with assorted images with burrs and no burrs
        - Each image went from Original folder -> Stored folder -> Final folder
        - Orange reads image from stored folder
        - Prediction from image is sent from python to Robotstudio, cleans stored folder places the used image in final folder (to prevent repetition)
        - UR3 takes a pre programmed path according to the classification results
        - UR3 resets to home position and python communication is closed
        - Test concludes when the simulation places the profile in the correct bin and the program resets.

Prototype Details: Process Loop

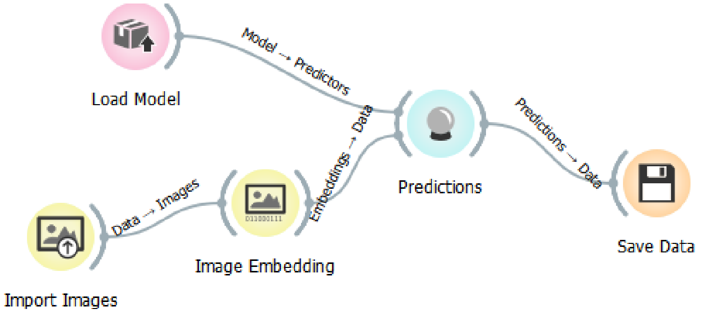


Simplified Orange program

Original



Simplified



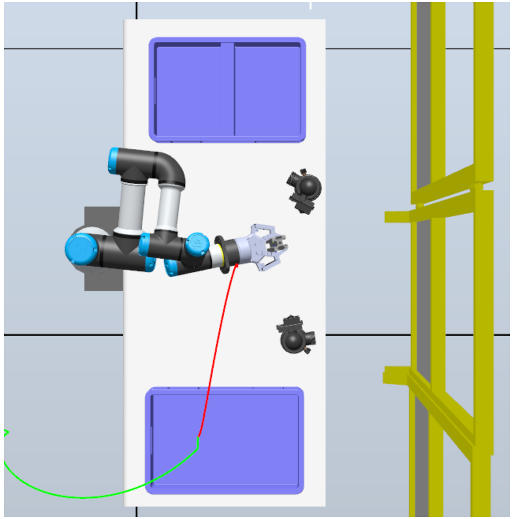
- Logical regression training model was selected (Better results than SVM and random Forest) AC > 92%
- Grayscale database was used for training

- Logic regression model is loaded into new orange file
- Picture saved by python in Store folder is loaded for classification
- Much faster orange results
- No need to retrain every new run

Python – Robotstudio connection and initiation of python script

1. Simulation starts in RobotStudio
2. RobotStudio creates server host; waits for python connection confirmation.
3. Cobot takes path according to classification results

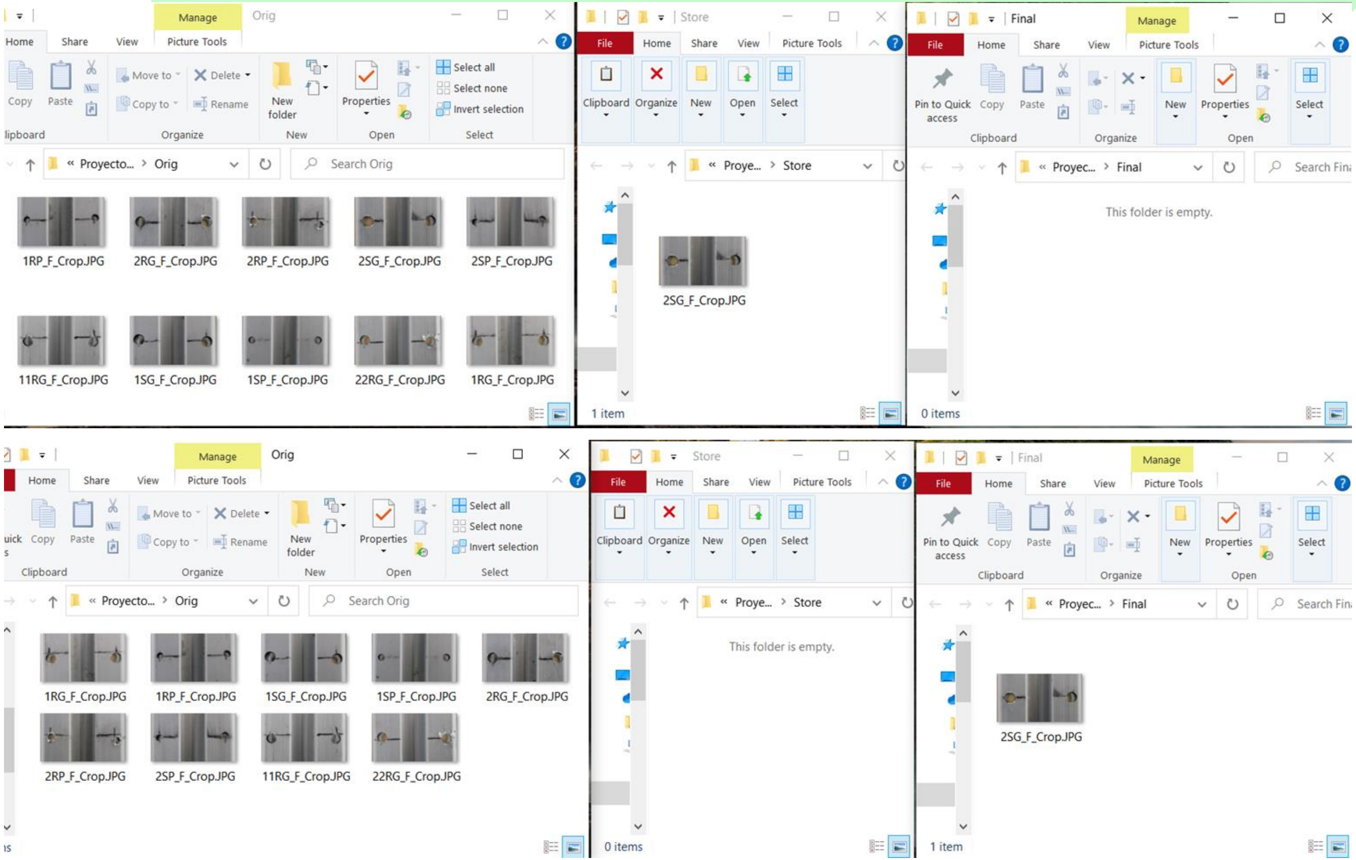
1. Once confirmed, python runs image filter script
2. After Orange finishes image analysis, user needs to press any key to continue
3. Classification results will be stored in a database
4. Python reads and sends results to Robotstudio



```
C:\Users\JAAG\AppData\Local\Programs\Python\Python39\python.
b'Conexion correcta'
C:/Users/JAAG/Desktop/ProyectoCobot/Orig/2RG_F_Crop.JPG
C:/Users/JAAG/Desktop/ProyectoCobot/Store/2RG_F_Crop.JPG
Press any key to continue . . .

C:/Users/JAAG/Desktop/ProyectoCobot/Final/2RG_F_Crop.JPG
Con rebaba
1
```

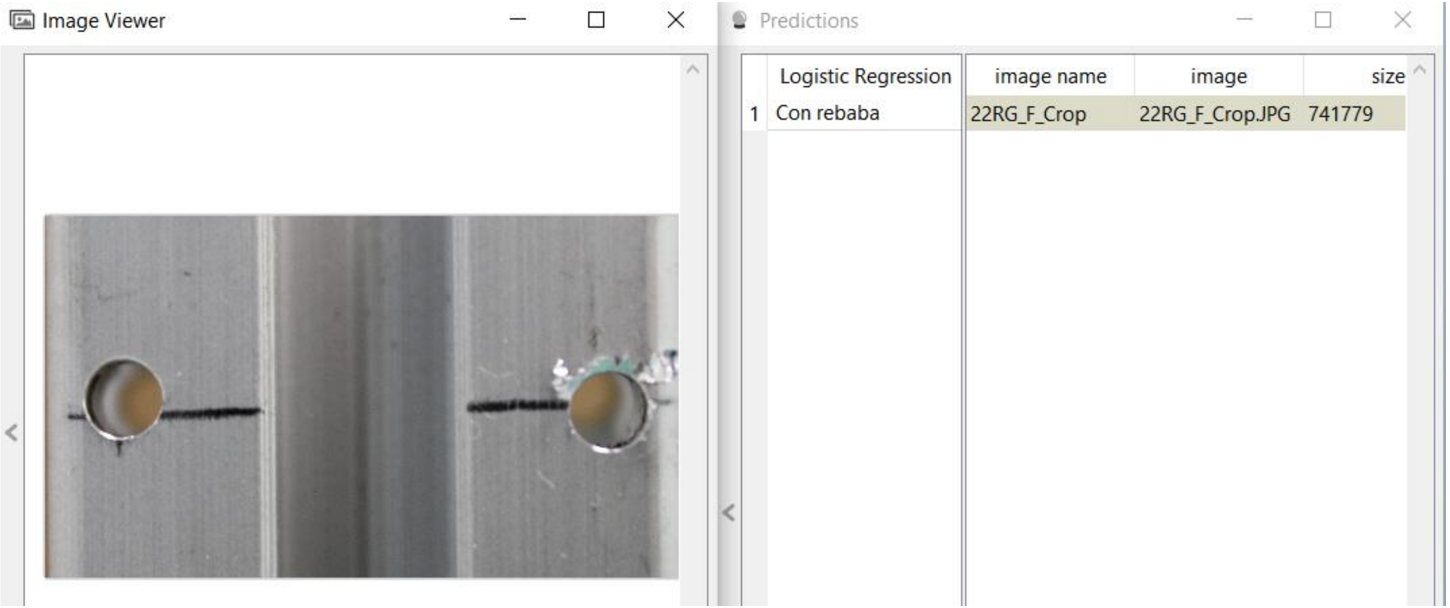
Python image manipulation



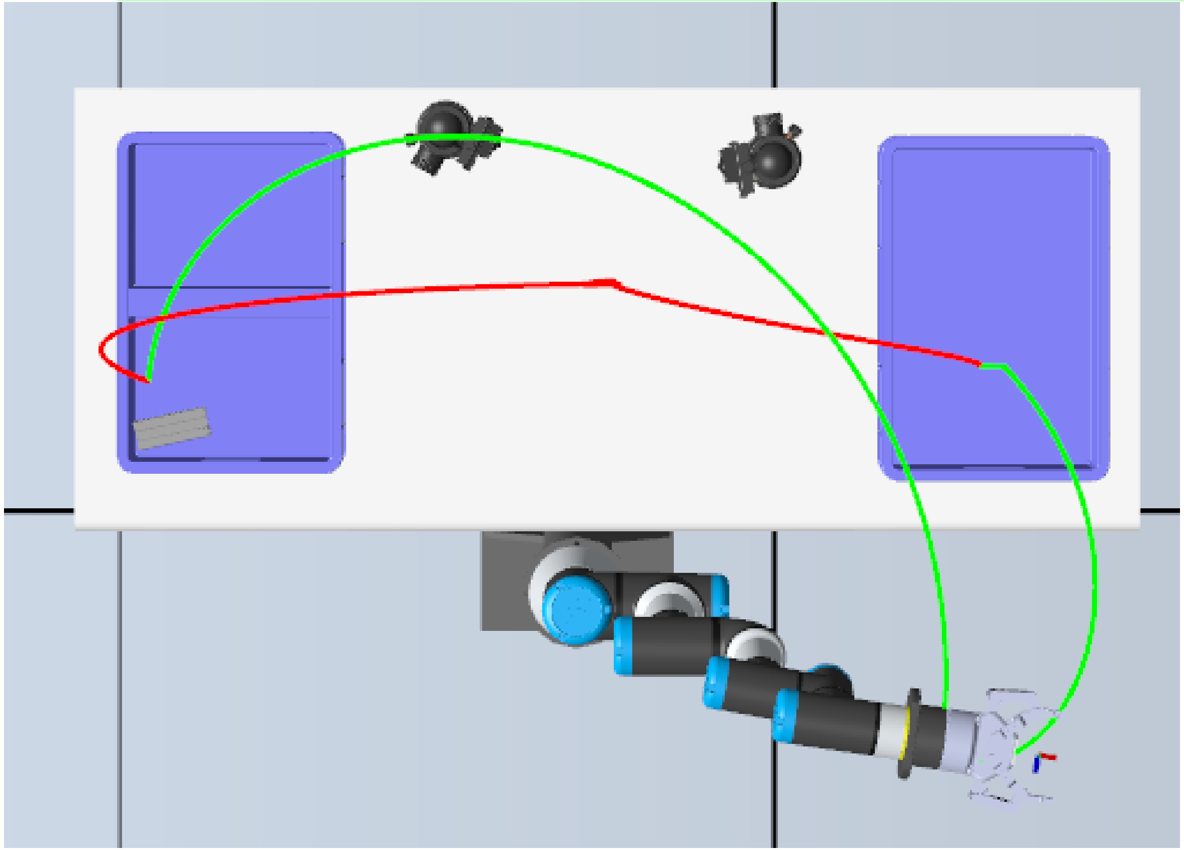
1. Orig folder holds pictures to analyze
2. Python extracts a random picture from Orig and moves it to Store folder
3. Orange reads picture from Store folder
4. Python moves picture from store folder to Final folder *This is to prevent image repetition

Orange single image results

- ❏ As it can be seen Logistic Regression model gives a correct prediction
- ❏ The prediction result is stored in a database and read by python
- ❏ RobotStudio receives prediction result from python



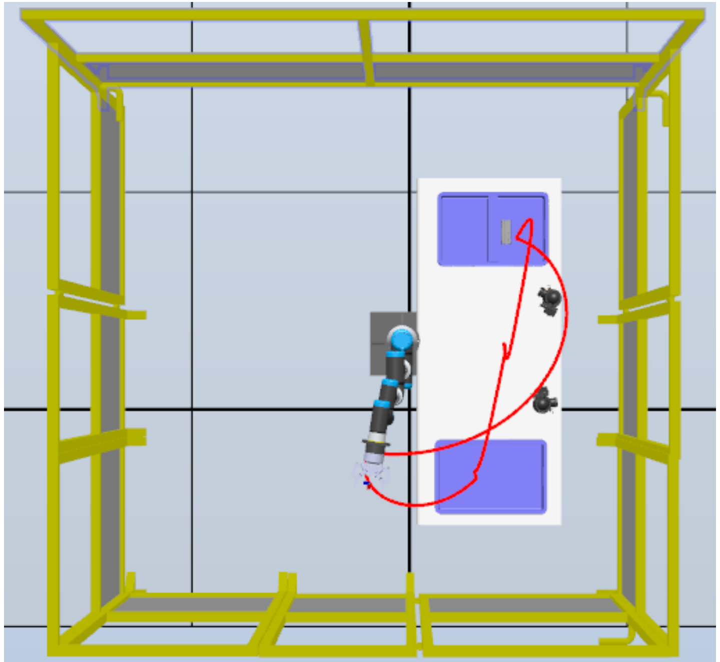
Cobot path for image with burrs



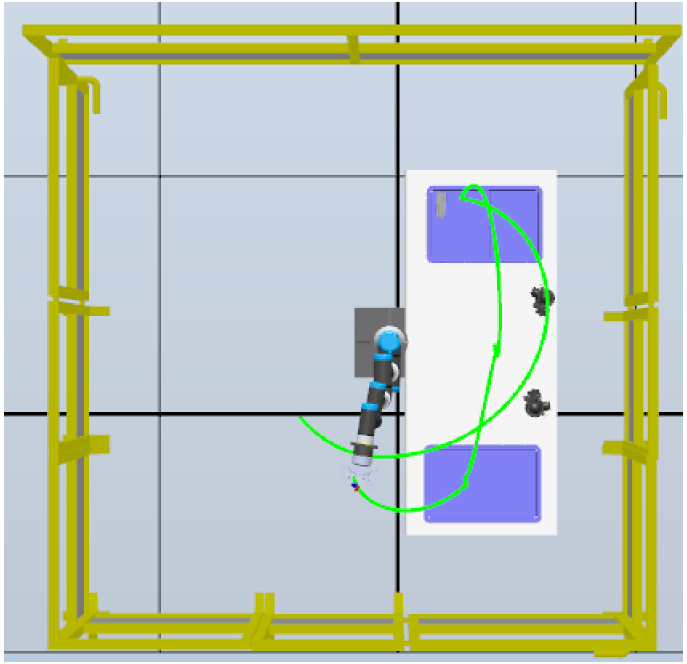
- Cobot wont move from classification area (between cameras) until it receives Orange’s result from python
- Red path means the cobot is holding the profile
- After leaving profile the Cobot returns to its original location.

Pre programed paths

Path with no burrs



Path with burrs



Video test results

Test results

- The model of classification of images with or without burrs in orange is of **92.9%**

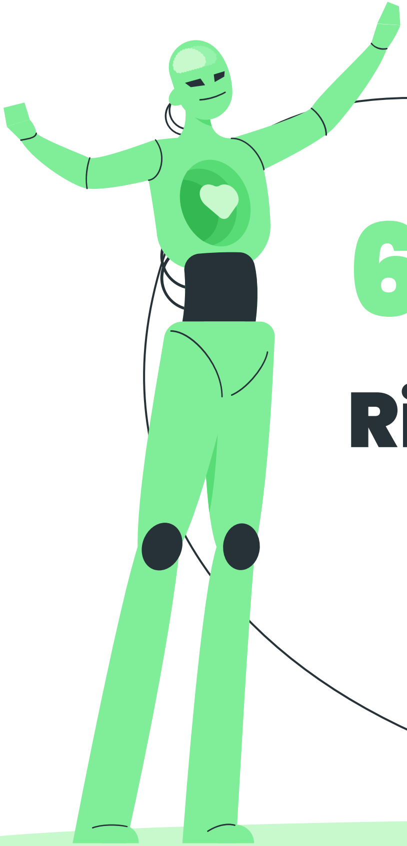
- Times of simulation experiment results:

Steps/Velocity	Slow	Medium	Maximum
Pick up and position the piece	14s	11s	9s
Python y Orange	35s	23s	23s
Position the piece in the bins	14s	10s	8s
Total Time	63 s	44 s	40 s

Orange test results

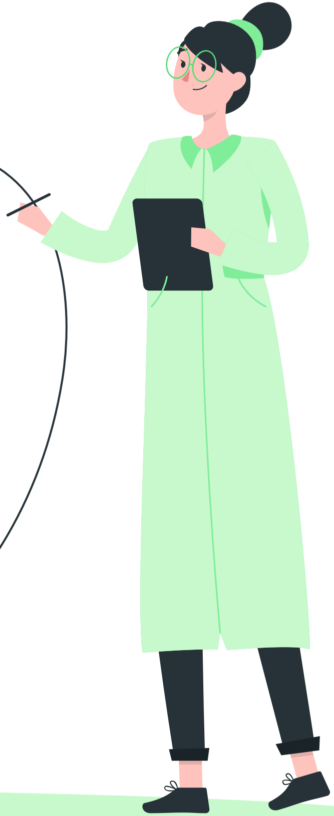
image name	image	Logistic Regression	Logistic Regression (Con rebaba)
string	string	Con\ rebaba Sin\ rebaba	continuous
meta	meta origin=C:/Users/JAAG/Desktop/ProyectoCobot/Store type=image	meta	meta
11RG_F_Crop	11RG_F_Crop.JPG	Sin rebaba	0.318308626
1RG_F_Crop	1RG_F_Crop.JPG	Con rebaba	0.879055988
1RP_F_Crop	1RP_F_Crop.JPG	Con rebaba	0.529156108
1SG_F_Crop	1SG_F_Crop.JPG	Sin rebaba	0.001326119
1SP_F_Crop	1SP_F_Crop.JPG	Sin rebaba	2.02E-08
22RG_F_Crop	22RG_F_Crop.JPG	Con rebaba	0.99999972
2RG_F_Crop	2RG_F_Crop.JPG	Con rebaba	0.999999126
2RP_F_Crop	2RP_F_Crop.JPG	Con rebaba	0.999826744
2SG_F_Crop	2SG_F_Crop.JPG	Sin rebaba	0.024250907
2SP_F_Crop	2SP_F_Crop.JPG	Sin rebaba	0.016766824

*Testing carried out with the original images flipped 180°



6

Risk Assessment



Risk Assessment

<table>
<tr><td rowspan="8">Major Risks</td><td>Reliability/ Performance</td><td>Fast Process</td></tr>
<tr><td>R1: Wrong classification of the aluminium profiles</td><td>FP1:Takes more than 5 seconds to classify</td></tr>
<tr><td>R2: Software failure</td><td>FP2: Process stops due to bad connectivity</td></tr>
<tr><td>R3: Requirement of multiple software programs management</td><td>FP3: Maintenance and upgrades take too much time</td></tr>
<tr><td>Costs</td><td>Safety</td></tr>
<tr><td>C1: Operational cost are bigger than revenues</td><td>S1: Possible accident that injures the user</td></tr>
<tr><td>C2: Initial investment is very high</td><td>S2: Possible accident that damages the Cobot</td></tr>
<tr><td>C3: Maintenance of system expenses</td><td>S3: Possible accident that damages the aluminium profile</td></tr>
</table>

Risk Assessment

			Business Impact				
			Extreme	Major	Moderate	Minor	Insignificant
			100%	80%	62%	25%	1%
Occurrency	(Almost) Certain	100 %		FP1			
	Probable	80%					
	Possible	62%	C1	FP2			
	Unlikely	25%	S1	R2, C2,S2	R1	S3	
	Rare	1%		R3,FP3	C3		

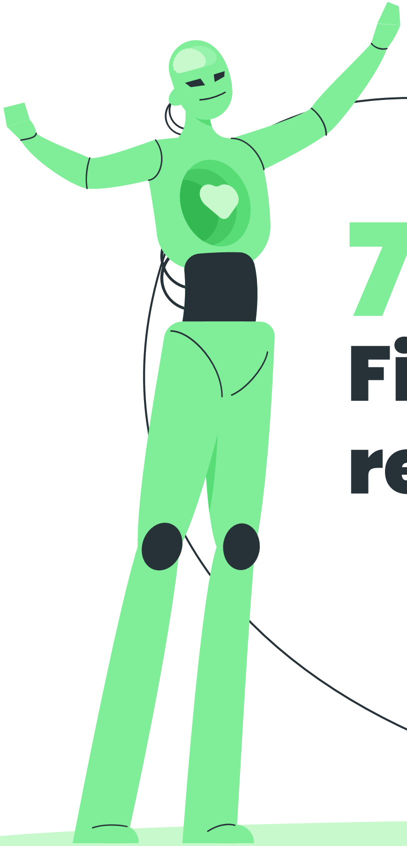
Tolerate	to Improve	Must be corrected

Mitigation Plan

	Reliability/ Performance	Risk	Actions to eliminate/mitigate Risk
Major Risks	R1: Wrong classification of the aluminium profiles	M	Improve the quality of the image acquisition.
	R2: Software failure	M	Regular check-ups to verify that the softwares are in optimals conditions
	R3: Requirement of multiple software programs management	L	Run the programs in a device that supports all the softwares running at the same time without saturating the RAM.
	Costs		
	C1: Operational cost are bigger than revenues	M	Try to cut unnecessary costs and match it with the sales prices.
	C2: Initial investment is very high	M	Try to cut unnecessary costs to low initial investment.
	C3: Maintenance of system expenses	L	Make a study of how often are maintenance shall be done.

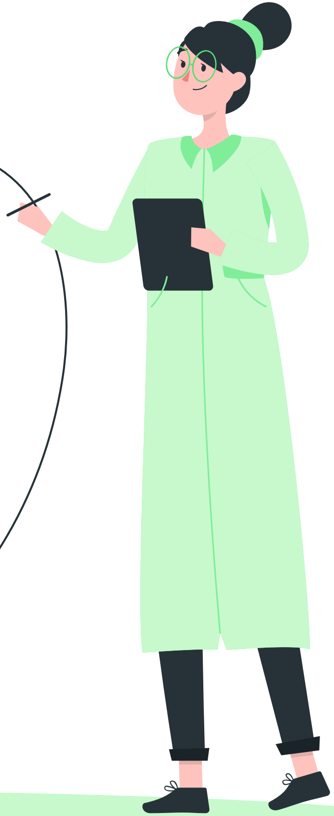
Mitigation Plan

Major Risks	Fast Process	Risk	Actions to eliminate/mitigate Risk
	FP1:Takes more than 5 seconds to classify	H	Optimize the movement path and velocity of the robot.
	FP2: Process stops due to bad connectivity	M	Implements communication protocols to avoid interference.
	FP3: Maintenance and Upgrades take too much time	L	Program maintenance when production line is not running. (at night or holidays)
	Safety		
	S1: Possible accident that injures the user	M	Redesign the working area to avoid accidents, Integrate a Stop Button
	S2: Possible accident that damages the Cobot	M	Redesign the working area to avoid accidents, Integrate a Stop Button
	S3: Possible accident that damages the aluminium profile	M	Ensure by testing that the robot is able to place the aluminium profile in a correct place



7

Final IP recommendations



Claims and IP Instruments

Copyright

Orange algorithm for the detection of burrs in aluminum profiles.

This patent protects the algorithm created to analyse compare and classify provided images for the detection of aluminum burrs. It also includes a machine learning stage to improve accuracy and precision through a previously loaded database.

Copyright

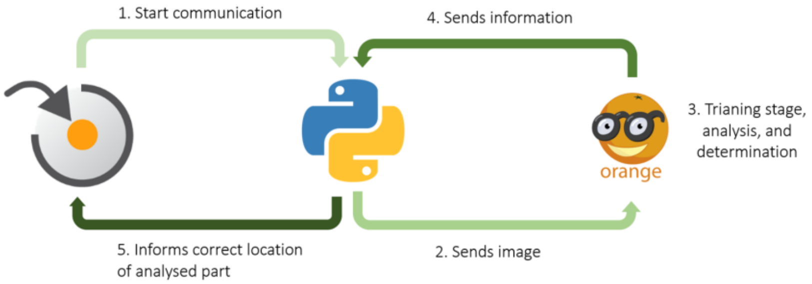
Python algorithm used as an intermediary for the synchronization between orange and robot studio.

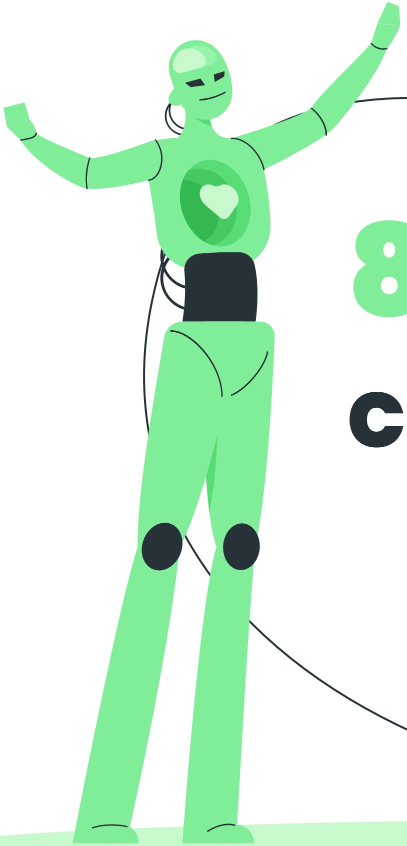
Algorithm that manipulates a robotstudio simulation with information previously processed by orange. It provides information to orange to complete the image classification process, which is sent back to python. This information is used to determine the specific placement of the aluminum profiles in robotstudio.

Patent

Computer vision device and quality inspection cell for burr detection in aluminum profiles.

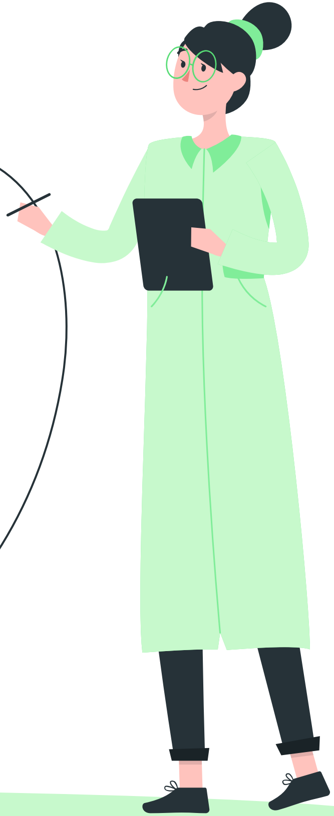
This is an inspection cell that incorporates a cobot and computer visión to fully automate the classification of metal burrs. This inspection cell takes into account the pick of the part, the inspection of the profile in an specific area and the correct distribution of the aluminum part.





8

Conclusions



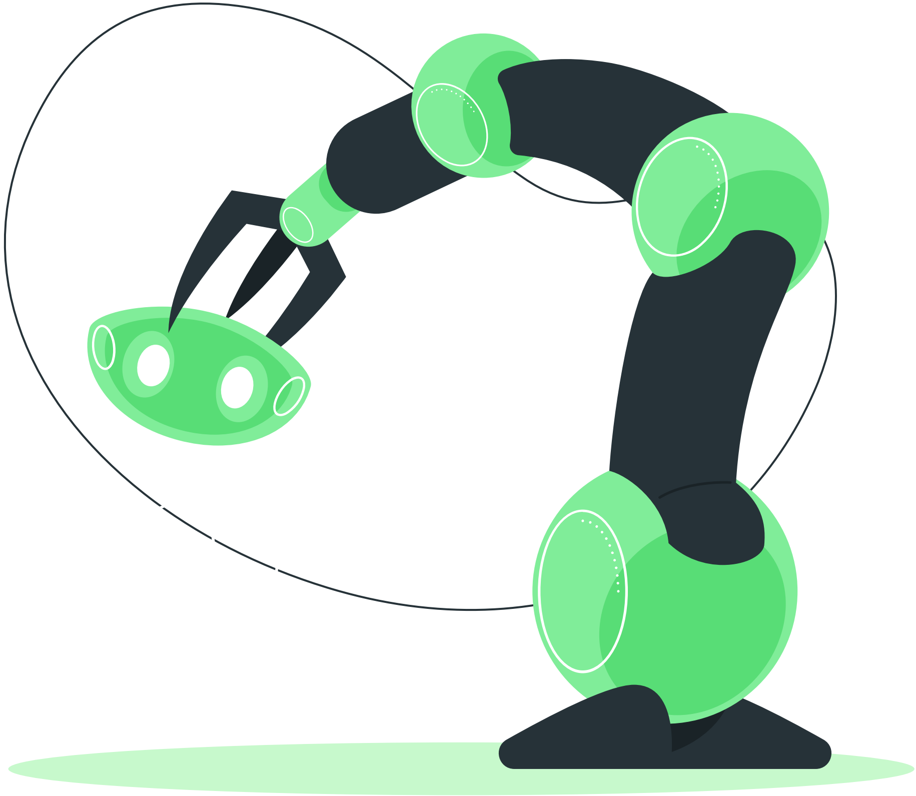
What's next

Make a physical prototype based on the simulation. (Implementation of the product).

Validation tests (On field testing for communication between softwares, robot trajectory, image capturing, etc.).

Optimization of classification model to reach more than or equal to 95% detection accuracy.

Quality Inspection Cell: Burrs detection

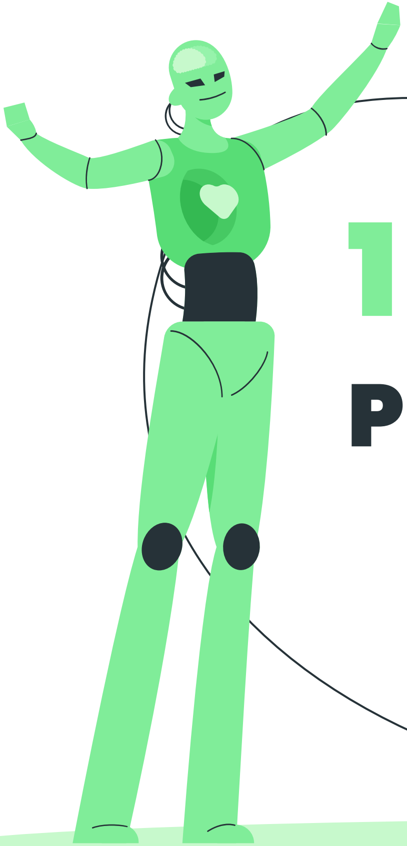


Mechatronic Design MR3009
Sergio Uribe

P4.
José Angel Soto Hernández A01039978
Nathalie Vilchis Lagunes A01364838
Hector Everardo Martínez Cisneros A01282300
Teclo Moreno Rodriguez A01252067
Estefany Morales Valdes A01281880
Diego A. Santisteban Pozas A01154423
Jose Antonio Arrambide Garza A00817790

01/09/2021

Table of Contents



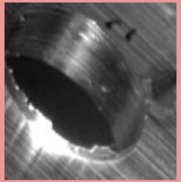
1

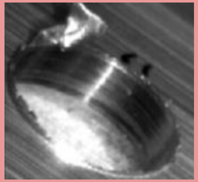
Project Definition

Project Definition

- Automate the inspection of drilled aluminium profiles.
- Discriminate Clean parts from Rejected parts
- Fuse computer vision and cobot integration.
- Machine Learning

Problem/Opportunity





Suggested Solutions

Major Risks

Technology Risks
- Lack of useful training data
- Inaccurate burrs detection

Business Risks
- Cost effectivity
- Competition

- Aluminium drilling implies material removal.
- Metal burrs may appear due to many factors.
- Tool velocity, precision, quality
- Potential risk for subsequent processes.
- Performance Failure

Main Beneficiaries

CID y T-Tec:
- QA Department
- Operator/quality Inspector

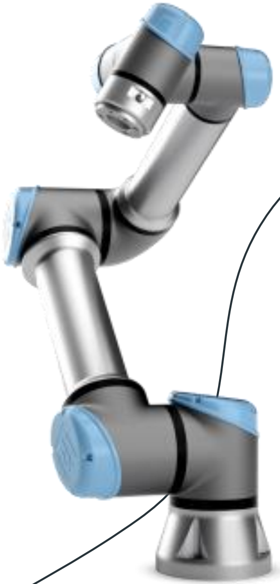
Other Customers

Project target

1. Automatic process of burrs detection with computer vision
2. Increase process efficiency
3. Less expensive by eliminating the human factor

Project Scope

→ Research
→ Quality control
→ Automation & manufacturing



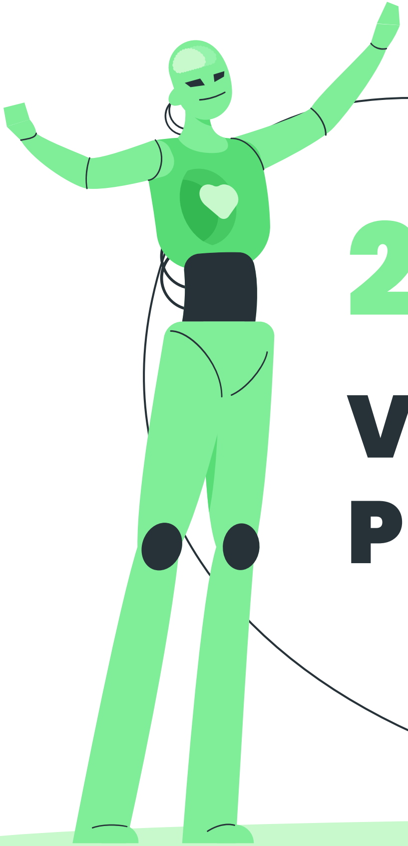
Project Deliverables

Design of the full inspection cell

Training and verification of the system

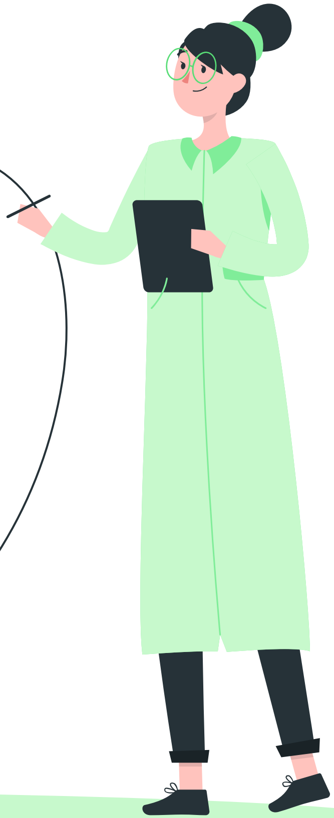
Machine Learning classifier

Program the cobot integrating the full function



2

Value Proposition



Value Proposition

Increase

- Adaptability
- Detection speed
- Accuracy
- Higher quality

Create

- User-friendly interface
- Database
- Alarms
- Machine learning

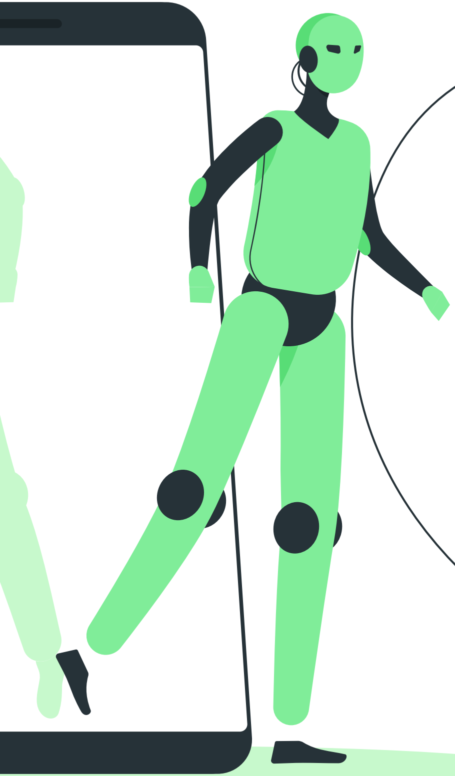
Quality Inspection Cell: Burrs detection

Reduce

- Price
- Workspace
- Hardware/Software requirements

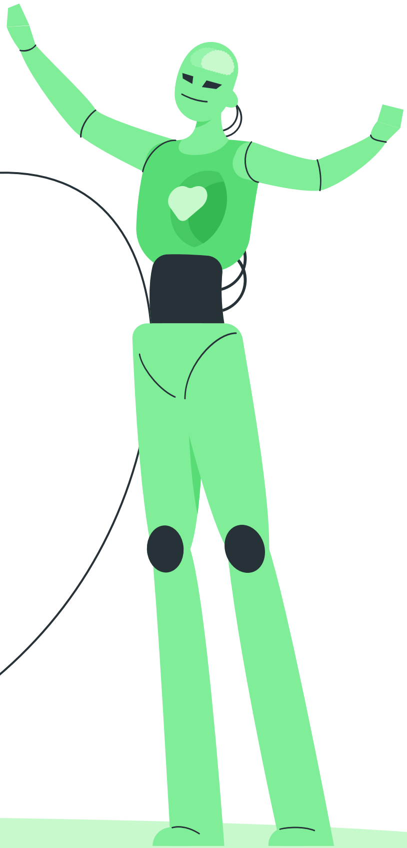
Eliminate

- Overhead
- Human error

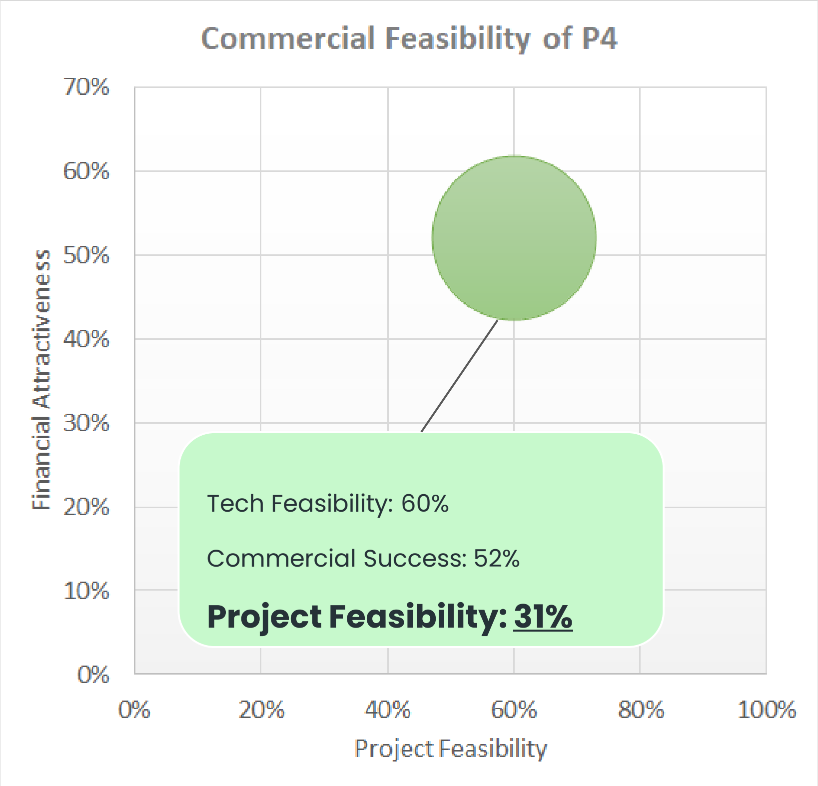


3

Business Case



Commercial Feasibility



The project is technologically challenging, but based on a **proven concept.**

A niche market is targeted, but the **growth is expected to be moderate** (<15%).

The delivered benefit rests on the **radical improvement in performance, cost, and quality.**

Added value and customer need should be highlighted and constantly improved upon to keep the strong competitors at bay.

Project Financials

Project development time:

18 weeks (4.5 months)

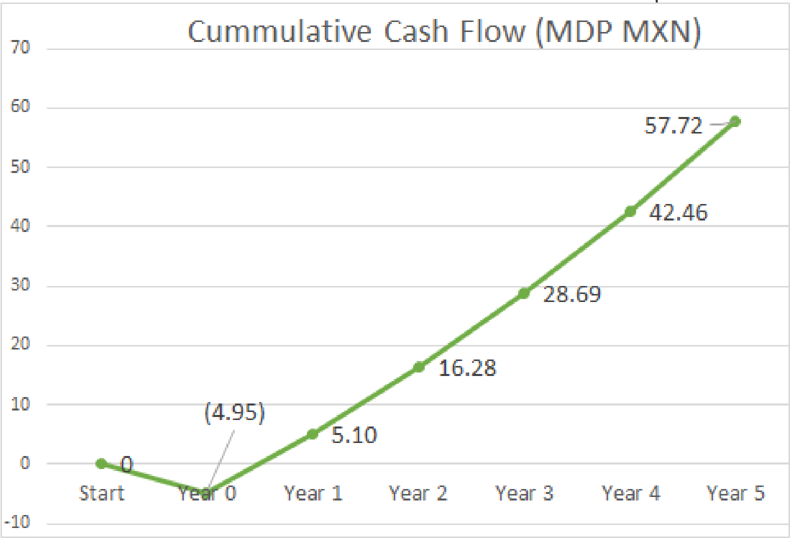
Assuming:

- **6 monthly sales**
- **30% mark-up**
- **10% market growth**

IRR: **213%**

ROI: **635%**

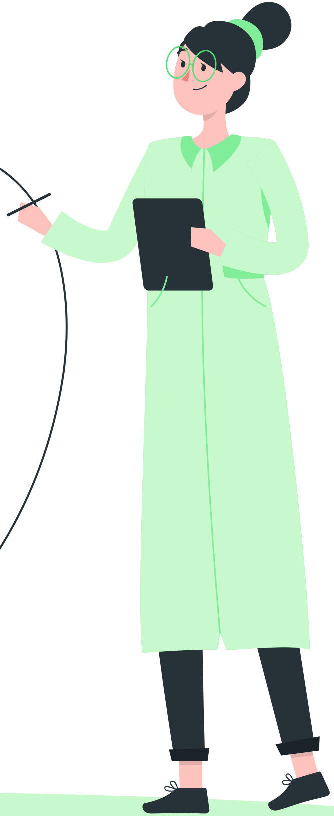
6 month payback period



Initial investment: 4,950,750 MXN

Note: projection up to year 5

4
Product Requirements Solutions



Qualifiers

Optimized layout for lean operation

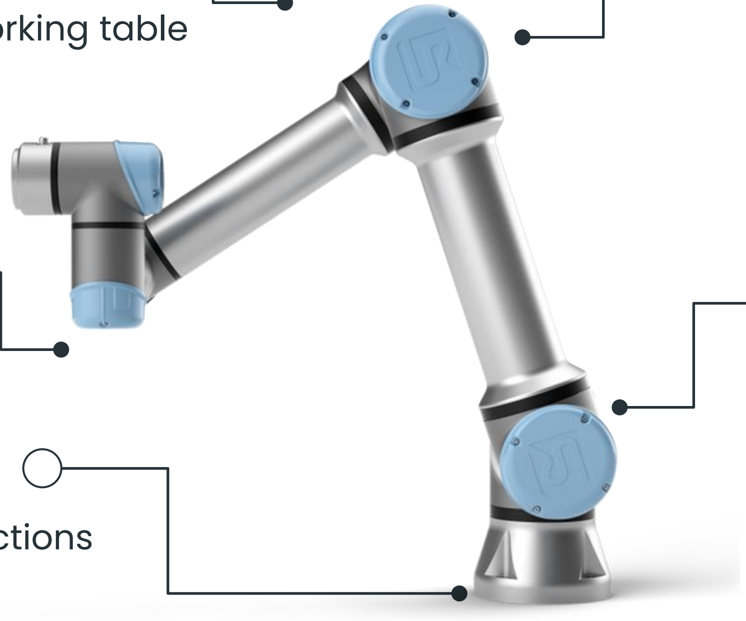
- ➔ Minimize the cycle time
- ➔ Integrated in a single working table

Reduce cost of the process

- ➔ Reduce workforce

Guarantee quality products

- ➔ Reduce mismatch inspections
- ➔ Reduce false negatives



Automatic detection of metal burrs

- ➔ High precision
- ➔ Fast response time
- ➔ Reduce human error

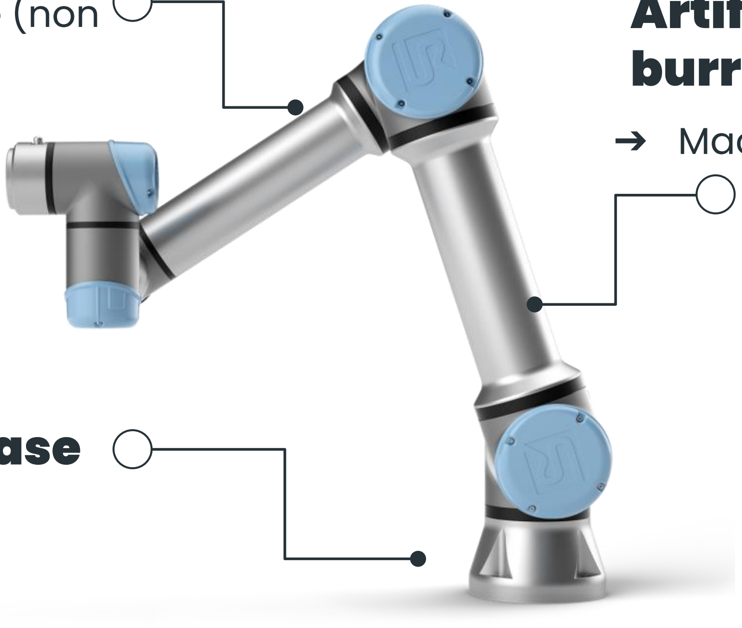
Safety Operation

- ➔ Reduce accidents
- ➔ Avoid product damages
- ➔ Avoid infrastructure to be damaged

DIFFERENTIATORS

User Friendly Workspace

- ➔ Interaction user-machine (non invasive interaction)



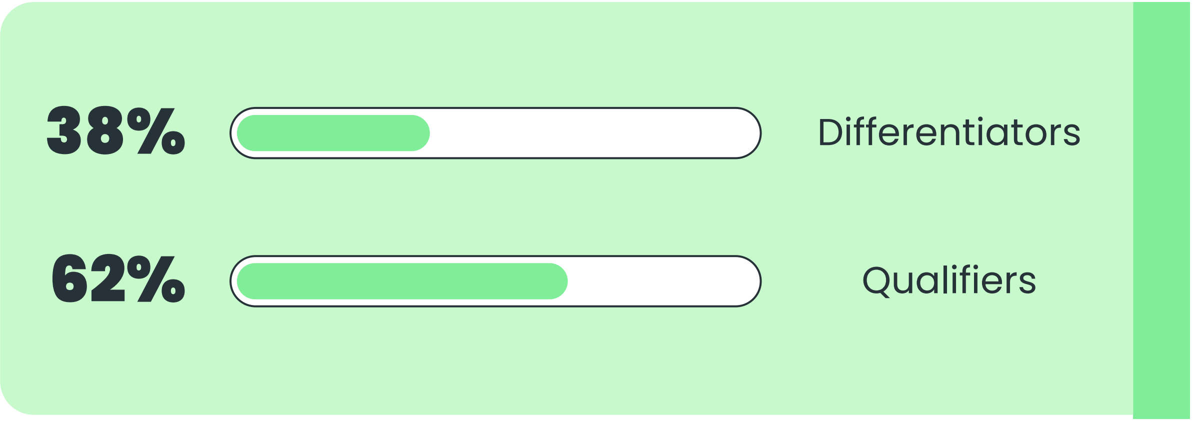
Artificial intelligence for burrs detection

- ➔ Machine learning

Exportation of database

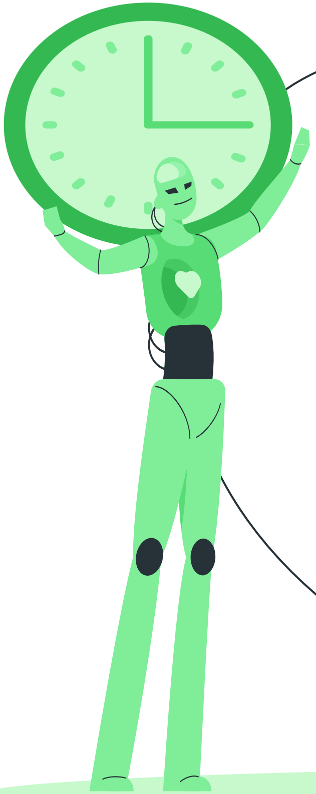
- ➔ Creating Knowledge

Product Requirements Solutions (PRS)



The solution needs more **added value** in order to get **more money** back

More Differentiators to distinguish the product



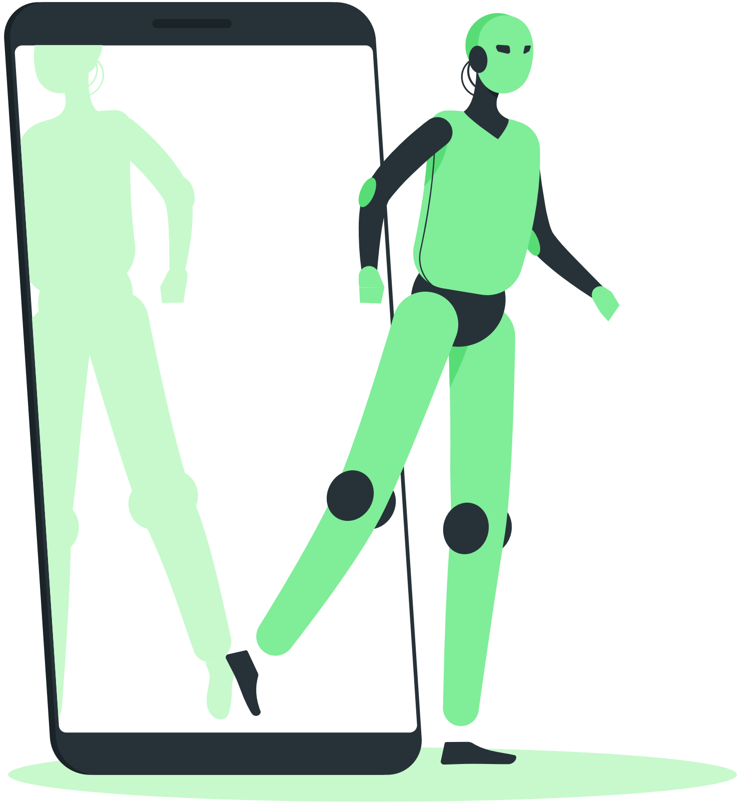
5

Project Plan

Project Plan (18 weeks)

Activities/Task	Responsible	Support by	Week number																			
			1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
TRL1: Project Definition	Ev.	Ev.	p	p	p	p																
Customer visit	Ev.	Ev.			A																	
Project Definition (Target, Scope, Deliverables)	Ev.	Ev.		A																		
Value Proposition definition	T & J	Ev.		A																		
Product Requirements Specification definition	H & N	Ev.			A																	
Business Case Analysis	E & D	Ev.				A																
Project Team definition and engagement	A & T	Ev.			A																	
Consolidate Master Plan for execution	A & T	Ev.				A																
Consolidate TRL1 presentation	Ev.	Ev.				MR																
Presentation TRL1	Ev.	Ev.				MR																
TRL2: Concept Definition						p	p	p	p	p	MR											
TRL3: Design											p	p	p	p	MR							
TRL4: Proof of Concept /Conclusions															p	p	p	p	p	MR		

Q&A



Thank you!

Quality Inspection Cell: Burrs detection TRL2

Mechatronic Design MR3009

P4.

José Angel Soto Hernández	A01039978
Nathalie Vilchis Lagunes	A01364838
Hector Everardo Martínez Cisneros	A01282300
Teclo Moreno Rodriguez	A01252067
Estefany Morales Valdes	A01281880
Diego A. Santisteban Pozas	A01154423
Jose Antonio Arrambide Garza	A00817790

29/09/2021

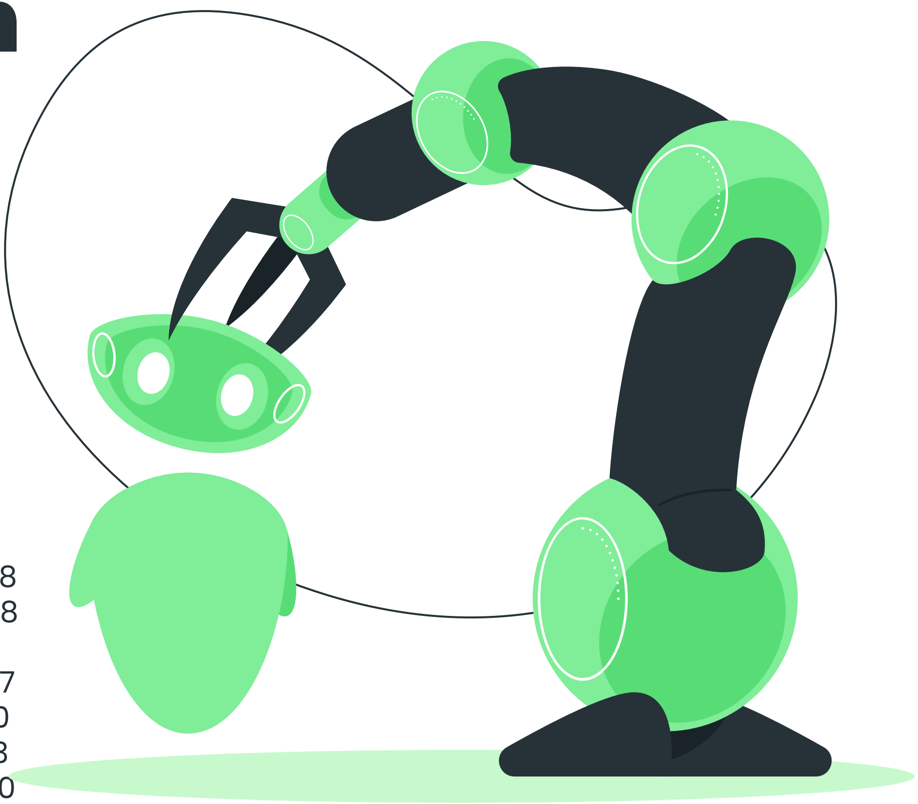
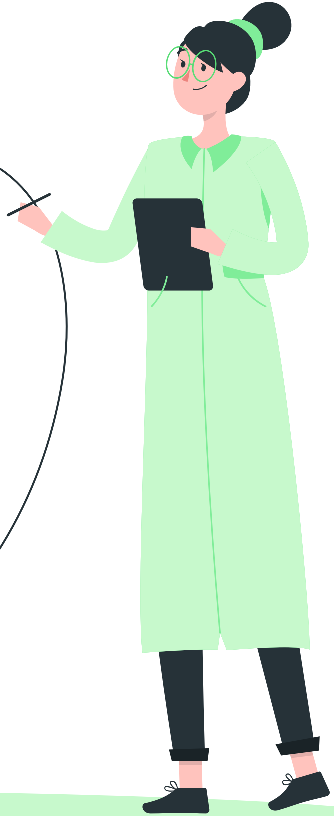


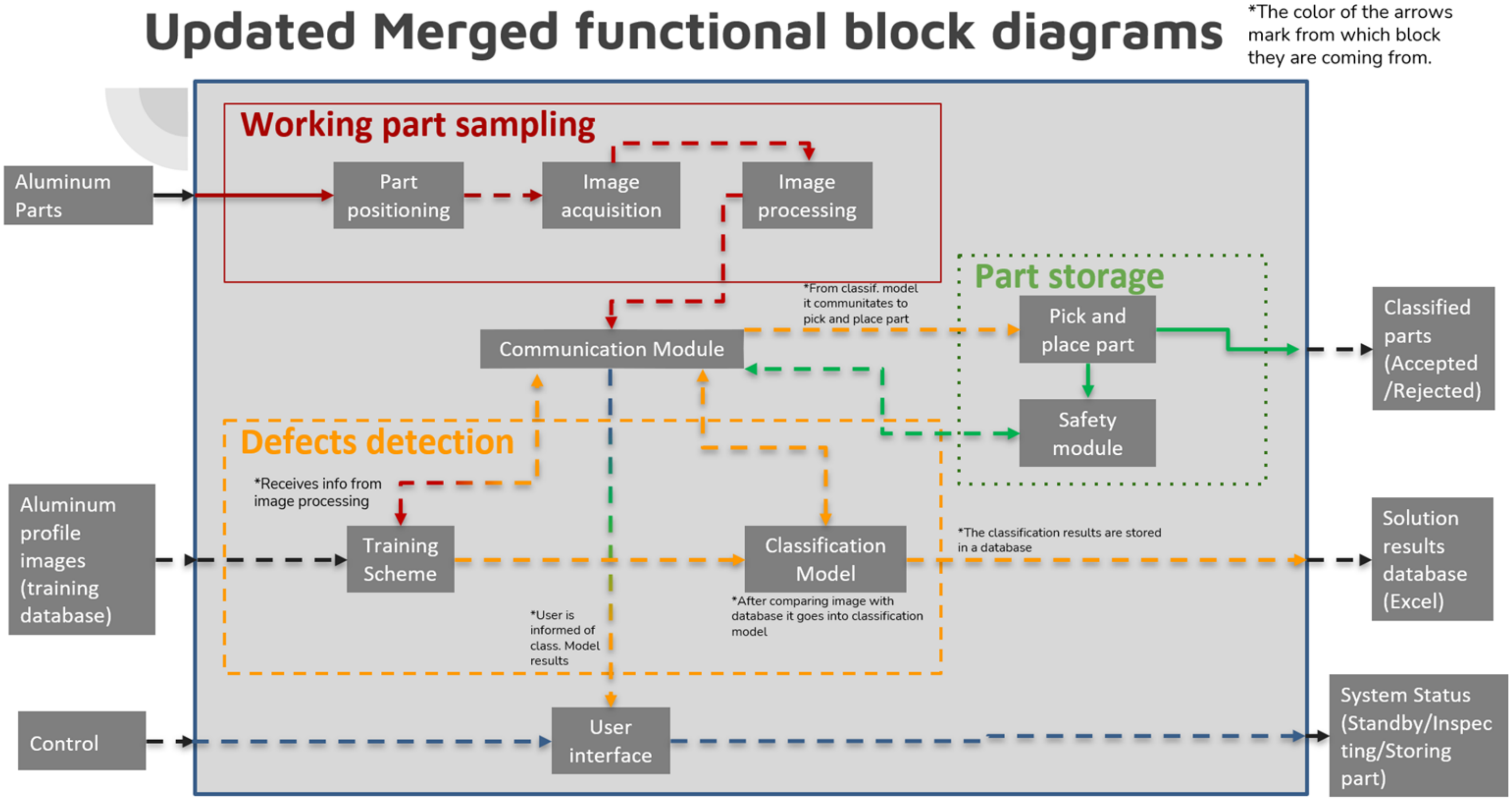
Table of Contents

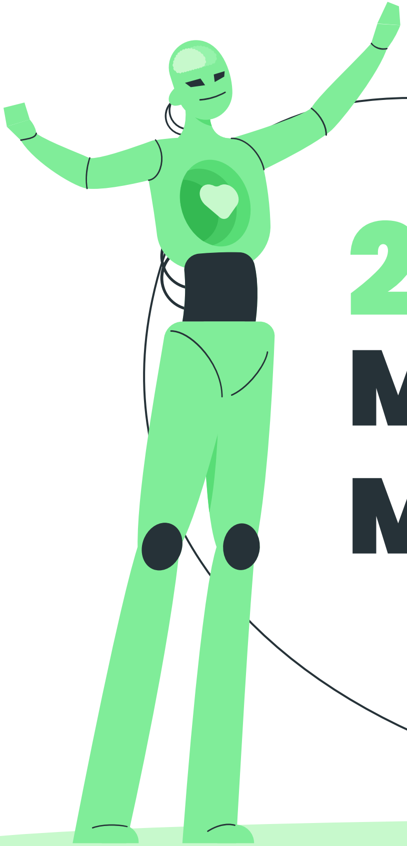
1

Functional Diagram

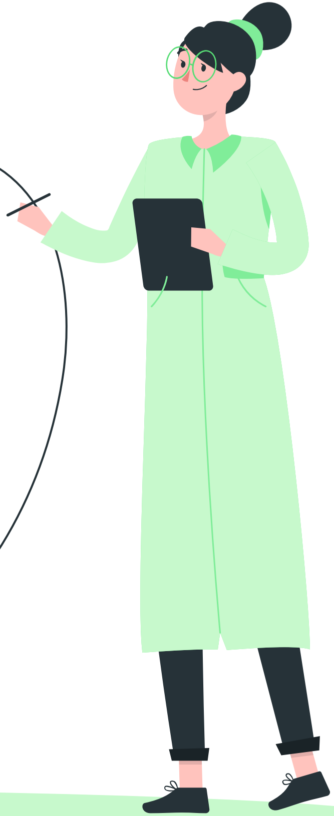


Functional diagram





2
Morphology Matrix



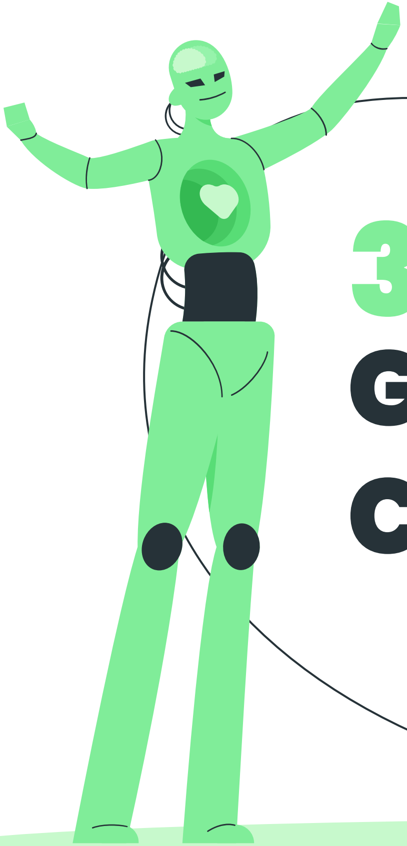
				Alternatives to Implement Functions			
Functional Modules	Working part sampling	Part positioning		Random location	Specified area of work (Human)	Conveyor	Dispenser of Parts
		Image acquisition	Part Location Image acquisition	Robot holding IP camera	No camera necessary (defined area of work)	IP Roof camera (for area scanning)	-
			Burr image acquisition	Robot holding IP camera	Specified area for image acquisition with multiples cameras	Fixed camera in workplace with the Robot rotating the part	-
		Image processing		MatLab	Python	Visual Studio	Insight Cognex
	Defects detection	Training scheme		Hold Out Sampling	Cross Validation	-	-
		Classification model (TBD after testing)		Logistic Regression	Support Vector Machine	Neural Network	Random Forest
	Part storage	Pick and place part		Cobot places classified parts in designed bins with mechanical gripper	Conveyor that classifies	Cobot with a vacuum suction gripper	
		Safety module		Wire mesh cage + Cobot collision function	Wire mesh cage + Cobot collision function + tray	Roof camera worker detection + Cobot collision function	Cobot collision function
	Communication module			Computer	Microcontroller (raspberry pi)	-	-
	User interface			LEDs + push button	LCD + push button	Mobile App	HMI Screen (Computer)

				Alternatives to Implement Functions			Low Cost
Functional Modules	Working part sampling	Part positioning		Random location	Specified area of work (Human)	Conveyor	Dispenser of Parts
		Image acquisition	Part Location Image acquisition	Robot holding IP camera	No camera necessary (defined area of work)	IP Roof camera (for area scanning)	-
			Burr image acquisition	Robot holding IP camera	Specified area for image acquisition with multiples cameras	Fixed camera in workplace with the Robot rotating the part	-
		Image processing		MatLab	Python	Visual Studio	Insight Cognex
	Defects detection	Training scheme		Hold Out Sampling	Cross Validation	-	-
		Classification model (TBD after testing)		Logistic Regression	Support Vector Machine	Neural Network	Random Forest
	Part storage	Pick and place part		Cobot places classified parts in designed bins with mechanical gripper	Conveyor that classifies	Cobot with a vacuum suction gripper	
		Safety module		Wire mesh cage + Cobot collision function	Wire mesh cage + Cobot collision function + tray	Roof camera worker detection + Cobot collision function	Cobot collision function
	Communication module			Computer	Microcontroller (raspberry pi)	-	-
	User interface			LEDs + push button	LCD + push button	Mobile App	HMI Screen (Computer)

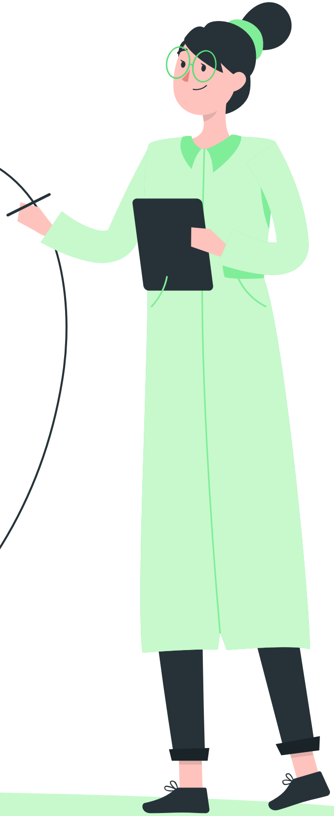
				Alternatives to Implement Functions			Fastest process
Functional Modules	Working part sampling	Part positioning		Random location	Specified area of work (Human)	Conveyor	Dispenser of Parts
		Image acquisition	Part Location Image acquisition	Robot holding IP camera	No camera necessary (defined area of work)	IP Roof camera (for area scanning)	-
			Burr image acquisition	Robot holding IP camera	Specified area for image acquisition with multiples cameras	Fixed camera in workplace with the Robot rotating the part	-
		Image processing		MatLab	Python	Visual Studio	Insight Cognex
	Defects detection	Training scheme		Hold Out Sampling	Cross Validation	-	-
		Classification model (TBD after testing)		Logistic Regression	Support Vector Machine	Neural Network	Random Forest
	Part storage	Pick and place part		Cobot places classified parts in designed bins with mechanical gripper	Conveyor that classifies	Cobot with a vacuum suction gripper	
		Safety module		Wire mesh cage + Cobot collision function	Wire mesh cage + Cobot collision function + tray	Roof camera worker detection + Cobot collision function	Cobot collision function
	Communication module			Computer	Microcontroller (raspberry pi)	-	-
	User interface			LEDs + push button	LCD + push button	Mobile App	HMI Screen (Computer)

				Alternatives to Implement Functions			Most Reliable
Functional Modules	Working part sampling	Part positioning		Random location	Specified area of work (Human)	Conveyor	Dispenser of Parts
		Image acquisition	Part Location Image acquisition	Robot holding IP camera	No camera necessary (defined area of work)	IP Roof camera (for area scanning)	-
			Burr image acquisition	Robot holding IP camera	Specified area for image acquisition with multiples cameras	Fixed camera in workplace with the Robot rotating the part	-
		Image processing		MatLab	Python	Visual Studio	Insight Cognex
	Defects detection	Training scheme		Hold Out Sampling	Cross Validation	-	-
		Classification model (TBD after testing)		Logistic Regression	Support Vector Machine	Neural Network	Random Forest
	Part storage	Pick and place part		Cobot places classified parts in designed bins with mechanical gripper	Conveyor that classifies	Cobot with a vacuum suction gripper	
		Safety module		Wire mesh cage + Cobot collision function	Wire mesh cage + Cobot collision function + tray	Roof camera worker detection + Cobot collision function	Cobot collision function
	Communication module			Computer	Microcontroller (raspberry pi)	-	-
	User interface			LEDs + push button	LCD + push button	Mobile App	HMI Screen (Computer)

Functional Modules				Alternatives to Implement Functions			SAFEST
Functional Modules	Working part sampling	Part positioning		Random location	Specified area of work (Human)	Conveyor	Dispenser of Parts
		Image acquisition	Part Location Image acquisition	Robot holding IP camera	No camera necessary (defined area of work)	IP Roof camera (for area scanning)	-
			Burr image acquisition	Robot holding IP camera	Specified area for image acquisition with multiples cameras	Fixed camera in workplace with the Robot rotating the part	-
		Image processing		MatLab	Python	Visual Studio	Insight Cognex
	Defects detection	Training scheme		Hold Out Sampling	Cross Validation	-	-
		Classification model (TBD after testing)		Logistic Regression	Support Vector Machine	Neural Network	Random Forest
	Part storage	Pick and place part		Cobot places classified parts in designed bins with mechanical gripper	Conveyor that classifies	Cobot with a vacuum suction gripper	
		Safety module		Wire mesh cage + Cobot collision function	Wire mesh cage + Cobot collision function + tray	Roof camera worker detection + Cobot collision function	Cobot collision function
	Communication module			Computer	Microcontroller (raspberry pi)	-	-
	User interface			LEDs + push button	LCD + push button	Mobile App	HMI Screen (Computer)

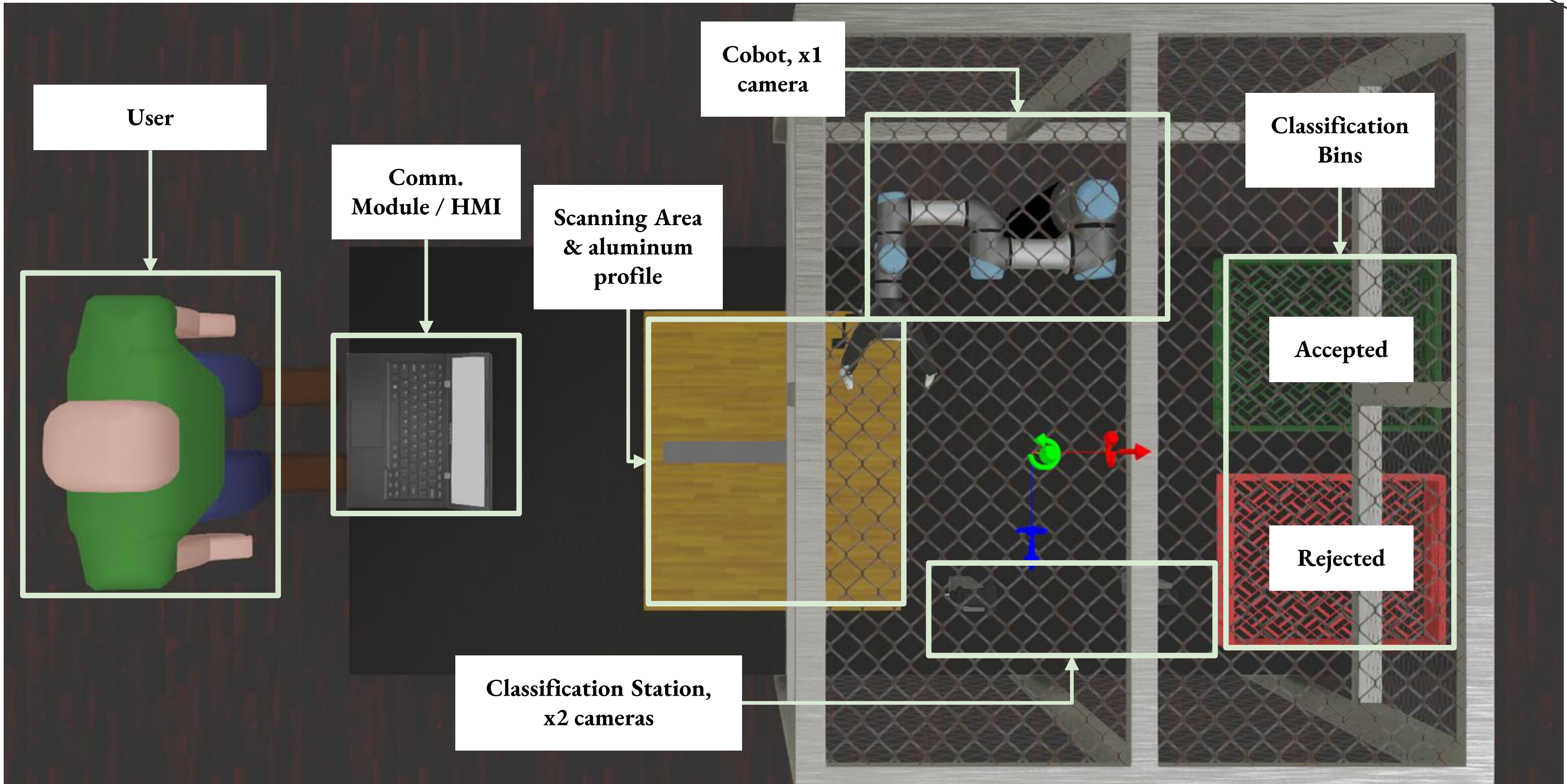


3
Generated Concepts



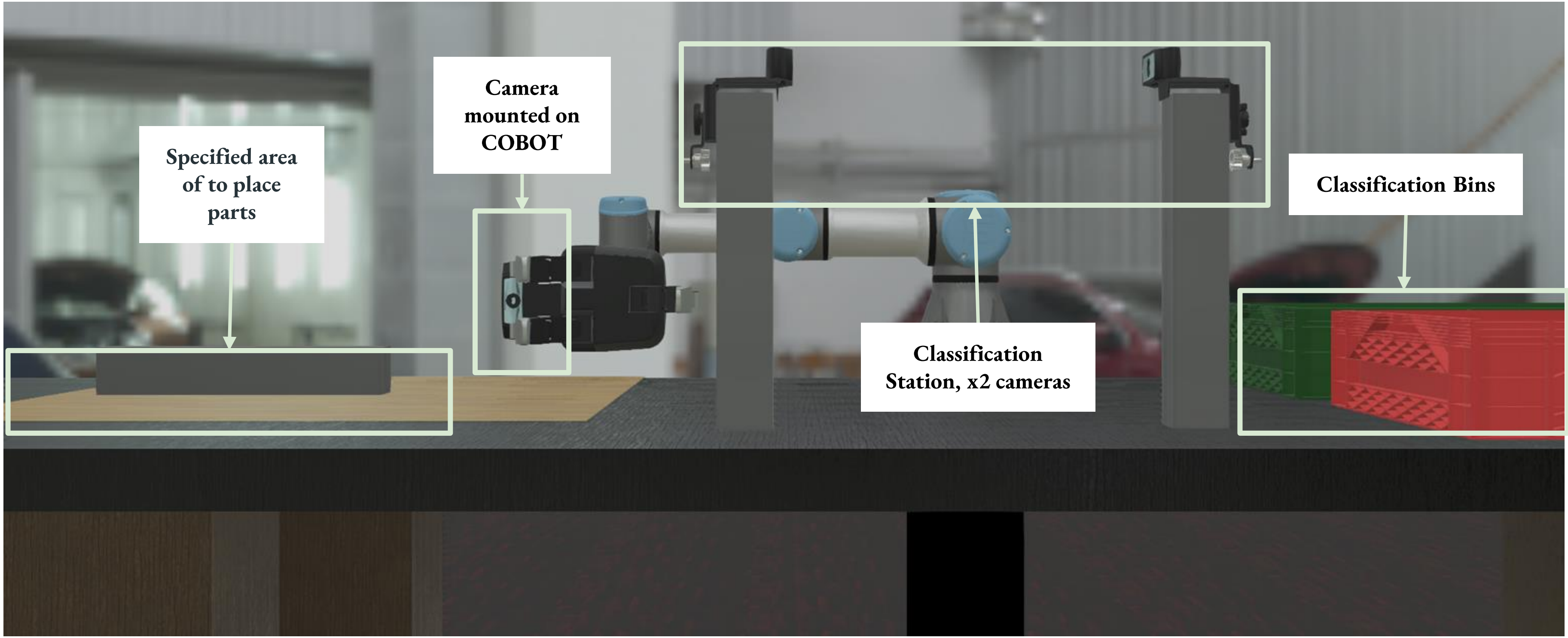
Most Reliable

Most reliable concept – Top view

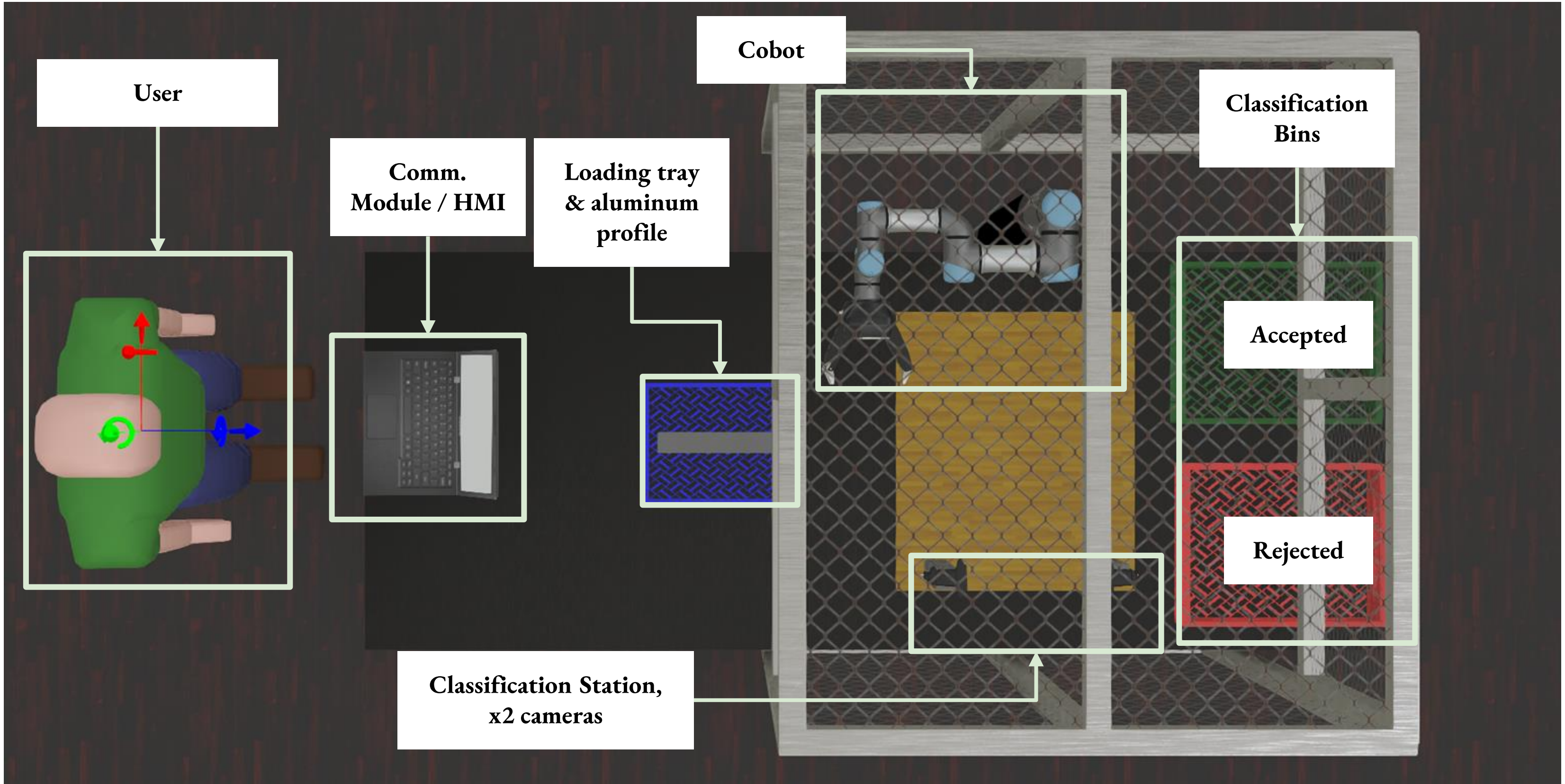


Most Reliable

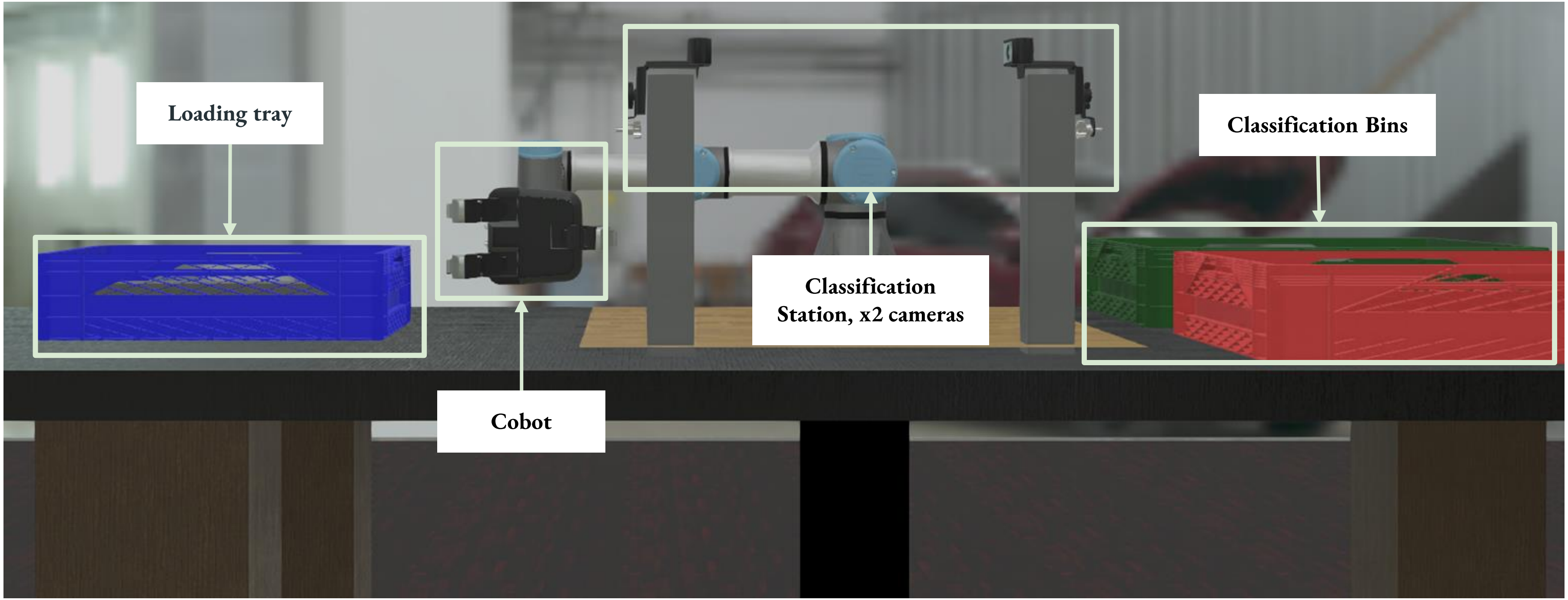
Most reliable concept – close up (w/no cage)



Safest concept – Top view

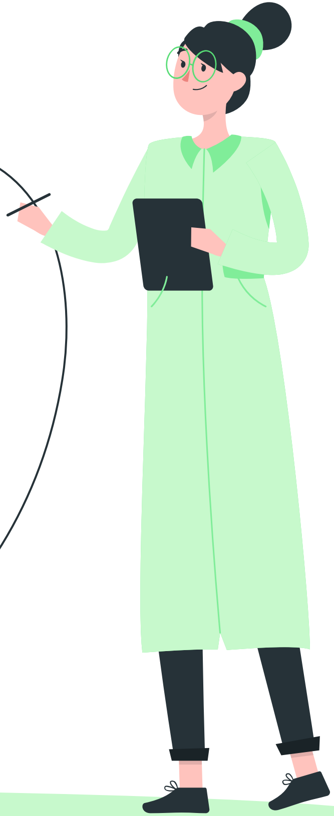


Safest concept – close up (w/no cage)



4

Selection Criteria

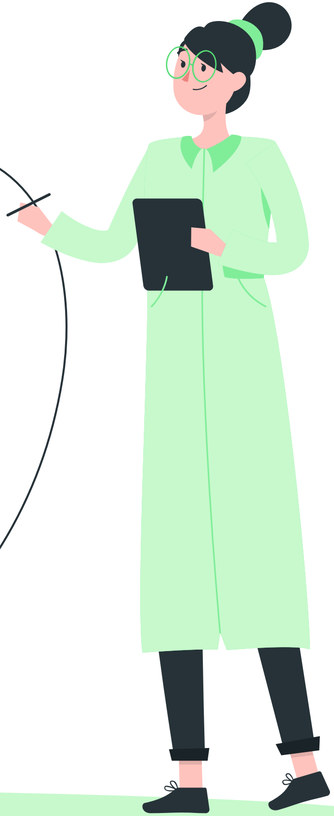


Selection Criteria

Based on the customer values:

- High precision
- Safety
- Low Cost
- Fast Process

5 Selected Concepts

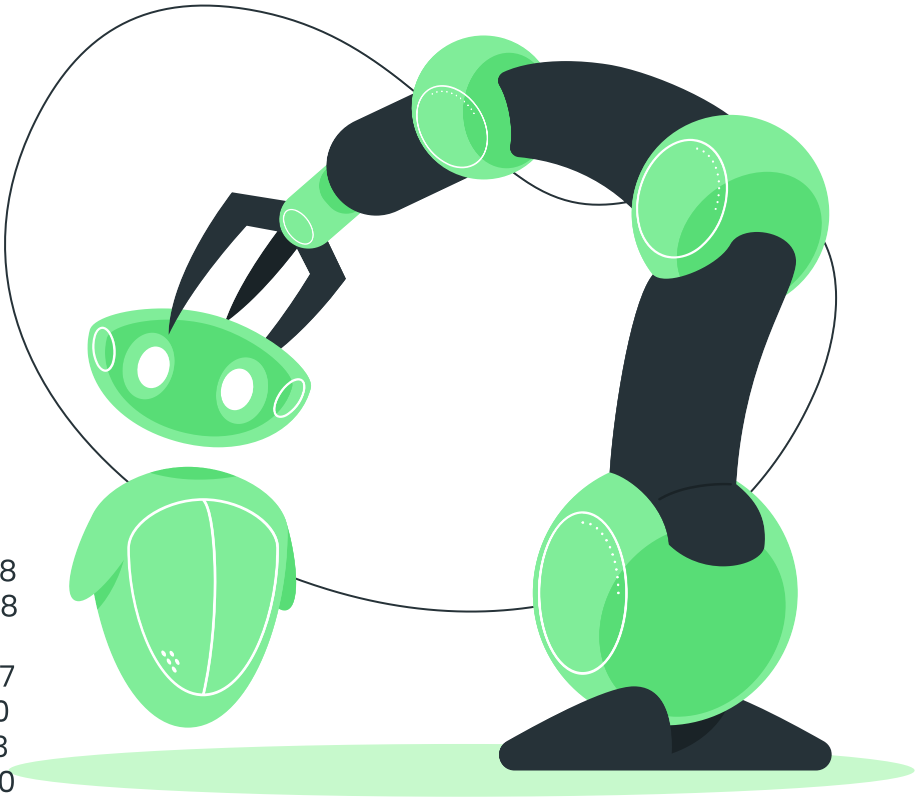


Pugh Matrix

Less than spec 0
Sames as spec 1
More than spec 1.1

Weight	Customer Value	Low Cost	Fastest Process	Most reliable	Safest
0.35	1. High Precision	0	1	1.1	1.1
0.35	1. Safety	0	1	1	1.1
0.2	2. Low cost	1.1	0	0	0
0.1	3. Fast process	0	1.1	1	1
1	**Total Score**	27.5%	77.5%	77.5%	80%
	Weighted total Score	22.0%	81%	83.5%	87%

Quality Inspection Cell: Burrs detection

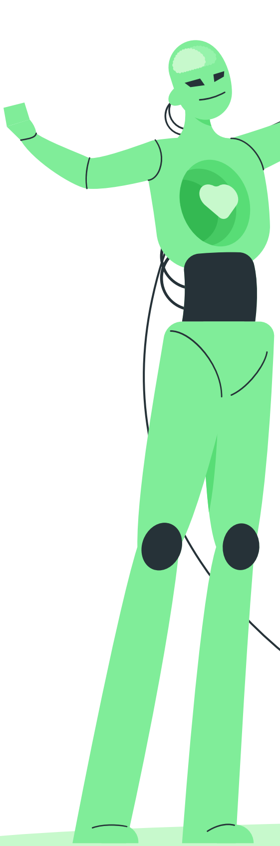


Mechatronic Design MR3009

P4.
José Angel Soto Hernández A01039978
Nathalie Vilchis Lagunes A01364838
Hector Everardo Martínez Cisneros A01282300
Teclo Moreno Rodriguez A01252067
Estefany Morales Valdes A01281880
Diego A. Santisteban Pozas A01154423
Jose Antonio Arrambide Garza A00817790

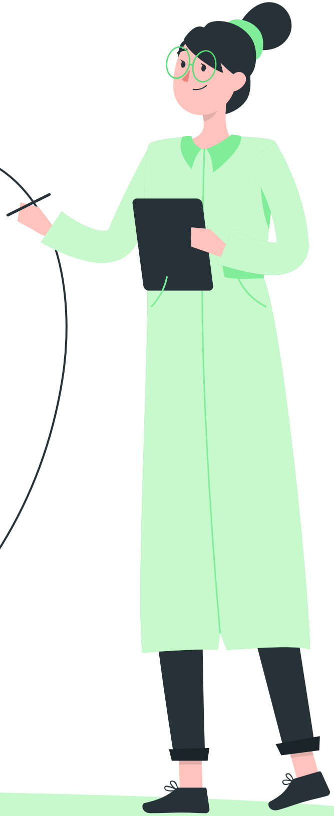
03/11/2021

Table of Contents

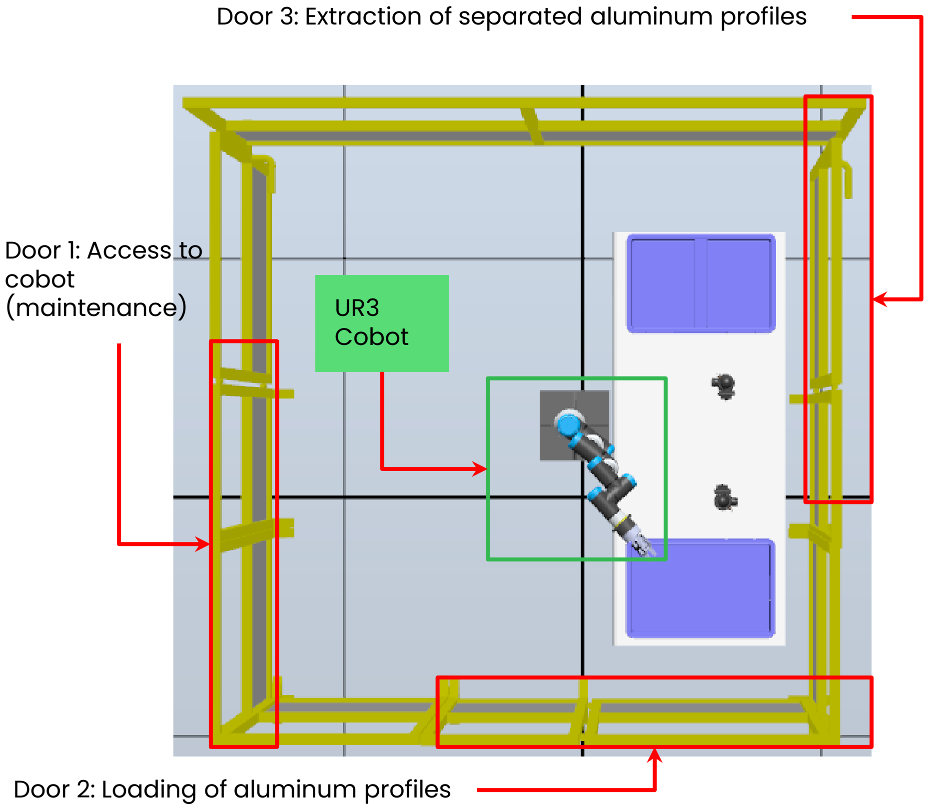


1

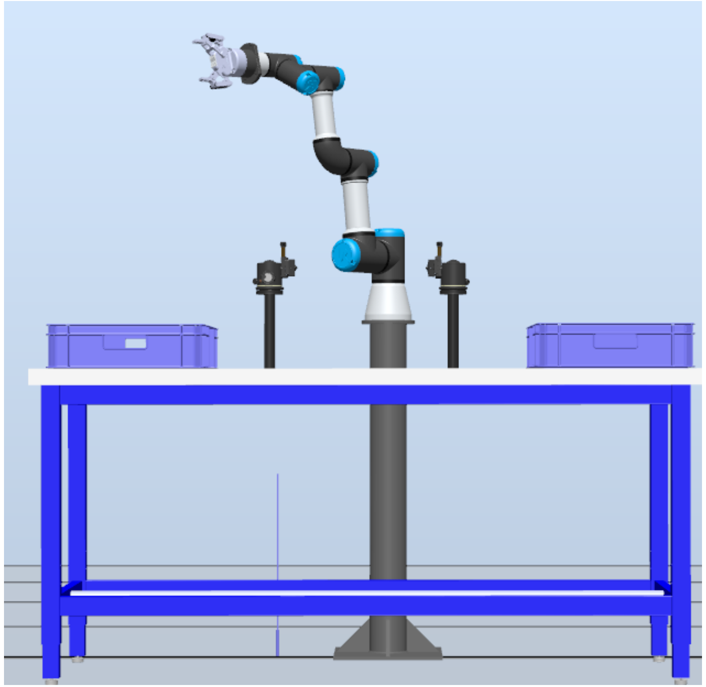
Detailed designs of selected concept



Quality Station (RobotStudio)



*The UR3 Cobot will be rec

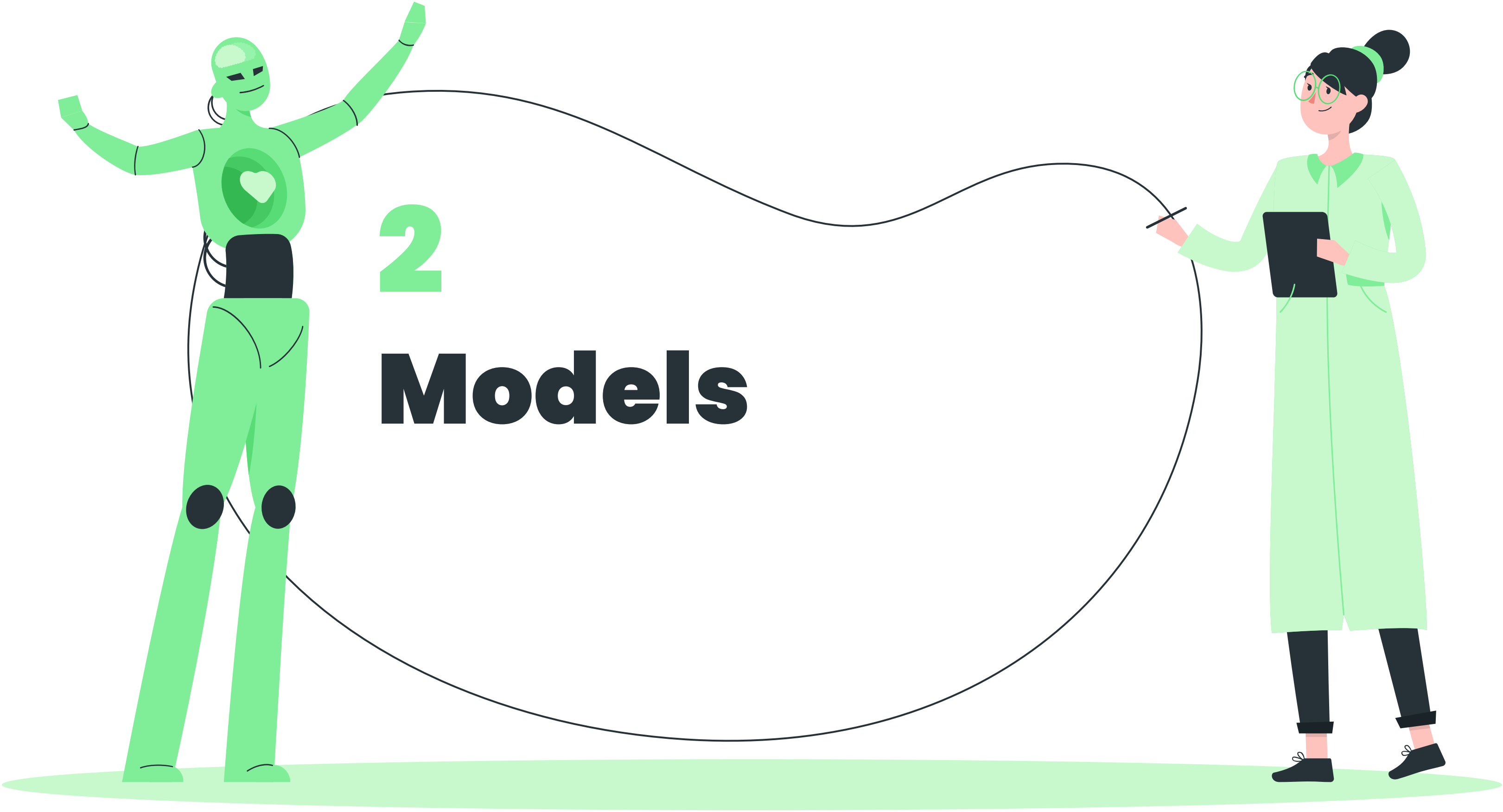


Operation

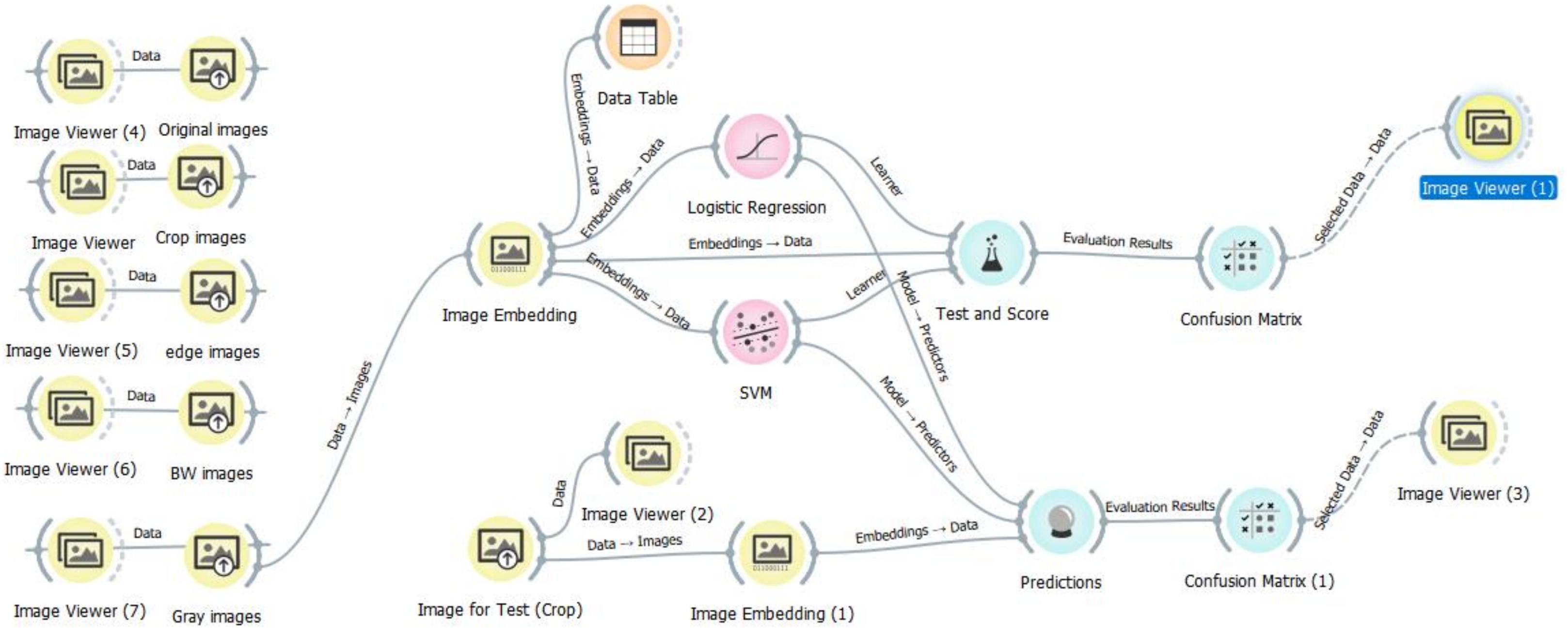
1. User loads the drilled aluminum profiles in a designated area and close the station door
2. User commands cobot to pick the aluminum profiles when ready.
3. The cobot moves the profile between the two ip cameras.
    a. The cobot slightly rotate profiles (For angular pictures)
4. The pictures will be send to python/orange trainer
5. Trainer communicates the classification results to cobot through python.
6. Cobot separates profiles according to results
7. User may extract classified profiles

2

Models

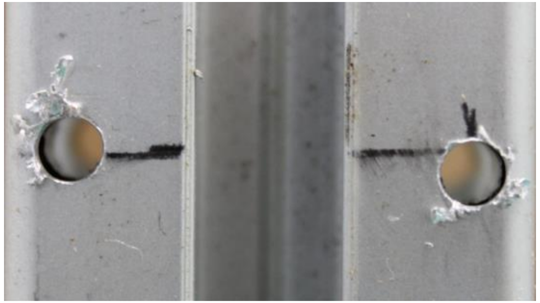


Classification training (Orange)



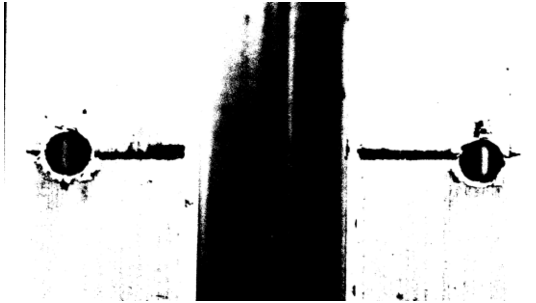
Filters

Front-Original

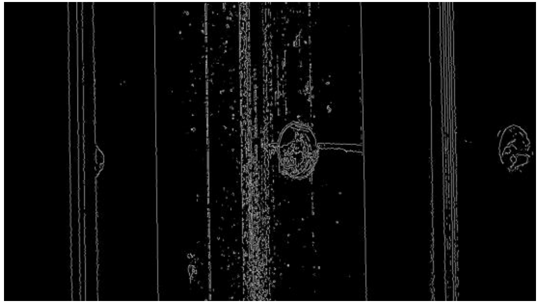


Angled-Grayscale

Frontal-Binary



Angled-Edge



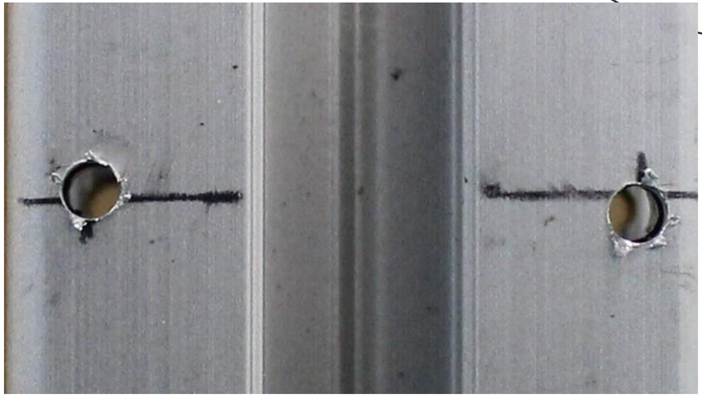
First Test

Datasets

- Original
- Gray
- Edge
- Binary

Best Model :

87.5% accuracy

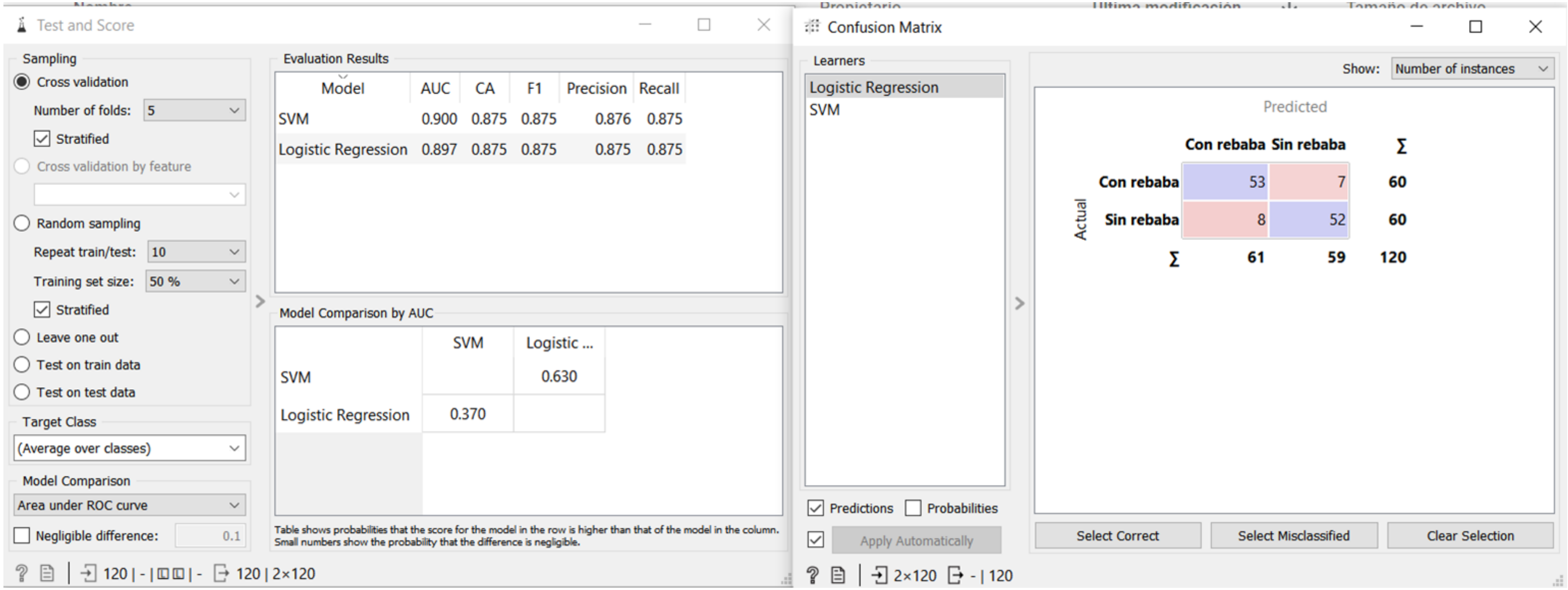


Conclusions

- Cropped images work better than originals
- The best results are with grayscale images
- Front images work noticeably better than angled ones
- Cross Validation with 5 divisions is the best performing sample type

Best Model from 1st Test 87.5 % accuracy

★ Logistic regression and SVM with Cross Validation (V3)



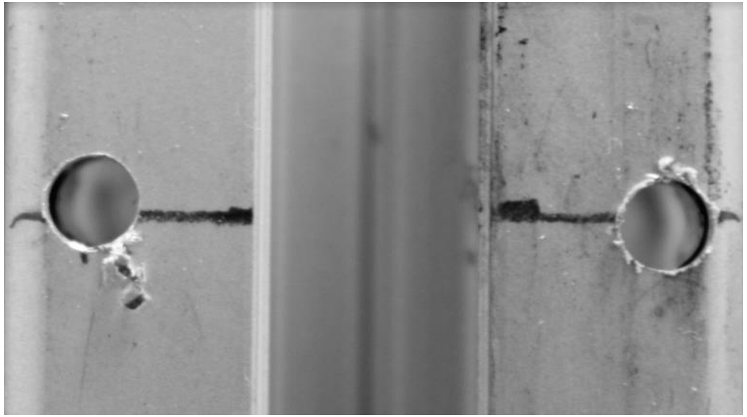
2nd Test

Datasets

Frontal crop images in grayscale

Best Model :

92.9% accuray

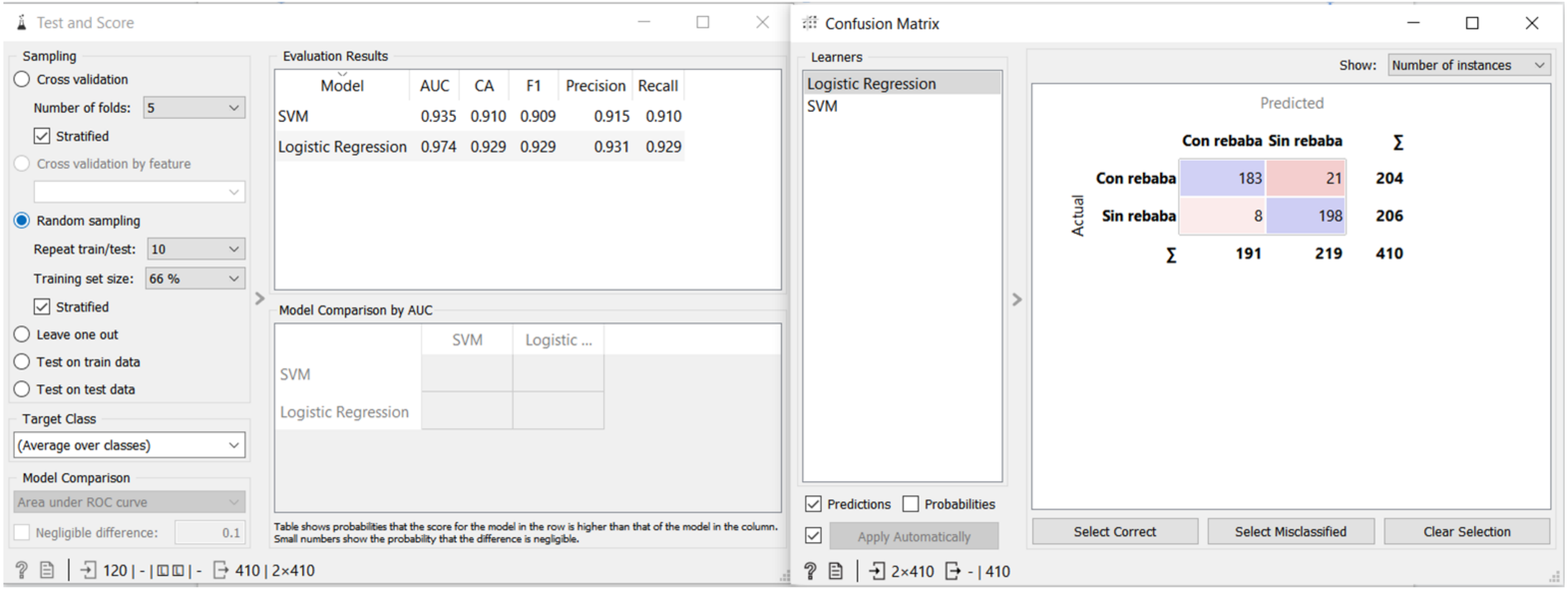


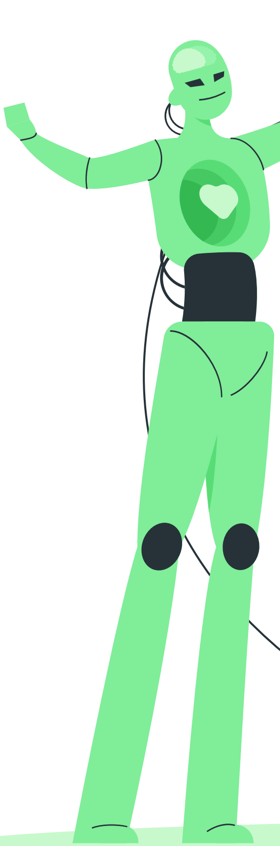
Parameters to vary:

- ❑ Embedder
- ❑ Sampling method
- ❑ Logistic Regression Strength
- ❑ Logistic Regression Regularization Type
- ❑ SVM type
- ❑ SVM Kernel

Best Model from 2nd Test： 92.9 % accuracy

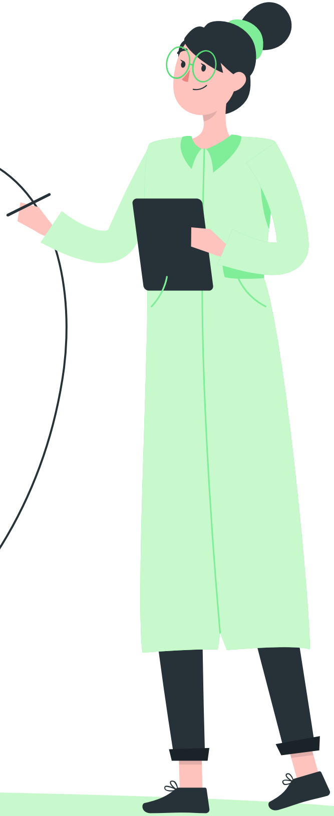
★ Logistic Regression with Random Sampling（VGG19）



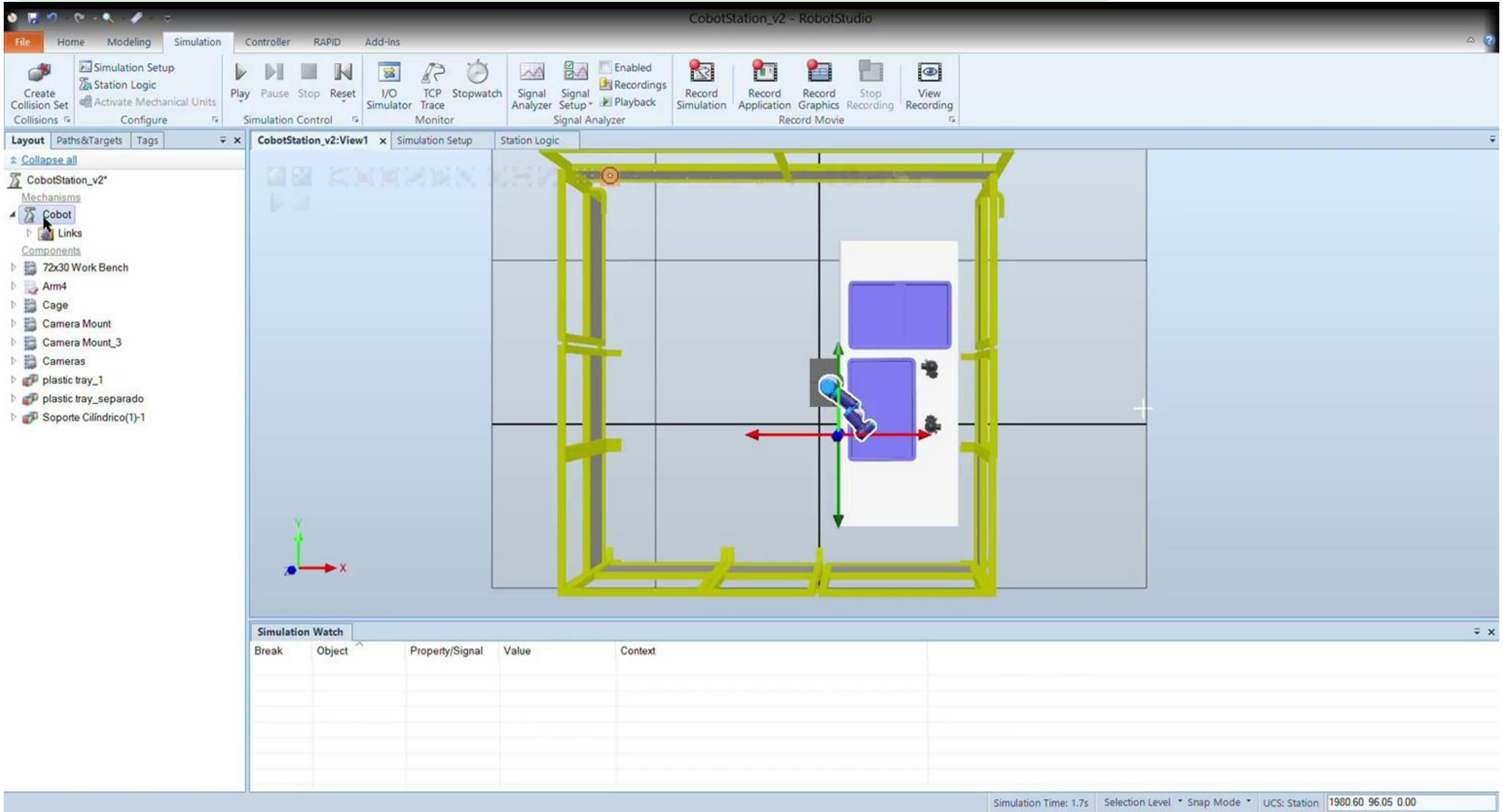


3

Virtual Prototypes



RobotStudio



Python & Orange

- Module for Python Script in Orange
- Library of Orange for Python

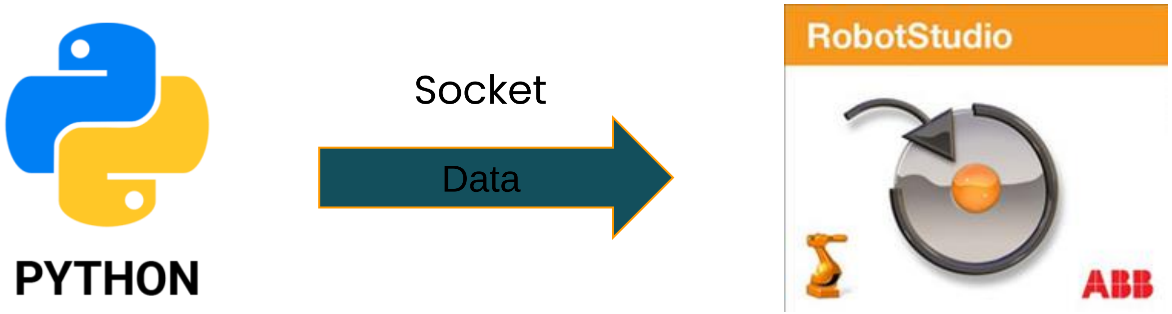
\- Classification models can be generated directly in Python

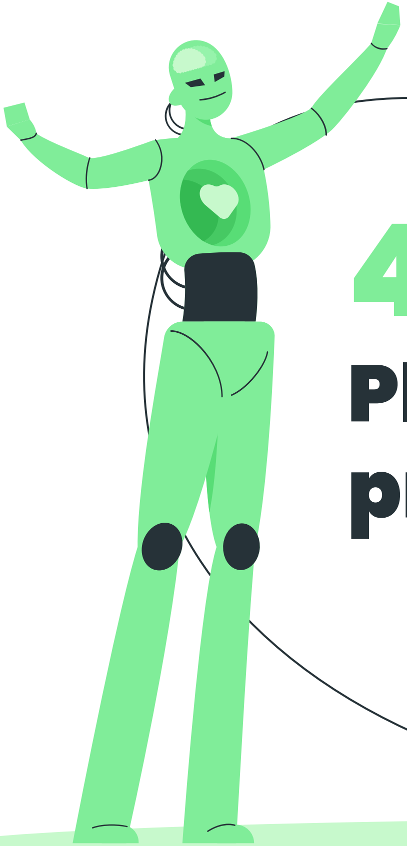
PYTHON

orange
DATA MINING
FRUITFUL&FUN

RobotStudio & Python Communication

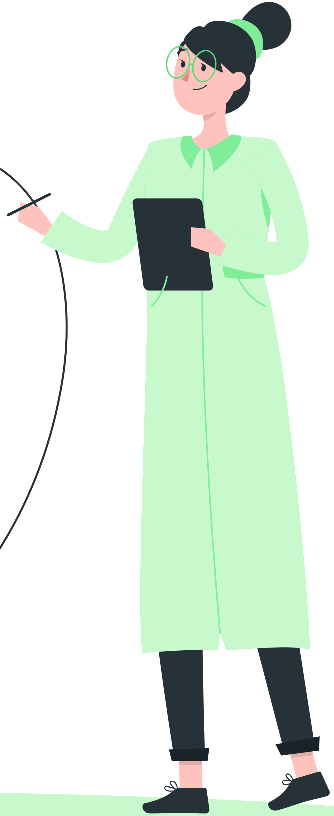
- Python needs to send the variable of accepted or rejected to RobotStudio
- Communication will be established through TCP/IP protocol (sockets)
- The client will be Python and server will be the server.





4

Plan for final prototype

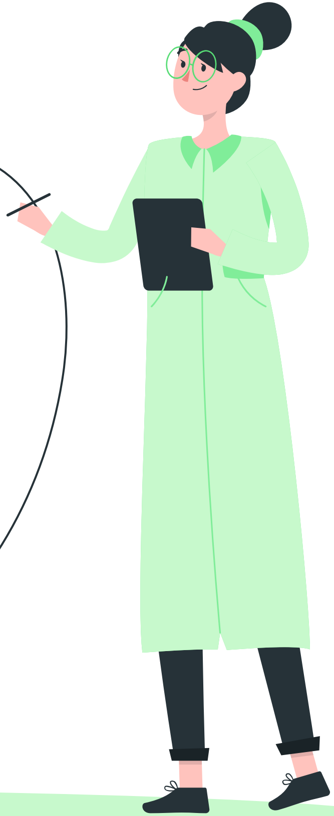


Plan for Final Prototype

- Integrate all the programs with real data exchange to simulate as close to reality as possible.
- Adjust the models, designs and approaches to ensure reaching the requirements of the client such as: Time, Quality, Reliability, Safetiness.
- Run enough tests in order to generate the specifications and recommendations of the final product.

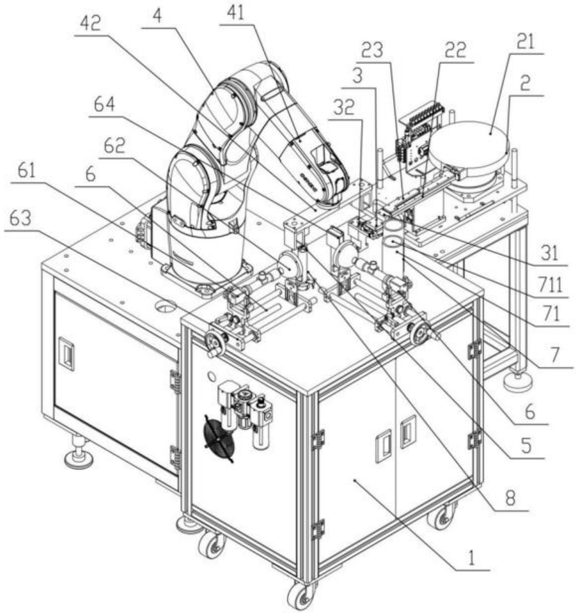
5

Freedom to Operate



360 degrees product burrs detection device of rotation type

- Applicant: Dongguan Shengxiang Precision Metal Co Ltd
- Application date: 2018-05-29
- Publication date: 2018-12-11

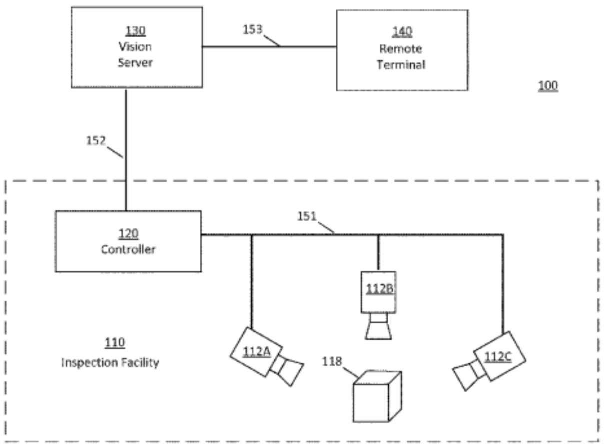


Extracted from:
https://patents.google.com/patent/CN108982509B/en?q=burrs+detection&oq=burr

Other patents...

Machine-vision system and method for remote quality inspection of a product

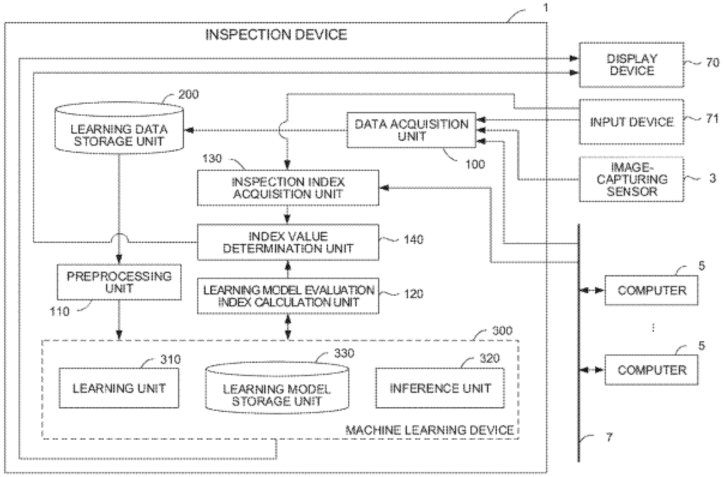
- Applicant: SIGHT MACHINE Inc
- Application date: 2018-04-10
- Publication date: 2019-03-21



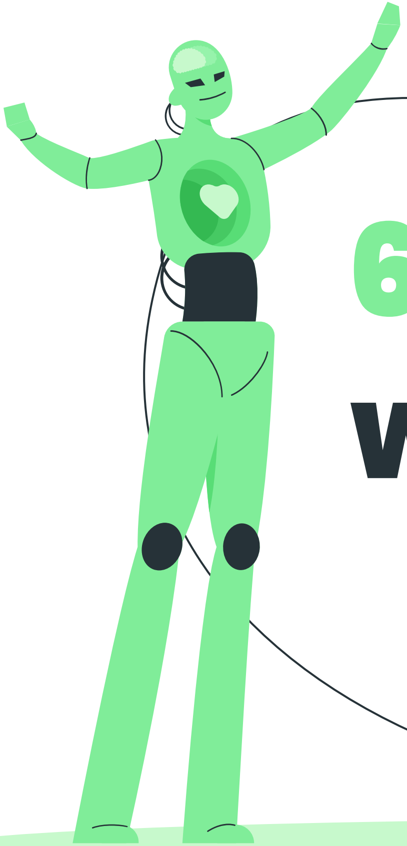
Extracted from:
https://patents.google.com/patent/US20200082225A1/en?q=(computer+vision+manufacturing+quality+control)&language=ENGLISH&oq=(computer+vision+manufacturing+quality+control)+language:ENGLISH&page=1

Inspection device and machine learning method- Analyzer

- Applicant: Keisuke Watanabe, Yasuhiro Shibasaki
- Application date: 2018-09-12 (JP) & 2019-09-12 (DE-US-CN)
- Publication date: 2021-01-27 (JP)

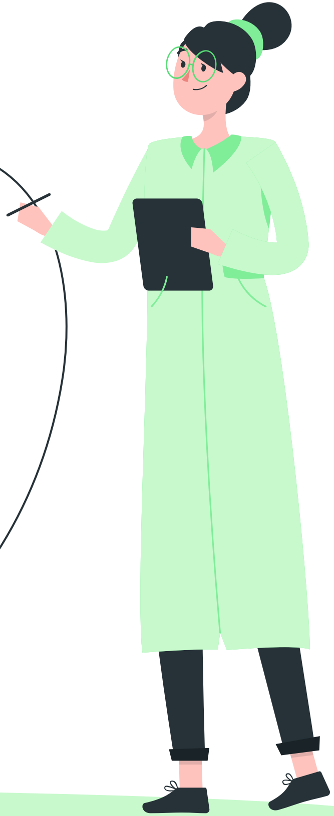


Extracted from:
https://patents.google.com/patent/US20200082225A1/en?q=(computer+vision+manufacturing+quality+control)&language=ENGLISH&oq=(computer+vision+manufacturing+quality+control)+language:ENGLISH&page=1



6

What's next?



Next Steps

- Interconnect all the systems for the final prototype.
- Testing results to define the reliability and limitations of the performance.
- Optimization of parameters to meet the requirements of the client.
- Evaluate the risk of the limitations and compare to the original project definition.
- Define the specifications, claims and recommendations of the final product.

TRL4: Proof of Concept /Conclusions																				
					/	/	/	/	/	/	P	P	P	P	P		P	P	P	MR
Testing and improvement	8	0	Hector	Team	/	/	/	/	/	/	/	/	/	/	/		P			
Summary of results			Teclo	Team	/	/	/	/	/	/	/	/	/	/	/		P			
Cost Estimation and final Business Case doc			Nathalie	Team	/	/	/	/	/	/	/	/	/	/	/		P			
Technology Readiness Assessment			Diego	Team	/	/	/	/	/	/	/	/	/	/	/			P		
Risk Assessment			Estefy	Team	/	/	/	/	/	/	/	/	/	/	/			P		
Final IPR recommendation			Antonio	Team	/	/	/	/	/	/	/	/	/	/	/			P	P	
Final Recommendations			Antonio	Team	/	/	/	/	/	/	/	/	/	/	/			P	P	
Consolidate TRL4 Report			Jose	Team	/	/	/	/	/	/	/	/	/	/	/					MR
Presenation TRL4			Jose	Team	/	/	/	/	/	/	/	/	/	/	/					MR

Major Risks

Time is the main concern. Right now we are all in on a solution that involves robot studio, python and orange. If for some reason the communication between the 3 were to fail or show to be inconsistent. Realistically, time wouldn't allow for a secondary solution to take place.

Robot Studio: working without an ABB Robot that already includes its own virtual controller is a challenge since the team lacks experience with the software and most digital resources assume that an ABB robot is chosen. However, as shown in the video, it is possible to create a full working station with a third party robot.

Computer vision: if the lighting conditions and other factors with which the training data was obtained can't be fully replicated or approached on the field, then we could see a drop in performance for the classification algorithm. Making a field run test and calibrating the algorithm would be an important step before implementing the final solution.