WP8 – Use-cases set-up and on-site validation

D8.2 - MERGING robotic system validation on THIMONNIER use-case

Initial due date: M47 – 30/09/2023

Submission date: M48 – 31/10/2023

Version number: V1

Responsible partner

THIMONNIER : B. GRAS

Contributing partners

CEA : G. ACHER, T. BOULMIER, N. BRUCKMANN, O. LEBEC, V. MOLINA, A. REBBOUH, B. GRADOUSSOFF

LMS : D. ANDRONAS, P. ANGELAKIS, K. KAVVATHAS

EPFL : H. SHEA

AIMEN : C. SANCHEZ, I. FERNANDEZ

THIMONNIER : N. CROZET, B. GRAS, S. MARQUAT, J. PERRIN, I. TAJMOUT

OMNIGRASP : F. GARGANO, D. GRASSO, V. CACUCCIOLO

Summary

This demonstration shows the robotized workcell prototype at THIMONNIER premises and its evaluation in near-real industrial conditions. This use-case addresses the insertion of flexible pouches into a feeding rail, to be installed as a repetitive process at the input of a liquid (or semi-liquid) food-packaging machine.

Executive summary

This report documents the integration activities at THIMONNIER premises for the deployment of a robotized workcell prototype for the automation of pouches insertion into a packaging-machine feeding-rail. The report outlines the core technological modules of the Merging solution, and it continues with a description of its operation. The controlled real scenario demonstrates the system handling strategies and capabilities. This report consists of a supplementary description of the THIMONNIER demonstration video (internet link for its download is provided in the first paragraph of this document) towards the complete comprehension of the implemented functionalities and integrated results.

Table of content

1. Introduction

This report describes the demonstration video deliverable that concludes the activities realized in task T8.2. The document highlights key aspects and functionalities of the demonstrated manufacturing process as well as outlines the core technological modules that contributed to the integrated result. The report consists of the following sections:

- Section 2: summary of integration activities
- Section 3: description of hardware integration and key hardware modules of the system
- Section 4: description of software integration and key software modules of the system
- Section 5: description of integrated demonstration scenario and respective handling strategies and system functionalities
- Section 6: concluding remarks and outlook

The video file that showcases the demonstrator's operation described here, can be found at this link hosted by LMS servers:

- https://syrios.mech.upatras.gr/nextcloud/index.php/s/nEeNpCwYETmHztc
- *using password : H2020_MERGING_WP8*

2. Integration activities

The demonstrator is the result of technical activities of work packages WP3, WP4, WP5 and WP6, as well as integration activities in the context of WP7.

Each building block has been developed separately by their responsible partners, then sent to Thimonnier separately for final integration.

Due to a lack of competences on ROS at Thimonnier during the period, we had to lead integration activities with physical workshops taking place between RTO and integration partners. Hereafter, some pictures show the workshop that has been done with OMNIGRASP. Other physical workshops, dedicated to control and software aspects, have been held at Thimonnier with CEA and LMS respectively[1]. The integration and validation involvement of Thimonnier contributed to the results obtained, through industrial feedback or direct technical involvement.

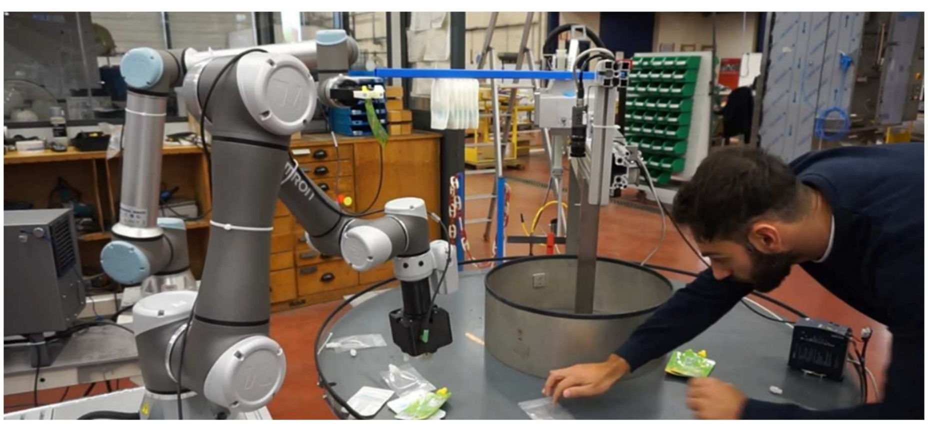


Figure 1: OMNIGRASP workshop at Thimonnier premises – localization building block on-site fine tuning.

[1] There is no picture of the participants (B. Gradoussoff, P. Angelakis, K. Kavvathas and D. Andronas).

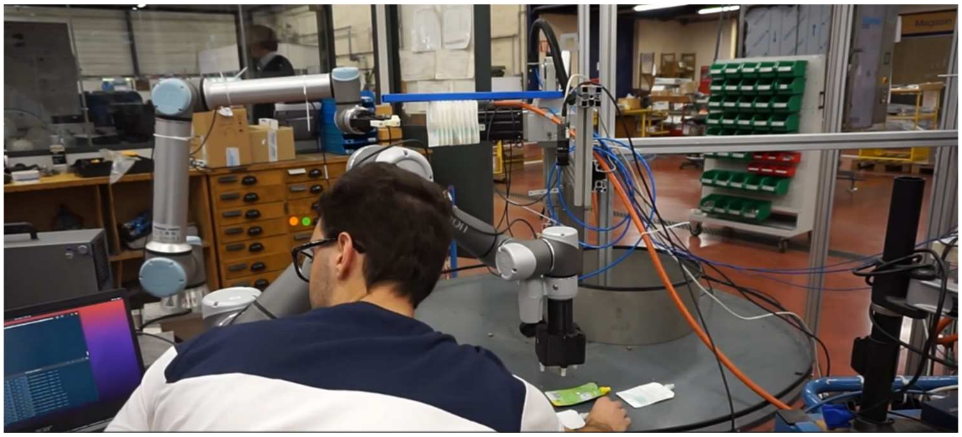


Figure 2: OMNIGRASP workshop at Thimonnier premises – ROS interfacing between test bench and gripper building blocks.

3. Hardware integration

In the context of WP8, CEA assisted Thimonnier for the design and integration activities required to implement the demonstrator at Thimonnier premises. As presented in Figure 3, the system's main resource is the dual arm manipulator. The manipulator is supported by a series of hardware and software modules that contribute to the enhancement of its perception (WP5 modules), its dexterity (WP3, WP4, WP6 modules) and its cognition (WP4, WP5 modules).

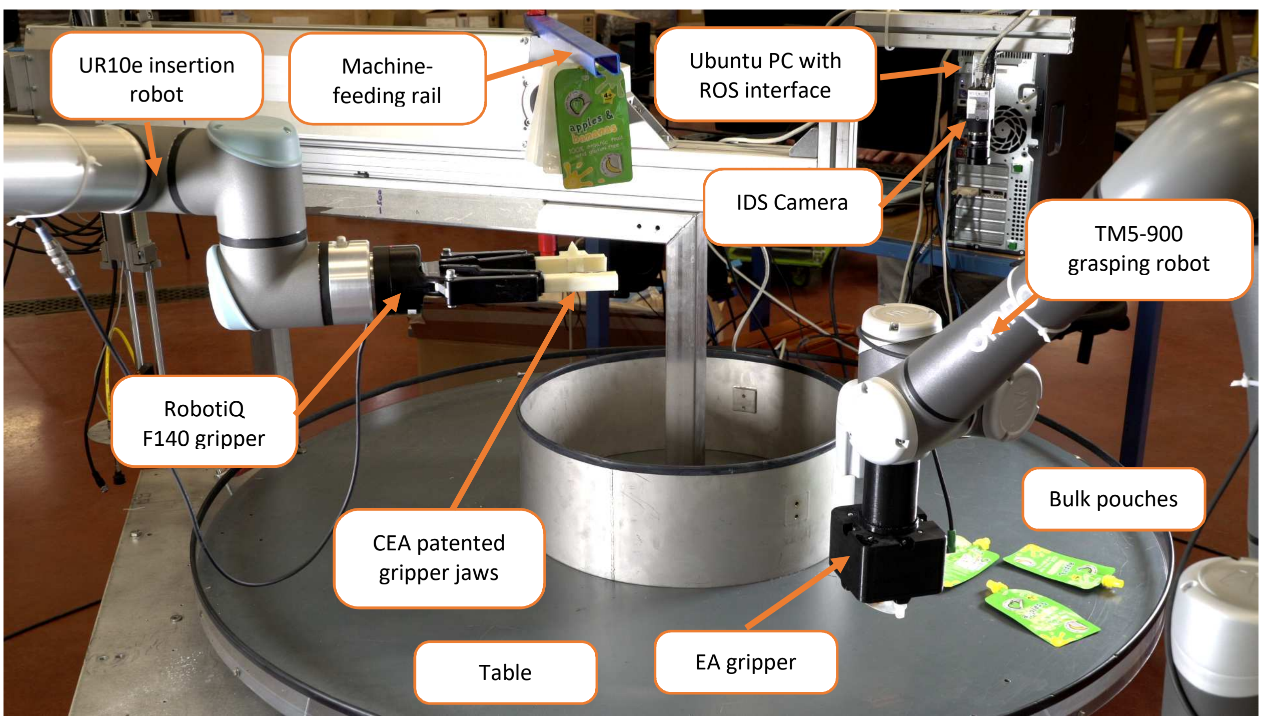


Figure 3. Overview of the Thimonnier demonstrator

Hereafter, the most important modules of the system are enlisted. In detail:

- UR10e insertion robot: The system is equipped with one Universal Robots UR10e arm. His main function is to insert grasped pouches into the final rail.
- TM5-900 grasping robot: This Omron robot can be mounted with two kinds of gripper: one substitution with a Venturi and a suction cup, or the electro-adhesive (EA) gripper made by OMNIGRASP (in collaboration with EPFL for the skins).
- Picking workbench: it is a workbench intended for the bulk placement of pouches by the operator. This is where the pouches will be automatically detected and grasped by the robotic system.
- Camera: it is an IDS camera, that is mounted over the test bench. The positioning of the camera ensures that most of pouches can be detected on the table.
- Ubuntu PC with ROS interface: In order to orchestrate the full cycle, and process every building block developed by the consortium on ROS. The test bench is equipped with a 20.02 Ubuntu PC device.
- Human machine interfaces: In this case, everything is made with the Ubuntu PC and a screen allocated to the demonstration.
- SPIRE: it consists of the framework by CEA for developing new robotic skills. The module provides interfaces for the execution and monitoring of tasks resource activities.

4. Software integration

In order to account for difficulties encountered on WP3 progress, and to limit its impact on the time line of the project, the final Thimonnier demonstrator integration has been realized in three steps, each allowing partial tests and validation of sub-assemblies of the elementary Merging building blocks, as follows:

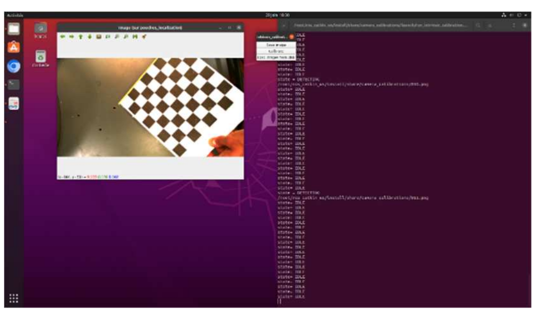
1. Robot control and grasping early tests with suction cup (without detection), during on-site CEA workshop;
2. Adding vision module, set the reference camera/robot, workcell controller & orchestrator (with suction cup), during on-site LMS workshop;
3. Adding dedicated EA gripper (replacing suction cup), during on-site OMNIGRASP workshop.

4.1. AIMEN detection module

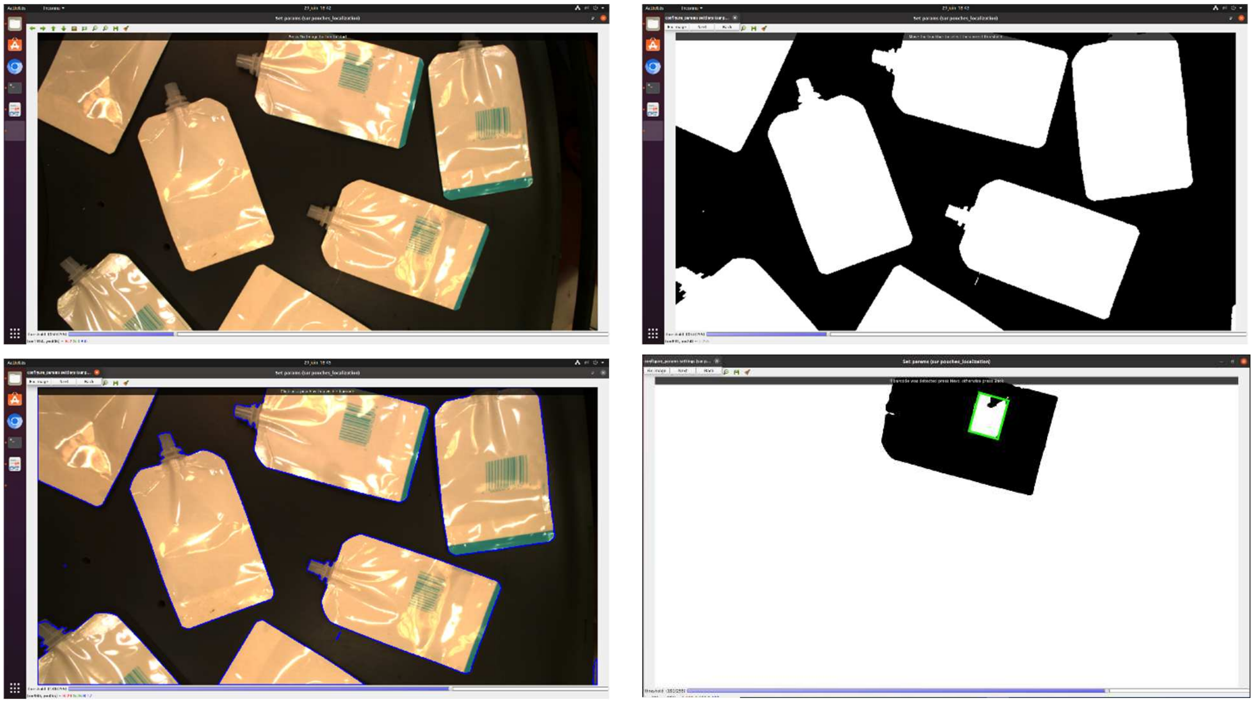
Below are the main steps of implementation and tuning of the AIMEN detection building block developed.

- Setup the detection module: First, we needed to install and connect the camera with the ROS interface. AIMEN sent us the block in a docker format. Once used with the docker environment, we were able to set the parameters for the detection:

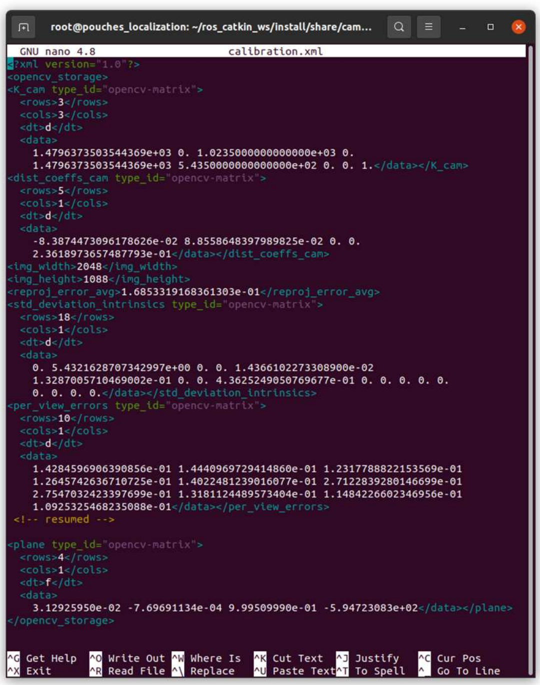
    - First plan calibration with the calibration chessboard :



- Then pouches format and threshold parameters settings:



- Once everything is set, the parameters are saved into a .xml file that will be called each time that the detection module runs:



- Use the detection module: AIMEN provided the procedure to launch the pockets detection using their calibrated module, as described below:

Code lines to access to the container when we boot the PC:

1/ Start container: *docker start 46c5d24326c7*

2/ Make graphic display accessible to the container: *xhost +*

3/ start container & connect to his terminal: *docker exec -it pouches_localization bash*

Now connected to container terminal.

At the first activation of the container, it is necessary to reconfigure the connection with the camera, follow these steps.

1/ Open the software to set camera parameters: *idscameramanager*

2/ If the message at the bottom of the newly opened window indicates that the camera is not configured correctly, you should press: Automatic ETH Configuration

3/ Message is now in yellow & camera statue is now available.

4/ Close idscameramanager windows

Launch folllowing command in the docker :

- roslaunch pouches_localization run_pouches_localization.launch

- rostopic echo /pouches_poses

- rostopic pub /get_capture std_msgs/Bool "data: true"

Each command has to be done on a specific terminal, at least 3 terminals are needed.

In order: (be careful of the order)

1/ launch following command:

- roslaunch pouches_localization run_pouches_localization.launch

This one create MASTER ROS (= roscore) + create node pouches_localization

2/ launch following command:

- rostopic echo /pouches_poses

3/ launch following command:

- rostopic pub /get_capture std_msgs/Bool "data: true"

This command is used to initiate pouch detection and activate the camera upon the first detection. In other words, this command sends a boolean message with the value 'true' to the topic '/get_capture.' The previously executed 'pouches_localization' node subscribes to the topic '/get_capture,' and that's how it receives the pocket detection command.

If everything has been correctly done : We get the locations in the terminal where the command 'rostopic echo /pouches_poses' is executed

CAUTION: Every time the command 'rostopic pub /get_capture std_msgs/Bool "data: true"' is executed, we initiate a pocket detection and retrieve both new and previously published positions on the topic since the creation of the MASTER!

4.2. OMNIGRASP EA gripper

Two versions of the EA gripper prototype developed for the Thimonnier use-case have been successively implemented and tested, in order to reach the demonstrator objective.

4.2.1. V1 prototype

OMNIGRASP provided the procedure to install and launch the EA gripping with their first module version, as described below. This early version had as main objective of validating the software compatibility for communication and control within the workcell.

<u>INSTALLATION</u>
Extract "gripper_package" into your catkin workspace
run catkin command:
cd catkin_workspace
catkin_make
<u>RUN</u>
Set the Arduino port in the launch file (default port is ACM0)
<param name="port" value="/dev/ttyACM0"/>
Run the code.
Verify that the code is executable.
cd catkin_ws/src/gripper_package/scripts/
chmod +x <node_name>
to run pouch gripper:
roslaunch gripper_package thimonnier.launch
<u>CONTROL</u>
Gripper is drived through 2 message /cmd_vel:
rostopic pub /cmd_vel geometry_msgs/Twist “angular.z = 2.0” to turn on HVPS
rostopic pub /cmd_vel geometry_msgs/Twist “angular.z = -2.0” to turn off HVPS

Nevertheless, it appears during the first tests that the V1 prototype was unable to lift the finally selected pouches for the insertion task, even if it was efficient on one kind of pouch without spout. This was due to a limited adhesion force (see paragraphs 3.3., 4.3. and 5. of the deliverable D3.5 ‘Report on gripping technologies - Part 2’, for more details).

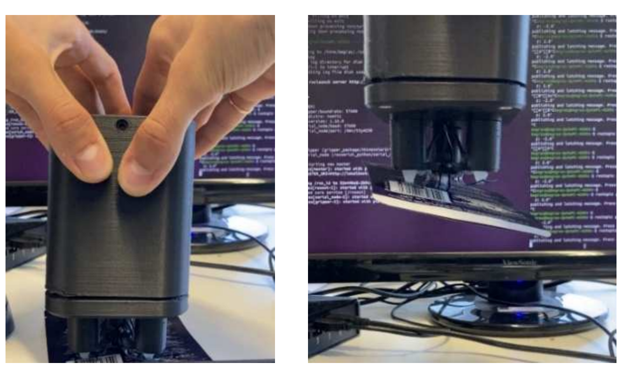


Figure 4: OMNIGRASP EA gripper V1 prototype: gripping pouch model without spout

4.2.2. V2 prototype

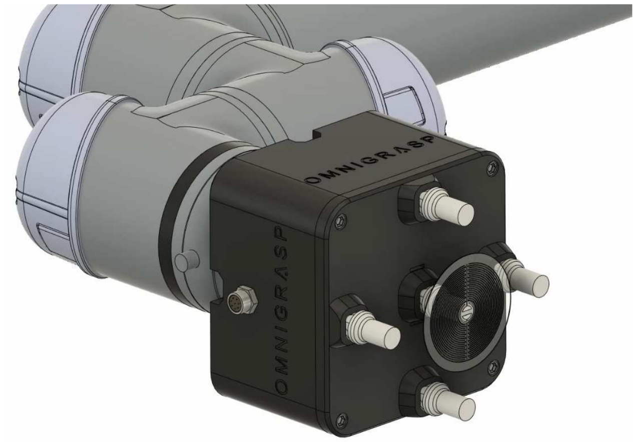


Figure 5: OMNIGRASP EA gripper V2 prototype: CAD view of the gripping side

On the V2 prototype, the High Voltage Power Supply (HVPS) is upgraded (possible to go up to 10kV). We then went through a trial-and-error process to define the best set of parameters: we aimed at using the lowest possible voltage to ensure safe working conditions for the electro-adhesive skins.

At Omnigrasp, with 3kV and 50Hz, we were able to pick every pouch. But at Thimonnier premises, we had to increase the voltage to 5kV to grasp the white PET pouch (pouch we weren't able to test at Omnigrasp). Note that higher voltage (for example, 7kV) means a higher probability of dielectric breakdown, which will break the device.

So now, the new gripper can lift most[2] of the different types of THIMONNIER pouches. The gripper package was slightly modified requiring to reinstall it on the ROS environment of the demonstrator.

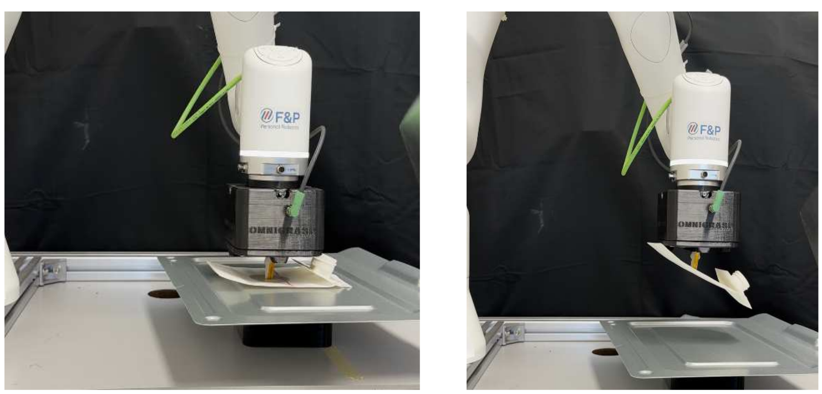


Figure 6: OMNIGRASP EA gripper V2 prototype: lifting a Thimonnier pouch (to be inserted in the rail)

4.3. LMS Workcell controller

During their visit on-site, LMS managed to run an orchestration code that is compatible with the DOHC. A full demonstration process cycle has been implemented for the Thimonnier use case as follows:

[2] Except for a non-woven type of pouches. There were also issue to lift Thimonnier white pouches.

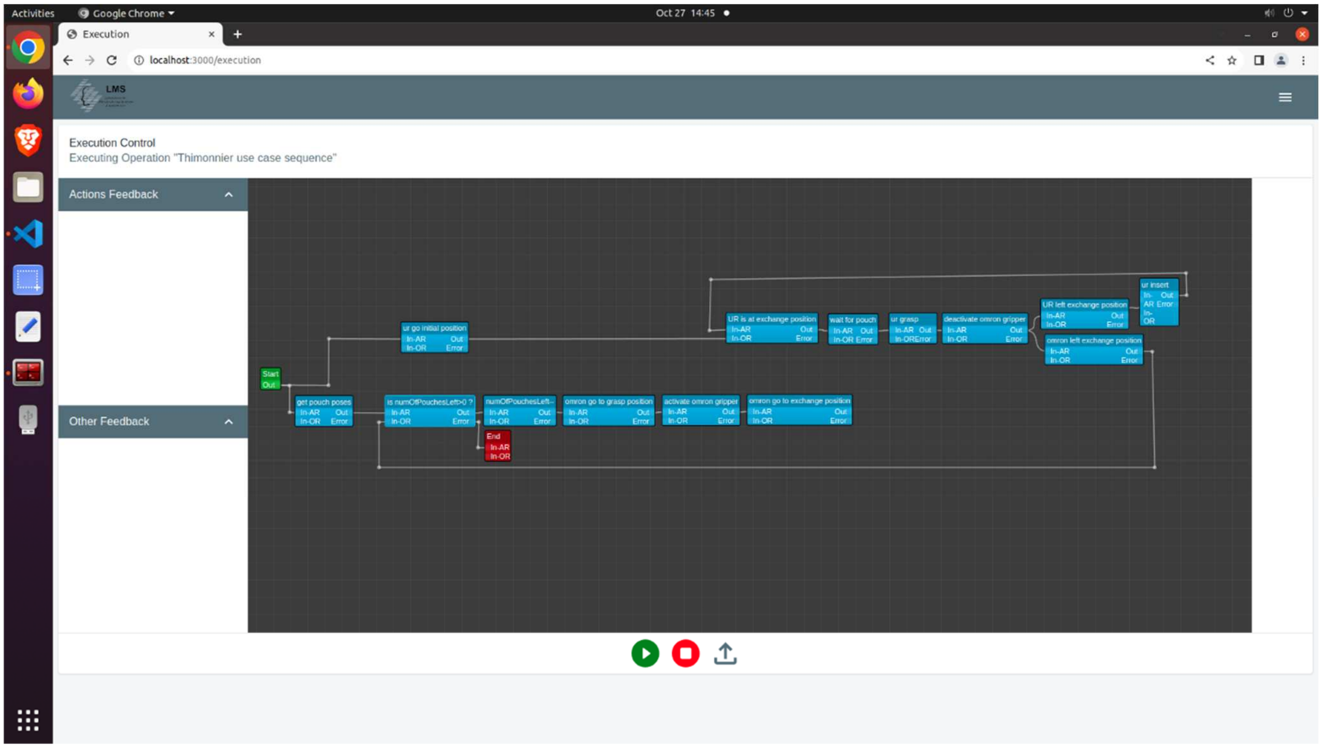
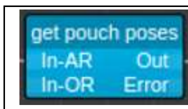
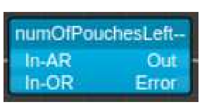


Figure 7: General screen view of the Thimonnier use-case process through the DOHC

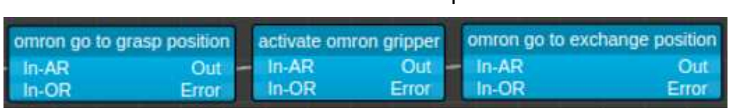


Send message to localization module and get pouch poses.

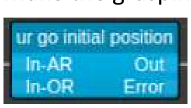
If loop in order to be sure that there are still pouches on the table.



Define the number of pouch that there are remaining on the table.



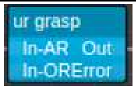
Make the grasping robot going to grasp position, grasp and then going back to exchange position.



Homing position for insert robot.

Make insert robot going to exchange position.

Delay.

	Insert-robot grasping routine. (refer to sequence 07).
	Deactivate gripper of grasping robot.
	Insert routine (refer to sequence 08).

4.4. CEA SPIRE package

Refer to the deliverable D7.2 “Workcell integration – Final version including interfacing” for detailed information.

5. Testing activities and results

This section discusses the achievements of task T8.2 whose objective is the delivery of an integrated solution that addresses the challenges of the Thimonnier use-case. The above-mentioned modules have been integrated towards the successful automation of the process of robotized insertion of bulk empty pouches into a rail (i.e. feeding rail to be installed at the entry of a pouches filling-machine).

5.1. Detection module early evaluation

The vision module has been tested in standalone configuration, before the on-site integration workshops. Firstly, tested with white pouches, everything was functional except two points: barcode detection and pouches overlapped detection.

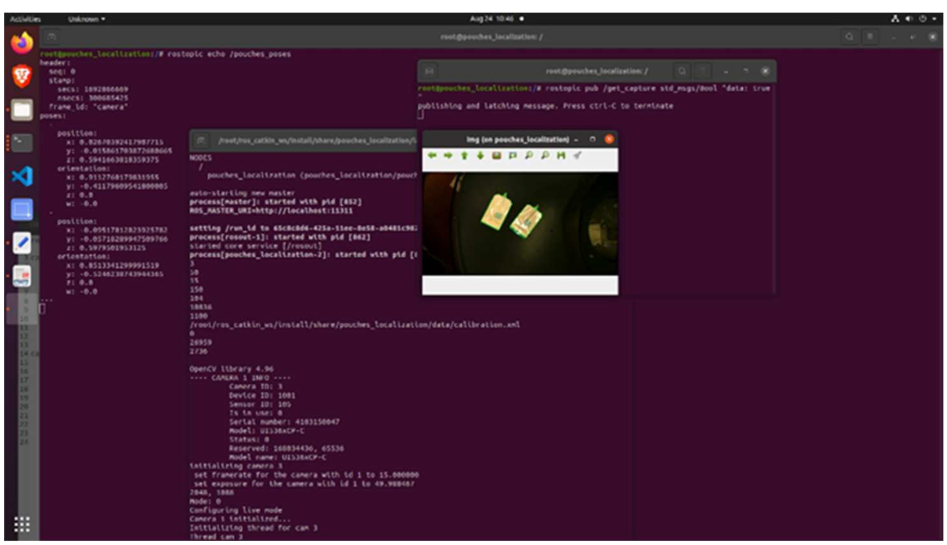


Figure 8 : VISION MODULE: full process of detection

Due to reflect on the pouch, sometimes the detection module is not able to detect the barcode. This leads to a default on the Z axis orientation: sometimes it will be up and sometimes it will be down.

As we can see in Figure 9, the two pouches have a barcode in front of the camera, but it’s not detected.

Figure 9 : VISION MODULE: Default with pouches detection: In red the faulty pouches

Because the vision module is working with threshold contrast definition, if we put pouches in bulk, it won’t be able to detect them independently. As we can see in Figure 10, there are 5 pouches but as is shown on the third picture, the blue outline corresponds to one big pouch whereas there are five.

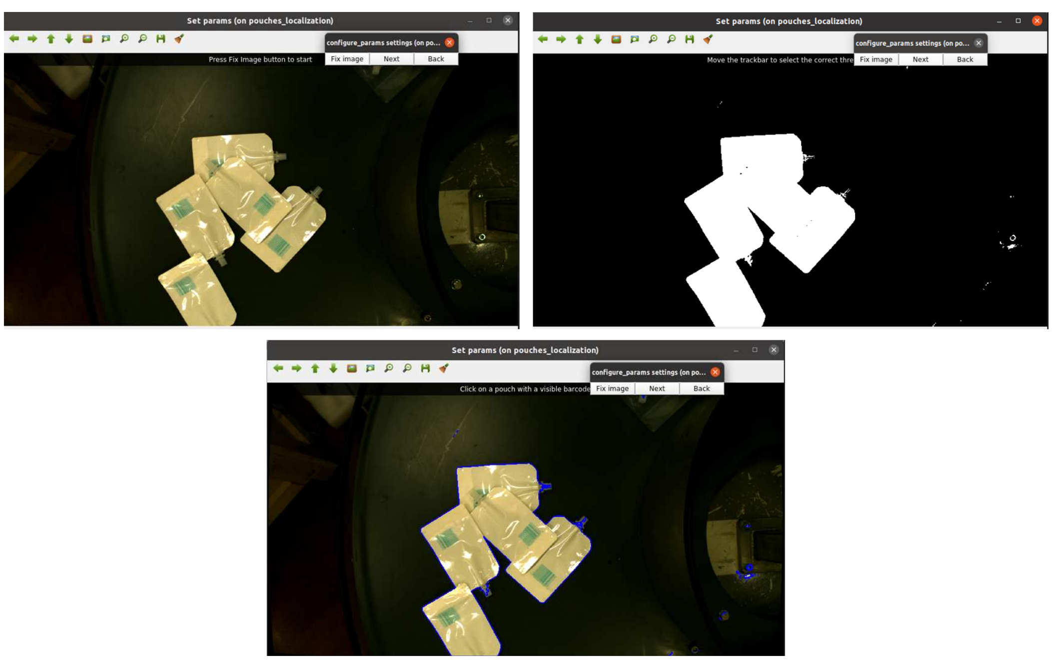


Figure 10 : VISION MODULE: Pouches detection in bulk

Once it was functional with white pouches, it was necessary to set references between OMRON robot and camera.

First, we took 14 pictures of different positions under the camera. We placed the calibration tool of the robot and took robot position for each picture.

	0.6937304180114167 - 0.08264025346320261 - 0.1560196863650965 0.6382895063634829 - 0.21059159981555173 - 0.15798791823732167 0.6875237581029142 0.07684115823750776 - 0.15234252590283903 0.5521341130620112 0.01042183697490813 - 0.15482000590865785 0.5563112995261105 - 0.11938403429954678 - 0.15587534979561796 0.6328821642686157 - 0.19465535856065377 - 0.15698209151347692 0.6902266798064153 - 0.15185197499537711 - 0.15508039106356503 0.7753545519294741 - 0.11383228123764574 - 0.15340276331616204 0.5298023585130027 - 0.07061068578542341 - 0.15621695001860675 0.5620008648842207 - 0.08639641864278674 - 0.15656194851933902 0.6962088717617267 - 1.093395510083639e-05 - 0.15278599469276055 0.6767736097098783 - 0.23204318907605215 - 0.15685896563835544 0.7572245878245254 - 0.25953146501921587 - 0.15610514331157732 0.8458980511353182 - 0.1737868738755115 - 0.15416708413227598 0.7360707201002944 0.05848003807245589 - 0.151465500375891 3

Then, we performed on a translation matrix:

<u>3x3 rotation matrix</u>

6.2905876040455266e-01 / 7.7730583081818283e-01 / 8.9845052363405065e-03

7.7733773050865951e-01 / -6.2908349523833129e-01 / -9.3522036683512961e-05

5.5793087326400873e-03 / 7.0428257666267369e-03 / -9.9995963414494238e-01

<u>3x1 translation matrix</u>

7.5235494101978463e+02 / -1.1389133806126289e+02 / 4.5114801745133076e+02

<u>quaternion x y z qx qy qz qw (xyz in milimeters)</u>

7.5235494101978463e+02 / -1.1389133806126289e+02 / 4.5114801745133076e+02 / 0.9025107 / 0.4306441 / 0.0040342 / 0.0019768

Last issue is when we tried to change the pouch format, we had the following issue: different pouches are not detected. Calibration process seems to be working, but due to the fact that there is no barcode on the pouches, it doesn't get a pouch pose. To continue working with different kind of pouches, we did a substitution mode that always goes on 3 fixed positions (always the same ones).

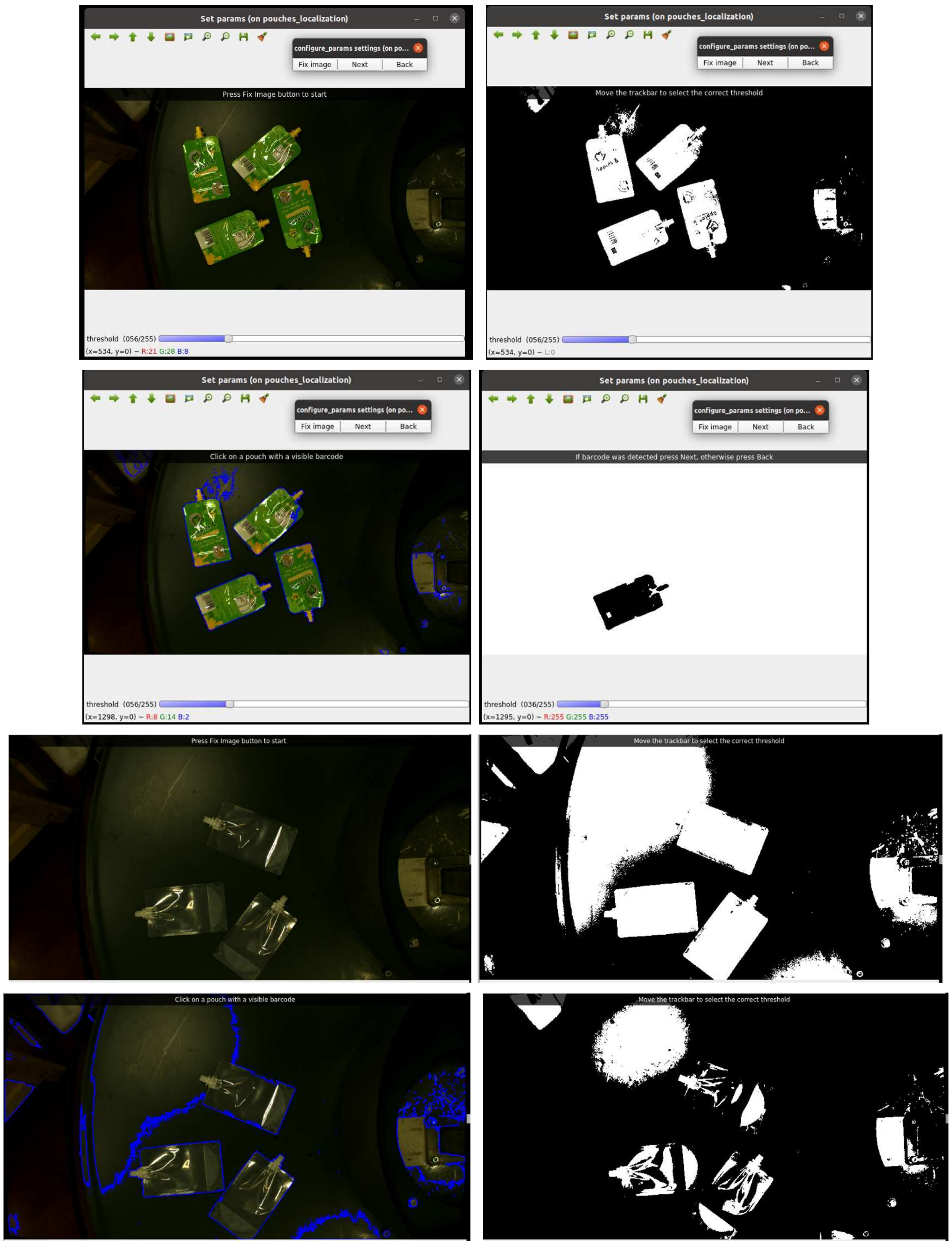


Figure 11 : Detection failure with different kind of pouches

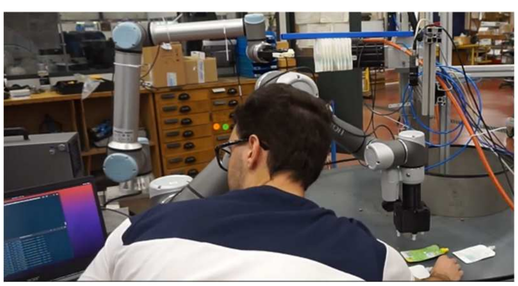
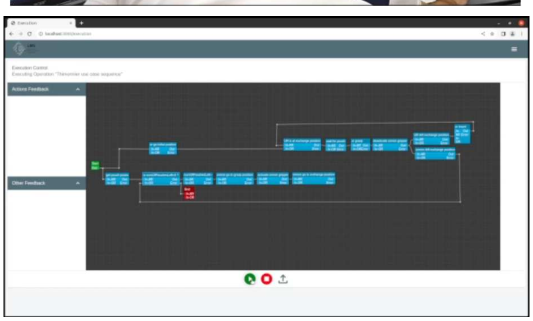
5.2. Demonstrator general process evaluation

Due to this limitation, the general sequence of the demonstration is as given in Table 1.

Table 1. Handling strategies evaluated in the integrated demonstration.

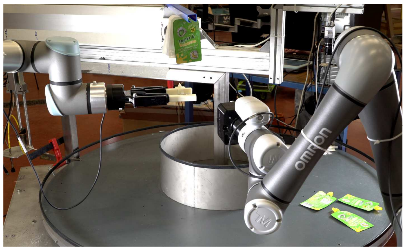
Sequence number	Name of activity
01	Human set pouches in position (no pouches overlaps)
02	Two robots and grippers homing
03	Pouches detection
04	Grasping robot goes to grasp position
05	EA gripper ON
06	Grasping robot goes to exchange position
07	Insertion robot takes pouch
08	Insertion robot inserts pouch into the rail

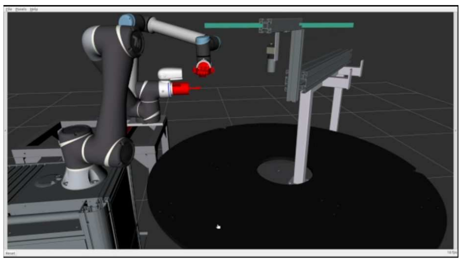
Below is the description of the Thimonnier demonstration reporting video.

<table>
<tr><td>Sequence: 01</td><td colspan="2">Human set pouches in position (no pouches overlaps)</td></tr>
<tr><td colspan="2">
</td><td><u>Setting the process cycle:</u>

• The operator puts pouches on the table, but not in bulk. As mentioned before, the AIMEN vision module will not be able to detect the pouches when they are overlapping each other.

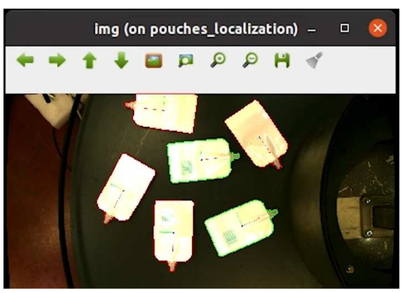
• Then launch the cycle on the workcell controller.</td></tr>
</table>

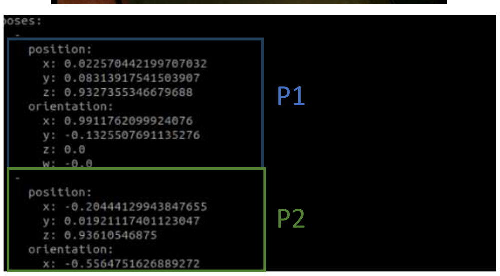
<table>
<tr><td>Sequence: 02</td><td colspan="2">Two robots and grippers homing</td></tr>
<tr><td colspan="2">



</td><td>

<u>Robot homing:</u>

- Both robots will go to homing position, and the insertion gripper will execute a calibration test.
- On another hand the Rviz digital twin will show on the screen the robots move.

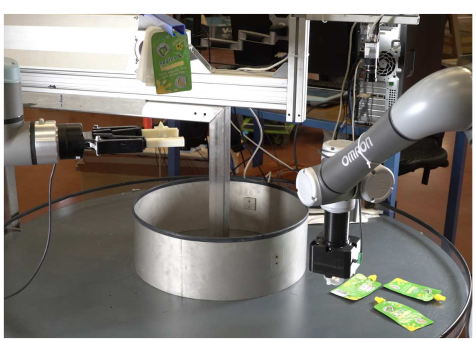
</td></tr>
<tr><td>Sequence: 03</td><td colspan="2">Pouches detection</td></tr>
<tr><td colspan="2">




</td><td>

<u>AIMEN Vision module:</u>

- The AIMEN vision module is started by the DOHC.
- It determines x,y,z and qx qy qz, as we can see in the picture on the left with a small reference set green, red and blue (x,y,z). qx, qy, qz correspond to the rotation of the detected pouch. Then it send it to ROS.
- Vision system module determines the number of pouches and their localization, then send it to the orchestrator DOHC.
- The ordering number of pouches is determined by the detection module (picture below)

</td></tr>
<tr><td>Sequence: 04</td><td colspan="2">Grasping robot goes to grasp position</td></tr>
<tr><td colspan="2">


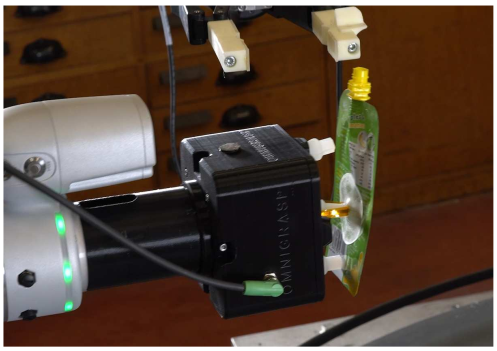
</td><td>

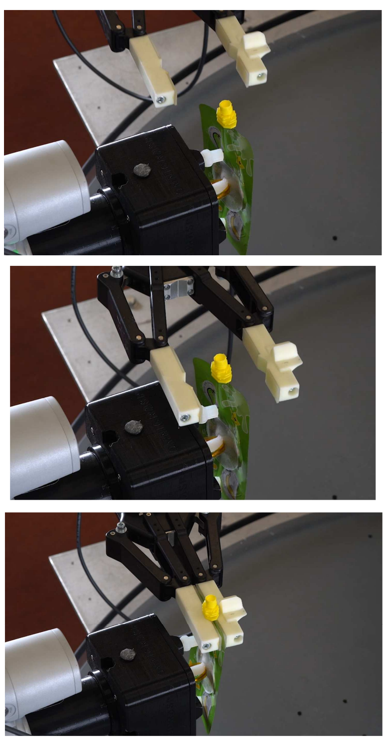
<u>Robots moves:</u>

- The grasping robot goes to the position sent by the localization module (following the pouch order of detection as described above) and orchestrated with DOHC.

</td></tr>
</table>

<table>
<tr><td>Sequence: 05</td><td>EA gripper ON</td></tr>
<tr><td></td><td><u>EA gripper ON:</u>

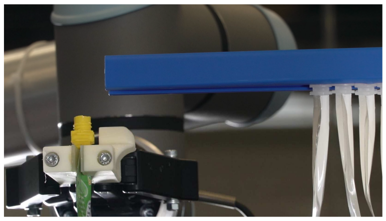
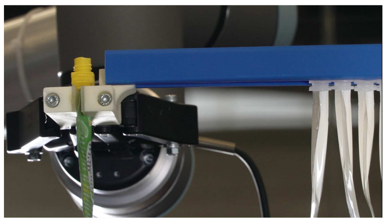
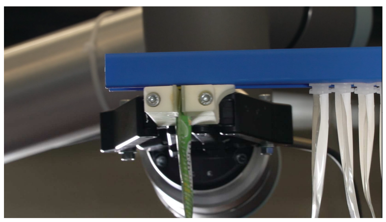
• Once the robot is in a grasping position, the orchestrator sends a command to set the HVPS on. Consequently, the EA pad sticks to the pouch that we want to lift.</td></tr>
<tr><td>Sequence: 06</td><td>Grasping robot goes to exchange position</td></tr>
<tr><td></td><td><u>Robots moves:</u>

• The grasping robot will go to the exchange position sent by orchestrator with DOHC.</td></tr>
<tr><td>Sequence: 07</td><td>Insertion robot takes pouche</td></tr>
<tr><td></td><td><u>Exchange:</u>

• Once grasping robot is in exchange position, the insertion robot opens the dedicated insertion gripper.

• Then the insertion robot moves forward to the pouch target point defined in the spire package.

• The gripper slightly closes, and the insertion robot goes up. Like that, the pouch tip will be blocked on the reference surface of the CEA patented gripper jaw and will be able to adjust the clearance. On the same time the electro-adhesive gripper is set OFF (thus releasing the pouch).</td></tr>
</table>

Sequence: 08	Insertion robot inserts pouch into the rail
  	**<u>Inserting phase:</u>** • The insertion robot holds the pouch into its dedicated gripper. It goes to the insertion start position. • Then goes up to insert the dedicated gripper jaw reference surface into the rail, taking advantage of its flexibility. • Once the reference surface of the dedicated gripper jaw and the bottom of the rail are in contact, the robot has only to move forward along the rail main axis to insert the pouch to its final position.

6. Conclusion and next steps

This report describes the video demonstration deliverable that concludes the activities of task T8.2. The report discusses the enabling technologies and the functionalities they bring to the integrated solution. A representative set of sequences was used to validate the envisioned solution. Next steps involve testing of the solution with technical experts and operators (see deliverable D8.4 'Report on MERGING robotic technologies assessment').