WP5 – Multi-level perception system for environment understanding

D5.3. Perception functionalities demonstration

Due date: M41 - 31/03/2023

Version number: V2

Responsible partner

AIMEN : D. Pérez Losada, A. Costas López, A. M. Pertusa Llopis, S. Muíños Landín, J. Masood

Contributing partners

EPFL : E. Judd

Summary

This document summarizes the multi-level perception system assessment. It describes the procedures used to validate the technologies involved in the perception system as described in D5.2, centred on perception for human robot interaction and perception for grasping and manipulation. The perception system for human robot interaction based on non-wearables devices includes the person detection and tracking system, the workers movement decomposition and the gesture recognition systems. The assessment related to process for the given use-cases was centred on the application of the perception technologies adapted to the considered processes. It includes fabric detection and planning for wrinkles removal for SELMARK, pose extraction for different types of pouches in the THIMONNIER use-case, continuous monitoring of fabric deformation in the VDL use-case and continuous gripping monitoring system to be included in all the processes once integrated in the gripper. It is important to note that the withdrawal of the gripper developer and integrator has limited content and scope of this early version of the deliverable. The force and torque sensing have not been tackled for the proposed industrial processes and it will be updated in a final version of this document. It is also important to note that the feedback from the electro-adhesion device was developed and tested in a laboratory setup, not integrated in the final handling device.

Table of content

1. Introduction

After the development of the technologies presented in D5.2, it is necessary to carry out their assessment. In this document we present the demonstration procedures to validate these technologies in laboratory setup simulating the realistic environments of Merging Use-Cases. This demonstration is organized as follows: we first present the validation of the technologies for the environment recognition, which will be followed by the validations of the technologies related to process monitoring and assessment of the soft force sensor technology.

2. Environment recognition

2.1. Human detection and tracking system

As established in D5.2, the developed human tracking system uses stereo images to detect and track people in the workshop. To validate the output of this system, it was decided to use a map of the workshop in which the localization data returned by the tracking system can be represented. This representation can give an idea about the precision of the system.

The map was created with a mobile robot equipped with two laser scanners and was localized regarding one of the fixed cameras in the working environment (camera-map calibration). In Figure 1, it can be seen the map of the workshop with an approximate position of the cameras (CAM1 and CAM2) as well as the monitored area highlighted in yellow (intersection of the field of views of both cameras). The cameras must be positioned to ensure the optimal overlapping between the field of view of every camera to minimize the dead zones. With this configuration, the system can monitor an area of approximately 60m^2 (10m x 6m) which is difficult to achieve by other type of sensors like 3D cameras. Of course, the precision of the detection is not the same in the nearby areas than in the more distant ones due to the angular nature of the optic (focal length of 8mm), but even for distances of 10m, the resolution for each camera is 7mm per pixel that can be enough for having a good precision in the detection of people. So, the precision of the system will depend more on the precision of the YOLO (You Only Look Once) detection itself and the precision of the stereo calibration. Person detection and tracking is based on YOLO object detection in real-time.

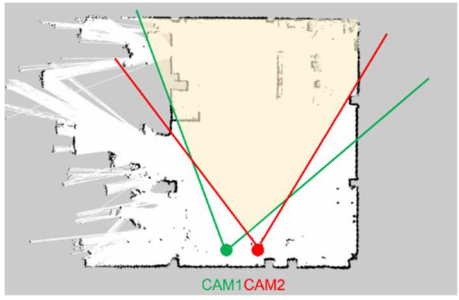


Figure 1. Map of the workshop and monitored area.

To validate the performance of the system, different tests were carried out acquiring from the cameras at 13 fps (frames per second) including the presence of several operators, wearing uniform. In this document we present the three tests based on the number of operators in the working area. In cases where there are more than one operator, special focus was put on the occlusions caused by different operators crossing each other.

Single Operator Detection and Tracking Scenario: In Figure 2 the result for the tracking of single operator can be seen. In the two images on the left, it is shown the output of the tracking system represented onto the 2D images, while in the images on the right it is shown the output of the system in 3D over the map of the workshop. The tracking system was able to precisely detect and track the single operator. Hence, we

can conclude that the performance of the developed system for single operator is good, since it only depends on the good performance of the YOLO. This was the simplest testing scenario.

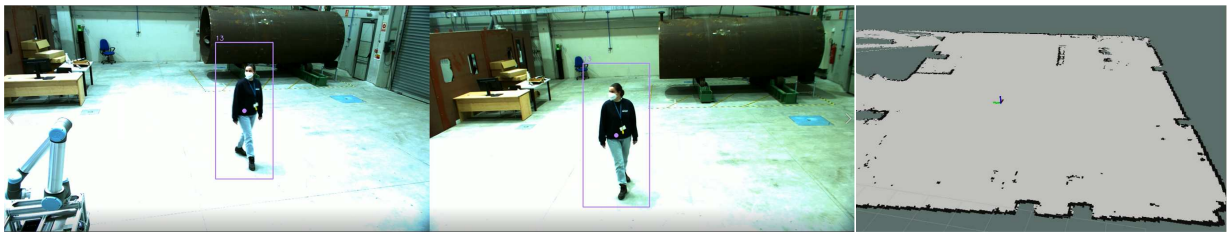


Figure 2. Tracking of one operator: tracking output in 2D images (left); 3D representation on a map (right).

Two Operators Detection and Tracking Scenario: The system demonstrates a good performance with two operators in the working area as shown in Figure 3. The system also demonstrated the robustness against the occlusions for multiple operator in the working area as shown in Figure 4 and Figure 5. A video showing the real performance could be download from the following link tracking_system.mp4. The system is capable of relocating a worker after a detection loss keeping the correspondence with previous detections.

Figure 3. Tracking of two operators.

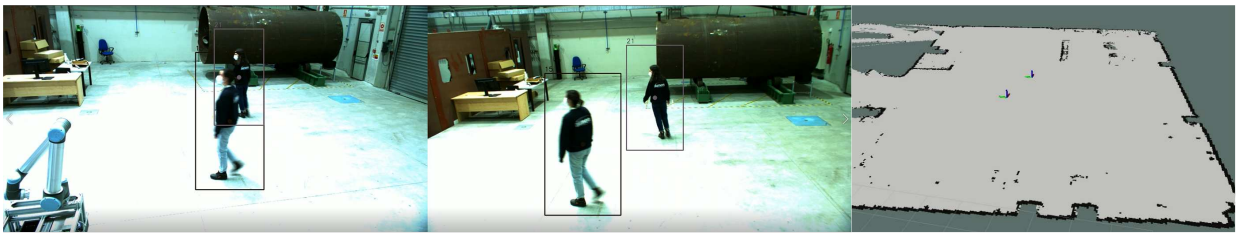


Figure 4. Result of tracking with partial occlusion in one of the cameras.

Figure 5. Result of tracking with multiple operators bounding boxes overlaps.

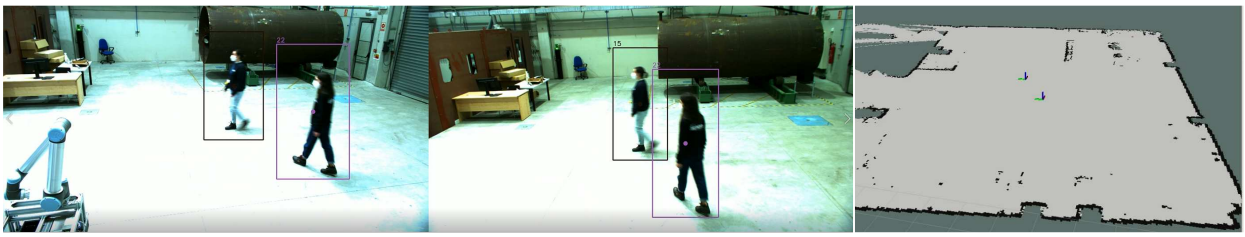


Figure 6. Good performance of the tracking after a cross of the operators.

Three Operators Detection and Tracking Scenario: Similarly, the single and double operator the system demonstrate a robustness for three operators in the working area wearing different uniforms for the case without occlusions as shown in Figure 7. However, we detected the loss of tracking of two operators due to the occlusion in the right camera where two operators were crosses as shown in Figure 8. We observed that after the occlusion the tracking system recuperate with its inherent performance as shown in Figure 9. Finally, the overlapping of the bounding boxes yield good performance as shown in the Figure 10.

Figure 7. Tracking of three operators.

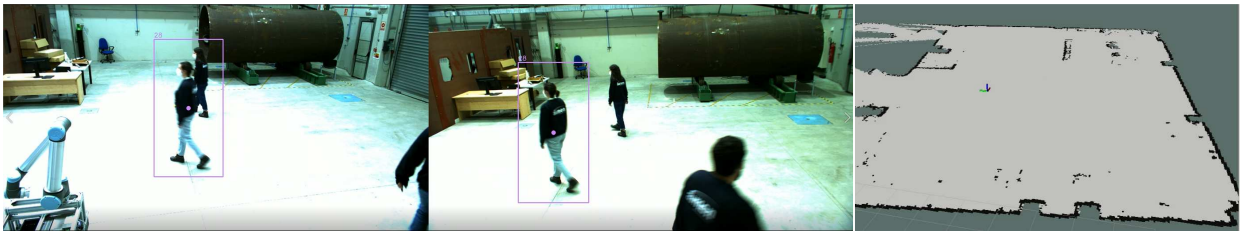


Figure 8. Loss of tracking of two operators due to an occlusion in one of the cameras due to a cross.

Figure 9. Recuperation of the tracking of three operators after an occlusion.

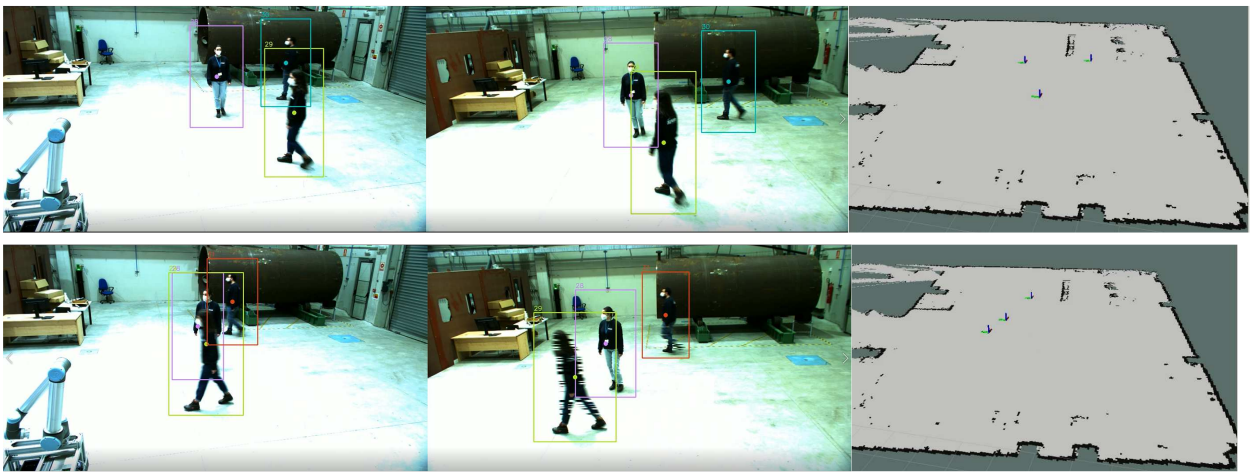


Figure 10. Good performance of the tracking system with overlaps.

2.2. Movement decomposition with OpenPose

In the same way as for the tracking system validation, for the movement decomposition system the map of the workshop was used. The movement decomposition system, as was established in D5.2, is based on a neural network called Openpose, capable of detecting important points of a human body (joints). In order to represent these joints in Rviz (ROS visualisation), it was implemented a module of representation that allows to represent each joint as a ball as it is shown in Figure 11 and Figure 12. For the movement decomposition with OpenPose we designed two test scenarios similar to the detection and tracking which were based on the number of operators in the working area.

Figure 11. Movement decomposition system output: result of OpenPose in 2D images previously filtered by YOLO neural network (left); Output represented in Rviz in 3D (right).

Figure 12. Movement decomposition system output: result of Openpose in 2D images previously filtered by YOLO neural network (left); Output represented in Rviz in 3D (right).

Single Operator Movement decomposition: During the validation, it was observed that the result of this module was good when the detection with Openpose was also good (**Erreur ! Source du renvoi introuvable.**). However, some weak points of the Openpose network were detected with the tests:

- Lack of repeatability
- Bad performance if the clothes colour is similar to the background colour
- Bad performance when there exist overlapping between operators
- Quality of detection decrease when the operator is in profile

All these points contribute to the system to be not very robust causing a bad result of the movement decomposition (**Erreur ! Source du renvoi introuvable.**) even when some filters are used to improve the repeatability. Further filtering and retraining of the neural network will be necessary for the deployment in real scenarios to ensure having a good repeatability. The real performance can be checked in the following video movement_decomposition_system.mp4. **Erreur ! Source du renvoi introuvable.** shows the person decomposition problem when the clothes and the background colour is same. We observed

that the skeletal disappear in the left-hand side of the **Erreur ! Source du renvoi introuvable.**, which pose a limitation of the technology and provide an avenue of future research.

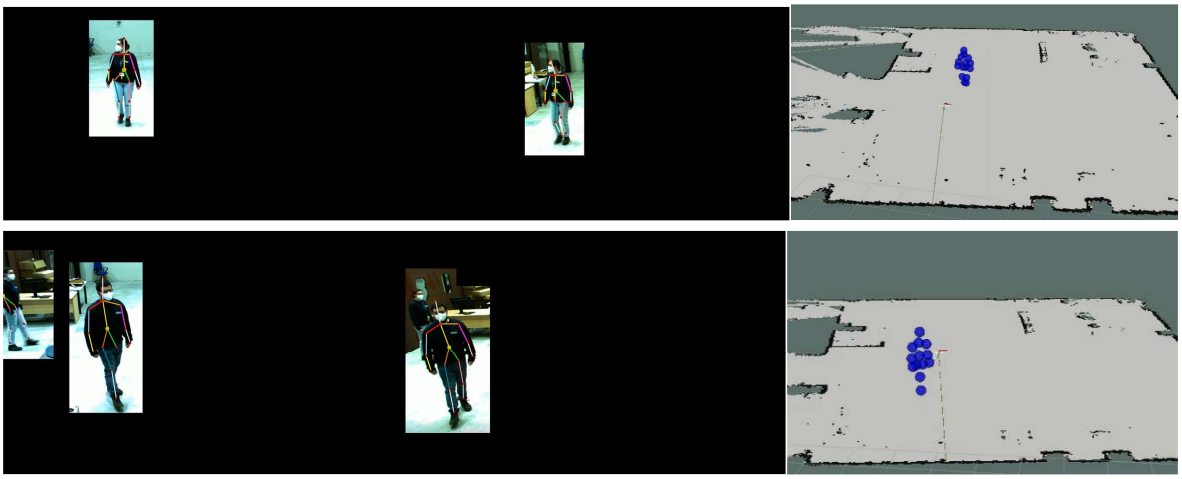


Figure 13. Good results of the movement decomposition system for single operator

Figure 14. Bad results due to the similarities between the background and the clothes color. The image segmentation is not able to isolate relevant features to identify the workers.

Two Operators Movement Decomposition: Similarly, the results of the movement decomposition system for two operator wearing different cloths in the working area are good. However, we observed the problem when the operator cloth matches with the background colour (Figure 16) or the camera experience occlusion (Figure 17) due to operator crossing. It results in situations where the segmentation from features identification used in OpenPose is not able to extract any relevant information, and so it is not able to give valid positioning result. Figure 17 shows to the tracking and decomposition problem when a worker is occluded by other, not being possible to track their movements with one of the cameras. In this case, the system keeps interpolating the movement considering the decomposition from one of the cameras and the last valid detection from the other one. These are the major limitations of the technology.

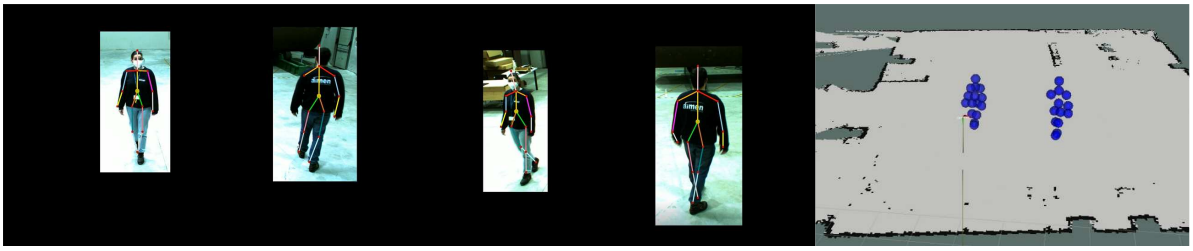


Figure 15. Good results of the movement decomposition system.

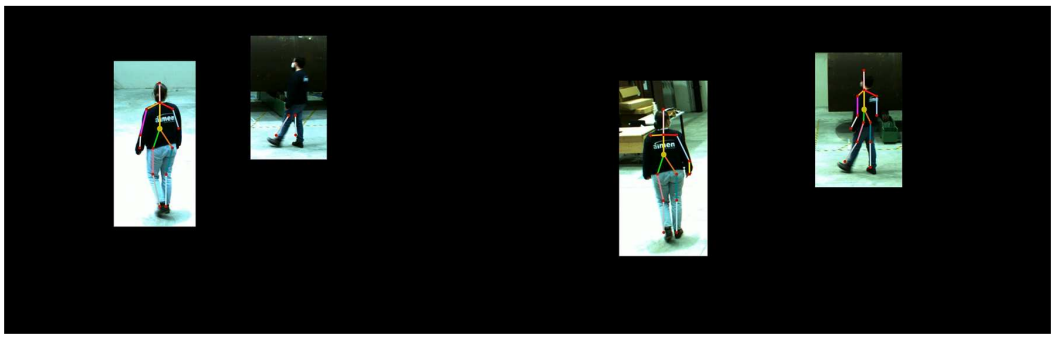


Figure 16. Bad results due to the similarities between the background and the clothes color. The image segmentation is not able to isolate relevant features to identify the workers.

Figure 17. Bad result due to an overlapping between operator in one of the cameras

2.3. Gesture recognition

The gesture recognition system was validated in the same working environment than the tracking and movement decomposition system. It should be noted that the performance of this module depends directly on the performance of the movement decomposition module, since it takes as input the data provided by it. For this reason, the validation was done with the single operator in a zone in which the background colour is different than the clothes colour. In these conditions, the system was calibrated to recognize the gestures of the Figure 18 taking several images for each gesture. Detection if very good for several gestures and keeping the number of them low it precludes the possibility of having false positives. The validation video can be found in the next link gesture_recognition_new.mp4.

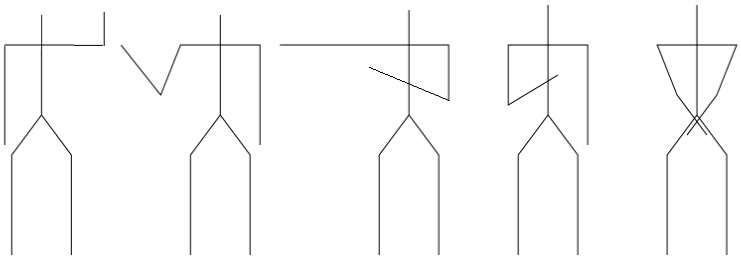
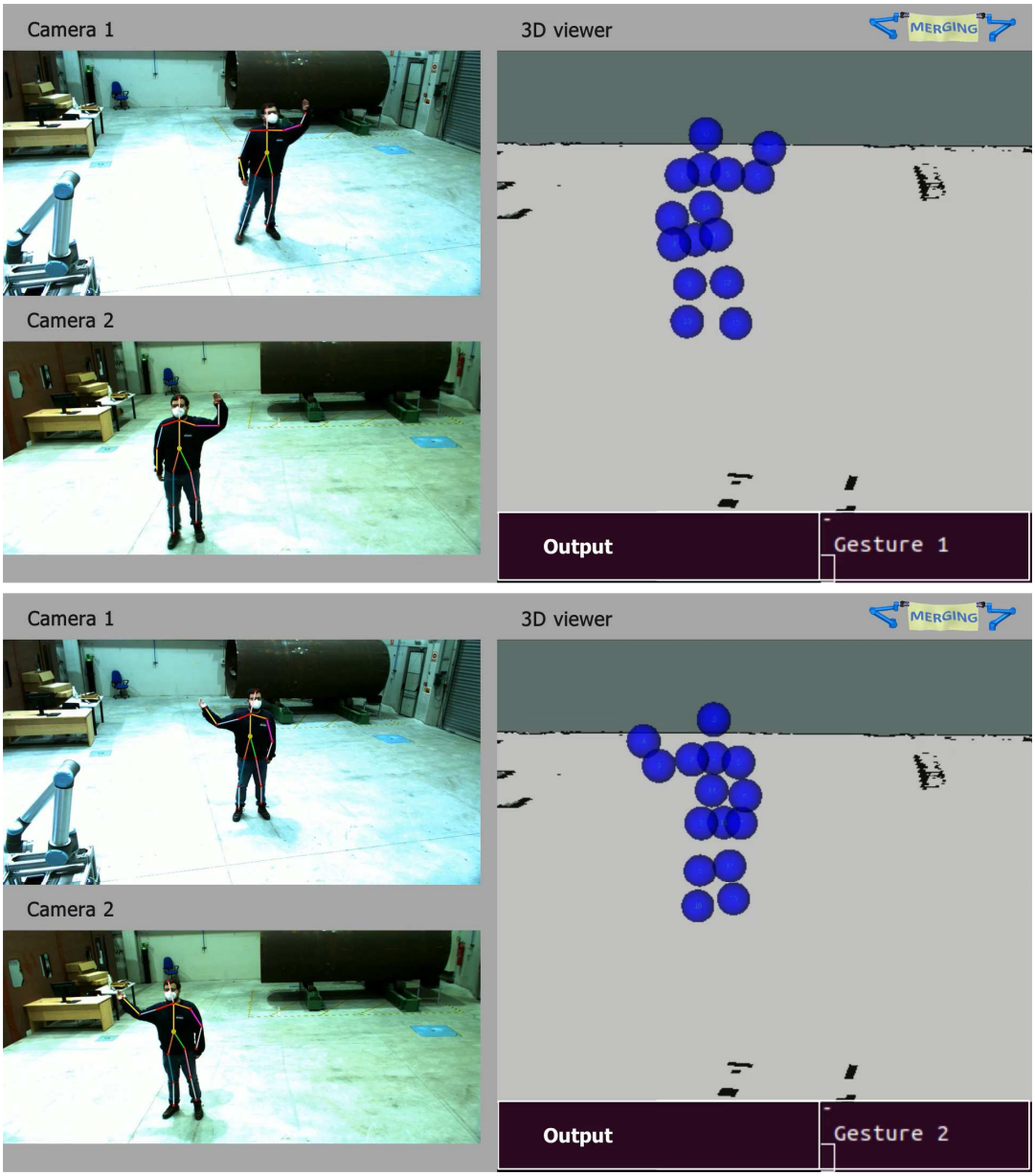
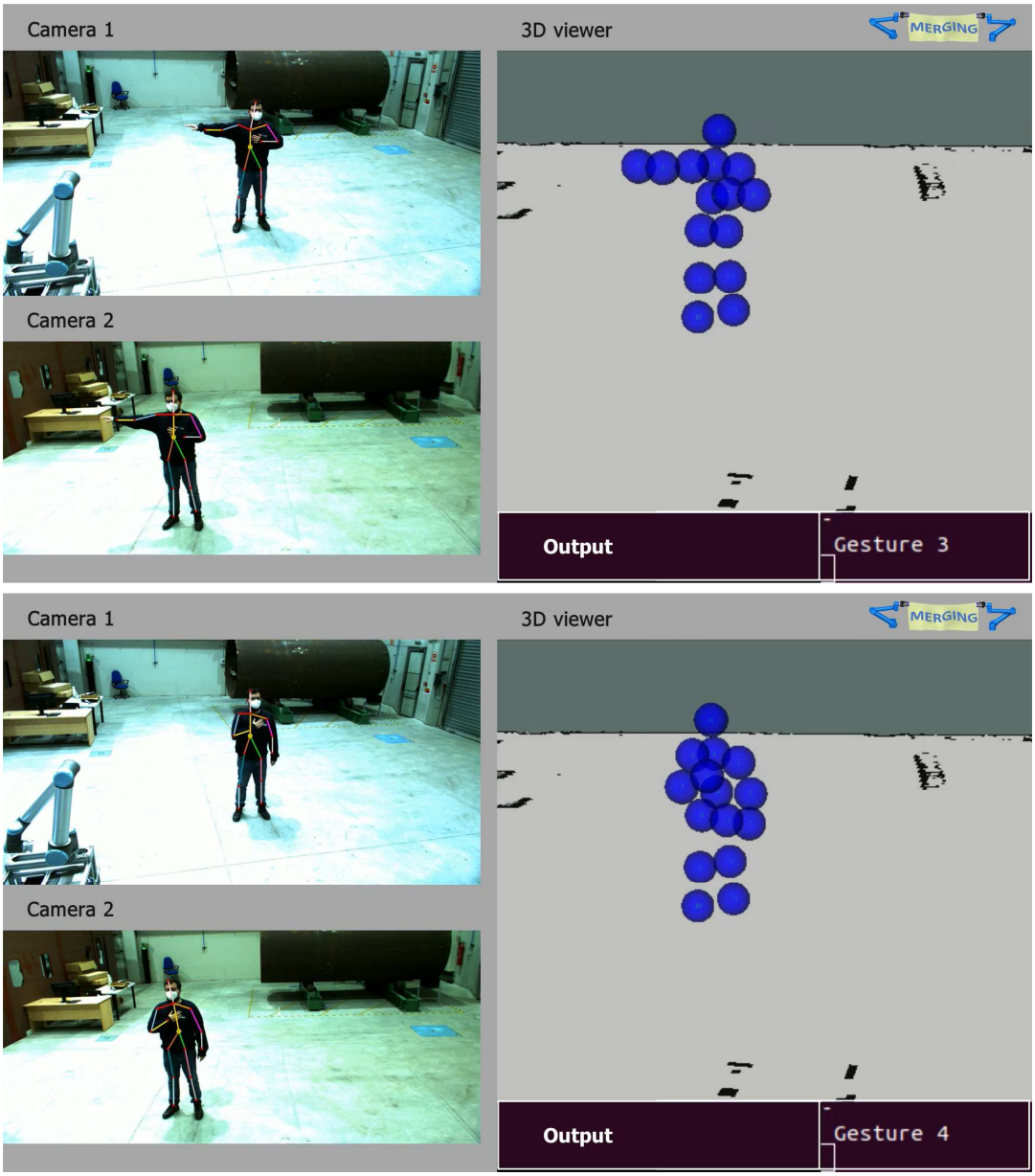


Figure 18. Gestures to be recognized by the gesture recognition system.

For five gestures we have a detection rate over 97% considering a good movement decomposition. With workers occlusions or more dynamic environment the results decrease in the detection quality but without having false positives, it only fails to detect any of the gestures. In the following images, the result of the gesture recognition module is shown:




Camera 1
3D viewer
MERGING
Camera 2
Output
Gesture 3
Camera 1
3D viewer
MERGING
Camera 2
Output
Gesture 4

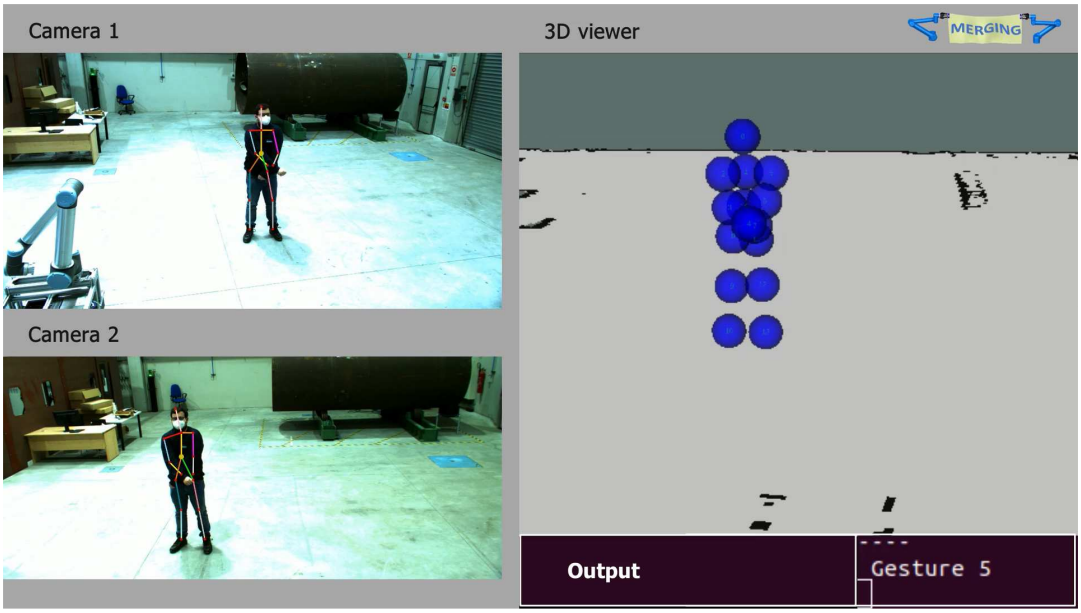


Figure 19. Result of test of the gesture recognition system.

3. Process monitoring

3.1. SELMARK use case

In the SELMARK use case, the perception system must be able to detect the fabric and the wrinkles on it to be smoothed by the robots. For this reason, it was decided to use both technologies, 2D and 3D vision, because the first works better for segmentation (distinguish fabric from background) and second is needed to detect the wrinkles and their size. In the following section, the validation for the fabric segmentation algorithm is presented while the validation of the wrinkles detection and the action to remove them is included in section 3.1.2.

3.1.1. Fabric segmentation

To assess the technology developed for the fabric segmentation in the SELMARK use case, AIMEN have prepared the following setup (Figure 20): ABB robot equipped with a MotionCam3D and a 2MPx RGB camera and a table covered with a white Nomex on which the fabric will be placed. It must be highlighted that we have used Nomex material as background in these tests to establish similar conditions to the real case at SELMARK facilities. Currently, this Nomex material is being used only inside the presses, but it is feasible to have the same background to homogenise the perception system performance. To obtain captures with significant information with the MotionCam3D, an optical aperture of 2.8 mm is set. Then, to acquire wrinkles shape, laser wavelength is configured in 2500-3000 nm depending on environment luminous conditions. Concerning MotionCam3D resolution, it was set to the highest mode (1600x1200) in order to obtain more points to process. Then, another crucial point was the scanning distance. In this case, it was established to 600 mm, which is the minimum scanning distance recommended by the user manual. Regarding the software, it was used a system equipped with Ubuntu 18, PCL 1.7 and OpenCV4.2.

Figure 20. Setup for segmentation algorithm validation.

Considering this setup, the fabric segmentation algorithm was verified for several types of fabrics, as it was already shown in D5.2, and it brought good result in almost all the cases, except for the case of the white fabric (Figure 21). The crucial problem found when scanning the white fabric on white Nomex was its difficulty to segment data due to the colour similarities between the background and the fabric.

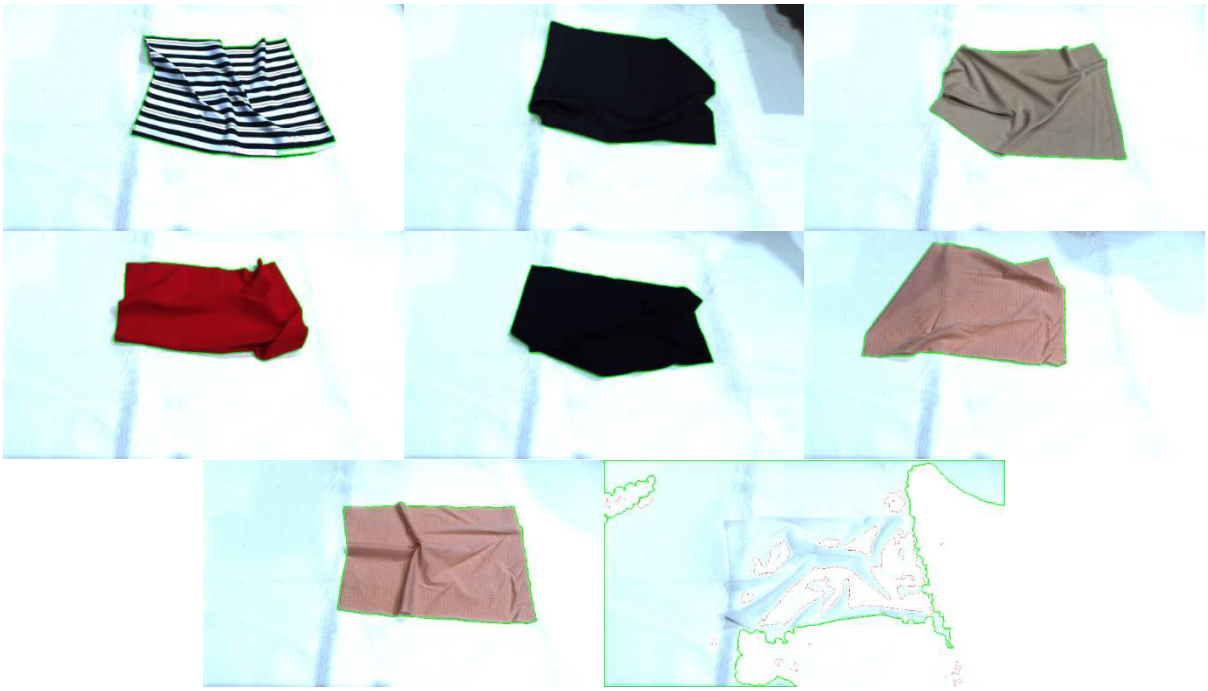


Figure 21. Result of the segmentation algorithm for several types of fabric: the fabric is correctly segmented except for the white one.

In order to solve this issue, SELMARK provided a dark green Nomex that was used to add colour contrast between the captured elements in the scene. With this change in the background, the acquisition problem was solved (Figure 22), providing more information of fabric points, being able to distinguish almost all wrinkles and codify all required data to use them as input to the neural network to obtain the wrinkle-removal action. An aspect that must be pointed out is that program configuration parameters do not have to be modified to obtain expected results.

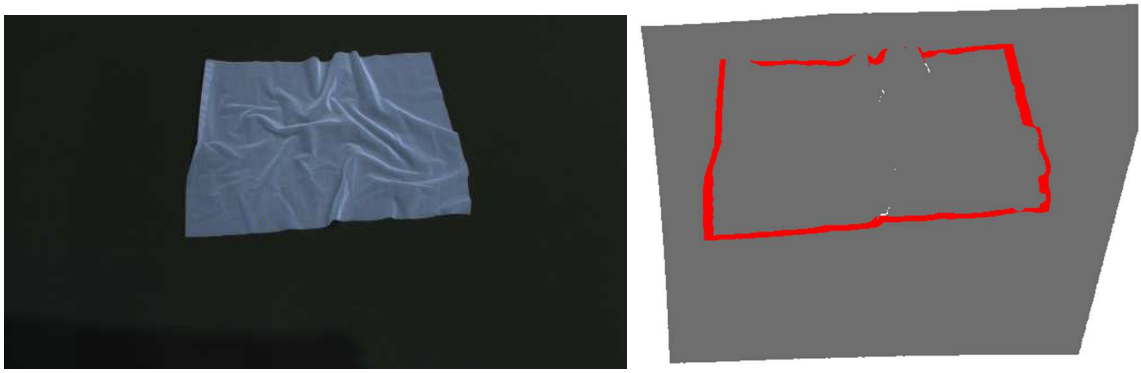


Figure 22. New background of dark Nomex for white-translucid fabric (left); Contour segmentation obtained.

Moreover, this algorithm was verified in the case of the detection of a stack of fabrics. Its performance turned out to be good as it can be seen in Figure 23. It is possible to detect the pile and the upper fabrics, but the resolution is not enough to isolate the identification of every fabric. For the given use-case, it is possible to ensure the flatness of the upper layer on the pile, being enough for the proposed manipulation.

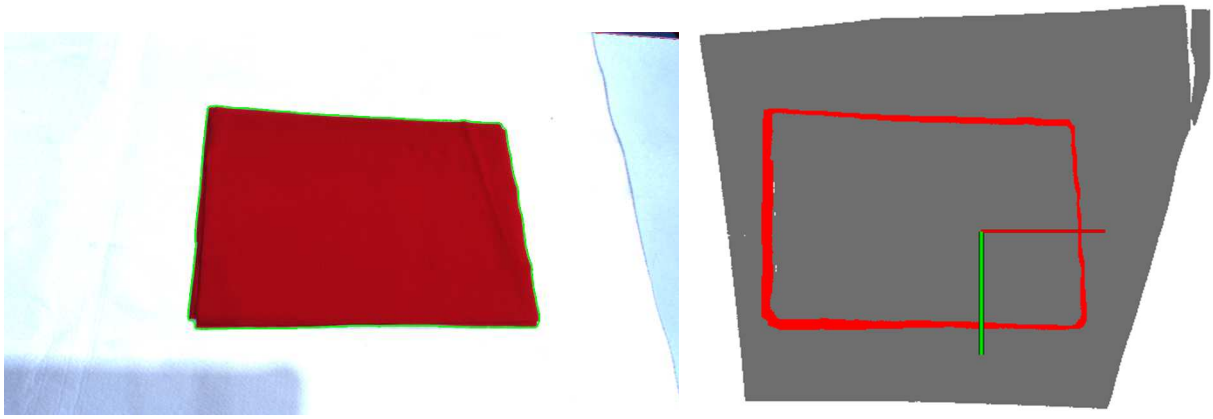
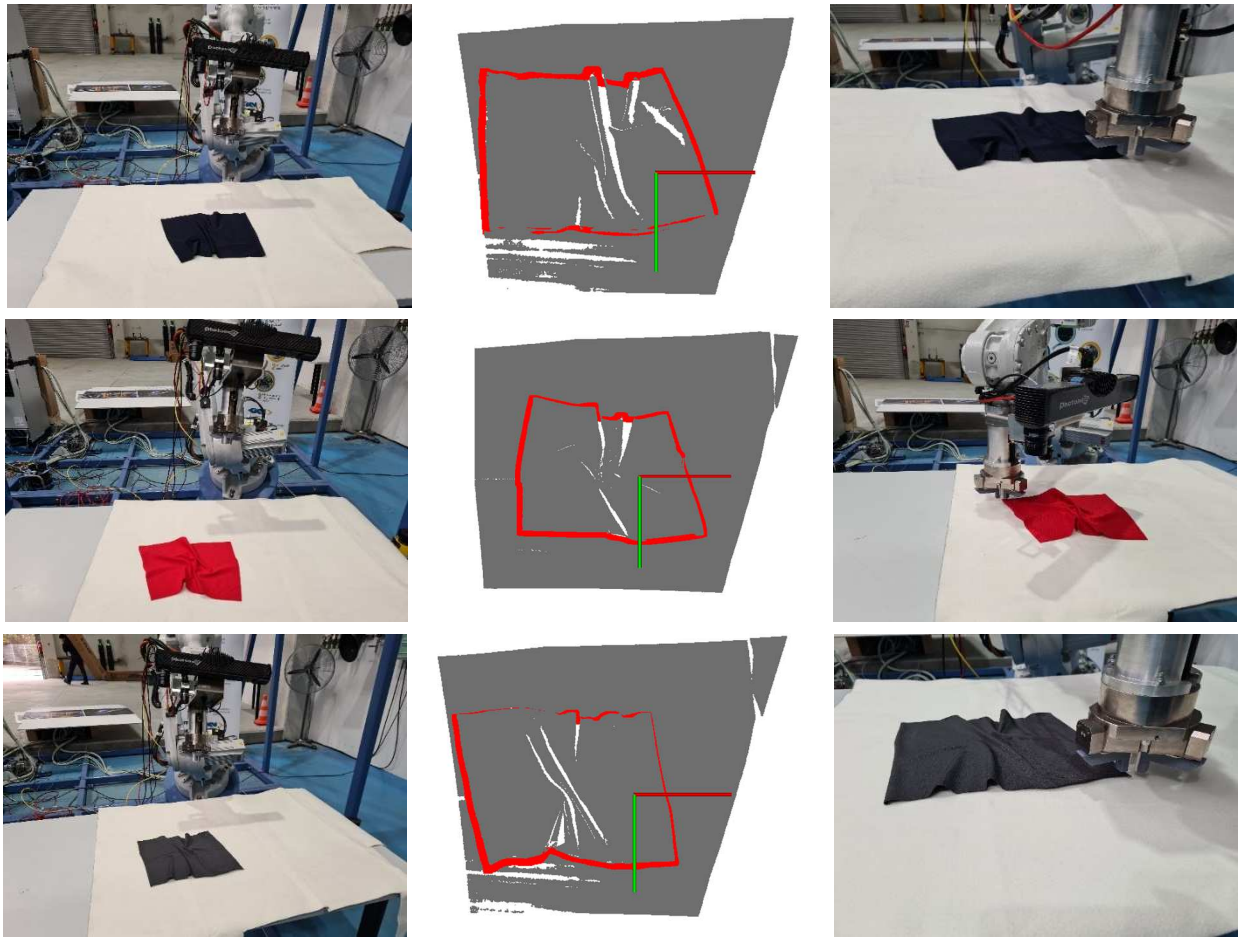


Figure 23. Segmentation of fabric stack (left); Contour in 3D (no-flat detection) of the fabric stack.

Once the performance of the algorithm was validated for all the types of fabric, it must be verified that the perception system is well calibrated with the robot. To do that, the fabric was scanned, the system detects the corners and the robot tool is sent to these detected corners. In the following images it can be seen that the robot goes to the detected corner correctly, so the whole system is well calibrated.



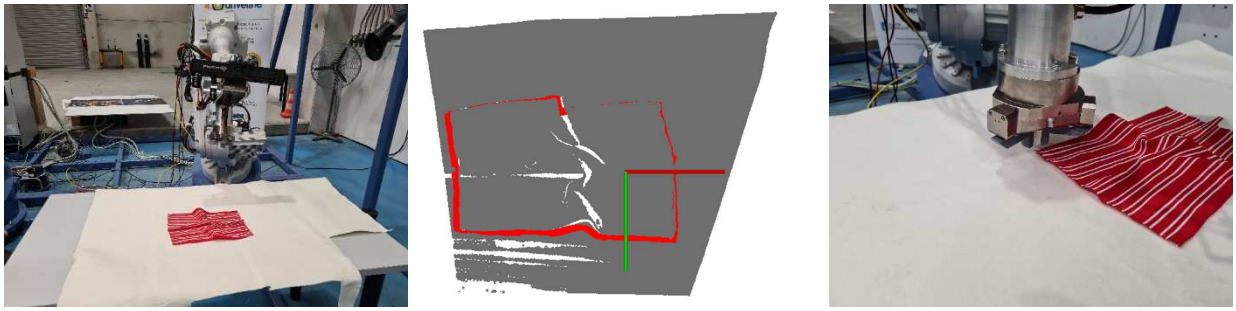


Figure 24. Tests of the calibration of the segmentation algorithm with the robot for several fabrics

3.1.2. Grasping and manipulation planning for wrinkle removal

3.1.2.1. Code organization and basic parameters used

The developments done in order to achieve a data-based solution rely on two main blocks: one for data generation and another for the development of a deep Reinforcement Learning solution. The different modules included in the solution attend to the different points addressed in the following diagram.

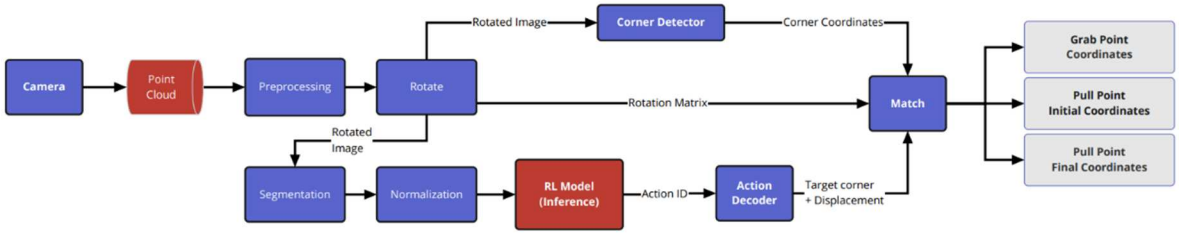


Figure 25. Data-based solution information flow

Apart from the *clothsim* environment built separately for training and modelling the wrinkle detection (*clothsim* is also used for modelling in Work Package 6 by LMS), the deep Reinfocement Learning framework was developed following a deep Q-Learning (dQL) approach.

All the data generated and used for training is organized as described in the deliverable D5.2 and will be made available as open access. Also the availability of a repository with the dQL environment and all the required functionalities will be evaluated.

The *clothsim* environment has been used for the training of the system producing initial states while the learning routine has suggested different actions considering the Q-values (Q-matrix) by an *argmax* function. After the application of the actions *clothsim* returned the transition of the fabric and the values of the Q-matrix have been updated following a Q-learning update rule where a reward function stablishes how good or bad the action was attending to the calculated entropy for each state. The DQL procedure has been trained built on a ResNet18 architecture as a backbone with an input shape (224,224,4) and the RL has been set with the following characteristic parameters: the gamma factor has been initially set as 0.75 and epsilon has variated from 1 to 0.3 in 8000 steps. During the whole learning process a log file captures the global status of the knowledge acquired by the system.

3.1.2.2. Results on virtual and real environment

a. dQL procedure training

We validated the development in real and virtual environment. The following figures show a few examples of the clothsim environment where the dQL routine has been learning.

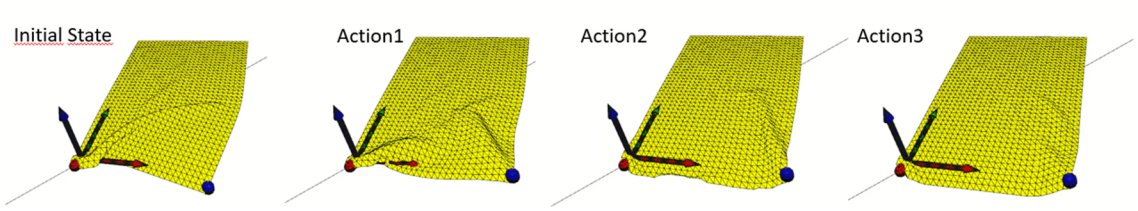


Figure 26. Simple example of fabric manipulation using the developed clothsim environment for training. The red point indicates the grab point and the blue one the pull point.

The simulation consists of a fabric in a given initial state with wrinkles and the strategy to remove them using the bimanual robotic solutions to grab one corner and pull from other in a given direction to remove the wrinkles. To understand the action-effects for a given state, it is necessary to simulate the fabric behaviour applying several forces in different directions, always pulling from other corners than the one used to grab the fabric. Figure 26 shows a manipulation sequence of simulation actions that allows to obtain an almost flat fabric after the Action 3. Figure 27 shows the simulation for a complex manipulation with the required actions to remove the wrinkles.

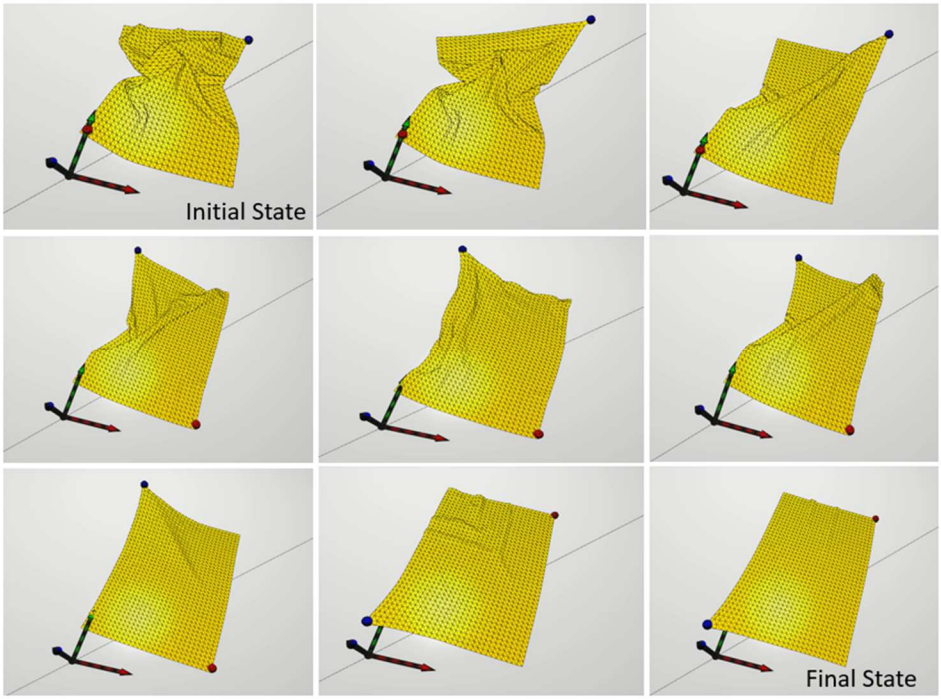


Figure 27. Example of fabric manipulation using the developed environment for training under a complex manipulation task. The red and blue points indicate grab and pull points respectively.

b. Manipulation action with dQL approach

The figure below shows two examples of the fabric manipulation in such virtual environment (*clothsim*) in order to reduce the wrinkles now representing the phi, theta, z information described in the deliverable D5.2. The images show the status of the fabric before and after the application of an action selected (red arrow on the left image) attending at the values of the Q-matrix. These values are developed during the training procedure where the system tries to stablish, for given state, which action drives the maximum entropy reduction. Indeed, the entropy is the metric chosen during the whole process for the wrinkle minimization in the fabric.

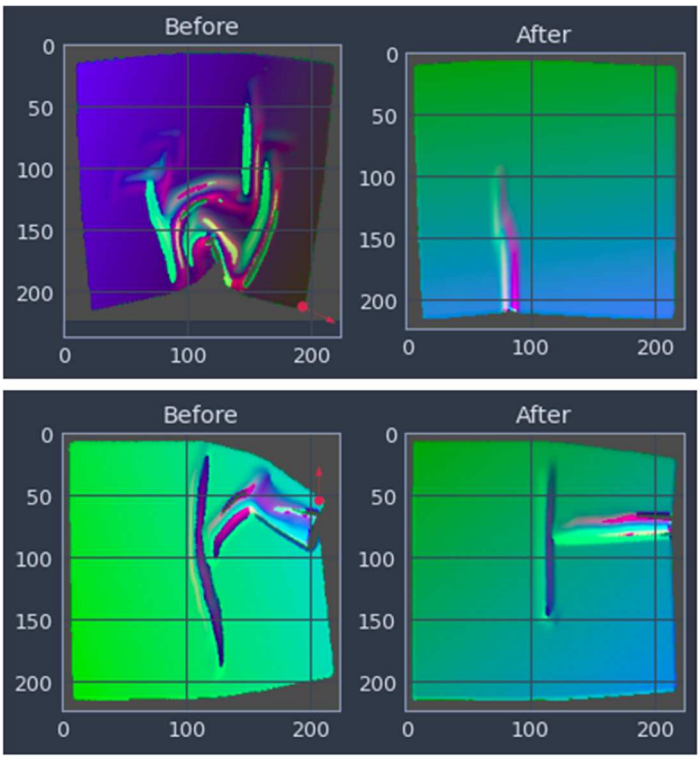


Figure 28. Examples of virtual fabric manipulation following dQL approach. The red arrow indicates the action suggested by the algorithm after learning.

c. Actions sequence for an optimal manipulation

In order to select the actions, the knowledge of the system is encoded in the classic Q-matrix which is inferred by the system for a given state. Such codification is done using a 6x6 matrix that consider corners to manipulate and directions that can be taken within the fabric manipulation. The final outcome of the procedure is a set of three points: one static point to fix the fabric, a second point that represents the corner to be manipulated and a third point which represents the point where this last corner has to be placed (i.e. grab point, pull point initial coordinates, pull point final coordinates). To decide what action to take, the system evaluates the Q-matrix and select the action that corresponds with the maximum value on the matrix through an *argmax* function. The Q-values are updated including a reward function (a scalar value) which becomes positive if the entropy is decreased. So the Q-values hold the information of the reduction of the entropy that the system expects by the application of a given action. In a way that though the application of the action that corresponds with the maximum Q-value, the entropy reduction is expected also maximum in a long term (considering a whole episode).

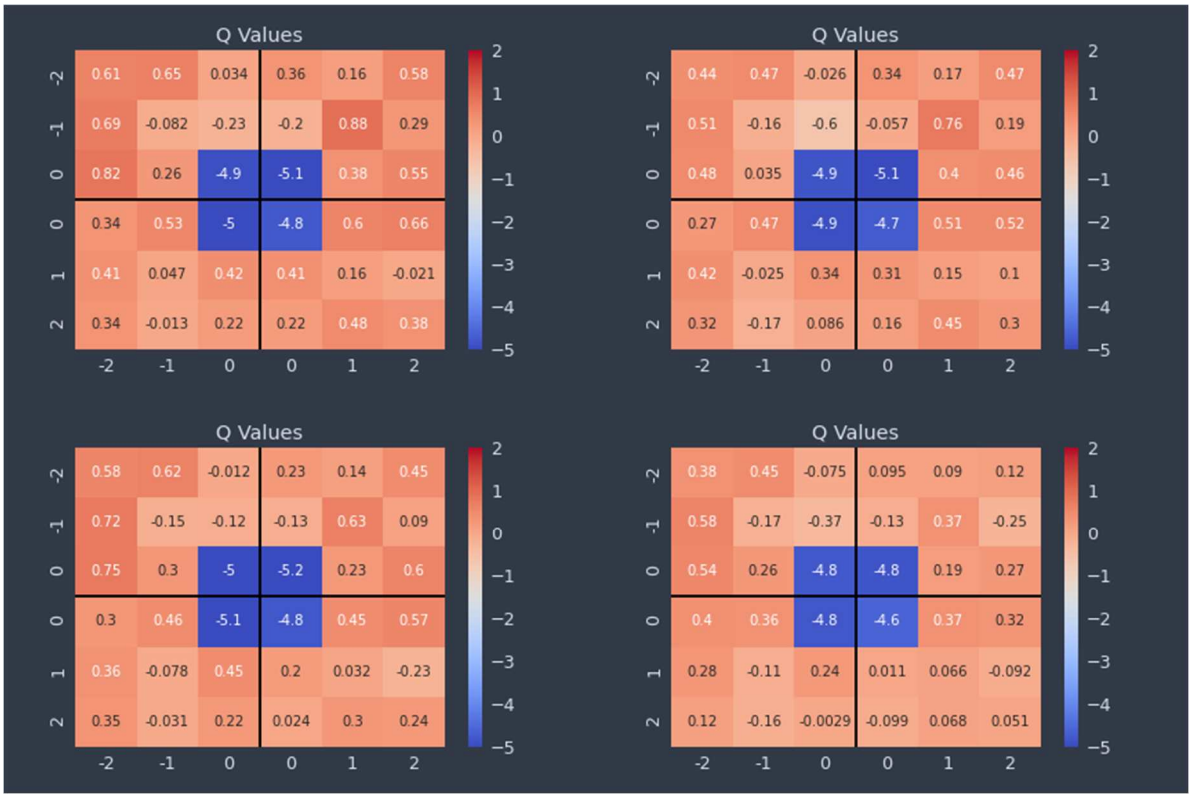


Figure 29. Different examples of Q-matrix which is estimated for each state in order to drive an action attending at the position of its maximum value. The minimum values a very small displacement of the corners meaning a very low reduction of the entropy so that they show low Q-values.

In order to quantify our solution, we validate the use of the entropy as a metric for the wrinkles of the fabric, and its minimization as the target of the algorithm for the development of a strategy to reduce such wrinkles. The following figures show how the entropy is reduced through the actions followed attending at the Q-matrix towards an acceptance distance from the target (plane). It iterates until a certain threshold that is configurable, affecting directly the number of actions necessary to reach the final state.

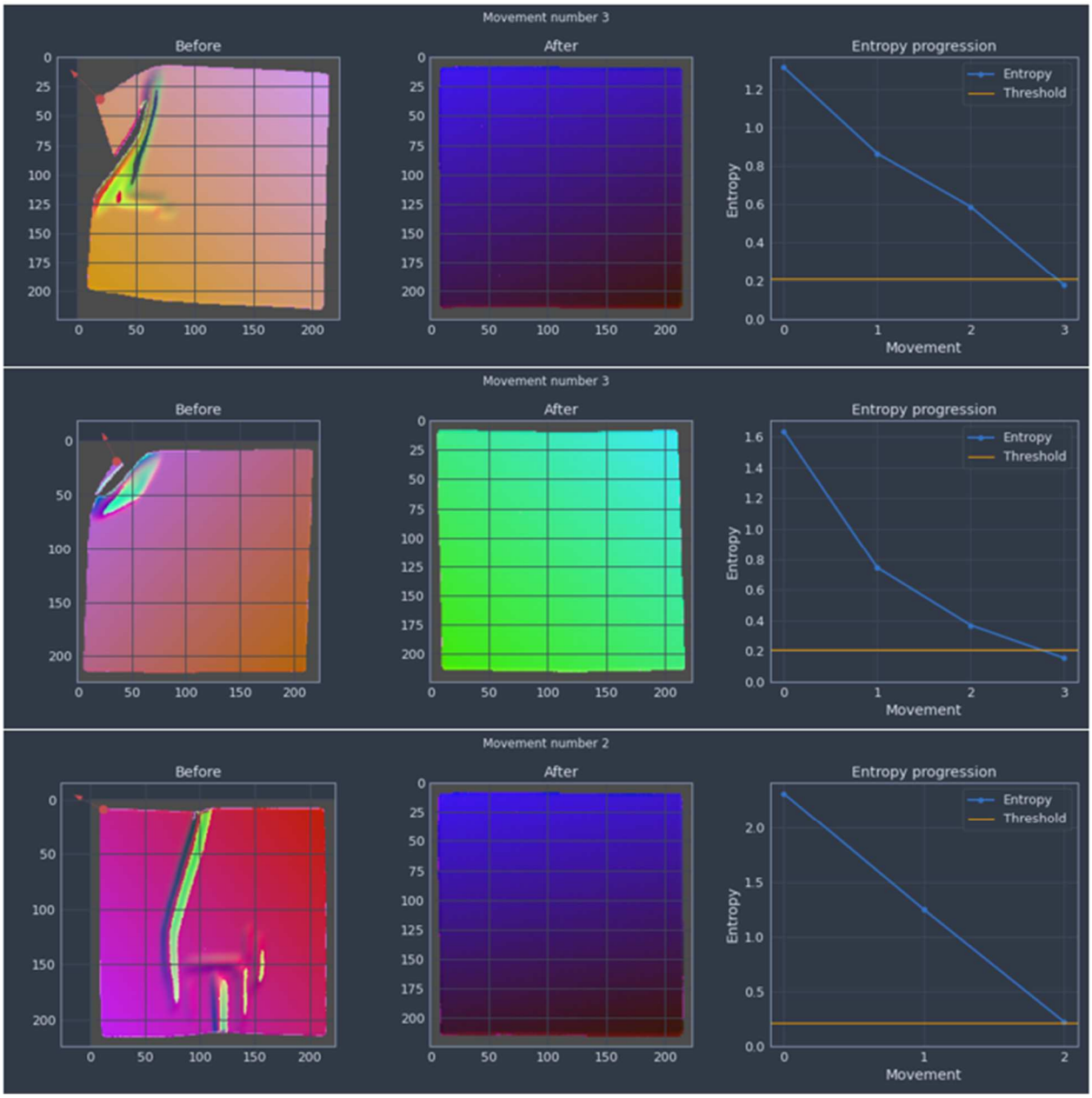


Figure 30. Examples of entropy evolution during synthetic fabric manipulation. The figure shows the initial state of the fabric and the final after different actions are applied. The red arrow indicates the action suggested by the system. The right side of the figure shows the entropy evolution during the application of the actions. It can be seen how it decreases until a certain threshold (orange horizontal line) is crossed, meaning that the fabric is close to an ideal (plane) state.

In addition, we also have validated the knowledge structure behind the action selection method which represents essentially the plan of the system to eliminate the wrinkles of the fabric in an optimal way. For that purpose we evaluate the entropy variation suffered under the application of different actions attending at the knowledge represented in the Q-matrix. For a given initial state, we apply different actions ordered by their optimality considering the Q-matrix. The figure below shows how the action corresponding with the maximum in the Q-matrix triggers the maximum decrease of the entropy, which is the target of the problem.

Figure 31. Example of entropy contrast between best (blue curve) and a suboptimal action (red curve) for a given concatenation of states.

We also compare the evolution of the entropy following what can be called the *optimal path* with a situation where not the best action is applied considering the Q-matrix in order to see how the election of the maximum value drives the most robust curve towards the minimization of the entropy. The figure below shows how the evolution of the entropy follows a decreasing curve towards the minimum entropy, while if considering always the same state of the fabric, not optimal actions are taken, the entropy follows a more noisy behaviour. This demonstrates also how the solution developed provides a long term optimal solution which ensures a continuous decreasing behavior of the entropy on such *optimal path*.

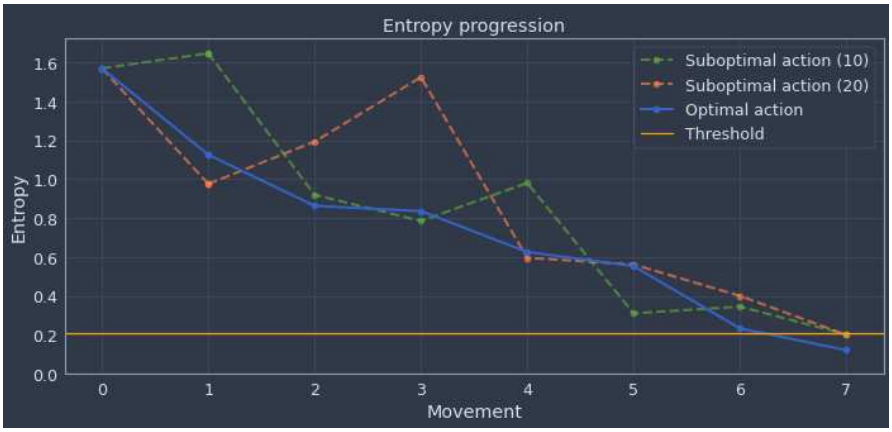


Figure 32. Optimal path. Comparison of the entropy evolution when always the maximum of the Q-matrix is taken as optimal actions (blue curve) with entropy evolution under the election of suboptimal actions (green and orange curves). The starting point for each transition is always considered as the state achieved by the application of the optimal (armax from Q-values). Suboptimal actions drive more irregular behaviour.

The validation in the real environment has been done using the high-resolution mode of the Photoneo camera. The interreflection mode was set to reduce reflections and the scan distance was set at 60-70 cm. The optics were set at f/2.8. The samples (fabric) used were provided by Selmark with a dimensions of 34,3 x 24,2 cm.

d. Validation in the real environment

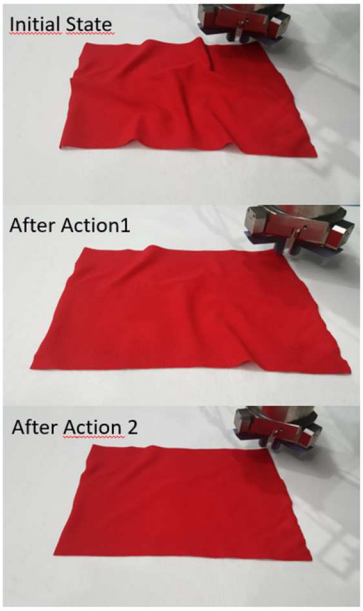


Figure 33. Example of real fabric manipulation results using the wrinkle removal planner developed. The figure shows three different steps during the manipulation following the actions suggested by the system. The sample manipulated is one from the sample set provided by Selmark. In the figure can be clearly appreciated how the wrinkles is reduced as the actions are applied.

One of the main problems that were faced was the manipulation of materials with pattern on their surface. Such scenario made impossible the use of 2D vision-based solutions and for that purpose, the use of point-cloud was proposed. The use of point-clouds also precludes the illumination dependency and the false segmentation due to the shadows. With that objective, we also tested our data-based solution in the real scenario using fabrics that present a certain pattern on the surface as can be seen in the figure below.

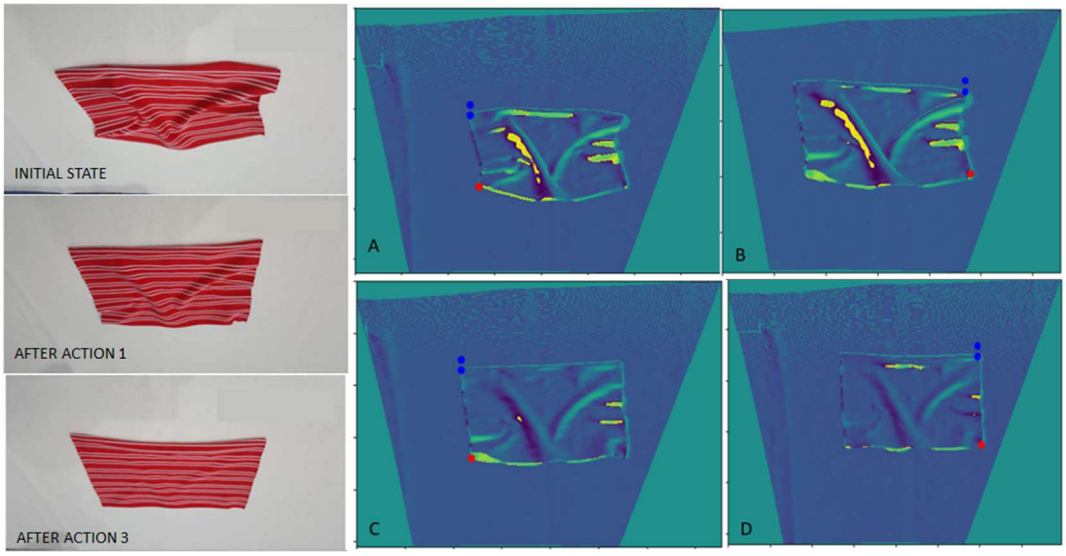


Figure 34. Example of real fabric manipulation. The left side of the figure shows the result under the application of different actions for a fabric that presents a pattern on its surface, which represents a bottleneck for conventional vision-based solutions. The right side of the figure shows the point cloud measured where the red point indicates the grab point and the blue ones the initial and final pull points for the manipulation of the fabric.

An additional problem using different methods arises when fabric textures show large reflections, as reflections increases the entropy estimation. The figure below shows the manipulation of such type of fabric using the dQL solution.

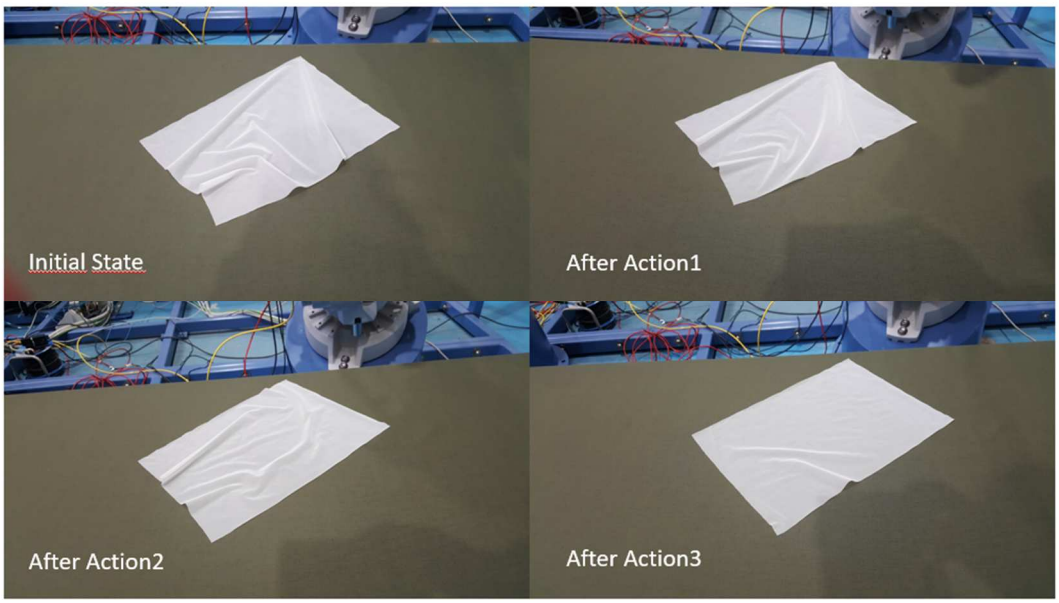


Figure 35. Example of real fabric manipulation. The example shows the evolution of the wrinkles for a fabric from the sample set that shows a texture with particularly high reflections.

We also see how the entropy for the real case follows a decreasing behaviour during the fabric manipulation. The image below shows the entropy evolution for the two examples above.

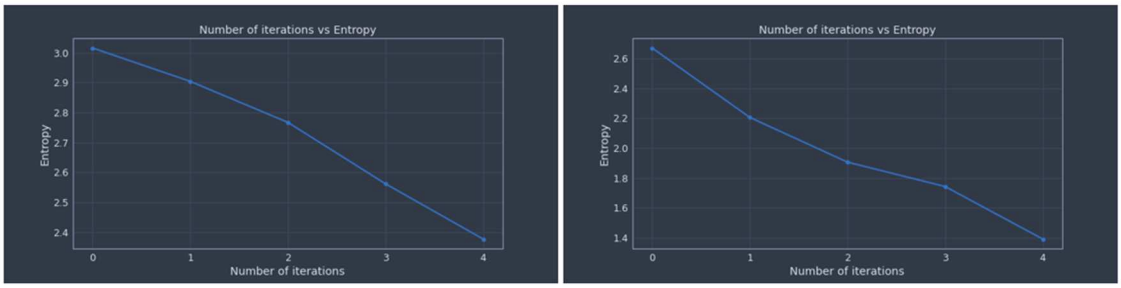


Figure 36. Entropy evolution for real fabric manipulation. Decreasing behavior of the entropy for real fabric examples following the actions attending at the Q-matrix values.

3.1.2.3. Further improvement

The presented and validated development has still a path for further improvement. The following key points have been stablished as a workplan for the optimization of the system:

- Reward function design. So far, the reward function has been set as the main goal-driver. Also the number of steps have an impact on the reward/punishment suffered by the system. However, some events within the real process can be considered and included as a specific reward or punishment that can optimize the manipulation.
- Learning parameters. The discount factor (gamma) has a strong impact on the "memory" of the learning system in order to consider immediate or future rewards with different weights. More testing with this parameter is planned. Furthermore, the exploitation vs exploration problem, quantified in the learning procedure by the epsilon parameter has a remarkable impact on the training procedure. Considering that increasing the training of the system is a key element to improve its performance, more experimentation on the impact of epsilon with the policy developed is needed and planned.
- Data augmentation. In order to improve the performance in the real environment, where some wrinkles were difficult to remove, a data augmentation procedure will be followed introducing noise (that will be characterized) in the training data to make the training set more similar to the validation (real scenario).

3.2. THIMONNIER use case 1

To assess the technology developed for the THIMONNIER use-case 1, where pouches enter to a rotative table to be manipulated in the way defined in deliverable D2.3 (Figure 37, left image), provided pouches have to be detected in a scene. Chosen pictures depict separated pouches and others with some overlap. In this case, a group of pouches is going to be processed in the given conditions to check the accuracy of bottleneck detection algorithm. Data acquisition environment consists of a dark background with the different elements on it. The different bags, provided by the end user, that are going to be scanned are shown in Figure 37, right image. These pouches have different bottlenecks: a flat-based one and a cylindrical one.

For this testing, it has been used a system equipped with Ubuntu 18 and Halcon software (version 20.11).

Figure 37. Layout for THIMONNIER use-case 1 (left); pouches to be localized (right).

In this use-case, it is necessary to remark that scanning surface should have been a metallic table. Nevertheless, as we do not have one that fulfils this condition in our installations, a dark background is set to recreate an environment the most similar possible to the real use case. Selected background is the dark Nomex already used in fabric segmentation.

The aim of this use-case, as it was already mentioned, is to detect the bottleneck of the pouches by using the CAD model and 2D image data. Here, initial image resolution was 1936x1096. However, to be able to process it in a reduced time, resolution is decreased by 50%. As a result, higher accuracy is achieved when detecting position and orientation of the bottleneck. Before obtaining these data, CAD design must be processed to compute different views to compare them with the input images. Regarding the setup followed to capture, robot, cameras and bags are positioned as it is shown in Figure 38.

Figure 38. Scanning setup for empty pouches.

3.2.1. Flat bottleneck pouches

In the case of the flat bottlenecks, result data is shown in Figure 39. Here, minimum score (the score represents the detection quality with a decimal value between 0.0 and 1.0) is set to 0.6 and required overlap is set to 0.5. Furthermore, number of located items at detection function is limited to 4 to avoid false detections. As it can be seen, bottlenecks are correctly drawn in position and orientation. Besides, when some overlap or different perspectives among pouches are depicted, they are also well-identified.

Figure 39. Flat bottleneck detection (detection in green colour).

3.2.2. White pouches

The other pouches to assess have a cylindrical bottleneck. The easiest one to process is the white one as it has higher contrast with the scanning background. In this case, same parameter adjustment is done producing the result shown in Figure 40. There, although different orientations of the bottleneck and bags are set, they can be correctly detected.

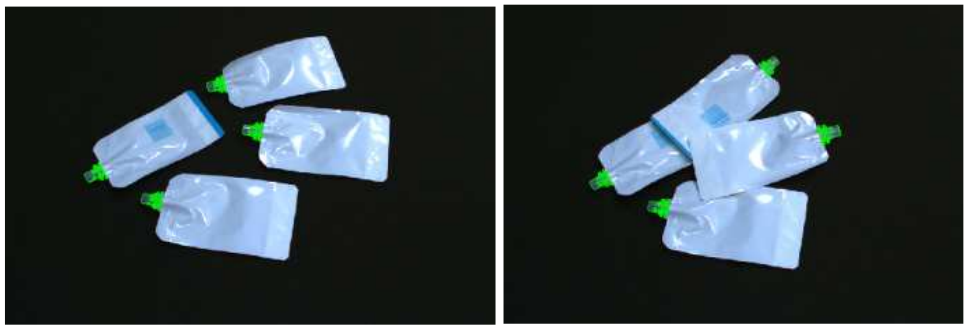


Figure 40. Correct detection of bottlenecks of white pouches (detection in green contour).

Nevertheless, when some overlap is produced and the barcode is seen, a bad detection might be produced, as shown in Figure 41, case in which detected bottleneck is placed in the barcode. This false detection can occur since the algorithm is searching image contours that will be compared to the ones obtained by the shape-based matching algorithm. If this affects the deployment process, it could be solved isolating sections after identifying the pouches corners to only process the corresponding image section.

Figure 41. Wrong detection in white pouch (detection in green contour).

3.2.3. Grey pouches

In the case of grey pouches, as they have a less contrasted colour with the surface, it results more difficult to distinguish all the bottlenecks. Here, image processing and CAD views generated are the same as the previous type of pouches. Regarding parameters configured to the algorithm, they are the same than for the previous bags. In this case, as the pouch has more shapes due to its noticeable shadows, bad detections may appear, whereas other detections are correctly obtained in orientation and position as it can be seen in Figure 42.

Figure 42. Detection of bottlenecks of grey pouches (detection in green colour).

3.2.4. Transparent pouches

The last type of pouch to assess is a transparent one, which result is shown in Figure 43. To increase image sharpness, a filter is applied to properly see the bottleneck since if we do not apply it, it is easier to the algorithm to put the bottleneck on the bag reflections. In that case, algorithm parameters are modified to constrain even more the algorithm to avoid wrong detections. For the image processed, we can see that two of the four bottlenecks are correctly identified, whereas one is orientated in the opposite direction. The last one is detected in a shiny part of the pouch.

Figure 43. Detection of bottlenecks in transparent pouch (detection in green).

One thing to consider in THIMONNIER use case 1 with last type of pouches is that the views of the CAD are obtained manually, where we set angle ranges referenced to the CAD coordinate system. In this case, it is convenient to reduce the field of view of the CAD to a range that only show the viewpoints in which then, we can find the component in the scene. If the range is too large, which means considering all viewpoints of the CAD, detection time will be increased as we have more possible orientations of the design. As a

result, it will be easier to get bad detections, as it is shown in Figure 44. Besides, if there are shiny parts on the image, it can cause them to be a false detection.

Figure 44. Bad detection of pouches bottlenecks.

3.3. THIMONNIER use case 2

THIMONNIER use case 2 was assessed following the same process as the previous one explained. For this test, filled pillow pouches are going to be scanned as 3D point-clouds. The developed technology to test is the clustering algorithm and point-cloud processing that was previously explained in the deliverable D5.2. To test this use-case, a system with Ubuntu 18, ROS Melodic and PCL 1.7 is used.

First step that must be done is to configure the Photoneo MotionCam3D to acquire relevant data. Optics aperture is set to f/2.8 and the device is placed at 60 cm approximately from the scanned elements (FIG SETUP BAGS), although more distance could be set since this is the minimum scanning distance for this Photoneo model. Then, laser wavelength can be set within the range from 800 to 1500 nm, which will provide a correct data acquisition. In this test, data is taken at 1000 nm. This configuration will provide a capture with enough information to process and to extract clusters. Besides, interreflections mode is set to reduce the amount of noise and reflects that may appear due to light conditions and surface characteristics, which are shiny or transparent.

To obtain data, environment is supposed to be a metallic table to be more realistic to the THIMONNIER Use-case 2. However, similarly to what it was done in the previous use case, a white Nomex is put on the table with the different objects on it (Figure 45). These objects will be placed with lack of overlap or with partial overlap in order to test if clustering extraction algorithm is able to identify all the items. In this case, background colour will not make a huge difference since it is going to be removed by making use of PCL plane-removal function.

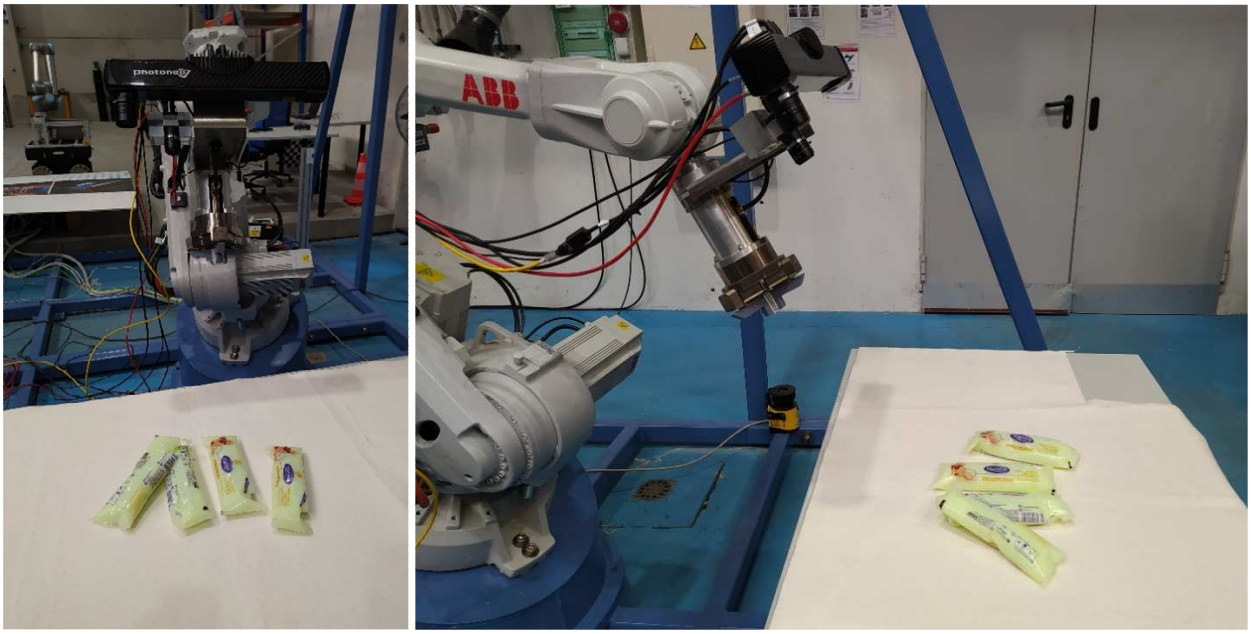


Figure 45. Robot configuration to scan filled pouches.

There here are three types of pouches to test provided by THIMONNIER: a white squared one or water bag, and two that shares its shape and are transparent, but with different liquid inside, being one light green and the other one transparent (Figure 46: left, centre and right, respectively).

Figure 46. Filled pouches from THIMONNIER Use Case 2.

3.3.1. Water pouches

The first type of bag is the most difficult to segment due to its deformation. When pouches are separated a selected distance, its identification results easier than if they are overlapped. In the last situation, as bags are deformable, there will be no distance between some overlapped parts, aspect which is the highest problem found here. With separated bags, all of them are correctly detected (Figure 47). Some algorithm parameter configurations must be set to filter a cluster size between lower and upper thresholds that will correspond to the detected one. Program is set to show each time one of the clusters. Furthermore, as it can also be seen in Figure 47 and Figure 49, clustering extraction algorithm is not affected by the orientation of the bags since they can be successfully identified as points are acquired.

Processing steps for the original image in Figure 47 are shown in Figure 48, where RANSAC filter is applied with a threshold of 1. This algorithm, in this input image causes a point reduction from 1459781 to 654034.

Then, voxel grid is also applied in boxes of size 2, producing a resulting cloud with 31941 points. Same parameters and processing functions are applied for all the following images displayed along this use case. In the next pictures, some scanning surface zones may remain in the RANSAC filtering due to shapes in the Nomex used, which usually appear in the bottom part of the image. This information can be filtered by the number of points to consider a cluster to avoid false cluster detections.

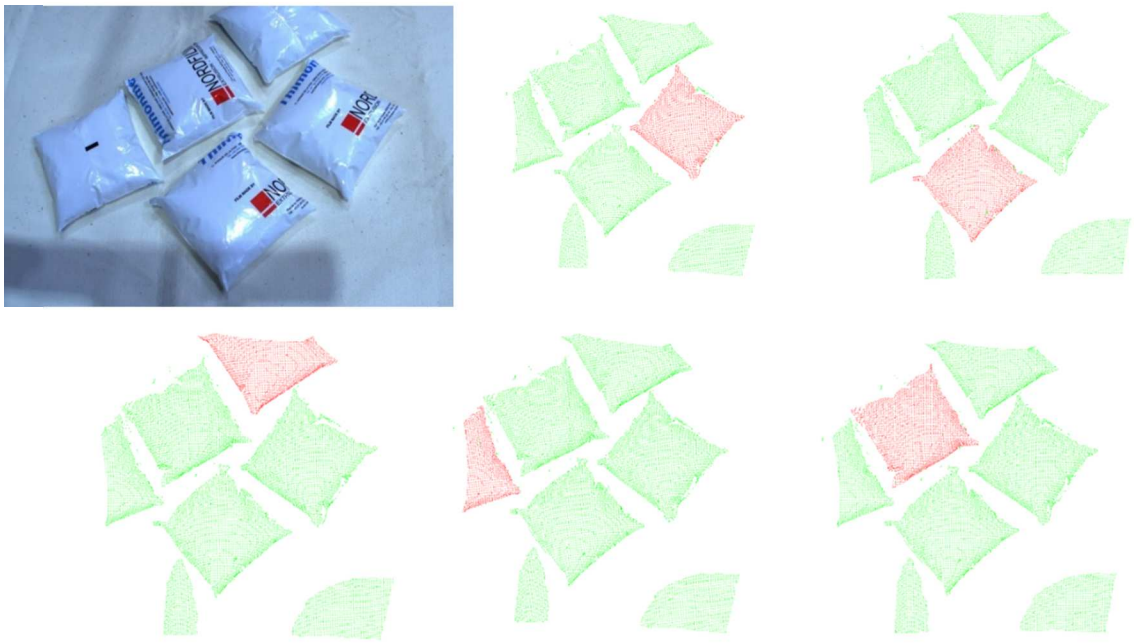


Figure 47. Original 2D capture and cluster extraction for separated water bags.

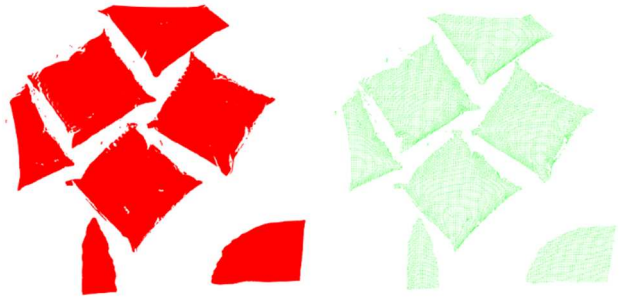


Figure 48. Ransac plane removal (left); voxelgrid downsampling (right).

When some overlap appears, clustering extraction becomes difficult due to bag deformation, pouches are connected and consequently, its obtained points, and algorithm extraction will only detect it as one bag despite cluster threshold limits. A result of this type of segmentation is shown in Figure 49. Here, overlapped bags are wrongly detected because overlapping area occurs at the same height, whereas separated ones are correctly identified. As for the empty pouches, if the deployment in real scenarios reveals the necessity of processing overlapped pouches, it will be necessary to search the cluster by means of identifying the pouches corners.

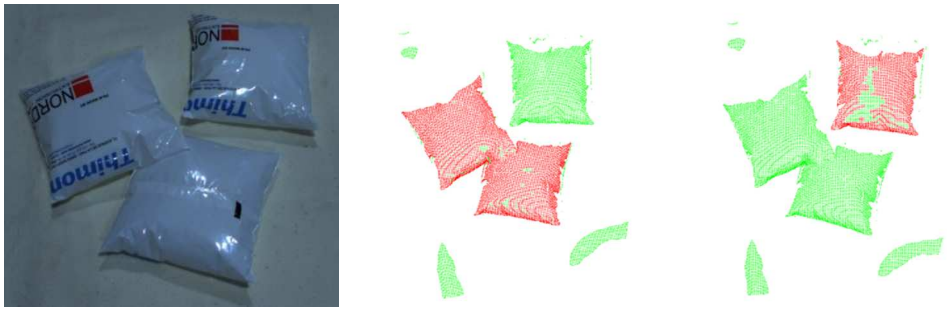


Figure 49. Original 2D capture and cluster extraction for overlapped water pouches.

3.3.2. Transparent pouches

Regarding acquisition process, it is the same as the previous situation. Nevertheless, this bag results more difficult to capture due to its physical characteristics. Although clustering extraction is achieved, parameters must be manually adjusted to obtain pouches.

In this case, Euclidean distance to consider whether a cluster is part of the others or not is increased to 13 mm. It is mainly due to the fact that only few points from the liquid and bag are obtained since emitted laser from the camera pass through the liquid. Thus, the scanning surface is what the camera is seeing, so different points corresponding to the text on the bag and surface are obtained. By setting this threshold to this value and decreasing the range of minimum and maximum points that are needed to consider a cluster, bags can be properly detected. These information results easier to obtain in the separated environment (Figure 50) than in the non-separated one (Figure 51). In the last one, some bad detection appears due to the points that are connected at the same height at overlapped zones, which mistakes the algorithm.

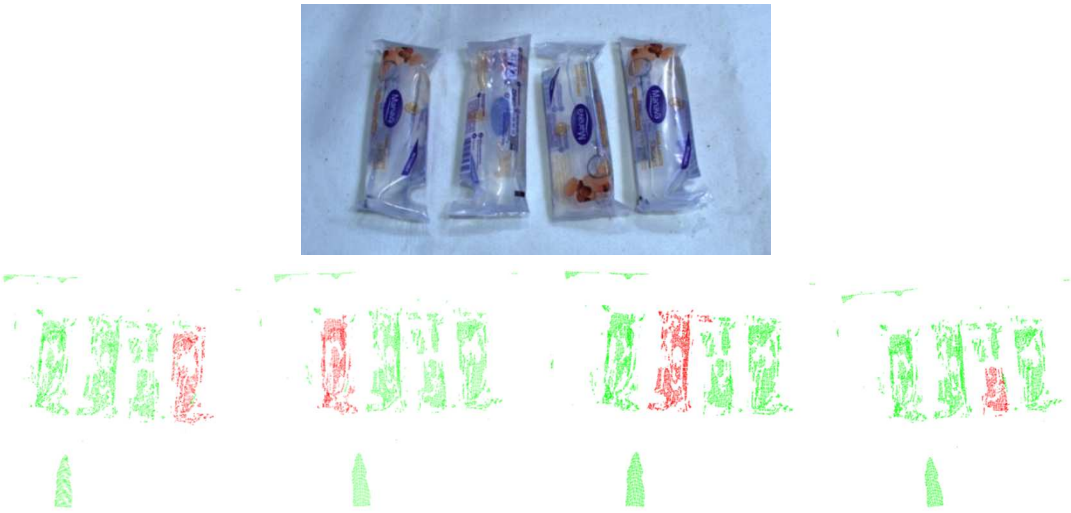


Figure 50. Original 2D capture and cluster extraction for transparent pouches.

Figure 51. Original 2D capture and cluster extraction for overlapped transparent pouches.

3.3.3. Green soap pouches

In the case of the green liquid pouch, data extraction results easier despite having the same geometry as the previous one. As the contended liquid is opaquer, more points of the pouch are acquired. Therefore, as we obtain a higher number of points with respect to the transparent liquid, to successfully distinguish clusters, thresholds must be more constrained.

In this case, when pouches are separated (Figure 52), they can be correctly segmented. However, if bags are too close, they might be detected as only one, as described in the case represented in Figure 49.

Figure 52. Original 2D capture and cluster extraction for separated green pouches.

When pouches are overlapped (Figure 53), as they have relevant volume, they can be properly detected. However, if a pouch divides one that is just under it, it can be wrongly detected. Besides, same occurs when clusters appear close to the other at the same height.

Figure 53. Original 2D capture and cluster extraction of overlapped green pouches.

3.4. VDL use case

For VDL use case, the MotionCam3D was fixed to a high structure in order to monitor the ply while an operator is manipulating it, simulating the comanipulation task. The tests were done with two different plies, one with white colour, similar to the ones used in the real application, and another with black colour made by carbon fiber to check the system limits. For both tests the camera configuration is the following:

- Operation mode: camera
- Laser wavelength: 4095 nm
- Coding strategy: interreflections, to reduce noise and material reflections.
- Resolution: 1120x800
- Coding quality: high (increased pre-filtering, it increases acquisitions time)

Besides, MotionCam3D *phoxi_camera* ROS module was used to enable continuous capture acquisition mode and publish these point-clouds to ROS messages. Then, an external ROS node is going to process and show each point-cloud in real time by making use of PCL. Down sampling with grid size of 9 is applied in X and Y coordinates, while Z has a less restrictive value (900) to simplify the point-cloud. Then, representation is done by setting point representation in size 4 to ease visualization process.

Figure 54. Setup for the validation of the continuous monitoring system.

With this configuration the results achieved for the white ply were successful as it can be seen in Figure 55. The point-clouds received from the MotionCam3D in free run mode were processed in real time. They are acquired at 12 frames per second and processes around 1 per second. It could be checked in this the following video ply_continous_monitoring.mp4.

Figure 55. Monitoring of the white ply.

However, the results for the black ply were not successful at all. The dark colours absorb the light projected by the MotionCam3D as it can be seen in Figure 56 producing empty areas in the point-clouds as it is shown in Figure 57. During the demonstration stage, if this effect appears from any of the fabrics, it will be necessary to introduce synthetic data generated dynamically using smoothed gradients to join the borders of the non-sensed area.

Figure 56. Ply absorbing the projected light in some areas.

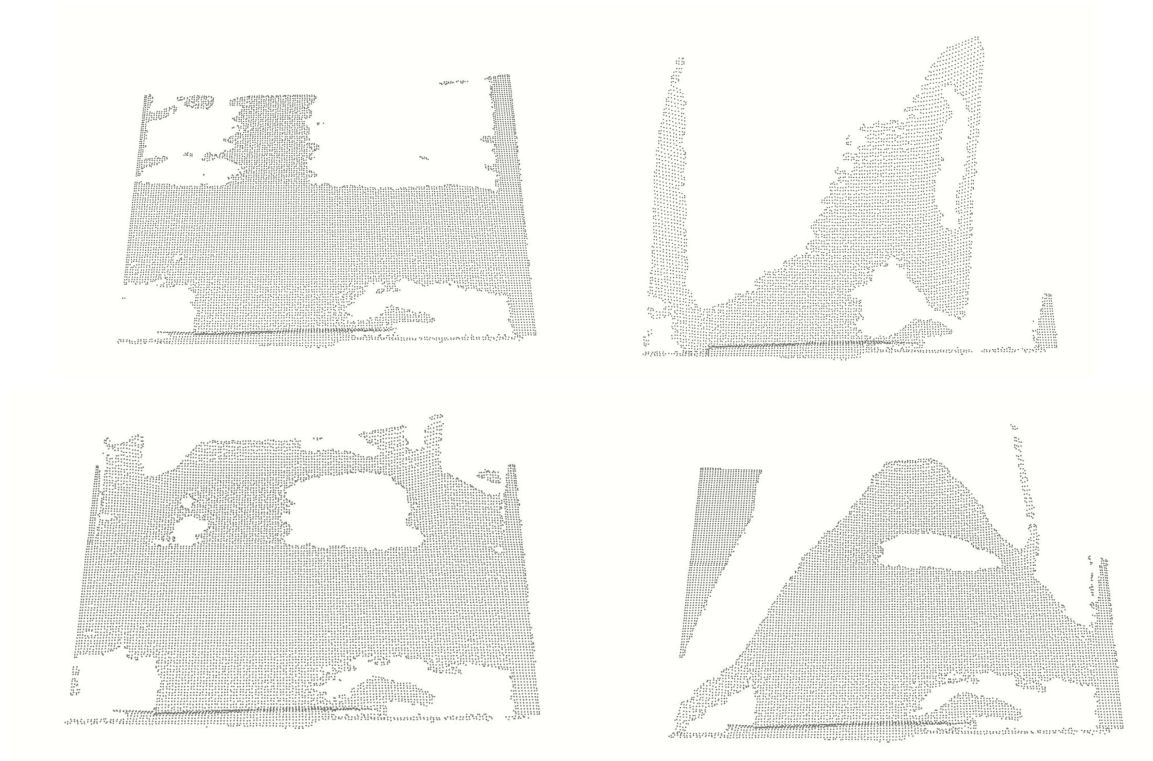


Figure 57. Monitoring of the black ply.

4. Continuous gripping monitoring system

The continuous monitoring system was validated using test data (i.e. data that was not used to train the data-driven model). The test data was acquired using the same data acquisition process that was used to acquire the training data (Figure 58). Data for various contact normal forces, velocities, gripper pose angles, and materials were used for testing. Examples of these data are in Figure 59, Figure 60, Figure 61, and Figure 62, respectively. The normal and shear forces at the contact with the object are measured using the soft force sensor developed in WP3.

A class prediction (gripper open, successful grasp, slippage, or grasp failure), accompanied with the ground truth, is provided above each graph in each figure. Colour coding of the classes (cyan, green, yellow, and red, respectively) allows comparison with the correct stage of the slip-grasp cycle in Figure 58. The prediction of a class (top row) at each time step is correct if the colour at that time step is the same as the colour of the ground truth (lower row) (see Figure 59 to Figure 62).

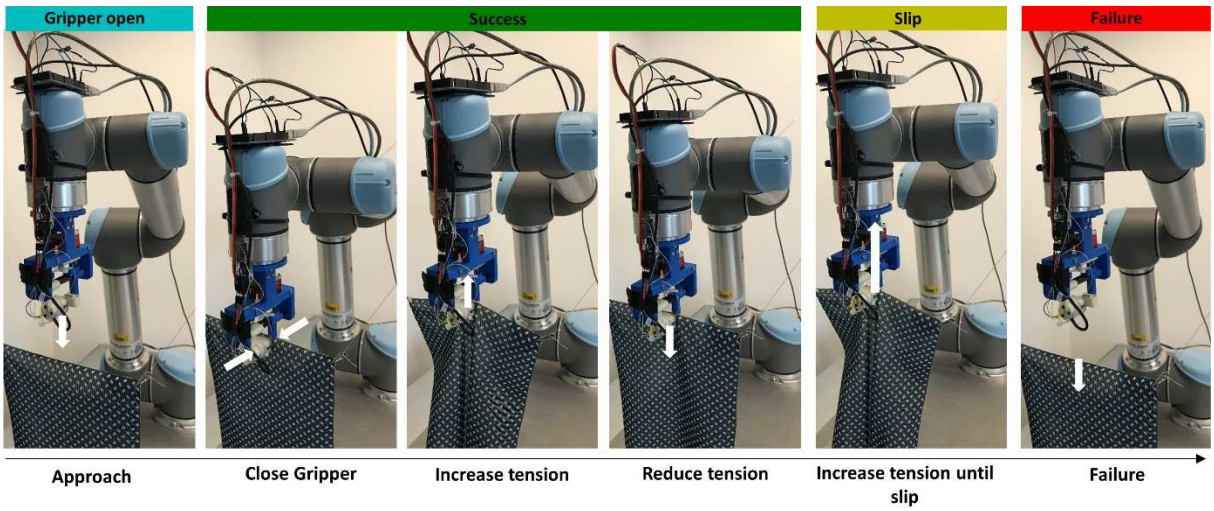


Figure 58 The repeatable data acquisition process of the grasp-slip cycle follows six steps. The gripper approaches the object suspended under it. The gripper is closed and the object is lifted upwards and then lowers to cause an increase and then reduction in tension but without slippage. The increase in tension is possible because the object is attached to the same surface than the robot arm. The object is lifted once again until slippage occurs between the two surfaces. After the object has slipped out of the grasp, the gripper remains closed as both the gripper and the object return to the initial positions.

The contact normal force will likely need to be varied in the real use cases due to the surface conditions and loading of the material. Therefore, test data was acquired for a range of contact normal forces. It was found that the profiles of the normal and shear forces were similar for all contact normal forces when the other parameters were held constant (Figure 59). Prediction accuracy was similar for each which shows the generalisability of the continuous monitoring system for different contact normal forces.

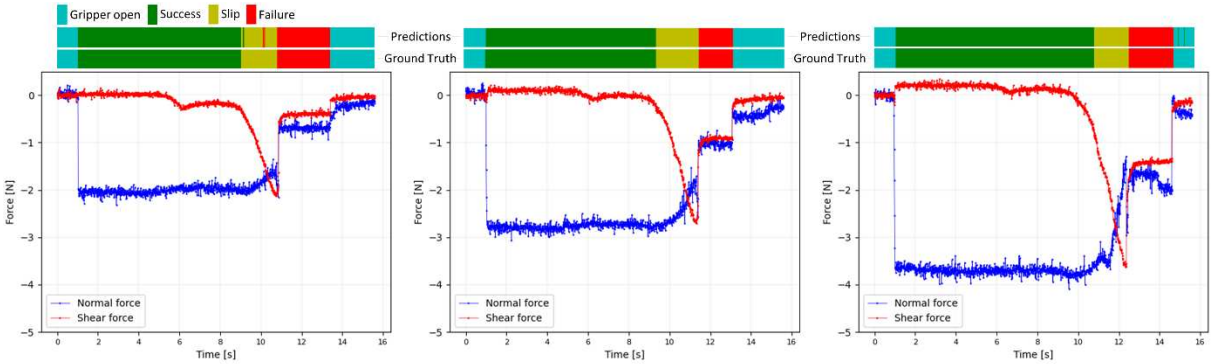


Figure 59 Varying the contact normal force from (a) lowest to (c) highest. Normal and shear forces are from the soft force sensor. The velocity of the gripper, gripper pose, and material were held constant.

The velocity of the gripper will likely vary in the use cases and, therefore, slippage will progress at various rates when it occurs (Figure 60). It was also found that the normal force reduces with a greater magnitude for faster cycles (Figure 60c) when compared to a slower cycle (Figure 60a). For instance, in Figure 60a the normal force reduced to -1.3 N after slippage but reduced further to -0.1 N in Figure 60c. However, training

was conducted with data for a range of velocities, so the model was still generalisable with a high prediction accuracy in all cases.

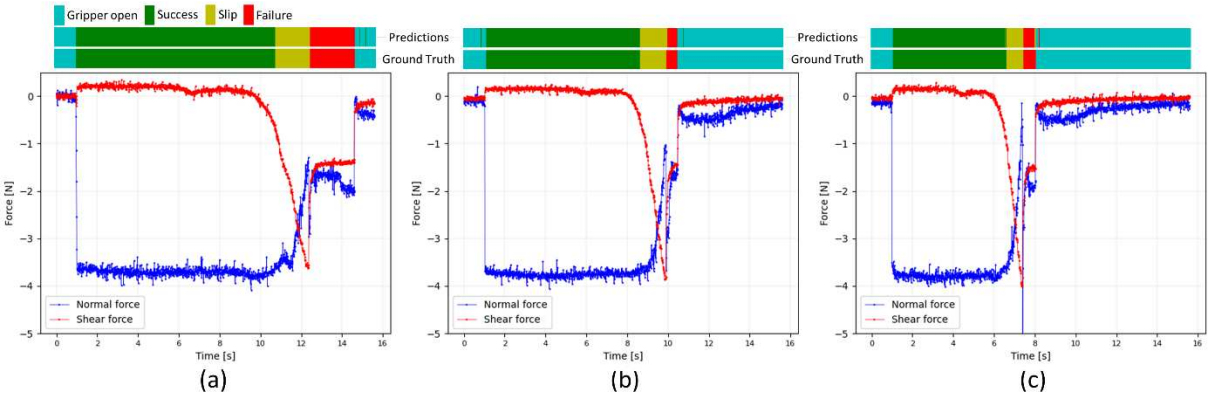


Figure 60 Varying the velocity of the gripper from slowest (a) to fastest (c). Normal and shear forces are from the soft force sensor. The contact normal force, gripper pose, and material were held constant.

The soft force sensor is able to measure "accurately" the shear force along a single direction. While it is not sensitive at all to shear in the direction orthogonal to this primary direction, it is nevertheless "sensitive" at various angles to the primary direction of shear. The shear force exerted on the sensor by the material is unlikely to always be along the most sensitive direction and therefore the trained model also needs to be sensitive to slippage that is not along this direction. Angles up to 20^o from the primary shear direction of the sensor were tested (Figure 61) with very little difference in the force profiles or prediction accuracy.

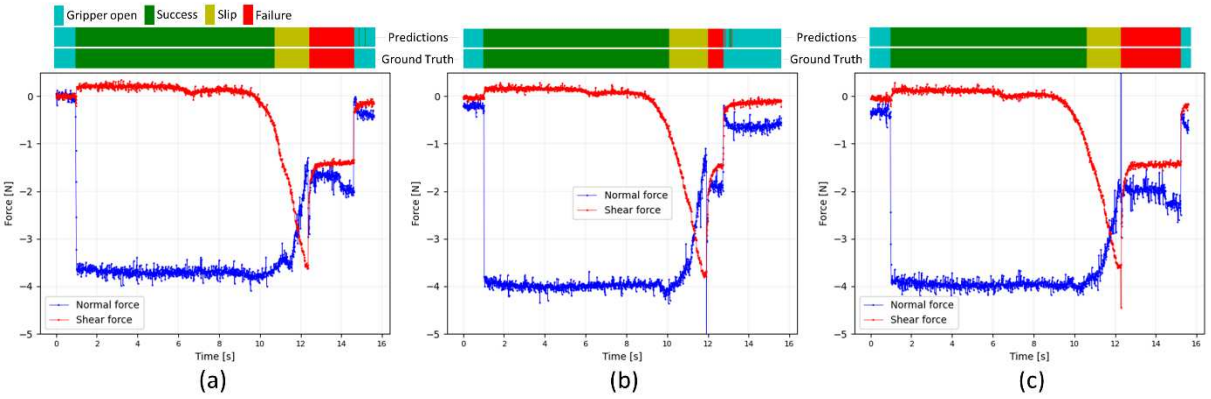


Figure 61 Varying the contact pose angle of the gripper at (a) 0^o, (b) 10^o and (c) 20^o from the primary shear force direction of the soft force sensor. Normal and shear forces are from the soft force sensor. The velocity of the gripper, contact normal force, and material were held constant.

The model should also be generalisable to different materials used in the different use cases. However, due to the large quantity of data needed for training a data-driven model, data for glass fiber (VDL) was not used in the training dataset (both SELMARK fabric and THIMONNIER pouches were used in the training data) as repetitive slippage during data acquisition could have damaged the sensor. Despite the large differences in the test data force profiles of each material (Figure 62), the model generalised well to the glass fiber (Figure 62c) despite not having seen data for this material during training. This was even true despite the lower contact normal force used which resulted in a marginally lower prediction accuracy than higher contact normal forces (Figure 59a)

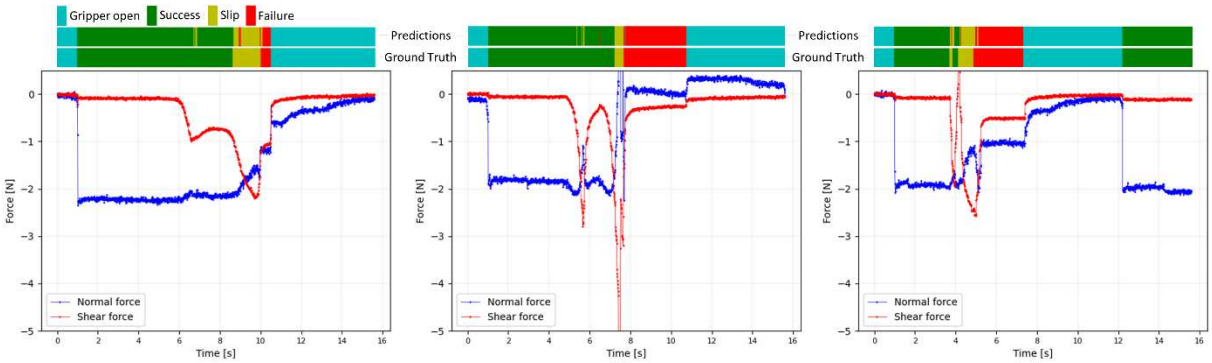


Figure 62 Varying the material between (a) SELMARK fabric, (b) THIMONNIER pouch and (c) VDL glass fiber. Normal and shear forces are from the soft force sensor. The contact normal force, velocity of the gripper, and gripper pose were held constant.

5. Conclusion

The assessment performed in the task 5.4 for the functionality demonstration of the perception modules and technologies described in the D5.2, allows to conclude that they are covering the requested specification to implement the final demonstrators. The modular solutions will enable an easy deployment, fast testing, and high scalability beyond the project execution. It also highlights the major limitations that should be considered during the demonstrations. During the integration and validation stage, configuration and optimization will be tackled on those technologies to ensure transferability to the proposed use-cases in the final benchmarking.

Functionalities / Tool Developed	**Successful Scenarios**	**Unsuccessful Scenarios**
Detection and tracking of persons	Single and multiple person detection and tracking with different clothing.	Scenario when a person crosses and adhere occlusions.
Movement decomposition	Single and multiple person movement decomposition.	Scenarios when a person cloth and the background matches and when there is an overlapping in one of the cameras between operators.
Gesture recognition	Five persons gestures with 97% detection rate.	Similar limitations as expressed above (movement decomposition) because the gesture recognition relates to the movement decomposition.
2D-3D fusion for soft material detection and identification	Selmark: All type of cloths with dark green Nomex to add colour contrast. Thimonnier use-case 1: Detection of flat, white, grey, pouches bottlenecks. Thimonnier use-case 2: Water and green soap pouches without overlapping.	Selmark: White cloth with light background. Thimonnier use-case 1: Transparent pouches detection and the pouches with high reflection. Thimonnier use-case 2: Overlapping pouches and transparent pouches.

	VDL use-case: White ply continuous monitoring.	VDL use-case: Black ply continuous monitoring.
AI-Based manipulation planning strategies for wrinkle removal	Selmark: Manipulation and planning of fabric without patterns and reflectivity.	Selmark: 1) Manipulation of fabric with patterns, bottleneck for conventional vision systems. 2) Fabric possesses higher reflections
Continuous feedback for real-time control	Selmark fabric, Thimonnier pouches and VDL glass fibre.	Data for glass fiber (VDL) was not used in the training dataset.