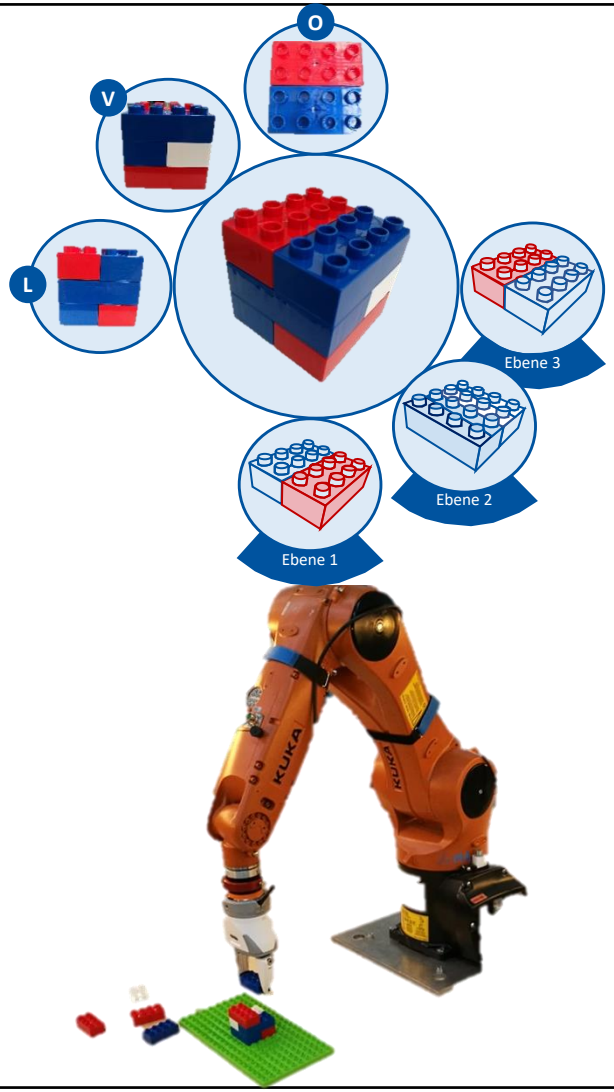


SuperRTwin 2021-2022

Human Robot Cooperation

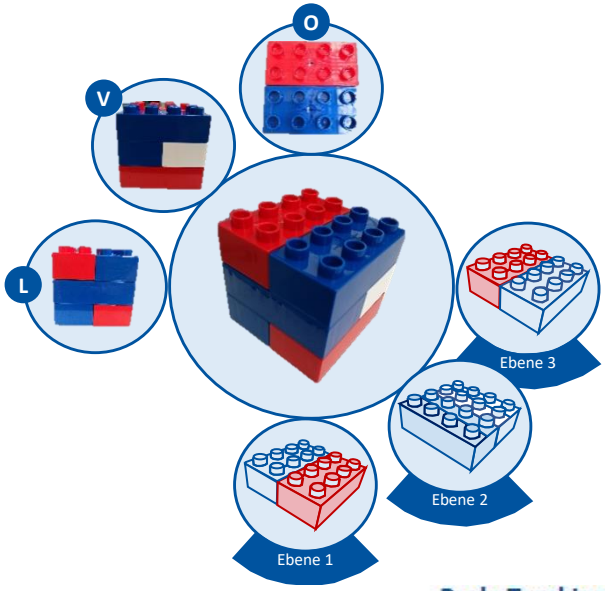
Khansa Rekik
Ganesh Gopalkrishna Hegde
Girija Bangalore Mohan
Shayari Bhattacharjee

Center for Mechatronics and Automation GmbH, Saarland University

31.03.2022, Saarbrücken

ZeMA Zentrum für Mechatronik und Automatisierungstechnik
M:sys

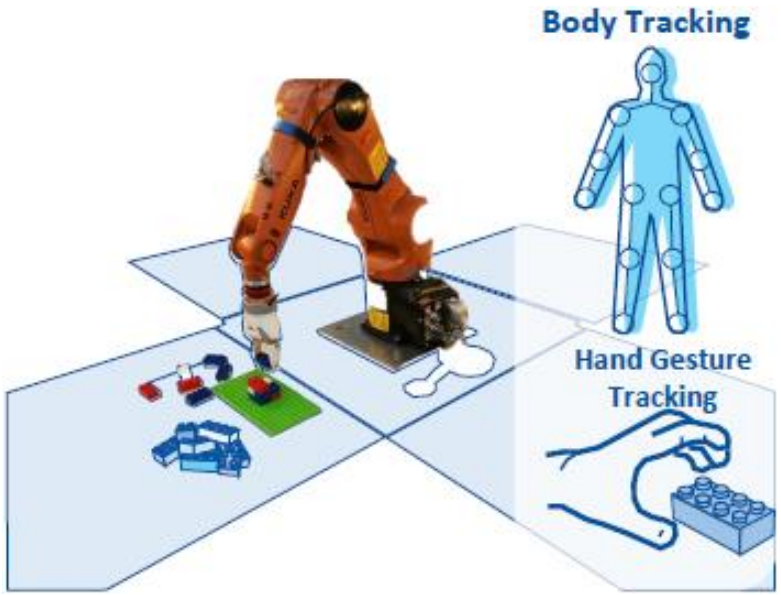
SupRTwin - Human Robot Cooperation



Rules of the game:

– A Human is supposed to help the robot assemble the LEGO part
– The human is only allowed to further assemble the LEGO part
– The Human is supposed to know the correct sequence
– The human is not allowed to disassemble or change the existing LEGO part
– The human is not allowed to enter the assembly area as long as the robot has a LEGO part is its gripper
– The robot is not allowed to move while the human is in the assembly area.

SupRTwin - Human Robot Cooperation



Camera system

– Scanning the LEGO parts and assembly before and after Human action to visualise the changes at every step.

Human Hand Tracking

– Live detection and tracking of Human Hands during assembly to predict human intention.

Digital twin

– Visualise the entire process as a simulation by adding a Human Avatar to the environment using Process simulate.

Camera System: Perception

PERCEPTION MODULE

LEGO DETECTION

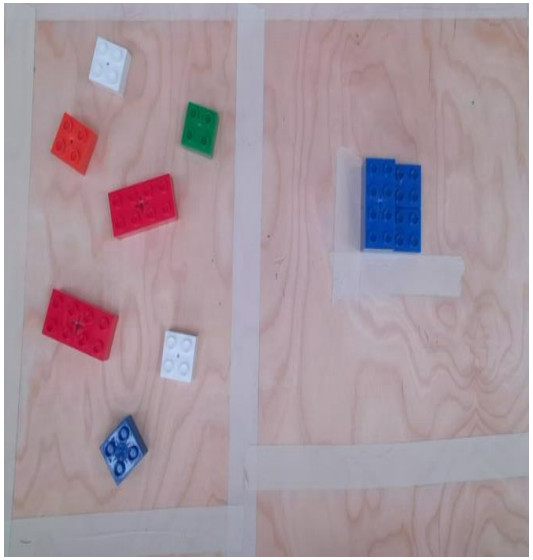
- Create a directory of the lego blocks available for assembly.
- The directory contains Unit ID(sorted- Left to Right, Top to bottom), center, color, lego label(small/medium/ large) and depth of the lego blocks.

ASSEMBLY OBSERVATION

- Capture depth maps before and after human intervention.
- Compare the depth maps to visualise changes made by human in terms of lego blocks.

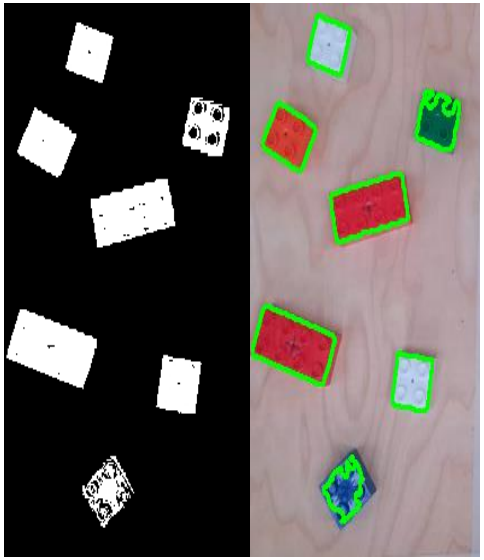
Camera System: Perception of Lego Blocks

1. **Creation of HSV database and contour area database to detect Lego colour and Lego Label**



2. **Setting of Region of Interest to denote the Lego Block detection.**

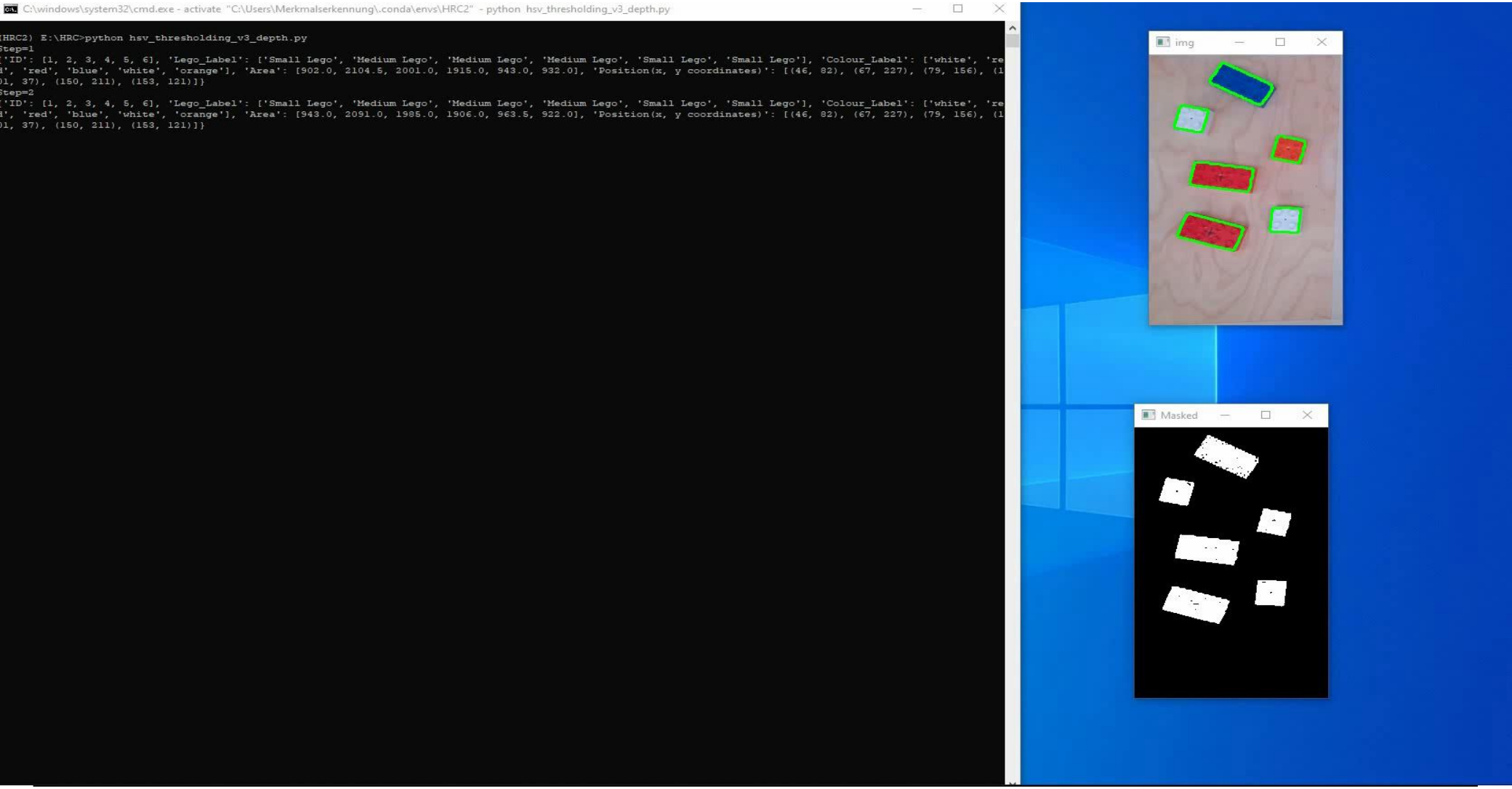
3. **Create a Custom Mask to identify different Lego Blocks and find contours and center.**



4. **Create a dictionary of Lego blocks information containing Unit ID, Center, Depth, Lego Label, etc and compare with the dictionary after Human intervention to find missing IDs.**

Step=1
{'ID': [1, 2, 3, 4, 5, 6],
'Lego_Label': ['Medium Lego',
'Small Lego', 'Small Lego',
'Medium Lego', 'Small Lego',
'Small Lego'], 'Colour_Label':
['red', 'orange', 'white', 'red', 'white',
'green'], 'Area': [2103.0, 1161.0,
905.5, 1874.0, 983.0, 941.0],
'Position(x, y coordinates)': [(43,
213), (39, 87), (74, 32), (110, 130),
(148, 235), (170, 78)]}

Camera System: Perception of Lego Blocks



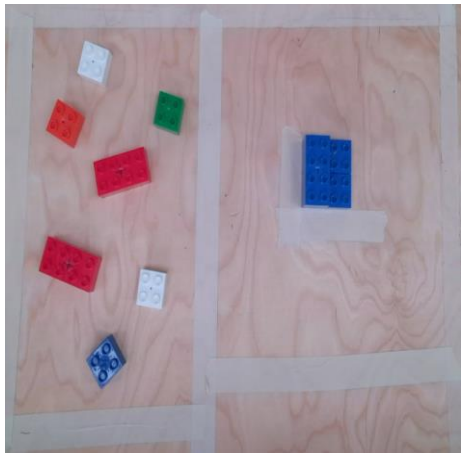
ZeMA Zentrum für Mechatronik und Automatisierungstechnik

M:sys

Camera System: Perception of Assembly Area

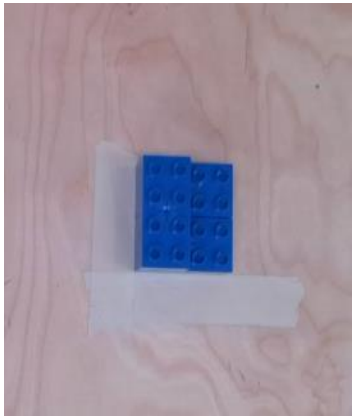
1

Reusing HSV database and contour area database to detect Lego colour and Lego Label



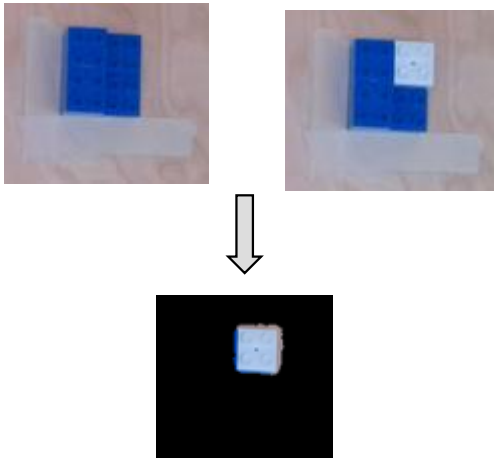
2

Setting of Region of Interest to denote the Assembly Area detection and collecting Depth maps before and after human intervention at every step



3

Compare depth maps before and after human intervention to create a mask of newly added components and use contour detection.

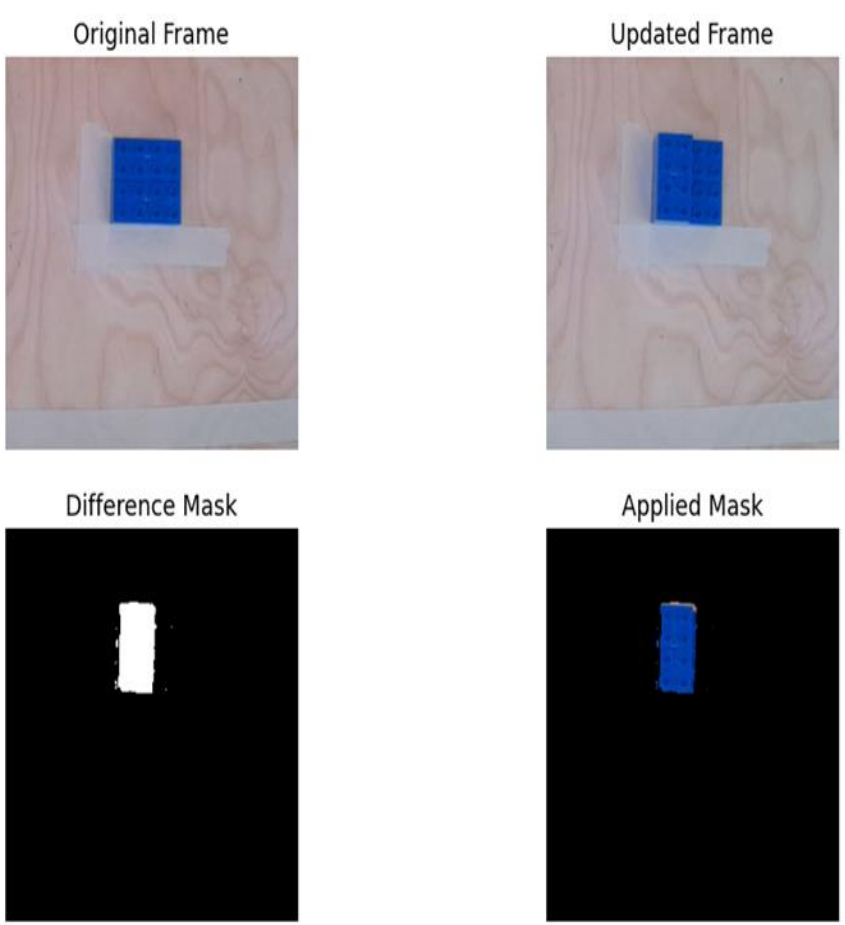


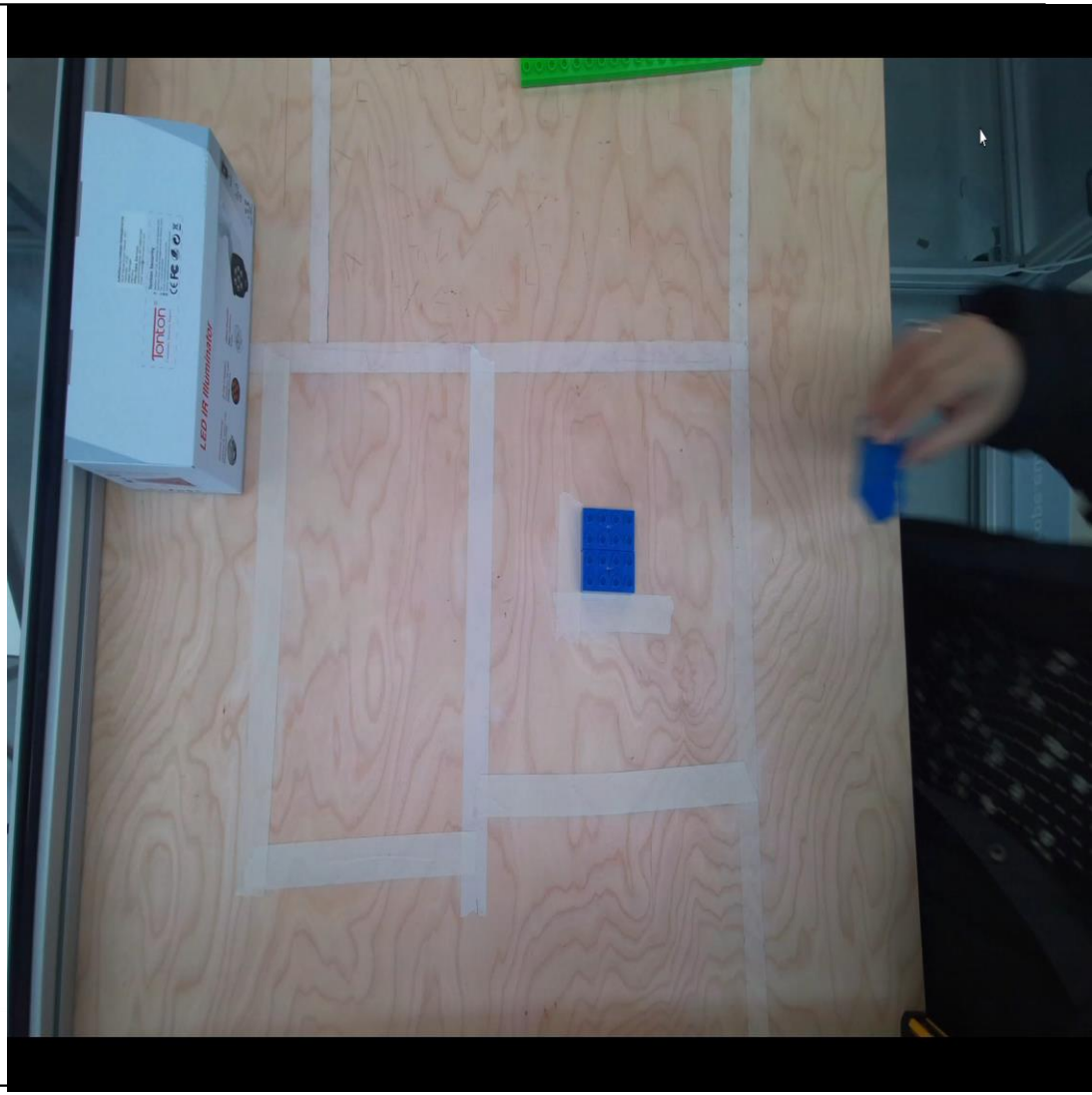
4

Get the characteristics of the newly added blocks and compare to the color and area values of IDs missing after human intervention to confirm objects removed from Lego Blocks Area are placed in Assembly area.

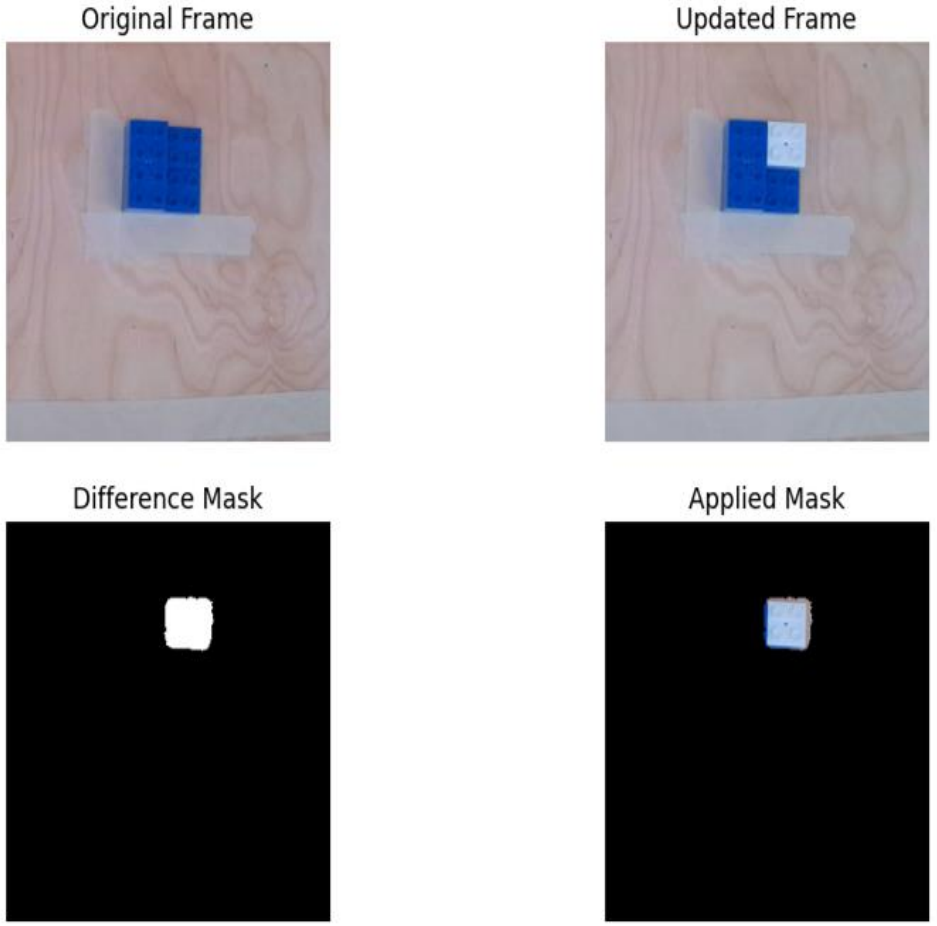
Step=1
assembly color:
['White']
assembly area:
[778.5]
assembly hsv values:
[(124.0, 54.0, 250.0)]
assembly lego label:
['Small Lego']

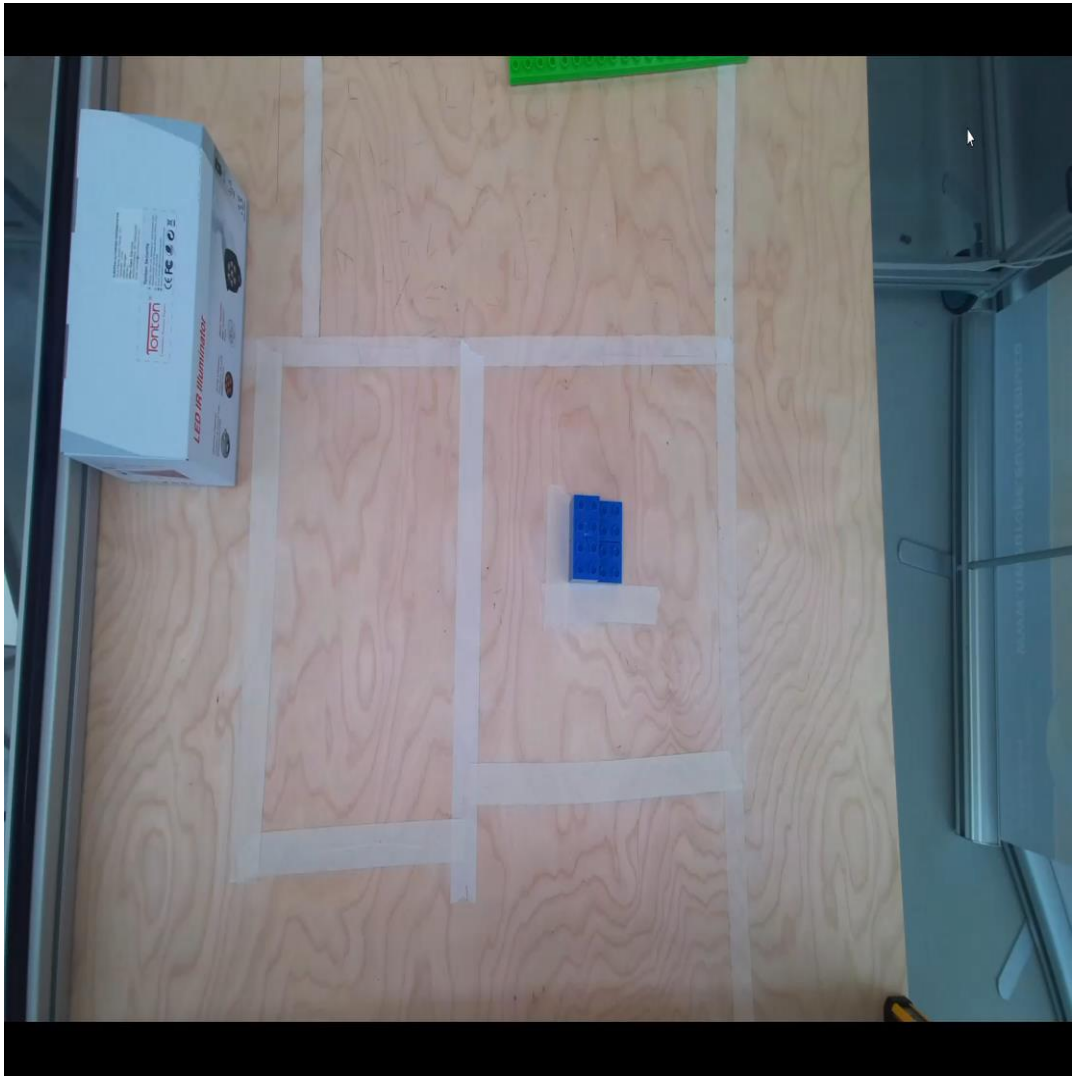
Camera System: Perception of Assembly Area - Step 1





Camera System: Perception of Assembly Area - Step 2





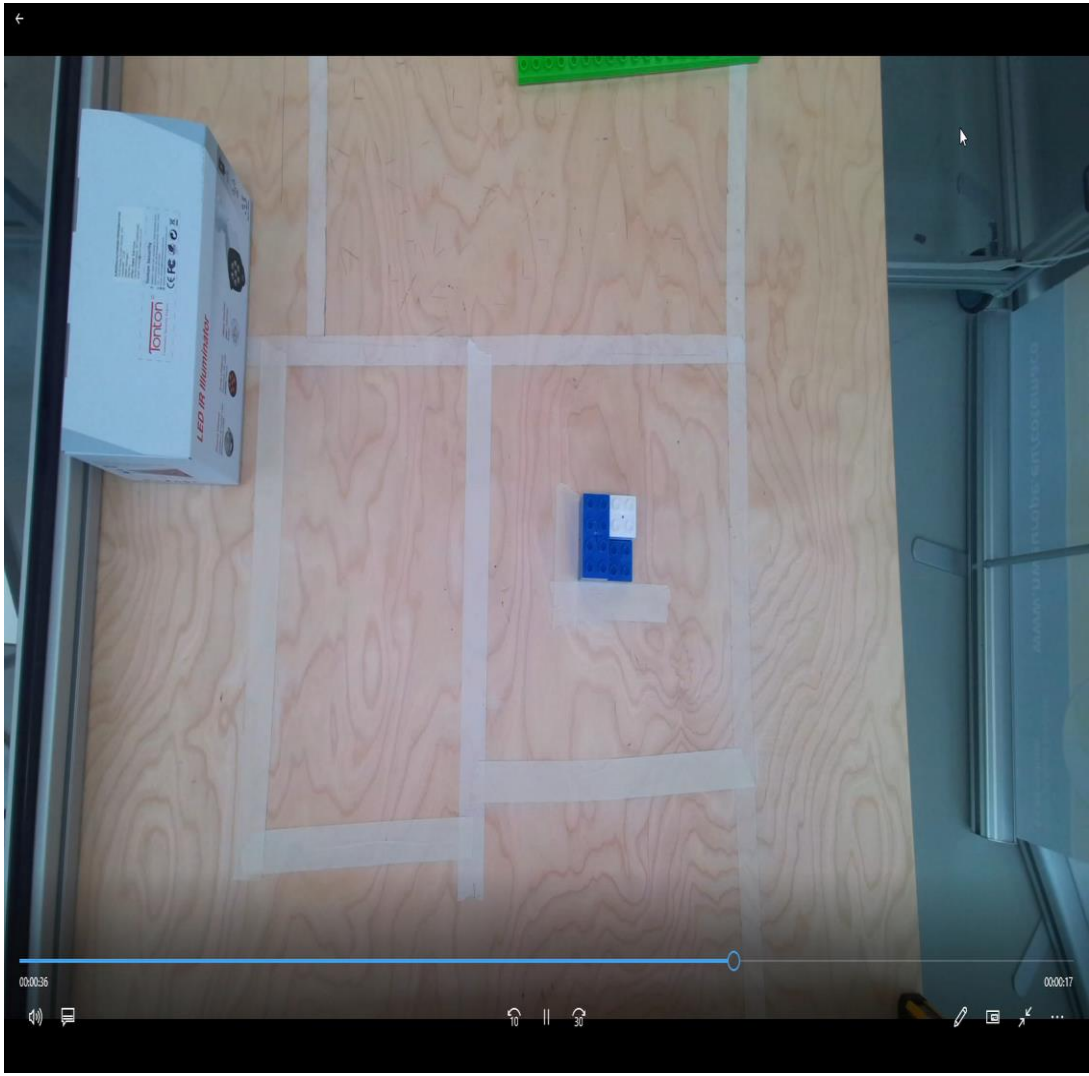
Camera System: Perception of Assembly Area - Step 3

Original Frame

Updated Frame

Difference Mask

Applied Mask

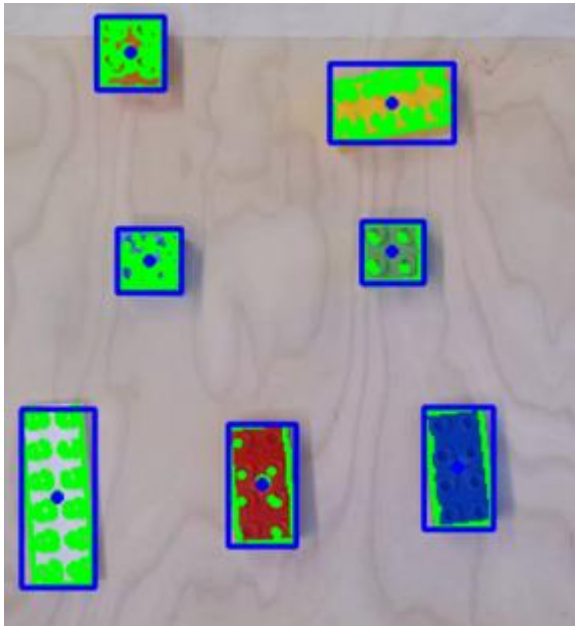


ZeMA Zentrum für Mechatronik und Automatisierungstechnik

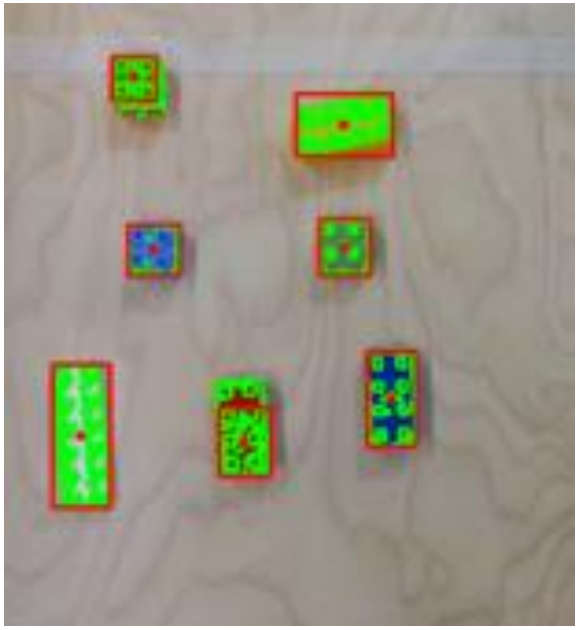
M:sys

Camera System:Shortcomings of Perception Module

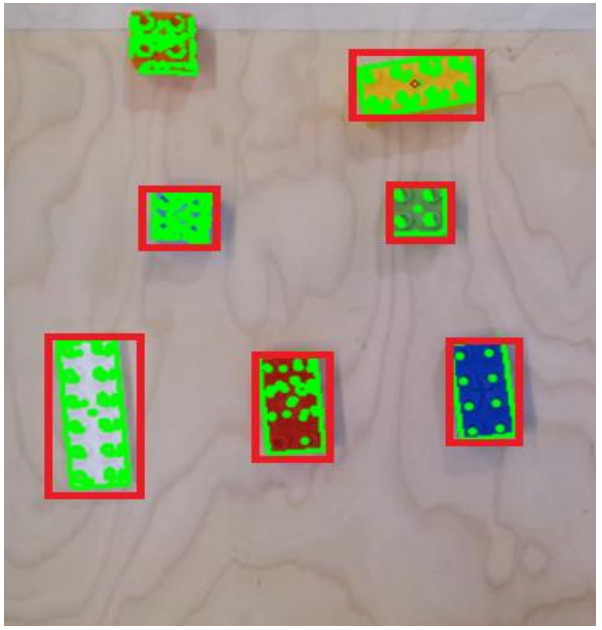
1. **The HSV values and the contour areas are susceptible to lightning conditions, camera angle and camera noise which causes deviations in assigning static IDs to the lego blocks..**



Perfect Lightning



Semi-Perfect Lightning

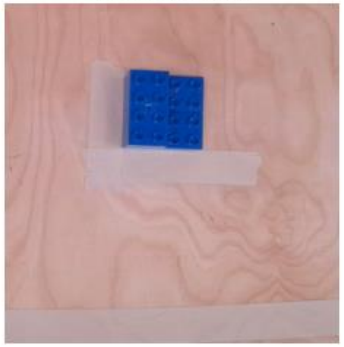


Bad Lightning.

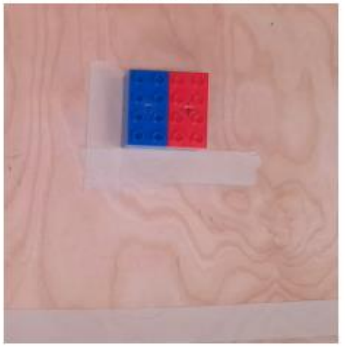
Camera System:Shortcomings of Perception Module

2. For the Assembly Area depth computation, the lego labels cant detect colour Red due to Infrared rays being absorbed by Red Blocks.

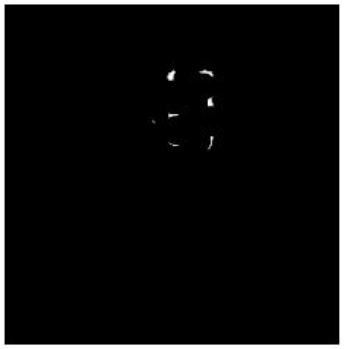
Original Frame



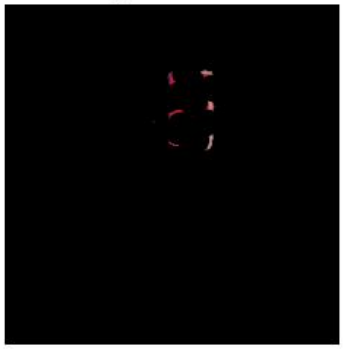
Updated Frame



Difference Mask



Applied Mask



Human Hand Detection and Tracking

Human Hand Detection .

1. Developing a model to detect human hand using depth information from the camera

- Research and literature survey on proven models for depth based data

2. Training the model

- Finding a suitable dataset for training, and training on pytorch

3. Testing on the real-setup

- test and evaluate in the assembly set-up

Human Hands Detection and Tracking

Dataset : MSRA Hands -
https://jimmysuen.github.io/txt/cvpr14_MSRAHandTrackingDB_readme.txt

MSRA Hands is a dataset for hand tracking. In total 6 subjects' right hands are captured using Intel's Creative Interactive Gesture Camera. Each subject is asked to make various rapid gestures in a 400-frame video sequence.

Each frame contains depth map of size 320 X 240 (only D channel) of a single hand in various poses. And location of 21 key points for each frame.

Model Description:

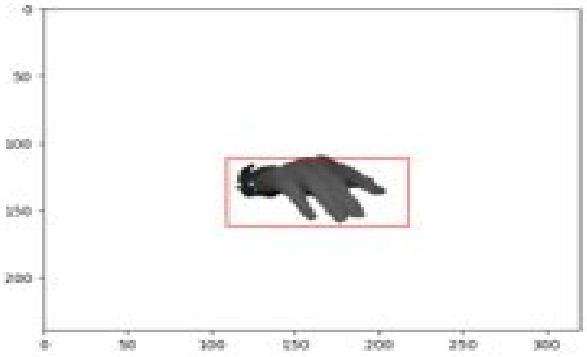
We used Keypoint RCNN from the model zoo of pytorch with ResNet 50 backbone.

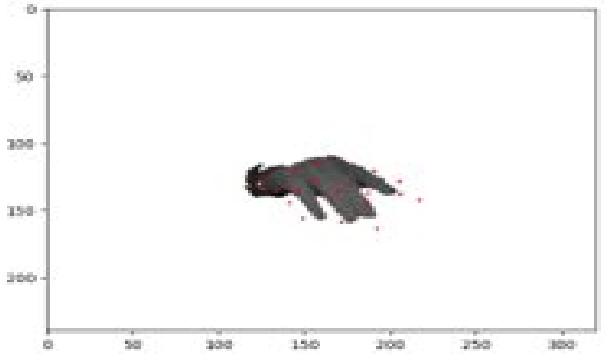
Training :

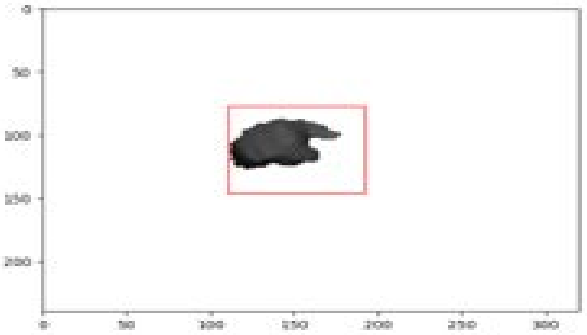
The model is trained on a CPU for 37 epochs, which took 40 hrs.

Batch-training was used, with batch-size being 10 frames.

Results on test data-set



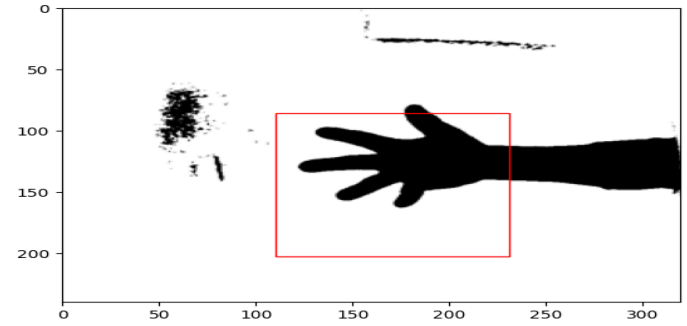


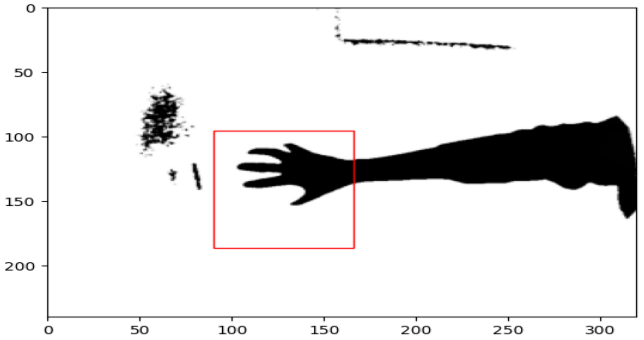


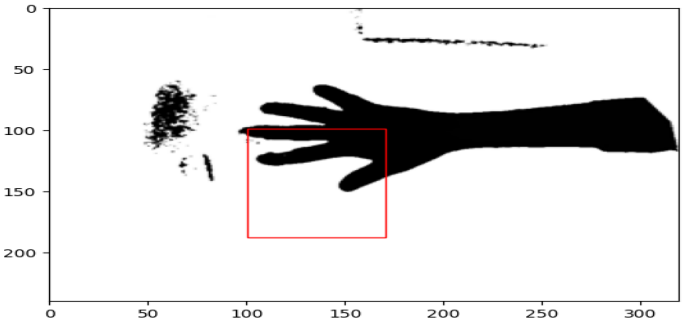
Human Hand Detection and Tracking

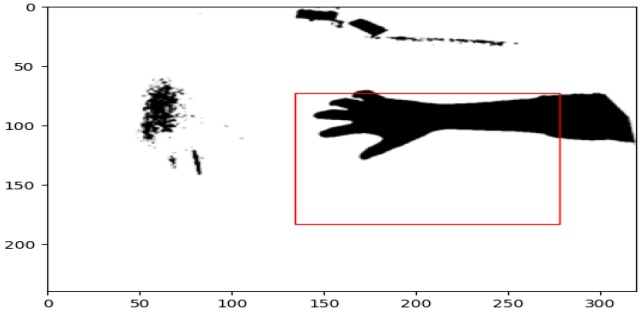
Results on Real set-up :

Here we added a preprocessing step, where background is subtracted from the original frame. To make the image same as in the dataset.









Problems and difficulties faced

- **Limited availability of proper datasets for hand detection containing depth information.**

  As most of the open-source datasets are only RGB images, we were not able to obtain a depth-based dataset of good size for training.

- **No pretrained models on depth-based hand detection:**

  Therefore it was not possible to take advantage of transfer learning to make the model more robust

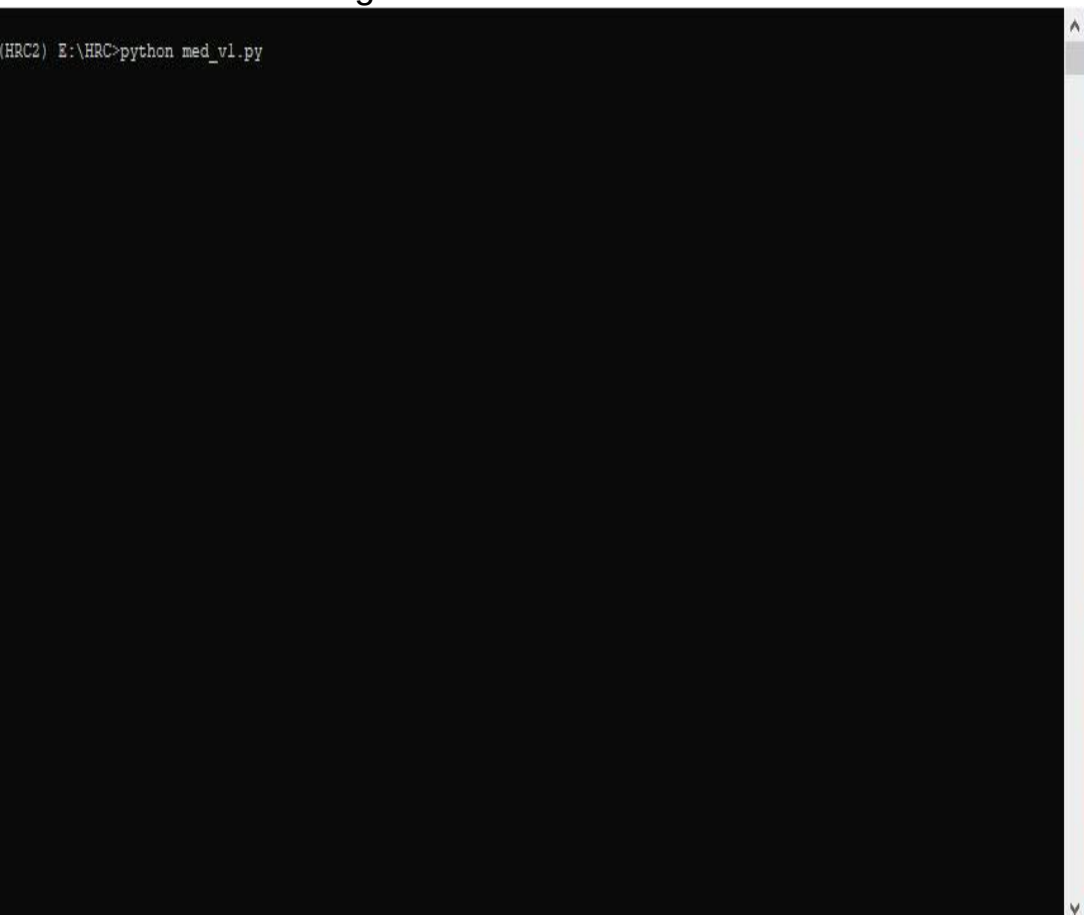
Human Hand Detection and Tracking

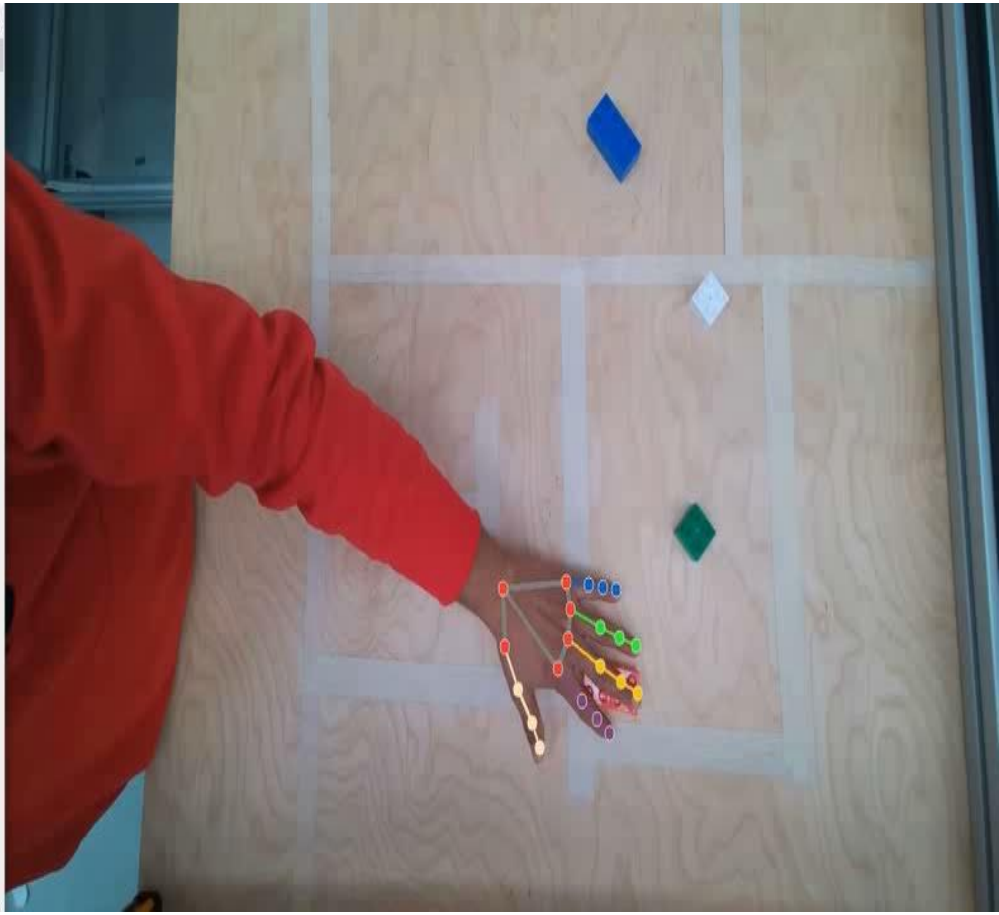
Tracking and prediction human actions:

- Tracking human hands and determining which lego human is going to pick by continuous observation of relative distances between hand and the lego blocks.
- Verify the predicted outcome by scanning the assembly area after human has left the area.
- Convey this information to the system to plan next steps to the assembly.

Human Hand Detection and Tracking

- Hand tracking algorithm using Google’s open source tool - Mediapipe-Hands.
- Here the algorithm predicts which lego human hand is trying to pick based on relative distance between hand and the lego blocks.





Digital Twin

Digit Twin is a virtual representation of a real device, product or process .It covers useful information aspects of the real asset through its whole life-cycle (design, engineering, production, maintenance) while being connected to the real asset (real-time 2-way communication) .

Why Digital Twin?

Analysis of current and historical data

- Predictive maintenance
- Fault/anomaly detection
- Shop Floor Performance Improvement

Remote Operations

- Remote collaboration (e.g. using VR technology)
- Remote maintenance
- Remote control

Real-time monitoring (Dashboards, Visualizations)

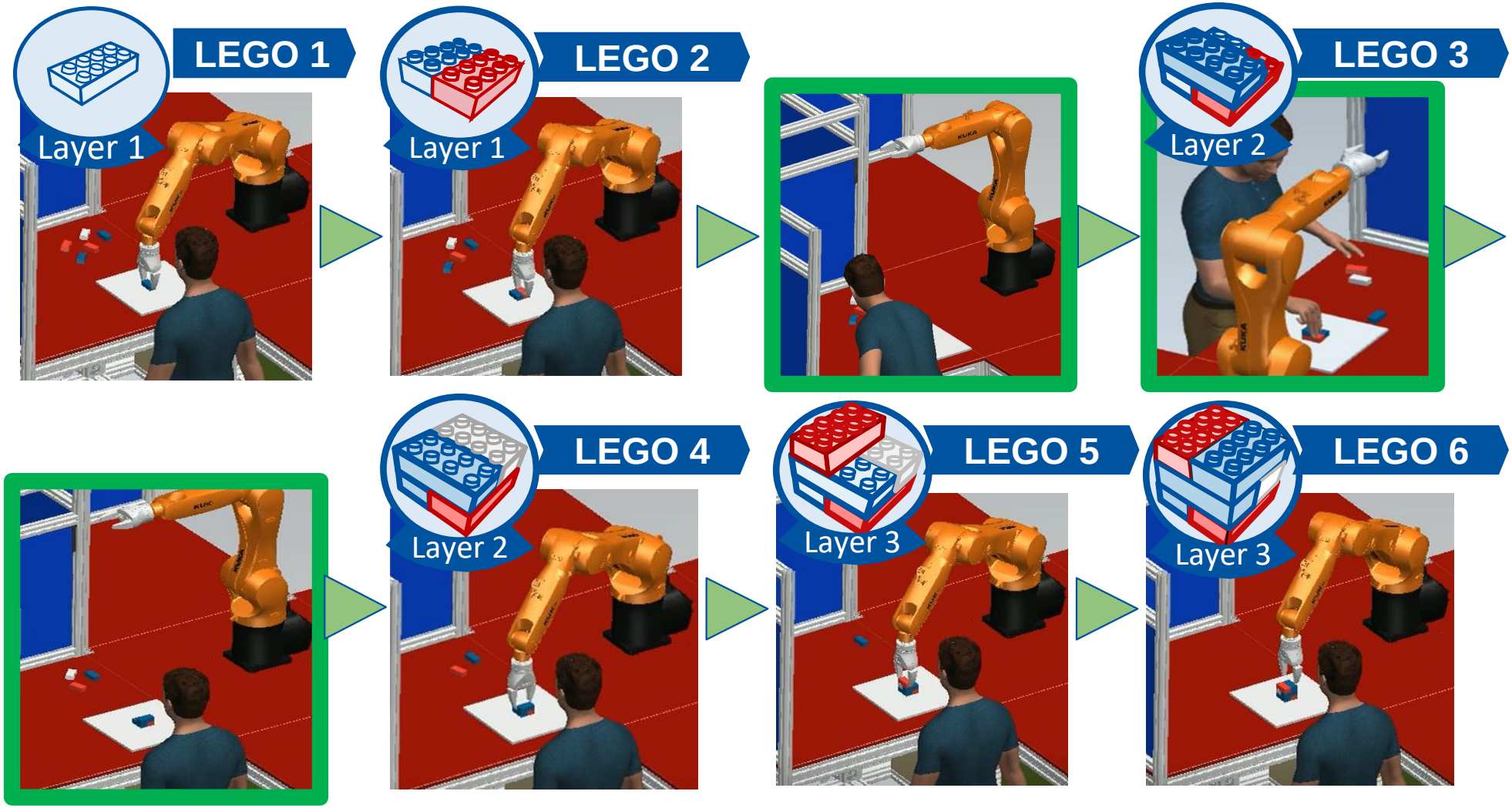
Digital Twin - Planning

Siemens Tecnomatix Process Simulate used as Digital Twin to simulate the process.

Simulation of the Lego assembly by robot hand is imported from Tecnomatix Robot Expert to Process Simulate.

Use Case of Human Intervention to help robot for Lego assembly is analyzed.

Digital Twin - Cooperation Point

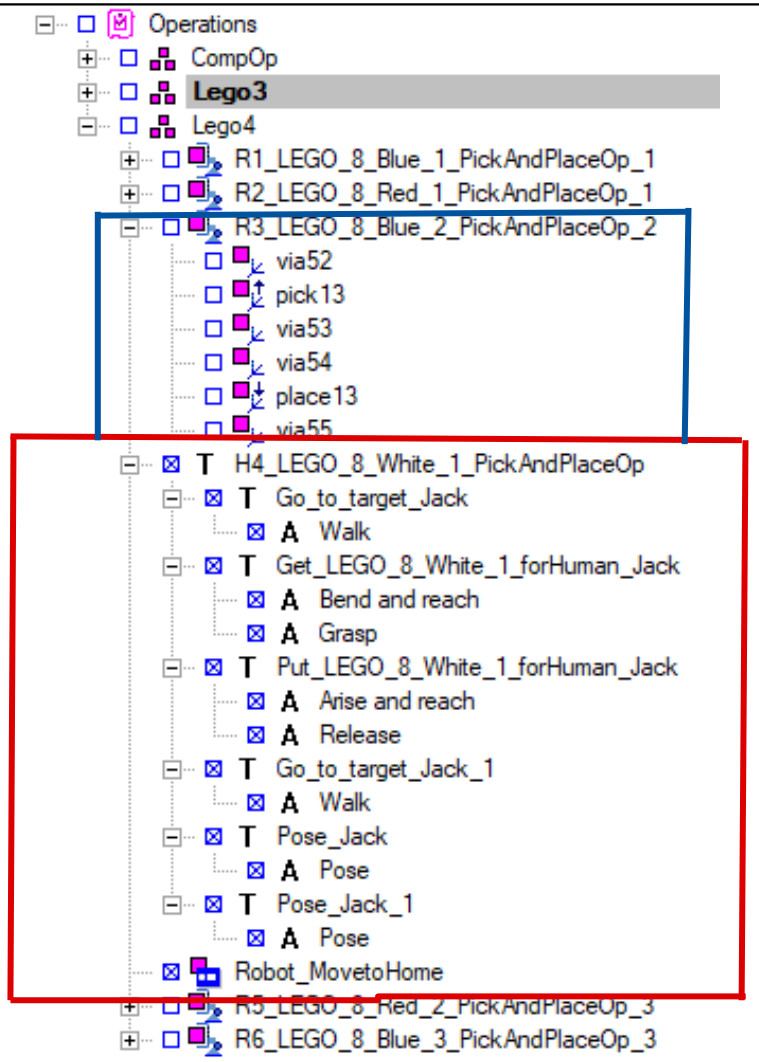


ZeMA
Zentrum für Mechatronik
und Automatisierungstechnik
M:sys

Questions arised:

- Human avatar.
- Posturing
    - Body posturing while walking towards assembly area, reaching the Lego parts, while assembly the Lego, walking back from assembly area.
    - Hand posturing while picking and placing the Lego, grasping of the Lego.
- Set Timing or pace for each Human movement.
- Via points/ trajectory of hand between pick and place coordinates.
- Action taken by Robot as soon as it detects human intervention.

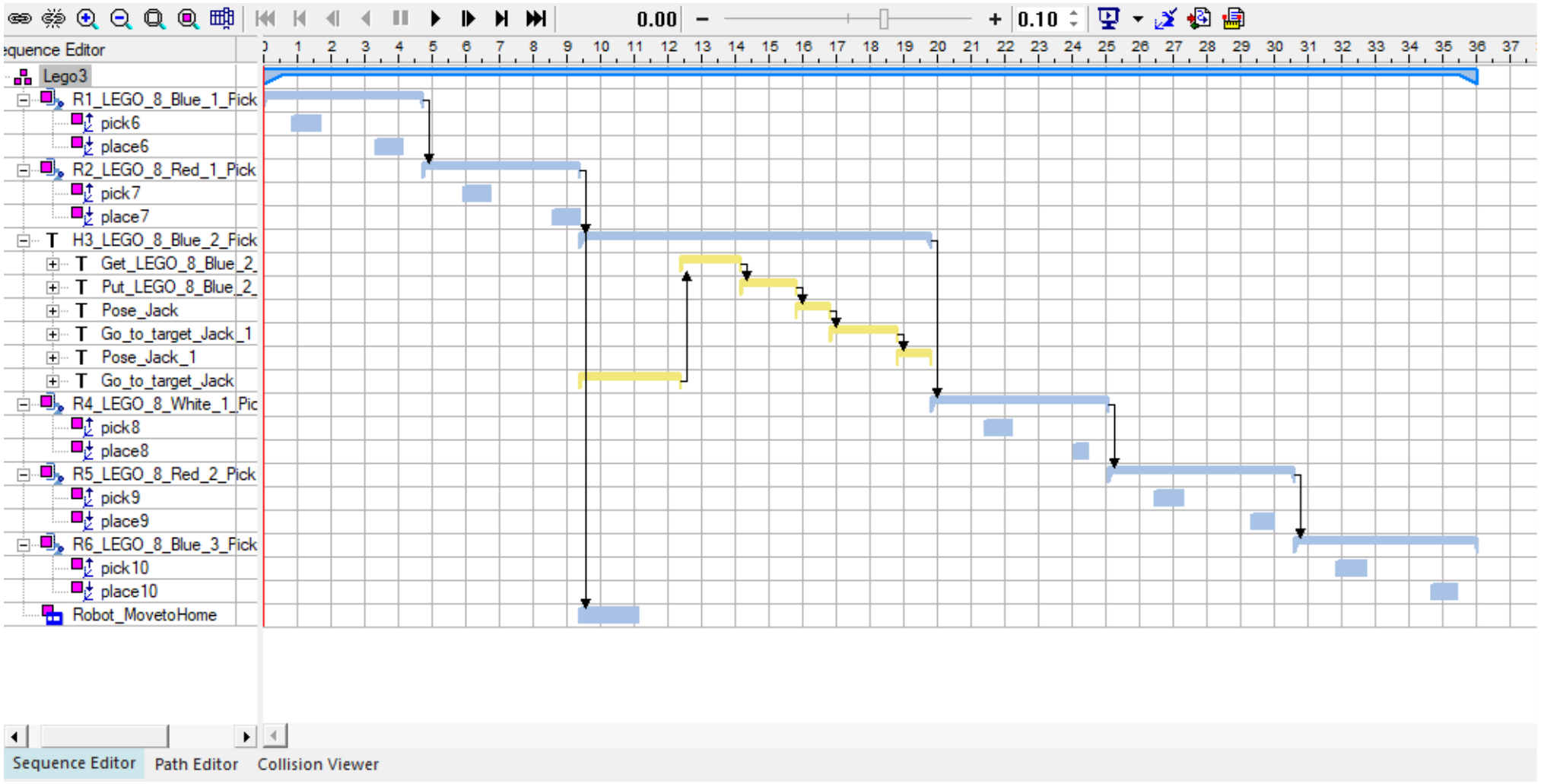
Operation tree



Operation tree for Robot task simulation

Operation tree for Human task simulation

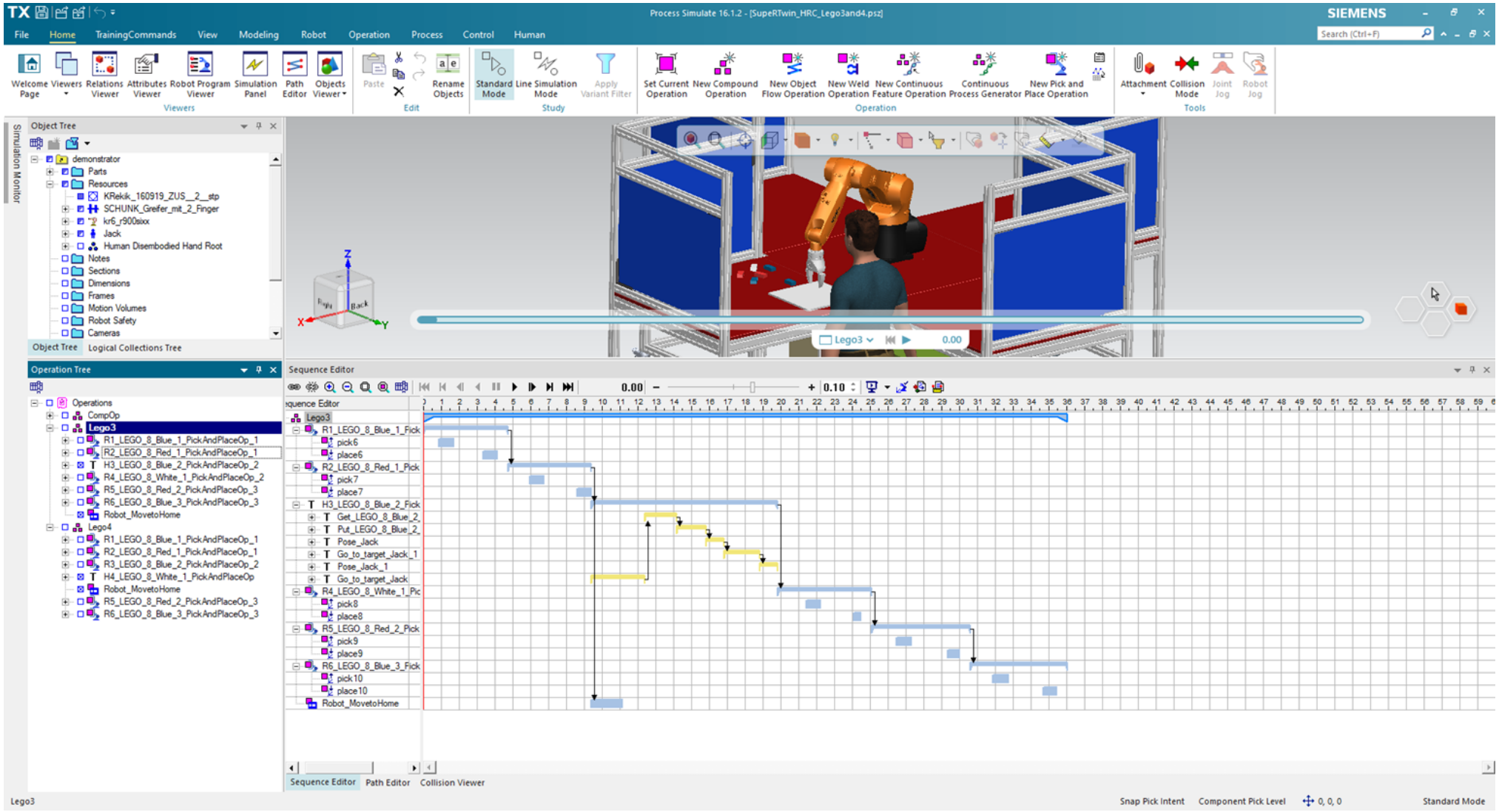
Sequence Diagram of complete simulation



ZeMA Zentrum für Mechatronik und Automatisierungstechnik

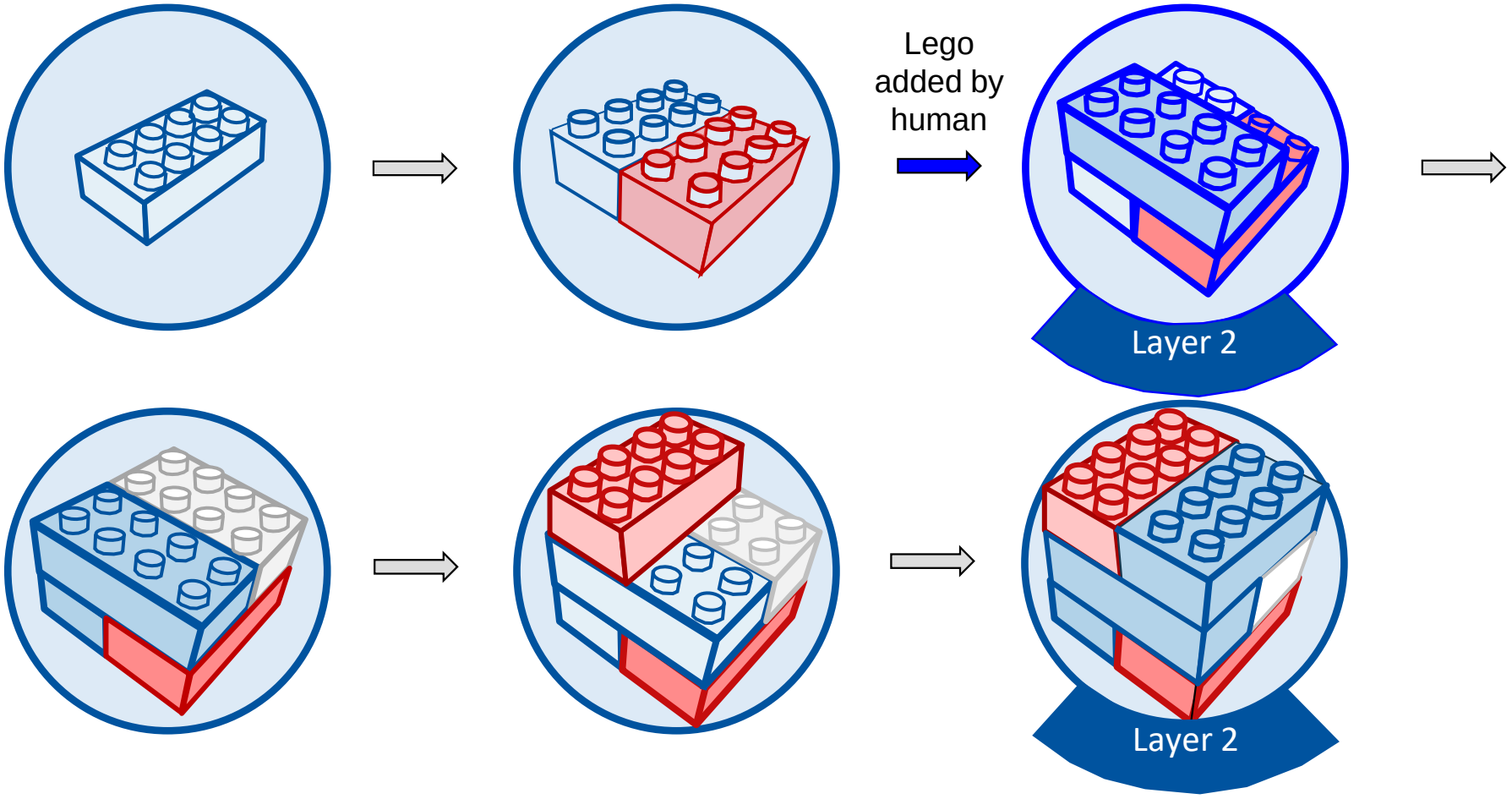
M:sys

Process Simulate User Interface

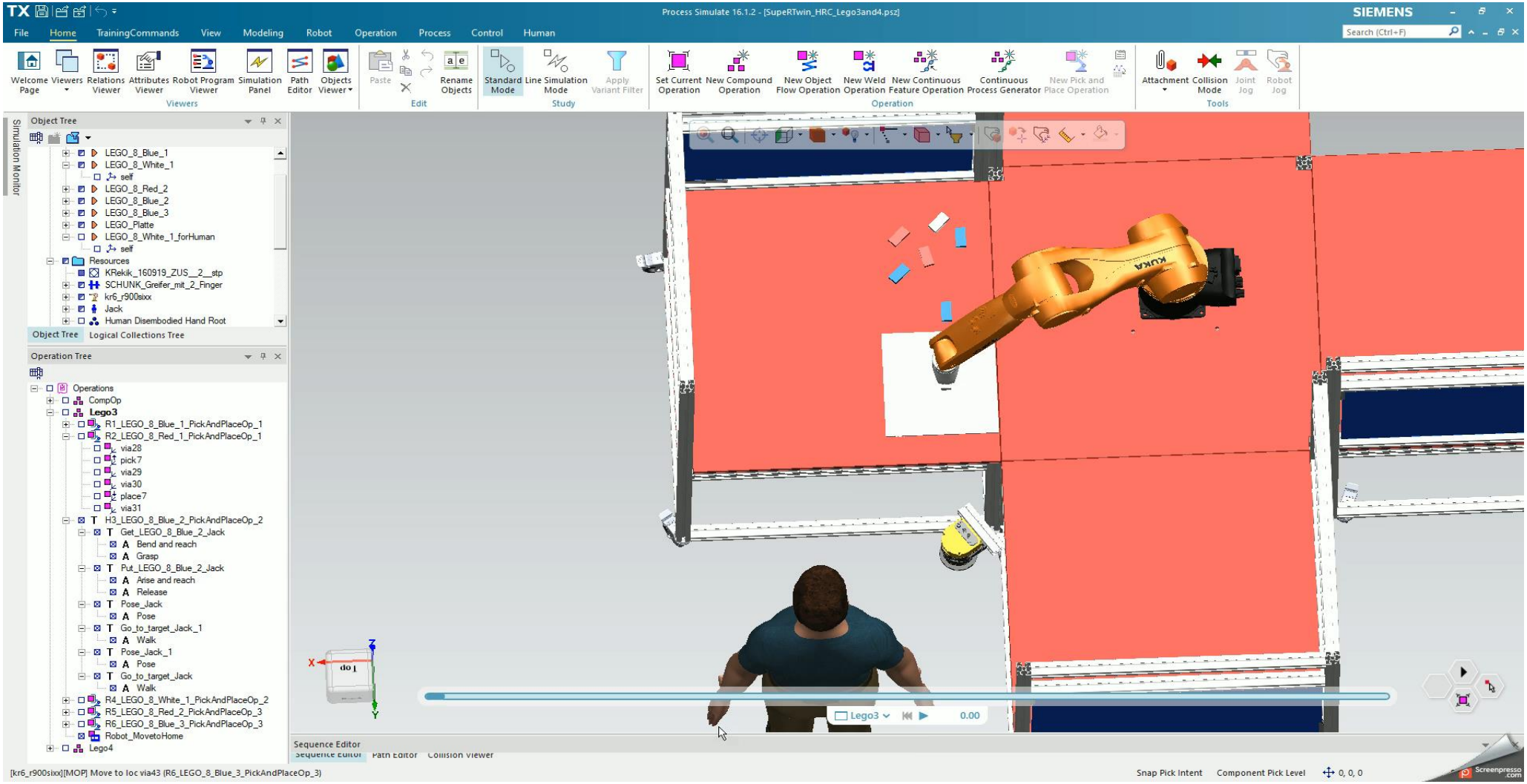


Zentrum für Mechatronik und Automatisierungstechnik

Cooperation Point (Human Intervention) - Sequence 1

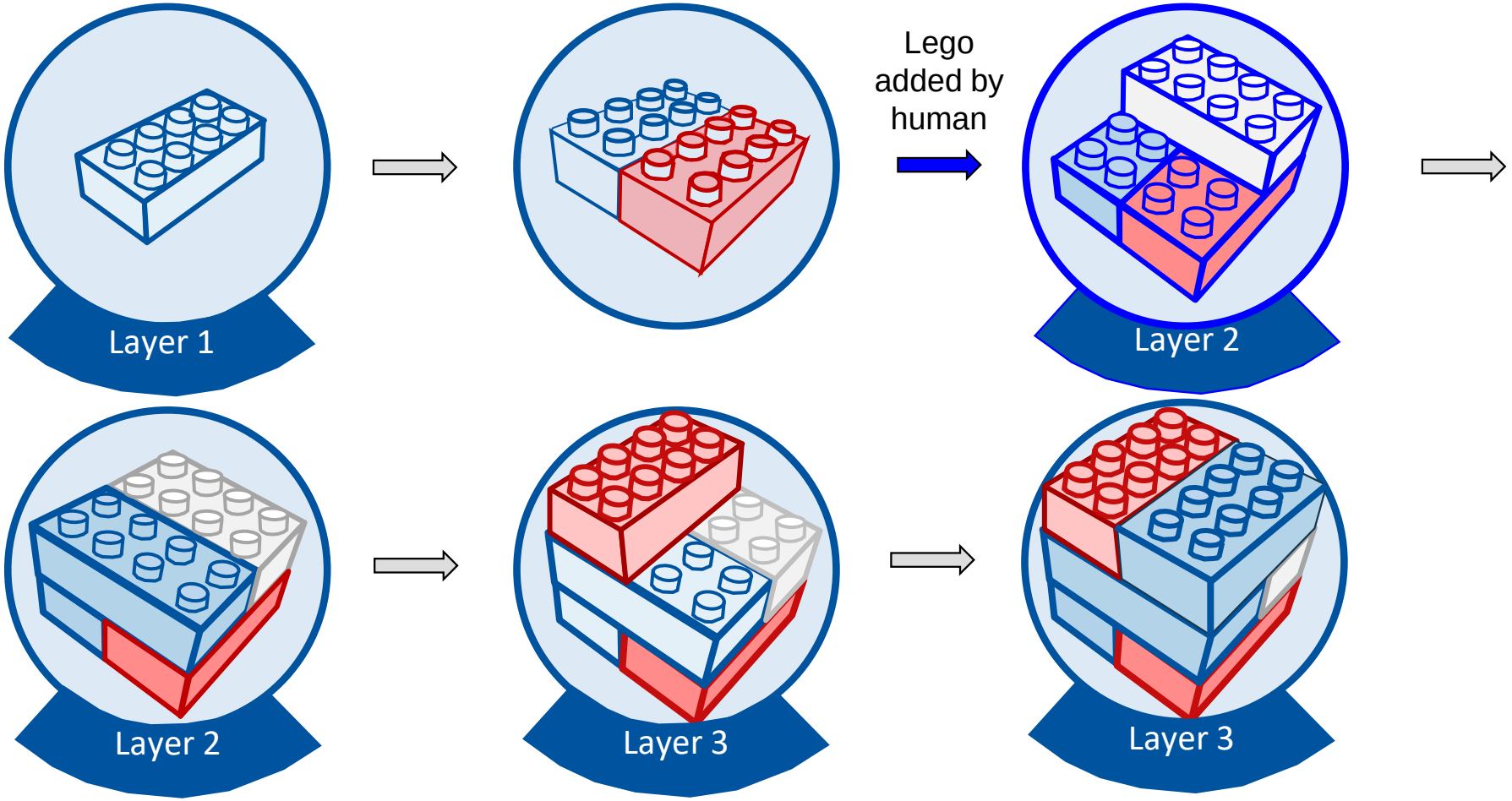


Human Robot Cooperation simulation - Sequence 1

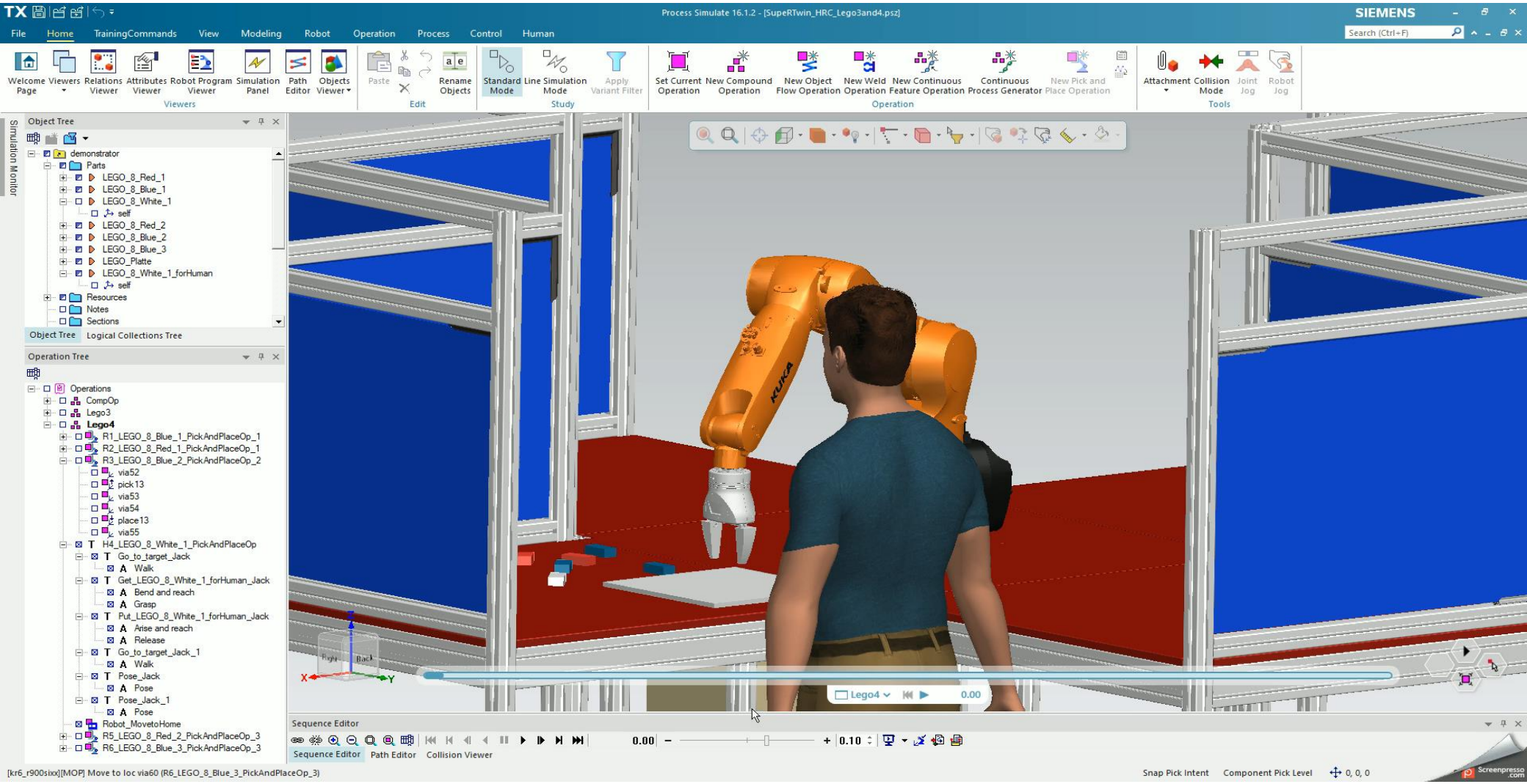


Zentrum für Mechatronik und Automatisierungstechnik

Cooperation Point (Human Intervention) - Sequence 2



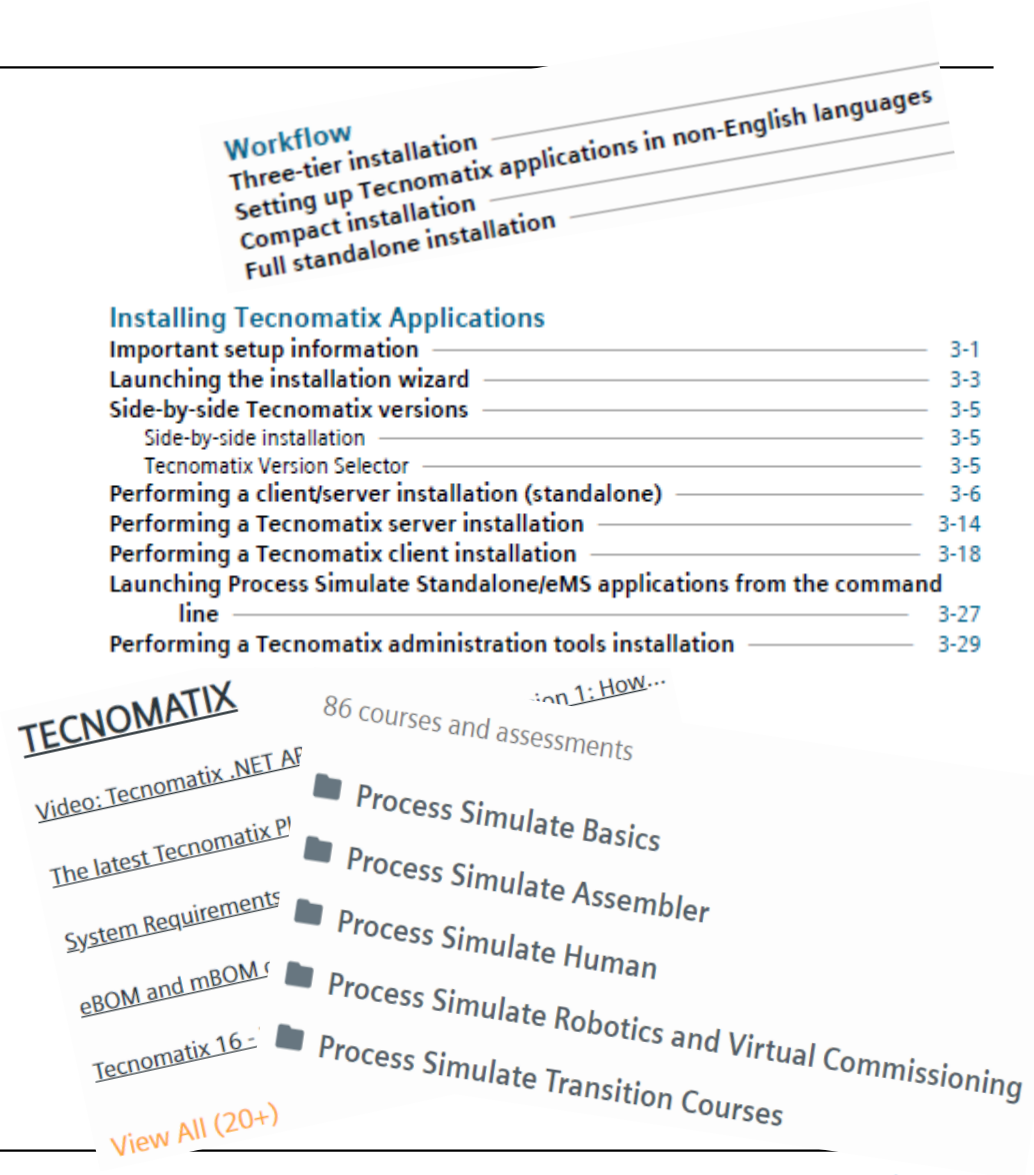
Human Robot Cooperation simulation - Sequence 2



Zentrum für Mechatronik und Automatisierungstechnik

Challenges Faced

- Installation of Process Simulate
  - New to the environment.
  - Three-tier Installation.
  - Lengthy installation document. (94 pages)
  - Decision: Various Options to choose.
- Large amount of video and document tutorials.
- To Design realistic Human postures, pace and movements.



Thank you