Interreg

EUROPEAN UNION

Grande Région | Großregion

Robotix-Academy

Fonds européen de développement régional | Europäischer Fonds für regionale Entwicklung

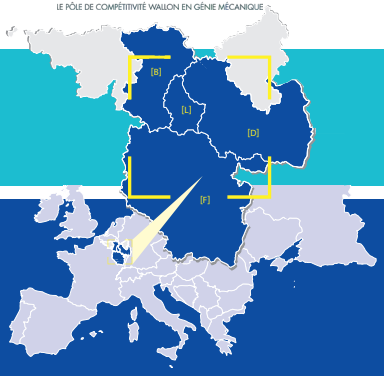
Axe prioritaire | Prioritätsachse 4
Compétitivité et attractivité
Wettbewerbsfähigkeit und Attraktivität

Robotix-Academy Conference for Industrial Robotics (RACIR) 2021

Partenaires du projet | Projektpartner:

ZeMA · uni.lu Université du Luxembourg · Liège université · Université de Lorraine · Umwelt-Campus Birkenfeld | Hochschule Trier · Pôle MecaTech – Le pôle de compétitivité wallon en génie mécanique

Rainer Müller, Peter Plapper, Olivier Brüls, Wolfgang Gerke, Gabriel Abba, Matthias Vette-Steinkamp, Ali Kanso (Hrsg.)



www.robotix.academy

Robotix-Academy Conference for Industrial Robotics (RACIR) 2021

Preface:

The Robotix-Academy Conference for Industrial Robotics (RACIR) is held on September 22, 2021 at Trier University of Applied Sciences, Environmental Campus Birkenfeld, Germany as a hybrid conference.

The Environmental Campus Birkenfeld was founded in 1996 and is the greenest university in Germany. Currently, around 2,300 Bachelor and Master students from more than 80 countries are studying on campus in the fields of environmental planning/environmental technology and environmental economy/environmental law.

The campus is internationally networked, with the strength of strong, regional roots. The students come from all over the world to Hoppstädten-Weiersbach - due to the groundbreaking and future-oriented education.

RACIR was moderated by Prof. Dr.-Ing. Matthias Vette-Steinkamp and accompanied by a roadshow of various manufacturers of robots and related equipment.

The topics concerned by RACIR are: robot design, robot kinematics/dynamics/control, system integration, sensor/ actuator networks, distributed and cloud robotics, bioinspired systems, service robots, robotics in automation, biomedical applications, autonomous vehicles (land, sea and air), robot perception, manipulation with multifinger hands, micro/nano systems, sensor information, robot vision, multimodal interface and human-robot interaction.

Acknowledgements:

The Robotix-Academy partners and the participating students are acknowledged for their contributions and participation to the conference.

The organisation committee and involved persons are also acknowledged for their help and support.

Content

An approach to overcome challenges prior to the actual peg-in-hole problem for the lithium-ion battery pack assembly

Böker, Moritz Emanuel
Faculty of Mechanics and Electronics
Heilbronn University
Heilbronn, Germany
moboeker@stud.hs-heilbronn.de

Joos, Markus
Faculty of Mechanics and Electronics
Heilbronn University
Heilbronn, Germany
joos_markus@t-online.de

Prof. Dr.-Ing. Timo Hufnagel
Faculty of Mechanics and Electronics
Heilbronn University
Heilbronn, Germany
timo.hufnagel@hs-heilbronn.de

***Abstract*—Lithium-ion battery packs based on the 18065 cell power countless electronic devices. These battery packs consist of cell holders containing holes where the batteries are inserted into. Each battery pack has its own shape, size and geometric constraints. Mostly these injection molded parts were designed to be assembled manually and not autonomously. This paper introduces an advanced human-inspired compliant approach to solve the lithium-ion battery based peg-in-hole problem. The here presented methods do not focus on how to center and lower the peg into the hole but rather on bringing the peg to the hole if there are displacement errors of up to several millimeters. Also additional challenges arising in the battery pack assembly are overcome: Despite a reduced number of mounting directions for a tilted peg and pins protruding from the hole surface the battery cell is guided to the edge of the hole. When it has arrived there, common peg-in-hole strategies explained in other publications can be applied to center and lower the peg and thus complete the assembly. All algorithms are force and geometric constraint based only - no vision system is needed! They were implemented and tested on a specially developed industrial robot plant in cooperation with the German battery pack manufacturer ANSMANN AG with a UR5e collaborative robot it is able to assemble a variety of battery packs differing in size, shape and other parameters. The inbuilt coarse force torque sensors of the UR5e are sufficient. The industrial plant has been tested for several month while mounting thousands of battery cells with a success rate of more than 99.5%.**

I. Introduction

Since the end of the last century until the very day, lithium-ion battery packs power a variety of not only handheld devices but also larger consumer goods. Their use and comfort is indispensable in our daily life: from our phones and laptops being the constant companion during the work day, until the electric car, bicycle or the vacuum cleaner we use at other times. Many but not all of these applications use the 18065 lithium-ion battery cells, which are connected in series and parallel yielding a powerful battery pack.

There are two strategies to design portable and thus battery-based products: The first one is to involve an already developed standard battery pack and to engineer the product around that battery. While the fixed specifications of the standard battery pack reduce the flexibility of the product design, they facilitate mass production because of their widespread application. They are designed to support autonomous assembly by e.g. incorporating chamfered edges.

The second strategy is to develop a product and have its battery designed specifically for it. Thus there are less restrictions during the design phase and the battery seemingly integrates to the product. But the downside is that this battery can only be used in that specific application. If it is not a widespread consumer product it probably will not be manufactured in highly automated mass production plants but still by hand. But how can different small series battery packs originally designed for manual assembly be put together autonomously by one and the same production plant? This industrial plant would need to be highly adaptive and thus be able to solve the well researched peg-in-hole problem in a variety of environmental settings. This is exactly what this paper is about.

II. State of the Art

Solving the peg-in-hole problem is one of the classical disciplines exercised in modern assembly processes involving industrial robots. Especially when it comes to fit a round peg into a round hole with a certain clearance between them, there are various successful approaches already applied in many industries throughout the globe. When there is a huge misalignment of the peg and the hole in addition to a vast search region caused by an extremely small clearance, [CB01] proposes ”blind searching strategies” without the aid of visual assistance. Several approaches define a trajectory to be travelled by the end effector (eef), covering the entire discretized search region:

The search region can be covered by joining the discretized points resulting in a zig-zag trajectory, or by using concentric circles with a spacing of double the clearance or a spiral path. [CB01] points out that an even more effective way is to tilt the peg ”into the hole to infer the direction of misalignment” as humans naturally do when inserting a peg into a hole. A prerequisite is that for the starting position the peg and the hole need to have an initial overlap. In more than 60 trials the success rate was $100\,\%$ with an average assembly time of

7.1 s, almost seven times faster than a worst-case spiral search.
Another approach is done by [Li+17]. Their human-inspired compliant peg-in-hole strategy is also based on a tilted peg and uses a coarse force sensor. According to [Li+17] a human hand and arm is superior to industrial robots: While a human has a much lower precision and stiffness than industrial manipulators, he still can assemble a peg into a hole even with the smallest clearance. This is because of his compliance caused by the muscles and tensions which can be adapted to an industrial robot.
There are three contact types of a tilted round peg with a round hole: a one, two and three points contact. When the peg contacts the hole at one or two points, the force acting on it can be measured with a coarse force sensor. While moving downwards the horizontal force direction "helps to slide the peg to achieve a three points contact" [Li+17].
While there a many possibilities for one point and two points contacts, the three points contact exists only once. It is established when the tilted peg has reached the lowest point within the hole. The three-dimensional clearance of the tilted peg within the hole is called the "constraint region". When the minimum of the constraint region is reached, the peg can be tilted back so that its longitudinal axis aligns with the center axis of the hole. Then it can be inserted completely into the hole [Li+17].
The human-inspired method by [Li+17] even works without jamming, when the clearance of the peg and the hole is smaller than the repeatabilty of the robot. The downsides are the low speed (about 40 s for one assembly) and the prerequisite that the starting position must lie in the constraint region. The here presented methods however concentrate on how to bring the peg to the constraint region in the first place.
[JPV17] deals with the autonomous assembly of flexible rubber pegs with a force-guided robot. By using a Gaussian mixture model the measured wrench (force and torque) acting on the peg can be analyzed and the discrete contact states of the peg and the hole being classified. In [JP13] each contact state is evaluated by using a Takagi-Sugeno fuzzy inference system that takes the wrench and pose (position and orientation) as an input and returns the model type as an output.
In general flexible parts are more challenging to handle as they exhibit smoother force transitions when a contact is established. Lithium-ion batteries in contrast are rather stiff, while the injection molded cell holders may deform more easily.
The peg-in-hole problem can also be solved by reinforcement learning (RL) algorithms [Ino+17; Bel+20], penalizing the robot for wrong actions or rewarding it for good ones. [Ino+17] trains and deploys two RL neural networks, one doing the search (bringing the peg to the hole until their centers coincide) and the other one the final insertion. These RL algorithms have been tested on a robot having an accuracy of 60 µm and a peg-in-hole clearance as low as 6 µm. The average assembly time is less than 5 s at 100 % success rate.

III. Scientific Gap

At first glance the peg-in-hole problem described in this section resembles the standard peg-in-hole problem for a cylindrical peg and hole. But eventually it will become obvious that further challenges increase its complexity and that the previously mentioned solutions are not sufficient alone to overcome them.

A. The peg

The cylindrical 18065 lithium-ion battery cell (see figure 1a) is the peg, having the following specifications:

- Diameter: $d_{\text{peg}} = 18.3 \pm 0.2\,\text{mm}$
- Edge: chamfered (chamfer radius: $0.6 \pm 0.1\,\text{mm}$)
- Purchased standard part – cannot be modified

B. The hole

The battery cells are inserted into various-shaped battery cell holders. They may be rectangular (see figure 1b), round or arbitrarily shaped and have a different numbers of holes to insert the batteries into. The specifications of the holes are:

- Diameter: $d_{\text{hole}} = 18.7 \pm 0.2\,\text{mm}$
- Edge: sharp
- Injection molded part – cannot be modified

Therefore the clearance between the peg and the hole results in: $c = 0.4 \pm 0.4\,\text{mm}$.

C. The UR5e robot arm

In order to solve this peg-in-hole problem, a UR5e industrial collaborative robot arm from Universal Robotics was chosen which has coarse force torque sensors built into its joints. Their specifications are [Rob]:

Sensor type	Range	Resolution	Accuracy
Force x, y, z	50 N	2.5 N	4.0 N
Torque x, y, z	10 Nm	0.04 Nm	0.30 Nm

D. Further challenges

The battery cells must be assembled in a certain order into the cell holder. They are inserted into the holes from the inside to the outside. If instead the outer holes were loaded first, there would be no chance anymore to reach the inner holes with a tilt-based search strategy. But even when the cell holder is assembled from the inside to the outside, the already mounted cells prohibit the cells to come to be assembled from certain tilt directions. This is called "occlusion by cells" from now on.
The battery pack consists of two interlocking cell holders: one being used to assemble the battery cells and one to be put on top of the mounted batteries afterwards to fasten them from the other side as well. The connection between the two cell holders is established by interlocking pins protruding from the hole surface of the cell holder in between the gaps left out by the inserted batteries (see figure 1b). As before with the occlusion by cells, these pins also restrict the tilt directions the neighboring battery cells could be mounted with. This is

referred to as ”occlusion by pins”.
While the cell holder is assembled, the tension within this injection molded part is rising, causing it to deform slightly and reduce the peg-in-hole clearance.
Last but not least, lithium-ion battery cells must be handled with extreme care, as they can catch fire if deformed, short-circuited or heated. Damage to the cells must therefore be prevented in any case.
Thus, the following requirements for a peg-in-hole solution in the battery pack assembly are derived::

1) The tilt direction a battery is assembled with must be adjustable for each battery individually to overcome the occlusion by cells.
2) Pins protruding from the battery holder surface must be detected and avoided in order to cope with the occlusion by pins.
3) The battery must be handled with great care throughout ”from pick to place”.
4) The assembly must be both as fast and as reliable as possible. A human is able to mount one cell in about three seconds in average.

The authors of this paper propose the hypothesis, that there is a human-inspired method that mostly fulfills the requirements defined above. As every human is capable of inserting a round peg into a hole blindfolded - just by having a force-feedback in its finger tips and geometric assumptions of the environment - a robot can be trained to do the same with a force torque sensor in its joints and by knowing some key parameters of the peg and hole scenario. Even when large displacement errors are present and no camera system is deployed.
The remaining of this paper is on how to solve the peg-in-hole problem in a human-inspired way taking the requirements mentioned above into account.

(a) The 18065 lithium-ion battery cell

(b) The battery cell holder

Fig. 1: The peg-in-hole setup

IV. Overall peg-in-hole procedure

The robot picks up a lithium-ion cell from the stack and moves to the calculated position of the desired hole. This predetermined and thus theoretical position deviates from the actual hole position of the cell holder by a few tenths up to a few millimeters.

a) Going-down-until-collision procedure: The robot then tilts the gripped cell by a tilt angle of $\beta_{\text{peg}} = 22.5°$ away from the vertical z-axis and moves downwards with respect to the cell holder coordinate system. Eventually it collides with the cell holder. The force acting in the vertical upward (-z) direction on the tool center point (TCP) of the robot is measured. The TCP lies at the concentric bottom of the gripped battery (see figure 2a). If the force rises above a given threshold, a collision will be detected.
This procedure is referred to by ”going-down-until-collision procedure” from now on. A threshold of $2.5\,\text{N}$ has proven well in practice.

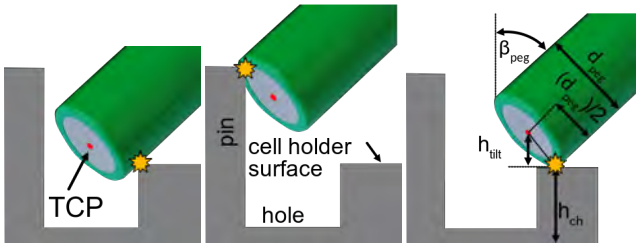
$$threshold_{\text{force(-z)}} = 2.5\,\text{N}$$
$$collision = force_{\text{(-z)}} > threshold_{\text{force(-z)}}$$

By now the robot does not know whether it has collided with the inner edge of the desired hole, the surface of the cell holder or a pin protruding from the cell holder. The different collision types are called ”hole collision”, ”surface collision” and ”pin collision”, respectively (see figure 2). Each of them demand a specific procedure to eventually assemble the battery. Thus, the robot needs to distinguish between the collision types.
This classification is done by comparing the height h_{coll} of the TCP, at the time the collision has occurred, with a reference height: To determine the reference height h_{ref}, it is assumed that the tilted battery is touching the cell holder surface with its lower edge (see figure 2c). h_{ref} is now the sum of the height h_{ch} of the cell holder from its bottom to its surface and the vertical distance h_{tilt} of the lower edge of the tilted battery up to the TCP (see figure 2c). This is expressed by the following relation

$$h_{\text{tilt}} = \sin(\beta_{\text{peg}} \cdot \frac{d_{\text{peg}}}{2})$$
$$h_{\text{ref}} = h_{\text{ch}} + h_{\text{tilt}} \tag{1}$$

If a collision occurs, the current height h_{coll} of the TCP is compared to the reference height h_{ref} and the collision is classified thereby:

$$h_{\text{tol}} = 2\,\text{mm}$$
$$collision_{type} = \begin{cases} collision_{\text{hole}}, \text{ if } h_{\text{coll}} < h_{\text{ref}} - h_{\text{tol}} \\ collision_{\text{pin}}, \text{ if } h_{\text{coll}} > h_{\text{ref}} + h_{\text{tol}} \\ collision_{\text{surface}}, \text{ else} \end{cases} \tag{2}$$



(a) hole collision (b) pin collision (c) surface collision

Fig. 2: The different collision types and their reference height

Depending on the type of the collision, a different reaction of the robot is demanded:

b) Avoiding-pin procedure: When a pin collision has taken place (see figure 2b), the robot has to avoid the pin as much as possible in the direction that leads to the actual hole . This avoidance maneuver is called "avoiding-pin procedure" and dealt with in chapter V. A collision with a pin is an indicator that the theoretical hole position deviates significantly from the actual one, resulting in a planar distance $d_{\text{hole2hole}}$ between the theoretical and actual hole of up to several millimeters. After the avoiding-pin procedure, the going-down-until-collision procedure (see chapter IV-0a) starts again.

c) Sliding-on-surface procedure: The second case is a surface collision (see figure 2c). This also means that $d_{\text{hole2hole}}$ amounts up to several millimeters. The robot then has to slide the lower edge of the battery on the cell holder surface in the direction that leads to the actual hole. This is called "sliding-on-surface procedure" and is explained in detail in chapter VI. If it registers a drop in the vertical force $force_{\text{(-z)}}$ acting upwards on the TCP, it will be assumed that the lower edge of the battery has reached the edge of the desired hole. If its transition criterion is fulfilled, a going-down-until-collision procedure is undertaken again.

d) Centering-lowering procedure: When a hole collision has occurred (see figure 2a), the calculated hole position matches well with the actual hole, caused by $d_{\text{hole2hole}}$ being just a few tenth of a millimeter. The robot is then centering and lowering the battery within the hole achieving a three points contact which is the desired destination. This procedure is therefore called "centering-lowering procedure" from now on and described in chapter VII.

If the hole has been found successfully at the end of the centering-lowering procedure, the cell will be tilted back into the vertical position. Then it is finally inserted into the cell holder by a specified force. The whole procedure is illustrated in figure 3. The following chapters will concentrate on the avoiding-pin and sliding-on-surface procedure as they show how to approach the peg to the hole. The centering-lowering procedure will be discussed only briefly, since many publications have already dealt with it.

V. Avoiding-pin procedure

Due to process fluctuations, a positional displacement $d_{\text{hole2hole}}$ of up to several millimeters between the calculated and actual hole position has to be expected. In such a case it may occur that the bottom or the edge of the tilted battery collides with one of the pins. The pins are protruding from the cell holder surface and are adjacent to the hole. It is pointed at by the tip of the battery which is also floating roughly above the actual hole. So practically speaking the battery is tilted away from the pin (see figure 2b). Before the battery can be inserted into the hole, the TCP has to avoid the pin first. How it does so will be explained in this chapter.

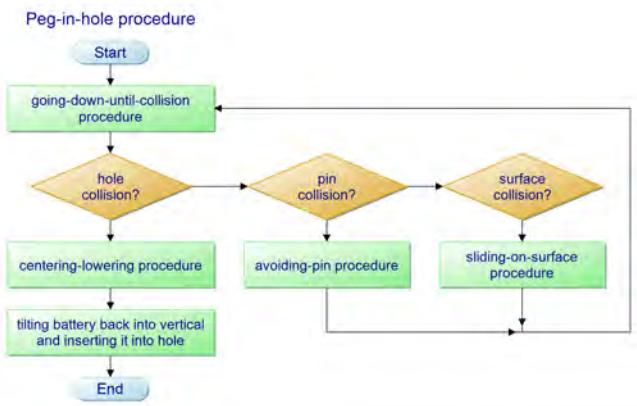


Fig. 3: The overall advanced peg-in-hole procedure

When a pin collision has occurred, the planar (xy) direction of the force caused by the pin acting on the TCP is determined.

$$\theta_{\text{force}} = \text{atan2}(force_{\text{y}}, force_{\text{x}}) \tag{3}$$

If the TCP is moving in the direction of θ_{force}, it reduces the force acting on itself by increasing the distance between the TCP and the pin. Because of the pins lying tangentially at the hole, θ_{force} may already point to the actual hole. But these force readings are not always reliable: The TCP lies in the concentric bottom of the gripped battery. If the collision location of the pin and the battery coincides with the TCP, the lever arm of the force acting on the TCP will be infinitesimally small. Then the values of the forces in x- and y-direction will be close to zero and lead to a rather random force direction θ_{force}. So in short: θ_{force} can point to the actual hole but is not reliable. Because of that there is a need of a second direction, stabilizing the avoiding-pin procedure:
Since the tip of the tilted battery points in the direction of the pin, the TCP has to move to the opposite direction away from the pin in order to come closer to the actual hole (see figure 2b). This direction is called "positive tilt direction" and is expressed by the following relation:

$$\begin{aligned} x_{\text{pos-tilt-dir}} &= \text{sign}(\beta_{\text{peg}}) \cdot \cos(\gamma_{\text{peg}}) \\ y_{\text{pos-tilt-dir}} &= \text{sign}(\beta_{\text{peg}}) \cdot \sin(\gamma_{\text{peg}}) \\ \theta_{\text{pos-tilt-dir}} &= \text{atan2}(y_{\text{pos-tilt-dir}}, x_{\text{pos-tilt-dir}}), \end{aligned} \tag{4}$$

where β_{peg} is the angle the battery is pitched or tilted by from the vertical axis. γ_{peg} is the the direction the tilt is applied to around the vertical axis of the cell holder coordinate system. If the TCP is moving into the direction implied by $\theta_{\text{pos-tilt-dir}}$, it is eventually leaving the pin behind but not necessarily going straight to the actual hole. So in short, $\theta_{\text{pos-tilt-dir}}$ may not always point to the actual hole, but is a constant and thus extremely stable.
In order to improve the avoiding-pin procedure, θ_{force} and $\theta_{\text{pos-tilt-dir}}$ are combined, yielding a resulting angle that ideally has the advantages of both inputs: leading away from the pin

to the actual hole and being robust. This is done by distributing a trust κ onto both directions according to

$$\theta_{\text{avoid-pin}} = \kappa \cdot \theta_{\text{pos-tilt-dir}} + (1 - \kappa) \cdot \theta_{\text{force}} \tag{5}$$

Because θ_{force} and $\theta_{\text{pos-tilt-dir}}$ are angles within $[-\pi, +\pi]$, equation 5 has to be slightly modified. The algorithm 1 shows how to merge two angles $a1$ and $a2$ lying within the range of $[-\pi, +\pi]$, which are trusted with κ and $(1 - \kappa)$ respectively, into a single angle which lies again in a range of $[-\pi, +\pi]$.

Algorithm 1 Merging two angles a1 and a2 with input and output range of $[-\pi, +\pi]$

```
angle = trust * a1 + (1 - trust) * a2
# if condition fulfilled: cond = True
# else: cond = False
cond1 = abs(a1) + abs(a2) > pi
cond2 = sign(a1) * sign(a2) < 0
cond3 = trust < 1.0
if cond1 and cond2 and cond3:
    angle = angle + trust * 2 * pi
endif
```

In order to avoid the pin and move closer to the actual hole, the TCP moves in direction of $\theta_{\text{avoid-pin}}$ for a distance of $step_{\text{avoid}}$ on the horizonal plane. In practice it has proven that $step_{\text{avoid}} = 1\,\text{mm}$ is a good approach. It can be increased further to leave the pin behind even more quickly but should not be chosen to large, as it easily would lead the peg to overleap the hole. After that the robot proceeds with a going-down-until-collision procedure again. At his point the reader may ask what value to assign to the trust κ. This question can be answered as follows:

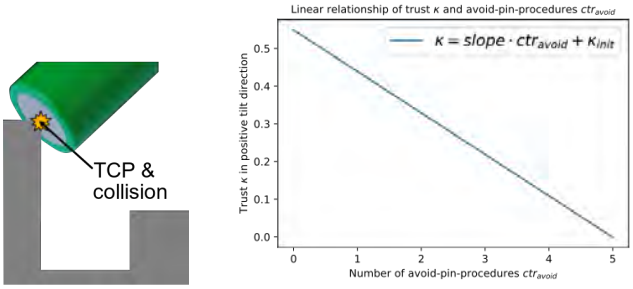
If the calculated hole position is far off the actual hole and extends into a protruding pin ($d_{\text{hole2hole}}$ of some millimeters), several avoiding-pin procedures may be necessary to avoid the interfering pin completely. The force direction will not be meaningful, if the collision location of the pin and the battery coincides with the TCP, yielding a lever arm close to zero (compare figure 2b with 4a). But if the location of the collision is closer to the upper edge of the battery and therefore further away from the TCP, the lever arm is longer and the force direction will be more accurate.

The first time an avoiding-pin procedure is executed, the robot does not know whether the pin collision occurred with the center or the edge of the battery. But the more times a pin collision is registered and thus an avoiding-pin procedure is undertaken, the more it can be assumed that the pin collision is taking place with the edge and not the center of the battery as the robot already has tried to avoid the pin. So the lever arm of the force acting on the TCP rises and the more trust ($(1-\kappa)\uparrow$) can be put in the force direction θ_{force} and the less trust ($\kappa\downarrow$) is put in the positive tilt direction $\theta_{\text{pos-tilt-dir}}$.

The number of avoiding-pin procedures is counted by ctr_{avoid}. As a reference value the maximum number of avoiding-pin procedures allowed before cancelling the mounting process and doing a complete retry is set by max_{avoid}. An initial trust of $\kappa = 0.55$ is given to the positive tilt direction $\theta_{\text{pos-tilt-dir}}$ and thus the force direction θ_{force} is entrusted the remaining $(1 - \kappa) = 0.45$. As ctr_{avoid} is increasing, κ decreases. These relations are defined in the following equations and illustrated in figure 4b:

$$\begin{aligned} \kappa_{\text{init}} &= 0.55 \\ slope &= -\kappa_{\text{init}} / max_{\text{avoid}} \\ \kappa &= slope \cdot ctr_{\text{avoid}} + \kappa_{\text{init}} \end{aligned} \tag{6}$$

This is to help the robot not only avoiding the interfering pin but also going already in the direction of the actual hole in one and the same step. Once the robot has successfully avoided an interfering contour, it continues descending downwards until another collision occurs, which may be a surface collision which the next chapter is about.



(a) pin collision at TCP

(b) The relationship of trust κ and number of avoiding-pin procedures ctr_{avoid}

Fig. 4: Graphics illustrating the avoiding-pin procedure

VI. Sliding-on-surface procedure

If the lower edge of the battery has collided with the surface of the cell holder in proximity around the actual hole (see figure 2c), the collision height h_{coll} is within the tolerance band of the reference height h_{ref}:

$$h_{\text{ref}} - h_{\text{tol}} \leq h_{\text{coll}} \geq h_{\text{ref}} + h_{\text{tol}}$$

So there is a positional displacement $d_{\text{hole2hole}}$ of some millimeters. Now the battery should slide on the surface until the actual hole is found. But how to find the actual hole if there is no vision system deployed?

The tip of the tilted battery points in the direction of the pin protruding from the cell holder (see figure 2c). So its lower edge can only come into contact with the cell holder surface where there is no pin. Practically speaking this is either on the left or right side, or in front of the hole. But not behind the hole, as there lies the pin. So if the battery follows the direction its tip is pointing to on the horizontal plane, there will be a chance of approaching the hole. This direction is

the opposite of the positive tilt direction $\theta_{\text{pos-tilt-dir}}$ and is thus called negative tilt direction:

$$\begin{aligned} x_{\text{neg-tilt-dir}} &= \text{sign}(\beta_{\text{peg}}) \cdot \cos(\gamma_{\text{peg}}) \\ y_{\text{neg-tilt-dir}} &= \text{sign}(\beta_{\text{peg}}) \cdot \sin(\gamma_{\text{peg}}) \\ \theta_{\text{neg-tilt-dir}} &= -\text{atan2}(y_{\text{neg-tilt-dir}}, x_{\text{neg-tilt-dir}}), \end{aligned} \tag{7}$$

There are two cases that need to be dealt with when trying to find the actual hole: The first one is that no pin collision has occurred before the surface collision. This means that the position of the TCP coincides with the precomputed hole position on the horizontal plane ($d_{\text{TCP2hole}} \approx 0\,\text{mm}$). But the precomputed hole position is displaced from the actual hole position ($d_{\text{hole2hole}}$ can amount up to several millimeters). Otherwise the battery would already touch the hole edges and not the surface of the cell holder. This case is called "no-pin-avoided case".

The second case is entered when one or several pin collisions have taken place before the surface collision. This time however, the TCP and the precomputed hole position do not coincide due to the avoiding-pin procedures which have moved the TCP away from the precomputed hole position (d_{TCP2hole} can amount up to several millimeters). Again, the precomputed hole position is displaced from the actual hole position ($d_{\text{hole2hole}}$ can amount up to several millimeters). This case is the "pin-avoided case".

As for the no-pin-avoided case it is hard to extract even more information from the environment apart from $\theta_{\text{neg-tilt-dir}}$ that would help finding the actual hole. While the forces acting on the TCP when the battery is sliding on the surface help to indicate when the hole itself is reached, they are not profitable in finding the direction to the hole. Therefore nothing else is left but moving the TCP along $\theta_{\text{neg-tilt-dir}}$. Therefore,

$$\theta_{\text{slide-surface}} = \theta_{\text{neg-tilt-dir}} \tag{8}$$

Now a transition criterion is needed which indicates when the actual hole has been found and the sliding-on-surface procedure is finished. During the sliding on the surface, a force is acting on the TCP in the vertical direction upwards. But if a hole is reached this force will drop to approximately zero because the lower edge of the battery crosses the edge of the hole and is now hovering above it. A threshold of $1.5\,\text{N}$ has proven practical.

$$\begin{aligned} threshold_{\text{force}_z} &= 1.5\,\text{N} \\ hole_{\text{found}} &= force_z < threshold_{\text{force}_z} \end{aligned} \tag{9}$$

When this is happening, the sliding-on-surface procedure is finished and the centering-lowering procedure can be activated in order to achieve a three points contact with the hole (see chapter VII).

If instead the pin-avoided case has occurred, it will be possible to gain more information from the current status. Here, the TCP and the calculated hole position differ from each other. Thus the horizontal distance between the two is

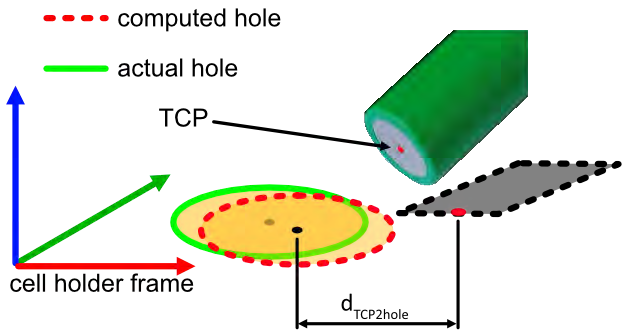


Fig. 5: The planar distance between the TCP to the computed hole

determined as follows:

$$\begin{aligned} pos_{\text{TCP}} &= (x_{\text{TCP}}, y_{\text{TCP}}, z_{\text{TCP}}) \\ pos_{\text{hole}} &= (x_{\text{hole}}, y_{\text{hole}}, z_{\text{hole}}) \\ d_{\text{TCP2hole}} &= \sqrt{(x_{\text{hole}} - x_{\text{TCP}})^2 + (y_{\text{hole}} - y_{\text{TCP}})^2} \end{aligned} \tag{10}$$

This relation is also depicted in figure 5.

Furthermore, the angle from the TCP to the theoretical hole position is:

$$\theta_{\text{TCP2hole}} = \text{atan2}(y_{\text{hole}} - y_{\text{TCP}}, x_{\text{hole}} - x_{\text{TCP}}) \tag{11}$$

If the distance of the TCP to the theoretical hole is far greater than the distance between the theoretical and actual hole ($d_{\text{hole2hole}} >> d_{\text{TCP2hole}}$), θ_{TCP2hole} not only points to the theoretical hole but also roughly to the actual hole from a distant point of view. This effect decreases as the TCP approaches the actual hole.

Therefore the smaller d_{TCP2hole} is, the less θ_{TCP2hole} can be trusted. Thus, the negative-tilt direction $\theta_{\text{neg-tilt-dir}}$ becomes more important while the TCP is coming closer to the theoretical hole (and d_{TCP2hole} is decreasing). Therefore, both angles $\theta_{\text{neg-tilt-dir}}$ and θ_{TCP2hole} are fused into a single direction by involving again a trust value κ:

$$\theta_{\text{slide-surface}} = \kappa \cdot \theta_{\text{TCP2hole}} + (1 - \kappa) \cdot \theta_{\text{neg-tilt-dir}} \tag{12}$$

At this point the reader may ask again what value to assign to the trust κ. In the avoiding-pin procedure the maximum permitted number max_{avoid} of avoiding-pin procedures and the length $step_{\text{avoid}}$ of a single avoiding-pin maneuver, are given. Based on them the maximum distance the TCP could ever have to the theoretical hole when entering the sliding-surface procedure can be computed by multiplying the two. This is necessary in order to normalize the actual distance d_{TCP2hole} of the TCP and the theoretical hole, yielding $\overline{d_{\text{TCP2hole}}}$.

$$\begin{aligned} d_{\text{TCP2hole-max}} &= max_{\text{avoid}} \cdot step_{\text{avoid}} \\ \overline{d_{\text{TCP2hole}}} &= \frac{d_{\text{TCP2hole}}}{d_{\text{TCP2hole-max}}} \end{aligned} \tag{13}$$

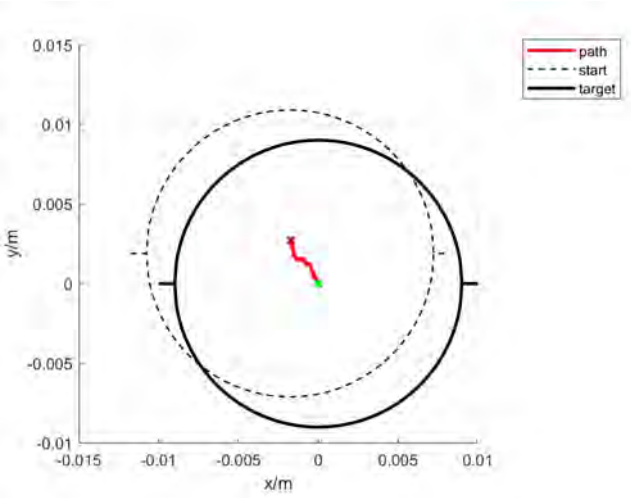


Fig. 6: Top-down view of the battery's trajectory overcoming a displacement error and being centered in the hole.

$\overline{d_{\text{TCP2hole}}} \to 1$, when a pin has been avoided several times. Then θ_{TCP2hole} is trusted more than $\theta_{\text{neg-tilt-dir}}$. In the same manner $\overline{d_{\text{TCP2hole}}} \to 0$, when the TCP has come closer to the theoretical hole and thus $\theta_{\text{neg-tilt-dir}}$ should be trusted more. Therefore $\overline{d_{\text{TCP2hole}}}$ can be used as the trust value κ, involved in equation 12:

$$\kappa = \overline{d_{\text{TCP2hole}}} \tag{14}$$

Figure 6 depicts real measurements of the peg's trajectory while sliding on the surface of the cell holder and then being centered and lowered within the hole. As for the no-pin-avoided case the pin-avoided case is terminated when the condition in equation 9 is true.

It may happen that while sliding on the cell holder surface towards the actual hole the tilted battery collides with a protruding pin. This may occur in both cases, the no-pin-avoided case and the pin-avoided case. It is indicated by the horizontal force rising above a certain threshold, e.g. $10.0\,\text{N}$:

$$\begin{aligned} force_{\text{xy}} &= \sqrt{force_{\text{x}}^2 + force_{\text{y}}^2} \\ threshold_{\text{force}_{\text{xy}}} &= 10.0\,\text{N} \\ collision_{\text{pin-horiz}} &= force_{\text{xy}} > threshold_{\text{force}_{\text{xy}}} \end{aligned}$$

If the condition $collision_{\text{pin-horiz}}$ is true, the robot should not continue pressing the gripped battery into the pin as this could damage the battery. Instead it should move horizontally around the pin while still trying to find the actual hole. Thus, the direction of the horizontal force vector of the pin acting on the TCP is evaluated:

$$\theta_{\text{force}_{\text{xy}}} = \text{atan2}(force_{\text{y}}, force_{\text{x}}) \tag{15}$$

The previous determined angle $\theta_{\text{slide-surface}}$ is then merged with $\theta_{\text{force}_{\text{xy}}}$ by trusting both directions equally. This new direction allows the robot to do both: circumnavigating the pin and approaching the actual hole.

$$\kappa = 0.5 \tag{16}$$

$$\theta_{\text{slide-around-pin}} = \kappa \cdot \theta_{\text{force}} + (1 - \kappa) \cdot \theta_{\text{slide-surface}} \tag{17}$$

Eventually the actual hole is found and the sliding-on-surface procedure is finished and the centering-lowering procedure is executed, as described in the next chapter. If nevertheless the hole cannot be found, a retry of the whole mounting process is launched.

VII. Centering-lowering procedure

While the avoiding-pin and sliding-on-surface procedures are helping the robot to overcome larger displacement errors and guiding the peg to the constraint region of the hole, the centering-lowering procedure is about inserting the tilted peg into the hole until a three points contact is established. If done so, the peg will be oriented back into the vertical in order to align its axis with the hole axis.

The centering-lowering procedure is entered, when the surface of the peg collides with the edge of the hole (see figure 2a) and a one point contact or two points contact is reached. Practically this is indicated by the collision height falling below the reference height:

$$h_{\text{coll}} < h_{\text{ref}} - h_{\text{tol}} \tag{18}$$

When the precomputed hole position matches the actual hole position well ($d_{\text{hole2hole}} \approx 0\,\text{mm}$, or a few tenth of a millimeter), the centering-lowering procedure is entered right away without the need of an avoiding-pin or sliding-surface procedure. The three points contact can be achieved by using the direction of the contact force vector as described in [Li+17], deploying RL algorithms as in [Ino+17; Bel+20] or by implementing other strategies presented in numerous peg-in-hole publications.

VIII. Results

During a several months period of testing in a real industrial production, the UR5e robot assembled more than 13,500 batteries with a success rate of greater than $99.54\,\%$ by using the methods presented in this paper. It took approximately $11\,\text{s}$ to pick up and fully assembly a single battery. The time needed for the assembly process itself depends on the positional deviation between the precomputed and actual hole $d_{\text{hole2hole}}$: the greater it is, the longer it takes to find the hole and assemble the battery. It may vary from $4\,\text{s}$ to up to $20\,\text{s}$ for a single try. Experiments have shown, that positional deviations up to $4\,\text{mm}$ millimeters can be overcome. However when is $d_{\text{hole2hole}}$ increasing, the error rate rises as well.

The following listing highlights the key parameters of the presented method:

- Success rate: $13557/13619 \approx 99.54\,\%$
- Cycle time of cell assembly: $\approx 11\,\text{s}$
- Of which the peg-in-hole takes: $\approx 4\,\text{s}$ to $20\,\text{s}$
- Maximum displacement error to be overcome: $\approx 4\,\text{mm}$

IX. Industrial validation and outlook

In collaboration with the German battery pack manufacturer ANSMANN AG a fully functioning robot plant (see figure 7) has been developed and tested for several months in real production, assembling thousands of battery cells. A UR5e robot arm is stationed at its center picking up a battery supplied by the trays on the left and right side. They contain batteries either with their positive or negative pole upwards. The robot chooses the polarity of a battery depending on the current position in the assembly sequence of the battery pack.

While having a battery gripped, the robot moves slightly above the hole position it is been given and tilts the battery. Then the peg-in-hole procedure described in this paper begins. After finding the hole at the end of the centering-lowering procedure, the robot tilts the battery back into the vertical axis in order to insert it completely into the hole by a specified force. If a cell cannot be mounted on the first run, there will be some more retries to find the hole. If the mounting still fails, maybe due to positional deviations too large for the robot to overcome, the gripped battery is sorted out and the robot continues with the next battery in the assembly sequence.

The current plant design has two major downsides: The first one is that the trays containing the batteries still must be filled manually. Secondly, the robot wastes valuable time picking up the batteries and bringing them to the cell holders. Therefore more research is invested into smart grippers: A new gripper is under development that sucks out the batteries directly out of the very box they are delivered and stored in. This would significantly reduce the time for equipping the robot plant with the batteries it needs for the battery pack assembly. Another approach is also investigated, that aims to feed the robot directly with batteries so it just can concentrate on inserting them into the cell holders. This approach would not only reduce but eliminate the time wasted for the movements from picking to placing. The robot would only need to move from one hole to another and insert the batteries by the hereby described method.

X. Conclusion

At the beginning it was shown that the lihtium-ion battery-based peg-in-hole problem dealt with in this paper is a lot more challenging than the general round peg-in-hole problems: A battery cannot be mounted from certain directions due to already mounted battery cells (occlusion by cells) and pins protruding from the cell holders surface (occlusion by pins). The cell holders containing the holes are not static but deform when they are assembled. Additionally lithium-ion batteries have to be handled with extreme care, proposing a compliant assembly strategy.

For each case, avoiding protruding pins during the hole search and approaching the hole while the battery is sliding on the cell holder surface, intelligent and yet simple strategies were presented: the avoiding-pin and sliding-on-surface procedure, respectively. Their only goal is to guide the peg into the constraint region where the actual peg-in-hole problem is solved.

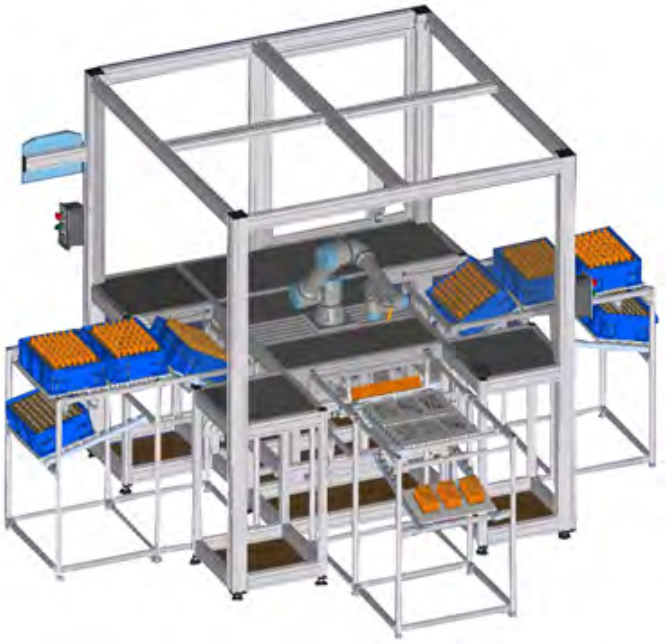


Fig. 7: The complete robot plant

Within the constraint region, the peg is simultaneously centered and lowered by using common peg-in-hole strategies published in other papers.

The explained methods have been implemented on a UR5e collaborative robot with inbuilt force torque sensors being the heart of a complete industrial robot assembly plant. No vision system was used for the peg-in-hole assembly. The robot plant is able to mount a variety of different shaped cell holders being given a very limited number of environmental constraints. Over a period of several months more than 13,500 batteries have been assembled with a success rate of greater than $99.54\,\%$.

Lastly an outlook over ongoing research was given, overcoming the downsides of the current plant design: reducing the time the robot plant needs to stop for being equipped with new batteries and eliminating the time it takes to pick up and move a battery from the tray to the cell holder.

References

[CB01] S.R. Chhatpar and M.S. Branicky. “Search strategies for peg-in-hole assemblies with position uncertainty”. In: *Proceedings 2001 IEEE/RSJ International Conference on Intelligent Robots and Systems. Expanding the Societal Role of Robotics in the the Next Millennium (Cat. No.01CH37180)*. Vol. 3. 2001, 1465–1470 vol.3. DOI: 10.1109/IROS.2001.977187.

[JP13] Ibrahim F. Jasim and Peter W. Plapper. “T-S fuzzy contact state recognition for compliant motion robotic tasks using gravitational search-based clustering algorithm”. In: *2013 IEEE International*

Conference on Fuzzy Systems (FUZZ-IEEE). 2013, pp. 1–8. DOI: 10.1109/FUZZ-IEEE.2013.6622415.

[Ino+17] Tadanobu Inoue et al. "Deep reinforcement learning for high precision assembly tasks". In: *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2017, pp. 819–825. DOI: 10.1109/IROS.2017.8202244.

[JPV17] Ibrahim F Jasim, Peter W Plapper, and Holger Voos. "Contact-state modelling in force-controlled robotic peg-in-hole assembly processes of flexible objects using optimised Gaussian mixtures". In: *Proceedings of the Institution of Mechanical Engineers, Part B: Journal of Engineering Manufacture* 231.8 (2017), pp. 1448–1463. DOI: 10.1177/0954405415598945.

[Li+17] Xiaoqing Li et al. "Human-inspired compliant strategy for peg-in-hole assembly using environmental constraint and coarse force information". In: *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. 2017, pp. 4743–4748. DOI: 10.1109/IROS.2017.8206348.

[Bel+20] Cristian C. Beltran-Hernandez et al. "Variable Compliance Control for Robotic Peg-in-Hole Assembly: A Deep-Reinforcement-Learning Approach". In: *Applied Sciences* 10.19 (2020). ISSN: 2076-3417. DOI: 10.3390/app10196923. URL: https://www.mdpi.com/2076-3417/10/19/6923.

[Rob] Universal Robots. *UR5e technical details*. URL: https://www.universal-robots.com/media/1802778/ur5e-32528_ur_technical_details_.pdf. (accessed: 19.07.2021).

Human-Robot Collaboration: a Survey on the State of the Art focusing on Risk Assessment

1st Luisa Hornung
Hochschule Karlsruhe - University of Applied Sciences
Research Group Robotics and Autonomous Systems (IRAS) at Institute of Applied Sciences (IAF)
Karlsruhe, Germany
luisa.hornung@h-ka.de

2nd Christian Wurll
Hochschule Karlsruhe - University of Applied Sciences
Research Group Robotics and Autonomous Systems (IRAS) at Institute of Applied Sciences (IAF)
Karlsruhe, Germany
christian.wurll@h-ka.de

***Abstract*—Counteracting the labour shortage, improving ergonomics in combination with increasing productivity, easy integration and programming are some of the benefits that Human-Robot Collaboration (HRC) offers. The innovative technology combines the strengths of humans and robots and helps to enable the Industry 4.0 approach. However, despite the many advantages, the widespread use of the production technology has still not been achieved. Reasons for the inhibited use include large knowledge and know-how gaps in the area of implementation and planning, the low economic efficiency due to low cycle times and the high safety requirements. In this paper, a study is presented that addresses the state of the art of HRC and in particular the implementation of risk assessment. The study includes two surveys with a total of 57 participants, the results of which are presented and discussed in this work.**

Keywords—Human-Robot Collaboration, Survey, State of the Art, Risk Assessment, Safety

I. Introduction

Due to the increasing demand for customized products that are available within a short period of time, combined with decreasing product life cycles, the requirements for fast response times in production are growing. This requires flexible and adaptable production technologies capable of handling the expanding range of variants and making it possible to introduce new products rapidly [1]. Furthermore, in order to maintain competitiveness in the long term, both new and innovative production systems and digital production strategies must be used [2], capable of linking people and technologies with advanced information and communication technologies [3]. According to [4] Human-Robot Collaboration (HRC) is one of such important cyber-physical technology that supports and enables the Industry 4.0 approach.

Through HRC it is possible to use the capabilities of humans and combine them with the strengths of robots [5]. The interaction of the robot's repeatability, accuracy and strength with the human's cognitive abilities makes the production technology very flexible and adaptable, solving the needs and requirements of the industry [6], [7]. The advantages of HRC are manifold: increased productivity, improved product quality and ergonomics, and the elimination of protective fences enables optimized use of factory space [6], [7]. HRC also has a positive effect on the employee structure in the company. According to [8], the use of HRC helps to counteract the shortage of skilled workers on the one hand, as the production systems can close the skill gaps. On the other hand, the use of new innovative technologies makes the company more attractive and thus attracts the younger generations and skilled workers.

But despite the numerous advantages, the sales volume of collaborative robots is barely 5 per cent compared to the total volume of robots sold in 2019 [9]. According to [10], the full potential of HRC has not yet been exploited as many HRC applications are limited to the same operations that a fully automated robotic application would perform. This is one of the reasons for the low sales volume of collaborative robots. References [10], [11] and [6] believe that the barriers to the use of collaborative robotics lie in the high safety requirements and resulting high costs for risk mitigation measures. These statements are supported by a study that identified conducting risk assessments for specific HRC applications as one of the main challenges in HRC implementation [12]. Other challenges that emerged from the study were the identification of appropriate HRC applications and the application of safety standards.

In order to obtain a comprehensive overview of the state of the art in the field of HRC, a two-part online study was conducted, targeting experts and end users. The study consisted of two independent surveys, one providing a general overview of HRC and the second specifically addressing one of the main challenges: risk assessment for HRC applications and its supporting tools.

The paper is structured as follows: first, Section II describes the research methodology, which includes the research approach, the structure of the questionnaire and the composition of the survey participants. Sections III and IV present and discuss specific parts of the results of the *General Overview Survey* and the *Risk Assessment Survey*. The paper ends with a conclusion in Section V.

II. Research Methodology

A. Research Approach

The study aimed to provide a comprehensive overview of the state of the art in HRC, reflecting the current situation of HRC in the industry. The focus was on the opinions and experiences of users, integrators and service providers who apply HRC in industry, and in particular the hardware installed, suitability of processes, benefits and hurdles of HRC.

By conducting two independent surveys, the objective was to reach different groups of experts and thereby obtain an unbiased general overview of the topic area in the first step and specifically address one of the main hurdles for the introduction of HRC in depth in the second step. In this way, a broad scope of data and information was collected. The survey on the topic of risk assessment dealt with the requirements of experts for software tools and in general, their advantages and disadvantages and in specific with desirable enhancements for HRC.

The standardized survey was chosen as the data collection method and the type of survey adopted was an *online survey*. The standardized survey ensures that different answers to a question are in fact due to different characteristics, attitudes and behavior of the persons and not to varying conditions during the survey situation. The prerequisites are (1) a uniform formulation of the questions, the answer options and the order of the questions, (2) standardized framework conditions and (3) standardized implementation of the survey [13].

In order to meet requirement (1), the general outline of the questionnaires were built according to the structure defined by [14] and the question phrasing followed the *10 rules of question phrasing* according to [13].

To comply with (2) and (3), the survey type online survey was selected. This has several advantages [15]: (a) the answers are more thoughtful as there is no time pressure, (b) the answers are more "honest" as the survey is anonymous, (c) completing the survey is possible in a short time and (d) there is the possibility to give hints (e.g. in case of incomplete questions). In addition, a broad spectrum of experts can be reached through an online survey.

The surveys were designed using the web portals UmfrageOnline[a] and EUSurvey[b], published via LinkedIn and email and conducted between January and March 2021.

B. Questionnaire

The *General Overview Survey* consisted of four thematic blocks with a total of 32 questions:

- Collaborative robotics basics,
- Process suitability,
- Training and
- Safety requirements and standards

Whereas the *Risk Assessment Survey* consisted of three thematic blocks with a total of 19 questions:

- Tools for conducting risk assessments,
- Risk assessment method and
- Requests for improvement with regard to HRC

C. Structure of the Survey Participants

A total of 29 participants participated in the *General Overview Survey* and a total of 28 in the *Risk Assessment Survey*. The characteristics of the participants are presented in TABLE I. respectively in TABLE II. The left column contains the parameter of the characteristic and the right column lists the number of participants exhibiting this characteristic. In TABLE II. the level of experience is defined as follows:

- *Expert knowledge*: participant mainly engages in the subject area of risk assessment and has comprehensive know-how.
- *Advanced knowledge*: participant has already carried out several risk assessments.
- *Basic knowledge*: participant has already carried out one risk assessment.

All participants with experience in the field of HRC also had experience in the field of industrial robotics.

TABLE I. GENERAL OVERVIEW SURVEY: CHARACTERISTICS OF THE PARTICIPANTS

Occupation	***No.***	***Industry Sector***	***No.***
Integrator	12	Mechanical Engineering	18
Manufacturer	3	Automobile	3
Entrepreneur	3	Electro industry	3
Consulting	2	Different sector	5
Service Provider HRC	2		
Different Occupation	7		
Experience in HRC	***No.***	***Number of HRC in operation***	***No.***
< 1 year	1	none	14
1 - 5 years	22	1	2
> 5 years	6	> 1	13
Experience in planning HRC	***No.***	***Number of implemented HRC***	***No.***
yes	24	1	2
no	5	1 - 5	13
		> 5	8

TABLE II. RISK ASSESSMENT SURVEY: CHARACTERISTICS OF THE PARTICIPANTS

Occupation	***No.***	***Experience in Risk Assessment***	***No.***
Consulting	17	< 5 years	18
CE Representative	6	5 – 10 years	3
Integrator	3	11 – 15 years	3
End user	2	> 15 years	5
Researcher	1		
Level of Experience	***No.***	***Experience in Robotics***	***No.***
Expert knowledge	19	Industrial robotics	24
Advanced knowledge	8	HRC	16

[a] https://www.umfrageonline.com/

[b] https://ec.europa.eu/eusurvey/

Occupation	*No.*	*Experience in Risk Assessment*	*No.*
Basic knowledge	1	none	1

III. General Overview Survey Results

The questions in the thematic blocks were not formulated as compulsory questions, resulting in the consequence of not every participant having answered every question. A *response rate (RR)* is thus specified for the interpretation of the results:

$$RR = \frac{number\ of\ participants\ answering\ the\ question}{total\ number\ of\ participants} \quad (1)$$

The questionnaire consisted of the question types *multiple choice* (MC), *single choice* (SC), *free question* (FQ), *ranking question* (RQ) and *applies/does not apply* (A/DA). In the following subsections, the results are supplemented by a tuple consisting of RR and the question type. The results are given as absolute values on the number of participants.

In the following, a portion of the results of the *General Overview Survey* results is presented and discussed. The report on the entire survey can be found in [16].

A. Collaborative Robotics Basics

The thematic block of questions focused on the areas of collaboration types/protection principles, the types of robots used and the gripping techniques employed.

The protection principles applied by the experts in the HRCs implemented and planned so far are shown in Fig. 1 in descending priority (RR=93%;MC). The experts primarily use the protection principle *power and power limitation (PFL)*, followed by *safety-rated monitored stop* and *speed and separation monitoring (SSM)*. The protection principle *manual guidance* is used the least.

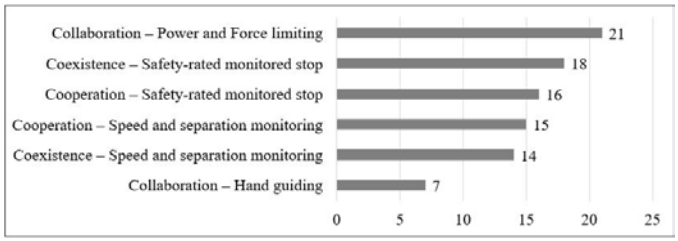


Fig. 1. Mainly applied protection principles in HRC applications.

The ranking of the collaboration types according to which the HRC applications are most commonly operated is assessed by the experts as follows (RR=93%;RQ):

1. Coexistence – Safety-rated monitored stop
2. Coexistence – SSM
3. Cooperation – Safety-rated monitored stop
4. Cooperation – SSM
5. Collaboration – PFL
6. Collaboration – Hand guiding

According to the experts' opinions, HRC applications are predominantly operated in the collaboration type *coexistence*, using either the protection principle *safety-rated monitored stop* or *SSM*. Although the protection principle *PFL* is most frequently used by the experts to safeguard HRC applications, HRC applications are barely operated in the collaboration type *collaboration*, even though this protection principle enables direct and close contact with humans [17].

In most applications, not only one protection principle is implemented, but a combination of several. HRC applications are often implemented as *coexistence* and protected by *SSM*. If a human approaches at a certain distance, the robot switches to another protection principle (e.g. *PFL* or *safety-rated monitored stop*). The flexible transition between protection principles enables efficient operation of the HRC application, as the robot can operate at full speed and handle dangerous components when the human is outside the protection zone.

In addition to the collaboration types/protection principles, the questionnaire covered the types of robots (RR=100%;MC) and gripper technologies (RR=97%;MC) employed in HRC applications. The results are shown in Fig. 2 a) and b). In addition to the robot types listed, Yaskawa, Mitsubishi, Stäubli, Doosan, Kawasaki, Rethink - Sawyer were each mentioned once. Similarly, servo grippers and passive gripping systems were each mentioned once corresponding the gripping technologies in use.

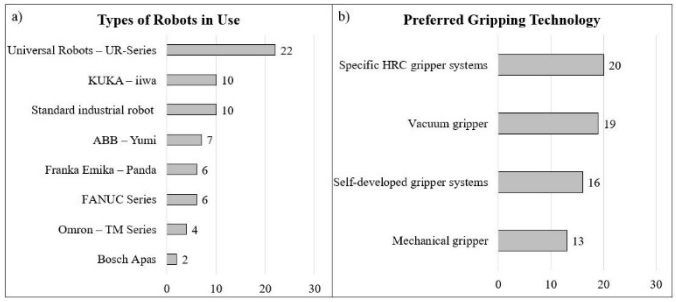


Fig. 2. a) Types of robots and b) Gripping technologies used in HRC.

Besides the market leader Universal Robot and KUKA iiwa, conventional industrial robots are also increasingly utilized in HRC applications. With today's state of the art, it is possible to equip conventional industrial robots with safety technology enabling them to be used in collaborative workplaces. Through safe interaction between humans and conventional industrial robots, the advantages of both, HRC and full automation, become available. In shared workspaces, the robot acts at low speed and ensures the safety of the human. In areas that are not accessible to humans, the robot can operate at full speed and power, counteracting the hurdle of the lack of economic efficiency.

Furthermore, the results show that special HRC gripping systems or vacuum gripping technology are predominantly employed. Both technologies have a low risk of injury compared to conventional gripping systems due to the design, energy supply and special sensitive functions. The gripper as a component is safe and HRC suitable and does not need to be safeguarded by additional risk reduction measures. Preferred gripper manufacturers are Zimmer Group GmbH (9 mentions),

SCHUNK GmbH & Co. KG (6 mentions), OnRobot A/S (6 mentions), RobotiQ (5 mentions), WEISS ROBOTICS GMBH & CO. KG (1 mention), Schmalz (1 mention) and own 3D printing (1 mention).

B. Process Suitability

The thematic block concerned suitable processed for HRC, the process characteristics a process should have in order to be suitable for HRC, the reasons for implementing HRC and the current hurdles.

Regarding suitable processes for HRC, the experts evaluate the process types *Handling/Pick and Place* (26 mentions), *Assembly* (23 mentions), *Quality Assurance* (21 mentions), *Palletizing* (19 mentions), *Screwing applications* (17 mentions), *Packing* (16 mentions), *Positioning aid* (16 mentions), *Lifting aid* (12 mentions), *Welding applications* (6 mentions), *Gluing processes* (3 mentions), *Grinding/Polishing* (1 mention) as more or less suitable for the implementation of HRC (RR=100%;MC). For the implementation of *Handling/Pick and Place*, *Assembly, Quality Assurance* and *Palletizing* tasks in HRC, a large number of sample applications, best practices and comprehensive experience exist as well as sensor systems and other HRC-capable hardware, enabling a safe implementation and efficient employment. The implementation of those processes using HRC offers the advantages of reducing monotonous activities, a low-effort adaptation to different products and an optimized use of factory space.

Processes involving hazardous tools or materials, such as welding, gluing or polishing tasks, are not suitable for HRC applications, or only to a limited extent. Due to the process-related hazards, safe application planning and ensuring the safety of humans is linked to a very high expenditure of time and costs, resulting in the possibility of high residual risks that cannot be mitigated. The experts disagree about welding applications in particular. There are already solutions on the market that enable welding in HRC, but the results show a critical attitude on behalf of the experts. In this context, the high approval of screw applications is striking. In screwing applications, a screwing tool is required which, due to its small surface area and rotating movements, poses a great risk to humans. The availability of HRC screwdrivers, which enable safe operation of screwing applications in HRC, is a possible reason for the high level of approval, despite the high process-related risks.

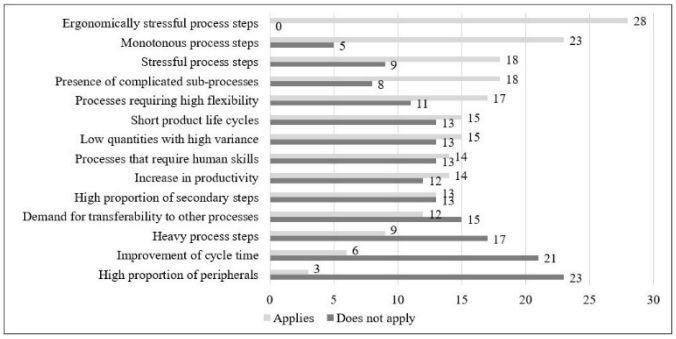


Fig. 3. Characteristics for HRC suitability.

The experts agree that processes that are (ergonomically) stressful, include monotonous process steps or complicated sub-process steps, offer great potential for optimization through the use of HRC (see Fig. 3). On the other hand, processes that require a high number of peripherals or processes for which the primary goal is to improve the cycle time or that involve heavy work steps are rather unsuitable for the use of HRC (RR=97%;A/DA). For some characteristics, experts disagree on whether the use of HRC is beneficial or not. These aspects depend on the expert's perspective, e.g. an existing characteristic may be beneficial through the use of HRC compared to manual operations, but disadvantageous for HRC compared to full automation.

Detailed information on areas where HRC provides an advantage or disadvantage compared to manual activity or full automation is provided in TABLE III. (RR=90%;MC). In TABLE III. (1) is *advantage of HRC compared to manual process*, (2) *disadvantage of HRC compared to manual process*, (3) a*dvantage HRC compared to full automation* and (4) is *disadvantage HRC compared to full automation*. The values refer to the number of experts who support the corresponding position.

TABLE III. HRC COMPARED TO MANUAL ACTIVITY AND FULL AUTOMATION

Characteristics	***(1)***	***(2)***	***(3)***	***(4)***
Counteracting the shortage of manpower	21	2	10	6
Repeatability	21	1	5	14
Low error rate/High quality	20	2	1	18
Precision	18	6	3	14
Increase in productivity	17	4	4	19
Economic efficiency	14	8	9	9
Compliance with low tolerances	14	6	2	15
Sensitivity	12	7	18	3
Improvement of cycle time	11	11	2	23
Easy to learn/operate/use	11	9	18	3
Collision detection	11	6	18	2
Fast amortization	9	10	12	7
Fast reprogramming	8	10	19	1
Elimination of time consuming training	8	10	17	3
Low investment	8	15	14	5
Response to uncertainties	8	13	12	5
High variance	7	13	19	2
Fast adaptability to changing conditions	7	1	20	13
Speed	7	13	1	21
Fast installation	6	15	17	4
Small quantities/single-piece production	4	16	17	1
Low space requirements	2	14	16	6

Advantages that HRC offers compared to manual work are considered by the experts to be disadvantages compared to full automation for the majority of the aspects and vice versa. But HRC offers advantages over manual activities as well as full automation in terms of counteracting labour shortages, sensitivity, ease of learning and collision detection, and disadvantages in terms of process speed. The experts' assessment reflects and confirms current knowledge and research results. In a specific application, the objectives must therefore be clearly defined and the optimal implementation strategy must be selected for the application in order to utilize the advantages of HRC and achieve the best possible design of the process.

Other findings of the survey are the results regarding the main reasons for implementing HRC applications (RR=100%;SC) as well as the reasons for the hesitant adoption of HRC applications (RR=100%;MC). *Improving ergonomics* is the most frequently cited reason for implementation with 11 mentions, followed by *saving labour/counteracting the labour shortage* and *(partial) automation with low investment* with 9 mentions each. The reasons given are in line with the findings on the advantages of HRC over manual activities and full automation from the previous analysis. The reasons for the hesitant adoption of HRC applications are listed in Fig. 4. The experts agreed that *the lack of HRC know-how, experience* and *skills*, the *economic inefficiency* and the *high safety requirements* to be fulfilled inhibit the use of HRC. These results indicate that there is a need for action, particularly in the area of know-how development and training, in order to accelerate the introduction of HRC into the industry. Economic production is directly related to the speed of the robot, which must be kept low due to the high safety requirements and the risk of collision. A higher movement speed leads to shorter cycle times and thus to an increase in economic efficiency. The results show that there is potential for optimization and a need for research, especially in the area of safety.

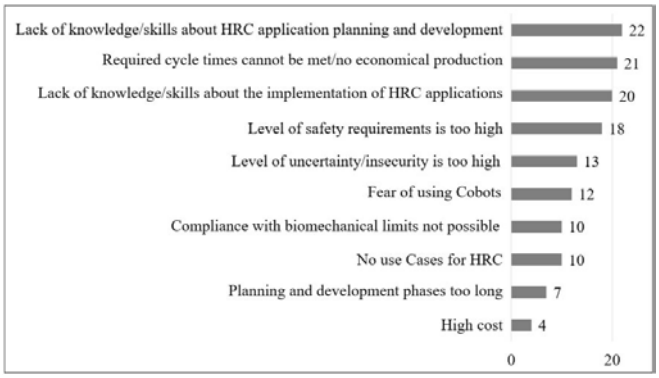


Fig. 4. Reasons for the hesitant adoption of HRC application.

IV. Risk Assessment Survey Results

According to the *General Overview Survey*, the questions in the *Risk Assessment Survey* were not mandatory and were developed using the question types SC, MC, A/DA. The RR is also used to interpret the results and the presentation of the results is also supplemented by a tuple consisting of RR and question type. The results are given as absolute values on the number of participants.

In the following, a portion of the results of the *Risk Assessment Survey* results is presented and discussed. The report on the entire survey can be found in [18].

A. Risk Assessment Tools and Methods

This section presents the results regarding the risk assessment tools used by the experts, the methodology and general aspects of the topic of risk assessment. *Tool* is defined as the totality of documentation tools for risk assessment and includes both MS Office templates and software.

For the documentation of the risk assessment (RR=96%;MC), the experts predominantly prefer MS Office templates (25 mentions), though the most frequently used are self-created Excel templates (11 mentions). In contrast, software solutions are utilized less (11 mentions). The most frequently used software is Safexpert with 7 mentions (RR=54%;MC). The tools in use are listed in Fig. 5 a). MS Office templates offer the advantage of being easily and flexibly adapted to individual needs. This allows the user to design the risk assessment documentation according to individual preferences and the type of machinery under consideration. Software provides a relatively rigid structure and therefore offers little scope for individual adaptations, but offers the advantage of standardization across all risk assessments conducted.

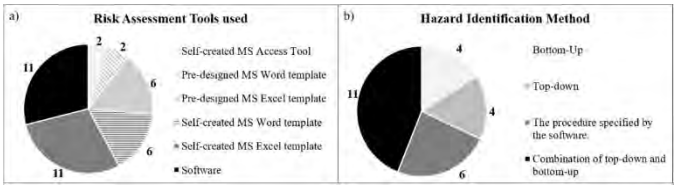


Fig. 5. a) Risk assessment tools used by the experts and b)

For the identification of hazards (RR=96%;SC), the majority of experts use the checklist procedure according to DIN EN ISO 12100 Annex B (17 mentions) and apply a combination of top-down and bottom-up methodology (11 mentions) (RR=100%;SC) (see Fig. 5 b)). An important question that needs to be addressed in the risk assessment is the completeness of the identified hazards. The use of the checklist method ensures each hazard listed in DIN EN ISO 12100 Annex B has been considered and, in conjunction with the combination of top-down and bottom-up, a high completeness of the identified hazards can be achieved.

The survey also included general statements on the conduction of risk assessments, assessed accordingly TABLE IV. by the experts (RR=93%;A/DA). The results highlight the importance of the experience and know-how of the person conducting the risk assessment in order to achieve a successful and high-quality risk assessment. A tool can only support the implementation and serves as documentation, but experience cannot (yet) be replaced by a software.

TABLE IV. Statements on Risk Assessment

Statement	***No.***
The quality of the risk assessment depends on the know-how, experience and accuracy of the user	25
The software available on the market is not a substitute for experience and expert knowledge.	23

Statement	*No.*
The risk assessment should be started already in the development phase of a product/application.	23
Checklists of potential hazard causes or consequences are helpful and should be used in the hazard identification phase of the risk assessment.	21
The most important aspect of risk assessment is the experience of the user.	21
A risk assessment should be carried out in a group.	18
A risk assessment should be conducted by an external service provider or system integrator.	3

A need for assistance in conducting the risk assessment (RR=86%;MC) exists in deciding if a *hazard is acceptable* (16 mentions), which *hazards are reasonably foreseeable and where the boundary to gross negligence lies* (14 mentions), which *hazards need to be considered* (12 mentions) and the *automatic transfer of hazards from application-specific standards into the risk assessment* (12 mentions). Risk assessment is partially a subjective process. In particular, risk estimation and evaluation are mainly based on the subjective opinion of the person conducting the risk assessment, and thus also the decision on the acceptable residual risk. The results of the survey indicate a need for support, especially in the subjective aspects of risk assessment, and that there are still uncertainties in these areas. An experienced user has less difficulties in assessing risks and deciding if the residual risk is acceptable. For an inexperienced user who wants to implement HRC applications in the production line and therefore needs to conduct a risk assessment, these processes represent a major hurdle. In order to facilitate the introduction of HRC and to simplify the process of risk assessment of HRC for inexperienced users, it is necessary to increase the objectivity of subjective processes.

B. Enhancements for Risk Assessment Tools for HRC

The requests for improvements and support to facilitate risk assessment for HRC are grouped into two categories. The first category describes possible HRC-specific additions to risk assessment tools such as guidelines, good practice examples or other assistance and support and the second category includes new functionalities by automating parts of the risk assessment process.

Regarding the first category, the results of the survey are listed in TABLE V. (RR=79%; A/DA). The majority of experts agree that there should be more guidance on risk assessment for HRC. In particular, matching specific hazards with appropriate risk reduction measures is considered promising with a total of 19 mentions, closely followed by support in deciding when a significant change exists with 18 mentions. In risk reduction according to the 3-stage procedure, the hazard is safeguarded in the first stage by design measures, in the second stage by technical measures and the third stage covers user information [19]. Due to the absence of the protective fence at HRC, the hazards must be secured in particular by stages 1 and 2. As the number of designed and implemented HRC applications increases, the experience and the range of possibilities for safeguarding certain hazards also rises. This experience should be shared among users, designers and safety engineers, and new users in particular can benefit from catalogues containing possible risk-reducing measures for specific hazards and the experience of experts. The availability of support and guidelines, assistance in deciding on significant changes and in performing the necessary collision measurements can counteract the uncertainties and challenges related to risk assessment and thus advance the adoption of HRC applications.

TABLE V. DESIRABLE ADDITIONS TO RISK ASSESSMENT TOOLS

Statement	*Applies*	*Does not apply*
For certain hazards, a catalogue of possible applicable safety measures.	3	19
Assistance in deciding whether changes to HRC applications are significant enough to require a renewed risk assessment.	4	18
More help and guidance on HRC-specific issues in general (e.g. more instructions, application examples, exemplary safety measures, instructions on how to conduct collision measurements, etc.).	5	17
More precise instructions on how to conduct collision force/pressure measurements.	6	16
Concretisation of ISO TS 15066.	6	16

The experts also consider the automation of sub-processes in risk assessment to offer great potential for support (see TABLE VI.) (RR=68%; A/DA). The visualization of paths and work areas of humans and robots as well as the development of standard modules for recurring sub-processes with 16 mentions each are seen as the most promising by the experts. Visualization enables the simple detection of overlapping paths or work areas, which facilitates the definition of hazardous areas and supports the fast detection of collision points. The use of standardized templates for certain (sub-)processes also leads to a considerable facilitation of the process, since first, all hazards, assessments, evaluations and mitigation of the specific process are pre-determined and second, the documentation effort can be significantly reduced. Yet, the development of standard templates proves to be difficult, as a multitude of parameters have to be considered influencing the hazards and their severity (e.g. workpiece, tool, gripping technique, working environment, etc.). In general, the automation of sub-areas of risk assessment is considered promising and facilitating by the experts. The experts are divided solely on the automation of hazard identification using simulations or artificial intelligence and the automated detection of critical geometries.

TABLE VI. DESIRABLE NEW FUNCTIONALITIES

Statement	*Applies*	*Does not apply*
Visualization of the path and working areas of the robot and the human.	16	3
Standard templates / modules for risk assessment of frequently recurring (sub-) processes (e.g. safety functions for specific activities).	16	3
Automatic detection of collisions and comparison with parameters for compliance with biomechanical limit values.	15	4
Simulation of human behaviour in certain situations / Inclusion of human behaviour in certain situations in the risk assessment (e.g. on which parts of the body a collision can occur).	14	5

Statement	*Applies*	*Does not apply*
Automatically calculate safety distances and speeds with the help of simulations / CAD data.	13	6
Automatic hazard identification using simulations or artificial intelligence.	12	7
Automatically detect critical geometries (sharp edges, sharp corners, etc.) with the help of simulations / CAD data.	13	7

Despite the significant number of opposing opinions, the experts were positive about the usefulness of a (partially) automatic identification of hazards with a total of 18 mentions (RR=93%;SC).

Automation of hazard identification based on simulations or artificial intelligence and automated detection of critical geometries are currently research-based and methodologies proven and tested in industry are not yet available. This fact explains the differentiated and contradictory opinions of the experts. The statement is supported by the question about the awareness of the methodologies developed in research (RR=75%;MC), the result of which is shown in Fig. 6. With the exception of the combination of HAZOP and UML, the experts are either unaware of most of the methods or consider them to be unpromising.

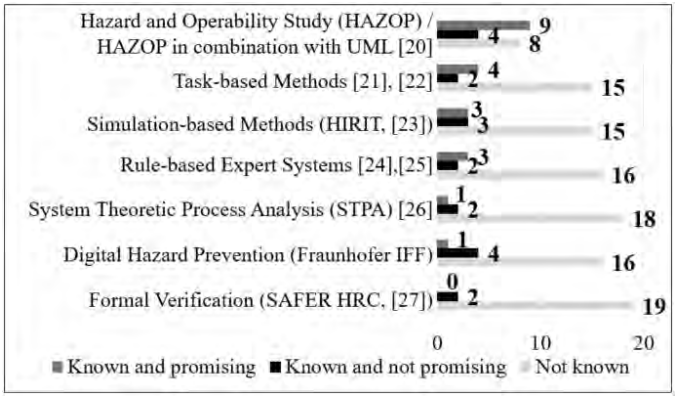


Fig. 6. Novel methods to facilitate hazard identification.

Another important aspect is the expert opinion regarding the calculation of the biomechanical limits on a simulation basis (RR=82%;A/AD). The results are shown in TABLE VII. The experts agree that collision measurements on a simulation basis significantly help risk assessment of HRC applications and if the quality of the simulation is high enough, the measurements could even be reduced. However, the majority of experts believe that simulations cannot completely replace measurements. The high effort required to develop a simulation meeting the requirements for quality and accuracy in order to be able to simulate collisions in detail is offset by the effort required to perform measurements in reality. In order to be able to obtain detailed and reliable statements through simulations, the simulations must be as close to reality as possible, including the modelling of the worker's behavior. One possibility to reduce the effort of model creation would be the use of virtual reality, eliminating the need for human modelling.

TABLE VII. SIMULATION-BASED DETERMINATION OF BIOMECHANICAL LIMITS

Statement	*Applies*	*Does not apply*
Simulation-based estimation of collision force/pressure could significantly help risk assessment of HRC applications.	20	2
With a detailed, high-quality simulation, it would be possible to significantly reduce the number of required collision force/pressure measurements.	19	3
With a detailed, high-quality simulation, it would be possible to significantly reduce the number of required collision force/pressure measurements.	9	13

V. CONCLUSION

As part of the study on the current state of the art of HRC, two independent surveys (*General Overview Survey* and *Risk Assessment Survey*) were conducted, the results of which were presented and discussed in this paper.

Coexistent implementations of HRC applications are the most common, whereas collaborative applications are only implemented to a small extent. In addition to the market leader Universal Robots and KUKA iiwa, conventional industrial robots equipped with appropriate safety technology are increasingly being used. With regard to gripping technology, the experts are focusing on the safe solutions and predominantly use HRC-capable gripping systems or vacuum gripping technology. For HRC applications, processes that have low process-related hazards are classified as HRC suitable. The biggest obstacle to the introduction of HRC applications is the large knowledge and know-how gap as well as the non-economical operation of HRC applications due to too high cycle times.

A key finding of the *Risk Assessment Survey* is the dependence of the quality of the risk assessment on the experience and know-how of the person conducting the risk assessment. For the documentation of the risk assessment, experts prefer MS Office templates instead of software solutions. If software solutions are applied, Safexpert is chosen most frequently.

A need for support in risk assessment for HRC in the form of general assistance, examples and guidelines as well as standard catalogues containing the suitable risk-reducing measures for specific hazards or in deciding when a significant change occurs, is considered useful and necessary. The survey also identified a need for (partial) automation of risk assessment, especially in the area of hazard identification. In research, there are already methodologies and procedures that simplify the identification of hazards, but these are not known or are not considered promising by the experts. Further support is needed in the area of measuring biomechanical limits. The study concluded that the measurements can be reduced with a detailed, high-quality and realistic simulation.

In conclusion, HRC should be employed in areas where the benefits of HRC can be optimally exploited. In terms of safety and risk assessment, there are a number of shortcomings that need to be addressed through research and development in order to further advance the technology.

ACKNOWLEDGMENT

This work was funded by the research project "ProBot", which is funded under the programme "Zukunft der Arbeit" by the German Federal Ministry of Education and Research (BMBF) and the European Social Fund (ESF) (funding label 02L17C550) and implemented by the Project Management Agency Karlsruhe (PTKA).

REFERENCES

[1] L. Barbazza, M. Faccio, F. Oscari, and G. Rosati. “Agility in assembly systems: a comparison model,” in Assembly Automation, 2017.

[2] D. T. Matt, and E. Rauch. “ SME 4.0: the role of small-and medium-sized enterprises in the digital transformation“ in Industry 4.0 for SMEs, Palgrave Macmillan, Cham, 2020, pp. 3-36.

[3] AMM S. Ullah. “W hat is knowledge in Industry 4.0?” in Engineering Reports, Wiley Online Library, vol. 2, No. 8, pp. e12217, 2020.

[4] L. Gualtieri, and E. Rauch, and R. Vidoni. “Emerging research fields in safety and ergonomics in industrial collaborative robotics: A systematic literature review,” in Robotics and Computer-Integrated Manufacturing, Elsevier, vol. 67, pp. 101998, 2021.

[5] B. Siciliano, and O. Khatib. “Springer handbook of robotics”, springer, 2016.

[6] L. Wang, S. Liu, H. Liu, and X. V. Wang. “Overview of human-robot collaboration in manufacturing” in 5th International Conference on the Industry 4.0 Model for Advanced Manufacturing, Springer, Belgrade, Serbia, pp. 15-58, June 2020.

[7] E. Matheson, R. Minto, E. G. G. Zampieri, M. Faccio, and G. Rosati. “H uman–Robot Collaboration in Manufacturing Applications: A Review,” in Robotics, vol. 8, No. 4, pp. 100, doi: 10.3390/robotics8040100, 2019.

[8] S. Kumar, C. Savur, and F. Sahin. “Survey of Human–Robot Collaboration in Industrial Settings: Awareness, Intelligence, and Compliance,” in IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 51, No. 1, pp. 280-297, doi: 10.1109/TSMC.2020.3041231, 2021.

[9] IFR International Federation of Robotics. “IFR Press Conference 2020”, url: https://ifr.org/downloads/press2018/Presentation_WR_2020.pdf, accessed: 16.07.2021.

[10] J. Saenz, N. Elkmann, O. Gibaru, and P. Neto. “Survey of methods for design of collaborative robotics applications-why safety is a barrier to more widespread robotics uptake,” in Proceedings of the 2018 4th International Conference on Mechatronics and Robotics Engineering, pp. 95-101, 2018.

[11] T. P. Huck, N. Münch, L. Hornung, C. Ledermann, and C. Wurll. “Risk assessment tools for industrial human-robot collaboration: Novel approaches and practical needs,” in Safety Science, Elsevier, vol. 141, pp. 105288, 2021.

[12] C. Gaede, F. Ranz, V. Hummel and W. Echelmeyer. “A study on challenges in the implementation of human-robot collaboration,” in Journal of Engineering, Management and Operations Vol. I: Die Zeitschrift pr{\"a}sentiert Ergebnisse aus der angewandten Forschung und Grundlagenforschung mit Schwerpunkten in den Bereichen Operations, Logistik, Produktentstehung, Industrie 4.0., LIT Verlag Münster, pp. 29, 2020.

[13] M. Tausendpfund. “Quantitative Methoden in der Politikwissenschaft,“ Springer VS, Wiesbaden, doi: https://doi.org/10.1007/978-3-658-20698-7, 2018.

[14] R. Jacob, A. Heinz, and J.P. Décieux. “Umfrage: Einführung in die Methoden der Umfrageforschung,“ Oldenbourg Wissenschaftsverlag, isbn:9783486714982, 2013.

[15] R. Porst. “Praxis der Umfrageforschung,“ Vieweg+Teubner Verlag, doi: https://doi.org/10.1007/978-3-322-84789-8, 1985.

[16] L. Hornung. “Ergebnisse der Umfrage Mensch-Roboter-Kollaboration,“ doi:10.13140/RG.2.2.29402.26560, 2021.

[17] DGUV, Deutsche Gesetzliche Unfallversicherung. “ Kollaborierende Robotersysteme,“ in Planung von Anlagen mit der Funktion „Leistungs- und Kraftbegrenzung“, DGUV FB HM-080, 2015.

[18] L. Hornung. “Ergebnisse der Umfrage Risikobeurteilung für Mensch-Roboter-Kollaboration (MRK),“ doi: 10.13140/RG.2.2.27292.00644, 2021.

[19] DIN EN ISO 12100. Safety of machinery – General principles for design – Risk assessment and risk reduction (ISO 12100:2010)

[20] J. Guiochet. “Hazard analysis of human--robot interactions with HAZOP-UML,” in Safety science, vol. 84, Elsevier, pp. 225-237, 2016.

[21] V. Gopinath, and K. Johansen. “Risk assessment process for collaborative assembly--a job safety analysis approach,” in Procedia CIRP, vol. 44, Elsevier, pp. 199-203, 2016.

[22] J. A. Marvel, J. Falco, and I. Marstio. “Characterizing task-based human--robot collaboration safety in manufacturing,” in IEEE Transactions on Systems, Man, and Cybernetics: Systems, vol. 45, No. 2, IEEE, pp. 260-275, 2014.

[23] P. Bobka, T. Germann, J. K. Heyn, R. Gerbers, F. Dietrich, and K. Dröder. “Simulation platform to investigate safe operation of human-robot collaboration systems,” in Procedia CIRP, vol. 44, Elsevier, pp. 187-192, 2016.

[24] R. Awad, M. Fechter, and J. van Heerden. “Integrated risk assessment and safety consideration during design of HRC workplaces,” in 22nd IEEE International Conference on Emerging Technologies and Factory Automation (ETFA), IEEE, pp. 1-10, 2017.

[25] D. L. Wigand, S. Krüger, S. Wrede, T. Stuke, and M. Edler. “Report on the Application of the COVR Toolkit and its Protocols for the Certification of Cooperative Robot Workstations,” in University of Bielefeld, 2020.

[26] N. Leveson. “E ngineering a safer world: Systems thinking applied to safety,” in MIT Press, 2011

[27] M. Askarpour, D. Mandrioli, M. Rossi, and F. Vicentini. “SAFER-HRC: Safety analysis through formal verification in human-robot collaboration,” in International Conference on Computer Safety, Reliability, and Security, Springer, pp. 283-295, 2016.

Collaborative or Industrial Robot? An Economic Calculation Scheme to Determine the Optimal Robot Technology in Fenceless Machine Tending

Christopher Schneider, Martina Hutter-Mironovová, Michael Klos, Mohamad Bdiwi, Matthias Putz

Abstract— **Current Human-Robot Interaction (HRI) evaluation methods deal primarily with multi-criteria target systems to identify the optimal possible robot cell. Those possibilities are often limited exclusively to either collaborative robots (cobots) or industrial ones. A methodological gap exists in the comparison of both robot technologies regarding their economic advantages. Therefore, we present a planning tool to determine the optimal fenceless robot-based machine tending system, strongly incorporating financial factors and the influence of human presence on the robot's operating speeds.**

I. Introduction

The market introduction of the first cobots ignited a huge Industry 4.0 hype around this new technology, mainly amplified by opening up robotization to small and medium-sized companies (SMEs) by providing higher usability and flexibility [1]. Due to lacking truly collaborative use cases, this euphory subsided increasingly over the last years. Most applications considered as collaborative were either coexistent, cooperative or ended up behind a safety fence [2]. This contradiction reveals a methodological gap in determining the required HRI technology, especially when comparing industrial robots with external safety devices as a suitable fenceless alternative. Furthermore, planning methodologies for cobot systems are still insufficient and cover only selected parts of the planning entities. Despite the promises of cobots to guarantee easy implementation and commissioning, it turns out that planning such a system includes even more parameters of uncertain data than a fenced-in robot system [2]. Obstacles can be traced back to the requirements that have been defined in ISO/TS 15066, especially regarding application-specific force and pressure measurements and the respective allowed operating speed in collaborative mode [3]. In fact, switching between industrial and collaborative speed by applying external safety devices enlarges the calculation complexity even more. Hence, the traditional generation of industrial robotics and the new generation of cobotics clash: while the cobot world claims that a cobot is the easiest and most profitable solution, the industrial world prefers fenceless industrial robots in many cases. Higher operation speeds, increasing usability, and decreasing prices of industrial robots challenge the stated benefits.

The traditional automation field of machine tending utilizes robot technologies for decades, usually focusing on high volumes of mainly automotive companies in partially or fully fenced in operating areas [4, 5]. Automation optimizes CNC manufacturing regarding motions and defects (see Lean Management and seven types of waste) [6]. On the one hand, robotized movements are faster due to the robot's operating speed and the utilization of double grippers and improve ergonomics by unburdening the operator. On the other hand, the higher accuracy and consistent workpiece positioning of robots eliminate defects occurring from clamping deviations. Within the last years, cobot installations increased rapidly in machine tending with growing potential to become one of the main cobot applications [7]. Especially SMEs face the challenge of recruiting CNC professionals due to demographic change while managing the needs for flexibility, automation, and digitization [8, 9]. Cobots pick up low in-house experience in automation planning and implementation of SMEs with easy programming, plug&play, and online-based training and configuration.

This paper shows how to strategically derive an economic-oriented calculation scheme for HRI based on the current workflow in CNC lathe machine tending. The proposed scheme can support the user in identifying the most efficient robot solution. It is based on the following four steps:

1) Deriving a process pipeline for CNC lathe machine tending from user surveys
2) Choosing the typical use case and defining its constraints
3) Parametrization of the relevant factors and their relations to each other
4) Illustrating the relationship between economic profitability and human-robot-interaction grade.

II. Literature Review

For a comprehensive understanding of the stated problem, the implicated disciplinary fields Human-Robot Interaction,

Corresponding author Christopher Schneider is with YASKAWA Europe GmbH Robotics Division, Yaskawastr. 1, 85391 Allershausen, Germany. phone +49 8166 90 236, email: christopher.schneider@yaskawa.eu.com

Dr. Martina Hutter-Mironovová and Dr.-Ing. Michael Klos are with YASKAWA Europe GmbH Robotics Division, Yaskawastr. 1, 85391 Allershausen, Germany, email: (martina.hutter, michael.klos)@yaskawa.eu.com

Dr.-Ing. Mohamad Bdiwi and Prof. Dr.-Ing. Matthias Putz are with the Fraunhofer Institute for Machine Tools and Forming Technology, Reichenhainer Str. 88, 09126 Chemnitz, email: mohamad.bdiwi@iwu.fraunhofer.de, matthias.putz@zv.fraunhofer.de

Flexible Manufacturing Systems, Machine Tending, and Planning Methodologies are outlined in the following.

For Human-Robot Collaboration (HRC), various classification approaches have been developed in research within the last years [2, 10–12]. While ISO 10218-1:2012 defines the term collaborative operation, ISO TS 15066 differentiates the operation modes (1) Safety rated Monitored Stop (SRMS), (2) Hand Guiding (HG), (3) Speed and Separation Monitoring (SSM) as well as (4) Power and Force Limiting (PFL) [3, 13]. Fenceless production concepts can be achieved with both industrial and collaborative robots by combining different modes. Industrial robots require an additional safety device (i.e., safety scanner) mandatorily to switch between modes 1 and 3. Due to their onboard technology (i.e., torque sensors), cobots can be operated either as standalone (mode 4) or in hybrid mode with external safety (mode 3 and 4) [3, 14–16].

Research in HRC planning implies, among others, the main fields task allocation, resource selection and layout development, mainly focusing on assembly operations. Task division algorithms, based on individual strengths, have been described by [17], [18], and [19]. Although these models consider various evaluation criteria, economic comparisons of different robot system alternatives are not observed. For the planning of HRC applications, different methods have been developed, such as [20], [21], and [22]. In [23], the process is analyzed and described regarding output and ergonomics. A multicriterial pairwise comparison scale evaluates the advantageousness of the alternatives manual execution and HRC. Another approach is a search method and tool for resource selection, layout development, and task assignment in the rough planning phase based on economic and technical criteria [24]. The planning of human-industrial robot cooperation (HIRC) is described in [25] and [26]. In [27], a software planning tool has been developed, which utilizes digital human modeling and industrial robot simulation for evaluation. In [28], multi-criterial evaluation of manual and hybrid workstations has been undertaken, mainly focusing on the economic figures of each alternative.

Different approaches to Flexible Manufacturing System (FMS) planning and design can be found in [16], [29], and [30]. Due to the high automation grade and the predominant fence utilization, those concepts lack transferability to fenceless manufacturing. Besides customized machine tending systems, standard solutions have been developed and are offered at the market. Until today, those systems exist mainly with a partially fenced-in industrial robot equipped with an additional laser scanner. Cobot systems are mostly individual solutions and, therefore, not comparable. In [31], a cycle timed-based method for layout determination of robotized machine cells is presented. A practical planning approach for machine tending with collaborative robots can be found in [32], utilizing the Return on Investment [ROI] as the target figure.

In the current research, machine tending as a potential application field for HRI has not been analyzed in much detail yet. Furthermore, most models lack in the comparison of multiple robot technologies as well as in considering the impact of the hybrid operation on cycle times.

Sophisticated methods, as well as multi-criteria target systems, increase the methodological complexity. A lack of simplicity can be noted for rapid resource suitability evaluation in the rough planning stage, hindering a broad application and acceptance in both research and industry.

III. METHODOLOGY

A. Process Pipeline for CNC Lathe Machine Tending

The fundament of process automation forms a solid understanding of the manual process. Therefore, existing customer projects have been compared to each other, and expert interviews with various CNC turning professionals have been conducted. As a result, the following process pipeline for lathe machine operation has been derived (Fig.1).

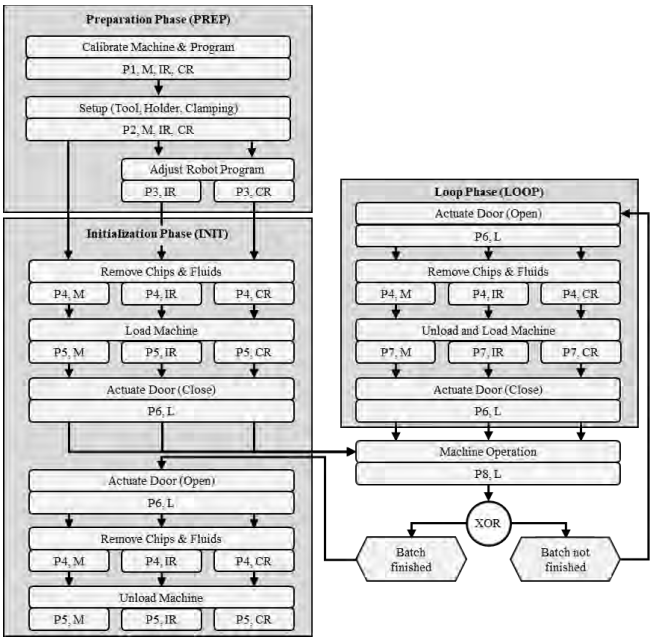


Fig. 1 Process Pipeline for CNC Lathe Machine Tending (own figure)

As can be seen, the processes are distinguished between cognitively challenging and repetitive tasks, which are executed either exclusively manually by the operator (M) or automated with an industrial (IR) or collaborative (CR) robot. While the industrial robot either operates at full speed (FS) or stops, the collaborative robot is usually moving in full speed (FS) and reduces to collaborative speed (CS), as soon as an operator approaches (hybrid mode). Another option is the exclusive operation in collaborative mode at the respective collaborative speed level (CS). Suitable methods to determine the execution times for the alternatives are Methods-Time Measurement (MTM) for the manual tasks and offline simulation studies for the robotized execution [33, 34]. Furthermore, there are lathe machine-based processes (L), which are neutral to the execution form. For both automated process chains, the manual process "robot program adjustment" (P3) is added, including tasks such as reteaching points or tool center point (TCP) calibration.

B. *Use Case Description and Constraints*

Due to the process complexity of milling operations and the respective geometrical variety of the workpieces, the research object has been narrowed to lathe machines and machining centers. According to state-of-the-art, chuck parts are mainly handled with robots, which qualifies them as the workpiece representative for this research.

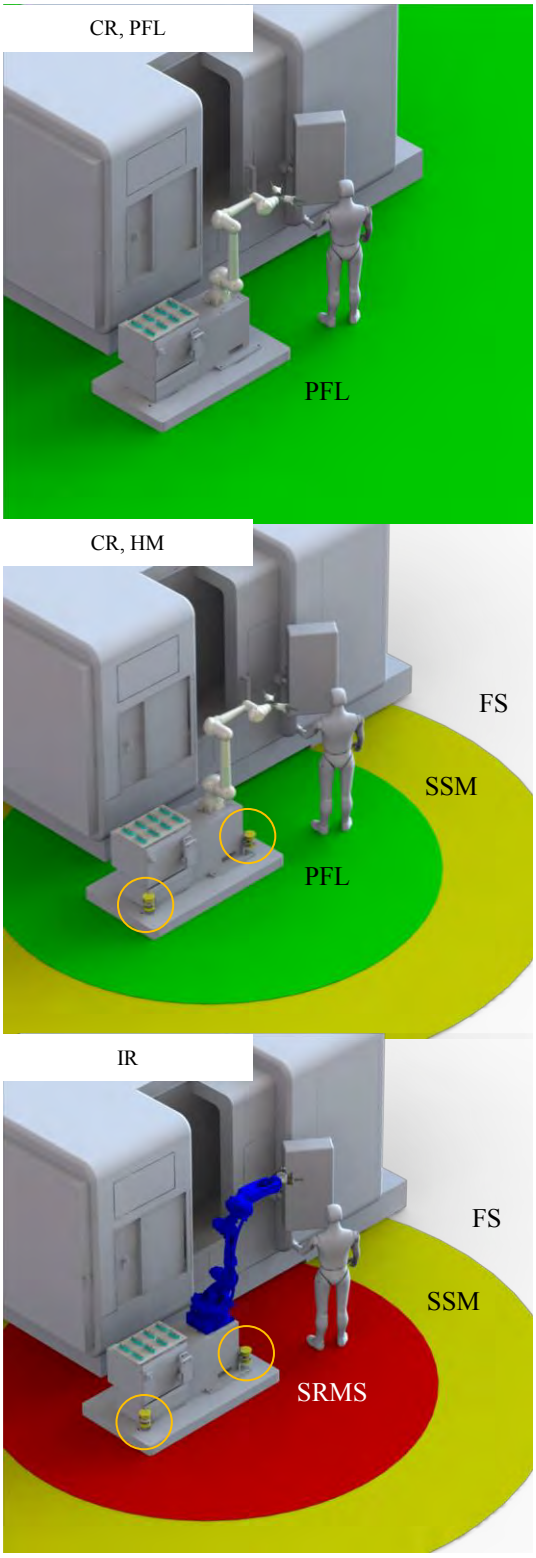


Fig. 2 Exemplary Technical Automation Alternatives (own figure)

It is assumed that the doors stand open at the beginning and are operated by an actuation system. Furthermore, the automated process chains include only those manual processes that must remain in that execution form to pursue a maximum automation level. A collaborative and an industrial robot with comparable technical specifications are used as technology representatives.

Please note that the maximum operating speeds of both robots differ due to kinematic individualities. Cash-out flow relevant factors (i.e., space occupation, maintenance effort, energy consumption) are presumed equal. Double grippers are assumed to handle raw and machine parts simultaneously for cycle time optimization. Based on the illustrated protection field configuration, floor-installed safety laser scanners with a 275 ° scanning angle enable the robot system to switch between operation modes. For this research, only full-speed operation (white zone), SRMS (red zone), and PFL (green zone) are considered. SSM (yellow zone) is excluded to limit this paper's scope and ensure the simplicity of the presented calculation scheme. This gap must be closed in future research by considering the required distances and allowed speeds regarding the robot's stopping times. Figure 2 illustrates the three concluded technical alternatives: 1) collaborative robot in exclusive PFL operation (CR, PFL), 2) collaborative robot in hybrid mode (CR, HM), and 3) industrial robot (IR).

C. *Planning Algorithm*

For this paper, the economic feasibility is emphasized using the net present value NPV [€] as the target figure. In its simplified form, it consists of the initial net investment I_0 [€], the periodical cash-in flows CIF [€] and cash-out flows COF [€], the discount factor q, the time index t [year] and the observed time period T [years] [35].

$$\mathrm{NPV} = -I_0 + \sum_{t=0}^{T}\big(CIF(t) - COF(t)\big) \bullet q^{-t} \qquad (1)$$

The values for I_0 and CIF show vast differences when comparing robot systems, while those for COF can be cut down due to the previously defined assumptions.

$$\triangle NPV = -I_0 + \sum_{t=0}^{T} CIF(t) \bullet q^{-t} \qquad (2)$$

Consequently, only cash-in flow determination is emphasized. To evaluate the single alternatives, the process times P [h] of each process module p must be identified, leading to the following variables: $P_{p,M}$, $P_{p,IR,FS}$, $P_{p,CR,FS}$ *and* $P_{p,CR,CS}$ *as well as* $P_{p,L}$. For better understanding, the calculation of the execution time (ET) [h] in the manual state is demonstrated first. Therefore, the production order is subdivided to single batches B_n, which includes a specific number of workpieces per batch n: the batch size S_n [workpieces]. Before a batch can be produced in the loop phase (LOOP), the enclosing preparation (PREP) and initialization (INIT) phase take place before and after. The sum of all three steps delivers the execution time.

$$ET_{PREP,Bn,M} = \sum_{p=1}^{2} P_{p,M} \qquad (3)$$

$$ET_{INIT,Bn,M} = 2 * \sum_{p=4}^{5} P_{p,M} + 2 * P_{6,L} \quad (4)$$

$$ET_{LOOP,Bn,M} = 2 * P_{6,L} + P_{4,M} + P_{7,M} \quad (5)$$

$$ET_{Bn,M} = ET_{PREP,Bn,M} + ET_{INIT,Bn,M} + ((ET_{LOOP,Bn,M} + P_{8,L}) \bullet S_n) \quad (6)$$

When calculating the execution times for one batch, the total number of batches N can be concluded. The available annual working time W [h] is defined as plain working time, while breaks are taken additionally.

$$N_M = \frac{W}{ET_{Bn,M}}, N_M \in \mathbb{N}^* \quad (7)$$

The available rest capacity R is equivalent to the time that is left over after subtracting the ETs of all executed lots ET_N.

$$R_M = W - N_M \bullet ET_{Bn,M} \quad (8)$$

Within this remaining time (R), a batch share can be produced in the loop phase. Therefore, the PREP and INIT phases need to be executed first. The leftover capacity can be used to manufacture a specific output O_R. As a result, either no or one batch share N_+ can be produced.

$$O_{R,M} = \frac{R_M - \left(ET_{PREP,Bn,M} + ET_{INIT,Bn,M}\right)}{\left(ET_{LOOP,Bn,M} + P_{8,L}\right)} \quad (9)$$

$$O_{R,M} \in \mathbb{N}, \rightarrow N_{+,M} \{0; 1\}$$

By adding up the batch sizes S_n of the single batches B_n with the remaining lot share O_R, the total annual output O_A is determined.

$$O_{A,M} = S_n \bullet N_M + O_{R,M} \quad (10)$$

As a calculation base for the labor release grade LRG, the manual annual execution time is determined, delivering the temporal binding of the operator to the machine. In manual operation, this factor is relatively high because all operations are executed by hand. Only when the machine is processing, the operator is relieved.

$$AET_M = (ET_{PREP,Bn,M} + ET_{INIT,Bn,M}) \bullet (N_M + N_{+,M}) + ET_{LOOP,Bn,M} \bullet (S_n \bullet N_M + O_{R,M}) \quad (11)$$

To calculate the robotized execution times, mixed operations at different speed levels must be considered. Therefore, the human-robot interaction grade α is introduced to cover the time slice, in which the operator stays within the robot's working space for potential interaction. For simplification, it is assumed that the operator stays either out of the robot's operating space (white area) or within (red or green area). Passing through the yellow area and the associated speed reduction in terms of SSM are not considered. Zone violation events are either planned activities (e.g., setup, programming) or unplanned ones (e.g., troubleshooting, in-process workpiece measurements). The behavior of the human-robot interaction grade and its influence on the robot's performance depends on the used technology and assumed consideration period. For cobots, a linear interaction-speed relation is noted because the robot is gradually decreasing from full to collaborative speed. Consequently, the execution times of both modes are set in relation to each other for mixed operation calculations.

$$ET_{PREP,Bn,CR} = \sum_{p=1}^{3} P_{p,CR} \quad (12)$$

$$ET_{INIT,Bn,CR,\alpha*} = 2 * P_{6,L} \quad (13)$$

$$ET_{INIT,Bn,CR,\alpha,FS} = 2 * \sum_{p=4}^{5} P_{p,CR,FS} \quad (14)$$

$$ET_{INIT,Bn,CR,\alpha,CS} = 2 * \sum_{p=4}^{5} P_{p,CR,CS} \quad (15)$$

$$ET_{INIT,Bn,CR} = ET_{INIT,Bn,CR,\alpha*} + ET_{INIT,Bn,CR,\alpha,FS} \bullet (1 - \alpha) + ET_{INIT,Bn,CR,\alpha,CS} \bullet \alpha \quad (16)$$

$$ET_{LOOP,Bn,CR,\alpha*} = 2 * P_{6,L} \quad (17)$$

$$ET_{LOOP,Bn,CR,\alpha,FS} = P_{4,CR,FS} + P_{7,CR,FS} \quad (18)$$

$$ET_{LOOP,Bn,CR,\alpha,CS} = P_{4,CR,CS} + P_{7,CR,CS} \quad (19)$$

$$ET_{LOOP,Bn,CR} = ET_{LOOP,Bn,CR,\alpha*} + ET_{LOOP,Bn,CR,\alpha,FS} \bullet (1 - \alpha) + ET_{LOOP,Bn,CR,\alpha,CS} \bullet \alpha \quad (20)$$

$$ET_{Bn,CR} = ET_{PREP,Bn,CR} + ET_{INIT,Bn,CR} + (ET_{LOOP,Bn,CR} + P_{8,L})) \bullet S_n \quad (21)$$

$$N_{CR} = \frac{W}{ET_{Bn,CR}}, N_{CR} \in \mathbb{N}^* \quad (22)$$

$$R_{CR} = W - N_{CR} \bullet ET_{Bn,CR} \quad (23)$$

$$O_{R,CR} = \frac{R_{CR} - \left(ET_{PREP,Bn,CR} + ET_{INIT,Bn,CR}\right)}{\left(ET_{LOOP,Bn,CR} + P_{8,L}\right)} \quad (24)$$

$$O_{R,CR} \in \mathbb{N} \rightarrow N_{+,CR} \{0; 1\}$$

$$O_{A,CR} = S_n \bullet N_{CR} + O_{R,CR} \quad (25)$$

To observe cash-flows, the annual output deviation, and the labor release grade between the automated and the manual execution is utilized.

$$\triangle O_{A,CR,M} = O_{A,CR} - O_{A,M} \quad (26)$$

$$AET_{CR} = ET_{PREP,Bn,CR} \bullet (N_{CR} + N_{+,CR}) \quad (27)$$

$$LRG_{CR} = 1 - \frac{AET_{CR}}{AET_M} \quad (28)$$

Finally, the overall cash-in flow CIF is calculated by multiplying the LRG with the annual labor cost of the operator C on the one side and the $\triangle O_A$ with the value creation per workpiece VC on the other.

$$CIF(t)_{CR} = LRG_{CR} \bullet C + \triangle O_{A,CR,M} \bullet VC \quad (29)$$

Then, the CIF and I_0 values for each automation alternative are inserted into the NPV equation.

$$NPV_{CR} = -I_{0,CR} + \sum_{t=0}^{T} CIF_{CR}(t) \bullet q^{-t} \quad (30)$$

For the technical variant "CR, PFL", the calculations can be strongly simplified since the allocation of the α-affected ET's can be neglected. Consequently, the same calculations can be used by cutting out the allocations.

On the other side, industrial robots, stop in the event of safety zone penetration. As a counterpart for the execution time in 100% full speed, the actual value in a 100% stop situation cannot be calculated, hindering time estimation in mixed operation. In theory, the order would never be completed in the 100% stop scenario and the execution time converges to infinity with the expression:

$$P_{p,IR} = \frac{P_{p,IR,FS}}{1-\alpha} \quad (31)$$

Exemplary calculations verify this exponential behavior. With an increasing share of stopping periods, the robot must subsequently compensate. Hence, the entire operating time (e.g., one year) is assumed, in which stopping events occur irregularly and need to be compensated for finishing the batch. By considering a short period of time (e.g., one shift) with regular stopping times (planned activities), the expected waiting times of the next cycle are not included. Instead, the execution time results from adding the unproductive time slice to the theoretical ideal execution time at full speed:

$$P_{p,IR} = P_{p,IR,FS} \bullet (1 + \alpha) \quad (32)$$

However, this calculation is theoretically correct and an idealistic assumption that is not transferrable to the unpredictable character of production. Therefore, equation (31) is used for the following calculations.

$$ET_{PREP,Bn,IR} = \sum_{p=1}^{3} P_{p,IR} \quad (33)$$

$$ET_{INIT,Bn,IR,\alpha*} = 2 * P_{6,L} \quad (34)$$

$$ET_{INIT,Bn,IR,\alpha,FS} = 2 * \sum_{p=4}^{5} P_{p,IR,FS} \quad (35)$$

$$ET_{INIT,Bn,IR} = ET_{INIT,Bn,IR,\alpha*} + \frac{ET_{INIT,Bn,IR,\alpha,FS}}{(1-\alpha)} \quad (36)$$

$$ET_{LOOP,Bn,IR,\alpha*} = 2 * P_{6,L} \quad (37)$$

$$ET_{LOOP,Bn,IR,\alpha,FS} = P_{4,IR,FS} + P_{7,IR,FS} \quad (38)$$

$$ET_{LOOP,Bn,IR} = ET_{LOOP,Bn,IR,\alpha*} + \frac{ET_{LOOP,Bn,IR,\alpha,FS}}{(1-\alpha)} \quad (39)$$

$$ET_{Bn,IR} = ET_{PREP,Bn,IR} + ET_{INIT,Bn,IR} + (ET_{LOOP,Bn,IR} + P_{8,L}) \bullet S_n \quad (40)$$

$$N_{IR} = \frac{W}{ET_{Bn,IR}}, N_{IR} \in \mathbb{N}^* \quad (41)$$

$$R_{IR} = W - N_{IR} \bullet ET_{Bn,IR} \quad (42)$$

$$O_{R,IR} = \frac{R_{IR} - \left(ET_{PREP,Bn,IR} + ET_{INIT,Bn,IR}\right)}{\left(ET_{LOOP,Bn,IR} + P_{8,L}\right)} \quad (43)$$

$$O_{R,IR} \in \mathbb{N} \rightarrow N_{+,IR}\{0;1\}$$

$$O_{A,IR} = S_n \bullet N_{IR} + O_{R,IR} \quad (44)$$

$$\triangle O_{A,IR,M} = O_{A,IR} - O_{A,M} \quad (45)$$

$$AET_{IR} = ET_{PREP,Bn,IR} \bullet (N_{IR} + N_{+,IR}) \quad (46)$$

$$LRG_{IR} = 1 - \frac{AET_{IR}}{AET_M} \quad (47)$$

$$CIF(t)_{IR} = LRG_{IR} \bullet C + \triangle O_{A,IR,M} \bullet VC \quad (48)$$

$$NPV_{IR} = -I_{0,IR} + \sum_{t=0}^{T} CIF_{IR}(t) \bullet q^{-t} \quad (49)$$

As a result, three net present values can be calculated, one for each technical alternative. By comparing these variants to each other, the relative advantageousness can be concluded as decision assistance.

$$NPV_{IR} > NPV_{CR}: Industrial\ Robot\ preferable \quad (50)$$

$$NPV_{IR} < NPV_{CR}: Collaborative\ Robot\ preferable \quad (51)$$

D. Relationship between Economic Profitability and Human-Robot-Interaction Grade

To illustrate the application of the proposed calculation scheme, an example with realistic values is given. The HRI grade α is used as the sensitivity target figure, progressing in steps of 0,1 from 0 to 0,9. By this, the influence of this factor α on the overall decision is illustrated (see Fig. 3). The industrial robot (IR) is the preferred solution up to an interaction grade of 20 % since this robot has the highest velocity and stops occasionally. With increasing interaction grade, the IR profitability decreases exponentially because the stopping time slices are getting higher, making this alternative

increasingly undesirable in dynamic environments. Up to an alpha value of 84 %, the NPV is absolutely advantageous (NPV > 0). Between 20 % and 50 % interaction grade, the collaborative robot in hybrid mode (CR, HM) is relatively advantageous because the robot can switch between full- and collaborative speed due to external laser scanners. With increasing interaction grade, the net present value decreases linearly and gets closer to the collaborative robot in exclusive PFL operation (CR, PFL) because the operating speeds increasingly converge to those in the PFL operation. The CR, PFL variant is the dominant solution at an interaction grade of 50 % or higher. If the robot operates most of the time at collaborative speed, the additional investment of the CR, HM is no longer beneficial since the operating speeds between FS and PFL operations are very small during the main times. Consequently, the decision-maker can make a sound investment decision based on the expected interaction grade.

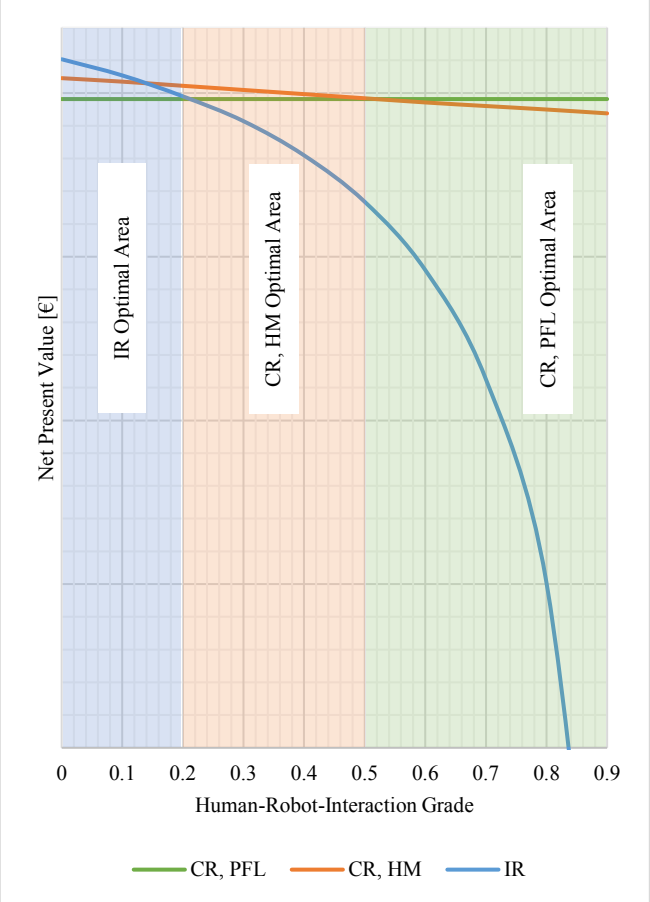


Fig. 3 Sensitivity analysis of the HRI grade α (own figure)

IV. Results and Discussion

The results of this paper are intended to help to manage the decision complexity in the early automation planning stage. With the mathematical model, a substantial discussion base about the practicability of collaborative robots has been presented to stimulate reconsiderations of the cobot hype while simultaneously encouraging traditional automation enthusiasts for this new technology. The focus of this work is the systematic categorization of the economic structure of fenceless robot cells. The financial framework for classic automation was adopted to cobots until now, without the precise exposure of the cobot-specific factors, such as the HRI grade. Particular interest lies in the behavior individualities for both robot types regarding their linearity and advantageousness, which has also been plausibly illustrated in the example. Innovative is the identification of an absolute unprofitable zone and the subdivision in different decision areas. Therefore, an economically justified assertation about the fitness of cobots as a credible automation alternative can be made, especially for low-volume production with various setups and process interventions. Limitations occur in the transferability to other use cases due to the strict application characterization. Partial fencing and SSM, as it is usual in machine tending, were excluded from consideration leading to mismatches with reality. Due to the strong focus on α, the decision is highly sensitive to the estimation uncertainties of this factor. For counterbalancing, a sensitivity analysis similar to figure 3 can assist in defining realistic solution spaces. A practical solution for accurate α-value estimation can be achieved by tracking the operator behavior in the manual state over a relevant period of time. State-of-the-art laser scanners provide a data collection interface to create heat maps of the violated safety zones.

V. References

[1] PwC, *The new hire: How a new generation of robots is transforming manufacturing*, 2014.

[2] W. Bauer, M. Bender, M. Braun, P. Rally, and O. Scholtz, *Lightweight robots in manual assembly - best to start simply!*, 2016.

[3] *ISO TS 15066:2016: Robots and robotic devices - Collaborative robots,* International Organization for Standardization (ISO), 2016.

[4] A. Lazinica and H. Kawai, *Robot Manipulators: New Achievements*. Rijeka, Croatia: InTech, 2010.

[5] U. Winkelhage, *The Digital Transformation of the Automotive Industry: Catalysts, Roadmap, Practice*: Springer International Publishing AG, 2018.

[6] J. R. Henry, *Achieving Lean Changeover: Putting SMED to Work*: CRC Press, 2017.

[7] BIS Research, *Global Collaborative Robot Hardware Market, Analysis & Forecast, 2016-2021 (Focus on Major Industries and Applications)*, 2016.

[8] P. Brown, A. Green, and H. Lauder, *High Skills: Globalization, Competitiveness, and Skill Formation,* 2nd ed. New York: Oxford University Press, 2003.

[9] T. Redman and A. Wilkinson, *Contemporary Human Resource Management: Text and Cases,* 3rd ed.: Pearson Education, 2009.

[10] M. Fechter, C. Seeber, and S. Chen, “Integrated Process Planning and Resource Allocation for Collaborative Robot Workplace Design,” *Procedia CIRP*, vol. 72, pp. 39–44, 2018.

[11] R. Parasuraman, T. B. Sheridan, and C. D. Wickens, “A model for types and levels of human interaction with automation,” *IEEE Transactions on Systems, Man and Cybernetics - Part A: Systems and Humans*, vol. 30, pp. 286–297, 2000.

[12] M. Bdiwi, M. Pfeifer, and A. Sterzing, "A new strategy for ensuring human safety during various levels of interaction with industrial robots," *CIRP Annals - Manufacturing Technology*, vol. 66, no. 1, 2017.
[13] *ISO 10218-1:2011: Robots and robotic devices - Safety requirements for industrial robots: Part 1: Robots,* International Organization for Standardization (ISO), 2011.
[14] M. Gurgul, *Industrial robots and cobots: Everything you need to know about your future co-worker*, 2018.
[15] M. F. Zaeh, Ed., *Enabling Manufacturing Competitiveness and Economic Sustainability*, 2013.
[16] M. I. A. Ferreira, J. S. Sequeira, G. S. Virk, M. O. Tokhi, and E. E. Kadar, *Robotics and Well-Being*. Switzerland: Springer Nature, 2019.
[17] A. Casalino, F. Cividini, A. M. Zanchettin, L. Piroddi, and P. Rocco, "Human-robot collaborative assembly: a use-case application," *IFAC PapersOnLine*, vol. 51, no. 11, pp. 194–199, 2018.
[18] F. Ranz, V. Hummel, and W. Sihn, "Capability-based task allocation in human-robot-collaboration," *Procedia Manufacturing*, vol. 9, pp. 182–189, 2017.
[19] J. E. C. Mateus, E. Aghezzaf, D. Claeys, V. Limere, and J. Cottyn, "Method for transition from manual assembly to Human-Robot collaborative assembly," *IFAC PapersOnLine*, vol. 51, no. 11, pp. 405–410, 2018.
[20] S. Bouchard, *Lean robotics: A guide to making robots work in your factory*: Levis, Quebec, 2018.
[21] P. Tsarouchi *et al.,* "A decision making framework for Human Robot Collaborative workplace generation," *Procedia CIRP*, vol. 44, pp. 228–232, 2016.
[22] G. Michalos, S. Makris, J. Spiliotopoulos, I. Misios, P. Tsarouchi, and G. Chryssolouris, "ROBO-PARTNER: Seamless Human-Robot Cooperation for Intelligent, Flexible and Safe Operations in the Assembly Factories of the Future," *Procedia CIRP*, vol. 23, pp. 71–76, 2014.
[23] S. Heydaryan, J. S. Bedolla, and G. Belingardi, "Safety Design and Development of a Human-Robot Collaboration Assembly Process in the Automotive Industry," *Applied Sciences*, vol. 8, no. 344, 2018.
[24] G. Michalos, J. Spiliotopoulos, S. Makris, and G. Chryssolouris, "A method for planning human robot shared tasks," *CIRP Journal of Manufacturing Science and Technology*, 2018.
[25] V. Gopinath, F. Ore, S. Grahn, and K. Johansen, "Safety-focussed design of collaborative assembly station with large industrial robots," *Procedia Manufacturing*, vol. 25, 503-210, 2018.
[26] F. Ore, L. Hansson, and M. Wiktorsson, "Method for design of human-industrial robot collaboration workstations," *Procedia Manufacturing*, vol. 11, pp. 4–12, 2017.
[27] F. Ore, B. R. Vemula, L. Hanson, and M. Wiktorsson, "Human-Industrial Robot Collaboration: Application of simulation software for workstation optimisation," *Procedia CIRP*, vol. 44, pp. 181–186, 2016.
[28] S. Consiglio, Gelinger, G., and N. Weinert, "Development of Hybrid Assembly Workplaces," *Annals of the CIRP*, vol. 56, no. 1, pp. 37–40, 2007.
[29] I. Gania, A. Stachowiak, and J. Oleskow-Szlapka, "Flexible Manufacturing Systems: Industry 4.0 Solution," *24th International Conference on Production Research (ICPR 2017)*, pp. 57–62, 2017.
[30] T. Koltai, K. E. Stecke, and V. Juhasz, Eds., *Planning of Flexibility of Flexible Manufacturing Systems*, 2004.
[31] A. Tubaileh, "Layout of robot cells based on kinematic constraints," *International Journal of Computer Integrated Manufacturing*, 2015.
[32] RobotIQ, *The Machine Tending Playbook*, 2018.
[33] D. W. Karger, *Engineered Work Measurement: The principles, techniques, and data of Methods-Time Measurement, background and foundations of work measurement and Methods-Time Measurement, plus other related material,* 4th ed. New York: Industrial Press, Inc., 1987.
[34] B. Leon, A. Morales, and J. Sancho-Bru, *From Robot to Human Grasping Simulation*. Switzerland: Springer International Publishing, 2014.
[35] P. Vernimmen, P. Quiry, M. Dallocchio, Y. Le Fur, and A. Salvi, *Corporate Finance: Theory and Practice,* 4th ed.: John Wiley and Sons, Ltd, 2014.

Convolutional Neural Network for detection of fluid level in bores for assembly automation

Alexej Simeth
Department of Engineering
University of Luxembourg
Luxembourg
alexej.simeth@uni.lu

Atal A. Kumar
Department of Engineering
University of Luxembourg
Luxembourg

Peter Plapper
Department of Engineering
University of Luxembourg
Luxembourg

***Abstract*—Assembly tasks often require high level of perception, skill and logical thinking, which is challenging for automation. Increased customization and shortening product life cycles are further inhibiting automation. Monitoring the fluid level in gluing processes in assembly of highly individualized products is one of such tasks. To automate the task, an AI-based model is proposed in this paper. Target is the automation of a gluing process in a final assembly. A convolutional neural network is applied to extract features of images taken of a gluing process. A support vector machine classifier is trained with these features and used to identify the level of liquid in bores. A Monte-Carlo-simulation is applied to validate the model. Overall, the proposed model achieves 98% accuracy in classifying liquid level. Testing the model on a technology demonstrator showcases similar results.**

Keywords—liquid detection, convolutional neural network, artificial intelligence, assembly automation

I. INTRODUCTION

The automation of a pick and place and gluing process in a high variance low volume assembly scenario is in focus of the authors' research. In this assembly, parts are inserted into carrier workpieces and then bonded with glue. A standard volumetric control of the gluing process it not possible due to the specific properties of the carrier workpiece and the in consequence unknown required glue volume. A model to detect the glue in order to start, monitor, and stop the gluing process is presented in this paper. Since there are different sensing tasks in the overall process which can be covered with an optical sensor, an industrial grade fixed lens camera is applied.

A brief overview of related work is given in the next chapter followed by the applied methodology, experimental setup, results, and conclusion in subsequent chapters. An elaborated version of this paper was presented on APMS conference 2021 [1].

II. RELATED WORK

To detect specific features, objects, parts, etc. in manufacturing automation. Existing detection problems are solved differently depending on the exact task. In the area of glue, liquid, and fluid detection in production environment, most solutions are proposed in the context of bottle filling and electronics manufacturing [2]–[8]. The majority of reviewed publications applies a static edge detection algorithm in order to identify the surface or the silhouette of the liquid [2]–[4]. Once the boarder of the liquid is detected it is compared to a predefined position or threshold in order to measure the width of an applied glue line [2] or to determine the upper filling level of a bottle [3], [4]. Another method is the image segmentation based on colour. In [5] the dark liquid inside the monitored bottle is separated from the image background solely based on colour values and the contour of the identified area is taken to estimate the filling level. A less complex algorithm based on changes in histogram is applied in [6] to measure the volume of a liquid and a bubble phase in translucent cylindric vessels. Key of this method is the experimental setup with special light source and background, which amplifies the liquid and bubble phase. In [7] the authors compare a conventional detection approach to identify liquid and bubble phase in bottles via mean filters with convolutional neural network (CNN) based approach. The classification results are slightly improved with the CNN, despite its very simple structure (three layers). To detect variation in a dispensed glue drop on a workpiece, the authors of [8] selected principal component analysis. Target is to identify whether a fault in the dispensing system exists.

Most of the introduced methods by other researchers apply conventional models to identify the surface, silhouette or colour of a liquid and compare it against predefined thresholds or references. Further they are depending on a constant environment with specific settings for each feature to be detected. In order to start, monitor, and stop a gluing process in a constantly changing production environment, e.g., high difference between products, robot mounted system, more robust and flexible detection algorithms are necessary.

III. METHODOLOGY

The authors present in this paper an AI-based model for the robust detection of fluid in bores in a workpiece as part of the automation of an industrial gluing process. The model was developed following the in workflow for supervised and unsupervised learning [9]. The glue detection problem is reduced to a binary problem with the two classes "full" and "empty", i.e., sufficient and insufficient amount of glue. An initial data set is generated by taking images and labelling these as "full" or "empty". In order to increase the amount of data, the initial data set is augmented according to the method of importance sampling [10]. Focus of augmentation are data points, which are expected to be more influential as data points close to the label change. The data set is split in the subsequent

step into training and test set randomly. The model is trained on the training set. In the developed model a pre-trained CNN is applied to extract image features, which are used for classification. Using a pre-trained CNN is usually significantly faster and simpler than designing a new network [11]. A machine learning algorithm is trained on the extracted image features by the CNN and used to classify the test set. The achieved performance and accuracy of machine learning algorithms is similar to deep learning classifiers by reducing complexity and computational effort [12]. In contrast to the initial publication [1], the less complex eight layer deep CNN AlexNet and a support vector machine (SVM) image classifier are applied in the proposed model.

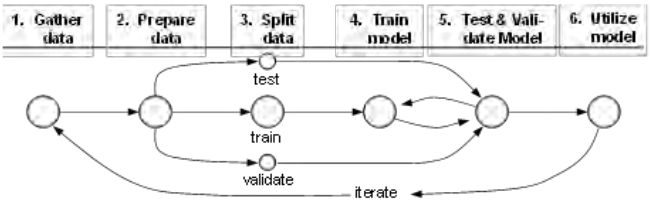


Fig. 1. Workflow for supervised and unsupervised learning [9, p. 22]

To analyse the impact of the randomly selected training and test set on the outcome, a Monto-Carlo-Simulation is conducted [13]. This cross-validation is conducted, since the learned interdependencies by training the model with the training set are highly depending on the random drawing of the data.

The model is then utilized in a live gluing process. New images are captured during gluing and are classified by the developed model. Based on the classification result the gluing process is automatically stopped.

IV. Experimental Set-up

A. Generation of data sets

The initial data set is created on a test stand by recording manually conducted gluing trials. As depicted in Fig. 2. a), a smartphone cradle is mounted on a horizontal aluminium profile above the workpiece in order to capture the gluing process. Via a fixed nozzle, glue is filled trough an inlet hole of the inserted part into the workpiece. The process is recorded and the region of interest (ROI), the outlet hole of the inserted workpiece, is cropped out of the video frames. The resulting images of 41x41x3 pixels are labelled into the categories (labels) "full" and "empty" and transformed into the required image format of the AlexNet input layer, which is 227x227x3 pixels. 605 initial images are obtained via the described procedure, which are later used for data set optimization. In TABLE I sample images of the two labels are given in row "Label".

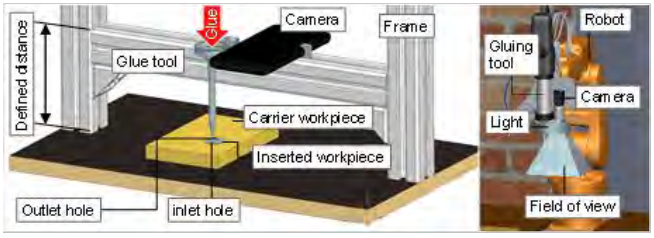


Fig. 2. a): Designed test stand. b): Demonstrator. Source: [1]

B. Optimization of data set

Enough data is required to train the developed model. The size of the data set is increased by augmentation of the existing data. A selection of created images is rotated and mirrored according to the importance sampling method. It is expected that images labelled as "empty" but close to the label "full" and vice versa have a high impact on the decision rule. Thus, these images are augmented. On overview of the final data set is given in TABLE I. The total data set size is increased from 605 to 3,000 images via augmentation of selected images following importance sampling.

TABLE I. Generated and Optimized Data Set

Data	Frames from videos taken on test stand			
Label	Empty (380 images)		Full (225 images)	
	Divide into more and less influential data for "importance sampling"			
Sublabel	Clearly empty	Close to full	Sufficiently full	Overfull
		augment	*augment*	
Data set	1830 images		1170images	

C. Validation

To balance the size of the classes 1170 images are randomly selected from the label "empty". The resulting data set is randomly split in the ratio 70:30 into training data and test data. The model is trained on the training set. I.e., the image features of the images in the test set are extracted by the CNN and used to train the SVM image classifier. The model is then used to classify the test set. The result of the classification is compared to the original label. The whole process is repeated 500 times in an MCS. Each epoch has a different random selection of the label "empty", training set, and test set.

After training and testing the model with images created on the test stand, the model is applied on a technology demonstrator. Here, new images are taken by a fixed lens camera mounted on a robot. The camera is held perpendicular above the carrier workpiece so that the ROI, i.e., the outlet hole, can be monitored. Glue is pumped automatically through a nozzle into the inlet hole of the inserted part. The scenery is illuminated with a red LED ring light with the target to reduce environmental impacts. The whole setup is depicted in Fig. 2. b).

The ROI is cropped out of the red image plane of the taken images of the live gluing process and the image patch is transformed into the required input size of the CNN. The image features are extracted by the CNN and classified by the trained SVM image classifier. The gluing process is stopped once an image is classified as "full". The trials are conducted in bright and dark environmental conditions.

V. Results

With the developed model an overall prediction accuracy of >97% is achieved. 500 different, randomly selected configurations of the class "empty", training set, and test set are calculated in the cross validation with an MCS. The prediction accuracy varies in an interval from 90-99% over

the 500 simulations (s. Fig. 3). Each wrongly classified image is analysed. It is found that these images are directly at boarder to the other label. Considering the whole video from where the images frames are grabbed, each of the wrongly classified images is either the last image, which is labelled as "empty" or the fist image of a video, which is labelled as "full" (cf. TABLE I).

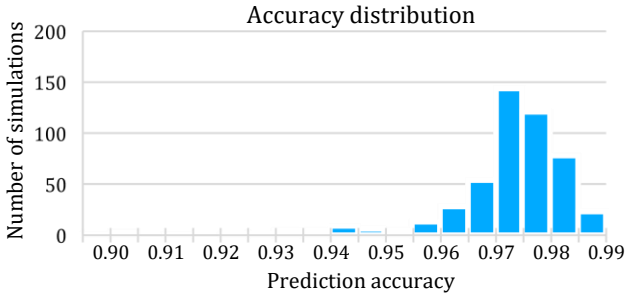
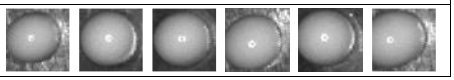


Fig. 3. Results of overall classifier performance with AlexNet feature extraction based on MCS

On a technology demonstrator the proposed model achieved similar results during gluing trials, where it is used to stop a gluing process based on the classification of the glue level. In total 35 gluing trials are conducted on the demonstrator using the developed model. In all cases, the gluing process was stopped correctly based on the glue level classification. The differences of the images used for training and the images taken on the demonstrator are significantly different. Further, the trials are conducted in a daylight and night scenario. A selection of images classified as "full" on the demonstrator is given in TABLE II.

TABLE II. "FULL" IMAGES FROM DEMONSTRATOR VALIDATION

Scenario	Sample images
Daylight scenario	
Night scenario	

VI. CONCLUSION AND FUTURE WORK

A new model to robustly detect glue level in workpieces is proposed in this paper. The authors present a hybrid detection model consisting of the pre-trained neural network AlexNet and a support vector machine image classifier. The convolutional neural network is applied to extract image features of images taken with a smartphone camera from a gluing process on a designed test stand. After training of the image classifier with the extracted image features, the classifier is used to classify new images. The classification is reduced to a binary classification problem and the generated data set is labelled into the two classed "full" and "empty".

The overall achieved average accuracy of the proposed model in classification of the test set images is >97%. The result is cross-validated with a Monte-Carlo-Simulation – 500 random distributions of training and test set data of the initial data set are simulated. The application of the developed detection model on a technology demonstrator indicated similar results. In all conducted gluing trials on the demonstrator, the model stopped the gluing process correctly based on glue level classification. The application of the detection model in a real gluing process has a huge impact on the boundary conditions. Especially the lighting situation and the image resolution is significantly different to the initial data set, which highlights the robustness of the AI-based detection model. In ongoing experiments, the proposed model is compared to other approaches from both computer science and production perspective.

REFERENCES

[1] A. Simeth, J. Plaßmann, and P. Plapper, "Detection of fluid level in bores for batch size one assembly automation using convolutional neural network," in *Advances in Production Management Systems. Artificial Intelligence for Sustainable and Resilient Production Systems. IFIP International Federation for Information Processing. APMS 2021, IFIP AICT 632.*, A. Dolgu et al., Ed. Cham: Springer International Publishing, 2021, pp. 1–8.

[2] Yuchung Huang, Chiahsiang Wu, and Chih-Yuan Chang, "An application of image processing in flat panels," in *2012 International Conference on Wavelet Active Media Technology and Information Processing (ICWAMTIP)*, Dec. 2012, pp. 8–11, doi: 10.1109/ICWAMTIP.2012.6413427.

[3] M. A. A. Felipe, T. V Olegario, N. T. Bugtai, and R. G. Baldovino, "Vision-based Liquid Level Detection in Amber Glass Bottles using OpenCV," in *2019 7th International Conference on Robot Intelligence Technology and Applications (RiTA)*, Nov. 2019, pp. 148–152, doi: 10.1109/RITAPP.2019.8932807.

[4] K. J. Pithadiya, C. K. Modi, and J. D. Chauhan, "Machine vision based liquid level inspection system using ISEF edge detection technique," in *Proceedings of the International Conference and Workshop on Emerging Trends in Technology*, Feb. 2010, pp. 601–605, doi: 10.1145/1741906.1742044.

[5] M. M. Gonzalez Ramirez, J. C. Villamizar Rincon, and J. F. Lopez Parada, "Liquid level control of Coca-Cola bottles using an automated system," in *2014 International Conference on Electronics, Communications and Computers (CONIELECOMP)*, Feb. 2014, pp. 148–154, doi: 10.1109/CONIELECOMP.2014.6808582.

[6] H. Ma and L. Peng, "Vision Based Liquid Level Detection and Bubble Area Segmentation in Liquor Distillation," in *2019 IEEE International Conference on Imaging Systems and Techniques (IST)*, Dec. 2019, pp. 1–6, doi: 10.1109/IST48021.2019.9010097.

[7] T. Beck, B. Gatternig, and A. Delgado, "Schaum-und Füllstanderkennung mittels optischer Systeme mit neuronalen Algorithmen," in *Fachtagung „Experimentelle Strömungsmechanik"*, 2019, pp. 17.1-17.7.

[8] Z. Huang, V. C. Angadi, M. Danishvar, A. Mousavi, and M. Li, "Zero Defect Manufacturing of Microsemiconductors – An Application of Machine Learning and Artificial Intelligence," in *2018 5th International Conference on Systems and Informatics (ICSAI)*, Nov. 2018, pp. 449–454, doi: 10.1109/ICSAI.2018.8599292.

[9] R. Akerkar, "Machine Learning," in *Artificial Intelligence for Business*, 1st ed., R. Akerkar, Ed. Springer International Publishing, 2019, p. 81.

[10] B. Arouna, "Adaptative Monte Carlo Method, A Variance Reduction Technique," *Monte Carlo Methods Appl.*, vol. 10, no. 1, pp. 1–24, Jan. 2004, doi: 10.1515/156939604323091180.

[11] MathWorks, "Pretrained Deep Neural Networks," 2021. https://www.mathworks.com/help/deeplearning/ug/pretrained-convolutional-neural-networks.html (accessed Apr. 08, 2021).

[12] A. Kumar T.K., R. Vinayakumar, S. Variyar V.V., V. Sowmya, and K. P. Soman, "Convolutional Neural Networks for Fingerprint Liveness Detection System," in *2019 International Conference on Intelligent Computing and Control Systems (ICCS)*, May 2019, pp. 243–246, doi: 10.1109/ICCS45141.2019.9065713.

[13] C. Andrieu, N. De Freitas, A. Doucet, and M. I. Jordan, "An introduction to MCMC for machine learning," *Mach. Learn.*, vol. 50, no. 1, pp. 5–43, 2003, doi: 10.1023/A:1020281327116.

A Concept for Object Detection and Localization based on AI and Computer Vision as a Basis for Bin Picking Applications

1st Rainer Müller
Chair for Assembly Systems
Saarland University.
ZeMA - Center for Mechatronics and Automation Technology gGmbH
Saarbrücken, Germany.
rainer.mueller@zema.de

2nd Ali Kanso
ZeMA - Center for Mechatronics and Automation Technology gGmbH
Saarbrücken, Germany.
a.kanso@zema.de

3rd Stefan Marx
ZeMA - Center for Mechatronics and Automation Technology gGmbH
Saarbrücken, Germany.
s.marx@zema.de

4thMd Jonybul Islam
ZeMA - Center for Mechatronics and Automation Technology gGmbH
Saarbrücken, Germany.
j.islam@zema.de

***Abstract*—With the deployment of Industrial Robots in the last decades, picking objects from scattered environments has become inevitable. Human hand-eye coordination allows humans to recognize and sort different objects. But the engagement is vastly cumbersome and pestilent as they have to do the identical task monotonously. For this particular reason, bin picking tasks have enormous development potential per effectiveness and efficiency comparing humans. The robot is a potential candidate to automate the picking process, but it still has deficiencies in detecting objects in different environments. The limited deployment of AI algorithms in existing solutions restricts the integration of automated picking applications. Moreover, existing solutions require expensive equipment and additional development time. In this paper, we are portraying a method to detect objects in heterogeneous environments using a 2D camera and AI algorithms. Thereby, the position and orientation of the object are determined. Moreover, the proper gripping pose of the object is calculated concerning the robot coordinate system. This method is executed and evaluated in detecting complete assembled nut-bolts on a plane surface.**

***Index Terms*—Bin Picking, Image Processing, Deep Learning, CNN, YOLO.**

I. Introduction

Over the last decades, the subject of picking up objects from specified locations or boxes is a research part in the field of automation technology. In ordered picking or machine tending, each of the scenarios needs inspection for picking objects. Predominantly, this task was perpetually performed by humans with much efficiency and quality. Human has extraordinary hand-eye coordination to recognize and pick up objects without any prior knowledge. But they are error-prone, and the fatigue or boredom effect can influence their efficiency. With the industrial revolution, this task seeks for more productivity along with more momentum of execution. For instance, picking up objects in an ordered environment in a typical warehouse burns 50% - 75% of the total operating cost [4]. Thus, the robotic solution for bin picking became inevitable. Industrial robots are manoeuvred in the production and assembly lines to generate high flexibility for the high level of automation. These robots are freely programmable and there are some common ways to do this which are more or less involved in the problem that they are not flexible and adaptive regarding changes in the environment. The use of robotic automation means an effective and efficient solution to the bin picking problem with reduced cost. Notwithstanding, the necessity of automated bin picking solution is implacable, robust use of AI-based solutions is still an extensive quest. Existing research has produced a wide variation and dimension of the described enigma. These methods are often expensive and utilized for individual solutions. High costs are thereby incurred, for example, for equipment such as cameras.

In this paper, we are presenting a solution for detecting objects and gripping pose estimation in bin picking scenario. Our approach consists of a camera mounted on the robot flange equipped with an OnRobot RG2 gripper [13]. With the camera, we observe the environment straight below the robot flange to detect the desired object. Beforehand, we modify a pre-trained model to append our desired object. We can use this appended pre-trained model for different objects in the future. Afterward, we utilize the OpenCV [1] [7] contour to figure out the object area of our intended object and process the best gripping pose for the object for 2 finger gripper. Eventually, we have transformed this gripping pose from pixel to robot-base coordinate system and convey the output to achieve our bin-picking task. For demonstration, we have implemented the system with the Kuka KR6 industrial robot.

II. Related Work

Zeng et. al [18] have illustrated a method for grasping and recognizing both known and unknown objects in a cluttered environment without requiring any task-specific training model. Initially, it uses a diagnostic target retrieval frame to map visual observations to actions that derive dense pixel probability maps of the possibilities of four different primitive retrieval actions. It then performs the action at its highest retention and recognizes the selected objects using a cross-domain image classification framework that matches observed images with product images. Since product images are available for a large number of objects (e.g., from the web), the system works out-of-the-box for intended objects without additional data acquisition or retraining. But it has a complex and expensive setup with four cameras and multiple fundamental setups. Petersen et. al [14] presented a more complete scenario focusing on three process decisions of picking, storing and routing of ordered picking scenario. This method leans more towards the best routing planning outline with a specific object in the current time frame. Moreover, Kraft et al [8] has introduced an approach that is capable of automatically generating good grips using a dynamic grip simulator and post-processing methods, along with an offline learning approach that can adjust grip priorities based on past performance. Our work is most aligned with Holz et. al [6]. Here authors have proposed a detecting method after picking up objects in the different unordered piles with compounds of simple shape and contour primitives. This method can give proper object grasping planning and afterward detect the grasped object to sort properly.
Primarily, we focused on detecting objects, and then we attain the best gripping position to act with and proceed for the path planning to grip and place our intended object.

III. Object Detection

A. Camera Calibration

The aim of camera calibration is to determine the transformation between the camera unit pixel and a physical unit of length (e.g., millimeter). The result can then basically be used to convert pixels of the camera into world coordinates, and vice versa. One of the most commonly used camera models to describe a mapping from three-dimensional to two-dimensional space is the pinhole camera model [see fig 1].

The coordinates of the projection of the point P onto the image plane in the coordinate system i can be described with the help of the second intercept theorem from mathematics. From the length relations between the camera focal length f and the ${}^{c}Z_p$-coordinate as well as between the ${}^{i}X_p$ -coordinate and ${}^{c}X_p$-coordinate respectively ${}^{i}Y_p$-coordinate and ${}^{c}Y_p$-coordinate follows [7]:

$$ {}^{i}X_p = f \times \frac{{}^{c}X_p}{{}^{c}Z_p} \text{ and } {}^{i}Y_p = f \times \frac{{}^{c}Y_p}{{}^{c}Z_p} \tag{1} $$

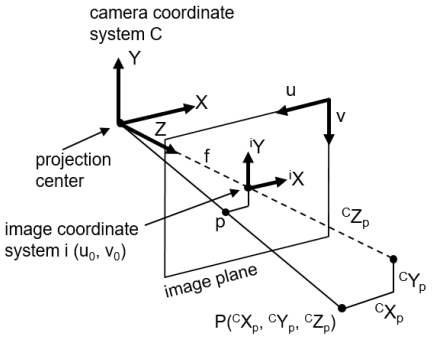


Fig. 1: Pinhole Camera Mode

With the parameters u_0 and v_0 first the displacement of the coordinate system i with respect to the coordinate system B in the corner of the image plane is described. Furthermore, two factors s_x and s_y (unit pixel/mm) are introduced, which define the pixel density along the respective image axis in relation to a physical unit of length. The pixel coordinates ${}^{i}u_p$ and ${}^{i}u_p$ of the point P result then in accordance with the equations (1):

$$ \begin{pmatrix} {}^{i}u_p \\ {}^{i}v_p \\ 1 \end{pmatrix} = \underbrace{\begin{pmatrix} f_x & 0 & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{pmatrix}}_{\text{camera matrix}} \cdot \begin{pmatrix} \frac{{}^{c}X_p}{{}^{c}Z_p} \\ \frac{{}^{c}Y_p}{{}^{c}Z_p} \\ 1 \end{pmatrix} \tag{2} $$

$$ \text{with } f_x = s_x \times f \text{ and } f_y = s_y \times f $$

$$ \begin{pmatrix} \frac{CP_p}{CP_p} \\ \frac{{}^{c}Y_p}{{}^{c}Z_p} \\ 1 \end{pmatrix} = \begin{pmatrix} f_x & 0 & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{pmatrix}^{-1} \cdot \begin{pmatrix} {}^{B}u_p \\ {}^{B}v_p \\ 1 \end{pmatrix} = \begin{pmatrix} a \\ b \\ 1 \end{pmatrix} \tag{3} $$

The identification of the parameters listed here can be done using the Python library OpenCV and a chessboard pattern. Using several images of the chessboard pattern from different poses, the camera matrix can be calculated. In addition to this, the transformation between camera coordinate system C and chessboard coordinate system CB is obtained as a result of the calibration.

$$ \begin{pmatrix} {}^{c}X_p \\ {}^{c}Y_p \\ {}^{c}Z_p \\ 1 \end{pmatrix} = \begin{pmatrix} d_{11} & d_{12} & d_{13} & t_x \\ d_{21} & d_{22} & d_{23} & t_y \\ d_{31} & d_{32} & d_{33} & t_z \\ 0 & 0 & 0 & 1 \end{pmatrix} \cdot \begin{pmatrix} {}^{CB}X_p \\ {}^{CB}Y_p \\ 0 \\ 1 \end{pmatrix} \tag{4} $$

B. Camera-Flange calibration

In order to make the data from the camera calibration described above available for the robot, the camera must be calibrated with respect to the robot flange. For this purpose, a method developed at ZeMA [11] [12] is used. The prerequisite for this calculation is a camera calibration that has already been carried out with the camera mounted on the flange. The extrinsic parameters, i.e., the transformation from the chessboard coordinate system CB to the camera coordinate

system C, are used for different positions and orientations of the camera above a stationary checkerboard pattern. Matching the camera position over the chessboard pattern, the respective poses of the robot flange with respect to the robot base coordinate system B have to be acquired. With this data, a system of equations of the form $\boldsymbol{AX = XB}$ can be set up, where $\boldsymbol{X}$ represents the transformation matrix ${}^{F}\boldsymbol{T}_{c}$ that must be identified.

C. Introduction to AI

Artificial Intelligence (AI) leverages computer systems and machines to imitate the problem-fixing and decision-making abilities of the human mind. Machine Learning (ML) is a branch of AI where the use and development of computer systems that are able to learn and adapt without following explicit instructions, by using algorithms and statistical models to analyze and draw inferences from patterns in data. Deep learning is a subset of machine learning, which is essentially a neural network multiple layer. These neural networks attempt to simulate the behavior of the human brain (but far from matching human brain's ability) allowing it to "learn" from large amounts of data.

D. Convolutional Neural Network[CNN]

One of the most popular Deep Neural Networks for object detection is the Convolutional Neural Network (CNN) [9]. It takes this name from the mathematical linear operation between matrices called convolution. CNN's efficiency lies in decreasing the Artificial Neural Network Parameters and its ability to handle a huge amount of data. Depending on this reasoning, researchers have used it in various spheres like Computer Vision, Natural Language Processing, Pattern Recognition, and other instances. First, we will go through the base of CNN [2].

1) Convolution Layer: In the case of Image processing, usually, we get color images with $height \times width \times color$ shape. For instance, fig. 2 has $4 \times 4 \times 1$ image as input examples. But in real-life scenarios images can be enormous. For instance, with an 8k image, the size can be $7680 \times 4320 \times 3$. Hence, reducing these huge sizes is essential. The convolutional layer reduces the images into a form that is easier to process, without losing features that are critical for getting a good prediction. In the beginning, the hidden neurons[1] of the next layer only get inputs from the corresponding part of the previous layer (output of the previous layer) rather than the full connectivity where the connection is extremely high. Another simplification way is to keep weight constant throughout all neurons which makes sure that the next layer connection remains smooth. By doing this, a lot of weight update is controlled. With these two assumptions, it provides an opportunity to detect and recognize features regardless of their positions in the image. These weights are given as matrices to find specific features from the image. These

[1]The word "hidden" implies that they are not visible to the external systems and are "private" to the neural network.

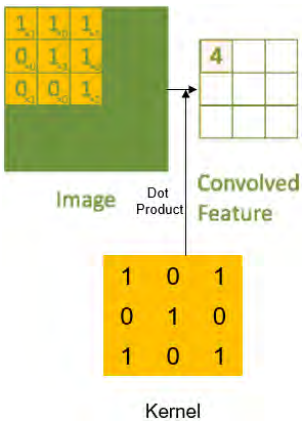


Fig. 2: Convolution Layer

matrices are also called a filter because they act like the classic filters in the image processing. However, these filters are initialized in the CNN, followed by the training procedure shape filters, which are more suitable for the given task. These filters strides through the whole image. Controlling these strides can decrease the parameters more and more, and at the same time reduce some of the side effects. Usually, filters move through the image grid with one jump at a time. This is known as Stride. The assumption is that the successive layer's node overlaps with their neighbors by gazing at the regions. Stride manipulates this overlapping condition. If we provide a stride of two, the filter will jump two grid lines as a stride which will reduce the size. If we have an image $N \times N$ and $F \times F$ filter, then the output O would be:

$$O = \frac{2 + N - F}{S} \tag{5}$$

Where N = input size, F = filter size and S = stride. But convolution has a drawback. As the filter moves forward with strides, it gives more emphasis on the middle parts of an image than in the corner. But corners can have important information which we are giving less emphasis. We can use 0 padding to recoup this situation. 0 Padding means adding an extra row and column of the outside of an image to get through the corners more. But adding an extra part will decrease the reduction size of the image but it captures more information about the image [see fig. 2]. With 0 padding the output would be:

$$O = \frac{1 + N + 2P - F}{S} \tag{6}$$

Where P is padding.

2) Pooling Layer: The pooling layer is responsible for reducing the spatial size of the convolved feature. It reduces the computing power required to process the data by reducing the dimensions to effectively train the model. There are two types of pooling. Max pooling and Average pooling. Max Pooling returns the maximum value of the portion of the image covered by the kernel. Average Pooling, on the other hand, returns the average of all the values of the part of the image covered by the kernel. Max Pooling also acts as a noise

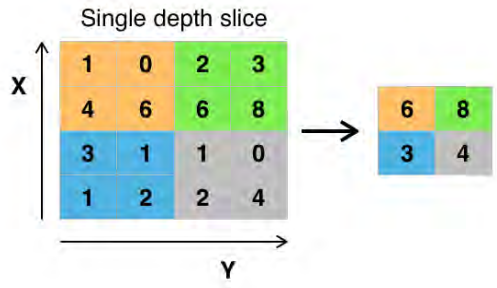


Fig. 3: Max Pooling

canceller. Completely discard noisy triggers and perform noise suppression along with dimensionality reduction [see fig. 3].

3) Fully Connected Layer: Fully Connected Layers form the last few layers in the network are just like feed forward network of ANN (see fig 4). In feed-forward network, every point is counted as feature. By verifying series of epochs, the model can differentiate between predominating and low-level features in images and classify applying the Softmax Classification technique.

$$softmax(x)_i = \frac{exp(x_i)}{\sum_j exp(x_j))} \tag{7}$$

The flattened output is fed to a feed-forward neural network and backpropagation applied to every iteration of training for validation for object detection decision. Output of pooling is flattened into a column vector so that we can feed it to feed-forward network. Each input, $g(Wx + b)$ is calculated for each of the hidden layer neuron (g = activation function, x = input, W = weight, b = bias). The weights are essentially reflecting how important an input is. Positive weights increase the output while negative decreases the output. The bias is used to shift the result of activation function towards the positive or negative side. Usually ReLU is used for the activation function, which is $ReLU(z) = \max(0, z)$. There can be n-number of hidden layers depending on the accuracy we need and computational power we have. For each neuron of the hidden layer, the weighted sum of inputs and weights is $Y = x_1w_1 + x_2w_2 + ... + x_nw_n + b$. After passing through all the hidden layer with same calculation, each class will get a predicted value using softmax function from each of the neuron. Then each output's error is calculated.

$$Error = actual\ value - predicted\ value \tag{8}$$

The goal is to minimize this error for same class and increase for different classes, this error minimization is done by Stochastic gradient descent which is derivative of loss in respect to weight.

$$w_{new} = w_{old} \ - \ \eta \sum_{i=1}^{n} \nabla \ loss\,(w) \tag{9}$$

Where, $\eta = \frac{learning\ rate}{step\ size}$. Higher the η, more it learns. But with higher η, there is risk of over fitting and lower η gives underfitting. So, we need provide a moderate learning rate. Each weight is updated to minimize the error for same class and increase for different classes. Thus, each class gets a value of training. While detection, same procedure is followed to decide the new object class which has least error with respect to class values. Fig 5 represents full CNN structure.

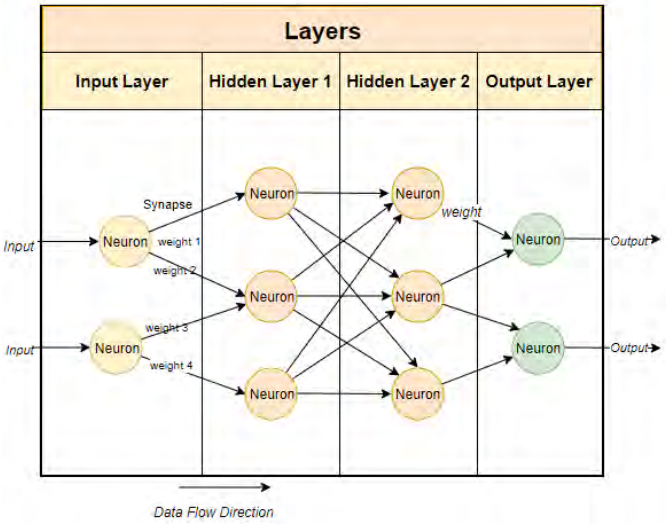


Fig. 4: Fully Connected Layers

4) YOLO: The abbreviation YOLO [15] [17] stands for the expression 'You Only Look Once'. It is an algorithm to detects and recognizes objects in images or video feeds. YOLO considers object detection as a regression problem and renders the class probabilities of the detected images. YOLO is based on CNN. It requires only a single forward propagation through CNN to detect objects. It yields that prediction in the entire image is done in a single algorithm run. CNN permits to prediction of different class probabilities and bounding boxes simultaneously. In this section, we will describe the YOLO in general. We have implemented YOLO version 3 over the YOLO version 2 which is described in section IV.

YOLO is crucial due to the fact of:

- Speed: This algorithm improves the rate of detection due to the fact it can predict the object in real-time.
- High Accuracy: YOLO is a predictive method that gives correct outcomes with minimum background errors.
- Learning Capability: It has amazing insight-gaining abilities that allow it to analyze the representations of objects and employ them in object detection.

YOLO works in 3 steps [17]:

- Residual blocks: In the beginning, an image is divided into $S \times S$ grids (see fig 6). In the grid, there are numerous grid cells of identical dimensions. Every grid unit will

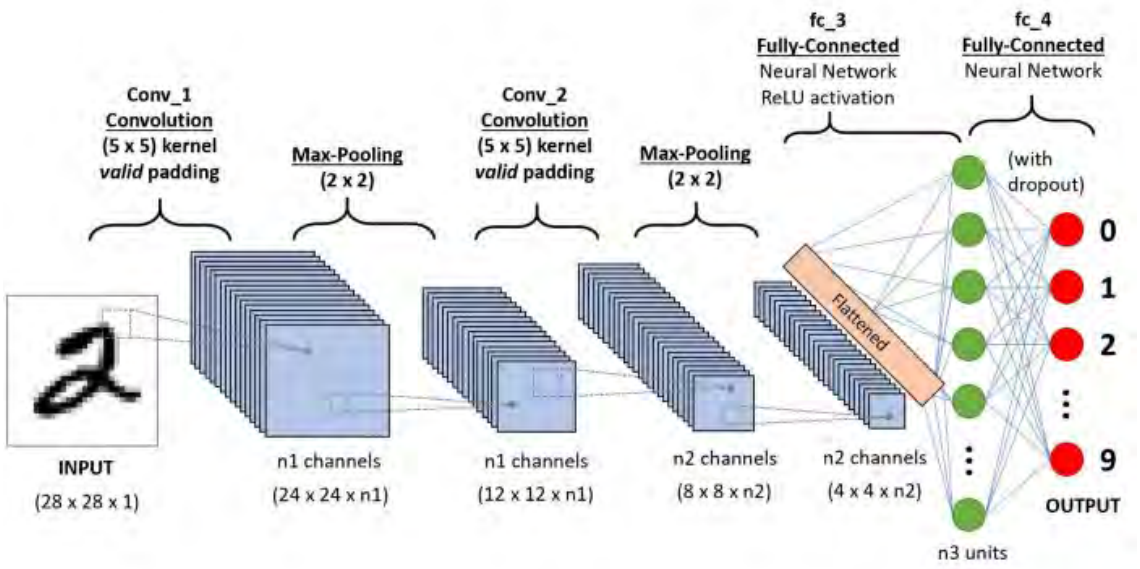


Fig. 5: Convolutional Neural Network[CNN]

locate items that seem inside of it. For instance, if an item middle seems inside a sure grid unit, then this unit could be liable for detecting it.

- Bounding box regression: A bounding box is an outline to highlight an object in an image. Each bounding box have width (b_w), height (b_h) and class (e.g. person, car, traffic light, etc.) where c = Classes and box center (b_x,b_y) [see fig 7]. The yellow outline is the bounding box. YOLO algorithm utilizes a single bounding box regression to predict the height, width, center, and class.
- Intersection Over Union (IOU): IOU is the description of how the boxes overlap. YOLO employs IOU to render a yield box that encircles the objects ideally. Each grid unit is accountable for divining the bounding boxes and their confidence scores. The IOU is equal to 1 if the predicted bounding box is the same as the real box. This mechanism eliminates bounding boxes that are not equal to the real box. In fig 8, there are two bounding boxes, the blue box which is the predicted box while the green box is the real box. YOLO ensures that the two bounding boxes are equal.

Fig. 6: Residual Blocks

IV. Experiment

To validate the concept, we implemented the object detection for demonstrative use case with bolts, nuts and washers. They are randomly placed on a flat surface with black background in the workspace of the used Kuka Kr6 robot (see fig. 9). A two finger gripper (OnRobot RG2 gripper [13]) and a 3d camera (Intel Realsense D435) are attached to the robot flange. The target is to detect and grip these objects and drop them at a predefined place.

A. Model Training

For the experiment, we have captured 365 photos of our object assembled nuts, unassembled nuts, bolts, and washers. Each picture can have multiple items from upper mentioned. We have used LabelImg [10] to label our images into two

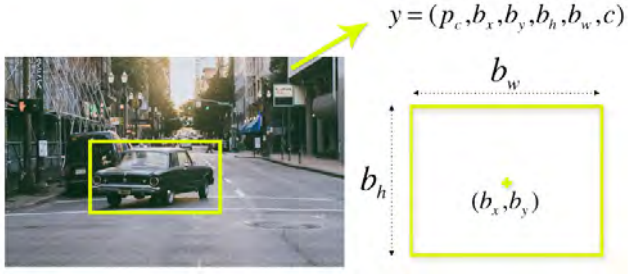


Fig. 7: Bounding Box

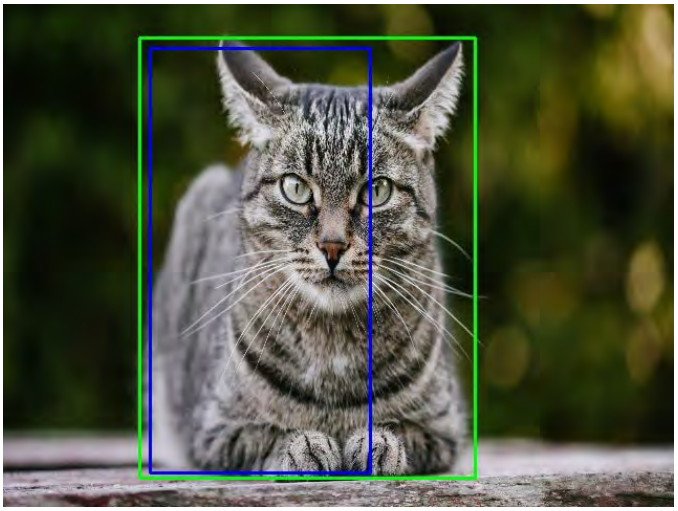


Fig. 8: Intersection Over Union (IOU)

classes. All properly assembled nuts, bolts and washers together are assigned to the "Assembled" class and all that are not properly connected are assigned as "Not assembled". Thus, we have around 700+ labels. Moreover, we also changed the shape of the images (rotate, flip, but do not resize) and were able to create some additional images.

For model training and object detection we have implemented YOLOv3 [16]. In comparison, YOLOv2 [19] has darknet-19 which is a custom deep architecture of a 19-layer network extended with 11 layers. YOLOv2 often grapples to detect diminutive objects because it downsampled the input which attribute to the loss of the fine-grained features. YOLOv2 uses identity mapping, concatenating feature maps from a previous layer to capture low-level features to coup up with the difficulty. However, YOLOv2's architecture has obstacles like no residual blocks, no skip connections and no upsampling. YOLOv3 solves these obstacles. YOLOv3 has 53-layer network trained on Imagenet [5]. For detection, YOLOv3 has 53 layers stacked up. Overall YOLOv3 has 106 layers of convolutional architecture. But v3 is slower than v2 as it boasts of residual skip connections, and upsampling. The standout feature of v3 is that it makes detections at three different scales. YOLO is a fully convolutional network, and its eventual output is generated by applying a 1×1 kernel on a feature map. [2]. In YOLOv3, 1 x 1 detection kernels are applied on feature maps of three different sizes at three different places in the network. The shape of the detection kernel is $1 \times 1 \times (B \times (5 + C))$. Here B is the number of bounding boxes a cell on the feature map can predict and C is the number of classes. YOLOv3 trained on COCO dataset as pre-trained model.

Fig. 9: System Setup

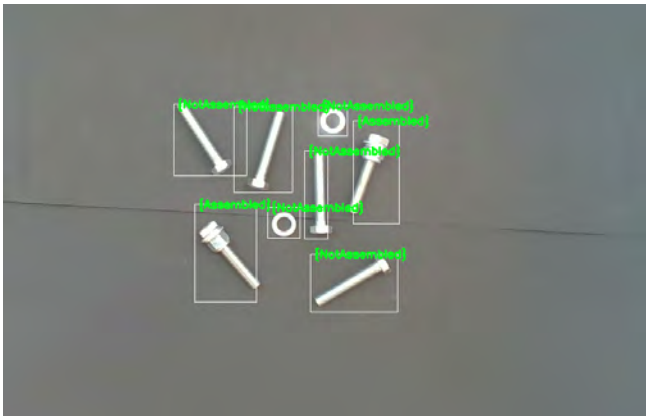


Fig. 10: Detected Objects

After detecting the object [Figure 10], we crop the detected part from the video feed which is "assembled" and apply the OpenCV contour function to find the object area. It is important to know that we are only interested in detected object and we need to find out a proper gripping point for the object to grip and change it's location. Applying OpenCV contour appears with it's challenges because before applying contour, we are required to crop the detected image and transformed it into a grayscale image. In this grayscale image, we solicit contour detection to get the definitive (only) object area. But grayscale transformation depends distinctly on the proper lighting condition. Thus, contour is variable with the lighting setup. With the contour from the object, we can perceive the contour center which is the center of the density of the contour. But we discovered that the contour center is not a gripping position as the assembled object has more density towards its nut-bolt connection position which is not up to the mark for gripping. So, we adopted the highest distanced points from the contour center point to create reference points distanced from the contour center point to produce a straight line using these reference points. Meanwhile, this straight line served to shift the center point along towards the bottom of the assembled object, so the gripper gets a proper gripping

[2]Feature map is finding edges for each smallest grid where currently the filter is going through.

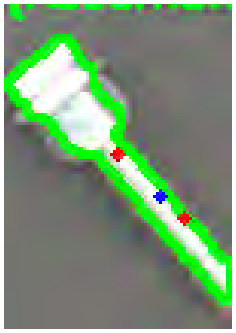


Fig. 11: Calculated gripping point (blue) for the assembled object. Upper left [red] is contour center and also one of the reference point and down right [red] point is another reference point which is adopted the highest distanced points from the contour center point.

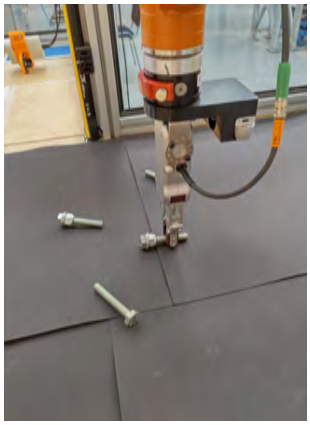


Fig. 12: Object Gripping.

point [Figure 11].

B. Path Planning

To finally pick up the assembled screws the gripping point P, identified in the image (see fig. 11), has to be transferred to the robot. Furthermore, the gripping position has to be extended by an orientation. The transformation from image coordinates system i to the robot base coordinate system B can be accomplished using the parameters of the camera calibration and the camera flange calibration. In the first step, converting the gripping position from pixel coordinates to the camera coordinate system C is calculated with aid of the equations (3) and (4). Merging the two equations yields the following system of equations from which the ${}^{C}Z_p$-value for the sought pixel can be extracted.

$$\begin{pmatrix} {}^{CB}X_p \\ {}^{CB}Y_p \\ {}^{C}Z_p \end{pmatrix} = \begin{pmatrix} -d_{11} & -d_{12} & a \\ -d_{21} & -d_{22} & b \\ -d_{31} & -d_{32} & 1 \end{pmatrix}^{-1} \cdot \begin{pmatrix} t_x \\ t_y \\ t_z \end{pmatrix} \tag{10}$$

Substitution of ${}^{C}Z_p$ into equation e1 then also provides the corresponding ${}^{C}X_p$ and ${}^{C}Y_p$ coordinates of the gripping point regarding the camera coordinate system C. For the following calculation these coordinates are extended with a one to homogeneous coordinates of the form

$${}^{C}\underline{r}_p = \begin{bmatrix} {}^{C}\underline{r}_p \\ 1 \end{bmatrix} = ({}^{C}X_p, {}^{C}Y_p, {}^{C}Z_p, 1)^T \tag{11}$$

The now missing transformation to the base coordinate system B can be expressed as a chain of transformations consisting of the transformation matrix ${}^{F}\boldsymbol{T}_C$ from camera coordinate system C to the flange coordinate system F and the transformation matrix ${}^{B}\boldsymbol{T}_F$ derived from the flange pose during image acquisition.

$${}^{B}\underline{r}_p = \begin{pmatrix} {}^{B}\underline{r}_p \\ 1 \end{pmatrix} = {}^{B}\boldsymbol{T}_F \cdot {}^{F}\boldsymbol{T}_C \cdot \begin{pmatrix} {}^{C}\underline{r}_p \\ 1 \end{pmatrix} \tag{12}$$

In this equation ${}^{B}\underline{r}_p$ describes the displacement vector of the gripping point P regarding the robot base coordinate system B. For a suitable orientation to gripping position the estimation is done, that an alignment of the gripper parallel to the normal vector of the table meets the requirements of the gripping process. This normal vector is obtained from the chessboard coordinate system CB. A second axis to determine the orientation is then given by the straight line, calculated to shift the gripping point towards the bottom of the screw (section IV.A). Two points P_{l1} and P_{l2} on this straight line inserted into equation (9) provide the position of the straight line regarding the chessboard coordinate system. The orientation of the screw with respect to the chessboard coordinate system can now be described as a simple rotation around the z-axis of CB. The angle of rotation is given by

$$\alpha = atan\left(\frac{y_{l2} - y_{l1}}{x_{l2} - x_{l1}}\right) \tag{13}$$

The resulting rotation matrix is then:

$${}^{CB}\boldsymbol{D}_p = \begin{pmatrix} \cos\alpha & -\sin\alpha & 0 \\ \sin\alpha & \cos\alpha & 0 \\ 0 & 0 & 1 \end{pmatrix} \tag{14}$$

The final orientation of the screw with respect to the base coordinate system can be composed of several rotation matrices similar to the position transformation.

$${}^{B}\boldsymbol{D}_p = {}^{B}\boldsymbol{D}_F \cdot {}^{F}\boldsymbol{D}_C \cdot {}^{C}\boldsymbol{D}_{CB} \cdot {}^{CB}\boldsymbol{D}_p \tag{15}$$

Where ${}^{B}\boldsymbol{D}_F$ and ${}^{F}\boldsymbol{D}_C$ can be extracted from the transformation matrices ${}^{B}\boldsymbol{T}_F$ and ${}^{F}\boldsymbol{T}_C$. Furthermore, ${}^{C}\boldsymbol{D}_S$ originates from the transformation matrix ${}^{C}\boldsymbol{T}_{CB}$, which was calculated in equation (4) during camera calibration. Finally the rotation matrix ${}^{B}\boldsymbol{D}_p$ has to be expressed in the Roll-Pitch-Yaw convention [3], to provide 6D pose vector of the form $\underline{w}_p = (x_p, y_p, z_p, a_p.b_p, c_p)^T$ to the robot. Here a_P, b_P and c_P describe the three euler angles. An example of the gripping pose that occurs can be seen in fig. 12.

V. Evaluation

With the specified training method and parameters, we have achieved more than 90% accuracy for the object detection on our specified class of “Assembled” and “Not Assembled”.

Moreover, for the contour detection we have obtained exceedingly satisfactory results. Nevertheless, finding the contour could be influenced by changing lighting conditions. For industrial implementation this stipulation should be taken into account. Another issue is the contour detection with different background colors. This is why we installed a black background to generate a high contrast to the objects. After acquiring an acceptable gripping position described in section IV.B, we transformed the gripping position in the robot base coordinate system. In this transformation, we have around 4-6mm accuracy error depending on the camera calibration and camera-flange calibration error. This is sufficient for our evaluation process of gripping the "assembled" bolts. Furthermore, our use case does not contain high requirements for the drop off position, thus minor errors don't influence our outcome.

VI. Discussion & Future Work

As we described in evaluation, we have achieved satisfactory results. For instance, we were able to minimize the cost with a less expensive camera. Our object detection is also quite economical. But as we are using contour, we required to set the object background black to differentiate between our object (which is bright) and background. To solve this problem, one future approach could be better training of our model with OpenCV Rectangle function as we can manipulate the image cropped from the video feed. Towards the bin picking, the objection detection should also be able to handle stacked bolts or more general stacked objects. Moreover, the localization is performed in a 2D scenario and thereby restricted to pre-calibrated planes. Our future work will look towards the use of a 3D camera to solve the upper mentioned problem. Coordinate transformation will then be possible for any position and orientation of the objects. Another focus point could be the improvement of the calibration errors and therefore make it feasible to implement for higher precision applications.

References

[1] OpenCV: **https://opencv.org/**.

[2] S. Albawi, T. A. Mohammed, and S. Al-Zawi. Understanding of a convolutional neural network. In *2017 International Conference on Engineering and Technology (ICET)*, pages 1–6. Ieee, 2017.

[3] M. H. Ang and V. D. Tourassis. Singularities of euler and roll-pitch-yaw representations. *IEEE Transactions on Aerospace and Electronic Systems*, (3):317–324, 1987.

[4] J. Coyle, E. Bardi, and C. Langley. *The Management of Business Logistics*. West Publishing Company, 1988.

[5] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei. Imagenet: A large-scale hierarchical image database. In *2009 IEEE Conference on Computer Vision and Pattern Recognition*, pages 248–255, 2009.

[6] D. Holz, M. Nieuwenhuisen, D. Droeschel, J. Stückler, A. Berner, J. Li, R. Klein, and S. Behnke. Active recognition and manipulation for mobile robot bin picking. In *Gearing Up and Accelerating Cross-fertilization between Academic and Industrial Robotics Research in Europe:*, pages 133–153. Springer, 2014.

[7] A. Kaehler and G. Bradski. *Learning OpenCV 3: computer vision in C++ with the OpenCV library*. " O'Reilly Media, Inc.", 2016.

[8] D. Kraft, L.-P. Ellekilde, and J. A. Jørgensen. Automatic grasp generation and improvement for industrial bin-picking. In *Gearing Up and Accelerating Cross-fertilization between Academic and Industrial Robotics Research in Europe:*, pages 155–176. Springer, 2014.

[9] Y. LeCun, B. Boser, J. S. Denker, D. Henderson, R. E. Howard, W. Hubbard, and L. D. Jackel. Backpropagation applied to handwritten zip code recognition. *Neural computation*, 1(4):541–551, 1989.

[10] T. Lin. Labelimg, 2015.

[11] R. Müller, M. Vette-Steinkamp, and A. Kanso. Position and orientation calibration of a 2d laser line sensor using closed-form least-squares solution. *IFAC-PapersOnLine*, 52(13):689–694, 2019.

[12] R. Müller, M. Vette-Steinkamp, T. Masiak, and A. Kanso. Best-fit method for the calibration of 3d objects using a laser line sensor mounted on the flange of an articulated robot. In *Tagungsband des 4. Kongresses Montage Handhabung Industrieroboter*, pages 207–216. Springer Vieweg, Berlin, Heidelberg, 2019.

[13] Onrobot. Onrobot RG2 Gripper.

[14] C. G. Petersen and G. Aase. A comparison of picking, storage, and routing policies in manual order picking. *International Journal of Production Economics*, 92(1):11–19, 2004.

[15] J. Redmon, S. Divvala, R. Girshick, and A. Farhadi. You only look once: Unified, real-time object detection. In *Proceedings of the IEEE conference on computer vision and pattern recognition*, pages 779–788, 2016.

[16] J. Redmon and A. Farhadi. Yolov3: An incremental improvement. *arXiv preprint arXiv:1804.02767*, 2018.

[17] Section.io. Introduction to YOLO Algorithm for Object Detection.

[18] A. Zeng, S. Song, K.-T. Yu, E. Donlon, F. R. Hogan, M. Bauza, D. Ma, O. Taylor, M. Liu, E. Romo, et al. Robotic pick-and-place of novel objects in clutter with multi-affordance grasping and cross-domain image matching. In *2018 IEEE international conference on robotics and automation (ICRA)*, pages 3750–3757. IEEE, 2018.

[19] J. Zhang, M. Huang, X. Jin, and X. Li. A real-time chinese traffic sign detection algorithm based on modified yolov2. *Algorithms*, 10(4):127, 2017.

Robotic stripping with external position/force control applied to the Yumi collaborative robot

Meryem Taghbalout
University of Lorraine, Art et Métier Paris Tech, LCFC
Metz, France
meryem.taghbalout@gmail.com

Jules Audrey MAMA TSANGA
University of Lorraine, Art et Métier Paris Tech, LCFC
Metz, France
jules-audrey.mama-tsanga9@etu.univ-lorraine.fr

Jean-François Antoine
University of Lorraine, Art et Métier Paris Tech, LCFC
Metz, France
jean-francois.antoine@univ-lorraine.fr

Gabriel Abba
University of Lorraine, Art et Métier Paris Tech, LCFC
Metz, France
gabriel.abba@univ-lorraine.fr

***Abstract*— In this paper, we propose to robotize wire and cable stripping using an external position/force control developed on the ABB IRB14000 robot called "YuMi". This command aims at controlling the external force applied on the robot and consequently the one applied on the wire to be stripped, which allows both to improve the robot behavior and to robotize an application currently not robotized. The first results of robotic stripping are presented.**

Keywords— Robot control, External position/force control, Robotic stripping, YuMi

I. INTRODUCTION

The stripping of electrical cables is a widespread process in the world and its importance extends to many areas. It is used for example in the industrial world, in the home for the connection of electrical installations, and in crimping for the preparation of cables. The concept of stripping is similar to that of degaussing in some areas where cables are stripped to recover the copper conductors. Degaussing can also be considered as the stripping of cables with a cross-sectional area greater than 16 mm^2 [1]. This technique was introduced during the second industrial revolution, which began in 1870. Indeed, the use of electricity in industrial production will make men rush to invent electrical machines [2]. Thus, from then on, in order to make the connections between machines, operators would need to strip the electrical cables.

Stripping an electrical cable means removing the protective layer called "insulation" from the conductive layer over a certain length of cable. It also exposes the core of the cable. Over time and with a growing market, the demand for innovative stripping techniques has continued to increase [3]. A number of companies have specialized in this process, offering ever more innovative techniques. A large number of patents support this. Nowadays, there are many special tools and machines created as a result of the various industrial revolutions.

A. What are the properties of the stripping process?

Stripping techniques and tools have continued to develop and evolve over the years, to the point where there are now a large number of them. To strip an electrical wire, a certain number of parameters must be taken into account for the stripping to be done correctly. According to the NF EN 2812 standard [4], stripping is correctly performed when:

- The insulation or shielding does not show any signs of deterioration that could cause short circuits in electrical installations
- The conductive core does not show any cracks or alterations that could cause the wire to break and cause an electric shock
- The original performance of the wire or cable as defined in the technical specifications is not reduced after the stripping operation.

In addition, this standard also identifies defects that may appear after the stripping operation [4] [5]. These defects include:

- A lack of torsion and discarding of the core
- Marks on the insulation (nick, burn, visible wire, etc.)
- Residual insulation
- Cut wires or strands
- Surface contamination (burnt insulation, glue residue, dust, etc.)

B. Stripping concepts and techniques

To strip an electrical cable, several techniques can be distinguished. Four main stripping methods are presented in the NF EN 2812 standard. They are based on mechanical tools made according to different and varied technologies. For this purpose, there are:

- Manual stripping: this practice is performed with a scalpel, razor blade or any other sharp object (cutting pliers, scissors, etc.). This practice is most often used when users do not have the proper tools to perform the stripping. This leads in the vast majority of cases to numerous stripping defects and for applications such as aeronautics; the use of manual cable stripping is prohibited when another stripping process can be used.
- Mechanical stripping: This method of stripping is the most common and most used by professionals to strip individual wires. It includes many tools of different strengths adapted to given cable sections. The special feature of these tools is their simplicity, which makes them ideal for use in a wide range of applications.
- Thermal stripping: it encompasses all devices powered by a voltage and composed of two parts: a voltage generator to control the temperature and a manual part consisting of two wires with clamps at their end. The two wires are powered by a current and heated to a temperature between 0° and 99°. The stripping is then done by pressing the clamps on the cable. The advantage of this device is that it does not require very strong pressure on the electrical cable. On the other hand, its design remains complex.
- Laser stripping: Laser is a process used by professionals for its precision, repeatability and robustness [6]. It is also used for applications where the presence of the slightest defect on the conductive core is not acceptable and when the external geometry of the cable to be stripped is incompatible with a mechanical process.

Mechanical stripping is the most widespread technique in the industrial world today. In the context of the project, it is the easiest to adapt. Moreover, in terms of cost, it is the one with the least financial requirements. We have therefore chosen in our project the mechanical stripping technologies for their feasibility and ease of implementation on the YuMi robot (Fig. 1) and also to allow us to validate the external control law we seek to apply on the robot.

Fig. 1 Robot IRB14000 from ABB (in LCFC lab)

The first part of this paper presents a state of the art on wire stripping, its properties and existing methods on the market to strip a wire. In the second part, we briefly present the control law developed on the robot to robotize the wire stripping application. In the third part, we present the technique adopted for this purpose. In the fourth part, we exploit the experiments carried out to test and validate this stripping. Finally, in the last part, the results obtained and the perspectives of this study are discussed.

II. CONTROL LAW

The external control of the position/force is implemented by using the dynamic model of the robot without using external sensors. This consists in organizing beforehand the force control loop on the position control loop and not in organizing these two loops simultaneously like the classical hybrid control [8]. Fig. 2 shows the schematic diagram for a single arm of YuMi.

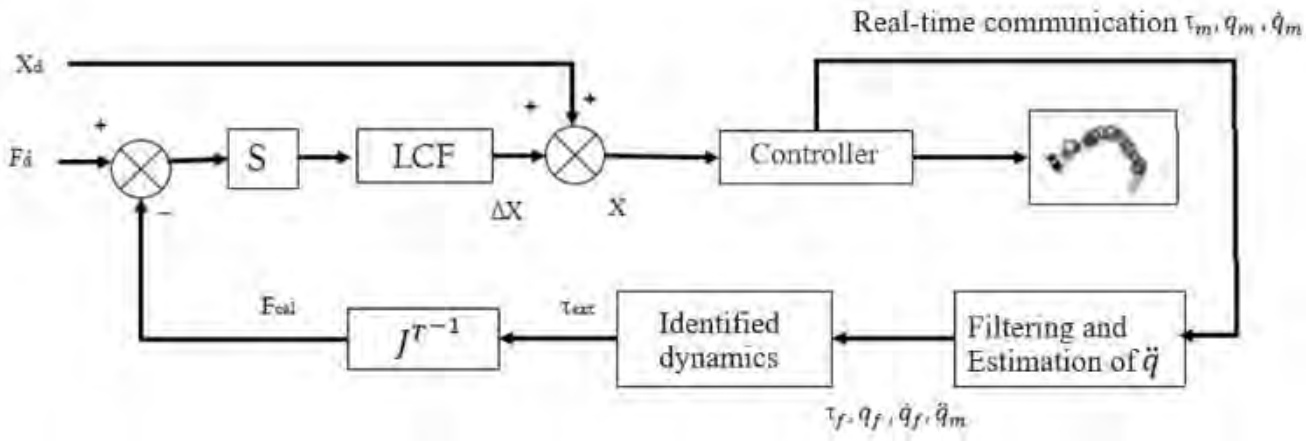


Fig. 2 Proposed control principle for a single YuMi arm

The desired external forces $\mathbf{F_d}$ and the calculated forces $\mathbf{F_{cal}}$ applied to the robot end effector are used to calculate a position deviation **X** and then added to the desired position set point $\mathbf{X_d}$ to form the new position set point **X** transmitted to the robot. Thanks to this, a constant desired force is imposed and then, with the selection matrix S, a diagonal matrix composed of 1 or 0 is chosen, the directions to be controlled in force; if this one is equal to 1, the servo-control of this direction is carried out. The position variation $\Delta\mathbf{X}$ resulting from the difference ($\mathbf{F_d}$ - $\mathbf{F_{cal}}$) is obtained from a force control law **LCF** to be defined according to the task to be performed and the desired results. Indeed, contrary to the hybrid control, the force constraints do not act directly on the joint torque of the actuators but on the operational position of the robot. A great advantage of this control is that it can be applied to relatively "closed" controller architectures, as in the case of our YuMi robot, which already has its own position controller and which is not suitable for implementing another position control either.

This allows us to take advantage of the robot controller as well as the new external control. The complexity of this law can thus be particularly high depending on the identified dynamic model of the robot; see our detailed work on this part in reference [9]. It also depends on the filtering of the data and the frequency of the measurements, i.e. the real-time communication with the robot; also refer to our detailed work on this topic in reference [10].

III. STRIPPING TECHNIQUE ADOPTED

In order to strip an electrical wire, it is necessary to perform an adequate technique beforehand in order to respect certain constraints. For example, it is important not to damage the conductive core, which could then break during handling and cause defects, and not to damage the protective sheath on the remaining length of the cable to avoid causing short circuits. This being the case, the stripping is spread over several steps [7], namely:

- Step 1: Setting the gauge, this is the first step of the stripping operation, which consists in adapting the tool or the machine to the wire to be stripped. The important characteristic to take into account here is most often the wire section.
- Step 2: Wire clamping, this step is relatively simple as it consists of placing the wire between the clamps of the stripping tool. During this step, it is also important to ensure that the clamping force does not damage the protective cover
- Step 3: Application of force to segment the insulation, this is an important step, as it is here that all parameters must be taken into account. The space left between the tool's clamps must be sufficient and precise so that the cut does not damage the conductive core. In addition, the segmentation obtained must be sufficient to facilitate the removal of the insulation in the next step.
- Step 4: Removal of the protective sheath is the last step of the stripping operation, which consists in removing the insulation from a given portion of wire. Automatic stripping tools and industrial machines allow to remove the insulation on the concerned part directly after the segmentation in a single action. For manual tools, it must be done by pulling on the wire or by separating the wire from its sheath (stripping).

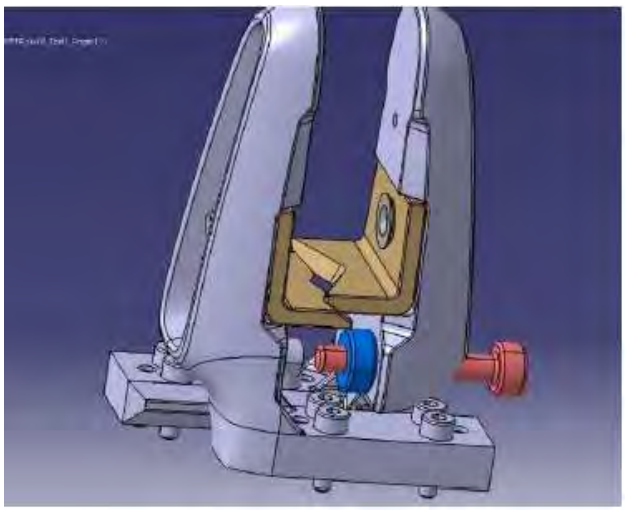


Fig. 3 Proposed CAD model of the clamp with identical V-shaped blades V

As part of our project, we have designed a tooling for the YuMi robot to strip electrical cables using a suitable technology that takes into account the constraints encountered in the execution of the stripping by the robot and that respects the required technical specifications. It consists of a pair of pliers with identical V-shaped blades (Fig. 3), a solution inspired by the classic stripping pliers that have V-shaped jaws and a locking screw to adjust the section of the cable to be cut.

A groove is made in each finger of the YuMi to accommodate the base of each blade. This groove has a radius r = 0.8mm which is twice the minimum value imposed by 3D printing. A smooth hole is also drilled in each finger and in each blade. A FHC M3 x8 screw and a M3 nut ensure the holding in position. This type of screw was preferred for space reasons because this screw can be completely embedded in the blade without causing problems to the overall design.

On each finger, roundings of $r \geq 0.4$ mm have been created on all the ribs in order to support the forces. An M3 x20 knurled screw and an M3 knurled nut can also be seen on one finger while on the other, an extension was created in order to reduce the length of the knurled screw used and to comply with the standards for knurled screws. The hole created, which rests on a vertex of the finger for stability and bearing surface, has a thread of pitch p = 0.25 mm and diameter d = 2.5 mm. The knurled locking system is explained here by the choice of manual adjustment, which has been taken into account.

The blades are made by folding a sheet of characteristics L × l × e = 25 × 20 × 2 mm. A chamfer was created on the base of the blade to facilitate its retention in position in the finger groove. A 120° chamfered notch was also made in the blade to create the cutting portion. The two blades were designed to be mounted back to back. With this in mind, chamfers of length 1 mm × 60° were added on the ends of the blades to provide a backlash free overlap of the blades and their adjustment into a self-contained position. This backlash-free overlap is useful because it enhances the cutting operation of the blades. Fig. 4 shows the proposed solution mounted on the robot.

Fig. 4 Proposed clamp mounted on the robot

IV. EXPERIMENTATION

Before presenting the first results of stripping with the control law, it is worth mentioning that we tried, with the same experiments, to strip wire only with the robot control. This means with a simple RAPID program that commands the robot to go from one position to another holding the cable between the clamps, but it turned out that the robot was not able to strip and it stopped because it considered itself to be in collision and, therefore, it deactivates the motors and displays the message "Motion Monitoring". In addition, we also noted that the robot was able to sustain much more than the external force on its arms, up to 40 N in a single direction, which confirms that the performance of a robot can be significantly improved by adding additional controls.

We now turn to the first results obtained during the stripping process. These results are based on an experiment carried out with a 1 mm diameter wire that we stripped with the right arm of the robot and a clamping tool that allows to hold the wire without any risk of slipping. Contrary to what happens when we use the other arm since we have not yet developed a tool that allows to hold the wire and to strip it well. Moreover, by changing each time the clamping force and the traction force controlled by the control law developed on the robot. We considered for these tests the same length to strip of 5 mm.

In the first result, we applied a maximum clamping force achievable by the robot of 20 N and commanded the robot to 30 N for the pulling force. It can be seen in Fig. 5 that the robot reached the desired force i.e. 30 N, and experimentally we found that the segmentation step was done at 15 s and the sheath removal takes place 3 s right after which explains the decrease in force from 15 s.

On the other hand, in the second case, we reduced the pulling force to 23N and the clamping force to 18 N because we noticed that the stripping could be done with less force and Fig. 6 proves it because the stripping is clearly better done than in the previous example. For Fig. 7 and Fig. 8, we have therefore maintained the same clamping forces and we have modified the tensile force to know its influence on the stripping. With a force of 35 N, the robot could not reach the force and it stripped before with a force of 30 N and for segmentation it needed a force between 15 N and 17 N to finalize the stripping operation.

In the last test, we decreased both forces. The stripping was perfectly performed with 20 N and 8 N of segmentation and clamping respectively, Fig. 9 shows the results. In this test, the stripping is performed in one-step with less force to be applied.

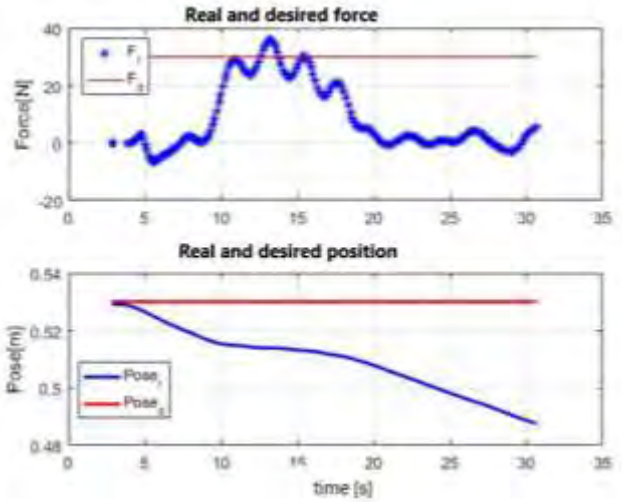


Fig. 5 Stripping results with 20 N clamping force and 30 N controlled force

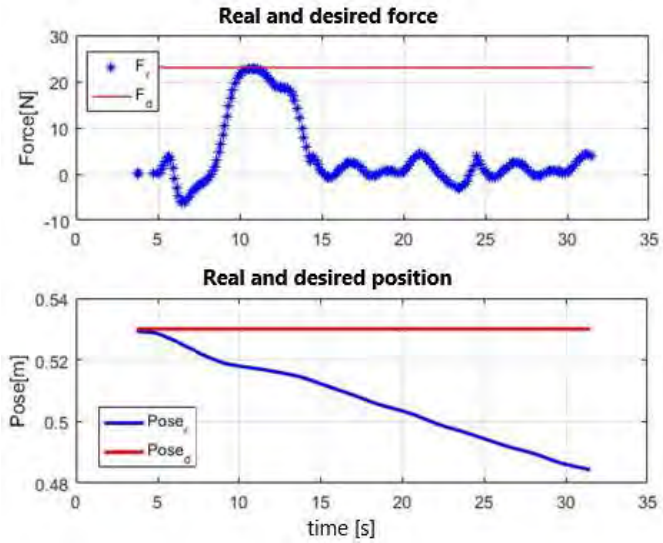


Fig. 6 Stripping results with 18 N clamping force and 25 N controlled force

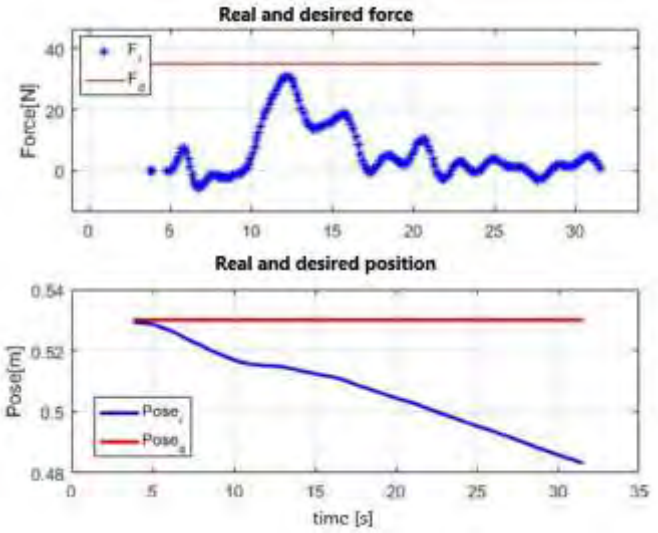


Fig. 7 Stripping results with clamping force 15 N and controlled force 35 N

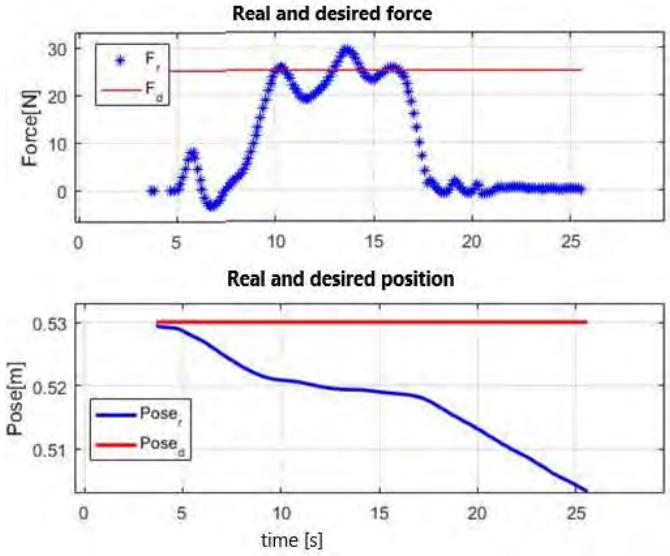


Fig. 8 Stripping results with 15 N clamping force and 25 N controlled force

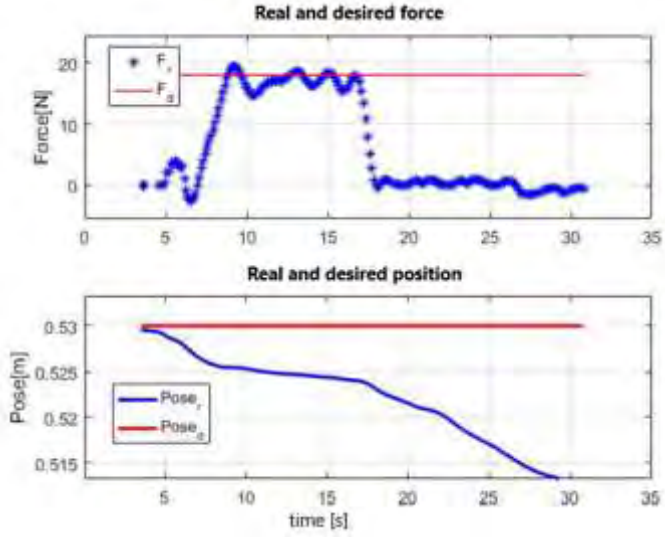


Fig. 9 Stripping results with 8 N clamping force and 20 N controlled force

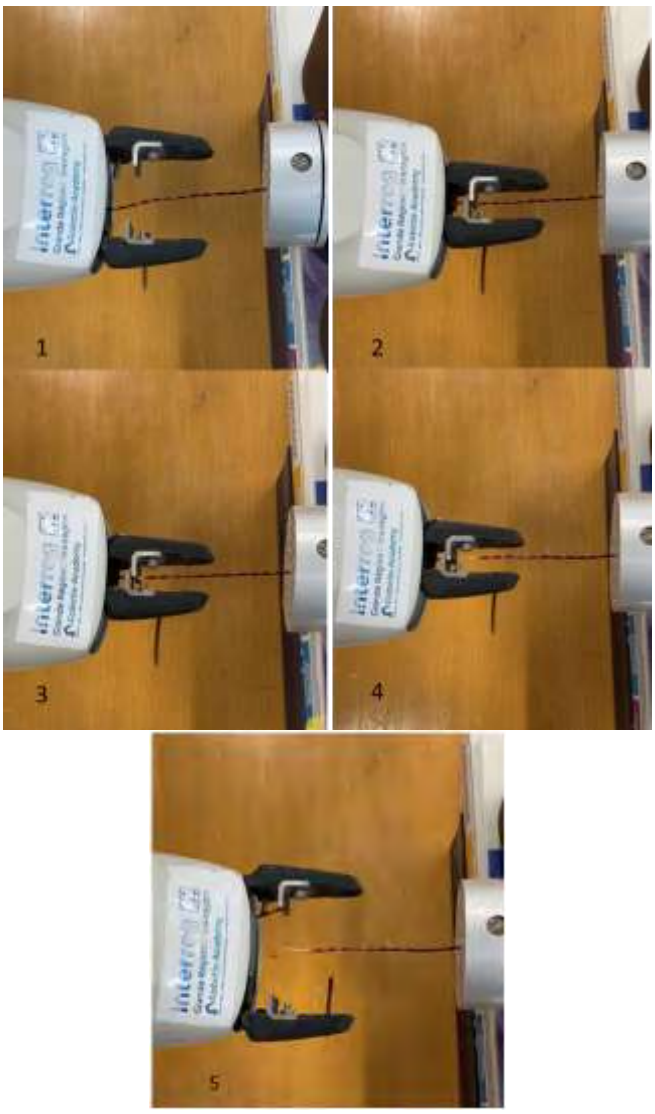


Fig. 10 Stripping steps with YuMi

V. CONCLUSION

The first results of robotic stripping with external position/force control with the robot confirm the applicability of robotic stripping and its steps are presented in Fig. 10; we have a defect-free stripped electrical cable respecting all the properties of manual stripping. These results also allow us to see that it is necessary to know in advance the force pattern to be applied for a specific length and diameter of the cable. We propose and take into consideration as a perspective of this work the identification of a force model from the experimental results allowing to know in advance the two forces to be applied on a cable to achieve a successful stripping respecting all the defined standards.

ACKNOWLEDGMENT

We would like to thank the Robotix Academy, contract number N°002-4-09-001 for funding this work as a part of the project funded by INTERREG V-A Grande Region program.

REFERENCES

[1] 2021.url:https://www.phoenixcontact.com/assets/downloads_ed/global/web_dwl_promotion/52006038_Werkzeug_fr-int_LoRes.pdf (visited in 2021).

[2] B. Segrestin. “Le tournant fayolien : des révolutions industrielles à la naissance de l’entreprise moderne”. In : Entreprises et histoire 2 (2016), p. 5-12.

[3] 2021. url : https : / / www . marketsandresearch . biz / report / 113071 / global -wire-stripping-machine-market-2020-by-manufacturers-regions-type-andapplication-forecast-to-2025 (visited in 2021).

[4] 2019. url : https://m.boutique.afnor.org/xml/1989134/false (visited in 2021).

[5] K. Stouffer, J. Falco and K. Scarfone. “Guide to industrial control systems (ICS) security”. In : NIST special publication 800.82 (2011), p. 16-16.

[6] 2021. url : https : / / www . lintech . fr / produits - et - solutions / cable - fil / denudage-laser/ (visited in 2021).

[7] B. Mercer.” Newnes industrial control wiring guide”. Routledge, 2001.

[8] M.H Raibert and J.J Craig. “Hybrid position/force control of manipulators”. In : (1981).

[9] M. Taghbalout, J.F. Antoine & G. Abba, (2019).” Experimental dynamic identification of a YuMi collaborative robot” . IFAC-PapersOnLine, 52(13), 1168-1173. MIM 2019, Germany.

[10] W. Yiguo M. Taghbalout, J.F. Antoine & G. Abba, (2020) “External communication of the robot ABB YuMi via virtual machine with analysis of hybrid position-force control” . RACIR 2020, Germany.

Evaluation of the feasibility of using camera and Aruco for position feedback on a 4 cable-driven parallel robot

LIU Menghang
Graduate student
University of Lorraine,
Arts et Metiers Institute of Technology

Atal Anil Kumar
Postdoctoral researcher
University of Luxembourg

Jean-François ANTOINE
Maître de conférences
IUT Nancy-Brabois

François Léonard
Professeur des universités
University of Lorraine, Arts et Metiers Institute of Technology,
University HESAM

Gabriel ABBA
Professeur des universités
ENI de Metz

Vianney Papot
University of Lorraine, Arts et Metiers Institute of Technology,
University HESAM

***Abstract*—A Cable-Driven Parallel Robot (CDPR) uses wires instead of rigid links to move the end-effector of the mechanism. It has the advantages of a simple model, large workspace and low price, and has gained the attention of many researchers in recent years. Additionally, with the increase in labor costs, it has become a trend to replace manual labor with automation technology. Various robots have been widely used in manufacturing industry, service industry, and medical industry. Therefore, the combination of parallel robots and visual inspection technology has become an exciting development trend. This article is dedicated to the evaluation of the feasibility of using camera and Aruco markers for position feedback on a 4 cable-driven parallel robot.**

Keywords—Cable-Driven Parallel Robot, Vision, Depth Camera, Aruco.

I. INTRODUCTION

Cable-driven parallel robot (CDPR) have been studied by an increasing number of researchers as an important type of mechanism. CDPRs have the advantages of simple design, large working area and low price. In recent years, they have been increasingly used in lifting, machining, port cargo handling, shipbuilding, subsea exploration, debris removal, etc. The study of the CDPR end-effectors is an important indicator of the motion state of the mechanism and provides a baseline for simplifying the mechanism and diagnosing faults. Its detection accuracy is an important aspect of measuring the structural performance of the mechanism and it is necessary to ensure that the end effector achieves the intended state of motion. Therefore, the study of the detection of the final trajectory of the CDPR is of great importance [1].

This article is dedicated to evaluation of the feasibility of using camera and Aruco for position feedback on a 4 cable-driven parallel robot.

A CDPR is composed of four basic components. A platform or end-effector, which is positioned within a workspace to fulfill a specific task, cables to control and move the platform, winches which change the cable length, and a frame upon which these cables are fixed (Figure 1) [2].

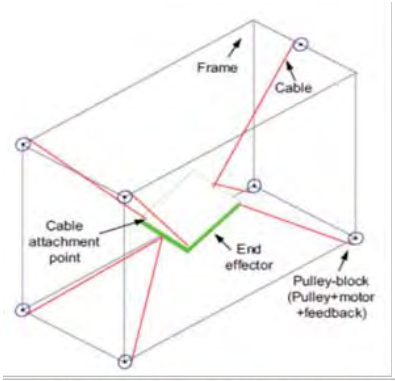


Figure 1: Schematic representation of Cable-Driven Parallel Robot

To position the robot, an Intel® RealSense™ Depth Camera D435 and an Aruco are used. The depth camera is fixed to the ceiling and the Aruco is pasted to the center of the platform (The end effector of the robot.). It can measure the relative pose of the platform by identifying the Aruco. The camera is connected to the PC and sends the measurement data to the PC. The experimental setup is shown in Figure 2. The operational flow of the system is shown in Figure 3.

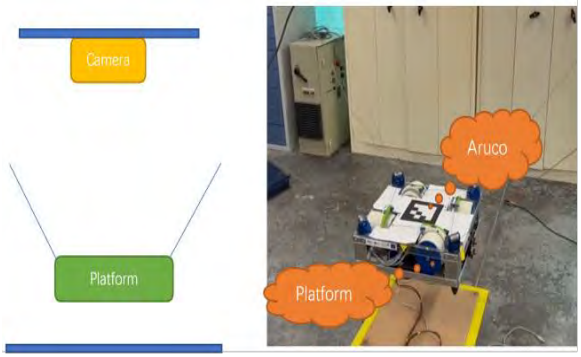


Figure 2: Measurement systems in the laboratory

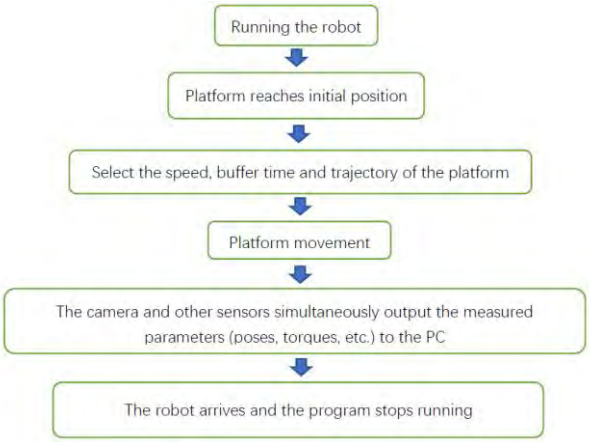


Figure 3: System flow chart

II. POSITIONING PLATFORM

The depth technology used in the camera is active stereoscopic projection. An active stereo display is a projector that projects an image, one moment projecting the signal seen by the left imager and the next moment projecting the signal seen by the right imager. At the moment when the signal is projected to the left imager, a control signal is sent from the workstation to control the left imager of the active stereoscope so that it opens, when the right imager is closed; conversely, when the image is projected to the right imager, the left imager is closed. At the same time, the independent IR projector on the camera sends a synchronisation message to the imagers.

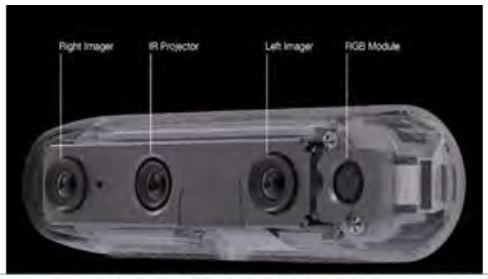


Figure 4: Intel® RealSense™ Depth Camera D435

The Aruco is used to determine the position and pose of the robot with respect to the camera in real-time. The Aruco tag was originally developed by S. Garrido-Jurado et al.in 2014 [3]. The Aruco library is a two-dimensional code tag library [4], which can be used for image recognition, graph localization and three-dimensional problem solving. The frame and four corners are mainly used for tag positioning The Hamming internal code [5] is mainly used for identity recognition, information validation

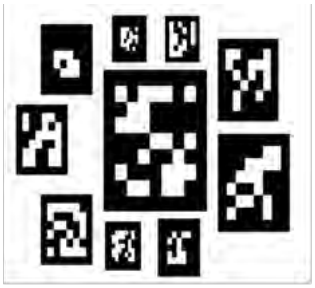


Figure 5: Aruco

and error correction. The detection of the Aruco code consists of two main parts: one part is the detection of the four corners of the Aruco code and the second part is the detection of the identification of the 2D code representing the numbers. Figure 6 illustrates the process of detecting Aruco codes.

As the Aruco ID is unique, the depth camera is programmed to recognize the code to detect the platform's position in real

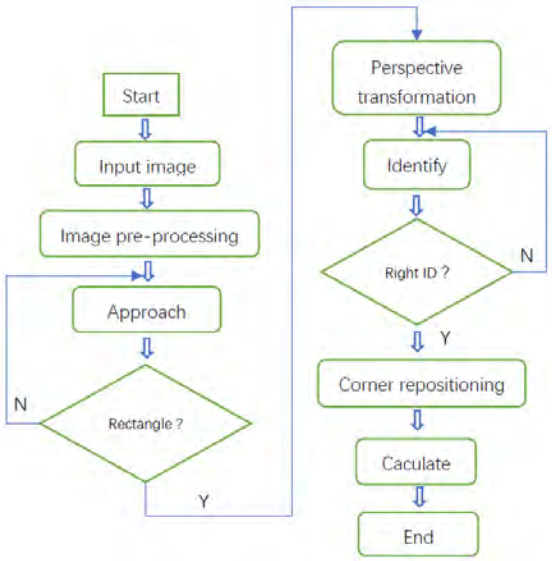


Figure 6: The process of detecting Aruco codes[6]

time. This method is much faster than monitoring by other sensors.

III. CAMERA PRECISION

The precision of the camera needs to be measured before the end-effector's precision can be improved. Precision measurement experiments with *Intel® RealSense™ Depth Camera D435* were carried out on a serial robot *IRB1200*.

The depth camera measures six parameters, three translational parameters x, y, z and three rotational parameters RV1, RV2, RV3.

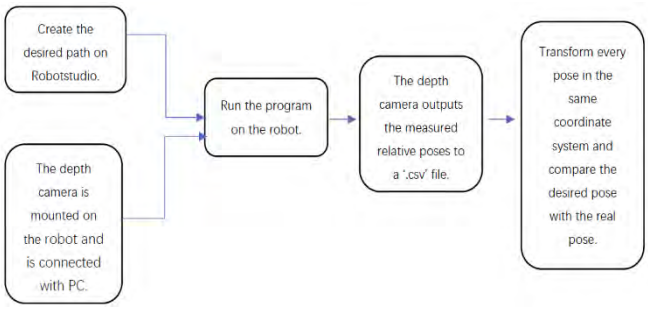


Figure 7: Experimental flow chart

A. *Stability of the camera*

The first step is to measure the stability of the camera. Each path consists of multiple poses and each pose is measured three times to obtain the standard deviation of the three data. The stability of the camera when measuring the path is then obtained by calculating the average of the standard deviations of the poses on each path. A total of six paths are measured: translation along the x-axis, y-axis and z-axis, and rotation along the x-axis, y-axis and z-axis.

	X(m)	Y(m)	Z(m)	RV1	RV2	RV3
Trans-X	0.0002	0.0002	0.0006	0.3140	0.0048	0.0302
Trans-Y	0.0000	0.0000	0.0002	0.1146	0.0018	0.0077
Trans-Z	0.0000	0.0000	0.0001	0.1664	0.0026	0.0117
Rotation-X	0.0001	0.0001	0.0005	0.4361	0.0089	0.0549
Rotation-Y	0.0000	0.0000	0.0002	0.0015	0.0007	0.0033
Rotation-Z	0.0000	0.0001	0.0003	0.1627	0.0370	0.0271

Table 1: Result of the stability

As can be seen from Table 1, the mean standard deviation of the translation parameters for all paths is less than 0.001m. This means that the camera is very stable in measuring the translation vectors.

Whereas for the rotation vector, the rotation vector RV1 has a large error, for the other two vectors, it is less than 0.01.

When the camera is measured multiple times in the same pose, the translation parameters are almost constant, while the measured rotation vectors have a large error.

B. *Translation and Rotation along x-axis, y-axis and z-axis*

The motion of an object in space has six degrees of freedom, they are: translation along the x-axis, y-axis and z-axis, and rotation along the x-axis, y-axis and z-axis. In order to measure the precision of the camera in each degree of freedom, this experiment uses the control variable method: each path moves along only one degree of freedom. Since there is an error between the camera's measured pose and the initial input pose, this experiment yielded both the error in percentage with $\frac{camera-Robot}{Robot}$ and the difference between the two with $|Camera - Robot|$ for error analysis. The results of the experiment are shown in Table 2.

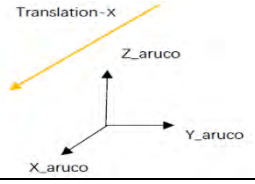
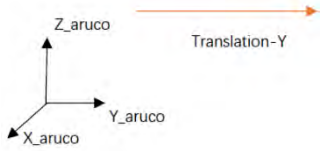
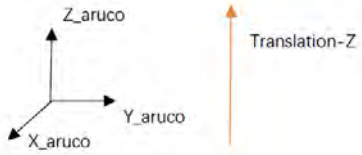
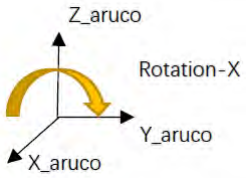
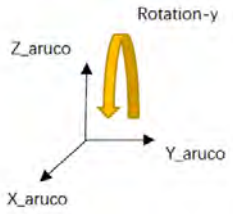
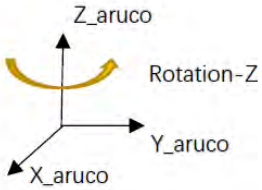
Trajectory	Result
Translation X 	An error in average 38% and in average 0.05m for a depth 5cm of ±0.3m is obtained.
Translation Y 	An error in average 17% and in average 0.0152m for a depth 5cm of ±0.18m is obtained.
Translation Z 	An error in average 6% and in average 0.0208m for a depth 5cm of 0.15m-0.65m is obtained.
Rotation X 	An error in average 2% and in average 3.4816 deg for an angle of -25deg to 25deg is obtained.
Rotation Y 	An error in average 58% and in average 6.9487 deg for an angle of -30deg to 15deg is obtained.
Rotation Z 	An error in average 6% and in average 2.1444 deg for an angle of -15deg to 35deg is obtained.

Table 2: Result of the experiment

In conclusion, the precision of the camera for translation along the X-axis is smaller than the other two with an error of 0.0515m between the input position and the real position, and the precision along the Y-axis and Z-axis are with an error of 0.0152m and 0.0208m between the input position and the real position.

And the precision of the camera along Y-axis rotation is smaller than the other two with an error of 6.9487deg between the input angle and the actual angle, and the precision along the X-axis and Z-axis are with an error of 3.4816deg and 2.1444deg between the input position and the actual position.

IV. Translation along Z-axis on the Cable-Driven Parallel Robot

After the camera precision measurements, the vertical translation along the Z-axis of the CDPR was studied and the transfer function was calculated. This experiment had three variables: different speeds of movement, different distances of movement and the presence or absence of buffer times. After a total of eight sets of experiments, the dataset with the best experimental results (buffer time = 5s, movement distance = 15cm and movement speed = 0.05m/s) was chosen to calculate the transfer function. Figure 8 shows the platform movement for the test. During the movement of the platform, the depth camera feeds the measured relative poses (Z_camera) to the computer and the robot also calculates the real-time poses (Z_rebuilt) of the platform from the cable length detected by the sensors and feeds it to the computer.

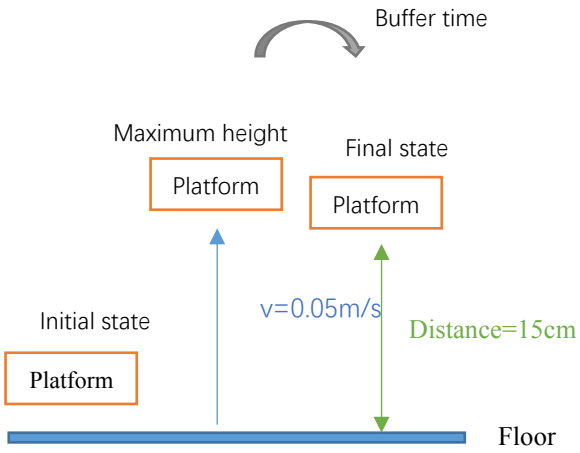


Figure 8: Platform movement process

Using MATLAB, here are the 2 transfer functions and the respective figures. In the figure 9, the three curves are: the height of the platform entered by the program over time (Z input), the height of the platform measured by the camera over time (Z camera) and the trajectory of the platform obtained by the calculated transfer function (Output model).

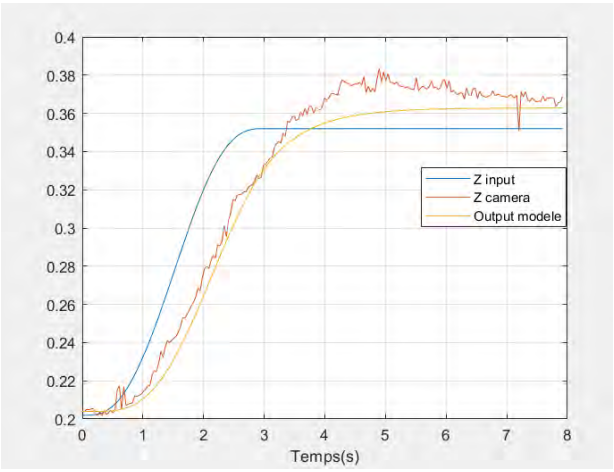


Figure 9: Z_camera, Z_input, Output of model

The transfer function between the input and output of camera is:

$$\frac{Z_{camera(s)}}{Z_{input(s)}} = \frac{12.62}{s^2+9.933*s+11.92} \quad (1)$$

With Static Gain: $K = 12.62/11.92 = 1.0587$; Natural frequency: $w_0 = \sqrt{11.92} = 3.45\ rad/s$; Damping: $\xi = 0.5 * 3.45 * 9.933/11.92 = 1.4374$.

In the figure 10, the three curves are: the height of the platform entered by the program over time (Z input), the height of the platform measured by the sensors over time (Z rebuilt) and the trajectory of the platform obtained by the calculated transfer function (Output model).

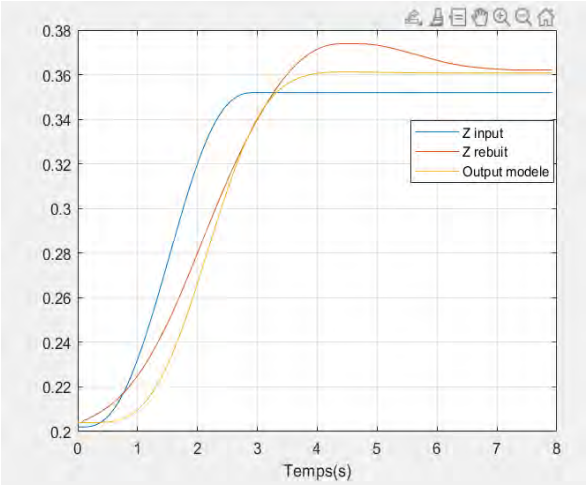


Figure 10: Z_rebuilt, Z_input, Output of model

The transfer function between the input and output of sensor is:

$$\frac{Z_{rebuilt(s)}}{Z_{input(s)}} = \frac{6.433}{s^2+4.199*s+6.155} \quad (2)$$

With Static Gain: $K = 6.433/6.155 = 1.0452$; Natural frequency: $w_0 = \sqrt{6.155} = 2.48\ rad/s$; Damping: $\xi = 0.5 * 2.48 * 4.199/6.155 = 0.8459$.

V. Conclusion

Both, the camera measurements and sensor calculations can be used to obtain a curve that approximates the input trajectory. The advantage of the camera is that it is more responsive and less subjected to external influences, as the cable's elasticity causes the wire winding to be slightly dependent to force and affects the sensor's calculations. The advantage of the sensor calculation is the greater detection range. The combination of the two can be used to improve the accuracy of monitoring to a greater extent.

References

[1] Shuaifei Zhu. 2019. Visual detection method of the end trajectory of the cable-driven parallel mechanism[D]. Henan University of Science and Technology.

[2] Atal Anil KUMAR. 2020. Design and Control of a Cable Robot for dexterous handling of parts on production lines[D]. University of Lorraine.

[3] S. Garrido-Jurado,R. Muñoz-Salinas,F.J. Madrid-Cuevas,M.J. Marín-Jiménez. Automatic generation and detection of highly reliable fiducial markers under occlusion[J]. Pattern Recognition,2014,47(6).

[4] Garrido-Jurado S , Muñoz-Salinas R, Madrid-Cuevas F J, et al. Automatic generation and detection of highly reliable fiducial markers under occlusion [J]. Pattern Recognition, 2014, 47(6):2280-2292.

[5] Gottesman, Daniel. A class of quantum error-correcting codes saturating the Hamming quantum bound[J]. Physical Review A, 1996, 54(3):1862-1868.

[6] Haiguo Fang. 2020. Research on 3D object detection and input technology based on Deep Learning [D]. Xiangtan University

Investigation of a concept for autonomous cleaning of solar panels with a robotic system under consideration of economic efficiency

Lisa Kopp, B.Eng.
University of Applied Science Trier, Environmental Campus Birkenfeld
Campusallee, 55761 Birkenfeld, GER
lis.kopp@umwelt-campus.de

Christian Konstroffer, B.Eng.
University of Applied Science Trier, Environmental Campus Birkenfeld
Campusallee, 55761 Birkenfeld, GER
c.konstroffer@umwelt-campus.de

Prof. Dr.-Ing. Matthias Vette-Steinkamp
University of Applied Science Trier, Environmental Campus Birkenfeld
Campusallee, 55761 Birkenfeld, GER
m.vette-steinkamp@umwelt-campus.de

Dr. Sebastian Groß
University of Applied Science Trier, Environmental Campus Birkenfeld
Campusallee, 55761 Birkenfeld, GER
s.gross@umwelt-campus.de

Abstract*—The generation of electricity through renewable energies is an important component of the German electricity mix. Photovoltaic and solar power systems have an important position among these green energies. Due to the exposed location of these systems to weather and environmental conditions, large efficiency losses can occur due to surface impurities. Even though cleaning solar plants can make sense from the point of view of increasing efficiency, this is not always the case from an economic point of view. In particular, small or difficult-to-access plants have a disadvantage here due to the high manpower requirements. So far, no mobile and autonomously acting systems can be found on the market. Either they have to be mounted on the plant or be controlled by operators. In addition, these robots must always be supplied with energy and operating materials via a pipeline. This problem was addressed here. The result of this work is a prototype of an autonomous cleaning robot that is independent of supply lines.

Keywords—autonomous Robot, solar, photovoltaic, cleaning

I. Introduction

The generation of green energy is an important factor in today's power generation. Solar and photovoltaic plants play a leading role in the process. The percentage of these plants, together with wind energy, bioenergy and hydropower, in German electricity generation is 50.6% [1]. Photovoltaic (PV) installations alone accounted for 10.5% of the German electricity mix, according to data from the transmission system operators. This share corresponds roughly to 51.4 TWh of electrical energy [1].

Photovoltaic systems are mostly found as rooftop systems or as large solar farms on open spaces. In Germany, around 90% of all PV systems are roof systems with a nominal output of up to 30 kWp (kilowatt peak). In 2020, for example, they generated a total output of 18.4 GWp [1]. A kilowatt peak is the unit of maximum output of solar modules. One kWp can generate about 1000 kWh of electricity per year, which requires about seven to ten square metres of solar surface [2].

Solar installations are exposed to external weather conditions all throughout the year, which is why dirt appears after a certain period of time. This can consist of dust, moss or bird droppings, for an example, and can lead to the performance being impaired [3].

A 2018 research study by Milk the Sun GmbH showed that solar installations with a tilt angle below 35° are particularly sensitive to soiling and that the effects of regular cleaning are the most significant [4]. Depending on the location of the PV systems, efficiency losses can vary. A measurement of a solar surface at the edge of a forest, for example, showed a loss in efficiency of up to 17% due to soiling by dust and moss [5]. The feed-in tariff for photovoltaic systems in April 2021 was 5.36 cents/kWh for systems up to 750 kWp [6]. For a PV system with an output of 500 kWp, which is a solar surface of about 5000 m², the average kWh production is about 500,000 kWh or 500 MWh per year. This would result in a feed-in tariff of 26,800 €. With an efficiency loss of 17%, the energy production would only be 415 MWh, i.e. a feed-in tariff of 22,244 € and a loss of 4556 €. Regular cleaning is therefore very important.

However, cleaning the solar surfaces is very difficult. Due to the fact that the PV systems are mostly found on roofs, there is an increased risk of injury from falling. The size of the surfaces and the slippery surface also make cleaning complicated. Incorrect cleaning can also damage the surface [7].

So far, the cleaning can only be carried out manually, which is often very expensive. Depending on the accessibility of the system, average costs of approximately 1.00 to 2.50 euros per m², additionally travel costs, are charged for photovoltaic cleaning. The photovoltaic cleaning prices are made up of the size of the system, the type and severity of the soiling, the access to the system and the technology used for cleaning [8].

Another reason why the conventional methods are limited is that the cleaning industry has a very high staffing requirement and therefore a shortage of skilled workers, which is why cutbacks make sense [9].

Possible solutions are autonomous robotic systems for cleaning PV and solar plants. Regular cleaning can prevent heavy soiling and therefore significantly reduce efficiency losses.

Due to the autonomous operation, no complex cleaning measures by external service providers are necessary. In addition, the robot system with an optimised concept enables more efficient, water-saving and thus more economical cleaning.

The automatic cleaning systems already available on the market can be divided into two categories. The first category are autonomous systems that use rainwater for cleaning, but are stationary, i.e. mounted on the PV system [10]. The second category are remote-controlled robots, which are placed on the PV system [11, 12]. In this case, at least one person is always necessary to control the robot and guide the supply cables along the system. These robots usually require an external power and water supply. Depending on the model and cleaning principle, demineralised water is used here [11]. Depending on the type of movement, an additional pneumatic supply is required [11]. The movement on the PV systems, i.e. an angled and slippery surface, is a problem that has already been solved, which many manufacturers and suppliers demonstrate with different solutions. Rubber tracks [13] or pneumatically operated types of movement are used for this purpose, for example [11, 12].

Partially automated systems, in contrast, are only used for large solar farms. Systems that work hand-operated are also used. Water (sometimes with detergent) is fed from a supply line and the modules are then cleaned with sponges or brushes. This type of cleaning is very resource-intensive, as cleaning is often done with a lot of water [11].

II. Challenges

A. *Structure of a solar module*

The materials used inside a solar module make it very sensitive. Therefore, PV systems require a certain cleaning.

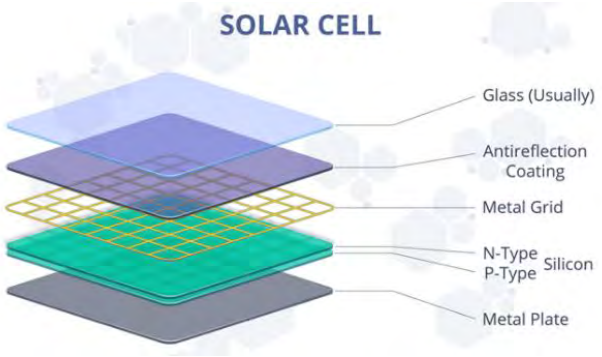


Fig. 1: Structure of a solar cell [14]

Solar modules consist of several layers. Figure 1 shows that the upper layer is usually tempered glass. Underneath, there is an anti-reflective film, metal and silicon, which convert the sun's radiation into usable energy [14]. On the one hand, the tempered glass serves as a protective layer, but it should also ensure that the solar radiation can be transmitted reliably. Depending on the manufacturer of the solar modules, other transparent materials can also be used as top layers. However, the material of the top layer is sensitive and requires special cleaning. Chemical cleaning products are usually not permitted, according to the manufacturers. Water and gentle brushes are recommended for cleaning so that the surface is neither damaged nor attacked by harsh cleaning products [15].

Several solar modules are lined up on roof surfaces or solar farms. They are attached by using frames, which are mounted on a base construction. There are gaps between the individual modules, which vary in size depending on the manufacturer. This is illustrated in Figure 2.

Fig. 2: Gap between the solar modules [1]

This fact plays a major role in the implementation of the path planning and the design of the sensors. The robot must be able to overcome the gaps as well as recognise that it is only a gap and not an outer edge.

B. *Requirements for an autonomous robot system*

The robot system must be able to navigate and move autonomously on the roof. The key is to build a suitable sensor system that can reliably detect the outer edges and distinguish them from the gaps between the solar panels. Reliable detection of the edges is made more difficult by the weather conditions and the smooth surface. No other obstacles are expected on the PV surface besides the outer edges. PV systems that have a distance of less than 5 cm to skylights, chimneys, antennas or other obstacles are not considered by the presented solution. The problem of robot control and path planning is considered to be minor, as corresponding applications are already available on the market in the field of service robots. Vacuum cleaning robots that can recognise steps are a good example.

The robot system has several weight-limiting factors, such as the maximum payload of the module and the worker. A safe snow load of 5400 Pascal per module is given by manufacturers [16]. With a module size of 1.64 m², this corresponds to a total load of 8856 N, which is equivalent to up to 903 kg with appropriate distribution on the solar panel [16]. Transporting the robotic system to the PV plant can take place under non-ergonomic conditions. The following options are considered here:

- Putting down at the ground level up to 1.80 m without any aids

- Putting down from a lifting platform onto the roof surface
- Putting down from a skylight onto the PV system
- Putting down from a ladder onto the roof surface

According to German labour law, a regular load may weigh a maximum of 10 kg, for pregnant persons only 5 kg [17].

The robot needs its own power supply, which shall be provided by a rechargeable battery. This is necessary because with a wired power supply, the cable would have to be guided by a worker or at least checked so that it does not get tangled. This would make autonomous operation impossible.

Water is needed for cleaning, but the robot will clean without a water supply. Only rainwater, which rains off before, during and after driving, is used.

Some robot solutions for PV systems use demineralised water. The reason for this is that, in contrast to water containing minerals, no lime stains can occur when drying. However, this danger only exists in direct sunlight or if the water evaporates in a short time [7]. If the cleaning process takes place during a rainfall, demineralised water is therefore not required. This is also demonstrated by the "SolarProtec" cleaning system from Osborn International GmbH [10]. Water volumes of more than 12 l/min are recommended for cleaning [18]. Even with a small area that would be cleaned within one minute, this would mean that the water to be transported would already require a load of about 12 kg. A robot of this size would no longer comply with the above-mentioned weight restrictions.

The robot must have a waterproof housing. This must keep out two types of water: Firstly, splashing water from the wet panels and the track drive, and rainwater. The housing must contain the battery, the on-board electronics and the electric motors. The sensors must also be installed waterproof, but outside the central housing. Special attention must be paid to the transitions between the electric motors and their shafts, as well as the cables leaving the housing. The housing must also have a removable lid, which must be closed watertight. The lid is needed to reach the battery and the on-board electronics.

III. Robot Construction

A. *Cleaning concept development*

Based on the previous conditions for cleaning solar installations, an optimized cleaning concept was developed with which the autonomous robot system can clean PV installations. For the tests, different cleaning tools were used and tested on glass surfaces. The cleaning tools used are shown in Figure 3.

The glass surfaces showed a medium degree of soiling. They were then cleaned with normal tap water using different cleaning tools. The cleaning was carried out in wiping and rotating movements. The result was that rough dirt, such as bird droppings, should be loosened first. The easiest way to do this was with a combination of water and a silicone sponge **(5).**

Loose dirt, such as dust and pollen, could be removed with all tools in combination with water. When it came to drying, however, there were big differences between the cleaning methods. The common household sponge **(2)** did not pick up the dirt, but spread it over the surface, which is why there were strong streaks. Other sponge structures, such as the bottle brush **(4)** and the microfibre cloth **(3)**, also did not leave a clean result. The microfibre cover **(1)**, on the other hand, cleaned the surface sufficiently and absorbed the dirt completely. Due to the structure of the cover, the wet glass surface was polished afterwards and no water residues or streaks were left.

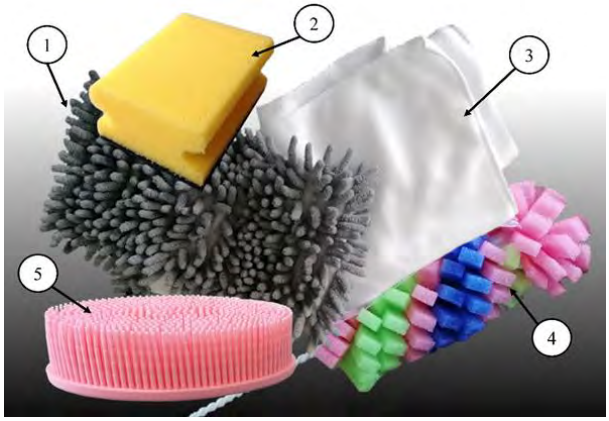


Fig. 3: used cleaning tools

The final cleaning concept has now been put together from a combination of the tools. The dirt is loosened with the help of rotating silicone sponges. The loosened dirt is then picked up by a microfibre cover and the glass surface is polished with it. Figure 4 shows the cleaned glass surface.

Fig. 4: dirty vs. cleaned glass surface

B. *Description of the robot construction*

The main components of the robot are mostly made of 3D printed parts. Manufacturing by 3D printing has the advantage that the components are light, cheap as well as quick and easy to produce. Figure 5 shows the CAD model of the developed robot.

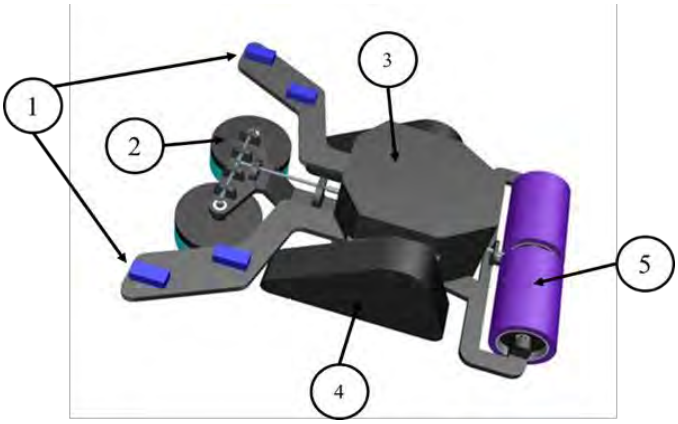


Fig. 5: CAD-model of the cleaning robot

At the front end of the robot are two sensor arms **(1)** in which the sensors for navigation are integrated.

The cleaning mechanisms are also located at the front and back of the robot. At the front end there are two rotating silicone sponges **(2)** that can loosen the rough dirt. At the rear end there is a rotating mechanism with a microfibre cover **(5)**. In this way, the surface is polished and the dirt is picked up.

The housing **(3)** should also be waterproof, which is why all unprotected openings are shielded from moisture caused by rainwater with a rubber membrane or sealing compound.

To ensure reliable traction and movement on the slippery solar modules, there are several possibilities. Since the robot is supposed to move autonomously, drives for which an additional supply line would be necessary are excluded. This is the case, for example, with vacuum feet or pneumatic nubs. A reliable possibility is a chain wheel drive with rubber tracks **(4)**. These are already present in other systems on the market and can negotiate a slope of up to 30° without any problems, even on wet surfaces [19].

Inside the housing there are various electronic components. On the right and left, there is a DC motor to drive the chain wheel drive. These motors are controlled by a motor control unit. A program on a microcontroller defines when and in which way the motors are controlled in order to let the robot move on the desired path. The cleaning mechanisms are each driven by a DC motor. With the help of bevel gears, the momentums are transmitted to the corresponding shafts.

IV. Sensors and Path Planning

A. *Placement of the sensors*

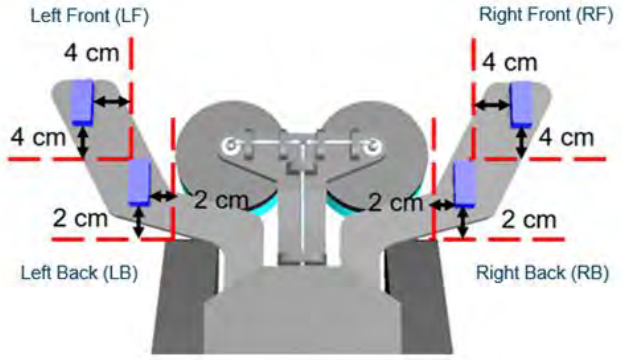


Fig. 6: Sensor placement on robot

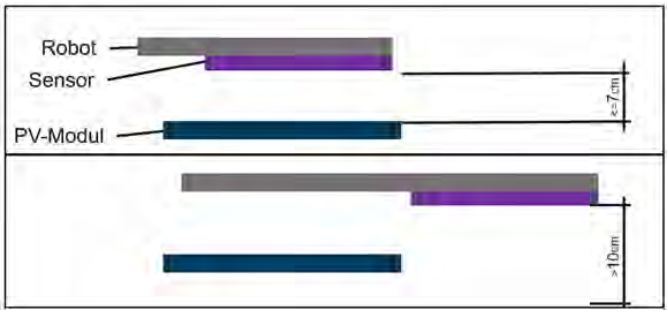


Fig. 7: Sensor placement over solar modules

The outer edges of the solar modules play a major role in path planning. There are four sensors on the front of the robot, two on the left and two on the right (see figure 6). Using the circuit diagram in Table 1, the robot drives left, right, straight ahead or turns. Their signal is switched binary between 1 or 0 depending on whether the PV system is detected below them or not. The sensors measure the distance between the sensor and the ground. The distance between the sensor and the PV module is defined by the height of the mounts, which is 7 cm (see Figure 7). If the distance is 10 cm or more, it is a gap between two solar modules or an outer edge of the modules; then the sensor value is switched to 0, otherwise it is 1 (Figure 7). If both sensors on one side are set to 0, the robot will move in the opposite direction, as there must be an outer edge there. If all 4 sensors are set to 0, the robot turns because it has reached the end of a path. If it switches individual sensors to 0 in between, this is a gap, as these are 1-3 cm apart, depending on the design [20].

The direction of travel, or deviations from it, are measured with an acceleration sensor. The "Joy-it MPU 6050" was used here [21]. With the help of this sensor, deviations are detected and counterbalanced when driving straight ahead. Furthermore, the sensor is used to detect how far the robot has rotated during the turning manoeuvre.

The ultrasonic sensors used cover the required measuring range with a measuring distance of 2 cm to 4 m [22]. During the measurement, the condition and moisture of the surface have no

significant influence on the measurement result. This is important because both the moisture of the solar system and the roof covering can differ and they can also be made of different materials.

TABLE I. SENSOR SIGNAL SCHEDULE

Direction of travel based on sensor signal					
Driving direction	DEC	FL	BL	FR	BR
Forward	15	1	1	1	1
Forward	14	1	1	1	0
Forward	13	1	1	0	1
Rotation left	12	1	1	0	0
Forward	11	1	0	1	1
Forward	10	1	0	1	0
Forward	9	1	0	0	1
Rotation left	8	1	0	0	0
Forward	7	1	1	1	1
Forward	6	0	1	1	0
Forward	5	0	1	0	1
Rotation left	4	0	1	0	0
Rotation right	3	0	0	1	1
Rotation right	2	0	0	1	0
Rotation right	1	0	0	0	1
Turnaround	0	0	0	0	0

The microcontroller "Elegoo Mega2560" was used to process the sensor signals in order to control the electric motors of the drive and cleaning system [23]. The "Saberthooth Dual" motor driver was connected between the drives and the microcontroller [24]. The "AZDelivery L293D" motor driver was installed between the microcontroller and the cleaning system [25].

B. *Description of the path planning*

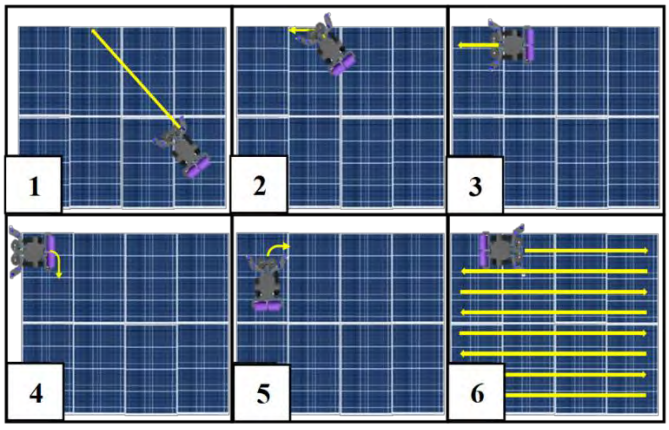


Fig. 8: Path development

Figure 8 shows the path that the robot follows in six steps.

Step 1: The robot is placed on the PV system and moves forward. When putting it down, it must be ensured that it reaches the upper edge of the roof.

Step 2: The robot reaches the upper edge. This is detected because two sensors on one side now switch to 0.

Step 3: The robot aligns its position and orientation parallel to the upper edge.

Step 4: The robot starts the cleaning unit and drives to the end of the roof.

Step 5: Once it reaches the end of the roof, it turns backwards and returns to the first track.

Step 6: The robot now drives along the tracks. It moves horizontally from one side of the plant to the other. Each lane is offset by 15 cm, so that each lane is offset by half. This guarantees that all dirt is removed with both the brushes and the microfibre cloth. The effective cleaning width per lane is therefore 15 cm.

The dirt on the system is loosened from the top down and can be rinsed off by the rain.

V. RESULTS

The completed prototype of the robot is shown in Figure 9.

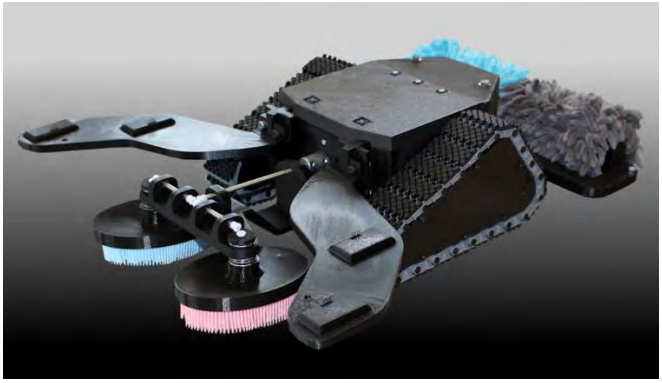


Fig. 9: Robot prototype

Simplifications were made for the construction of the prototype, as the focus was on testing the sensor technology and path planning as well as the cleaning system. The energy supply was provided by cable, which means that no battery was needed during the tests and the required energy demand of the system could be measured. It must be considered that the weight of the battery will lead to an increase in the total energy requirements.

The robot has a total weight (without battery) of 2.25 kg, which corresponds to 22.1 N. The tracks have a contact area of about 0.02 m². Thus, the robot applies a pressure of 1103.06 Pa. This is less than the specified 5400 Pa of a solar module [16]. In this calculation, the contact surface of the cleaning system was not taken into account, as there is almost no load there.

The robot has a cleaning speed of 0.5 m/s. The exact cleaning time for a PV system could not be tested experimentally, so the time for cleaning a roof can only be estimated. The following

formula can be used to calculate the time required to clean a roof.

$$T_C = T_S + N * (2\,T_T + V_R * L) \quad (1)$$

T_c is the overall cleaning time
T_s is the starting time until the first rotation
N are the number of lanes, which is calculated below
L is the length of the roof
V_R is the speed of the robot

$$N = \frac{B_R}{B_{eff}} \quad (2)$$

B_{eff} is the effective cleaning width per lane
B_R is the width of the PV system

For a large private PV system of 60 m², the robot needs about 630 seconds, or 10.5 minutes [26]. The calculation was based on the assumption that the PV system is rectangular and has a width of 6 m and a length of 10 m. The robot was put down at the bottom right corner so that it first had to drive 15 m to the top right edge, which takes 30 seconds travel time. For one turn, 5 seconds were needed.

A battery of 500g would already cover the energy demand for 2 hours and 12 minutes. An average battery has an energy density of 200 Wh/kg [27]. The robot requires a measured power of 45.3 W at an operating voltage of 12 V. For the microcontroller, the voltage must be regulated to 5 V.

VI. Conclusion

Within the paper, a new cleaning system was developed in combination with an autonomously driving robot. For this concept and the result, new applications in the cleaning of PV and solar plants are considered conceivable and possible.

In addition to the already mentioned possibilities to bring the robot to elevated positions, such as roofs, transport with a drone would also be conceivable. This would offer further advantages for the PV system cleaning business model. The advantage could be that the robot could be brought to the roof with little time effort. There, the drone would drop the robot off and pick it up after cleaning. This extension is a complicated task, but would revolutionise the entire concept and could make the cleaning market for hard-to-reach and relatively small PV modules attainable.

Another possibility as an alternative to cleaning during rainfall would be the external supply of water with the help of heavy-duty drones. This would make the robot autonomous, but still ensure a reliable water supply.

Acknowledgment

The project was realised within the competition "Trinatronics 2021", which is organised and financed by the alliance TriRhenaTech. In total, the construction of the robot was funded with 1057€. Further thanks go to our colleagues at ENSISA - École nationale supérieure des ingénieurs sud Alsace in France, for their support.

References

[1]. Henrik Te Heesen, Volker Herbort, Martin Rumpler (2021) Studie zum Ertrag von Photovoltaikdachanlagen 2020 in Deutschland. University of Applied Sciences Trier
[2]. E.ON Energie Deutschland Was bedeutet die Maßeinheit kWp? https://www.eon.de/frag-eon/themen/solar/fragen-und-antworten/was-bedeutet-die-masseinheit-kwp/. Zugegriffen: 13. August 2021
[3]. Pro Energy Solutions Reinigung PV-Anlage. https://www.pro-energy-solutions.de/photovoltaik-reinigung/
[4]. Milk the Sun GmbH (2018) Reinigung von Freiflächenphotovoltaikanlagen. Wann lohnt sich die Reinigung von PhotovoltaikFreiflächenanlagen in Deutschland?
[5]. Moor D (2016) Bauwerksintegrierte Photovoltaik - Details im Fassadenbau. Stahlbau 85(S1):173–181. doi:10.1002/stab.201690156
[6]. EIGENSONNE GmbH Einspeisevergütung 2021: wie viel bekommen Sie für Ihren Strom? https://www.eigensonne.de/einspeiseverguetung-eeg-umlage/. Zugegriffen: 18. August 2021
[7]. Time Solution - Fensterputzlager (2012) Reinigung von Solar- und Photovoltaikanlagen. Fensterputzlager.de
[8]. 11880 Gebäudereiningung (2021) Photovoltaik Reinigung Preise | Beispiel | PV-Reinigung-Intervall. https://www.11880-gebaeudereinigung.com/preisvergleich/photovoltaikreinigung-preise. Zugegriffen: 09. August 2021
[9]. Bundesverband der Deutschen Volksbanken und Raiffeisenbanken Gebäudereiniger. VR info Branchenspezieal 2008
[10]. Tanja Peschel (2015) Automatisches Reinigungssystem für große PV-Anlagen. VDE Verlag GmbH. https://www.sonnewindwaerme.de/photovoltaik/automatisches-reinigungssystem-fuer-grosse-pv-anlagen. Zugegriffen: 14. August 2021
[11]. SERBOT AG GEKKO Solar Hightech Roboter. Mobiler Reinigungsroboter für Photovoltaik-Dachanlagen. https://www.serbot.ch/de/photovoltaik-reinigung/gekko-solar-roboter. Zugegriffen: 14. August 2021
[12]. Staudinger GmbH Helios PV Reinigungsroboter. Über uns. http://www.pv-roboter.de/. Zugegriffen: 14. August 2021
[13]. Christian Ulrich (2021) Solar-Cleaner: Ferngesteuertes Kettenfahrzeug zur PV-Anlagenreinigung. https://www.ulrichc.de/project/order/solar-cleaner/. Zugegriffen: 14. August 2021
[14]. Benjamin Stafford (2019) What Are Solar Panels and How Do They Produce Electricity? https://matmatch.com/blog/what-are-solar-panels-and-how-do-they-produce-electricity/
[15]. Solar Frontier Europe GmbH Installations- und Bedienungsanleitung. PowerSets mit 1-phasigem Wechselrichter
[16]. Heckert Solar Datenblatt Nemo 2.0

[17]. VFR Verlag für Rechtsjournalismus GmbH Lastenhandhabungsverordnung (LasthandhabV) - Arbeitsrecht 2021. Arbeitsrechte.de
[18]. Michael Mattstedt (2020) Trend zum „Wasser-Sparen“ schadet dem Photovoltaik-Betreiber. Ökologische Solarreinigung
[19]. Photovoltaik Liebold GbR Roboter Reinigung. https://www.pv-liebold.de/reinigungssysteme/roboterreinigung/. Zugegriffen: 15. August 2021
[20]. IBC SOLAR AG (2014) Installationsanleitung. Photovoltaikmodule der Typen IBC SoloSol xxx CS
[21]. SIMAC Electronics GmbH Joy-IT Bewegungssensor MPU6050. https://joy-it.net/de/products/SEN-MPU6050. Zugegriffen: 18. August 2021
[22]. ELEGOO Official ELEGOO Ultrasonic Distance Sensor Module Kit (Pack of 5, HC-SR04). https://www.elegoo.com/products/elegoo-ultrasonic-sensor-kit. Zugegriffen: 18. August 2021
[23]. ELEGOO Official ELEGOO MEGA 2560 R3 Board with USB Cable Compatible with Arduino IDE. https://www.elegoo.com/products/elegoo-mega-2560-r3-board. Zugegriffen: 18. August 2021
[24]. RobotShop Sabertooth Dual 12A 6V-24V R / C Regenerativ Motor Treiber. 12A 6V-24V R / C Regenerativ Motor Treiber. https://www.robotshop.com/de/de/sabertooth-dual-12a-6v-24v-r-c-regenerativ-motor-treiber.html. Zugegriffen: 18. August 2021
[25]. AZ-Delivery Vertriebs GmbH 4-Kanal L293D Motortreiber Shield Schrittmotortreiber für Mega 2560 und UNO R3, Diecimila, Duemilanove. https://www.az-delivery.de/products/4-kanal-l293d-motortreiber-shield-schrittmotortreiber. Zugegriffen: 18. August 2021
[26]. Jens Piasecki (2018) Photovoltaik als Solaranlage - Vorteile, Kosten, Wirtschaftlichkeit. Energieheld GmbH
[27]. Wissenschaftliche Dienste Deutscher Bundestag Energiespeicher der Elektromobilität. Entwicklung der Energiedichten

Detection of cuts on vascular stents using convolutional neural networks

Tobias Christopher Elbert
Department of Robotics and Remanufacturing
University of Applied Science,
Envivronmental Campus Birkenfeld
Campusallee, 55761 Birkenfeld
tobiaselbert@outlook.de

Matthias Vette-Steinkamp
Department of Robotics and Remanufacturing
University of Applied Science,
Envivronmental Campus Birkenfeld
Campusallee, 55761 Birkenfeld
m.vette-steinkamp@umwelt-campus.de

Wolfgang Gerke
Department of Robotics and Remanufacturing
University of Applied Science,
Envivronmental Campus Birkenfeld
Campusallee, 55761 Birkenfeld
w.gerke@umwelt-campus.de

***Abstract*—Within the InStent project, project partners are working on a solution to automate the quality assurance process and thus increase process efficiency, as well as minimize the risk of an undetected defect. Therefore, an automated optical inspection is to take place by means of a robot. In this context, problems with the handling of the stents, the image acquisition and the classification of the images have to be solved. [1]**

To inspect the stents for cuts, one of the possible defect classes, the application of Convolutional Neural Networks will be investigated. Conventional image processing methods, such as corner detection and various point operators, are used to preprocess the images to achieve the best possible result in classification. The chosen inductive learning approach assumes the existence of training data. Following this, a dataset of 932 images is generated. The results from the test data set show that the method promises sufficient accuracy to be used in production as an assistance system in quality assurance. However, in order to achieve this, it is still necessary to extend the training data set and, if required, to adjust the sensitivity of the test procedure.

***Keywords*—Convolutional Neural Network, InStent, Vascular Stent, Cut Detection, Image Classification, Image Preprocessing, Transmitted Lighting**

I. INTRODUCTION

According to the German Society of Angiology, vascular disease is one of the most common diseases. A common method of treating vascular disease is the usually minimally invasive insertion of a vascular stent. Stents are medical implants that are inserted into narrowed vessels to widen them and thus ensure continued normal blood flow.

In addition to the already high demands on product quality within the medical technology industry, errors in production pose a serious risk to patients. These special quality expectations require the highest standards of quality assurance methods on the part of the manufacturer. At the same time, in a high-wage country like Germany, production costs play an essential role in order to be competitive on the international market. Manual inspection of the stents is a time-consuming but also tiring task for the employees. Even partial automation of the inspection process would have the potential to not only significantly improve the effort required and the associated costs, but also the quality of work. This is to be achieved by a robot-guided optical inspection system in which image acquisition and evaluation are automated.

For the evaluation of the acquired images, conventional methods and modern approaches from the field of computer vision are available. Since both the geometries of the individual stent types and their defects can vary greatly, inspection using only conventional methods of image processing proves to be difficult, but the application of these methods is not ruled out. Using conventional methods, it has already been possible to detect braiding defects on stents [2]. As a counterpart, artificial intelligence methods, on the other hand, have already proven to be powerful in similar tasks in other fields (cf. with case study on the classification of defects in solar cells) [3]. This suggests that such methods may also be suitable for stent inspection. Since these AI methods are less demanding in terms of image acquisition compared to conventional methods, they could represent a simpler and more flexibly applicable solution option. Therefore, it is investigated how machine learning methods, supported by classical image processing methods, can be applied to combine the advantages of both disciplines and thus ensure reliable quality assurance.

II. STATE OF THE ART IMAGE CLASSIFICATION WITH METHODS OF MACHINE LEARNING

Machine learning training algorithms usually fall into the following three categories:

- Supervised Learning
- Unsupervised Learning
- Reinforcement Learning

For classification tasks, an inductive learning approach is usually chosen (supervised learning). This learning approach assumes a set of examples that can be learned from, i.e. whose correct result is already known [4, p. 19 f.], [5, p. 15 f.] The regularity derived from the training data is used to assign similar data to a class that is not already covered in the same form by the examples.

Commonly used learning models of supervised learning are support vector machines, decision trees and artificial neural networks. The latter have proven to be particularly well suited for the classification of image data [6], since the huge number of attributes (in this case the values of all pixels of an image) can be processed comparatively efficiently with them by convolution and pooling operations.

Support Vector Machines try to separate an n-dimensional point cloud from each other by a hyperplane for classification [5, pp. 19–20]. Decision trees, on the other hand, take the approach of breaking down a complex problem into a finite number of easy-to-make distinctions, which can then be assigned to a class [7, pp. 217–219]. In contrast, the structure of artificial neural networks is based on the biological functioning of the brain [7, p. 275]. This structure essentially consists of neurons and weighted connections between the neurons. The quality of a neural network is not proportional to its size and complexity. For many problems, comparatively simple structures are already sufficient [8]. The learning process in an artificial neural network is done by adjusting the weights, which establishes a relationship between a characteristic input and the corresponding output. The adjustment of the weights within the neural network is done by a gradient descent procedure in which the error, that is, the deviation of the value predicted by the neural network from the actual value, is minimized [9].

For neural networks there are a variety of possible architectures, for the respective application purposes [10]. For processing image data, Convolutional Neural Networks have proven to be particularly powerful [6]. In this architecture, so-called convolutional and pooling layers are used to highlight features on images while drastically reducing the number of pixels. In a convolutional layer, a convolution operation is performed as known from classical image processing [11, p. 33]. Here, a convolution kernel is applied to a section of the image, and the result of this operation is a scalar. By shifting the section pixel by pixel by a defined step size (stride), a new image is created pixel by pixel, the so-called feature map. Different strategies (padding) can be used for the edge areas, which expand the image in different ways to ensure a constant number of pixels. Alternatively, no expansion can be done. In this case, the reduced number of pixels must be considered in the following layers. This procedure is called valid padding.

In a pooling layer, the number of pixels is reduced by combining an image section (usually 2x2 pixels) into just one pixel. One of the most commonly used methods is MaxPooling, where the highest value of the matrix is selected and written into a new feature map. [12, p. 9]

III. Generation of a Training/Test Data Set and Classical Image Processing to Prepare the Image Data for the Neural Network.

Due to the project being in its early stages, no functional inspection station existed over its duration. Nevertheless, in order to generate image data for training and testing the Convolutional Neural Network, it was necessary to set up a provisional camera setup, in which one limits oneself to the available resources of the later inspection station.

As shown in Fig. 1 the setup used an RGB matrix camera to capture the image sections. The stent is placed in a holder on a luminescent plate, which diffusely illuminates the stent from the underside.

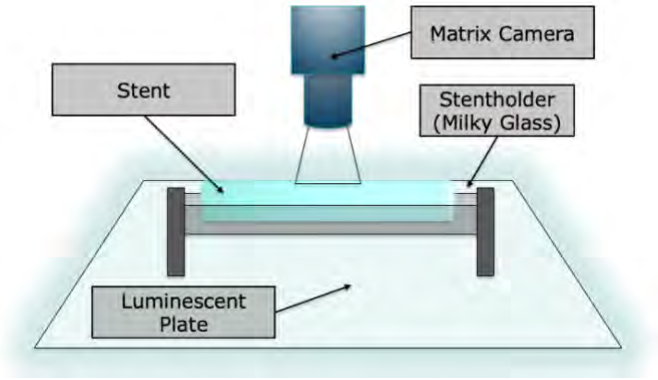


Fig. 1. Diagram of the camera setup

With the appropriate aperture setting, the illumination resulting from the transmitted light causes the wire to appear nearly black in the image, while the background disappears into an overexposed area. Thus, the contour of the stent, over which the stent can be examined for a tear site, is clearly visible.

Training a neural network directly with the original images would be feasible, but not recommended, since the error region is too small with respect to the acquisition area. By detecting the vertices, it is possible to define a region of interest starting from the wire intersection points. Fig. 2 shows a flowchart of this process, including the sorting into the predefined classes.

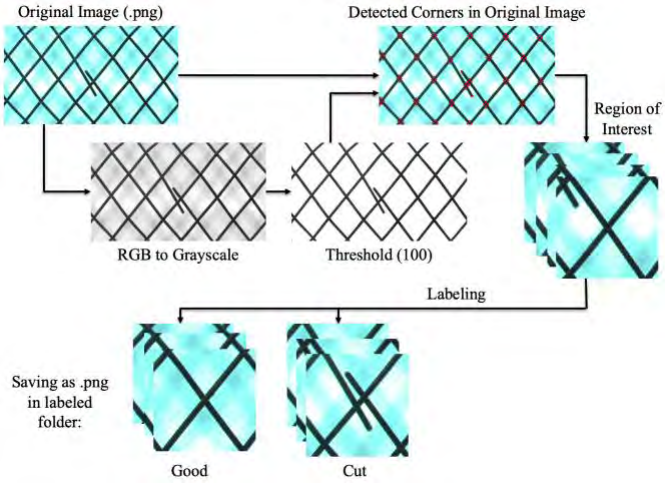


Fig. 2. Flowchart of the segmentation and labeling process

Following this, the extracted image areas are converted to grayscale and a thresholding procedure is applied to remove the background. In addition, the image area is inverted so that the irrelevant area receives the pixel value 0 (See Fig. 3).

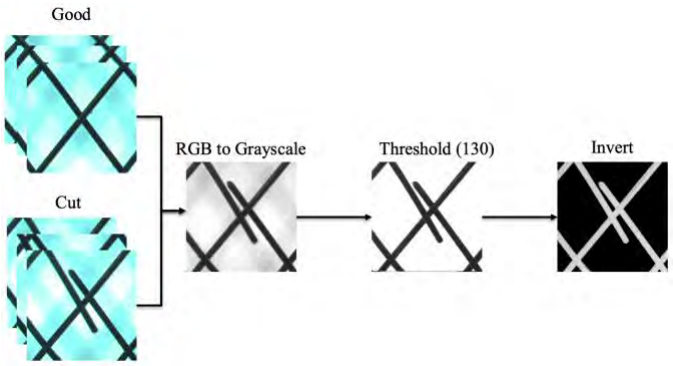


Fig. 3. Flowchart of image pre-processing following **Error! Reference source not found.**

The generated dataset, minus the images unsuitable for the training process, comprises 932 image sections (691 training / 241 validation). Since this is a relatively small size for a dataset in the deep learning domain, it can be artificially enlarged by transposing and rotating the original image section several times. This process is depicted in Fig. 4 and is known as data augmentation. It allows increased performance without manually acquiring more data [13].

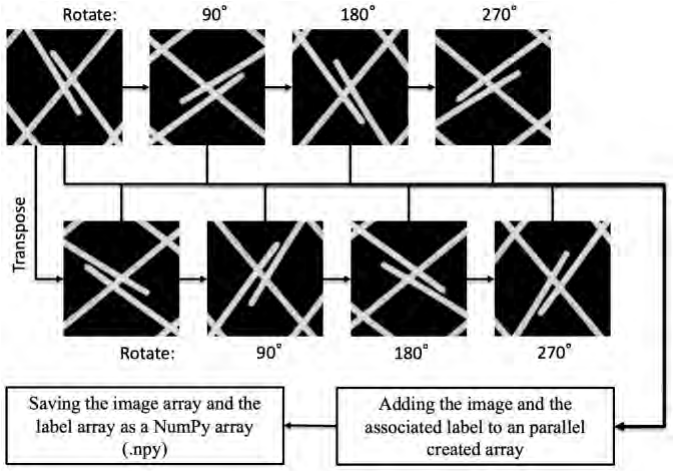


Fig. 4. Transpose and rotate the preprocessed data to enlarge the data set following **Error! Reference source not found.**

IV. Neural Network for Cut Detection

The architecture of the applied Convolutional Neural Network consists of four alternating Convolutional and MaxPooling layers followed by three fully connected layers and the output layer. (See Fig. 5)

When traversing the feature learning section, the number of pixels reduces from 130 x 130 to 5 x 5 pixels in 8 feature maps. Accordingly, the layers of the Classification section have 200 neurons in each layer. The output layer has two neurons for the classes "good" and "cut". The framework employed here is Keras from Tensorflow.

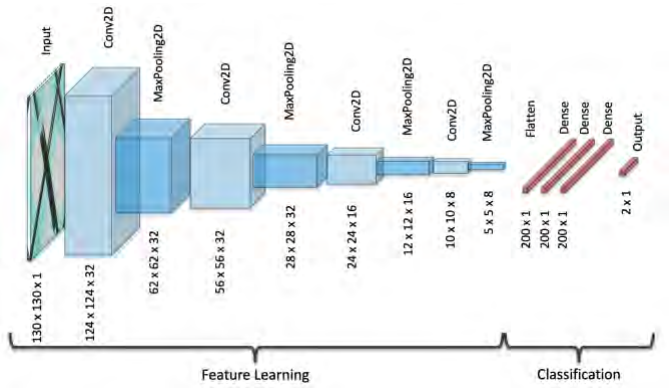


Fig. 5. Architecture of the used CNN with designation of the layers as well as the respective tensor size

By considering the feature maps, a number of 32, 32, 16 and 8 feature maps have been found to be suitable for the different convolutional layers. This involves initially extracting larger features (7x7 pixels) in the first two layers, following which increasingly smaller features (5x5 and 3x3 pixels) are extracted as the image size decreases. A stride of 1 and valid padding is used. For the training process, the Adam Optimizer is used with a dynamic learning rate, 100 epochs and a batch size of 10 images. Since this is a classification problem, an accuracy-based optimization is performed.

V. Results

During the training process, the learning curve characteristic for machine learning algorithms emerges, in which the accuracy asymptotically approaches a steady-state value. After about 60 epochs, this value is 98%. However, this value is not very meaningful because there are numerous, slightly modified copies of the original image sections in the training data set due to data augmentation.

Since this is a typical test procedure with the possible results being positive or negative, a four-field table can be used to evaluate the test data set. The specific results are shown in TABLE *I*.

TABLE I Four-field table of the distribution of results

		Predictor (CNN)		
		Good	***Cut***	***Sum:***
Criterion (Class)	***Good***	154 (True Negative)	10 (False Positive)	164
	Cut	14 (False Negative)	63 (True Positive)	77
	Sum:	168	73	241

Sensitivity:	81,82%
Specificity:	93,90%
Positive Predictive Value:	86,30%
Negative Predictive Value:	91,67%
Hit rate:	90,04%

When looking at the incorrectly classified images, it is noticeable that a significant proportion of them consists of images that cannot be explicitly classified. This is the case, for example, when the location of a cut leads to a distortion of the braiding pattern of the surrounding image areas. Although these image areas indicate the occurrence of a cut in the immediate vicinity, they do not show it directly.

To avoid distortion of the results by these types of images, all images that cannot be explicitly classified are removed from the test data set. The corrected data can be seen in TABLE *II*.

TABLE II. FOUR-FIELD TABLE WITH CORRECTED RESULT DISTRIBUTION AFTER SUBTRACTION OF THE AMBIGUOUSLY DEFINED IMAGE SECTIONS

		Predictor (CNN)		
		Good	***Cut***	***Sum:***
Criterion (Class)	***Good***	139 (True Negative)	5 (False Positive)	144
	Cut	10 (False Negative)	62 (True Positive)	72
	Sum:	149	67	216

Sensitivity:	86,11%
Specificity:	96,53%
Positive Predictive Value:	92,54%
Negative Predictive Value:	93,29%
Hit rate:	93,06%

VI. CONCLUSION AND OUTLOOK

Although the achieved accuracy values do not yet correspond to a production-ready result, they prove that it is possible to detect cracks in stents by combining classical image processing and machine learning. Since the dataset used is still very small compared to other popular deep learning databases (e.g. MNIST), an extension of the dataset holds a promising opportunity to further improve the accuracy. By shifting the sensitivity of the testing procedure, the rate of false negative classifications could be reduced to the point where the procedure provides sufficient confidence for the particularly high demands. However, the possibility of this procedure completely replacing humans at this point is unlikely. For reasons of economy, image sections classified as cut should still be checked by humans for false positives. It is therefore not so much a fully autonomous inspection procedure, but rather an assistance system designed to direct the employee's attention to potential fault points. The latter then makes the decision as to whether it really is a faulty stent. The amount of work required to inspect a stent is thus considerably reduced.

REFERENCES

[1] 'InStent'. http://www.instent.de/ (accessed Jan. 12, 2021).

[2] E. Lange, 'Automatisierung der optischen Inspektion von Einfadenstents', p. 102, 2017.

[3] P. Zschech, C. Sager, P. Siebers, and M. Pertermann, 'Mit Computer Vision zur automatisierten Qualitätssicherung in der industriellen Fertigung: Eine Fallstudie zur Klassifizierung von Fehlern in Solarzellen mittels Elektrolumineszenz-Bildern', *Prax. Wirtsch.*, Jul. 2020, doi: 10.1365/s40702-020-00641-8.

[4] E. Alpaydin, *Maschinelles Lernen*, 1st ed. München: Oldenbourg Wissenschaftsverlag, 2008.

[5] A. Burkov, *Machine Learning kompakt: Alles, was Sie wissen müssen*. Frechen, GERMANY: mitp, 2019. Accessed: Oct. 13, 2020. [Online]. Available: http://ebookcentral.proquest.com/lib/sulb/detail.action?docID=5808518

[6] T. Guo, J. Dong, H. Li, and Y. Gao, 'Simple convolutional neural network on image classification', *2017 IEEE 2nd Int. Conf. Big Data Anal. ICBDA*, 2017, doi: 10.1109/ICBDA.2017.8078730.

[7] E. Alpaydin, *Maschinelles Lernen*, 2., Erweiterte Auflage. Berlin ; Boston: De Gruyter Oldenbourg, 2019.

[8] T. Dietterich, 'Overfitting and undercomputing in machine learning', *ACM Comput. Surv.*, vol. 27, no. 3, pp. 326–327, Sep. 1995, doi: 10.1145/212094.212114.

[9] S. S. Du, J. D. Lee, H. Li, L. Wang, and X. Zhai, 'Gradient Descent Finds Global Minima of Deep Neural Networks', p. 11.

[10] F. van Veen, 'The Neural Network Zoo', *The Asimov Institute*, Sep. 14, 2016. https://www.asimovinstitute.org/neural-network-zoo/ (accessed Nov. 30, 2020).

[11] S. Selle, 'Künstliche Neuronale Netzwerke und Deep Learning', Saarbrücken, May 12, 2018. [Online]. Available: https://www.htwsaar.de/htw/wiwi/fakultaet/personen/profile/selle-stefan/Selle2018e_Kuenstliche_Neuronale_Netzwerke.pdf

[12] M. Ziegler, 'Object Recognition with Convolutional Neural Networks', p. 15, 2018.

[13] K. Chatfield, K. Simonyan, A. Vedaldi, and A. Zisserman, 'Return of the Devil in the Details: Delving Deep into Convolutional Nets', *ArXiv14053531 Cs*, Nov. 2014, Accessed: May 22, 2021. [Online]. Available: http://arxiv.org/abs/1405.3531

Development of a Real-time Observation System to Track Body/Hand Gestures for Industrial Robot Using Azure Kinect Camera

1st Prof. Dr.-Ing. Rainer Müller
Chair of Assembly Systems
ZeMA-Zentrum für Mechatronik und Automatisierungstechnik gGmbH
Saarbrücken, Germany
rainer.mueller@zema.de

2nd Dr.-Ing. Ali Kanso
Chair of Assembly Systems
ZeMA-Zentrum für Mechatronik und Automatisierungstechnik gGmbH
Saarbrücken, Germany
a.kanso@zema.de

3rd Hussein Sheaib
Chair of Assembly Systems
ZeMA-Zentrum für Mechatronik und Automatisierungstechnik gGmbH
Saarbrücken, Germany
h.sheaib@zema.de

4th Xiaomei Xu
Chair of Assembly Systems
ZeMA-Zentrum für Mechatronik und Automatisierungstechnik gGmbH
Saarbrücken, Germany
xiaomei.xu@zema.de

***Abstract*—Human Robot Collaboration (HRC) applications face challenges in implementing a central system that controls the flow of the tasks set to achieve a certain goal as well as the safety measures to avoid unnecessary overlaps between the human and the robot that may lead either to injuries or to technical malfunctions. This paper presents an observation system bases on 3D camera system and the robot controller. Therefore, an optimization of the work flow is achieved while maintaining the safety measures. The camera system detects and observes the 3D pose of LEGO part, while the robot controller is responsible for the execution of the robot tasks and the commands set by the observation system.**

I. Introduction

Smart manufacturing is a main concept that the fourth industrial revolution is built upon [1] [2], as well as the introduction of three features: horizontal integration through value networks, end-to-end digital integration of engineering, and vertical integration [3]. This integration is dependent on advances and breakthroughs in digital technologies and automation. One of the main features of this integration into a smarter manufacturing and production system is human robot collaboration, where humans and robots can work together and create a shared workspace. Simultaneously, the system benefits from the advantages of robots in terms of accuracy and speed, as well as the flexibility and skills of a human. For example, a robot observes a human in an assembly operation to learn the assembly tasks sequence [4]. However, there are some challenges that need to be targeted to provide an efficient collaborative solution [5]: safe measures must be provided by a observation system to avoid any robot human collisions, as well as techniques that benefit from the cognitive skills of the human such as understanding gestures expressed by the body or the hand of a human.

This paper attempts to apply a validation scenario for body tracking, hand pose estimation, and 3D position estimation concepts, which are integrated into an observation system. A convolutional neural network is used to track body motion by building the body nodes and connecting them to simplify the body structure [6]. Moreover, the hand pose estimation is developed to recognise the numbers that different gestures are trying to express [7]. Body tracking and hand pose estimation are used to track the human body movement and understand the hand pose expression. The Azure Kinect camera integrates two interesting sensors: a depth camera that supports time-of-flight (ToF) technology, and a color camera of 12 megapixels with a maximum resolution of 3840x2160. Furthermore, the experimental data show that the depth of the Azure Kinect Camera is highly correlated with the color of the LEGO part. Delta X and Delta Y (the difference between the desired X and Y coordinates and the actual X and Y coordinates) show a significant correlation with the image pixel coordinate system. This paper discusses the different mathematical methods to reduce the error of the 3D position calculation.

A simple use case will verify the practicability of the observation system. The robot arm is configured with a camera to observe whether the LEGO components are placed in areas of the same color. In the event of detecting a human, the system will automatically recognise the human body and send a command to stop the robot from working. After the human interaction such as swapping, removing, adding, and changing position, the workspace is scanned again and the LEGO parts are sorted by the robot to the correct area.

II. State of The Art

Real-time observation systems (or surveillance systems) can have different deployments. For example, in [8] a real-time observation system is implemented where a robot continuously observes the behaviour of a human for it to be able to recognize gestures from people and turn this robot to a social robot. Likewise, [9] a real time face re-identification system is developed to achieve effective interactions between robots and humans, i.e. to be able to build and update the run-time user database in a low latency model where it should get high recognition accuracy. Moreover, [10] implements a real-time collision avoidance system that is composed by three parts: perception of the environment, collision avoidance algorithm, and robot control. An approach is developed to calculate the respective distances between the robot and the obstacles in the workspace from the depth data of the Microsoft Kinect

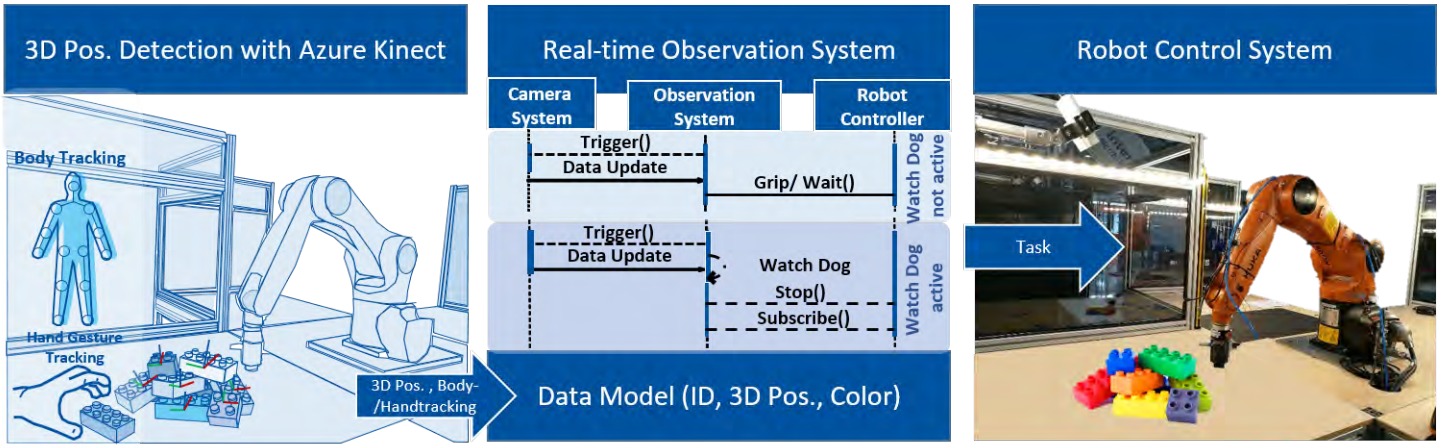


Fig. 1. Overview of the observation system

Camera. The distances between the robot and the obstacles (humans included) are required to generate repulsive commands for the robot to avoid collisions.

There are different approaches for body tracking, especially regarding whether the algorithm follows a top-down or a buttom-up approach, i.e. whether the algorithm detects the humans in the image and then start predicting the body parts of each human or it detects a body part and then predicts the other body parts to get the whole human. [11] builds on the pictorial structure model (PSM) [12] and extends it with a "mixture of trees" model by partitioning the space of human pose into clusters, which can be modeled and analyzed in a simpler manner. A top-down approach is used by modelling a Mask R-CNN, where first humans are detected and then **K** masks are predicted, one for each body keypoint, e.g. left shoulder or neck [13]. On the other hand, [6] takes a bottom-up method that uses part affinity fields to associate between the different body parts belonging to the same human.

In [14], the hand pose estimation problem is presented as an image database indexing problem, i.e. the database is searched for the closest match for an input hand image. On the other hand, [15] uses the depth data of the Kinect Camera to segment the hand region. After building a 3D hand model that gives the hand a 21 part representation, a random decision forest is trained on depth images. Each pixel is classified to a hand part and then this classification is passed to a local mode finding algorithm that estimates the joint locations of the hand. [7] uses the depth data of the Kinect Camera to extract the hand region out of the frame and then with the help of functions that output specific contours and values, the number of extended fingers are estimated.

Techniques to find the 3D position of an object vary according to the system used, e.g. a single RGB camera, a depth camera, a stereo camera system, or an industrial camera. [16] proposes a method to find the 3D position of a flying ball using triangular geometric relations, thus following a sequence where at first they compute the projection matrix and then detect the object of interest and finally estimate the 3D position of that object. [17] uses two deep learning models, one that extracts the images information, and another that extracts the temporal information for better prediction. Building a pipeline for both neural networks where the input of the "temporal network" is the output of 25 "image networks". This pipeline is able to predict the 3D position of an object that may be in trajectory.

III. Concept

A. Overview

The observation system is implemented to communicate between the camera system and the robot control system in a way where the observation system sends commands and triggers to either systems and in return receives feedback from them. As demonstrated in "Fig. 1", the observation system consists of two phases, when "Watchdog" mode is activated and when not activated. When the "Watchdog" mode is not activated, the system starts by triggering the camera system to scan the environment. The camera system then detects the LEGO parts and their 3D position as well as it keeps monitoring if any hand or body is present near to the workspace. Afterward the observation system sends a command to the robot control system, e.g. to grip the LEGO parts to a certain position. At the moment a hand or body is detected by the camera system, the observation system moves into the "Watchdog" mode and the observation system sends a STOP command to the robot control system and waits its feedback that it stopped. After the hand or body leave the environment, the "Watchdog" mode is deactivated and the camera system is again triggered to scan the environment and update the database.

B. Body Tracking

OpenPose (Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields) [6] adopts a buttom-up system that in return achieves high accuracy and real-time performance. The method first estimates the body locations and then with each stage it refines the part affinity fields (PAFs) which are needed to be able to learn to associate body parts with

individuals in the image. The input of the model is an image of size w*h and the output is a 2D locations of keypoints for each individual in the image. This process goes through three stages: First, the first ten layers of the VGGNet (VGGNet is a Convolutional Neural Network architecture) are used to generate feature maps **F** for the input image.

Fig. 2. Example of a confidence map and part affinities [6]

Then, a two branched CNN is used, the first branch is responsible for the prediction of a set of 2D confidence maps **S** of body part locations, e.g. neck, elbow, shoulder, etc.. The second branch is responsible for the prediction of a set of 2D vector fields **L** of part affinities which encode the degree of association between the body parts. Set **S** has J confidence maps and set **L** has C vector fields. An example of confidence maps and part affinities is shown in "Fig. 2". Finally, the confidence maps and the vector fields are parsed to output the 2D keypoints for the people present in the image. Since this method relies on confidence maps to detect the body parts and then for vector fields to associate between them, it is able to detect body parts even the whole body is not visible in the image. The trained Tensorflow model [1] used in this project outputs the 2D keypoints for a single individual. Moreover, this model uses Tesnorflow Mobilenet Model instead of the CaffeModel which reduces its size from 200MB to 7MB.

C. Hand Pose Estimation

The hand pose estimation approach used in this project is inspired by [7]. The algorithm is based on three stages: hand region segmentation, hand shape analysis and hand gesture recognition. With the hand region segmentation, the human's hand region will be extracted in every frame with the help of the depth map output of the Kinect sensor. Then, the shape of the segmented hand region is analyzed by getting contours, convex hull and convexity defects. Finally, by analyzing the convexity defects, the number of extended fingers is determined.

In this project the PY4KA [2] is needed to deploy the Azure Kinect Camera sensors and functions. The first step is performed by a function that takes in the depth map and returns the segment of the hand region. In the second step we get the largest contour found in the segment as well as the convexity defects in it if any are detected. To get the max contour within the segment, the OpenCV function *findContours()* is used, also to get the convex hull that envelopes the contour area, the OpenCV function *convexHull()* that takes in the max contour and gives back the convex hull. Finally, the OpenCV function *convexityDefects()* that takes in both the max contour and the convex hull to give back the convexity defects, i.e. the points farthest away from the convex hull and logically caused by the wrist and the fingers.

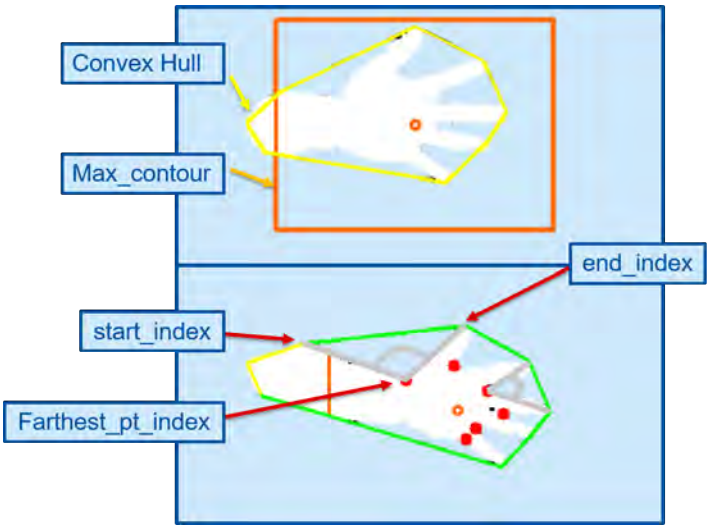


Fig. 3. Values to analyze hand region [7]

A convexity defect is an array of four integers. Relative to us are three of these four integers, namely the *start_index* (the index of the point at the beginning of the defect), the *end_index* (the index of the point at the end of the defect), and the *farthest_pt_index* (the point with the farthest distance away from the convex hull within the defect). The tip to find the number of extended fingers now is two form an angle within each defect between the *farthest_pt_index* and both the *start_index* and the *end_index*. These values are shown in "Fig. 3".

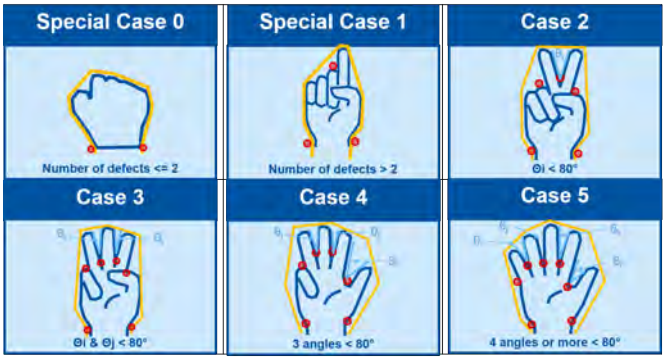


Fig. 4. Estimation of extended fingers

As shown in "Fig. 4", if the angle within each defect is smaller than a predefined threshold angle (we set it at 80°),

[1]https://github.com/quanhua92/human-pose-estimation-opencv
[2]https://github.com/etiennedub/pyk4a

then we assume that this convexity defect is located between two fingers, otherwise it is not. There are two special cases: when the number of defects is less or equal to 2, we predict that we are showing a closed fist and thus we get a 0 back. If the number of defects is greater than 2 and the none of the angles satisfy the condition, we predict that we are showing number 1. Then with each angle that satisfies the condition we add 1 to the number of extended fingers, with a maximum number of 5.

D. 3D Position Estimation

1) Experiment: The experiment consists of setting 15 LEGO parts of a certain color at 15 specific locations on the workspace. Then we move the robot to the center of each LEGO part to save the 3D position of the each LEGO part in the robot coordinate system.

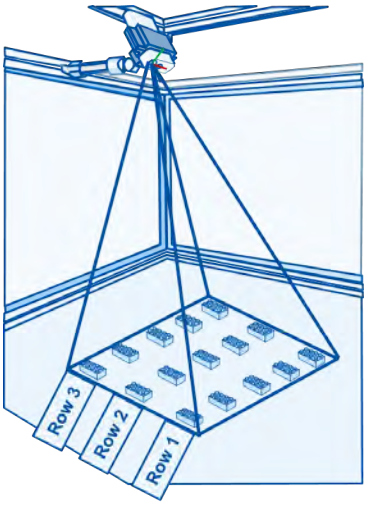


Fig. 5. The layout of the experiment.

Then we capture the scene with the Azure Kinect Camera in order to save the depth values in a JSON file and the color image in PNG format. The color image is of resolution 1280x720 and the depth image is of 640x576. The field of view of the depth sensor is different than that of the RGB sensor. That means that a pixel coordinate of the same point varies between the two sensors. In order to align the depth values to the RGB field of view, we use a function called *transformed_depth* from the PYK4A library. The color image is used to get the pixels of the center of each LEGO part, and the depth values are used to get the depth of that exact location. [7]

$$\begin{pmatrix} u_p \\ u_v \\ 1 \end{pmatrix} = \begin{pmatrix} f_x & 0 & u_0 \\ 0 & f_y & v_0 \\ 0 & 0 & 1 \end{pmatrix} * \begin{pmatrix} {}^{C}X_p/{}^{C}Z_p \\ {}^{C}Y_p/{}^{C}Z_p \\ 1 \end{pmatrix} \quad (1)$$

The matrix in "(1)" is called the camera matrix and we obtain it by calibrating the camera with a chessboard. We set a 9x7 chessboard at 21 different poses and we capture these different scenes with the camera set at a 1280x720 resolution then we input these images as well as the chessboard dimension and the edge length of a square in the chessboard into the camera calibration program. The program then outputs the aforementioned camera matrix, the distortion coefficients, the translational vector, and the rotational matrix. We assume that ${}^{C}Z_p$ is exactly the depth value captured by the depth sensor, and since we already have u_p and u_v from the color image, the only unknowns left are ${}^{C}X_p$ and ${}^{C}Y_p$. After calculating the unknowns and in order to verify the accuracy of our calculations we transform the robot coordinates to the camera coordinate system by multiplying them by the transformation matrix ${}^{C}T_B$ as shown in "(2)".

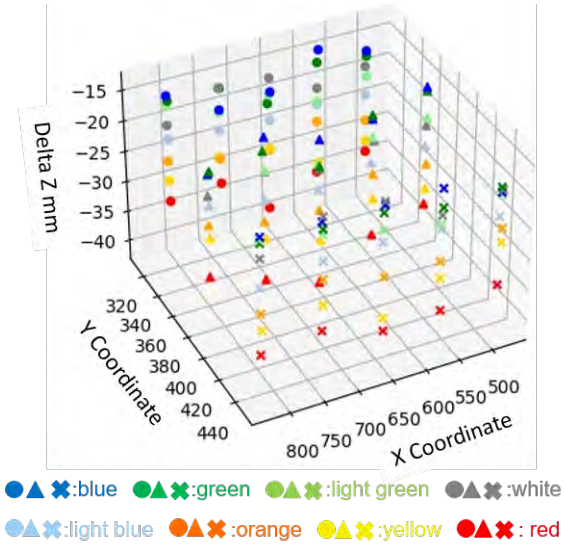


Fig. 6. Plot of Delta Zs

$$\begin{pmatrix} {}^{C}X_p \\ {}^{C}Y_p \\ {}^{C}Z_p \\ 1 \end{pmatrix} = {}^{C}T_B * \begin{pmatrix} {}^{B}X_p \\ {}^{B}Y_p \\ {}^{B}Z_p \\ 1 \end{pmatrix} \quad (2)$$

We assume that these transformed coordinates are the target values and that the calculated coordinates through "(2)" are the actual values in order to check the difference and understand how accurate is our approach.

From data in "Fig. 6", it is obviously clear how depth is highly dependent on the color. In each position the lowest Delta Z is between -15mm and -21mm and it belongs to blue LEGO parts. The sequence of the increase of Delta Z then goes as follows: $blue \rightarrow green \rightarrow lightgreen \rightarrow white \rightarrow lightblue \rightarrow orange \rightarrow yellow \rightarrow red$. The highest error in Z direction varies between -35mm and -40mm and belongs to red LEGO parts. This indicates the necessity for finding an approach to minimize Delta Z.

The general reasons for the inaccuracy of the depth sensor is the noise from the environment. For example, if the pixels are not located in the IR illumination mask, then the pixels are invalidated and have zero depth values. Moreover, if a pixel contains a saturated infrared signal then it also gets invalidated since the phase information is lost. In the same logic, if the

Fig. 7. Reasons for the inaccuracy of the depth sensor [18] [19]

strength of the infrared signal is not strong enough to generate depth, then the pixels at these positions are also invalidated. In addition to the factor of reflectivity, since the depth sensor follows the technology of flight then the reflectivity of the object is critical to the depth estimation.

The main reasons for the inaccuracy of the position estimation:

- Depth sensor and RGB sensor have different coordinates systems.
- Depth sensor and RGB sensor have different resolutions, which will influence the conversion from the depth image to RGB image.
- Depth sensor and RGB sensor have blink spots when converted to each other.

2) Weight Function: To minimize Delta Z, we use a weight function where the previous values of Delta Zs are saved in our database and the Z value of any new LEGO part is fixed with the weight function. First, we get the pixels of the center of the LEGO part as well as the color of that part. Second, we get the depth value of the center of the LEGO part.

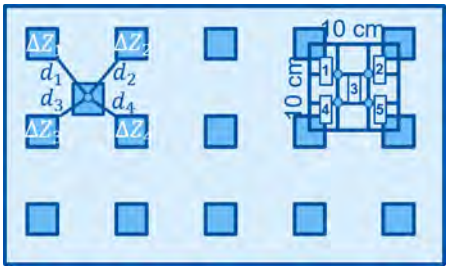


Fig. 8. Weights proportional to the distances

Then, we search our database for the four nearest neighbors and we set a weight for the Delta Z of each neighbor depending on how close is it to the new LEGO part.

$$\Delta Z = w_1 \cdot \Delta Z_1 + w_2 \cdot \Delta Z_2 + w_3 \cdot \Delta Z_3 + w_4 \cdot \Delta Z_4 \quad (3)$$

Using "(3)" we calculate the Delta Z that in return corrects the Z value of the new LEGO part. We set the new LEGO part at 5 different locations (See "Fig. 8"): [1: TOPLEFT, 2: TOPRIGHT, 3: CENTER, 4: LOWLEFT, 5: LOWRIGHT]. With this approach we minimize Delta Z from 15mm-40mm to -3.08mm-4.69mm.

3) Correction by Trendline: The plots of both Delta X and Delta Y show a possibility to calculate a trendline for the values of these Deltas for each color in each row. (see "Fig. 9")

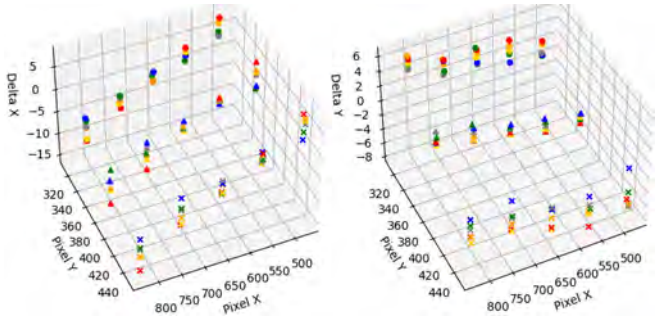


Fig. 9. Plots of Delta X and Delta Y

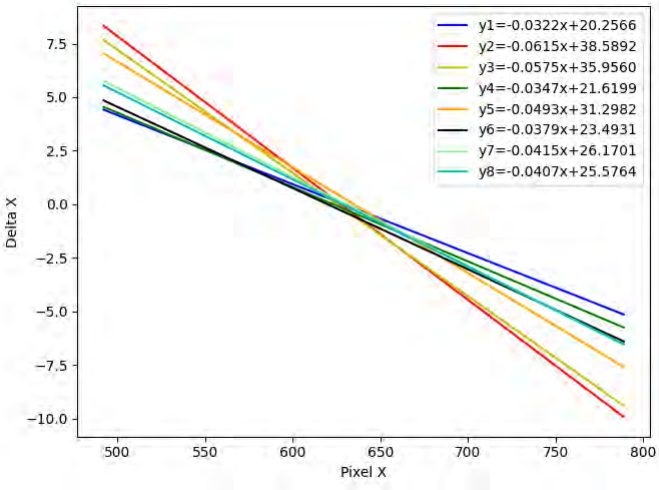


Fig. 10. Trendlines of Delta X in row 1

"Fig. 10" shows the trendlines of Delta X for every color in the first row (see "Fig. 5"). First, we assume that the ratio ${}^{C}X_p/{}^{C}Z_p$ is equal to a variable a, where $a_{target} = {}^{C}X_{p\_target}/{}^{C}Z_{p\_target}$ and $a_{actual} = {}^{C}X_{p\_actual}/{}^{C}Z_{p\_actual}$. "(4)" is dissected from the first row of "(1)".

$$\frac{{}^{C}X_p}{{}^{C}Z_p} = \begin{pmatrix} \frac{1}{f_x} & 0 & -\frac{u_0}{f_x} \end{pmatrix} \cdot \begin{pmatrix} u_p \\ u_v \\ 1 \end{pmatrix} \quad (4)$$

Then with some mathematical operations we arrive at:

$${}^{C}Z_{p\_target} = \frac{{}^{C}X_{p\_actual} - \Delta {}^{C}X_p}{a_{target}} \quad (5)$$

"(5)" shows that the target Z value is equal to the corrected X value divided by the a_{target}. This approach did not succeed to minimize Delta Z at all. Reasons were that a general trendline that combines the three rows for each color was not found and with each capture the trendline would have a slight change that deviates the actual value away from the desired value [19]. So regarding the general trendline obstacle, we had an idea where we cover each possible row in the workspace, in a way that eliminates the necessity for a general trendline. However, the results were still far away from the target.

E. Image Processing

Fig. 11. Contour extraction process

1) Contour Detection: As shown in "Fig. 11", the color image is converted to the HSV colormap (Hue Saturation Value). The HSV parameters are manually set so that the colors: blue, green, yellow, red, orange, light blue, and light green are detected. The *findContours()* function of OpenCV with these two modes *RETR_EXTERNAL* and *CHAIN_APPROX_SIMPLE* gets the contours by extracting the corner points of the image.

2) Shape Detection: An erosion mask of 5x5 kernel is used to reduce noise from the image. Then, to detect the shape and size of the LEGO part the width and height values of the bounding box of the contours are used to decide if the following contour is a square or medium rectangle or a big rectangle to get a good estimation for the midpoint of the contour.

F. Database

The database model is implemented with SQLite. A "LEGOS" table is created with the following entries: [ID, x, y, z, a, b, c, ColorOfTheSortArea, IsSorted]. ID is the primary key to each LEGO part, 'x,y,z' are the 3D position values of the LEGO part, 'a,b,c' represent the orientation parameters (roll, pitch, and yaw) of the LEGO part, ColorOfTheSortedArea specifies which area should the LEGO part be at and IsSorted is whether the LEGO part is already sorted at the correct color area or not. As mentioned before, the observation system updates upon triggering the camera system the database with the changes.

IV. Application

A. Experimental Setup

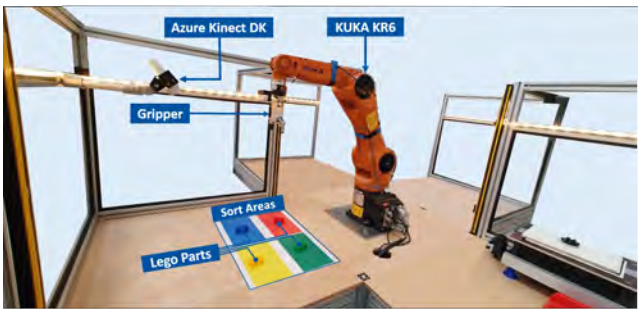


Fig. 12. Experimental setup of the Kinect Camera, KUKA KR6 Robot, and LEGO parts

The Azure Kinect Camera is set on the side bar as shown in "Fig. 12" and the LEGO parts are placed in the workspace. A flexible two finger gripper is used to grip LEGO parts and set them into desired positions. A human can appear from the left side of the robotic arm.

B. Body Tracking

Fig. 13. Body tracking keypoints

Facing the camera, the body tracking approach is tested. First, to check if all body keypoints are detected when the whole body is visible, and second to check if visible body keypoints are detected when part of the body is only available in the image. As shown in "Fig. 13", both scenarios are validated since the in the left image the full set of body keypoints is shown. On the right side and even though only about 50% of the body is visible, the keypoints are still correctly detected.

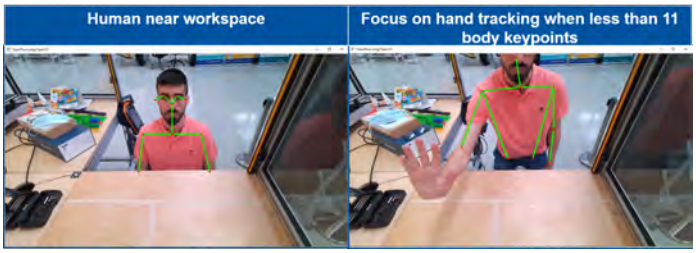


Fig. 14. Body keypoints at close range

C. Hand Pose Estimation

The hand pose estimation approach is tested by showing the camera different representations of different numbers and checking if they are correctly predicted. The method yields good results as shown in "Fig. 15".

D. Observation System

The previous approaches are combined with the 3D position estimation approach in one program. At the beginning, the camera detects the LEGO parts and then if a body is detected the system starts tracking the body. If the body keypoints visible become less than 11, we assume that the body is too close to the workspace and that the human wills to show a hand gesture, so the system moves to detect the number of extended fingers. In "Fig. 14", it is seen that when a human is close enough to the workspace to give a hand gesture that in fact less than 11 keypoints will be visible. Then after the human

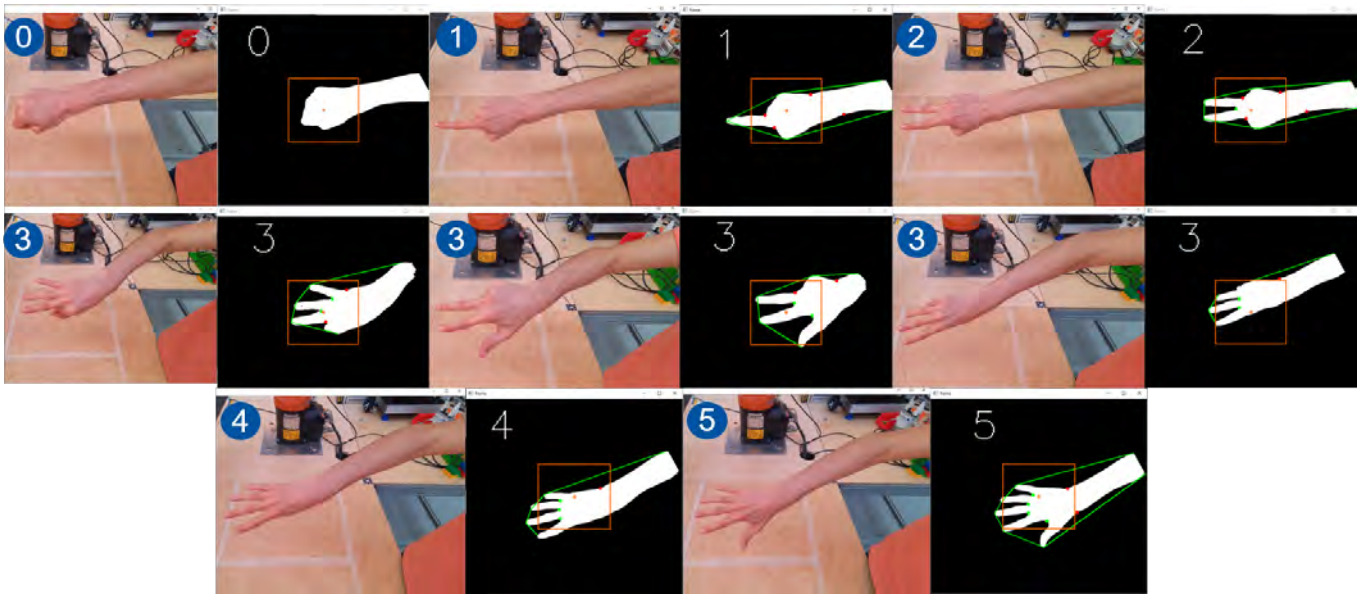


Fig. 15. Hand pose estimations

leaves the workspace, the camera then detects the LEGO parts again and saves the data in the database. We have four sort area colors: red, blue, yellow, and green. Red and orange LEGO parts should be sorted to the red sort area, blue and light blue to the blue sort area, yellow to the yellow sort area, and green and light green to the green sort area. These sort areas are seen in "Fig. 12" with the appropriate LEGO parts positioned on them. The 3D position of the midpoint of each LEGO part is calculated and then corrected with the weight function. Since the contours are a bit smaller than the actual LEGO parts and since they do not possess the same sharp edges, the midpoint calculated in pixels may not be exactly at the middle of the LEGO part. Moreover, the algorithm was able to correctly save the LEGO parts in the SQLite database and was able to detect the color of the LEGO part and to sort it at the correct sort area.

V. Summary

The paper represents a real time observation system, which consists of body tracking, hand pose tracking, and 3D position estimation. The camera system detects 3D positions of objects of interest in the workspace and at the same time observes body or hand presence in the workspace. Upon the presence of a body or hand, the observation system asks the robot to stop if it is performing a task. When the human leaves the workspace, the camera system updates the database in real time. The observation system specifies the required tasks from the robot. The depth data of the Azure Kinect Camera are used in this paper in the methods to estimate hand poses and to estimate 3D positions of objects of interest.

Points of weakness and limitations will be addressed in future work. The 3D position estimation accuracy of the different coordinates between the RGB sensor and the depth sensor is still limited. The transformed depth image that transforms the depth values to the RGB camera point of view should be improved. Additionally, more work will be put on getting a more accurate estimation of the midpoints of the LEGO parts. Moreover, the body tracking and hand poses will be converted to commands that the observation system then sends them either to the camera system or to the robot control system.

In the future the observation system will be used in a use case and will be combined with the digital twin tools with AI to build an intelligent robotic system. The intelligent robotic system will focus on how the robot understands an existing malfunction and adapts itself to the environment to solve the problem based on the existing condition.

References

[1] H. Kagermann, J. Helbig, A. Hellinger, and W. Wahlster, *Recommendations for Implementing the Strategic Initiative INDUSTRIE 4.0: Securing the Future of German Manufacturing Industry ; Final Report of the Industrie 4.0 Working Group*. Forschungsunion, 2013.

[2] E. Manavalan and K. Jayakrishna, "A Review of Internet of Things (IoT) Embedded Sustainable Supply Chain for Industry 4.0 Requirements," *Computers & Industrial Engineering*, vol. 127, pp. 925–953, 2019.

[3] S. Wang, J. Wan, D. Li, and C. Zhang, "Implementing Smart Factory of Industrie 4.0: An Outlook," *International Journal of Distributed Sensor Networks*, vol. 12, no. 1, p. 3159805, 2016.

[4] S. Ji, S. Lee, S. Yoo, I. Suh, I. Kwon, F. C. Park, S. Lee, and H. Kim, "Learning-based automation of robotic assembly for smart manufacturing," *Proceedings of the IEEE*, vol. 109, no. 4, pp. 423–440, 2021.

[5] V. Villani, F. Pini, F. Leali, and C. Secchi, "Survey on human-robot collaboration in industrial settings: Safety, intuitive interfaces and applications," *Mechatronics*, vol. 55, pp. 248–266, 2018.

[6] Z. Cao, T. Simon, S.-E. Wei, and Y. Sheikh, "Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields," in *CVPR*, 2017.

[7] M. B. Menua Gevorgyan, Arsen Mamikonyan, *OpenCV with Python Blueprints*. Packt Publishing Ltd., 2020.

[8] A. S. Phyo, H. Fukuda, A. Lam, Y. Kobayashi, and Y. Kuno, "A human-robot interaction system based on calling hand gestures," in *Intelligent Computing Methodologies* (D.-S. Huang, Z.-K. Huang, and A. Hussain, eds.), (Cham), pp. 43–52, Springer International Publishing, 2019.

[9] Y. Wang, J. Shen, S. Petridis, and M. Pantic, "A real-time and unsupervised face re-identification system for human-robot interaction," *Pattern Recognition Letters*, vol. 128, pp. 559–568, 2019.
[10] F. Flacco, T. Kröger, A. De Luca, and O. Khatib, "A depth space approach to human-robot collision avoidance," in *2012 IEEE International Conference on Robotics and Automation*, pp. 338–345, 2012.
[11] S. Johnson and M. Everingham, "Clustered pose and nonlinear appearance models for human pose estimation," in *Proc. BMVC*, pp. 12.1–11, 2010. doi:10.5244/C.24.12.
[12] P. Felzenszwalb and D. Huttenlocher, "Pictorial structures for object recognition," *International Journal of Computer Vision*, vol. 61, pp. 55–79, 01 2005.
[13] K. He, G. Gkioxari, P. Dollár, and R. B. Girshick, "Mask R-CNN," *CoRR*, vol. abs/1703.06870, 2017.
[14] V. Athitsos and S. Sclaroff, "Estimating 3d hand pose from a cluttered image," in *2003 IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2003. Proceedings.*, vol. 2, pp. II–432, 2003.
[15] C. Keskin, F. Kıraç, Y. E. Kara, and L. Akarun, *Real Time Hand Pose Estimation Using Depth Sensors*, pp. 119–137. London: Springer London, 2013.
[16] T. Kim, Y. Seo, and K.-S. Hong, "Physics-based 3d position analysis of a soccer ball from monocular image sequences," in *Sixth International Conference on Computer Vision (IEEE Cat. No.98CH36271)*, pp. 721–726, 1998.
[17] F. Pedrazzini, "3d position estimation using deep learning,"
[18] Microsoft, "Azure kinect dk depth camera."
[19] M. Tölgyessy, M. Dekan, L. Chovanec, and P. Hubinský, "Evaluation of the azure kinect and its comparison to kinect v1 and kinect v2," *Sensors*, vol. 21, p. 413, 01 2021.

MaRIO - A Cost-Effective Solution for Human-Robot Interaction

Henrik Schu
Sustainable Production Processes and Industrial Robotics
Trier University of Applied Sciences, Environmental Campus Birkenfeld
Birkenfeld, Germany
s17c71@umwelt-campus.de

Jonathan Brun
Ecole Nationale Supérieure d'Ingénieurs Sud Alsace (ENSISA)
Université de Haute-Alsace (UHA),
France
jonathan.brun@uha.fr

Romain Gondé
Ecole Nationale Supérieure d'Ingénieurs Sud Alsace (ENSISA)
Université de Haute-Alsace (UHA)
France
romain.gonde@uha.fr

Maxime Kieffer
Ecole Nationale Supérieure d'Ingénieurs Sud Alsace (ENSISA)
Université de Haute-Alsace (UHA)
France
maxime.kieffer@uha.fr

Allister Pereira
Sustainable Production Processes and Industrial Robotics
Trier University of Applied Sciences, Environmental Campus Birkenfeld
Birkenfeld, Germany
s18a57@umwelt-campus.de

Kevin Fuchs
Sustainable Production Processes and Industrial Robotics
Trier University of Applied Sciences, Environmental Campus Birkenfeld
Birkenfeld, Germany
s17af7@umwelt-campus.de

Matthias Vette-Steinkamp
Sustainable Production Processes and Industrial Robotics
Trier University of Applied Sciences, Environmental Campus Birkenfeld
Birkenfeld, Germany
m.vette-steinkamp@umwelt-campus.de

Lars Schaupeter
Sustainable Production Processes and Industrial Robotics
Trier University of Applied Sciences, Environmental Campus Birkenfeld
Birkenfeld, Germany
l.schaupeter@umwelt-campus.de

Rodolfo Orjuela
Institut de Recherche en Informatique, Mathématiques, Automatique et Signal (IRIMAS) UR7499
Ecole Nationale Supérieure d'Ingénieurs Sud Alsace (ENSISA)
Université de Haute-Alsace (UHA)
France
rodolfo.orjuela@uha.fr

Jean-Philippe Lauffenburger
Institut de Recherche en Informatique, Mathématiques, Automatique et Signal (IRIMAS) UR7499
Ecole Nationale Supérieure d'Ingénieurs Sud Alsace (ENSISA)
Université de Haute-Alsace (UHA)
France
jean-philippe.lauffenburger@uha.fr

***Abstract*—The availability of open-source as well as cost-effective solutions is ever increasing as time progresses. MaRIO, a robot capable of human-robot interaction, was initially intended for the purpose of adherence to COVID-19 measures regarding the wearing of masks in closed spaces. However, the system can also be seen as an example of how a robot can be constructed within a very tight budget and still perform a variety of tasks. This paper also aims to describe how the developmental process of MaRIO can be implemented using open-source tools, while shedding further light on the current state of the project and further objectives. These objectives are aimed at further developing the platform while also ensuring that the robot remains a cost-effective solution for tasks that it would be later capable of handling.**

***Keywords*—Human-robot interaction, open-source, image processing, machine learning, deep learning**

I. INTRODUCTION

The subject of robotics has taken great strides in recent decades and continues to hold a high degree of relevance. Today, that same subject has been the focus of research in a variety of fields, with one of these, namely human-robot interaction, also garnering considerable attention [1, 2].

Human-robot interaction (HRI) may be defined as a field of robotics dedicated towards robot systems that are directly used by or work alongside humans. Such a purpose thereby also requires communication between human and robot, which can be achieved either through remote (such as robots for explosive ordnance removal or EOD [3]) or proximate (such as the well-known Pepper from Softbank Robotics [4]) means [1]. This communication also includes understanding whose turn it would be to respond to a stimulus [5]. Effective communication, being a must in the case of HRI, can take place through a number of methods which include verbal cues, gestures, prosody and gaze among others [2, 5–7].

The term "open-source" is widely used in today's day and age, and for good reason. The existence of open-source solutions has

enabled even the average human being to access, design, develop and implement important information and technologies [8]. Furthermore, it also enables the dissemination of information among a greater community of individuals working together on a specific topic. This results in a wider availability of solutions and greater participation in discussion towards a working solution [9]. Open-source products see application in a variety of applications and continue to hold relevance due to their numerous advantages [8].

The robot for Mask Recognition and Interaction for Order (MaRIO) is a result of the integration of the concept of human-robot interaction with open-source solutions. Through the use of hardware such as the Raspberry Pi and Arduino as well as image processing software such as OpenCV in combination with Python libraries and a deep learning algorithm for mask detection, MaRIO was built with the initial idea of adherence to COVID-19 measures concerning the wearing of masks [10–13], which continue to be in effect as of the time of publishing this paper. Through this idea, the goal of creating a robot capable of autonomous movement within a closed environment while ensuring that people in front of it in close proximity are wearing their masks was conceived.

This paper intends to document a useful example of human-robot interaction and open-source products that can be put together to create a low-cost system that can be further developed for a variety of other tasks not specific to the original goal. This project was created due to the Trinatronics 2021 competition, organized by the TriRhenaTech, an alliance of universities for applied sciences at the Upper Rhine region [14].

II. STATE OF THE ART

MaRIO's development covers a wide range of topics whose relevant literature is described within this section.

A. The Open-Source Approach

OpenCV and Python have been used in scientific work pertaining to human-robot interaction [7, 15–17]. The Raspberry Pi, being a cost-effective computing device, is also an often-used solution within the open-source ecosystem [15–17]. Arduino boards and the tools used alongside are also well-known for their role in the above mentioned ecosystem [17–19]. These devices are typically used with compatible hardware either from their respective producers [15, 18] or from third-party producers [16].

B. Human-Robot Interaction (HRI)

There exists a growing market in robots capable of interaction with humans. Pepper and Nao are well-known humanoid robot examples in this case, which are also capable of autonomous movement [20–22]. Given the close proximity of many HRI methods, robots with HRI features must also come equipped with the necessary safety standards and flexibility of movement and communication while also exhibiting an appearance that makes them desirable enough to interact with, besides other factors [20–23]. HRI is an integral part of human-robot collaboration (HRC) and the list of possible applications extends across a wide range of fields besides industrial applications, with education, public services and healthcare being examples [20–22, 24].

C. Mask detection using image processing and machine learning

The use of facemasks is currently a major topic in society. According to the World Health Organisation (WHO), face masks are a very effective way to reduce the spread of the Covid-19 pandemic [13]. In the last year, many measures have been developed to counter Covid-19 through the methods of Artificial Intelligence (AI), more specifically Machine Learning (ML) and Deep Learning (DL) [25]. One possible use of DL is the development of facemask recognition software. Such DL models have been published in scientific papers [26] or on the developmental platform GitHub, in which case the latter was used to fulfil the objectives of the project [27].

D. Autonomous Platforms

Classic industrial robots currently lack flexibility. They traditionally perform pre-programmed, repetitive tasks in a highly constrained workspace. They are also unable to adapt to new environments or situations [28].

The development of autonomous robots has increased their range of tasks significantly in recent years. Autonomous robots can, for example, build an intelligent logistics network [29, 30] or investigate in areas that are too dangerous for humans to enter such as mine shafts or nuclear power plants [31, 32]. However, autonomous robots can also assist humans with everyday tasks. For example, many households already have autonomously driven lawn mowing or vacuum cleaning robots [33, 34]. Robot autonomy is also an important part of HRI, as can be seen in the previously mentioned examples of Pepper and Nao [22].

The difficulty of such systems lies in safe movement and navigation in free space. To recognise all obstacles and not present a danger to the surroundings and itself, a multitude of sensors and intelligent software is necessary [35].

III. METHODS

The developmental process of MaRIO is subdivided into the categories listed below. A vast majority of the system was designed from the ground up, the exception being applied to some aspects of the software in order to complete the project within the given time constraints.

A. Construction

The assembly of the robot was conceived and visualized with the help of the computer-aided-design (CAD) package Siemens NX. Figure 1 illustrates the robot with the parts of its construction labelled. The chassis, mask holder and head are comprised of 5 cm thick wooden plates that were processed

using a laser machine and the open-source vector graphics editing software, Inkscape. The cutting pattern is produced with help of the open-source box generator Boxes.py, which provides a variety of patterns that can be flexibly tailored as per requirements. The chassis is separated into two parts, so as to separate the motors and a majority of the cabling from the electronics. A wooden plate protects the underbelly of the robot, while a plexiglass roof shields the electronics from the external environment, while also providing a brief visual of the system setup. The head, which houses the display as well as the Raspberry Pi, comprises a slot for concealed positioning of the camera as well as additional slots for a speaker and for any quick work required while working on the software and electronics. A PVC rod was chosen for the neck of the robot, with the mask holder placed high enough within arm's reach.

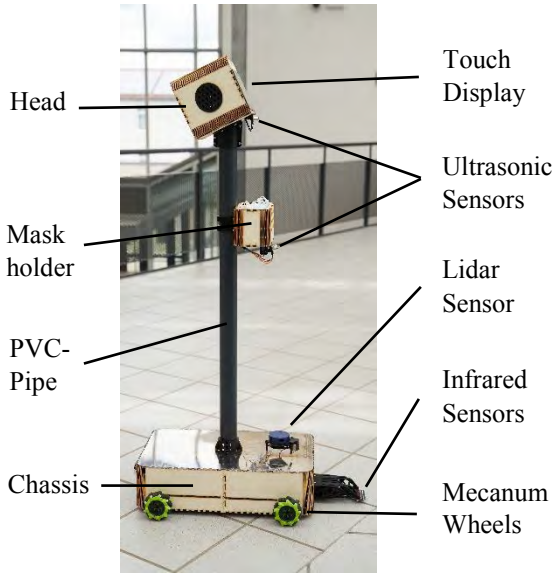


Figure 1: *A side view of MaRIO with all visible parts labelled*

A number of 3D-printed parts, constructed using NX, were employed in order to attach and reinforce specific components, such as the mask holder, head, neck and battery. The head is also inclined at an angle of 25° using 3D-printed parts. An additional attachment is positioned at the front of the robot for an ultraviolet sensor. This attachment is constructed in order to be long enough to give the robot time to respond to the sensor as required (see Section 4.2). All 3D parts were created using PLA filament.

It is important to note here that while Siemens NX is not open-source software, it need not be a necessary tool for visualizing the design of a system like MaRIO. If required, CAD alternatives also exist, among many others, depending on the needs and understanding of potential users [36].

B. Implementation of Electronic and Electrical Parts

a) Main controlling components: The latest version of the Raspberry Pi computers, being a tried, tested and proven solution for a large variety of applications, has proven itself worthy in being the main coordinating unit of MaRIO. The Raspberry Pi 4 features 8 GB of RAM and a 1.5 GHz Broadcom BCM2711, quad-core Cortex-A72 64-bit SoC and uses an OpenGL® ES 3.1 graphics API for GPU (graphics processing unit) computing, which in most cases satisfies the requirements for the average university robotics project.

Combining a Raspberry Pi computer with an Arduino Mega board via USB enhances the flexibility of the system to control sensors and actuators. An ATmega328P microcontroller developed by the Atmel Corporation (now acquired by Microchip Technology) powers the Arduino Mega. The division of labor was implemented in such a way that the Arduino Mega, alongside another Arduino Mega would be used to control as many sensors and actuators as possible. This would leave the Raspberry Pi with the objectives of communicating with the Arduino UNO and, more importantly, directly operating the mask recognition algorithm as well as the LiDAR (light detection and ranging) sensor, both of which are the most resource-intensive tasks in this system. Due to the latter two operations demanding so much computing power, a combination of active and passive cooling through a heat sink fitted with fans had to be used to keep the temperatures from going too high. Without cooling, the Raspberry Pi would reach a temperature beyond 80° Celsius, which would not only lead to a shutdown and failure of the Pi, but also reduce its lifetime.

A further division of labor takes place between both Arduino Mega boards. Directly connected to one of these are an SD card reader and an amplifier, which in turn leads to a speaker. The SD card loads audio output commands that are used in the mask recognition software (see section 4.3). The second board assumes control of the motors for robot motion.

b) Image Processing: The task of image processing for the purpose of recognizing masks is performed using a 5 MP camera module powered by an OmniVision OV5647 CMOS sensor and an aperture of f/1.8. Video data is fed to the Raspberry Pi, which differentiates between faces that do not wear a mask and those wearing a mask using an externally developed machine-learning algorithm [insert GitHub reference here]. The entire process of mask recognition is explained in further detail in section 4.4.

c) Motion: The movement of the robot would be coordinated via the Arduino using 4 Mecanum wheels, each powered by a single motor. Mecanum wheels possess the ability of turning the device to which they are equipped left and right while rotating about a single axis [37, 38]. This is accomplished through tiny mini wheels that comprise the contact surface of the wheel,

which rotate about their own position in order to "turn" the device to which the wheels are equipped. This movement would be additionally guided by a YDLIDAR X2 sensor and an Ultrasonic sensor. The YDLIDAR X2 sensor, as can be noticed in the name, is the sole LiDAR sensor in the robot system and is positioned right above the robot chassis in front of the neck. It is tasked with detecting any objects that could lie in front of the robot's intended movement path. The ultrasonic sensor is positioned under the mask holder and is attached to a servo motor such that it rotates only within a specific angle while also facing the frontal side of the robot. The rotation enables the sensor to detect any obstacles placed within the height of the robot. In the event that the robot encounters a lower floor surface, such as the next step of a staircase from the top end, an ultraviolet sensor is positioned at the front of the robot, giving it enough time to stop moving.

The robot motion is currently so programmed, that the robot only moves forward, stopping only to turn about its position depending on the input received from the obstacle detection sensorics before resuming forward motion.

d) *Power supply:* A 15000 mAh Li-ion battery pack was developed in-house, using a pack of 2 parallel sets of 5 batteries connected in series. The batteries are of the type ICR 18650, each possessing a voltage of 3.7V. The Pi and the robot display are connected to the battery pack via step-down transformers.

C. Software

Figure 2: *MaRIO's display showing the programmed environmental coverage from the LiDAR sensor*

The vast majority of the software aspect of the system has been written using Python. The code for motor control was however, for obvious reasons, implemented in C++ via the Arduino IDE. A program written specifically for the LiDAR sensor communicates with the Arduino UNO based on the input received from the robot's external environment, which then adjusts the movement of the 4 motors that influence the motion of the robot. An optional graphical interface has also been designed to better visualize how the LiDAR sensor interprets its surroundings.

Figure 3 (from top left clockwise): *Home screen for the mask detection interface, a screen indicating that the system does not recognise a mask on the individual standing in front of the camera and a screen indicating that the individual in front of the camera is wearing a mask*

The interface for human-robot interaction is controlled by a deep learning algorithm developed externally and shared over GitHub [27]. The deep learning algorithm was developed in Python using OpenCV, Keras and TensorFlow libraries, all of which are open-source programs for image processing and AI respectively. The data set used for this algorithm comprised 4095 images, which were divided into individuals wearing a mask and individuals not wearing a mask. The algorithm also makes use of the MobileNetV2 convolutional neural network architecture, which is designed for use on mobile devices [39].

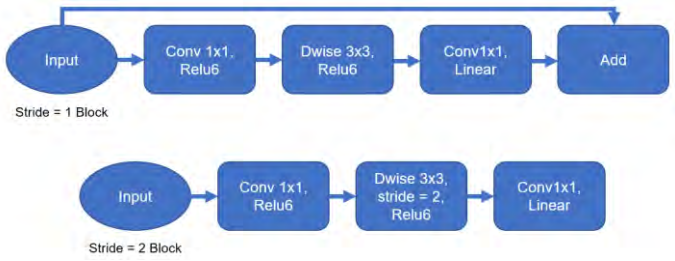


Figure 5: *Arrangement of convolutional blocks for the MobileNetV2 architecture. More information regarding the architecture can be found in the publication from Sandler et al [39].*

The theoretical accuracy after training via version 2.5.0 of the TensorFlow GPU was recorded at 98%. Considering an individual standing in front of a camera, an accurate result can be generated within 2 meters of distance between the camera and the individual. The detection accuracy reduces when considering a moving individual and may result in errors in correctly recognizing a face wearing a mask. Although the algorithm is capable of detecting several faces at a given time, the human-

robot interface has currently only been designed to read one face at a time.

D. Human-Robot Interaction

Abbildung 4: *A still depicting the results of the applied HRI algorithm*

The HRI feature of the robot is described in this section. All necessary programs for the complete functioning of the system are run on startup, this including the program for mask recognition. In order to elicit a response from the system, a single individual is required to pose within arm-distance of the robot. Following this, an image of a face with a green or red mask will be displayed, with green indicating that the individual is wearing a mask and red indicating that the individual is not wearing one. At the same time, audio messages are played based on the outcome, either acknowledging the person wearing a mask or requesting an individual not wearing a mask to either wear one or take one from the mask holder. In the case of the latter, the individual is requested to stay where they are until the robot is able to recognize a mask on their face.

IV. RESULTS

Figure 5: *Front-facing view of MaRIO*

As of the status of the project, the robot currently exists in a partially finished condition. Two versions of the robot were built under consideration of the conditions specified for the Trinatronics 2021 competition as well as the budget limit of 600 EUR per robot. While common issues between both builds were experienced and solved, a number of tasks and features could have been useful for the system but could not be fully implemented within the small time-frame of approximately two months. The most notable of these as well as the project accomplishments will be mentioned here.

Additional programming is required for some parts that have not been fitted but not programmed yet, such as the rotating ultrasonic sensor at the mask holder. An extremely important point to consider is the lack of sensors for obstacle detection. As it stands, only the frontal region of the robot is partially protected with the help of the LiDAR sensor, which is barely sufficient considering that the robot does not move backwards. The LiDAR sensor has been proven to do a good job at detecting walls and large obstacles. However, these tests were still in their early stages and it is clear that more sensors need to be integrated around the robot.

While the current motors responsible for robot motion appear to respond well to all tests conducted so far, it would still be safer to upgrade to ones that are more powerful. The current motors are designed with much smaller and lighter robot kits in mind and the combined weight of the robot platform goes much beyond the weight of the average mini-robot kit, which is usually less than a kilogram. However, while autonomous robot motion still requires more work, the robot does a somewhat decent job of moving about its environment with the current equipment under consideration.

The human-robot interaction, designed with regard to mask detection, provides accurate and desirable results. Lowering the response time and improving the look of the interface may definitely be a task for future work, although not as great a priority as the above-mentioned tasks. Although the Raspberry Pi 4 possesses the ability to perform compute intensive operations such as running image processing and machine learning algorithms, the response time is greatly reduced. Active and passive cooling is also necessary considering how quickly the Pi tends to heat up to the point of automatic shutdown without a sufficient cooling apparatus. Further experiments involving the use of additional hardware to accompany the Pi with such operations have not been conducted and may be investigated in future developments of the robot platform.

V. CONCLUSION

Although a complete and fully functioning system is difficult to implement from the ground up within a span of two and half months, it can definitely be proven from the work done so far, that a well-built robot with software specialized for human-robot

interaction can definitely be a reality. This is especially true as long as costs are kept at a minimum and with the help of open-source software. Additional resources and tests are definitely required in order to understand better how to manage the resources of the Raspberry Pi 4, while also placing focus on upgrading various aspects of the entire robot system for better results. With the required effort, MaRIO can be further developed into a system that can do much more than mask recognition. Examples of possible cases of application could be at old-age homes and at demonstrations in schools to generate additional interest among younger generations. This paper marks only the end of the beginning of a project that will continue to be worked on as time passes.

VI. ACKNOWLEDGEMENTS

Special thanks goes to Thomas Bartscherer, Sebastian Groß, Michael Mattern, Maximilian Hector and Vishnu Venkatesh for their roles in this project as well as this paper.

VII. REFERENCES

[1] M. A. Goodrich and A. C. Schultz, "Human-Robot Interaction: A Survey," *FNT in Human-Computer Interaction*, vol. 1, no. 3, pp. 203–275, 2007, doi: 10.1561/1100000005.

[2] S. H.-W. Chuah and J. Yu, "The future of service: The power of emotion in human-robot interaction," *Journal of Retailing and Consumer Services*, vol. 61, p. 102551, 2021, doi: 10.1016/j.jretconser.2021.102551.

[3] N. Tenhundfeld, H. M. Barr, E. O'Hear, A. Atchley, and J. E. Cotter, "Effects on Human-Likeness on Robot Use in High-Risk Environments," *Human Factors and Ergonomics Society*, 2021.

[4] *Pepper le robot humanoïde et programmable | SoftBank Robotics.* [Online]. Available: https://www.softbankrobotics.com/emea/de/pepper (accessed: Aug. 9 2021).

[5] G. Skantze, "Turn-taking in Conversational Systems and Human-Robot Interaction: A Review," *Computer Speech & Language*, vol. 67, p. 101178, 2021, doi: 10.1016/j.csl.2020.101178.

[6] H. A. Frijns, O. Schürer, and S. T. Koeszegi, "Communication Models in Human–Robot Interaction: An Asymmetric MODel of ALterity in Human–Robot Interaction (AMODAL-HRI)," *Int J of Soc Robotics*, 2021, doi: 10.1007/s12369-021-00785-7.

[7] B. Vaughan, E. Gilmartin, and N. Campbell, "Designing and Implementing a Platform for Collecting Multi-Modal Data of Human-Robot Interaction," *Acta Polytechnica Hungarica*, vol. 9, no. 1, 2012.

[8] M. J. Heron, V. L. Hanson, and I. Ricketts, "Open Source and Accessibility: Advantages and Limitations," *J Interact Sci*, vol. 1, no. 1, p. 2, 2013, doi: 10.1186/2194-0827-1-2.

[9] J. Đurković, V. Vuković, and L. Raković, "Open Source Approach in Software Development - Advantages and Disadvantages," *Management Information Systems*, vol. 2, no. 3, pp. 29–33, 2008.

[10] T. Czypionka, T. Greenhalgh, D. Bassler, and M. B. Bryant, "Masks and Face Coverings for the Lay Public : A Narrative Update," *Ann Intern Med*, vol. 174, no. 4, pp. 511–520, 2021, doi: 10.7326/M20-6625.

[11] G. Lu *et al.,* "COVID-19 in Germany and China: mitigation versus elimination strategy," *Global health action*, vol. 14, no. 1, p. 1875601, 2021, doi: 10.1080/16549716.2021.1875601.

[12] T. Maertl *et al.,* "Physical Activity during COVID-19 in German Adults: Analyses in the COVID-19 Snapshot Monitoring Study (COSMO)," *International journal of environmental research and public health*, vol. 18, no. 2, 2021, doi: 10.3390/ijerph18020507.

[13] World Health Organisation, "Mask use in the context of COVID-19: Interim Guidance," WHO/2019-nCoV/IPC Masks/2020.5, 2020.

[14] *Trinatronics.* [Online]. Available: https://www.trirhenatech.eu/lehre/Trinatronics/ (accessed: Aug. 9 2021).

[15] J. Vega and J. M. Cañas, "Open Vision System for Low-Cost Robotics Education," *Electronics*, vol. 8, no. 11, p. 1295, 2019, doi: 10.3390/electronics8111295.

[16] I. D. Orlandi, D. Cinalli, and P. Cruz, "CarNotFound: Project and Development of an Autonomous R/C Car," vol. 2, no. 1, pp. 2–18, 2019.

[17] A. Ambikapathy, J. Sandilya, A. Tiwari, G. Singh, and L. Varshney, "Analysis of Object Following Robot Module Using Android, Arduino and Open CV, Raspberry Pi with OpenCV and Color Based Vision Recognition," in *Lecture Notes in Electrical Engineering*, vol. 690, *Advances in Power Systems and Energy Management*, N. Priyadarshi, S. Padmanaban, R. K. Ghadai, A. R. Panda, and R. Patel, Eds., Singapore: Springer Singapore, 2021, pp. 365–377.

[18] J. Vega and J. Cañas, "PiBot: An Open Low-Cost Robotic Platform with Camera for STEM Education," *Electronics*, vol. 7, no. 12, p. 430, 2018, doi: 10.3390/electronics7120430.

[19] J. Vega and J. M. Cañas, "PyBoKids: An Innovative Python-Based Educational Framework Using Real and Simulated Arduino Robots," *Electronics*, vol. 8, no. 8, p. 899, 2019, doi: 10.3390/electronics8080899.

[20] A. K. Pandey and R. Gelin, "A Mass-Produced Sociable Humanoid Robot: Pepper: The First Machine of Its Kind," *IEEE Robot. Automat. Mag.*, vol. 25, no. 3, pp. 40–48, 2018, doi: 10.1109/MRA.2018.2833157.

[21] J. Dawe, C. Sutherland, A. Barco, and E. Broadbent, "Can social robots help children in healthcare contexts? A scoping review," *BMJ paediatrics open*, vol. 3, no. 1, e000371, 2019, doi: 10.1136/bmjpo-2018-000371.

[22] M. Garcia, L. Béchade, G. Dubuisson-Duplessis, G. Pittaro, and L. Devillers, "Towards metrics of Evaluation of Pepper robot as a Social Companion for Elderly People," *Proceedings of the 8th International Workshop on Spoken Dialog Systems*, 2017.

[23] F. Tanaka, K. Isshiki, F. Takahashi, M. Uekusa, R. Sei, and K. Hayashi, "Pepper learns together with children: Development of an educational application," in *2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)*, Seoul, South Korea, 2015, pp. 270–275.

[24] S. Robla-Gomez, V. M. Becerra, J. R. Llata, E. Gonzalez-Sarabia, C. Torre-Ferrero, and J. Perez-Oria, "Working Together: A Review on Safe Human-Robot Collaboration in Industrial Environments," *IEEE Access*, vol. 5, pp. 26754–26773, 2017, doi: 10.1109/ACCESS.2017.2773127.

[25] S. Agarwal *et al.,* "Unleashing the power of disruptive and emerging technologies amid COVID-19: A detailed review," May. 2020. [Online]. Available: http://arxiv.org/pdf/2005.11507v6

[26] L. Bellatreche, V. Goyal, H. Fujita, A. Mondal, and P. K. Reddy, *Big Data Analytics*. Cham: Springer International Publishing, 2020.

[27] GitHub, *GitHub - chandrikadeb7/Face-Mask-Detection: Face Mask Detection system based on computer vision and deep learning using OpenCV and Tensorflow/Keras.* [Online]. Available: https://github.com/chandrikadeb7/Face-Mask-Detection (accessed: Aug. 6 2021).

[28] A. Saffiotti, "The uses of fuzzy logic in autonomous robot navigation," *Soft Computing*, vol. 1, pp. 180–197, 1997.

[29] K. Gao, J. Xin, H. Cheng, D. Liu, and J. Li, "Multi-Mobile Robot Autonomous Navigation System for Intelligent Logistics," in *2018 Chinese Automation Congress (CAC)*, Xi'an, China, 2018, pp. 2603–2609.

[30] D. Wahrmann, A.-C. Hildebrandt, C. Schuetz, R. Wittmann, and D. Rixen, "An Autonomous and Flexible Robotic Framework for Logistics Applications," *J Intell Robot Syst*, vol. 93, 3-4, pp. 419–431, 2019, doi: 10.1007/s10846-017-0746-8.

[31] C. Baker *et al.,* "A campaign in autonomous mine mapping," in *IEEE International Conference on Robotics and Automation, 2004. Proceedings. ICRA '04. 2004*, New Orleans, LA, USA, 2004, 2004-2009 Vol.2.

[32] Institute of Electrical and Electronics Engineers; City University of Hong Kong, *Proceedings / M2VIP 2001, 8th IEEE Conference on Mechatronics and Machine Vision in Practice, 2001: Regal Kowloon Hotel, Hong Kong SAR, China, 27 - 29 August*. Hong Kong, 2001.

[33] J. Forlizzi and C. DiSalvo, *HRI 2006: Proceedings of the 2006 ACM Conference on Human-Robot Interaction March 2-4, 2006, Salt Lake City, Utah, USA toward human robot collaboration*. New York: ACM Press, 2006.

[34] M. Veloso, J. Biswas, B. Coltin, and S. Rosenthal, "CoBots: Robust Symbiotic Autonomous Mobile Service Robots," *Proceedings of the Twenty-Fourth International Joint Conference on Artificial Intelligence (IJCAI)*, 2015.

[35] T.-J. Lee, D.-H. Yi, and D.-I. D. Cho, "A Monocular Vision Sensor-Based Obstacle Detection Algorithm for Autonomous Robots," *Sensors (Basel, Switzerland)*, vol. 16, no. 3, 2016, doi: 10.3390/s16030311.

[36] M. T. Vasques, M. Mori, and D. C. Laganá, "Three-dimensional printing of occlusal devices for temporomandibular disorders by using a free CAD software program: A technical report," *The Journal of prosthetic dentistry*, vol. 123, no. 2, pp. 232–235, 2020, doi: 10.1016/j.prosdent.2018.12.017.

[37] A. Gfrerrer, "Geometry and kinematics of the Mecanum wheel," *Computer Aided Geometric Design*, vol. 25, no. 9, pp. 784–791, 2008, doi: 10.1016/j.cagd.2008.07.008.

[38] J. E. Mohd Salih, M. Rizon, and S. Yaacob, "Designing Omni-Directional Mobile Robot with Mecanum Wheel," *American J. of Applied Sciences*, vol. 3, no. 5, pp. 1831–1835, 2006, doi: 10.3844/ajassp.2006.1831.1835.

[39] M. Sandler, A. Howard, M. Zhu, A. Zhmoginov, and L.-C. Chen, "MobileNetV2: Inverted Residuals and Linear Bottlenecks," *The IEEE Conference on Computer Vision and Pattern Recognition (CVPR)*, pp. 4510–5420, 2018. [Online]. Available: http://arxiv.org/pdf/1801.04381v4

www.robotix.academy