*Journal of Machine Engineering, 2018, Vol. 18, No. 3, 28–41*
*ISSN 1895-7595 (Print) ISSN 2391-8071 (Online)*

Received: *22 December 2017* / Accepted: *06 June 2018* / Published online: *28 September 2018*

*assembly, machine learning, neural network, industrial robot*

Paul BOBKA[1*], Jakob HEYN[1]
Jann-Ole HENNINGSON[1], Martin RÖMER[1]
Thomas ENGBERS[1], Franz DIETRICH[1]
Klaus DRÖDER[1]

# DEVELOPMENT OF AN AUTOMATED ASSEMBLY PROCESS SUPPORTED WITH AN ARTIFICIAL NEURAL NETWORK

A central problem in automated assembly is the ramp-up phase. In order to achieve the required tolerances and cycle times, assembly parameters must be determined by extensive manual parameter variations. Therefore, the duration of the ramp-up phase represents a planning uncertainty and a financial risk, especially when high demands are placed on dynamics and precision. To complete this phase as efficiently as possible, comprehensive planning and experienced personnel are necessary. In this paper, we examine the use of machine learning techniques for the ramp-up of an automated assembly process. Specifically we use a deep artificial neural network to learn process parameters for pick-and-place operations of planar objects. We describe how the handling parameters of an industrial robot can be adjusted and optimized automatically by artificial neural networks and examine this approach in laboratory experiments. Furthermore, we test whether an artificial neural network can be used to optimize assembly parameters in process as an adaptive process controller. Finally, we discuss the advantages and disadvantages of the described approach for the determination of optimal assembly parameters in the ramp-up phase and during the utilization phase.

## 1. INTRODUCTION

The set-up process of the assembly plant -also called ramp-up- is a central problem in automated assembly. Depending on the complexity of the assembly task, setting the mounting parameters can be very time-consuming and therefore costly. This concerns, for example, the parametrization of gripping, motion and placement processes. Parameters such as the gripping position, robot path, acceleration, speed and the operating parameters of the gripping tool, compressed air supply pressure or force control, must be taken into account. Camera calibration is usually necessary for the localization and measurement of components to achieve the desired accuracy. A coordinate transformation must be identified to define the robot and camera coordinate systems relative to each other, if a camera is used to determine gripping positions for the robot [1].

---

[1] Technische Universität Braunschweig, Institut für Werkzeugmaschinen und Fertigungstechnik (IWF), Braunschweig Germany
[*] E-mail: p.bobka@tu-braunschweig.de
https://doi.org/10.5604/01.3001.0012.4605

### 1.1. THE CHALLENGE

The explicit descriptions of the systems, the calibration process, tolerances of the plant and the assembled components, as well as the prevailing environmental conditions are potential sources of error. Because of the multitude of potential error sources, the search for and the elimination of a fault or fault combination can be very labor-intensive and complicated. Especially non-linear effects are a major problem because their relations are difficult to identify and to understand by humans. The identification of the causal relationships is difficult because the causes and effects are in different physical domains, for example, optical, geometric thermal, structural, electrical, magnetic or dynamic effects and combined effects. These include temperature effects, rotation errors, wear, camera lens distortions. Even worse, however, are discontinuous nonlinear error effects, such as rapid ambient light changes, or vibrations from environmental processes. Therefore much time is usually wasted in order to identify the relevant correlations. Figure 1 shows the potential error sources and their effects in a generic assembly system.

**potential error sources:**

- Inexact location description
- Inhomogeneous lens distortions and camera calibration
- Equipment and Product tolerances
- Wear and aging
- Exposure or radiation effects
- Temperature effects
- Humidity and air pressure
- Vibrations
- Flow effects
- Tribological effects
- Electrical and Magnetic effects
- Etc.

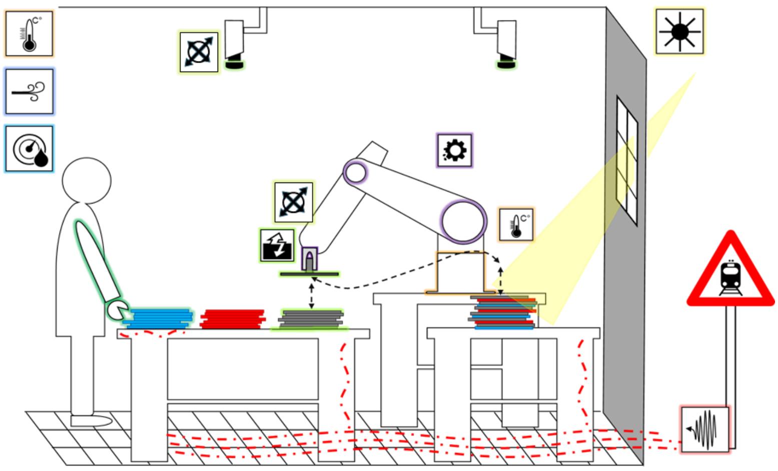


Fig. 1. Influences and combinations of influences that aggravates automated assembly

The necessary coordinate systems (e.g. robot, gripping tool, assembly fixtures, camera) can be defined by means of coordinate transformations to each other. In industrial applications, computer aided engineering (CAE) systems are often used to plan the work cell of an application and the program sequences. Program sequences can be created and optimized with regard to collision, travel times and dynamics with CAE systems. However, the CAE model is always a simplified representation of the real plant. Therefore, parameters determined in a simulation by means of the CAE system can only lead to an optimal assembly program in reality on the condition that the model is sufficiently precise. The CAE model is a simplified, bounded representation of reality to keep modelling efforts manageable. It is not possible to take all the real assembly influences into account, since for technical reasons, and above all for economic reasons, the degree of detail of the models is limited [2]. Therefore not all error influences that arise in reality can be consider in simulations.

Figure 1 shows a number of negative influences on repeatability and precision in assembly processes. These can only be taken into account to a limited extent in a simulation model. Common influences on deviations in a final assembly are geometric tolerances in the gripper tool, in assembly fixtures or in the assembled components themselves. Thermal influences, for example in the warm-up phase of a robot, can lead to variable positioning results. A common problem for industrial image processing is variable lighting or shading conditions that can lead to poor image processing results. Considering all the relevant disturbances, devising an adaptive control scheme to compensate for error influences is a challenge, since the error effects are superimposed and influence each other.

### 1.2. PREVAILING SITUATION

The problem with the current approach is finding a parameter set that can optimally represent all relevant variable influences during operation. Figure 2 details the flow of information during the lifecycle of an assembly plant.

Major properties and specifications of an assembly system are defined in the planning phase of the system. To simplify the ramp-up phase and to guarantee a robust operation it is necessary to plan for enough technical reserves. However, these technical reserves are associated with additional costs in plant construction. Cost drivers include, for example, the required robot repeatability or the required tolerances in the camera technology or assembly fixtures.

To achieve the required specifications under the influence of relevant error sources, a test phase is usually necessary parallel to or directly after the construction of the system. During the test phase, emerging assembly deviations are usually corrected by additional calibration processes, improvement of coordinate transformations or by additional offset parameter [3].

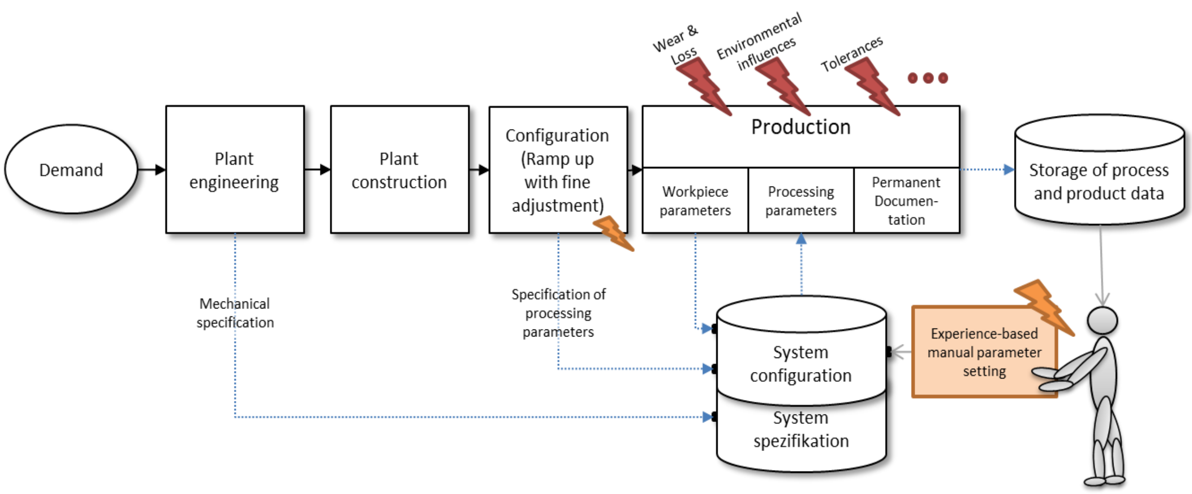



Fig. 2. Manual laborious search for suitable parameter sets

Experienced, well-trained personnel are needed to quickly identify all relevant influences and relationships and to adapt parameters in a focused way. Either experience-based or methodical approaches are used to quickly generate a sufficient parameter configuration by these Experts (for example by means statistical design of experimen) [4, 5]. Due to time and economic limitations, it can be assumed that a search for the best parameter set in the test phase can only be limited.

Upon completion of the test phase, the productive operation phase starts. During the productive phase parameters of the assembly system must constantly be adjusted. The causes for this are manifold and can be attributed, for example, to system wear, aging, changed component tolerances or changed environmental conditions. The parameter adjustments are determined based on the operators experience, often ad-hoc during the production operation. This continuous manual intervention is depicted in Fig. 2.

### 1.3. MOTIVATION AND TARGET

Since the experience-based set up strategies are labor intensive, automation concepts for the parametrization of assembly processes are needed, especially with regard to flexible production concepts and small lot sizes. Continuous optimization and parametrization during the production phase would also be beneficial for the stability and productivity of the assembly process.

The first objective is to arrive at a suitable parameter set efficiently when setting up or retooling assembly systems. The second objective is adaptive compensation of non-linear, combined, multi-dimensional error influences during operation.

The third objective is to optimize the assembly parameters up to the technical maximum, despite economic (time and financial) limitations. Goals in this optimization are, for example, the increase in assembly accuracy and the increase in the output quantity, or the reduction of wear and energy consumption. Finding a balanced optimum between these goals usually requires a lot of effort and experienced staff. This optimization should therefore be done as autonomously as possible. The vision is to reach an optimum between these target variables, under varying influence parameters with the least possible effort.

## 2. IMPROVEMENT OF A PICK AND PLACE PROCESS USING AN ARTIFICIAL NEURAL NETWORK

The idea to use Artificial Neural Networks (ANN) in the control handling systems is well established [6-9].The central approach applied in this paper is to use ANNs to predict error magnitudes and use them in plant control for error compensation [10]. The ANN is used to approximate linear, as well as non-linear, multi-dimensional influencing variables or combinations of influencing variables. Predicted errors are used to improve plant control. The ANN will enable a product-related control loop for plant control in order to be able to adapt the process parameters during operation with variable influences.

ANNs represent a promising approach to parameter optimization in handling technology and automated assembly for a number of reasons. One of the most important features of ANNs, that offers great potential in automated assembly, is the ability to learn an approximation of any continuous mathematical function [11]. Due to this property, it should be possible to map continuous, nonlinear relationships that are attributed to physical, chemical or other causes with an ANN. ANNs could also be used to model unknown relationships when extensive data sets containing peripheral information are used as training data [12]. For example, if an ANN is trained with mounting deviations and the time of day, it could learn and predict continous time-dependent mounting deviations [13]. A theoretical example is the influence of mechanical vibrations caused by a tram passing next to the factory building, as shown in Fig. 1. If the positioning accuracy of a precision assembly process is periodically poor due to the rhythm of the tram timetable, an ANN could recognize the connection between time and positioning accuracy. A human being would have to actively search for and recognize the temporal pattern in order to identify the tram influence. In addition, a continuously (online) learning ANN could take into account timetable changes after a certain period of time, whereas a person would have to be commissioned to investigate the pattern again.

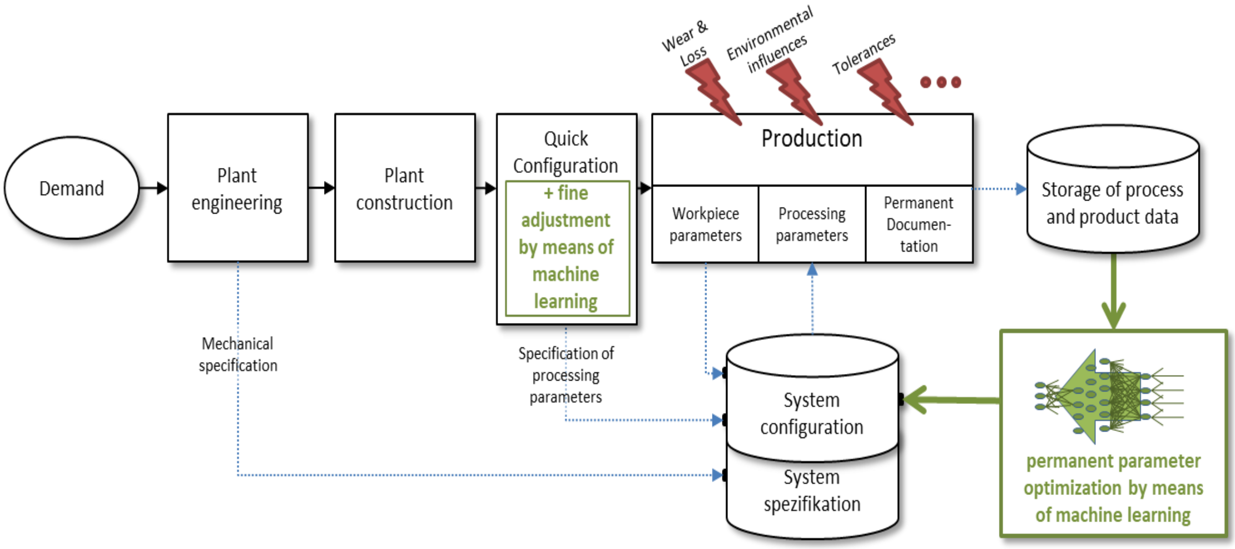


Fig. 3. Automatic process parametrization by an artificial neural network

ANNs offer the potential to account for complex relationships, which are represented directly or indirectly in a given dataset. The recognition of patterns in data sets with many different parameters and unknown correlations is an interesting feature for automated assembly, since these multiple (cross-) correlations make a targeted optimization of the assembly line very complex for humans. Another feature of an ANN is its high tolerance to internal and external errors (Internal errors are not considered in this paper). Tolerance to external errors means that ANNs can also work with incomplete/unknown or damaged input values. A properly trained ANN has a generalizing character that can enable it to process unknown, incomplete or faulty data correctly. For automated assembly, this capability offers the potential to increase the robustness of the overall system by providing extensive, redundant data sets alone. For example, a failure of a sensor could be compensated by the ANN estimating a sensor value via the remaining inputs. The ability to learn, tolerance towards errors and generalization of correlations are arguments for the use of ANN in automated assembly [14].

The approach presented in the remainder of this paper is an autonomous optimization of assembly parameters by an artificial neural network (ANN). How assembly parameters can be adapted by an ANN to increase the assembly quality is shown in an application study on the pick and place process of Lithium-ion pouch cell components. This approach simplifies the ramp-up process significantly and is therefore cheaper.

The ANN is applied to the task of finding causal relationships in the stored process and product data and predicting assembly deviations. Then, the mounting deviations predicted by the ANN are used to fine-tune the plant control and to improve the assembly quality. The ANN is introduced in two distinct phases of the process life cycle as shown in Fig. 3: as an adjustment of process parameters in the ramp-up phase and as a continuous optimizer in the production phase.

## 3. CASE STUDY: PICK AND PLACE OF LITHIUM ION BATTERY COMPONENTS

The stacking of battery components is a currently relevant pick and place process. Mounting accuracy is important in this process to ensure a good performance of the finished product. This assembly process of planar components is used as the demonstration example in this paper. The aim of this experiment is to show how a ANN can be used to increase the accuracy of a pick and place process. For this purpose, a test rig with very limited accuracy was set up. The ramp-up was reduced to the basic functions without investing time in the calibration, to specifically demonstrate the ability of an ANN to deal with a badly calibrated plant.

### 3.1. EXPERIMENTAL SETUP AND TEST PROCEDURE OF THE PICK AND PLACE PROCESS

The assembly process of lithium ion battery components can be reduced to a simple pick and place process. This is a generic handling process with four degrees of freedom ($X$, $Y$, $Z$, $\Theta$). Similar handling processes are found in many different industries, for example the precise assembly of surface mounted devices (SMD) on printed circuit boards in the electronics industry or the packaging of products in the food- and pharmaceutical industries. Figure 4 illustrates the setup of the experiment, which consists of a robot with gripper, two tables and two cameras for detecting and measuring components. The experimental setup is devised to represent the generic pick and place process: variably deposited electrodes are picked up from a table in “*Field A*” and deposited on a measuring table in “*Point B*” with a defined orientation by a robot.

At the start of the handling process camera one above *Field A* is used to detect the position of the electrodes. Camera one is a zoom camera with a 2592 × 1944 pixel

(5.04 mega pixel) sensor. With a fixed image width of 665 mm (zoom off) and a divisor of 4 for the edge scanning (Nyquist-Faktor) [15], the camera one offers a resolution of 1.026 mm without subpixel interpolation. The position detection in *Field A* is realized by a classical pattern matching. In this case the image processing software HALCON was used for the experiment, but any suitable image processing software could be used. The components can be deposited in varying positions and orientations ($X$, $Y$, $\Theta$) in *Field A*. Based on a successful pattern matching, the center position and orientation of the part in *Field A* is calculated and transmitted to the robot controller. The robot moves to the detected position in *Field A*, reorients the gripper to the angle $\Theta$ and grips the part by vacuum. In order to guarantee the basic function of the camera-supported gripping, a rough coordinate transformation from the camera coordinate system to the robot coordinate system was created before the experiments. However, an exact calibration of the camera coordinate system to the robot coordinate system was not carried out. Furthermore, the precise alignment and calibration of the camera one was deliberately dispensed with.

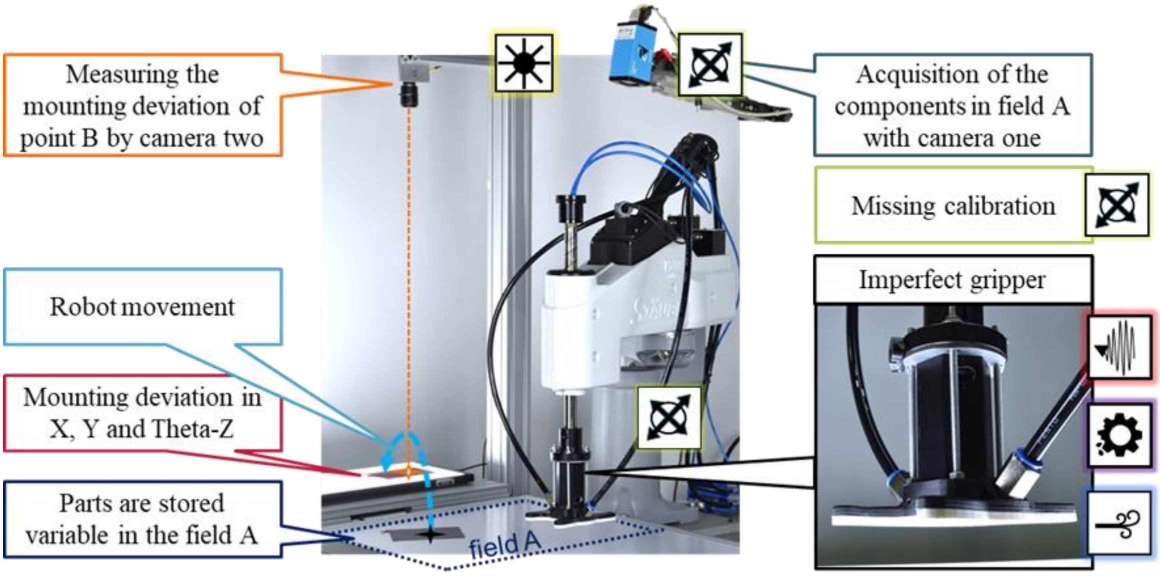



Fig. 4. Experimental Setup

After the robot has gripped the component, it moves to a taught-in assembly *Point B* and places it there. The robot then moves to a waiting position outside the field of view of camera two so that the resulting mounting deviations at *Point B* can be measured. This second camera, used to measure the mounting deviation, has a sensor with 3840 × 2748 pixels (10.55 mega pixel). In our setup this camera therefore offers a resolution in the detection plane of ~ 0.16 mm, based on an image width of 158 mm and a divisor of 4 for the edge scanning, without subpixel interpolation. The lens distortion of the camera two was calibrated with a calibration standard. An exact calibration to the robot was not made, since only offsets from a desired positon are to be measured.

With the second camera and image processing, the deviations in $X$, $Y$ and $\Theta$ at mounting *Point B* are determined. In order to repeat the process automatically after measurement, the robot places the component at a new location in *Field A*. In this way, data about the pick and place process could be collected quickly (one repetition of the cycle takes about 10 s). The procedure of the pick and place process and experimental data acquisition is shown in Fig. 5.

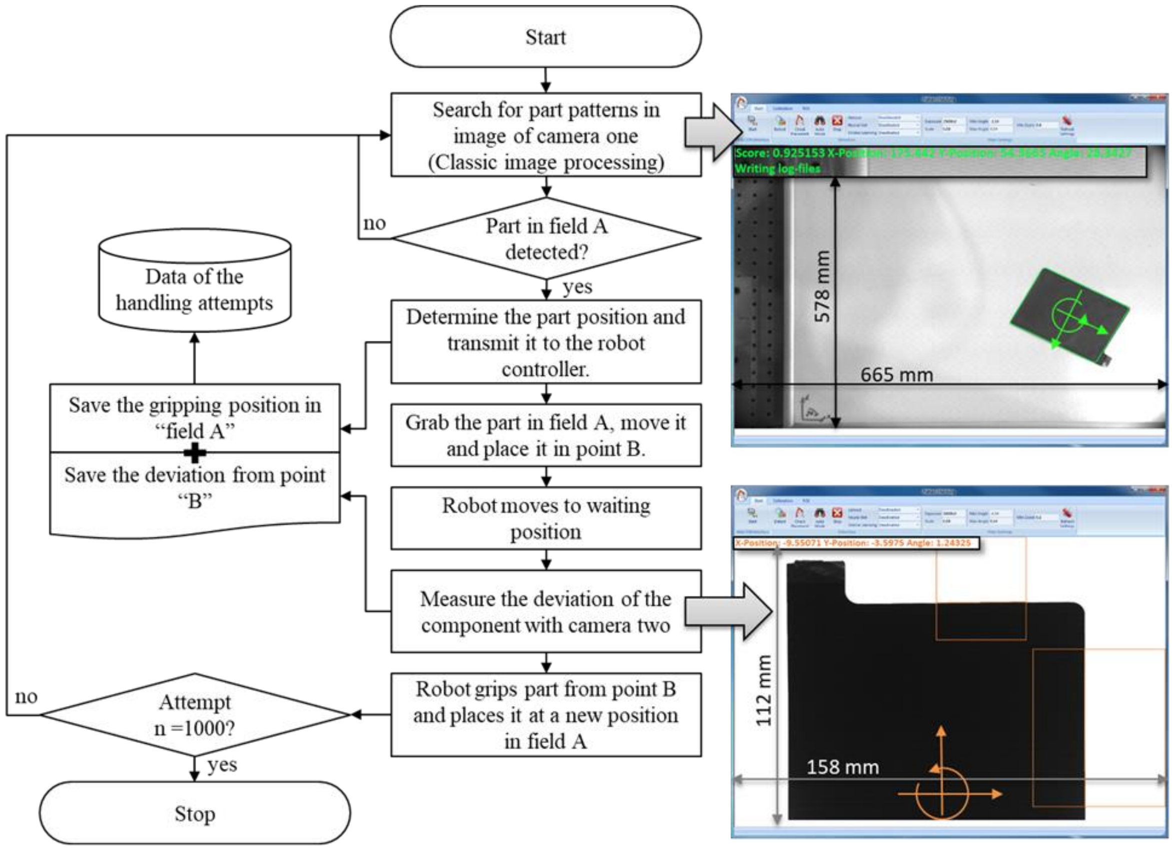



Fig. 5. Procedure of the pick and place experiments

### 3.2. ANALYSIS OF INITIAL EXPERIMENTAL RESULTS WITHOUT ANN

The pick-and-place attempt was repeated 1000 times. From each test the detected part position in field A and the measured assembly deviation at point B were stored. Figure 6 shows the detected positions of the parts in field A. The detection zone of the camera one represented by field A has been sampled extensively. Figure 7 shows the corresponding measured mounting deviations at point B. The open-loop process without any correction resulted in a flat sickle-shaped distribution of the mounting deviations. In order to clearly identify this characteristic distribution of the mounting deviation, a certain number of tests are necessary.

The deviations from the setpoint at position B shown in Fig. 7 are large compared to the desired assembly accuracy. On the X-axis, deviations between -25 mm and +20 mm were measured. On the Y-axis deviations between -22 mm and +2 mm were measured. One possible reason for this is the very poor alignment and missing calibration of camera one, which is used for the detection of the component in field A. In addition to the camera resolution and calibration, the robot kinematics and the gripping system also have an influence on the assembly quality. In the experiments an industrial SCARA robot, model "Stäubli TS 80" was used. This robot offers a high repeat accuracy of ± 0.01 mm. An inexact robot can thus be excluded as the cause of the detected deviations. The gripping system is a very simple vacuum surface aspirator produced by additive manufacturing. The handling accuracy of the gripper was not determined. However, it has been found that the gripper vibrates due to low rigidity and the vacuum pads are not particularly flat because of the additive manufacturing process used to fabricate it. Therefore the handling accuracy is assumed to be very limited.

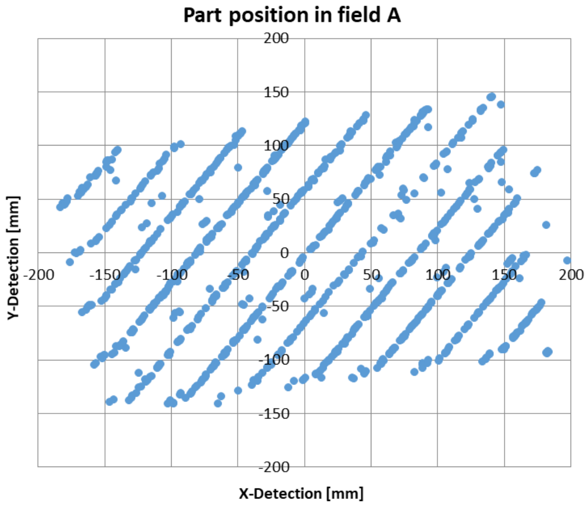



Fig. 6. 1000 detected positions of the parts in *Field A*

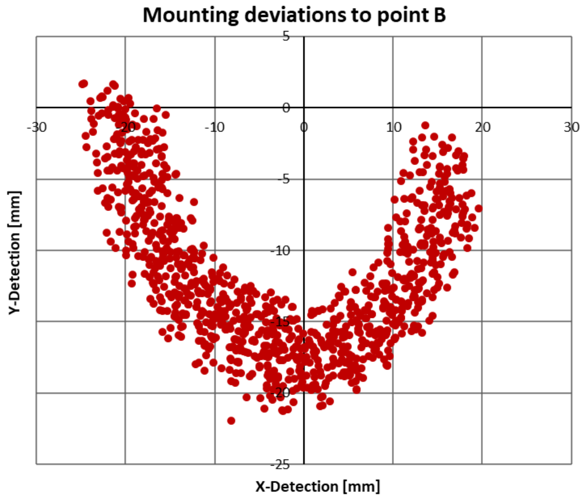



Fig. 7. 1000 measured mounting deviations to *Point B*, after the pick-and-place process

Overall the experimental handling system comprises a multitude of error sources, which influence the measured assembly deviations. To eliminate all of these, time consuming manual labor would be necessary. Therefore the experimental setup is a good representation of the challenges faced in the ramp-up of real automated assembly systems.

### 3.3. REDUCTION OF THE ASSEMBLY DEVIATION BY AN ARTIFICIAL NEURAL NETWORK

The deviations of the assembly position should now be reduced by an ANN. To this end the ANN should represent the transformation of the measured pick position in field A to the positioning deviation at *Point B*. This "error estimation" can then be used in a feed-forward control scheme for the place position.

Through the accuracy investigations, 1000 pick and place tests were recorded as system observations: 800 were used as training data and the remaining 200 as validation data in the offline training step. An extract of the saved file with 1000 attempts is shown in Table 1.

Table 1. Extract of the saved file with 1000 attempts

| | Inputs_data (part positions in field A) | | | Outputs_target (mounting deviations to point B) | | |
|---|---|---|---|---|---|---|
| Row Nr. | *X*-Detection | *Y*-Detection | *RZ*-Detection | *X*-Deviation | *Y*-Deviation | *RZ*-Deviation |
| 1 | -49.513462 | 78.765 | -28.0123 | 4.63602 | -17.8819 | 0.614641 |
| 2 | 13.016266 | -116.947 | 18.2779 | -6.708044 | -14.6227 | 0.298875 |
| 3 | 25.25279 | -92.7833 | 31.9414 | -10.798522 | -13.0408 | 0.412524 |
| up to 1000 | … | … | … | … | … | … |

The ANN was trained line by line in the illustrated application. Each line contains the input data and the corresponding output data for the training. The gripping positions represent our input data for the ANN. The assembly deviations represent the output data for the training of the ANN. For training, it is important to transfer the determined gripping position with the associated measured assembly deviations as a batch to the ANN. If the input values and output values were swapped or mixed, the ANN would be trained incorrectly.

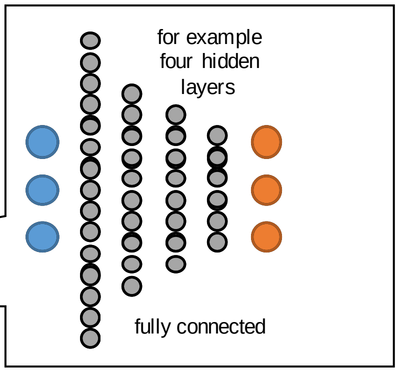
In order to learn the correlation between pick pose and assembly deviation, a deep feedforward artificial neural network [16] with a total of 6 layers and 45 neurons was constructed. To enter the position in $X$ and $Y$ as well as the rotation $\Theta$ of the detected component, the network requires three input neurons. For the prediction of the assembly deviation at *Point B* with $X$ and $Y$ and the rotation $\Theta$, the network requires three output neurons. Between the input layer and output layer, the network has four hidden layers, with fifteen, ten, eight, and six neurons each.

A deep neural network was observed to be beneficial for this application. With a simple network with one hidden layer, the desired predictive accuracy could not be achieved, because deep neural networks have better parameter efficiency in multidimensional interrelations than networks with only one hidden layer and many neurons [16]. The hidden layers 1-4 have activation function of the type: exponential linear unit (ELU) [17]. In this work Python 3 with Tensorflow was used for the modeling of the ANN. Alternatively, other systems are possible. Part of the Python Tensorflow code is shown in Fig. 8.

After the training of the ANN, a second experiment was conducted with the assembly system. The experimental procedure was basically identical to the first experiment. However, the trained ANN was used to generate offsets ($\Delta X$, $\Delta Y$ and $\Delta\Theta$) for the placement processes. The trained ANN was given the recognized position of the component in the *Field A*. The trained ANN can then determine the probable position deviation ($\Delta X$-Prediction, $\Delta Y$-Prediction and $\Delta\Theta$-Prediction). The predicted position deviation is added as an additional offset to the robot pose. In the experiment, the positional deviation was added directly to the gripping position, since the largest error influences were assumed to be in the gripping position calculation.

It is also conceivable to add the positional deviation to the place position. This experiment was carried out 180 times. The result of these pick and place tests are shown in Fig. 9.

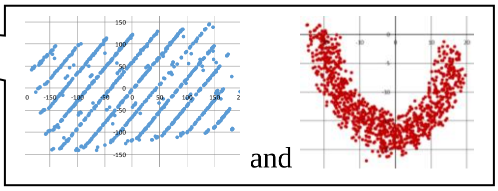
```
# Import tensorflow, numpy, and csv-libraries
[...]
# Read and split data in test and training-data
[...]

# Set number of epochs and number of datapoints
n_epochs = 100000
batch_size = 1
neuralNetName = 'DeepNetELU-15-10-8-6'

# Set number of hidden layer nodes
hidden_layer_nodes_1 = 15
hidden_layer_nodes_2 = 10
hidden_layer_nodes_3 = 8
hidden_layer_nodes_4 = 6
```



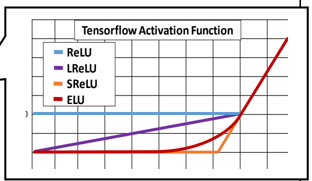

```
# Set number of in and outputs
nr_of_inputs = 3
nr_of_outputs = 3
```

Inputs: X-Detection, Y-Detection, RZ-Detection
Outputs: X-Prediction,Y-Prediction, RZ-Prediction

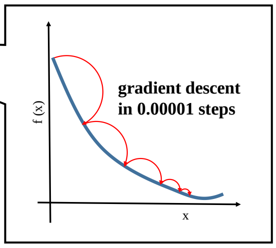
```
#Create placeholder for in- and outputs
inputs_data = tf.placeholder(dtype=tf.float32, shape=[batch_size, nr_of_inputs], name='input')
outputs_target = tf.placeholder(dtype=tf.float32, shape=[batch_size, nr_of_outputs])
```




```
#Declare model
W1 = tf.Variable(tf.truncated_normal(shape=[nr_of_inputs, hidden_layer_nodes_1], stddev=0.1))
b1 = tf.Variable(tf.constant(0.0000001, shape=[hidden_layer_nodes_1]))
W2 = tf.Variable(tf.truncated_normal(shape=[hidden_layer_nodes_1, hidden_layer_nodes_2],
stddev=0.1))
b2 = tf.Variable(tf.constant(0.0000001, shape=[hidden_layer_nodes_2]))
#Declare W3, b3, W4, b4 similarly
# [...]
W5 = tf.Variable(tf.truncated_normal(shape=[hidden_layer_nodes_4, nr_of_outputs], stddev=0.1))
b5 = tf.Variable(tf.constant(0.0000001, shape=[nr_of_outputs]))

#Declare model operations
hidden_outputs_1 = tf.nn.elu(tf.add(tf.matmul(inputs_data, W1), b1))
hidden_outputs_2 = tf.nn.elu(tf.add(tf.matmul(hidden_outputs_1, W2), b2))
# [...]
final_outputs = tf.identity(tf.add(tf.matmul(hidden_outputs_4, W5), b5))
```

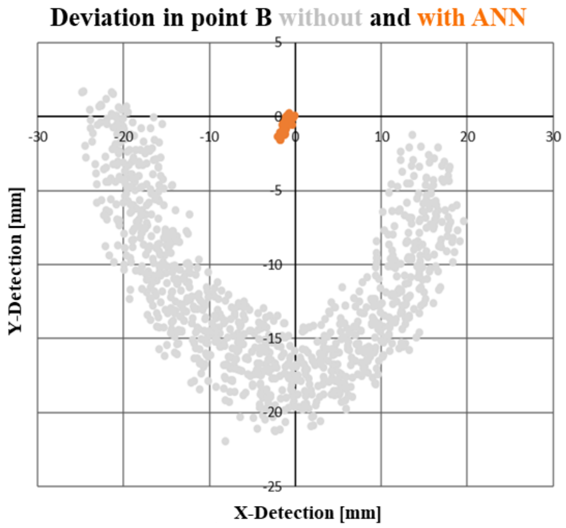



```
#Declare optimizer and loss function
loss = tf.reduce_sum(tf.abs(outputs_target - final_outputs))
optimizer = tf.train.GradientDescentOptimizer(0.00001)
train_step = optimizer.minimize(loss)
```




```
#Start session
with tf.Session() as sess:
    #Initialize variables
    init = tf.global_variables_initializer()
    sess.run(init)
    #Train the model

    #Split data into batches to feed to the model (test and training)
    #[...]

    #Train and test the model
    sess.run(train_step, feed_dict = {inputs_data: input_batch, outputs_target: output_batch})
    sess.run(loss, feed_dict = {inputs_data: input_test_batch, outputs_target:
output_test_batch})

    #Get the prediction of the model
    prediction = sess.run(final_outputs, feed_dict = {inputs_data: input_test_batch})

#Save the model#[...]
```

Fig. 8. Partial Source code of the ANN application in TensorFlow




**Mounting Deviation without ANN**

| | ΔX | ΔY | ΔΘ |
|---|---|---|---|
| **Average** | -3.15 | -11.65 | 0.51 |
| **Dispersion** | 12.40 | 5.31 | 0.32 |

**Mounting Deviation with ANN**

| | ΔX | ΔY | ΔΘ |
|---|---|---|---|
| **Average** | -1.18 | -0.55 | -0.01 |
| **Dispersion** | 0.63 | 0.59 | 0.24 |

Fig. 9. Validation result from 190 Pick and place attempts

This is an exemplary non-optimized network. In order to quickly achieve a high prediction accuracy about the deviations, the number of neurons and hidden layers, for example, must be adapted to the quantity and quality of the training data.

The result clearly shows an improvement in mounting accuracy. In additional investigations, the ANN was further trained online. It was observed that the accuracy can be improved further with additional training data. For industrial operations, an ANN could thus be used as a continuous assembly optimizer. The simple implementation as an additional offset could also prove advantageous in already constructed systems.

## 4. CONCLUSION AND OUTLOOK

In this paper it is shown how artificial neural networks can be used to improve mounting accuracy in automated assembly systems. Our goal was to show the potential of machine learning techniques in general and artificial neural networks in particular in automated assembly processes.

Error influences in automated assembly can be caused by a multitude of sources, many of which cannot be accounted for prior to the ramp-up phase. These error influences are the reason why adaption of assembly parameters during the ramp-up can be regarded as a critical process step. The labor-intensive selective search for error reasons and their elimination is critical. As an alternative approach, this work demonstrates how artificial neural networks can significantly reduce the impact of errors in automated assembly. For the investigation and demonstration, a camera-based pick and place process was set up. Then pick and place tests of the assembly system were recorded. For this, the pick positions and the corresponding assembly place deviations were measured. Based on this data, an ANN was trained offline. In our experiments, a deep feed-forward neural network was necessary to estimate the measured deviations with adequate precision. This trained network was used in order to generate additional offset values for the pick and place process.

### 4.1. ADVANTAGES AND DISADVANTAGES OF THE PRESENTED APPROACH

The presented approach is fast to implement and can potentially compensate any systemic error influence. 1000 process cycles of the assembly process, which could be acquired in a few hours, were sufficient to reach the desired training quality. In a final experimental validation run, the trained network was shown to reduce the measured deviations by one order of magnitude.

The main disadvantage is that an explicit error analysis does not take place. The description of the causes of the error is not necessary. An extensive understanding of causes of errors and possible explicit solutions is not necessary, thus learning effects regarding the design of the assembly plant become harder to realize.

### 4.2. OUTLOOK

The example presented is transferable to different pick and place processes, with different components and robots. Also conceivable is the improvement of precision assembly systems and assembly systems with more degrees of freedom. The potential usefulness of ANNs in industrial processes reaches far beyond these examples. Artificial neural networks could also be used in other robot-based industrial processes such as automated welding, bonding and 3D printing. For example, the robot paths and the associated measurements of the sweat, adhesive or 3D printing webs could then serve as training data for the ANN.

## REFERENCES


[1] BOLMSJO G., 2014, *Reconfigurable and Flexible Industrial Robot Systems*, Adv. Robot. Autom., 3/117, DOI: 10.4172/2168-9695.1000117.

[2] SCHERF, H., 2010, *Modellbildung und Simulation dynamischer Systeme, Eine Sammlung von Simulink-Beispielen*, l, Oldenbourg Wissenschaftsverlag, Available online at http://lib.myilibrary.com/detail.asp?id=609459.

[3] SCRIMIERI D., OATES R.F., RATCHEV S.M., 2015, *Learning and reuse of engineering ramp-up strategies for modular assembly systems*, J. Intell. Manuf., 26/6, 1063–1076, DOI: 10.1007/s10845-013-0839-6.

[4] GRAVEL D., ZHANG G., BELL A., ZHANG B., 2009, *Objective metric study for DOE-based parameter optimization in robotic torque converter assembly*, IEEE/RSJ International Conference on Intelligent Robots and Systems, St. Louis, MO, USA, 10.10.2009 - 15.10.2009, 3832–3837.

[5] SU Ch.T., CHIANG T.L., 2003, *Optimizing the IC wire bonding process using a neural networks/genetic algorithms approach*, Journal of Intelligent Manufacturing,14/2, 229–238, DOI: 10.1023/A:1022959631926.

[6] MONKMAN Gareth J., 2007, *Robot grippers*, Weinheim, Chichester, Wiley-VCH, http://search.ebscohost.com/login.aspx?direct=true&scope=site&db=nlebk&db=nlabk&AN=190005

[7] ARAI F., RONG L., FUKUDA T., 1993, *Trajectory control of flexible plate using neural netw*ork, Proceedings IEEE International Conference on Robotics and Automation, Atlanta, GA, USA, 2–6 May 1993, IEEE Comput. Soc. Press., 155–160.

[8] KOBER J., PETERS J., 2014, *Learning Motor Skills. From Algorithms to Robot Experiments*, Springer Tracts in Advanced Robotics, 97, http://dx.doi.org/10.1007/978-3-319-03194-1.

[9] ZHONG X., LEWIS J.N.N., FRANCIS L., 1996, *Inverse robot calibration using artificial neural networks*, Engineering Applications of Artificial Intelligence, 9/1, 83–93, DOI: 10.1016/0952-1976(95)00069-0.

[10] WEI Z.P., FANG G., 1999, *Model Predictive Control for Robot Manipulators Using a Neural Network Model*, Australian Conference on Robotics and Automation, 62-67, http://www.araa.asn.au/acra/acra1999/papers/paper13.pdf.
[11] CYBENKO G., 1989, *Approximation by superpositions of a sigmoidal function*, Math. Control Signal Systems, 2/4, 303–314. DOI: 10.1007/BF02551274.
[12] KUSIAK A., 1994, *Artificial Neural Networks for Intelligent Manufacturing*, Dordrecht, Springer Netherlands. https://ebookcentral.proquest.com/lib/gbv/detail.action?docID=3109078.
[13] WANG D., BAI Y., 2005, *Improving Position Accuracy of Robot Manipulators Using Neural Networks*, IEEE, Instrumentation and Measurement Technology Conference Proceedings, Ottawa, ON, Canada, 16-19 May 2005, 1524–1526.
[14] MAREŠ T., JANOUCHOVÁ E., KUČEROVÁ A., 2016, *Artificial neural networks in the calibration of nonlinear mechanical models*, Advances in Engineering Software 95, 68–81, DOI: 10.1016/j.advengsoft.2016.01.01.
[15] DAS A., 2015, *Introduction to Digital Image, Guide to Signals and Patterns in Image Processing, Foundations, Methods and Applications*, Springer International Publishing, 1–42, https://doi.org/10.1007/978-3-319-14172-5_1.
[16] GÉRON A., 2017, *Hands-On machine learning with Scikit-learn and TensorFlow, Concepts, tools, and techniques to build intelligent systems*, O'Reilly, http://proquest.safaribooksonline.com/9781491962282.
[17] CLEVERT D.A., UNTERTHINER T., HOCHREITER S., 2015, *Fast and Accurate Deep Network Learning by Exponential Linear Units (ELUs)*, Published as a conference paper at ICLR 2016, arXiv:1511.07289.