A Rapid Automation Framework for Applications on the Micro- and Nanoscale

Claas Diedrichs, Malte Bartenwerfer, Manuel Mikczinski, Sören Zimmermann, Tobias Tiemerding, Christian Geldmann, Ha Nguyen, Christian Dahmen and Sergej Fatikow

Division of Microrobotics and Control Engineering
University of Oldenburg, Germany
claas.diedrichs@uni-oldenburg.de

Abstract

Automation on the micro- and nanoscale requires special robotic approaches that differ from the established approaches in macro-robotic. Robotics on the micro- and nanoscale does not use kinematic chained systems, but direct actuation principles. Especially on the nanoscale reliable sensors for positioning are rare. Internal position sensor for linear or rotatory systems exist, but measure at the actuator base, which is normally several mm or cm away from the tool center point. Due to thermal drift, electrostatic charge and other effects on the small scale this positions are often not reliable enough. Therefore, optical and scanning electron microscopes are used as main sensor and hence automation on this scale requires a focus on object-tracking and perception based on visual information. As most automation processes on this scale are prototypic nature, a supporting framework should allow for rapid prototyping of image-processing and automation.

A new, free available framework is developed and deployed during several projects concerning automation on the micro- and nanoscale. The first part of this framework is a powerful drag and drop image processing system which is based on the OpenCV image processing library. The second part is a scripted automation system, based on the python scripting language. The system can be extended with different robotic systems, tools and sensors. The system was used in several projects, from assisting manual operations to semi- and fully-automated assembly steps. Six of these projects are presented and the usage of the framework in this projects is discussed.

1 Introduction

Automation on the micro- and nanoscale requires special approaches that differ from approaches used in the macro-robotic. For example other driving principles are used on the micro- and nanoscale [Kortschack and Fatikow, 2004]. These are direct driving principles and not kinematic chains as used in macro-robotic systems.

On the small scale, the most common used sensor is an optical system, often optical microscopes and cameras [Sievers et al., 2006; Sievers and Fatikow, 2006]. The scanning electron microscope is the most common optical system used on the nanoscale [Wich et al., 2009a].

Internal position sensor for linear or rotatory systems exist, but measure at the actuator base, which is several mm or cm away from the tool center point. Due to thermal drift, electrostatic charge and other effects on the small scale this positions are often not reliable enough for positioning tasks[Wich et al., 2009b; Diederichs et al., 2012]. Therefore, the previously mentioned optical and scanning electron microscopes are used as the main sensor and automation on this scale requires a focus on object-tracking and perception based on visual information. Furthermore the most automation processes on this scale are prototypic nature. A supporting framework should allow for rapid prototyping of image-processing and automation. The proposed framework has been developed during many projects concerning automation on the micro- and nanoscale. It is freely available and open-source for the windows, linux and Mac-OS platform 1,2.

The paper is structured as follows. In the next Sec. 2, the software framework for automation on the micro- and nanoscale is presented. Sec. 3 presents different applications on the micro- and nanoscale that where automated using the framework.

1http://automation.offis.de
2https://github.com/OFFIS-Automation/Framework
2 Software Framework for Automation on the Micro- and Nanoscale

To tackle the challenges mentioned in Sec.s 1, a software framework for rapid automation has been developed in the recent years. Several goals has been pursued driven by the following needs:

- Provide an easy to use image-processing system, to allow for object detection and position calculation without the need to learn a programming language.
- Provide a basis for vision-based automation tasks, that is easy to learn for simple automation and yet powerful enough for complex automation tasks programmed by experienced automation engineers.
- Encapsulate low-level functionality of tools, robots, sensors etc. into remote-controllable units (so called RC-Units). Each RC-Units offers only high-level commands for the automation.
- Allow for direct telecontrol of RC-Units using gamepads, joysticks or haptic interfaces. This enables the manual prototyping of assembly steps.
- Provide easy extensibility in order to include new custom RC-Units as well as problem-tailored image-processing solutions.

Fig. 1 shows the high-level architecture of the framework. The framework itself does not provide specific RC-Units (apart from the vision RC-Unit, described in Sec. 2.2). Instead, it provides software interfaces to add RC-Units as plugins into the framework. A main idea of this framework is that an RC-Unit already provides an abstraction to the hardware and encapsulate low-level functionality. For instance, a gripper should provide methods like open and close, instead of setVoltage or setCurrent. Thus, the operator does not need to know about the working principle of the RC-Unit and can automate a system by arranging this high-level operations. Hence, the entire system is easy to automate even with limited programming-language knowledge. Additionally, automation sequences can be easily adapted to other projects where e.g. a different gripper is used.

2.1 RC-Units

RC-Units are the interface between the framework and the hardware. They have two purposes: First, an RC-Unit is the hardware abstraction layer (HAL). Second, it implements low-level functionality and publishes high-level commands to the framework. RC-Units are developed in C++ programming language using the QT library\(^3\). To communicate with the framework, object methods written in C++ can be directly published. The framework takes care of the data conversion and the hook into the automation system. This makes it straightforward to implement custom RC-Units, as the developer can focus on implementing the HAL and the pure functionality.

Additionally, each RC-Unit can be directly controlled by telecontrol (e.g. gamepads, joysticks or haptic interfaces) by implementing and publishing object methods with a specific signature. This methods are called in a loop while the operator uses a telecontrol device.

2.2 Image-Processing System

One goal of the framework is to provide easy image-processing without the need of a programming language. Additionally, the image-processing system should be fast and reliable. The framework provides a plug-and-play image-processing system that is based on the well-established fast and robust image-processing library OpenCV\(^4\) [Bradski, 2000]. The OpenCV functions are wrapped into image-processing filters. Each filter has special purpose, and consists of several ports. For example, a simple thresholding filter has two input ports (input image and threshold value) and one output port (the thresholded image). Filters can be arranged to filter chains. In each filter chain, the filters are executed consecutively. The ports of the filters can be connected with each other, or can be set to fixed values, if possible. For instance, the input image of the above mentioned threshold filter must be connected to an output image of a different filter whereas the threshold value can be a fixed value. An example filter chain is shown in Fig. 2.

The framework supports multiple filter chains. Each filter chain is executed in its own CPU-thread, facilitating a real parallel execution on multicore systems. Additionally, it is possible to exchange data between different filter chains and to trigger filter chains; thus it is possible to create a pipelined image-processing.

\(^3\)http://www.qt-project.org

\(^4\)http://opencv.org
Figure 2: A simple filter chain that detects objects in a simple image. First, the image is thresholded. On the thresholded image, the contours of objects are detected. The last filter then computes the bounding boxes of the found contours.

Every output port of a filter can be visualized directly; therefore, a port is simply dragged to a visualization area. This way, not only the final result of the filter chain can be obtained, but also all intermediate steps can be observed. Additionally, values of input ports that are not connected to a filter can be changed inside the visualization area in a convenient way.

In this manner, complex image-processing chains can be created in a few minutes. This system allows rapid prototyping and testing of different approaches, especially if the image-processing task is not yet well known.

The vision system itself can also be extended by plugins. The vision-plugins are also written in C++. A vision plugin is a subclass of an interface class. It must define its input and output ports, and implement a single execute method. The input ports are assigned by the vision system before the execute method is called. A developer does not need to care about the data flow and type safety, and thus can concentrate on the image-processing code.

The vision system can be controlled by the automation system. This is realized by providing a vision RC-Unit. This RC-Unit publishes methods to retrieve data from the output ports and to set values on input ports. Additionally, filter chains can be paused and resumed. It is also possible to wait for a port to generate a value.

2.3 Scripted Automation System

The framework is designed to facilitate fast implementation of simple as well as complex automation sequences. Therefore it includes the free scripting language Python\(^5\), which is easy to use for simple sequences and a full-featured object-oriented programming language. The framework includes additional libraries for Python, making it possible to call methods of the connected RC-units as well as perform simple user interaction.

The user interaction module can be used to create simple graphical user interfaces to collect data from the user and to show progress information. Additionally, it is possible to collect user input using the vision system. For example, this can be used to obtain a position inside an image from the user by using mouse input.

Furthermore, the framework includes a debugger for the scripting language, thus making it possible to step through the automation sequence to control movements and calculated results.

The framework automatically extracts the documentation from each instantiated RC-Units. This documentation can help the user to implement the automation sequences.

At the beginning of a project, it is normally not specifically determined how to automate a particular task. Therefore, the framework includes methods for manual control of RC-Units.

2.4 Telecontrol

To allow manual control on the micro- and nanoscale, telecontrol of robots and tools is necessary. Telecontrol can also be used to obtain a better understanding of a process by manually testing, e.g. an assembly step.

The framework currently features two telecontrol frontends, gamepad and haptic interfaces. Gamepads can be used for manual positioning of robots or tools. Most operators are familiar with the use of gamepads. Therefore a very short learning period for telecontrol-manipulation is possible.

The framework gives the user the possibility to alter the gamepad behavior, e.g. it is possible to invert axes or add a multiplier to the gamepad joysticks to realize different speeds.

If an RC-Unit is able to give force-feedback, a haptic device can be used alternatively. The framework currently supports the Phantom devices from Sensable\(^6\). However, the framework offers a interface to add other haptic devices, which has been already been demonstrated [Bolopion et al., 2011; 2012].

Using a haptic device, the operator can feel the forces on the small scale, using not only vision feedback but also force-feedback.

On the RC-Unit (e.g. robot) side, a gamepad command is implemented as a set of methods. A special method is called when a gamepad button is pressed or

---

\(^5\)www.python.org

\(^6\)http://www.sensable.com/products-haptic-devices.htm
released. If gamepad joysticks are used, the RC-Unit receives the deflection of the joysticks normalized to [-1:1]. In this normalized value, the multipliers or inverts are already applied. The method receiving this deflection values is called in an infinite loop, as long as the operator presses an assigned dead-mans-control button.

For a haptic-device, an RC-Unit receives three position values from the frame work and, after the movement, returns three position and three force values. Using this technique, the RC-Unit can abstract from the haptic interface or gamepad and focus on the control of the hardware.

3 Applications
The framework described above was used in several projects, ranging from manual manipulation using a haptic interface on the nanoscale to fully automated cooperative handling of spheres on the microscale. It has been used also for assembly of nano-sized parts, for the testing of 2D materials (e.g. graphene) and for semi-automated characterization of wood fibres inside a scanning electron microscope. Another project was the semi-automated characterization of stick-slip drives.

3.1 Automated Assembly of Nanobits
In the NanoBits project exchangeable tips for atomic force microscopy (AFM) are developed. The AFM technique is a tactile based microscopy which uses ultra fine tips to scan surfaces of samples collecting topographic information. For this technique the tip radius is crucial, but obviously during its lifetime this tip suffers wear and fatigue. For this reason, a system is developed which allows to exchange the tip of this systems. The tips are stored in reservoirs, which are filled by microrobotic assembly [Bartenwerfer et al., 2011a]. These assembly steps are realized using several techniques offered by the automation framework.

In a fully automated process, Nanobits are detected, detached and placed into a reservoir. Fast image processing tools are applied to the visual feedback of an SEM image to detect relevant features inside the image, whereas an automation script controls movements of a gripper which is used for manipulation of the Nanobits. The detection of horizontal positions is implemented using cross-correlation based template matching. For the detection of the vertical position of the Nanobit, two approaches have been studied (Fig. 3). In both cases, the open gripper firstly is positioned above the Nanobit in a way, that the left part of the Nanobit is visually located between the gripper jaws. Secondly, the gripper is lowered in steps of 1 μm. In the first approach, it additionally performs small movements orthogonal to its gripping direction. When the gripper is on the same height as the Nanobit, these movements lead to a bending of the Nanobit, which is detected using template matching. In the second approach, the mean grayscale value of a region of interest (ROI) around the left side of the Nanobit is monitored. When the gripper approaches the Nanobit, electrons escaping the Nanobit are absorbed by the gripper material, leading to a reduced grayscale value inside the ROI. If the grayscale value drops below a pre-defined threshold, gripper and Nanobit are supposed to be vertically aligned and the Nanobit can be detached by closing the gripper.

Figure 3: Detection of the vertical position of nanobits: bending based detection (left) and grayscale value based detection (right).

For the placement of the Nanobit into the reservoir, the same grayscale based approach is used. At the moment when the tip is inserted into the reservoir, the brightness inside a ROI around the center of the reservoir is explicitly reduced. The gripper movement stops and releases the Nanobit into the reservoir.

3.2 Automated Testing of 2D Materials
Two-dimensional (2D) materials have become one of the most active research areas within the last decade. Since the first experimental fabrication of graphene and the demonstration of its extraordinary electrical characteristics in 2004, a variety of 2D materials has been discovered and studied. Reliable fabrication-, handling-, modification- and functionalization techniques of this new material class are still in its infancy or completely unknown and therefore, industrial exploitation of 2D materials is limited until today. Nanorobotic processing techniques are promising candidates to overcome these constraints, as they are not diffraction-limited and can be adapted to several application scenarios, e.g. nanolithography or mechanical and electrical characterization. However, only moderate effort has been spent on computerized control of nanorobotic processing of 2D materials until now, weakening the competitive strength of this technique. A first experimental approach of automated mechanical characterization of 2D materials using SEM based visual servoing has been presented recently [Zimmermann et al., 2013]. The idea of this approach is to determine the mechanical properties of a 2D material deposited onto a periodically perforated substrate. Due to the predefined shape of the used perforation holes, nanoindentation measurements can provide direct information on the local mechanical properties of the ma-
aterial such as Young’s modulus or pretension. Therefore, a fully automated sequence is developed to align a nanorobotic driven force sensor with sub micrometer accuracy and to conduct nanoindentation measurements within the center of each hole.

The automated nanoindentation uses both image-processing and automation. Image-processing is used with two processing pipelines that are dependent on the visual feedback of a SEM:

1. Template matching to detect the cantilever.
2. Detection of holes inside a Region of Interest (ROI).

To increase the reliability, the image data of the SEM is enhanced using supplied filters e.g. gaussian noise removal or binary morphology. Fig. 4 shows the results.

Data acquired by the image processing pipelines, e.g. the hole positions, are used by an automation sequence. The sequence moves the cantilever tip to the center of each detected hole. Then a touchdown of the tip to the nanomembran is performed, followed by a slowly indentation with a total distance of 150µm. The whole measurement sequence is monitored and the sensor data (e.g. position of the cantilever or force) is written to a file for further analyses with a frequency of 10 Hz. A single hole measurement can be done in eight seconds; resulting in a total measurement time of 16 minutes, which includes calibration and reference measurements [Zimmermann et al., 2013].

3.3 Semi-automated Testing of Wood Fibers

For classical, organized objects and structures on the microscale more automated systems have become available during the last two decades. However, un-organized and un-structured objects are challenges for current developments. Both, wood fibers and processed paper fibers have diameters in the range from 10 to 80µm and range in length from 0.6 mm to several mm. Their appearance ranges from rod-like (fresh wood fibers) to totally unordered, curly, and kinky fibers (processed fibres). Current models however, require validation data on single fiber level in axial, tangential or transverse directions.

One approach to assess the transverse fibre property of compression relies on a modular nanorobotic toolbox, called PS-AMiR [Mikczinski, 2011]. From this toolbox a coarse and a fine positioning module are used for the compression setup. The coarse positioning module carries the sensor and the fine positioning module carries the sample. Observation of the cross-section of the fiber under SEM is realized [Mikczinski et al., 2013] by placing the sample in an up-right position. Fig. 5 shows a pine fibre ready for compression.

Data acquired by the image processing pipelines, e.g. the hole positions, are used by an automation sequence. The sequence moves the cantilever tip to the center of each detected hole. Then a touchdown of the tip to the nanomembran is performed, followed by a slowly indentation with a total distance of 150µm. The whole measurement sequence is monitored and the sensor data (e.g. position of the cantilever or force) is written to a file for further analyses with a frequency of 10 Hz. A single hole measurement can be done in eight seconds; resulting in a total measurement time of 16 minutes, which includes calibration and reference measurements [Zimmermann et al., 2013].

3.3 Semi-automated Testing of Wood Fibers

For classical, organized objects and structures on the microscale more automated systems have become available during the last two decades. However, un-organized and un-structured objects are challenges for current developments. Both, wood fibers and processed paper fibers have diameters in the range from 10 to 80µm and range in length from 0.6 mm to several mm. Their appearance ranges from rod-like (fresh wood fibers) to totally unordered, curly, and kinky fibers (processed fibres). Current models however, require validation data on single fiber level in axial, tangential or transverse directions.

One approach to assess the transverse fibre property of compression relies on a modular nanorobotic toolbox, called PS-AMiR [Mikczinski, 2011]. From this toolbox a coarse and a fine positioning module are used for the compression setup. The coarse positioning module carries the sensor and the fine positioning module carries the sample. Observation of the cross-section of the fiber under SEM is realized [Mikczinski et al., 2013] by placing the sample in an up-right position. Fig. 5 shows a pine fibre ready for compression.

Data acquired by the image processing pipelines, e.g. the hole positions, are used by an automation sequence. The sequence moves the cantilever tip to the center of each detected hole. Then a touchdown of the tip to the nanomembran is performed, followed by a slowly indentation with a total distance of 150µm. The whole measurement sequence is monitored and the sensor data (e.g. position of the cantilever or force) is written to a file for further analyses with a frequency of 10 Hz. A single hole measurement can be done in eight seconds; resulting in a total measurement time of 16 minutes, which includes calibration and reference measurements [Zimmermann et al., 2013].

3.3 Semi-automated Testing of Wood Fibers

For classical, organized objects and structures on the microscale more automated systems have become available during the last two decades. However, un-organized and un-structured objects are challenges for current developments. Both, wood fibers and processed paper fibers have diameters in the range from 10 to 80µm and range in length from 0.6 mm to several mm. Their appearance ranges from rod-like (fresh wood fibers) to totally unordered, curly, and kinky fibers (processed fibres). Current models however, require validation data on single fiber level in axial, tangential or transverse directions.

One approach to assess the transverse fibre property of compression relies on a modular nanorobotic toolbox, called PS-AMiR [Mikczinski, 2011]. From this toolbox a coarse and a fine positioning module are used for the compression setup. The coarse positioning module carries the sensor and the fine positioning module carries the sample. Observation of the cross-section of the fiber under SEM is realized [Mikczinski et al., 2013] by placing the sample in an up-right position. Fig. 5 shows a pine fibre ready for compression.

The compression tests requires two tasks inside the SEM. First, the sensor needs to be placed in a parking position, from which the measurements start. This parking position is reached by teleoperated moving the sensor to the fibre and then both towards the substrate. Due to the flexibility of the fibre it tends to slip under the sensor. This movement needs to be compensated by the operator. Second, after reaching the parking position the measurement cycles start. Those measurement cycles can again be divided into two tasks, which are automated. First, a single approach is performed to determine the maximum travel towards a maximum sensor excitation. This preserves the sensor from overload in subsequent measurements. Second, the measurement cycles are performed. The operator chooses the number of loading and un-loading cycles and starts
the measurement. Now, the fibres are compressed with the previously determined maximum distance/travel and the results are recorded by the framework at a predefined sensor sampling rate.

3.4 Manual AFM Handling of Microspheres

The directed handling of individual microspheres offers a great possibility for the assembly of prototypic photonic crystals. These devices consist of periodic structures with different indices of refraction. This can be realized in different ways, but on one of the most straightforward approaches is the assembly of such components; this can be done by the directed ordering of spheres into a certain pattern. In such structures, the alternating index of refraction is given by the sphere itself and the surrounding air/vacuum. Essential for optical crystals is the requirement, that the periodic length is comparable to the wavelength of the light traveling through the crystal. Depending on this periodic length, wavelength depended devices such as gates, filters, and switches can be realized.

In the microsphere handling project, the automation framework has been used in order to assist the manual assembly of the spheres with the haptic interface and to assist the user in finding microspheres suitable for the assembly. The entire hardware part of the system consists mainly of a high-resolution SEM and a robotic handling system of six degrees of freedom realized by two different kinds of actuators. A force sensitive end effector is used as manipulator, at which the spheres tend to stick. Hence, pickup and deposition of the spheres is performed only by contacting and wiping off the spheres using adhesion forces. The automation framework offers the connection to a haptic interface for the user, which allows to perform precise and intuitive pickup sequences of individual spheres. Challenging for this task, is the update frequency of at least 100 Hz, which is mandatory in order to provide the user an smooth and continuous impression of the force. Furthermore, the automation framework processes the images of the SEM, detects and marks spheres which particular properties. Thus, the end effector can be positioned close to a suitable sphere by the automation framework and only the actual pickup sequence has to be done manually.

Figure 6 shows a typical preliminary result of a microsphere handling sequence towards an photonic crystal. The pre-patterned underlying substrate is already equipped with several spheres. The spheres are made of silica and have diameter of about 1 μm [Bartenwerfer et al., 2011b].

3.5 Automated Cooperative Sphere Handling by Mobile Microrobots

Robot-based microhandling and microassembly are well-established technologies in research and industry. The automated assembly of a microspectrometer is an example of such a process [Das et al., 2009]. A traditional prerequisite for assembly is the palletizing of the employed components. This extends to microassembly for many components such as microspheres that cannot be efficiently produced and transported in a well-ordered form.

Figure 7: Microrobotic setup with three mobile microrobots working cooperatively.

The goal of this project was the automated robot-based separation and palletizing of microcomponents [Jasper et al., 2011]. The microcomponents are initially assumed to be randomly located on a specimen holder. The components can be of mixed size, shape and quality. These components are then optically inspected and classified with an optical microscope. Objects of spe-
cific classes are selected, separated from other attached objects, picked up and moved to a target pallet. On this target, the objects are aligned in a regular grid with other objects of the same class. In order to be applicable for industrial processes, the operation needs to achieve a high throughput; a throughput of one operation per second is targeted. Two microrobots equipped with a microgripper and a sharp tungsten tip, respectively, can be effectively utilized to solve this task cooperatively [Jasper and Edeler, 2008]. The setup is shown in Fig. 7.

![Independent phases of the place process](image)

**Figure 8:** Independent phases of the place process

Based on the above described framework, the automated palletizing of two different microspheres initially lying in an unordered agglomerated structure was demonstrated. The achieved sorting throughput is in the range of one sphere per second while maintaining a placement accuracy of 3µm. The processes performed with the framework include image-processing to detect and classify the spheres. Other topics where collision avoidance and the cooperative control of two robots working together (see Fig. 8). The cooperative handling could be realized with the automation system; a main advantage is the fact that the movement abstraction is implemented in the robot's RC-Unit. This made the sequence of the cooperative handling straightforward.

### 3.6 Semi-automated Characterization of Stick-Slip Drive

Stick-slip drives have received much attention in recent years, since they have a noticeably simple design and very good working properties. The design of stick-slip drive is very simple consisting of a few parts and it can be therefore well miniaturized and controlled. Moreover, a stick-slip drive can exhibit movements with nanometer resolution in a scanning mode and also movements with large traveling in a stepping mode. These characteristics make it possible and attractive for applications on the micro- and nanoscale.

As presented in [Edeler, 2011], a stick-slip drive can be used to serve not only for positioning system, but also as a force generator. For a physical characterization of this drive, a teststand was developed whose schematic view is shown in Fig. 9. The purpose of this teststand is semi-automated experimentally physical characterization of the stick-slip drive, where the runner’s positions and generated forces depending on preload are automatically measured. In the automation framework, two main modules are used: sensor module and actuation module. The sensor module is used to get the runner’s position, generated force, as well as spring preload instantly. Based on this information, the actuation module is used to control piezoelectric actuators leading to the movement of runner. In this scenario, closed-loop as well as open-loop control can be used to characterize the drive.

**References**


