Michael Obermayr, BSc

Object Detection and Automated Motion Analysis for Single-Molecule Manipulations via STM

MASTER'S THESIS

to achieve the university degree of

Diplom-Ingenieur

Master's degree programme:

Technical Physics

submitted to

Graz University of Technology

Supervisor

Assoc.Prof. Dipl.-Ing. Dr.techn. Oliver T. Hofmann

Institute of Solid State Physics

Graz, June 2024

AFFIDAVIT

I declare that I have authored this thesis independently, that I have not used other than the declared sources/resources, and that I have explicitly indicated all material which has been quoted either literally or by content from the sources used. The text document uploaded to TUGRAZonline is identical to the present master's thesis.

Date, Signature

Declaration on the use of AI

The author declares the use of two different generative AI tools during the creation of this thesis with consent of supervisor.

Firstly, during code development in Python, the author used GitHub Copilot, an AI programming tool based on GPT-4 [1] [2]. It was employed throughout the code to generate small, specific code snippets to boost efficiency. No large fragments or approaches were generated. Most of these snippets served as a starting point and were further revised and tailored to meet the author's specific requirements.

Secondly, ChatGPT-3.5 was used to enhance the language and writing style of this thesis [3]. The model was tasked solely with improving provided draft text, and asked to not create new ideas or content. All generated output was carefully revised, altered, and rewritten before inclusion in the manuscript.

The author wishes to emphasize repeatedly, that generative AI was never used to obtain literature sources, ideas, theories, or original text content. Its use was limited to refining existing content or speeding up the coding process. The omission of these tools would not affect the quality of the thesis results but rather the efficiency of the working pace.

Abstract

Scanning tunnelling microscopy (STM) enables precise manipulation of single atoms and molecules on surfaces. Recent AI-driven advancements in manipulating arbitrary molecules pave the way for automatic assembly of artificial nanostructures, with advances in path planning, object detection, and process acceleration as further necessary steps.

In this work, we address the challenge of object detection as well as an approach to accelerate the process, based on tunnelling current analysis. While the first is tackled by developing a single-shot object detection pipeline, the employment of a neural network to predict molecular movements aims to reduce the need for frequent, time-consuming imaging between voltage pulses. A major challenge is creating a suitable dataset for neural network training, which we address using machine vision to automatically extract data from an extensive measurement set. Significant effort is spent on ensuring the swift adaptability of all developed algorithms to arbitrary molecules.

While the AI's translation prediction performance is not sufficient to fully replace imaging steps, detailed insights into single-molecule motion on surfaces can be derived from the dataset, allowing to study its rotational and translational behaviour. Consistency studies of the manipulation outcome under repeated experiment conditions highlight the stochastic nature of the process and enhance our understanding of the underlying mechanism

The object detection algorithm reliably identifies predefined object classes and its integration into a reinforcement learning-based framework is another significant step toward fully autonomous nanofabrication. Further adaptations could make it suitable for automatic on-surface synthesis with atomic precision.

Zusammenfassung

Mit Rastertunnelmikroskopie (STM) wird erstmals die präzise Manipulation von einzelnen Atomen und Molekülen auf Oberflächen möglich. Jüngste KI-getriebene Fortschritte in der Manipulation beliebiger Moleküle eröffnen nun die Möglichkeit eines vollautomatischen Aufbaus von künstlichen Nanostrukturen. Dafür müssen als nächster Schritt sowohl Module zur Wegfindung und Objekterkennung entwickelt werden, als auch der Gesamtprozess beschleunigt werden.

In dieser Arbeit befassen wir uns sowohl mit der visuellen Objekterkennung als auch mit einem Ansatz zur Prozessbeschleunigung, basierend auf einer Analyse des Tunnelstroms. Während erstere Aufgabe mit der Entwicklung einer Pipeline zur Single-Shot Objekterkennung gelöst wird, zielt der Einsatz eines neuronales Netzes zur präzise Vorhersage von Molekülbewegungen darauf ab, die notwendige Anzahl von zeitaufwändigen Bildaufnahmeschritten zwischen Spannungspulsen zu reduzieren. Dabei ist die Erstellung eines geeigneten umfangreichen Trainingsdatensatzes ein kritischer Schritt, der mithilfe maschineller Bildverarbeitung zur automatischen Datenextraktion bewältigt wird. Wesentliches Augenmerk liegt bei sämtlichen Algorithmen auf Generalisierbarkeit und schneller Adaptivität an beliebige Systeme.

Die Vorhersageleistung des vollständig trainierten Netzes ist nicht ausreichend um sämtliche Bildaufnahmeschritte vollständig zu eliminieren. Der automatisch extrahierte Datensatz erlaubt jedoch detaillierte Einblicke in die Bewegung einzelner Moleküle auf Oberflächen und deren Rotations- und Translationsverhalten. Der Vergleich von Molekülbewegungen unter wiederholten experimentellen Bedingungen liefert Einblicke in die Konsistenz und Stochastizität der ablaufenden Prozesse und hilft, die zugrundeliegenden Mechanismen besser zu verstehen.

Die Objekterkennungs-Pipeline identifiziert und labelt zuverlässig Objektklassen und stellt einen wertvollen Baustein zur vollautomatischen Nanofabrikation dar und könnte nach Weiterentwicklungen auch zur Automatisierung von Oberflächensynthese mit atomarer Präzision genutzt werden.

Danksagung

Nur zwei Namen finden sich auf der Titelseite dieser Arbeit, doch deutlich mehr Menschen haben indirekt Kleines und Großes beigetragen. Zuallererst möchte ich meinem Betreuer Oliver T. Hofmann danken für die Freiheit, Ideen und Ansätze frei verfolgen und entwickeln zu können. Trotzdem konnte ich mir jederzeit Input und Feedback holen und fühlte mich stets gut unterstützt. Gleichzeitig möchte ich meinem Zweitbetreuer Bernhard Ramsauer danken, der bei vielen kleinen Problemen meine erste Ansprechperson war. Dafür, dass er sich wunderbar leicht von seiner Arbeit ablenken ließ, um mir mit meiner zu helfen und viele Stunden verbrachte, mit mir Dinge zu diskutieren und zu lösen. Ebenfalls möchte ich mich bei der ganzen Arbeitsgruppe bedanken, nicht nur für die fachliche Unterstützung, sondern vor allem für ein superangenehmes Umfeld, in dem ich mich von Tag eins an wohlgefühlt habe. Für die vielen Kaffee- und Eis-Pausen, den Kebab-Montag, das gemeinsame Kochen, die ganzen Filmabende, die Kletterabende, fürs Kart fahren und für die Konferenzwoche in Berlin. 10/10, would recommend.

Danke an meine Freunde hier in Graz, insbesondere meiner WG-Familie, für so viele lustige Tage, Abende und Erlebnisse. Die letzten Monate und Jahre waren unglaublich wundervoll und prägend. Besonders dankbar bin ich meinen Eltern für ihre stetige Unterstützung, nicht nur während des Erstellens dieser Arbeit, sondern während meines gesamten Studiums. Ihr habt mir vieles so leicht gemacht, mich nie gedrängt und mir immer fest beide Daumen gedrückt. Zuletzt möchte ich meiner Anna danken für deine Zeit in Nähe und Ferne, für deine Liebe und deine Geduld. Mein Herz gehört dir.

Contents

1 Introduction

1.1 Motivation for the thesis

Perpetual technological advances in miniaturization and manufacturing capabilities opened up the possibilities of smaller and smaller devices, particularly in semiconductor technology. The pinnacle of miniaturization lies within utilizing single atoms and molecules as structural building members for devices, which is still a far stretch from the conventional approach of lithographic methods. Scanning tunnelling microscopy (STM) on the other hand offers, besides its sub-atomic resolution imaging capabilities, the possibility to manipulate single atoms or molecules on a substrate surface.

As demonstrated in the 1990s, small structures consisting of multiple particles can be assembled with accurately using STM [4]. However, this process is time-consuming and becomes even more challenging when dealing with molecules due to their directional dependencies, influenced by factors such as geometry and polarizability. The behaviour also depends on the tip shape and is nontrivial to derive from theory. Assembling larger and more complex nanostructures manually becomes impractical due to increased assembly time and increased need in precision. Hence, there is a need to replace manual manipulation with automated machine-guided steps.

For this, we can envision a complete framework to assemble large nanostructures on surfaces in a fully autonomous manner. Requiring merely a blueprint of the desired structure, this framework should be able to position all particles on the target sector of the surface precisely and reliably and evaluate its performance along the way. Additionally, we demand adaptability for arbitrary molecule species on arbitrary metallic surfaces to open up a large field of applications. While the manipulation of a multitude of molecule species is conceivable, we start focusing for demonstrational purposes on a single species.

The necessary building block for this framework can be grouped into three categories (see also Fig. 1):

- Automated manipulation of single particles:
  This is the core functionality, as each assembly of large structures can be broken down to single manipulation steps of particles. Here, we can build upon the work in [5], where the fully automatic manipulation of an unknown molecule on a Ag(111) surface with the use of Reinforcement Learning was demonstrated. The algorithm ensures rapid and reliable translation and rotation of arbitrary molecules, which meets our requirements for manipulation speed and adaptability. Thus, this block has already been completed successfully.

- Detection of surface objects
  Before assembly, all objects on the target sector must be located and differentiated by their species automatically. Object detection techniques can be applied here, which also allow for the tracking the assembly progress at intermediate steps as well as the confirmation of completion at the end of the process.

- Assembly Instructions
  As a last essential block, the framework needs to process the gathered information,

consisting of the blueprint and the location of all target molecules and obstacles. Generating collision-free paths for molecules to their respective goal positions involves path finding algorithms, optimizing routes for faster structure assembly.

While these three building blocks are essential for automatic nanostructure assembly, the incorporation of efficient feedback loops to assess outcomes and understand quantum mechanical processes enhances speed and comprehension. The thesis focuses on developing and investigating such a feedback loop, which also involves touching the subject of Machine Vision. The feedback variable for this loop will be the tunnelling current, as was already suggested in [6].

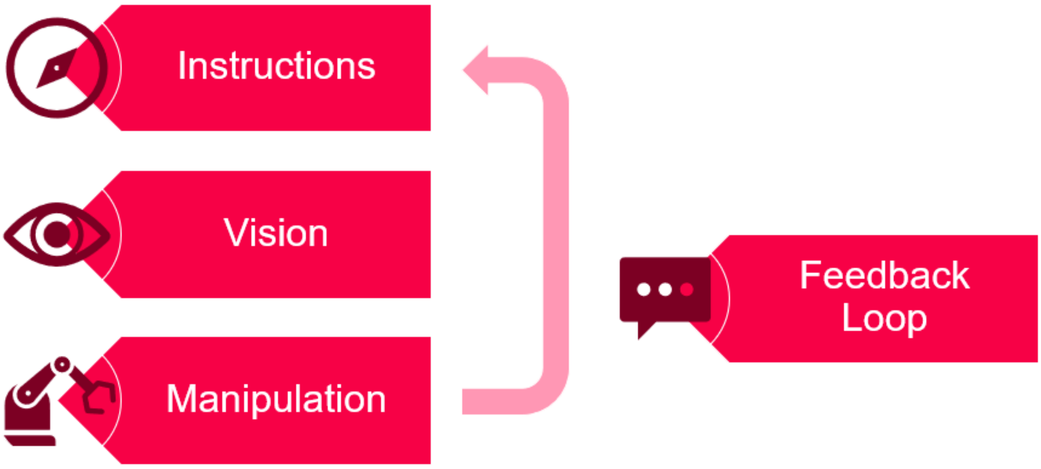


Figure 1: The fully autonomous assembly of artificial nanostructures requires the three main building blocks shown here. Additionally, a feedback loop is beneficial to comprehend the physics and potentially increase the assembly rate.

2 Fundamentals and Background

2.1 Surface Manipulations with STM

The development of Scanning Tunnelling Microscopy (short STM) in the 1980s marked a milestone, enabling the imaging and observation of metallic surfaces and adsorbed objects on there with atomic resolution [7]. Subsequent experiments demonstrated precise rotation and translation of single objects of surfaces, exemplified by the iconic ‘quantum corral’ micrograph, where a single electron confined within a two-dimensional ring of adatoms can be observed [4]. In further experiments, multiple modes of translating adatoms or molecules were described [8–10]. ‘Vertical translation’ is one of these modes, where the STM tip is located near an adsorbed molecule, which can be picked up with a voltage pulse and then transferred to another location with a second pulse, while traversing over obstacles and surface defects [8]. However, these movements are not uniform and exhibit inherent stochastic behaviour, varying across different surface-adsorbate systems.

Thus, deterministic algorithms do not seem like promising solutions for automating manipulations, as they require an understanding of the molecule behaviour in order to perform meaningful actions, which is not given for a new unknown system [5]. Machine Learning approaches offer greater potential, as they can infer this behaviour gradually. Recently, such approaches have found applications in various related applications, like in-situ tip conditioning or manipulating atoms, molecules or nanowires over surfaces using scanning probe microscopes [5, 6, 11, 12]. Throughout these manipulations, tunnelling currents are recorded, which are thought to contain valuable information about the manipulation outcome [6].

2.2 Preceding work and data origin

The present thesis strongly builds upon data collected in two preceding experimental campaigns, where Reinforcement Learning was employed to manipulate single molecules autonomously and optimally with an STM tip.

2.2.1 Manipulation of DDNB on Ag(111)

In [5] the manipulation of molecules of 2,5-di(ethynyladamantanyl)-4-(dimethylamino)nitrobenzene (DDNB) adsorbed on a Ag(111) surface was optimized automatically. DDNB was chosen for its high mobility while adsorbing with its NO_2 group specifically on Ag(111) top sites and its strong internal dipole, which enables precise translation and rotation, as demonstrated elsewhere [13–15]. It can adsorb in six distinct orientations on the surface, each differing by a 60° rotation.

The reinforcement learning agent could chose one of 15x15 possible STM tip positions, arranged in a grid projected on the XY plane of the adsorbed molecule, approached the chosen position in constant current mode and lowered the vertical height by 1 Å before applying a voltage pulse to induce movement. This grid was sampled non-uniformly as the agent emphasized actions with a larger translation towards some goal position due to a higher value of the reward function. The agent's performance was tracked via counting the number of required manipulation steps along a predefined square racetrack with a length of 40 nm. Eventually this track could be reliably completed within 20 min in roughly 60 manipulation steps, averaging 0.63 nm per step.

Two notable observations were made: Firstly, the success rate of a single step, defined by whether the molecule moved towards the goal, did not rise above 80%, even after extensive training. his raises questions about whether the Reinforcement Learning approach itself falls short of achieving full accuracy or if the quantum mechanical nature of the process limits the reliability of manipulations due to its inherent stochastics. Secondly, the process remains time-consuming for large structures of possibly hundreds of particles. Consequently, further acceleration warrants investigations to ensure viable time frames.

The raw measurement data with about 12,000 manipulation steps recorded from this work serves as the foundation for developing a feedback loop based on tunnelling current and an automated statistical analysis of molecule movement. The raw data contains image snippets of the molecule between manipulation steps, time series of the tunnelling currents as well as manipulation files with all involved parameters. Notably, the image snippets have a relatively low resolution, a deliberate choice to expedite steps for the original purpose. 45 % of the images are of the size 64x64 pixels with 0.106 nm per pixel and the other 55 % of 32x32 pixels with 0.212 nm per pixel. This poses a hard restriction for location accuracy.

2.2.2 Manipulation of H_2Pc on Ag(111)

In the second relevant work, phthalocyanine (H_2Pc) molecules were manipulated on an Ag(111) surface with an adapted Reinforcement Learning framework [16]. H_2Pc is an non-polar molecule with a 1-fold mirror symmetry, adsorbing in six distinct orientation states on specific lattice sites.

No tunnelling currents were recorded and saved during the experiments, rendering it unsuitable for the desired feedback loop. However, the raw data from the 1200 manipulation steps can be used for statistical movement analysis to compare observations with DDNB for generality. Additionally, this dataset contains a significant number of overview images featuring multiple instances of various object classes, making it well-suited as a sandbox problem for object detection frameworks.

2.3 Object Detection

2.3.1 Problem and datasets

The goal of object detection is to detect all occurrences of predefined object classes within an image and determine as their location by estimating axis-aligned boxes [17]. This differs from image classification, where classes are assigned to the image as a whole. Modern object detection models are predominantly based on Deep Learning and necessitate as such substantial amounts of labelled data for effective training via supervised learning. Numerous datasets are publicly available for this purpose, with one of the most challenging and widely used being the Microsoft Common Objects in Context (MS-COCO) dataset [18]. This dataset consists of ca. 330.000 images, each containing on average 7.7 instances of 91 object classes in their natural context.

2.3.2 Metrics

The standard criterion for evaluating the accuracy of individual object predictions is Intersection over Union (IoU). IoU represents the ratio of the overlap area to the union area between the predicted bounding box and the ground truth bounding box. Correctly categorized objects with an IoU greater than a predefined threshold value (usually between 0.5 and 0.95) count as True Positives and those with lower IoU as False Positives [17]. Objects, that are not associated with a detection during evaluation count as False Negatives. Subsequently the statistical variables Precision and Recall can be computed separately for each object class:

$$Precision = \frac{TruePositive}{TruePositive + FalsePositive} \tag{1}$$

$$Recall = \frac{TruePositive}{TruePositive + FalseNegative} \tag{2}$$

For each class Precision and Recall are calculated for all predictions, ordered in descending order of confidence and then plotted against each other. The area under this precision-recall curve is referred to as Average Precision (AP) [19]. Averaging the AP across all classes leads to the mean Average Precision (mAP), which is the standard metric to compare the model performance. The mAP is often subscripted with a number, indicating the IoU threshold used in the calculation.

2.3.3 Object Detection Algorithm 'You Only Look Once'

Traditional neural-network based models divide the object detection task by first identifying object-like regions and then categorizing them [17]. However, newer models can perform these tasks simultaneously, leading to the emergence of Single Stage Detectors. One notable milestone in this evolution was the introduction of 'You Only Look Once' (YOLO), a model that achieved state-of-the-art accuracy while significantly reducing inference times compared to similar powerful models. Subsequent iterations of the YOLO framework, such as YOLOv7 and the latest version, YOLOv8, have continued to push the boundaries of performance. In 2023, YOLOv7 outperformed all known real-time object detectors in terms of both speed and accuracy (mAP) on the MS COCO dataset [20]. The most recent iteration, YOLOv8, further improves upon its predecessors in both speed and accuracy [21]. Fig. 2 illustrates the mAP comparison of various YOLO models evaluated on the MS COCO dataset, with YOLOv8 (represented by the blue line) demonstrating superior accuracy and speed across all size variants. Consequently, YOLOv8 stands out as one of the most powerful frameworks for object detection tasks.

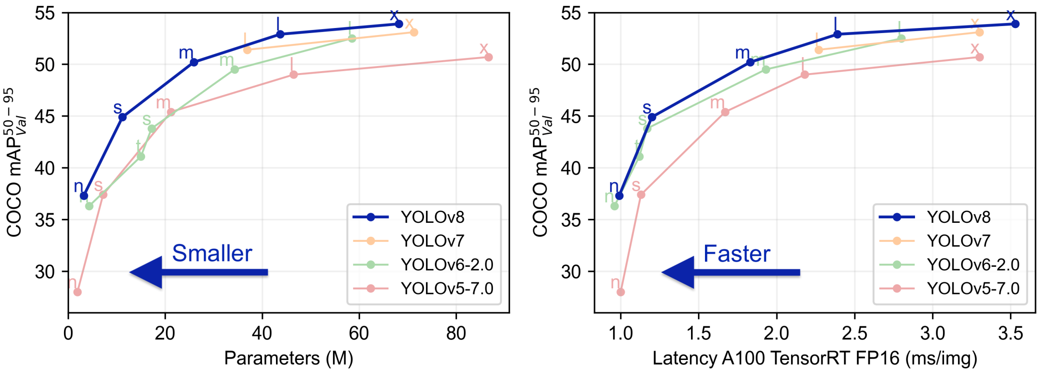


Figure 2: Comparison of mean Average Precision with IoU cutoffs ranging from 0.5-0.95 on the MS COCO validation dataset (COCO mAP50-95Val) for various YOLO object detection variants. The left plot illustrates the mAP versus the corresponding number of model parameters. On the right, the trade-off between speed (latency on an A100 TensorRT FP16 GPU) and accuracy for the same model is depicted. Each model version is represented by a unique colour, with lowercase letters indicate the respective size variant (n = nano, s = small, m = medium, l = large, x = extra large). Figures sourced from [22]

The performance of object detection models typically benefits from using larger training datasets. However, in some scenarios the data availability is limited and various approaches have been developed to achieve high accuracy with minimal data. One effective technique is Fine-tuning, which involves initially training a model on a large set of base classes (e.g. MS COCO). Subsequently, the model is fine-tuned on a target dataset that includes the new classes specific and relevant for the application [19]. By deploying Fine-tuning in conjunction with data augmentation methods, such as image rotation or flipping, it becomes possible to achieve robust performance with only a small set of labelled data.

3 Methodology

Most of the work was carried out in Python3 using a number of publicly available libraries, including *Opencv* for image manipulations and *Keras* for implementing artificial neural networks. The developer's repository is currently not publicly accessible.

3.1 Detecting Objects in STM images

As previously introduced in Sec. 1, full automation in single-molecule manipulations necessitates the ability to locate and identify objects and obstacles on the surface without human intervention. Machine Vision techniques play a crucial role in this process by analysing recorded scanning tunnelling microscopy (STM) images and providing feedback to the manipulation algorithm. To address this requirement, our approach involves two distinct levels of Machine Vision integration.

Firstly, the automatic manipulation framework needs a high-level overview of the surface to locate targets and obstacles for pathfinding and progress tracking, depicted by the second building block in Fig. 1. For this purpose, existing object detection algorithms can be adapted and applied on overview micrographs with relatively low magnification factor. The goal is to develop an algorithm that can quickly adapt to specific systems with minimal annotated training data, ideally achieving single-shot detection capability.

Secondly, object detection is essential for the feedback loop, introduced in Sec. 1 and illustrated in Fig. 1 to automatically assess the manipulation outcomes. Here, our objective is to analyse small surface snippets with high magnification factor to precisely determine the position and rotational state of an adsorbed molecule. Given the relatively uncrowded and simple nature of these snippets, classical Machine Vision techniques such as Edge Detection can be employed instead of Deep Learning Models. This reduced complexity and enhances algorithm comprehensibility while still achieving accurate object detection.

3.1.1 Developing an Object detection pipeline with YOLOv8

To acquire overview object detection capabilities, our aim is to precisely locate and identify objects within an overview image using only a single annotated training image, ensuring swift adaptability.

For demonstration purposes we employ the YOLOv8 object detection framework on a series of images with phthalocyanine adsorbed on a Ag(111) surface, stemming from unpublished experimental data [16]. YOLOv8 was selected due to its user friendliness and superior performance as outlined in Sec. 2.3.3. To achieve single-shot detection capability, we employ a pretrained model as a starting point and fine-tune it using a single annotated dataset multiplied by Data Augmentation techniques.

To simplify the task, we define three object classes that the algorithm should locate and identify:

- Adsorbed phthalocyanine molecules, with a distinctive four-leaf clover shape

- Silver adatoms, scattered across the surface due to repeated tip forming
- Defects, including both substrate lattice vacancies and impurities such as CO molecules

To evaluate the performance, a set of STM images was manually labelled using the web tool 'Roboflow' to define 'ground truth' labels. However, it's important to note that due to the inherent ambiguity in some images, this 'ground truth' may contain minor inaccuracies. In this labelled process 143 images were annotated, totalling 2314 objects (with on average 16.4 per image).

Following this dataset creation, a streamlined testing pipeline for the object detection was set up. As shown in the schematic in Fig. 3, the pipeline consists of three main modules. The first is a custom labeling tool for annotating one or multiple input images with bounding boxes to provide the original data for fine-tuning. These labelled images are then passed through the augmenter module, where random transformations are applied to introduce variability. This augmentation process enables a single input image to be expanded into a training set containing, for example, 1000 annotated images. These augmented images are subsequently used to fine-tune a YOLOv8 model, pretrained on the MS COCO dataset, to recognize the three defined object classes.

To provide ease of use and rapid adaptability the entire pipeline is controlled by a .yaml configuration file, which defines all major parameters and data directories.

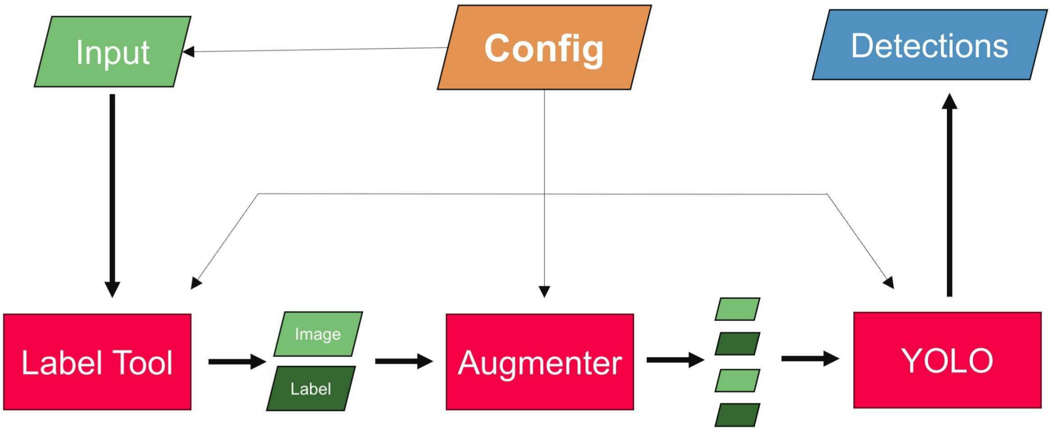


Figure 3: Schematic of the object detection pipeline utilizing YOLOv8. The configuration file (orange block) controls all parameters of the three modules (red blocks) and defines the input images for training (greed block at the top left) as well as image directories to finally compute object predictions (blue block).

3.1.2 Overlap Maximization algorithm

Second to the object detection algorithm described above, it is crucial to automatically analyse small STM snippets, each containing a single molecule and their most commonly clean surface surroundings. Given the simple nature of these snippets, an edge-detection based algorithm can be used, offering the additional advantage of transparent functionality. The primary aim is to compare two successively recorded images to extract changes in position and rotational state initiated by the STM tip pulse. For this aim we state two additional requirements: maximizing accuracy in translation and rotation determination within the limitations posed by image resolution and ensuring the algorithm's adaptability to arbitrary molecule shapes.

The algorithm performs three major steps. Firstly, the STM snippets are preprocessed, by background removal utilizing Otsu's method for automatic threshold determination [23], normalization to 8-bit pixel values (0-255) and applying a clustering algorithm to remove all but the largest cluster of nonzero pixels, to remove any occurring contamination on the surface images.

The second step, while optional, significantly enhances the algorithm's speed by implementing computationally inexpensive steps. Treating each pixel as a particle with mass based on its greyscale value allows computation of the gravitational centre of mass and the main axes for the moment of inertia tensor. Comparing these values between the two images provides rough estimates for the translation vector and rotation angle.

The third step constitutes the core of the algorithm, where the overlap of the subsequent images is maximized, starting from the translation and rotation estimates, by shifting and rotating one image, with any empty pixels padded with zeros. The overlap is calculated by summation of the element-wise product of the inverted pixel values and subsequent normalization. This inversion accentuates object edges, thereby facilitating more precise localization. The shift and rotation values yielding the largest overlap is the sought-after translation and rotation values between the two images.

This overlap-maximization algorithm is performed twice. The first execution occurs between subsequent images to extract the translation vector and rotation angle. The second execution is between each image and a uniform predetermined reference image to determine the absolute initial orientation state relative to the reference orientation. This redundancy allows to verify, whether the initial orientation and rotation angle add up to the subsequent initial orientation. Any found inconsistencies point to erroneous results of the overlap maximization algorithm. In the DDNB dataset, 7% of the points showed inconsistencies, primarily in the lower resolution 32x32 pixel images. As manually verified, most angle inconsistencies arise from orientation angles being wrong by 180°, resulting in pairs of inconsistencies for each error. Using a simple correction algorithm to adjust a single orientation angle for each inconsistency pair reduced the inconsistency rate from 7% to 0.3%. Manual inspection confirmed flawless error correction for each adjustment, although this correction may not be applicable for every molecule.

The manual inspection of randomly sampled data points confirms the reliability of the algorithm. It is designed to be adaptable to general molecules with arbitrary shapes and number of orientation states. However, for symmetric molecules, ambiguities arise, as orientation states become indistinguishable. Rotation angles must be either grouped into a reduced number of orientation groups or entire actions duplicated while changing orientation and rotation to the corresponding mirror-symmetric angles. The algorithm can be employed for molecule snippets with arbitrary pixel size and colour depth, the here utilized datasets contain greyscale images with dimensions of 64x64 pixels. As in both datasets, the molecules can adsorb in six possible states, only integer multiples of 60° need to be considered for rotations during overlap maximization. Due to its 2-fold mirror symmetry, the H_2Pc molecule (symmetry point group D_{2h}) in the second dataset required condensing the six orientation states into three groups.

3.1.3 Relevance and determination of the pivot point

2-dimensional objects moving in a 2-dimensional isotropic plane do not exhibit a distinctive point of motion, as translation symmetry applies. Rotations, however, disrupt this symmetry, leading to translation shifts when the rotation point is displaced.

Let us exemplify this by considering the movements of the tiled object in Fig. 4 on a XY grid. Starting at the far left, the object is rotate around two distinct centres, depicted by a red and orange cross. Obviously, the final orientation coincides but the location does not. The location difference $\vec{\Delta t}$ can be calculated with the following formula:

$$\vec{\Delta t} = \vec{\Delta p} - \hat{R}\vec{\Delta p} \tag{3}$$

Here, $\vec{\Delta p}$ is the vector between the two rotation centres and $\hat{R}$ is the two-dimensional rotation matrix.

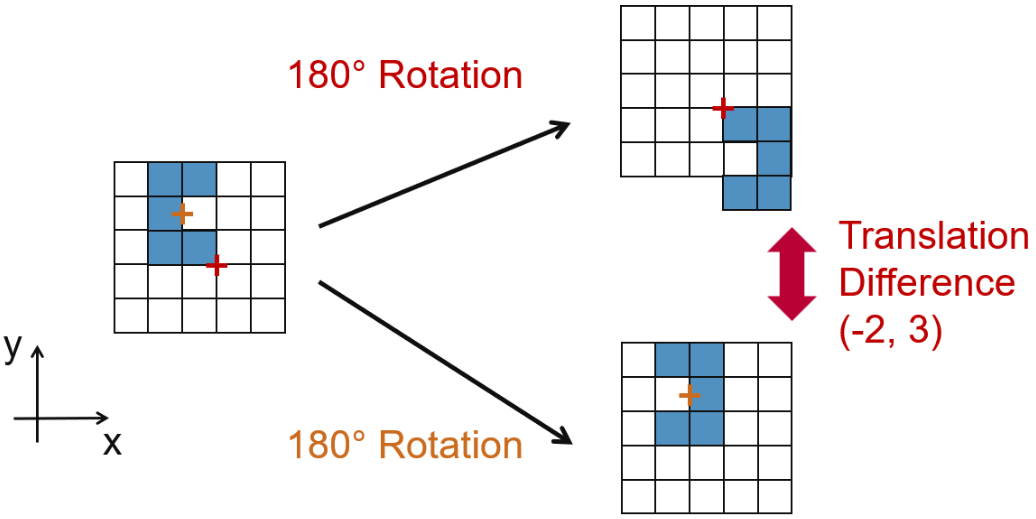


Figure 4: Starting from the left, the tiled object in blue is rotated around two different points in the 2-dimensional plane twice by 90°, leading to different locations on the grid.

For the system at hand, DDNB on Ag(111), previous studies found that NO_2 adsorption group acts as a specific anchor point and pivot point for rotations [15]. Thus, it is imperative to use this pivot point as the actual point for rotations in the algorithm in 3.1.2, to avoid the introduction of virtual translations, which do not happen in reality. Alternatively, the rotation point can be shifted retrospectively by transforming translation outcomes with equation 3, which was done during the motion analysis in Sec. 3.2 to refine the experimentally determined pivot point.

3.2 Motion Analysis

The automatic data extraction detailed in Sec. 3.3.1, utilizing the developed overlap maximization algorithm, allows for the rapid condensation of large STM manipulation images and data into datasets containing only essential manipulation parameters and outcomes. Beyond predicting manipulation outcomes from tunnelling currents and relative STM tip positions, this dataset can be used to study movement behaviour. Statistical analysis of the action outcomes is performed alongside a spatial dependence investigation to correlate movement behaviour with tip position during manipulation. Additionally, a consistency study compares actions with highly similar manipulation parameters. These analyses provide insights into single-molecule movement behaviour and the underlying physical mechanisms.

The efficiency of the data extraction and analysis enables large-scale statistical analysis of single-molecule movement behaviour with swift adaptation to arbitrary surface-adsorbate systems. It is therefore a valuable tool for analysing and understanding numerous molecules in future work in detail.

3.3 Tunnelling current based feedback loop

In our pursuit to enhance both the comprehensibility and efficiency of the single-molecule manipulation framework, we aim to integrate a feedback loop based on the tunnelling current, as proposed in [6]. The tunnelling current, a readily measurable physical quantity, is influenced by the interaction between the STM tip and the substrate and present adsorbates. Our central hypothesis is that this tunnelling current comprehensively encodes the processes occurring beneath the tip, containing all requisite information for their description. Specifically, we examine whether the relative motion of a single molecule is entirely encoded in the tunnelling current curve and can be extracted. In combination with the precise knowledge of the initial position and orientation, the final state of a molecule could be predicted. This would remove the need to image the surface after each manipulation step, allowing for considerable time savings in the overall manipulation process.

To explain this further, let us refer to Fig. 5, which illustrates the conventional three steps of single-molecule manipulation. An initial imaging step is necessary to precisely determine location and orientation of the molecule, designated to be moved. Building upon this information, optimal manipulation parameters are selected by the Reinforcement Learning agent and realized in a voltage pulse, which induces movement. Subsequently, a second imaging step is requisite to evaluate the manipulations success, providing feedback for Reinforcement Learning and utilizing the new position and orientation as a basis for subsequent manipulations. As depicted in 5, the two imaging steps are time-intensive compared to the manipulation itself, as the STM tip must scan the entire area of interest pixel by pixel. If the tunnelling current proves capable of predicting manipulation outcomes precisely, the fully trained agent could skip these imaging steps, which strongly accelerates the molecule assembly process. In the DDNB system for instance, imaging took approximately 30 seconds, while a manipulation step ranged from 0.5 to 6 seconds, which translates to a potential acceleration factor of more than one order of magnitude.

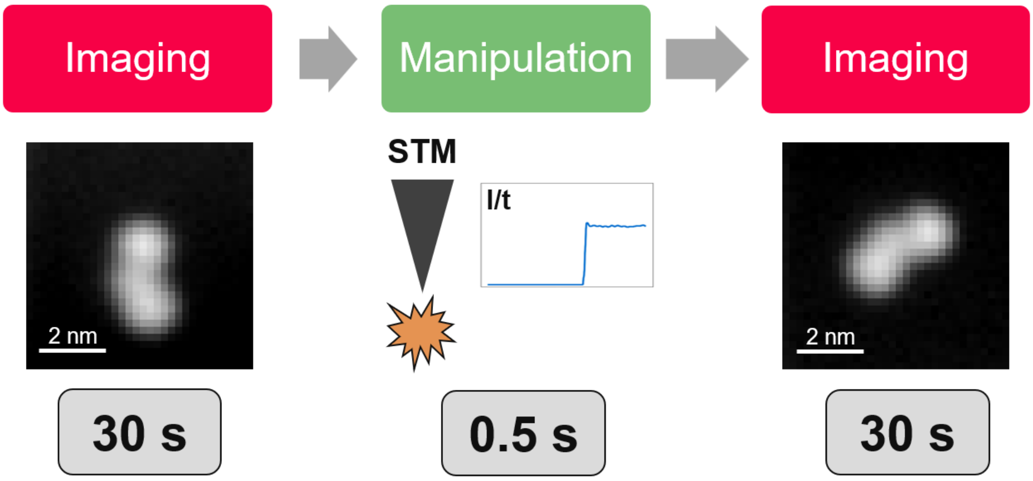


Figure 5: The manipulation process can be divided into three distinct steps: Two imaging steps to determine the exact location and orientation of the molecule for precise manipulation as well as to assess the outcome and the actual manipulation step in between, where the tunnelling current is recorded continuously (denoted by I/t). The typical required time in seconds for these steps is illustrated at the bottom.

To correlate the tunnelling current with the manipulation outcome, the raw measurement data must be condensed to a comprehensive dataset comprising all relevant manipulation parameters, along with the respective translation and rotation outcomes. Automating this data extraction is imperative, as the measurements encompass 10,000 data points.

3.3.1 Automatic data extraction and structuring

We combine the overlap maximization algorithm from Sec. 3.1.2 with automatic datafile readout to automatically compile a comprehensive dataset containing all relevant parameters and variables for our investigations from the large set of raw measurement data from [5] and [16]. As shown in the schematic in 6 for each manipulation image snippets of the molecule before and after are bundled together with the tunnelling current time series and manipulation files to form a data point.

The overlap maximization determines the pixel-precise location of the molecule as well as its rotational state, which is combined with the absolute coordinates in the microscope system to compute the 2-dimensional vectors of the relative STM tip position and translation as well as scalar values for the initial orientation state and the rotation angle. While the translation vector is combined with the rotation angle to form a 3-dimensional

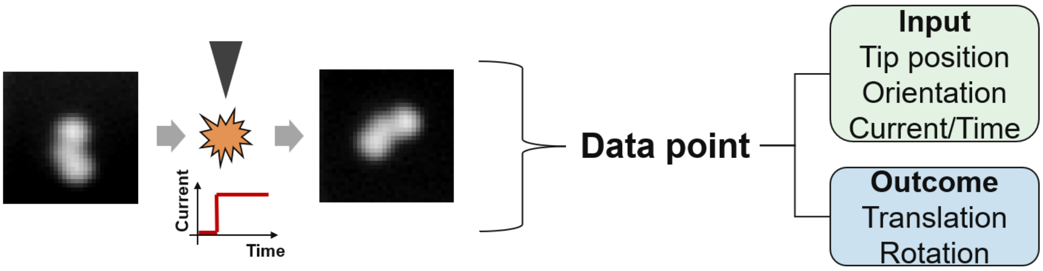


Figure 6: Each data point in the measurement data is extracted from two STM images before and after manipulation and the corresponding manipulation files. Image analysis with direct file readout automatically compiles a dataset, where each point consists of a set of input and outcome variables.

outcome state, the relative tip position and the orientation angle are appended with the tunnelling current times series of length 316 to form a 319-dimensional input state. Additional input parameters like the total time of the voltage pulse could be added to test for encoded information resulting in more accurate predictions.

Notably, for an isotropic system, the initial orientation angle could be eliminated by appropriately transforming the tip position and the translation vector. However, this reduces the prediction performance for the DDNB dataset, indicating some anisotropy on the Ag(111) surface, which does not follow 6-fold rotational symmetry.

Importantly, the data extraction process is quickly adaptable to various surface-adsorbate systems, given similar structuring of the raw measurement data. For example, adapting from the DDNB to the H_2Pc raw data and subsequent execution of all algorithms took less than an afternoon.

3.3.2 Current Analysis with Neural Networks

Tunnelling currents in STM experiments are influenced by manifold of static and dynamic parameters, involving surface and tip geometries and polarizabilities, to name just a few. Formulating an empirical or even analytical mapping between adsorbent movement and tunnelling current is highly complex and not straightforward, even with large amounts of labelled data at hand. However, this can be solved with Artificial Neural Networks (ANN), automatically find nonlinear correlations, given enough training data.

The idea is to feed the input portion of previously extracted data points into an artificial neural network (ANN) to predict the corresponding outcomes. While a simple fully connected network could be employed, mixed-input neural networks are more efficient. In these networks, the tunnelling current time series is separated from the input data and processed through a convolutional network before rejoining in a combined network to predict outcomes, as shown in Fig. 7. This is advantageous because convolutional networks excel with image-like data, such as time series, which exhibit near-range order, while requiring significantly fewer model parameters.

The two outcome variables differ distinctly. The 2-dimensional translation vector takes continuous values due to image resolution limits, even though the DDNB molecule adsorbs on discrete lattice sites, making translation vector prediction a regression problem. In contrast, only one of six angles can occur for rotation, making the prediction a classification problem. While these tasks could be handled jointly in a single network, using two separate specialized networks for each prediction problem avoids the need to balance two loss functions, reducing complexity. Therefore, two specialized networks with identical layer architecture were programmed using the Keras library in Python, differing mainly in their loss functions during training and evaluation.

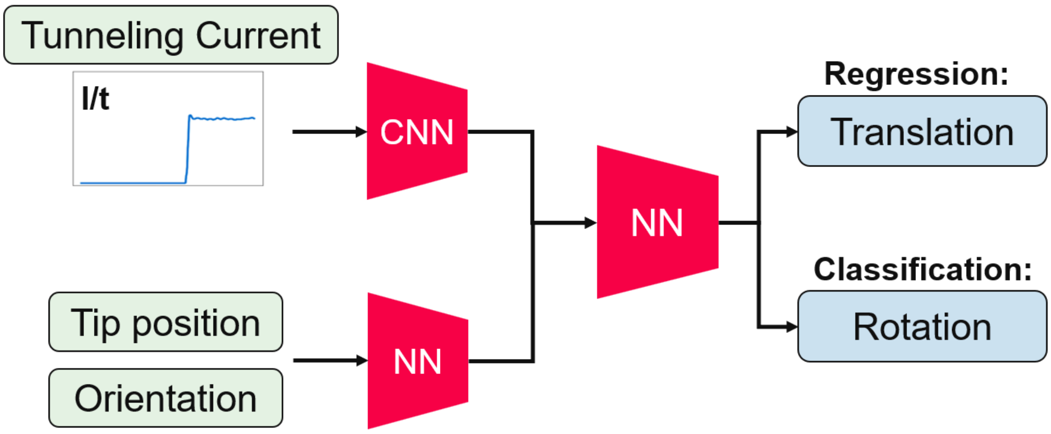


Figure 7: Schematic architecture of network model for predicting the outcome of single-molecule manipulations steps. The input is split in into an image-like part fed into a convolutional network (CNN) and a part fed into a fully-connected neural network (NN). The branches are rejoined before the two outcome variables are predicted, which pose a different fitting problem due to their variable nature.

4 Results and Discussion

4.1 Object Detection with YOLO

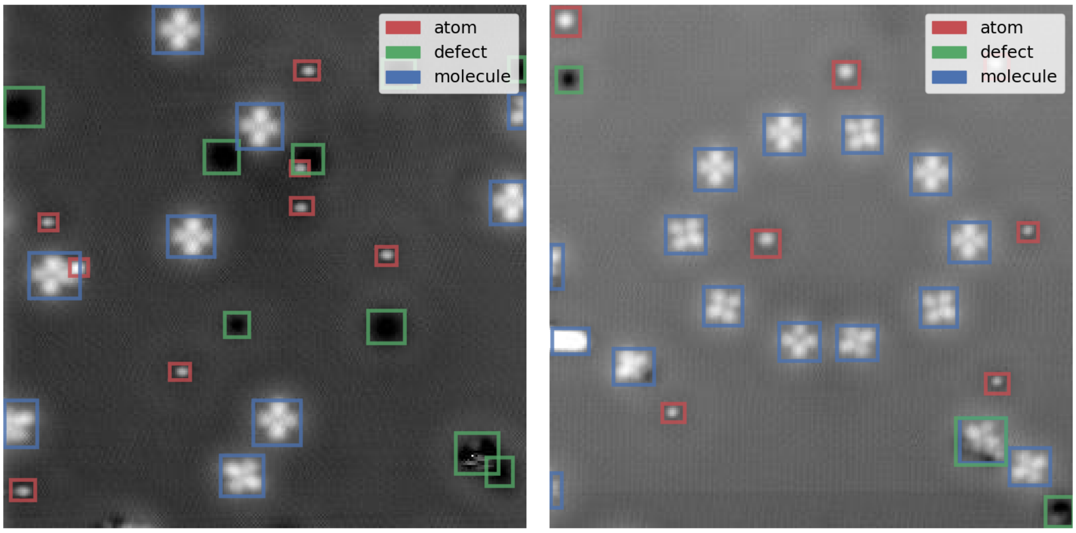
The pipeline described in Sec. 3.1.1 serves as a versatile tool for swiftly experimenting with various systems and parameter configurations. While a vast amount of parameter combinations and settings are possible, systematic optimization was not pursued in this work, as the primary aim was to showcase the algorithms viability. Nonetheless, the models predictions prove to be satisfactory, even with single-shot training, where only a single annotated image is used to generate training data.

Fig. 8a depicts an STM image with bounding box annotations of the three object classes, serving as the single original image for fine-tuning the pretrained model. This image was augmented to a dataset of 1000 elements by applying affine random transformations, including shifts, scalings, rotations, flips and random changes in brightness. After fine-tuning a pretrained YOLOv8 model with this set, it achieves a score of mAP = 0.73 on a test set consisting of 142 different and unknown STM images. Fig. 8b illustrates one of these test images alongside the models predictions, demonstrating correct detection of all objects except for a small defect partly obscured by a phthalocyanine molecule.

Upon qualitative inspection of predictions for the test dataset, three categories of erroneous behaviour can be identified. Firstly, the model can detect objects of similar size as in the training data, but struggles to recognize them, if the size is vastly different e.g. due to different magnification factors of the STM micrograph. Fig. 9 shows two examples of this behaviour, where each image features a single object on the Ag(111) surface. Whereas in one case, the algorithm locates the adatom correctly but classifies it incorrectly as a molecule (the largest object class the model has seen in training), in the other case no object is predicted at all.

Secondly, the model encounters difficulty with highly blurry images, which was observed when testing a system of adsorbed molecules P3N3Clx on Cu(111) with different numbers of bound Cl atoms. It has to be noted though, that these specific images also posed a challenge for human observers, which suggests that this blurriness may exceed the capabilities of Machine Vision altogether.

Thirdly, objects and artifacts, whose class is not present in the training image lead to confusion and wrong classifications. For instance, a few test images contain virtual markers coming from the STM imaging software. While the detection correctly identifies and locates these markers as objects, it mistakes them for adatoms or defects, as this new class is not represented in the fine-tuning dataset.



(a) Original annotated training image **(b)** Object predictions in a test image

Figure 8: The left image shows the original training image with its manually annotated object bounding boxes. After fine-tuning the pretrained YOLOv8 model with the augmented set of this image, the model can reliably detect most objects in unseen images, like in the test image at the right.

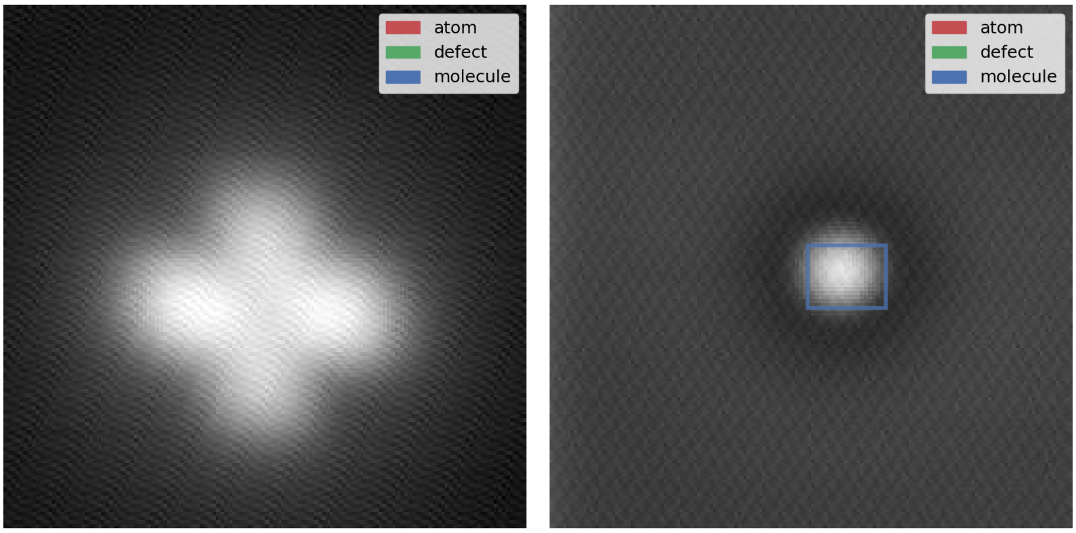


Figure 9: Examples of incorrect detections of YOLOv8 model due to a higher magnification factor. While the phthalocyanine molecule in the left image is not detected at all, the adatom in the right image is incorrectly categorized as a phthalocyanine molecule

We now like to examine these error classes and discuss strategies to address them. For the first class, handling images with different magnification factors presents the most imminent challenge but also offers relatively straightforward solutions. One approach is to include images of every anticipated magnification factor in the training set to prepare the model adequately. However, this approach enlarges the necessary training dataset, potentially hindering adaptability to new surface-adsorbate systems. Alternatively, introducing transformations with high scaling factor during the augmentation process may help to mitigate magnification factor issues. Here only snippets containing entire objects should be selected for magnification, to ensure the inclusion of meaningful training data. Nonetheless, the resulting magnified snippets may suffer from reduced quality due to inherently lower information content than in physically magnified STM images.

The challenge of newly occurring objects in test images, could be addressed by either rigorous inclusion of all classes in the training set, or, if not viable, the step-wise introduction of new classes during evaluation. For this purpose, image locations with a high objectiveness likelihood but small confidence for class categorization might be defined as a new class if they appear repeatedly. This goes far beyond the scope of this thesis warrants further investigation in future research.

Evidently the algorithm proved to deliver satisfactory detection and while further parameter optimization holds promise for improved results, it was not pursued extensively, as the phthalocyanine system served primarily as a sandbox for the overall framework. In scenarios requiring the distinction of a larger number of object classes or where objects exhibit greater similarity, the model's accuracy may diminish compared to the simple scenario demonstrated here. However, tuning the settings should yield significant improvements, as the YOLOv8 framework has the capability to detect and distinguish objects in images at macroscopic level, which exhibit a much higher complexity.

4.2 Motion Analysis

The following chapter summarizes insights from an in-depth analysis of 12,300 manipulation steps of DDNB molecules on an Ag(111) surface, illustrated with graphs. We begin with a statistical analysis to better understand the data, followed by a detailed study of the spatial dependence of molecule movement on the STM tip position during pulsing. Finally, comparing repeated experiments with similar parameters allows us to examine consistency, providing insights into the physical mechanisms and revealing potential impacts of time dependencies, such as changes in tip geometry over time. Subsequently, these insights from the consistency study of the DDNB dataset are compared with the H_2Pc dataset to substantiate the general validity of the results.

4.2.1 Statistics of Motion

Under the assumption that the DDNB molecule remains rigid during motion, the 2-dimensional translation vector and directional rotation angle are sufficient to describe the tip-induced motion. In this subchapter, we focus on the analysis of the DDNB dataset. While the H_2Pc dataset yields similar conclusions, the DDNB dataset offers better statistical significance due to its larger sample size.

Fig. 10 shows a histogram of the effective rotation angle, calculated from the orientation states of subsequent STM images. As previously described, the molecule can adsorb in one of six orientations on the hexagonal surface. Note that this angle does not necessarily represent the true rotation angle, as only initial and final states are compared, meaning that an angle of 300° is equivalent to -60°.

The histogram shows that 0° is the most common rotation angle among the six possible orientation changes, with decreasing occurrences for larger angles and symmetry around 0°. This suggests that each orientation changes can be understood as a series of 60° rotation steps, which all happen with a relatively low probability, as bin counts would be more balanced if multiple full rotations would be common. Thus, it is reasonable to assume that rotation angles do not exceed 180°, allowing the rotation direction to be inferred. The ratio of neighbouring bin counts is approximately 3:1, which hints at the probability of an individual 60° rotation step. However, this ratio is strongly influenced by the uneven sampling procedure and the molecule anisotropy, which reduces the reliability of this estimation.

Next, we examine the distribution of translation vectors. Fig.11 shows a histogram of one Cartesian vector component. The graph is roughly symmetric, ranging from -2 nm to 2 nm, with the highest density around 0 nm. A peak structure is clearly distinguishable at regular intervals adjacent to the central peak, with a mean spacing of 0.265 nm. This spacing closely matches the vertical distance between two Ag(111) lattice sites (0.250 nm), confirming that DDNB adsorbs specifically at lattice sites.

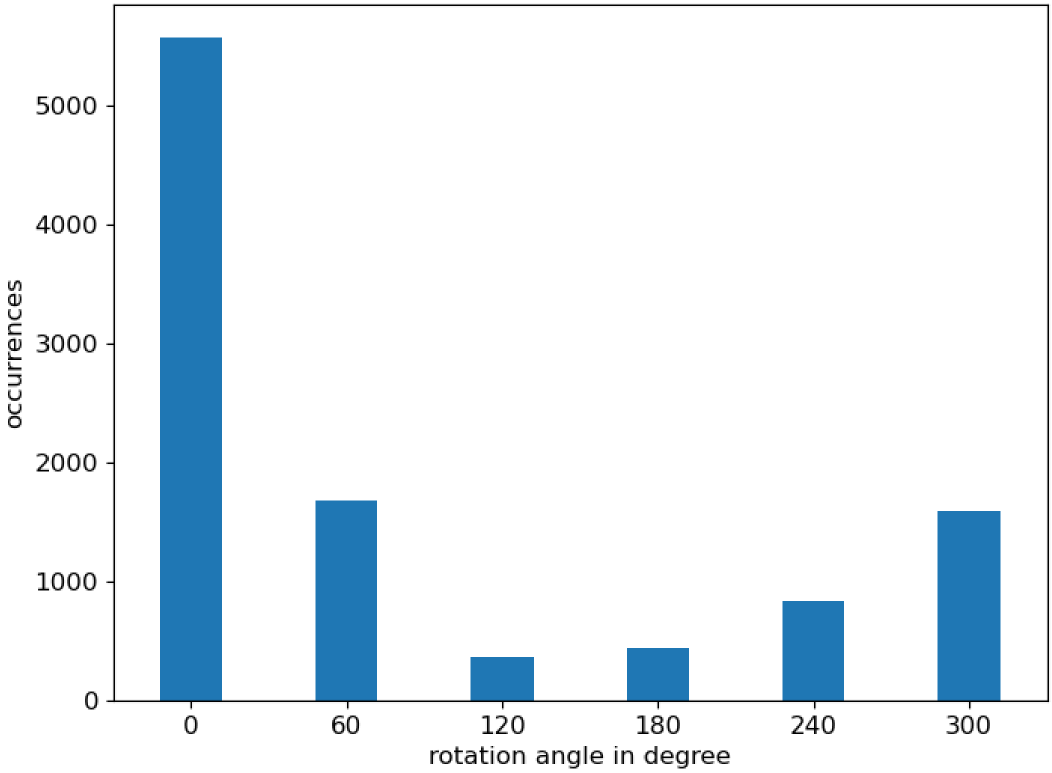


Figure 10: Histogram of the six rotation angles in the data set, calculated from the molecule orientation in subsequent images.

The peaks have a finite width of approximately 0.2 nm, attributed to image resolution limits. This is evident when a similar histogram is plotted for data points with an original image resolution of 64x64 pixels (instead of 32x32), where the peak width halves to 0.1 nm.

Similar peak structures can be seen in histograms of the translation in x-direction and the absolute translation distance, with peaks corresponding to each lattice site hop. Notably, 45% of all data points have an absolute translation of less than half a lattice constant (0.145 nm), while 16% have translations of more than three lattice constants. This indicates that manipulation steps with no translation are common, whereas those with large translations are rare.

A 2-dimensional visualization of all translation outcomes is shown in Fig. 12. It depicts a DDNB molecule on the hexagonal Ag(111) surface, with a lattice constant of 289 nm. The molecule's contour is marked by a black dashed line, and a black cross at its pivot point, which serves as the centre for translation and rotation. Each manipulation step's orientation-corrected translation destination is plotted as blue dots with low alpha values. The distribution of these dots is inhomogeneous, due to the non-uniform sampling of tip positions and the irregular translation specificity for each site, both explained in Sec. 4.2.2. Multiple destination locations can be seen to aggregate into dark blue groups at lattice points. A significant fraction of dots is located at the pivot point (=coordinate

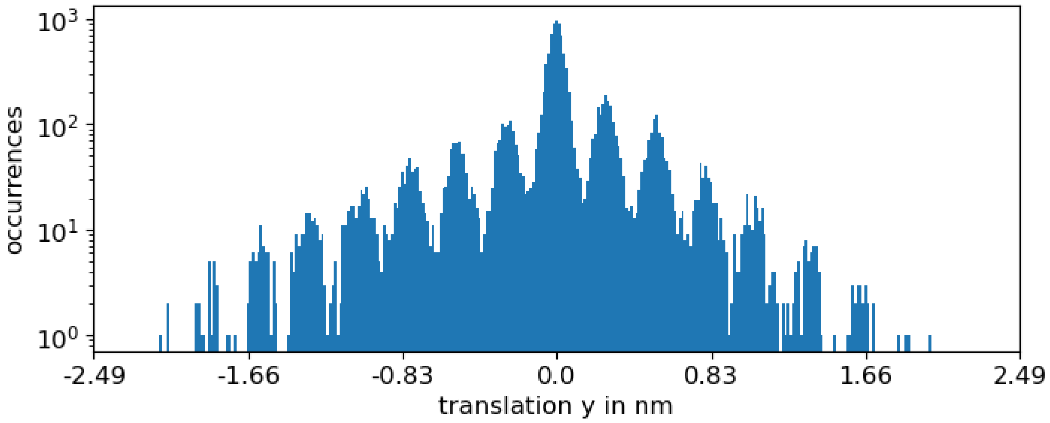


Figure 11: Logarithmic histogram of the y-components of all translation vector. Peaks with a regular spacing of 265 nm are visible.

origin), indicating manipulations with no translation. The surface lattice structure only arises for datasets with thousands of manipulation steps, which highlights the importance of rapid automatic dataset extraction and analysis for single-molecule manipulation.

The agreement with the expected discrete adsorption behaviour is only visible, when selecting the right pivot point during the data extraction algorithm (or by subsequent correction). If it is off by a small margin, virtual translations occur when the molecule is rotated, which leads broadens the translation peaks in Fig. 11 and Fig. 12. Thus, this graph helps to refine a roughly determined pivot point. When the correct position is found, this graph provides compelling evidence for the specificity of adsorption sites and helps to determine the surface lattices orientation, found to be rotated 2° clockwise relative to the Cartesian coordinate system. While manipulation destinations group accurately around nearby lattice sites, they coincide less precisely for sites farther from the pivot point. Increasing the surface lattice constant by 10% improves this alignment, suggesting experimental distortions in the data set. The agreement with the expected discrete adsorption behaviour validates the automatic data extraction process and is only apparent for large datasets.

The DDNB dataset reveals a wide range of translation vectors. As explored further in Sec. 4.2.2, these vectors strongly depend on the STM tip position during the voltage pulse, since the tip induces an attractive potential, acting like a target for translation. Consequently, molecule translation should be roughly proportional to the distance between the tip and the molecule's anchor point, defined as the pivot point. This expectation is evaluated in Fig. 13, a 2-dimensional logarithmic histogram of the ratio of molecule translation to tip distance for all data points, both corrected for the initial molecule orientation. Ratios above 2 are excluded for clarity and a red dashed square at ratios ±1 for the x- and y-components helps assess the results. If the molecule would translate exactly the distance between the initial pivot point and the tip position, it would appear at the (1,1) coordinate.

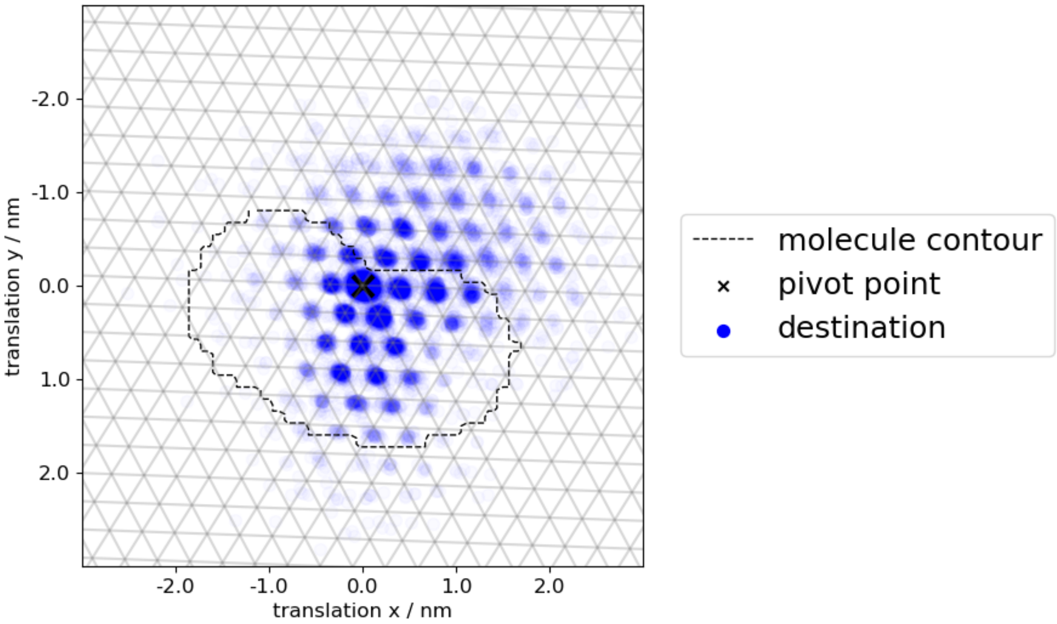


Figure 12: DDNB molecule, outlined by the dashed contour with a black cross at its pivot point, adsorbed on a hexagonal Ag(111) lattice rotated by 2°. For all 12,300 manipulations the orientation-corrected destinations are shown as light blue circles, aggregating at lattice sites. The inhomogeneous distribution arises among other things from non-uniform sampling of tip positions.

A strong central peak is visible at (0,0), representing steps where the molecule does not move, and is elongated along the horizontal and vertical axes, indicating single-direction translations. Notably, there is no peak at (1,1), where directly proportional translation would appear. Instead, data points are spread across the plot, with many showing translation beyond or opposite to the tip direction. The upper right quadrant contains the most points, followed by the upper left and lower right quadrants, which both indicate molecule movement towards the tip in one direction and repulsion in the other. This behaviour does not surprise since the sampled tip positions do not necessarily align with the lattice sites, and the molecule aims to minimize its distance to the tip while being constrained to a discrete grid of adsorption sites.

It's important to note that the displayed ratio does not account for absolute deviation from the target position. When the tip is placed very close to the pivot point, even a small overshoot can result in a disproportionate large ratio. However, in 80% of the data points plotted in Fig. 13, the tip was placed at least three surface lattice constants away from the pivot point. Plotting a similar histogram, which contains only these distant target positions, reduces the number of stray points with a negative translation/tip distance ratio, but about 8% of points still show a negative ratio. This indicates that single-molecule manipulation with STM tip voltage pulses is a stochastic process, where

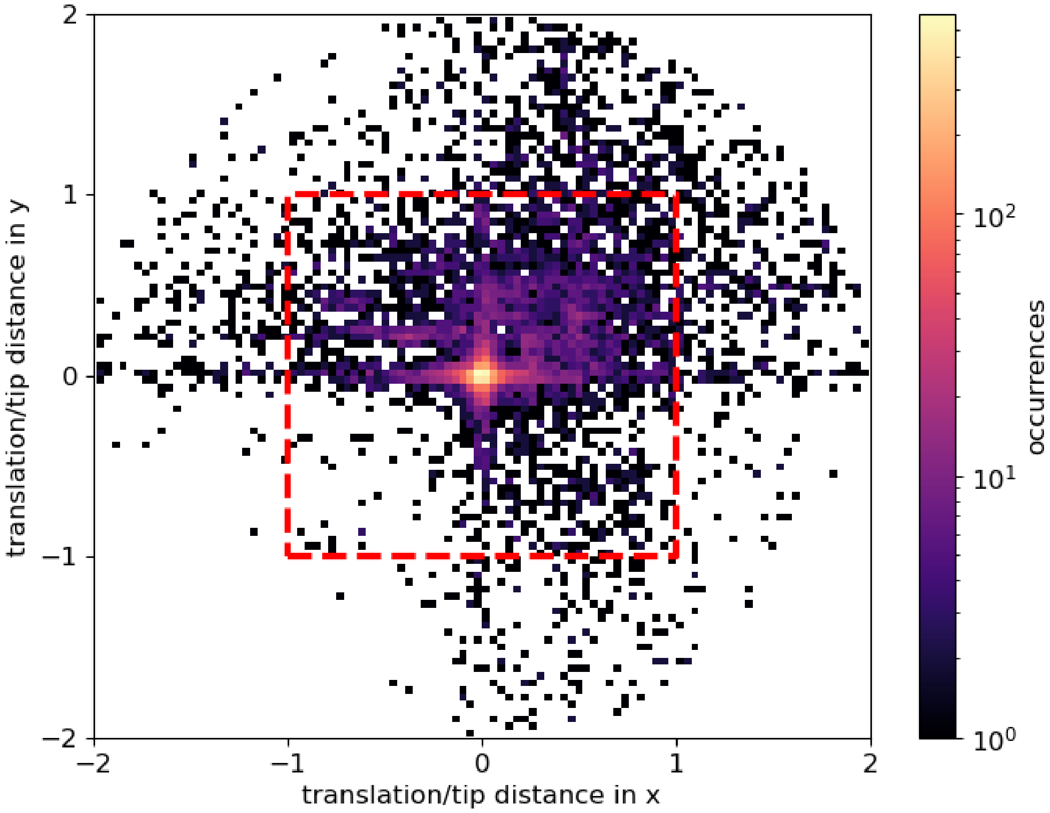


Figure 13: 2-dimensional logarithmic histogram of orientation-corrected molecule translation, divided by STM tip distance in both Cartesian coordinates. The red dashed square indicates the ratio of -1 and 1, combined ratios below -2 or above 2 are omitted.

repeated experiments with constant parameters yield a wide range of distinct outcomes. This hypothesis is readdressed in Sec. 4.2.3.

This chapter can be summarized with three key observations:

1. A significant portion of data points show no rotation, no translation, or neither.
2. The molecule translates to discrete lattice sites, although the translation vectors appear continuous due to measurement uncertainties from low image resolution.
3. Single-molecule manipulation shows weak proportionality to the tip position, which defines the target position. Despite this, the DDNB molecule generally tends to move towards the STM tip.

In the next chapter the translation dependency on the tip position is further explored to gain insights into the underlying physics.

4.2.2 Spatial Dependence Study

The initial goal in creating the DDNB dataset was to optimize manipulation parameters, including the STM tip position, for maximal translation towards a predefined goal location. As a result, a large 2-dimensional array of tip positions was sampled non-uniformly, now allowing for analysis of movement behaviour based on tip position. For this analysis, we grouped the 12,300 manipulation steps into evenly-spaced bins of similar tip positions. Note that these bins do not fully align with the original sampling grid due to a different molecule localization algorithm used in the original study.

Fig. 14 visualizes the number and positions of bin elements. It shows an STM image of the DDNB molecule with bin counts plotted where the tip position for each bin is located relative to the molecule. The hexagonal Ag(111) surface lattice is indicated with light grey gridlines, and the molecule's pivot point is marked by a small blue cross. The sampled grid forms a square, rotated approximately 30° relative to the image frame, with bin counts ranging from 1 to 700. Since the reinforcement algorithm sampled positions with large translations more frequently, the bin count distribution already indicates regions with reliable translation. All images in this subchapter use this data binning to compare manipulation outcomes for different tip positions.

We begin by examining the consistency of inducing single-molecule translation. Given the smeared translation peak, we use a threshold of half a surface lattice constant (0.145 nm) to distinguish between manipulation steps where the molecule moves and those where it remains stationary. Fig. 15 shows pie charts plotted on the STM image, depicting the percentage of experiments where the observed absolute translation exceeds this threshold. Translation success varies significantly across the molecule and lacks the mirror symmetry along the short axis that the molecule has in the gas phase. Two regimes of reliable translation are evident: one northeast and one south of the pivot point at the molecule's edges, while the translation success directly above the molecule is moderate.

A high success rate is only one ingredient for fast translation; a more expressive metric is the mean translation vector, shown in Fig. 16. Arrows starting at the respective tip positions depict the mean translation vector of the molecule to scale, with colour shade highlighting the magnitude. The translation-sensitive positions for DDNB are distributed asymmetrically, with three main regions identifiable.

The first two regions are located east and south of the pivot point, where a large bin count in Fig. 15 already indicated efficient movement. In both regions, which are roughly enclosed by a 1.5 nm by 0.8 nm rectangle, the arrows point away from the molecule centre in a divergent, explosion-like pattern. The maximum mean translation distance here is about three lattice constants, or 1 nm. The third region with noticeable average translation is north and northwest of the pivot point, where the arrows are smaller but follow a different pattern. Instead of diverging from the molecule centre, they follow the molecule's outline, forming a partial vortex. For most other positions, the mean translation is near zero, including positions inside the molecule. These positions showed a moderate translation success rate in Fig. 15, which now indicates that the translation direction at these positions is random and averages to zero over many experiments.

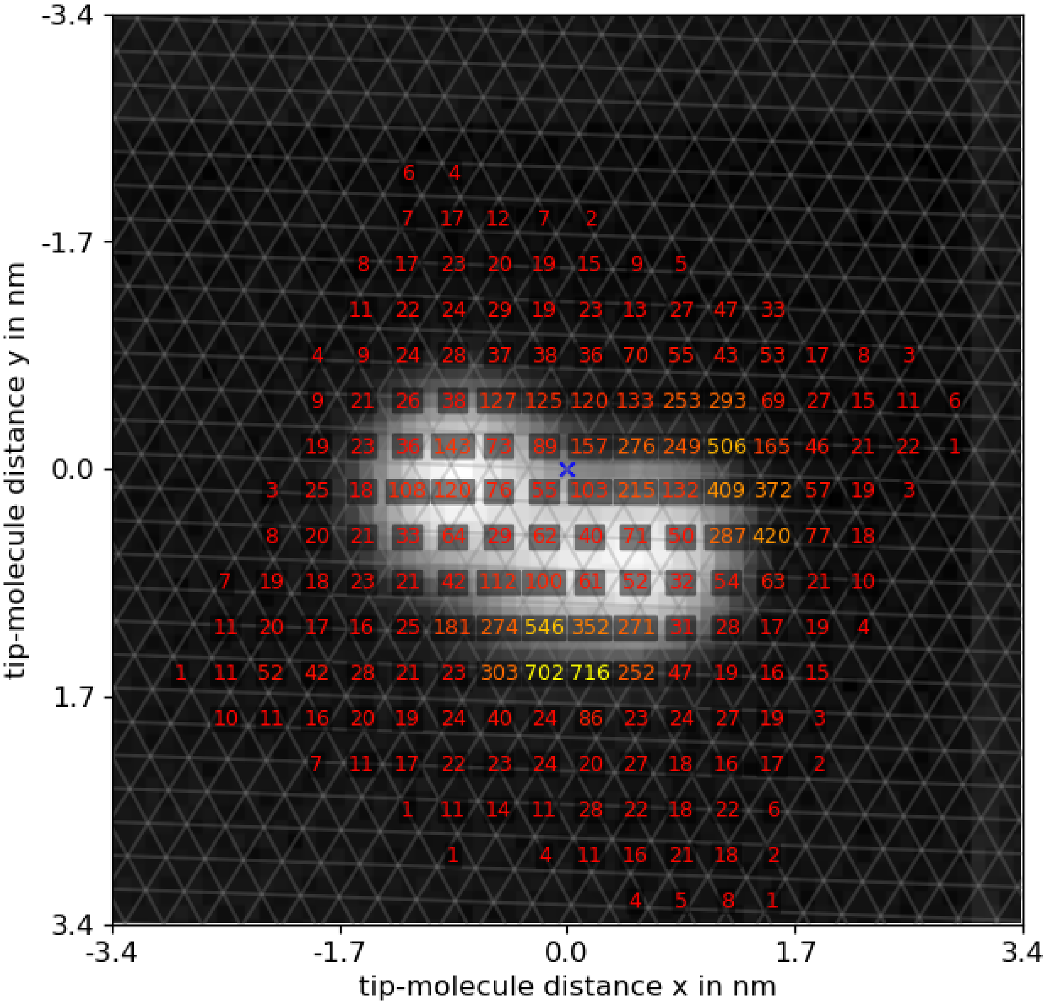


Figure 14: Number of experiments performed for each STM tip positions, binned in a 20x20 grid, with the colour coding the bin count and a blue cross marking the pivot point.

Next we explore the influence of tip position on molecule rotation. Similar to the previous plots, Fig. 17 depicts a DDNB molecule on the hexagonal Ag(111) grid, with the molecule's pivot point marked by a blue cross. At each tip position a pie chart shows the ratio of molecule rotations between initial and final orientations for all experiments of this bin. Colours representing the six possible angles highlight the orientation direction, with blue shades indicating clockwise orientation change and red shades indicating counterclockwise orientation change. Additionally, a yellow arrow indicates the axis and direction of the permanent dipole of the DDNB molecule.

As discussed in [13], the orientation behaviour is analogous to a dipole in an electric field. The plot confirms that the internal molecule dipole orients itself towards the STM tip, which acts as the negative pole during pulsing. Clockwise and counterclockwise rotation regimes are separated by the dipole axis, with predominant 180° flips at the

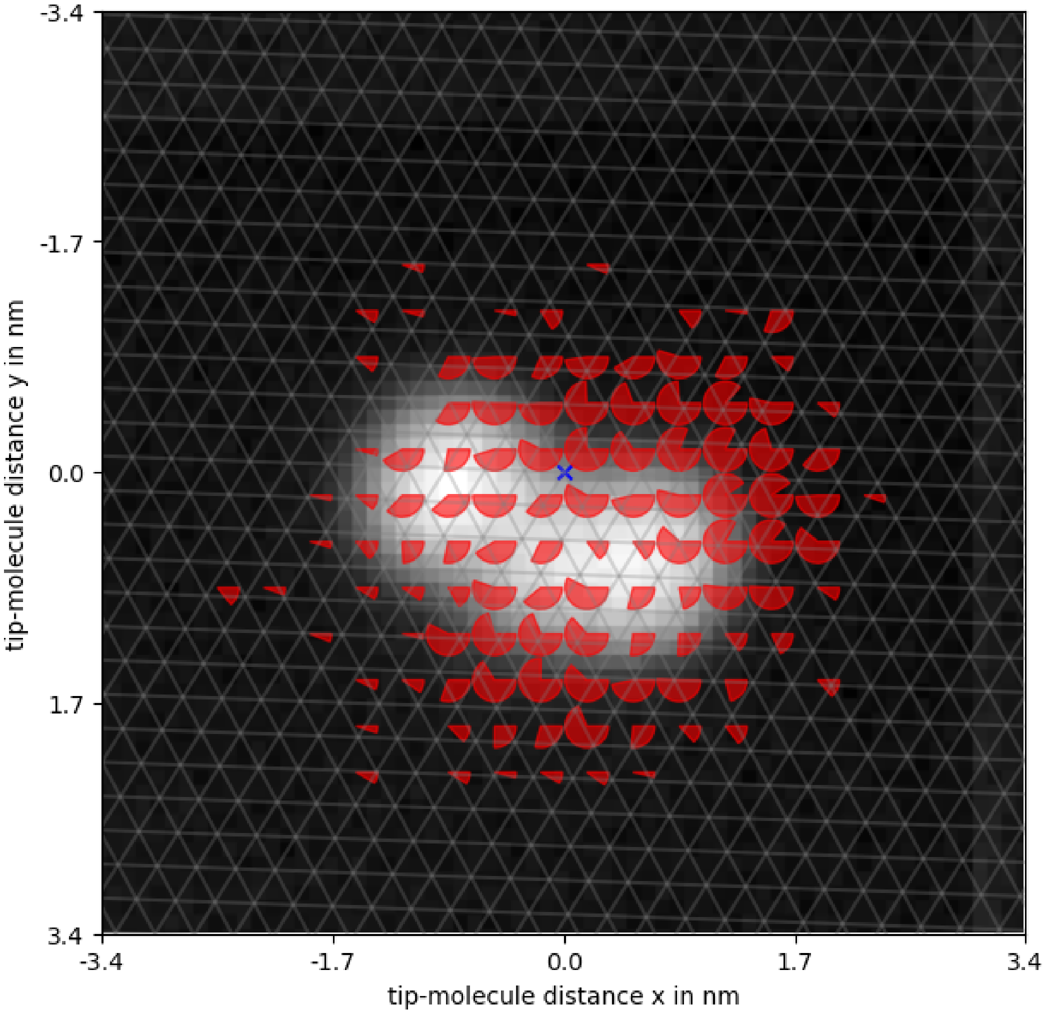


Figure 15: Pie charts depicting the share of successful translations (exceeding (0.145 nm) for different tip positions relative to the molecule. A blue cross marks the pivot point.

dipole's tail and no rotation at its head. In most pie charts, the 0° slice is the largest, followed by single 60° rotation steps in both directions. This further indicates that single-molecule manipulation is not a fully deterministic electrostatic process but rather a series of probability-driven events.

A similar analysis of the much smaller H_2Pc dataset revealed predominant translation at the molecule edges and less systematic rotation. It is likely that the nonpolar molecule becomes polarized during voltage pulsing, influencing the manipulation outcome. Like the DDNB molecule, all induced translation vectors diverge from the molecule centre and exclusively pull, rather than push.

In conclusion, reliable translation of the polar DDNB molecule is achievable only when the STM tip is positioned at specific locations around the molecule. These positions are

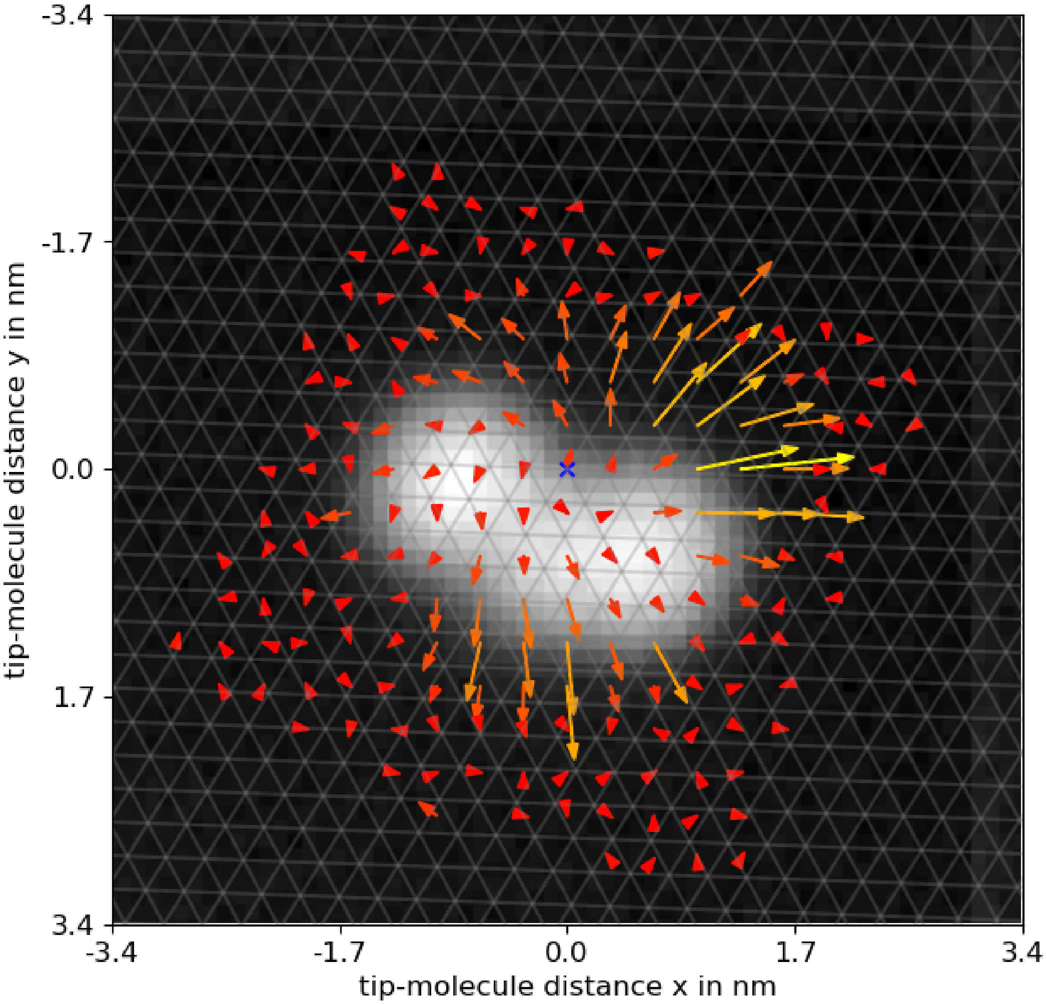


Figure 16: Mean translation vector for different tip positions relative to the molecule. Arrows indicate direction and magnitude of translation, starting from the respective tip position and scaled accordingly. A blue cross marks the pivot point.

unique to a molecule and challenging to predict. The molecule's permanent dipole shapes the movement behaviour but cannot fully explain it. Although the rotation angles are comparable to a dipole in an electric field, different and even opposed rotations occur. Moreover, the translation vector map cannot be explained by the simple electrostatic model, lacking symmetry along the internal dipole axis or the molecule's mirror symmetry axis. Instead, the manipulation process might involve a complex interplay of electrostatics, electron shielding, molecule polarizability, and quantum-mechanical stochastics. In the following chapter, qualitatively analyse movement consistency across repeated experiments is discussed to address this stochasticity.

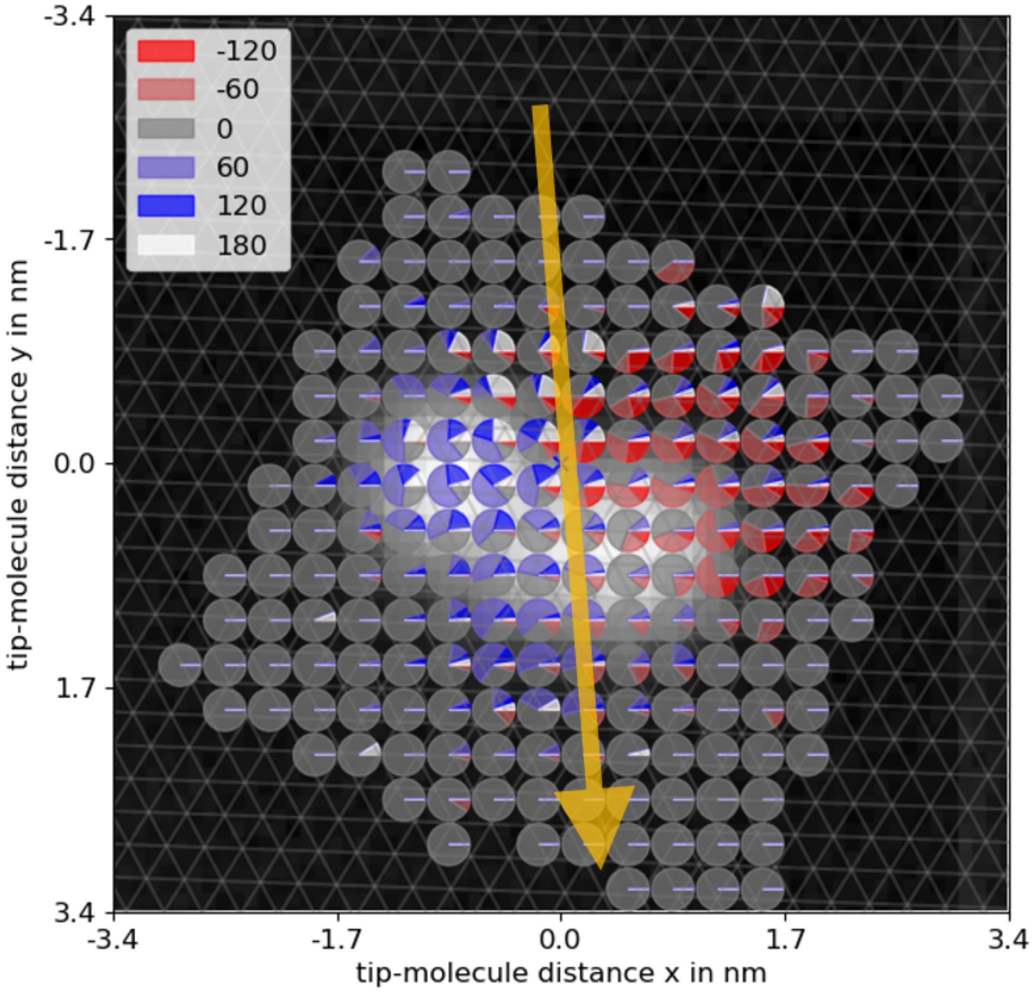


Figure 17: Pie charts depicting the share of induced rotation angles for different tip positions relative to the molecule. Blue shades represent clockwise rotation and red shades counterclockwise. A blue cross marks the pivot point and a yellow arrow the molecule dipole, which roughly splits the blue and red regime.

4.2.3 Consistency Study

In the previous chapter Sec. 4.2, we grouped manipulations with similar tip positions into bins and analysed their averages to study their influence. In this chapter, however, we focus on a single bin and compare all experimental outcomes within it to gain an understanding of the consistency and variance of single-molecule manipulation under repeated experimental conditions.

Fig. 18 illustrates the translation outcomes of DDNB manipulations with an exemplary tip position. The black dashed line outlines the DDNB molecule, with a black cross at the molecule's pivot point and a pink cross marking the designated tip position, situated 1.04 nm right and 0.22 nm below the pivot point. We collect 126 experiments with the

STM tip positioned within a tolerance radius of 0.1 nm around this marked position. For each experiment, the translated pivot point is represented by coloured dots, with the colour indicating the elapsed time for each step since the start of the measurement campaign. Due to measurement uncertainty arising from image resolution, the dots are not precisely situated at the surface lattice sites.

Evidently, the dots are scattered over a large area, spanning several lattice constants, with some points falling short of the goal position, some surpassing it, and many deviating from the goal direction sideways. Interestingly, the scattering is anisotropic, as almost all points lie on one side of the connecting line and outside the molecule. This might be due to distortions of the electric field formed during pulsing, caused by the molecule orbitals, resulting in a less attractive potential inside the molecule for the pivot point. In turn, the tip shape is unlikely to be responsible for this anisotropy or the large spread of translation vectors, as indicated by the unsystematic distribution of elapsed experiment time. Despite the experiment extending over two weeks and several tip changes occurring, no clear pattern between translation outcome and elapsed time is discernible. Thus, the tip shape appears to have little significant impact on the manipulation, which is an important observation, as otherwise, the comparison of data points with significant differences in elapsed time would be problematic.

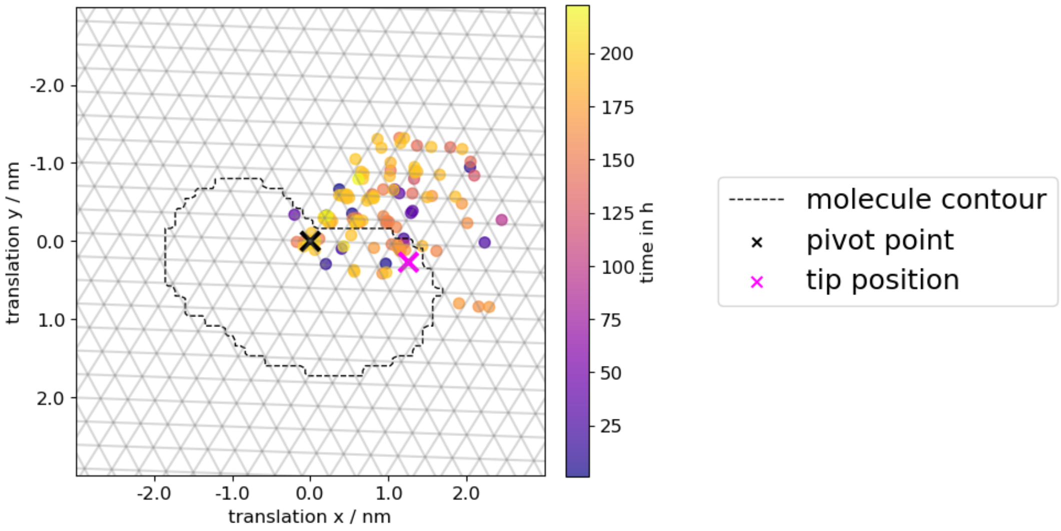


Figure 18: Comparison of 126 repeated experiments with the tip placed 1.06 nm from the DDNB pivot point, marked by a black cross. A black dashed line represents the molecule outline, and a light grey grid the hexagonal Ag(111) lattice. Coloured dots depict the pivot point location after each manipulation step, with the colour indicating the elapsed time since the start of the measurement campaign.

In Fig. 19, we analyse two different repeated experiments to quantify the spread of translation vectors. We exclude actions where the absolute translation distance is smaller than half a lattice distance and fit a 2-dimensional Gaussian probability density function to the remaining data points. The blue shades depict this fit, with each shade representing probability density values spaced by a 0.5 sigma interval. The standard deviation of the Gaussian density function is calculated as 0.468 nm in the x-direction and 0.385 nm in the y-direction for the first experiment. In the second experiment, with the tip located 2.366 nm south of the pivot point, the translation vectors have a fitted standard deviation of 0.269 nm in the x-direction and 0.534 nm in the y-direction. Similar standard deviations and translation vector spreads are observed across different repeated experiments.

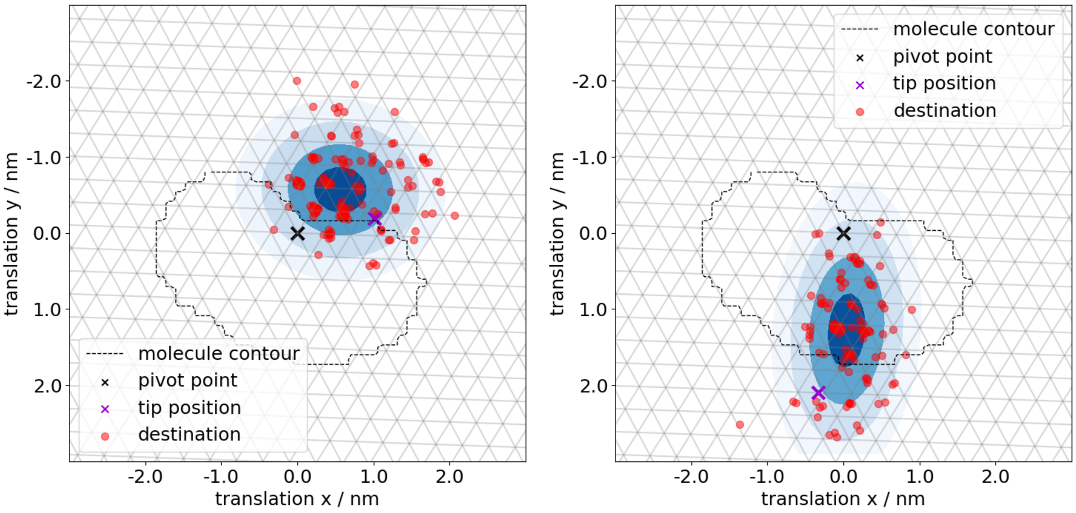


Figure 19: Comparison of repeated experiments with two tip positions, marked by a violet cross and the DDNB pivot point marked by a black cross. The black dashed line represents the molecule outline, the light grey grid the hexagonal Ag(111) lattice. Red dots depict the pivot point location after each manipulation step, neglecting steps, where no translation occurs. Fitted 2D Gaussian probability density functions are plotted in blue shades.

In Fig. 18, we examined the relationship between the translation vector and the elapsed measurement time for a set of repeated manipulation experiments with a constant tip position. Now, in Fig. 20, we replace the time parameter with the final orientation state by plotting arrows instead of dots at each coordinate where the pivot in the 126 experiments is translated to. These arrows point in the direction of the molecular dipole moment, and their colour additionally represent the angle to enhance visibility.

Fig. 17 from the previous chapter suggests that the molecule dipole orients itself towards the STM tip, which acts as the negative pole of the electric field during pulsing. This expectation can indeed be partly confirmed in Fig. 20, where the coloured arrows roughly point towards the tip position, marked by a pink cross. However, in a considerable number of experiments, the molecule is not in the orientation state we would expect from a purely electrostatic process. This cannot be attributed to too little energy in the system, as earlier work showed that the process of translation starts at higher energies than the process of rotation ([13], [14]). The fact that the molecular orientation is not always in the same state as a classical dipole, despite enough energy to overcome rotation barriers, suggests that single-molecule manipulation does indeed involve probabilistic processes. Note that unanticipated orientation cannot be explained by premature termination of a manipulation step, as tunnelling current curves of, for example, the far right light green arrow are static in a final time interval, rejecting the notion that pulsing time might be the limiting factor.

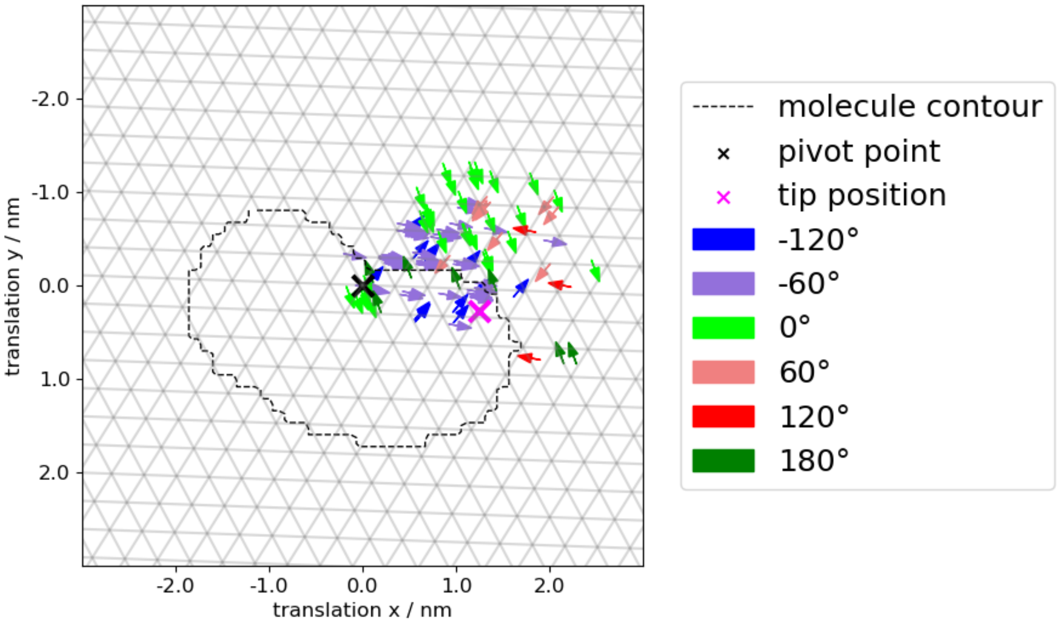


Figure 20: Comparison of 126 repeated experiments, where the tip, marked by a pink cross, is placed 1.06 nm from the DDNB pivot point, marked by a black cross. A black dashed line represents the molecule outline and a light grey grid the hexagonal Ag(111) lattice. Arrows depict the pivot point location after each manipulation step pointing into the direction of the molecules dipole moment with the colour indicating the orientation again for better visibility.

Similar analyses conducted on the H_2Pc dataset yield comparable insights into movement consistency. In Fig. 21, outcomes for repeated manipulation steps with constant tip positions are presented. The black cross denotes the molecule centre, serving as the rotation centre for H_2Pc due to the absence of a dipolar binding mechanism, while the pink cross indicates the tip position, located 0.765 nm from the centre. 50 experiments are collected with the tip positioned within a 0.08 nm tolerance radius around this marked position. Although H_2Pc adsorbs in six orientation states, only three are distinguishable due to its mirror symmetry. Hence, coloured lines, rather than arrows, are plotted to depict the orientation at the translated molecule position after the manipulation step, running along the axis of the pair of smaller lobes.

The molecule destinations after translation are scattered around the pivot point, indicating that this behaviour also occurs for non-polar molecules. For the tip position shown, these destinations are almost homogeneously scattered around the pink cross within a radius of about three lattice constants. Despite lacking a permanent dipole, the molecule orients its minor axis towards the tip position, with even higher reliability than the DDNB molecule, which does have a permanent dipole. This suggests that the applied electric field induces a dipole with a pronounced preferential direction in the molecule, causing this alignment.

Examinations of whether molecule translation depends on the elapsed experiment time also did not reveal any obvious patterns for H_2Pc. Thus, it is reasonable to assume that, as with DDNB, the tip shape does not strongly influence the outcome of single-molecule manipulations for H_2Pc, ensuring that manipulations with different tip shapes remain comparable.

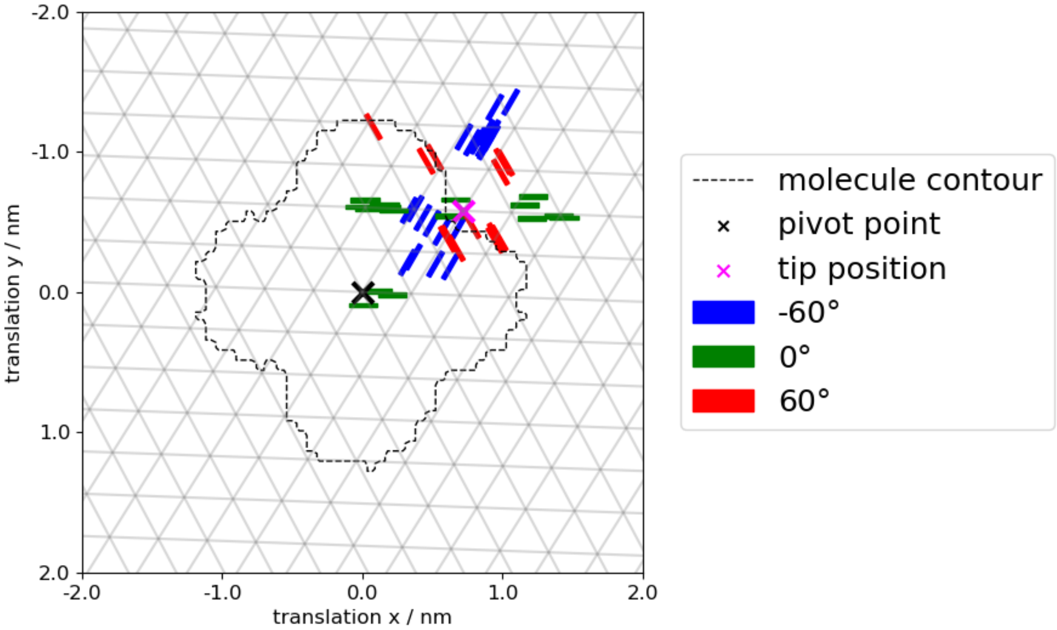


Figure 21: Comparison of 50 repeated experiments, where the tip, marked by a pink cross, is placed 1.06 nm from the DDNB pivot point, marked by a black cross. A black dashed line represents the molecule outline and a light grey grid the hexagonal Ag(111) lattice. Arrows depict the pivot point location after each manipulation step pointing into the direction of the molecules dipole moment with the colour indicating the orientation again for better visibility.

In this chapter, we assessed the consistency of manipulation experiments for both the DDNB and H_2Pc system. Analysis of repeated experiments with arbitrarily chosen STM tip positions yielded comparable results for both molecules when examining sufficiently large sets. Two primary observations emerged. Firstly, translation varied significantly across repeated experiments, indicating a strong stochastic characteristic or alternatively, dependence on additional currently inaccessible parameters. Secondly, induced molecular rotation displayed a high dependence on the associated translation vector, often orienting towards the STM tip akin to a dipole in a classical electrostatic field, even though a considerable number of exceptions occurred. This again suggests a combination of electrostatic and probabilistic effects governing the underlying physics. The consistency of these findings for both molecules, despite differing dipolar properties, underpins their generalisability.

In conclusion, controlling parameters such as initial molecule orientation, pulsing time, voltage, and STM tip position relative to the molecule is insufficient for ensuring consistent molecular movements across repeated experiments. Consequently, predicting molecule movement solely based on these parameters is not feasible. In the following chapter, we explore improving the prediction performance by incorporating additional parameters, particularly the tunnelling current curve recorded during voltage pulsing in single-molecule manipulation.

4.3 Tunnelling Current Analysis for DDNB system

The tunnelling current based movement prediction was performed on the exemplary dataset of manipulations of DDNB was performed with two mixed-input neural networks, as introduced in Sec. 3.3.2. They exhibit the same layer architecture, but differ in their loss function, tracked metrics and also in the activation function of the final layer. Thus, one of them is suited for fitting continuous translation vectors in a regression type problem and the other for classifying discrete rotation angles. Tab. 1 compares their differences. Both optimized models have approximately 19,300 free parameters, which is significantly lower than the 50,000 free parameters a comparable fully connected network would require.

Table 1: Network parameters for the classification and regression task

Parameter	Classification Network	Regression Network
Final layer activation	softmax	linear
Loss function	categorical crossentropy	mean squared error
Tracked metric	accuracy	mean average error

The dataset was split into a training set used to fit the network, a validation set used for tracking the training progress and a test set for final evaluation, with the proportions 64:16:20. Both models were trained using the optimizer ‘Adam’ with a learning rate of 0.001 during a maximum of 100 epochs, but were stopped early if the respective metric for the validation split did not improve for 10 epochs.

4.3.1 Translation vector prediction

The performance of the translation prediction networks was evaluated using the mean average error (MAE) between true value and prediction per Cartesian component of the translation vector in the test split of the dataset. We benchmark our models performance against the Null Hypothesis, which states, that no molecule movement takes place at all.

The error for this hypothesis is equal to the mean absolute translation distance and has per component an average magnitude of 0.414 nm

The highest achievable mean absolute error (MAE) across the entire test split is 0.226 nm, surpassing the average image resolution limit of 0.166 nm but falling significantly below the lattice constant of the hexagonal Ag(111) surface, which is 0.289 nm. This relatively low MAE value is possible due to a substantial portion of manipulations without translation, making them easier to predict and consequently lowering the MAE. However, when focusing on actions with an absolute translation of at least half a lattice constant (0.145 nm), the best achievable MAE rises significantly to 0.362 nm.

Fig. 22 illustrates the comparison between the true translation values for a single Cartesian coordinate and the network's predictions. Translations range from −1.5 nm to 1.5 nm, with approximately half of the data points exhibiting a translation of around 0. The predictions also span the entire range. Although deviations tend to be larger for true values further from zero, the predictions generally follow the curve's trend and exhibit a symmetric distribution.

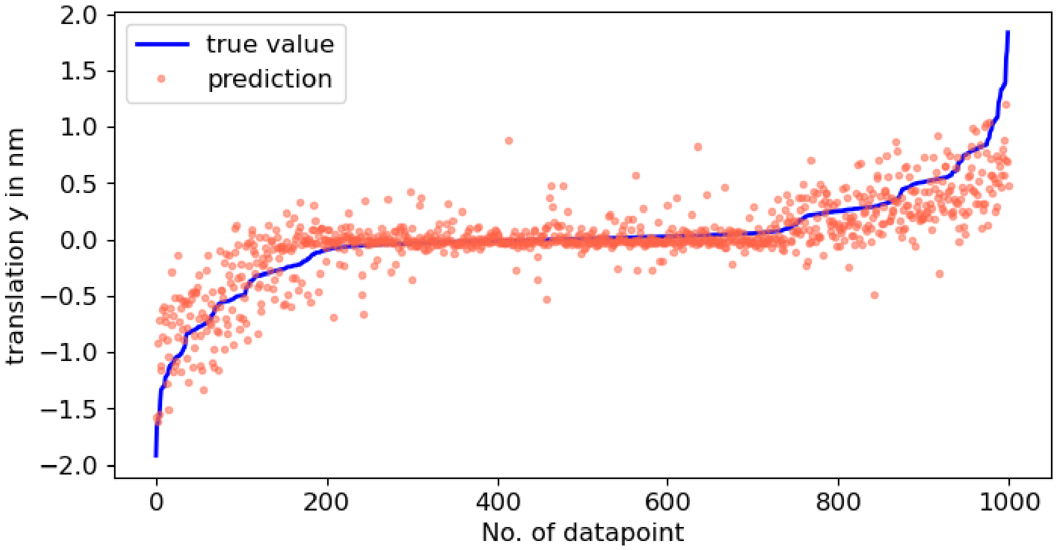


Figure 22: The blue line depicts the translation of the molecule in a single Cartesian direction for 1000 test data points, ordered by magnitude. Red dots represent the network's predictions, showing deviations from the true values across all ranges, with smaller errors observed around the true values near 0

The minimal set of input parameters, which led to the best performance is constituted of the tunnelling current time series, the relative tip position measured at the beginning of the manipulation and the initial orientation. Tab. 2 breaks down the contribution of these parameters by comparing the prediction performance for five training and evaluation runs.

Table 2: Best mean average error (MAE) on the test set in five training runs with different input parameter combinations, compared to the Null Hypothesis of no translation occurring. The average error across these runs follows a descending trend.

	Tip position	Tunnelling current	Orientation	MAE in nm
Null Hypothesis	-	-	-	0.414
Model 1	Included	-	-	0.342
Model 2	Included	Included	-	0.256
Model 3	Included	Included	Included	0.226

The first line in Tab. 2 shows the Null Hypothesis, which assumes no translation in all data points. Since many steps are without translation and translation occurs in all directions roughly equal, this hypothesis is not as unfavourable as it seems initially. The associated MAE reflects the average translation per Cartesian coordinate.

The tip position and tunnelling current time series contribute differently to the mean average error. While the tip position provides directional information in 2D, the tunnelling current encodes movement magnitude in 1D. This distinction is evident when comparing predictions and true values in a plot akin to Fig. 22. Using only the tunnelling current and neglecting the tip position yields accurate predictions for zero values but completely symmetric errors around zero for larger true translations.

Surprisingly, incorporating the initial orientation slightly improves prediction performance. Attempting to eliminate it by transforming all orientation dependent quantities with a rotation matrix results in slightly worse predictions. This suggests spatial anisotropy in the STM system, possibly arising from non-rotationally symmetric tip shapes or an external directional bias.

Examining Fig. 23 allows us to better understand the distribution of prediction errors. This plot illustrates prediction values plotted against true translation distance in the y direction. While the error spread may appear constant across the range of true values, a notable concentration of points is observed near the zerotranslation line, resulting in a smaller standard deviation. Specifically, for y-translations exceeding and falling below ±0.5 nm, the standard deviation is 0.338 nm, whereas for translations within the range of ±0.04 nm, the standard deviation reduces to 0.08 nm.

As previously documented in [14] and discussed in Sec. 4.2, the translation of DDNB on Ag(111) occurs discretely between lattice sites, contrary to the assumption of continuous values, made thus far in this chapter. Hence, to align with this discrete behaviour, the extracted translation vectors, which contain measurement uncertainties, are discretized by assigning them to the nearest lattice point of the hexagonal Ag(111) lattice, rotated -2° relative to the STM coordinate system. When the model's predictions are confined to lattice sites, comparison with the discretized translation values yields an improved MAE of 0.205 nm, slightly surpassing the performance of continuous prediction. This enhancement is primarily attributed large share of near-zero translations, as most errors are nullified upon discretizing true and predicted values. This statement is proven by applying the discretized model to manipulation steps with an absolute translation of

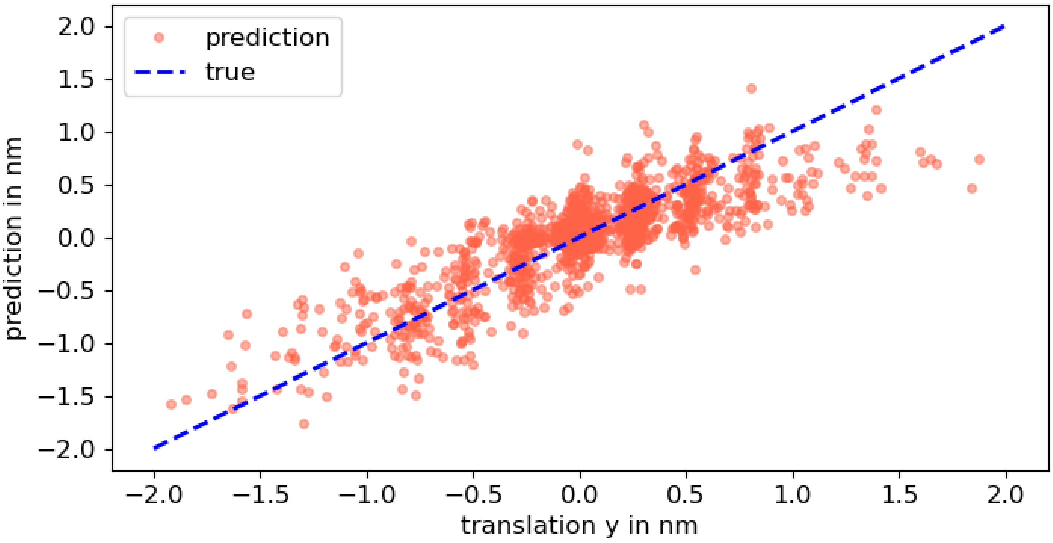


Figure 23: Red dots represent the predicted translation on the y-axis compared to their true values, which are additionally visualized by a blue dashed line.

least half a lattice constant (0.145 nm). In this scenario, performances of 0.354 nm are reached, fairly equal to the continuous model.

The minimal observed error for actions with zero translation raises the questions, whether actions with and without translations can be distinguished reliable. Using a threshold of half a surface lattice constant (0.145 nm) for distinction, actions with an absolute translation above are designated as 1, and those below as 0. A fully trained classification network akin to the one employed for rotation prediction reaches an accuracy of 90% for distinguishing movement from no movement, with an even error distribution of false negatives and false positives.

As described above, the minimal set of best performing input parameters consisted of the tip position, the tunnelling current time series and the initial orientation. Hereinafter additional input parameters are listed, which did not lead to improvements of the MAE when predicting the translation vector components.

- **Total voltage pulse time:** While the pulse time potentially influences the outcome by regulating the energy input, its addition to the model does not augment prediction performance. It's worth noting that the reliability of the extracted pulse time may be compromised due to the absence of clear documentation for the stored STM files.

- **Initial tip approach:** In the original experiment, the STM tip approached the designated position in constant current mode before vertically descending by 1 Å above the surface prior to applying a voltage pulse. As the model lacks information regarding the molecule shape, the binary data, whether the tip is above or not, is none of the direct input parameters. Although explicit inclusion of this information led to more consistent errors across different initialization runs, it did not reduce them further. Rather, it serves to avoid minor minima during the optimization process.

- **Minimum and maximum tunnelling current values:** Hypothetically, incorporating these values could guide the optimization process. However, neither the global min/max values nor the mean of the lowest/highest portion of current values improved error or error consistency.
- **Outcome correlations:** Lab based observation suggest correlations for manipulation outcomes. A successful translation in one step may increase the probability of a repeated successful translation in the subsequent step. Introducing a binary input parameter encoding whether the preceding manipulation step did lead to a translation of more than half a surface lattice constant did not impact the model error. No outcome correlations were observed in the current dataset, where subsequent steps are spaced by multiple seconds, though this dynamic might differ for manipulation sequences with shorter time in between.

Further optimization of network layer structure or depth, batch size, learning rate or loss function did not lead to improvements for the mean average error on the test data set. The mixed-input model outperforms comparable fully-connected networks of equal layer depth, despite having only a fraction of the free parameters.

Another tested approach involved reverse predictions, aiming to determine the initial molecule's orientation and position relative to the tip based on the tunnelling current curve and estimated final molecule location and orientation, with deliberately added errors. This technique might be useful for chaining multiple manipulation steps without intermediate imaging. By precisely identifying the initial state, reverse prediction would help to prevent cumulative error accumulation, reducing the uncertainty range significantly. However, only very poor reverse prediction performance was achieved.

4.3.2 Rotation angle prediction

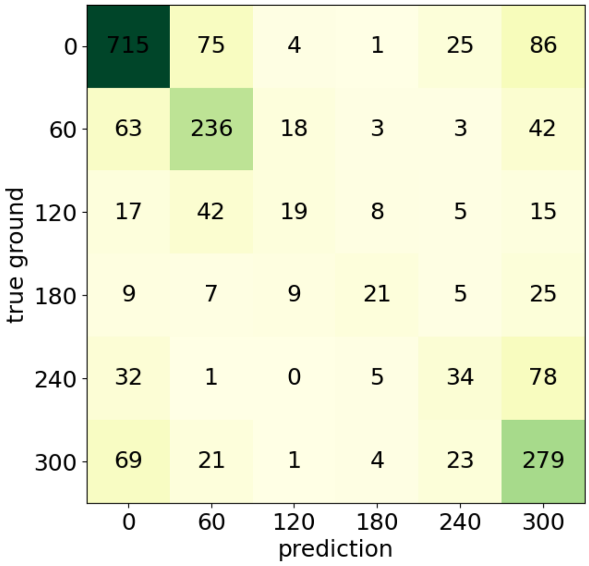
For predicting the rotation angle, which has six classes, the metric Categorical Cross-Entropy is a combination of a SoftMax activation function with Cross-Entropy loss, commonly used for multi-class classification. The metric for performance evaluation is the prediction accuracy in the test data set, which is the fraction of predicted labels coinciding with the true labels. We benchmark this metric against a prior, which is the most frequent rotation class (0°) with a share of 45.2%.

The same minimal set of input parameters used for predicting the translation vector components gives in the mixed-input classification network a maximum accuracy of 65%. This value is far from optimal, especially when considering, that the prior is 45%. Tab. 3 again breaks down the contribution of the different input parameters.

Table 3: Accuracy on test set in five training runs using different combinations of input parameters, compared with the prior, stating that no rotation occurs in all actions. The average value across these five runs follows the ascending trend.

	Tip position	Tunnelling current	Initial orientation	Accuracy
Null Hypothesis	-	-	-	44.2 %
Model 1	Included	-	-	58.1 %
Model 2	Included	Included	-	63.0 %
Model 3	Included	Included	Included	64.9 %

The classification and its errors become clearer when looking at the confusion matrix in Fig. 24, which groups the predicted classes based on their true counterparts. In an ideal scenario of perfect accuracy, only the diagonal elements would have nonzero values. It is noteworthy that more frequently occurring rotation angles are predicted more frequently, reflecting their prevalence in the dataset. The model tends to predict 0° most frequently, given its high occurrence in the dataset (45.2%). Consequently, the bin for correctly categorized 0° rotations has the highest number of entries, followed by accurately predicted rotations of +60° and -60°. Conversely, less common rotation angles are seldom categorized accurately due to their lower frequency. Erroneous predictions often cluster around the correct 0°, -60°, and +60° bins, indicating that predictions commonly deviate by a single rotation step.



true ground \ prediction	0	60	120	180	240	300
0	715	75	4	1	25	86
60	63	236	18	3	3	42
120	17	42	19	8	5	15
180	9	7	9	21	5	25
240	32	1	0	5	34	78
300	69	21	1	4	23	279

Figure 24: Confusion matrix displaying the prediction of rotation angles. Columns represent predicted rotation values, while rows depict true observed rotation angles, classified into six categories. The shading reflects the number of elements in each bin, with darker shades indicating a higher count.

4.3.3 Dataset size requirements

The network's performance, as described above, was achieved using the entire dataset comprising over 12,000 points, with 64% allocated for training and the remainder for validation and testing. While larger datasets generally enhance predictive accuracy, obtaining such extensive datasets is not always feasible. Hence, understanding how network performance varies with the size of the training data becomes imperative.

To explore this aspect, subsets of varying sizes were extracted from the initial segments of the complete dataset for training the network, followed by validation and testing. This sampling approach mirrors real-world scenarios, where a limited number of manipulation steps are taken with minimal prior knowledge. Notably, the reinforcement algorithm adjusts the translation and rotation priors, affecting metrics such as mean average error (MAE) and class accuracy. Consequently, comparing subsets based solely on MAE and accuracy might be misleading due to the strong influence of priors. Instead, we normalize MAE and accuracy by their respective priors to assess the additional information extracted by the model. These relative metrics are illustrated in two graphs in Fig. 25 for different subset sizes, with each data point representing the average value of five training runs, while error bars denote the standard deviation.

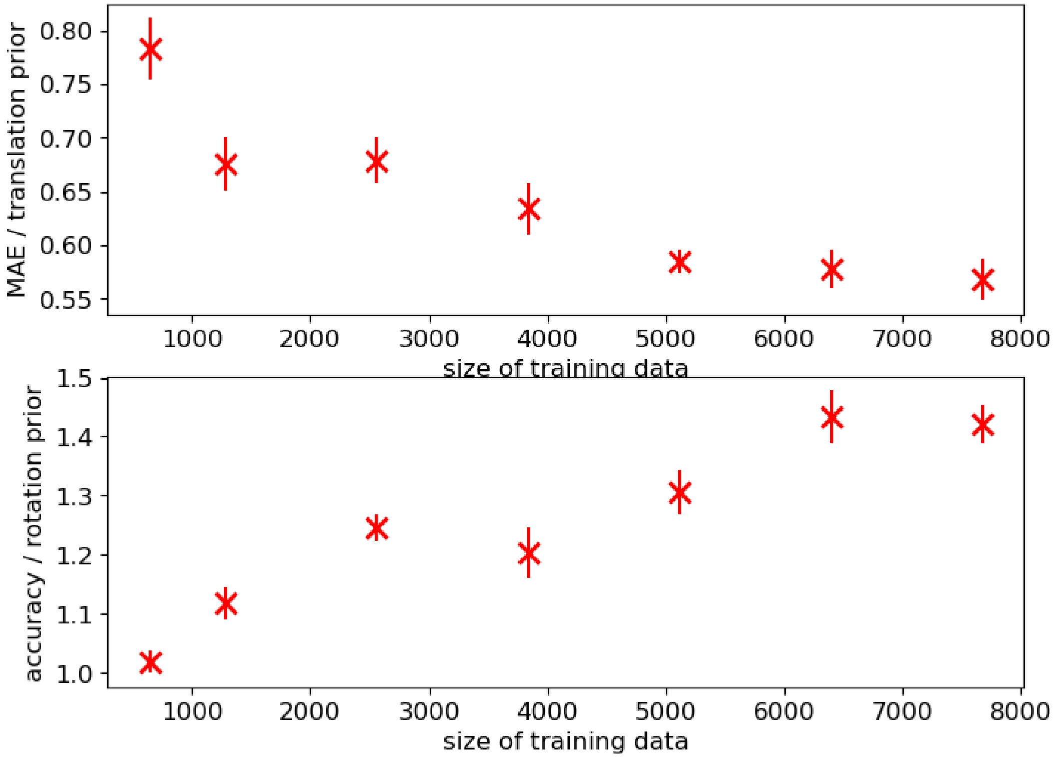


Figure 25: Mean squared error (MAE) and class prediction accuracy for various training set sizes, normalized by their respective prior values. Five training runs were conducted, with the cross denoting the mean value and the error bar representing the standard deviation.

The two metrics exhibit improvement with larger training set sizes: the relative MAE decreases, and the relative accuracy increases, although this trend is not strictly linear. Notably, there is a dip in relative rotation class accuracy at approximately 4000 dataset size, attributed to a change in image resolution from 64x64 to 32x32 pixels, which increases localization errors via the localization-maximization algorithm. Interestingly, this does not result in a similar decrease in MAE or translation prior as shown in the plot above. A slight dip in the MAE plot at 1300 dataset size is likely incidental, unrelated to changes in image resolution or other experimental factors.

The observed improvement in relative metrics suggests that the model gains more knowledge from larger datasets, albeit with diminishing returns. Higher resolution image snippets are undeniably advantageous for achieving quick and robust training results. However, faster imaging with lower resolution can provide more training data, compensating for localization errors as long as the MAE remains significantly above the resolution error limit. If other algorithms within the single-molecule manipulation framework necessitate a large number of data points, all available data should be utilized for the tunnelling current feedback loop. Alternatively, Fig. 25 indicates that the sixth data point with a dataset size of 5500 is a favourable compromise between performance and training speed for the mixed-input network with 19,300 free parameters. In scenarios with fewer available data, shrinking the network size to fewer parameters would be beneficial.

5 Conclusion and Outlook

This thesis set out to develop two components for an existing reinforcement-learning based framework for single-molecule manipulations: an efficient feedback loop to accelerate the process and improve understanding of the underlying mechanisms, and a single-shot object detection pipeline to further automate the process. A key requirement for all algorithms was the adaptability to arbitrary surface-adsorbate systems.

5.1 Feedback loop

Of all attainable parameters, the tunnelling current was identified as the most informative parameter for predicting manipulation outcomes with the highest possible accuracy to skip time-consuming imaging steps. The neural-network-based analysis achieved a mean average error (MAE) of 0.226 nm for translation vector components and 65% accuracy for rotation class prediction. A lower boundary for achievable MAE is the mean image resolution limit of 0.166 nm. When the initial state is precisely known through imaging, the predictive performance is sufficient to locate the DDNB molecule within a probability of residence area, where translation behaviour of roughly identical direction and magnitude is expected (as discussed in Sec. 4.2.2). This might vary for other molecules with different translation behaviours.

However, maintaining sufficient prediction accuracy over multiple manipulation steps without intermediate imaging is challenging. Location prediction errors are expected to escalate quickly, as inferred initial states are used in successive steps, leading to rapid error accumulation due to the divergence of translation vectors from the molecule centre. A compromise would be to schedule an imaging step after every second manipulation step, reducing time costs by nearly half, though at the expense of marginally decreased translation efficiency per step, if no rotations occur. This reduction is only achievable for the trained agent, as the training process requires precise feedback derived from imaging.

A primary concern, however, is the low accuracy of rotation prediction. Incorrectly inferred molecule orientation leads to significant deviations in the tip's relative position, resulting in vastly different translation outcomes. This is particularly problematic for the asymmetric DDNB molecule but might be of minor concern for more symmetric molecules or atoms with rotational symmetry. Given that rotation prediction accuracy is unlikely to exceed 90%, chaining multiple pulsing steps with a fixed-interval imaging step appears unfeasible.

Further efforts to extract more information from the discussed input parameters are unlikely to yield significant improvements. This is evident from the fact that various model configurations, hyperparameter combinations, and training set sizes have resulted in similar performances, with the best identified configuration being only marginally better. Another reason is the highly stochastic nature of the molecule manipulation process, where the system moves from identical initial states to a wide range of different outcome states. While different tunnelling current curves are recorded for these outcomes, these curves do not seem to fully capture the 3-dimensional information of the rotation angle and translation vector. They primarily consist of a time series of current plateau steps, each with a constant scalar current value, which resembles a 2-dimensional information package. Therefore, even for a specific experiment with a fixed initial molecule orientation and relative tip position, the movement outcome appears underdetermined by the current curve alone. Additional live-measured parameters are not available in STM experiments, one might try to increase manipulation specificity through methods like tip functionalization.

Currently, the only feasible strategy may be to predict whether the molecule state has changed and repeat the manipulation if it hasn't. This binary prediction has shown a reliability of 90% and could reduce some imaging steps. However, the time gain is minimal since the fully trained agent has a manipulation failure rate of only about 20%, translating to a reduction of the imaging steps by 20%.

More significant gains could be achieved using a Bayesian probability approach. Instead of a single point prediction, the network could pass down for each step the entire probability density function for location and orientation. Since pulsing steps are fast and binary prediction of motion/motionless is reliable, multiple pulsing steps can be performed, each updating the residence probability density, until movement is detected. Occasional imaging steps would then be used to precisely locate the molecule again. Under the assumption that each manipulation requires three pulsing steps, an imaging step is performed every five manipulations and the translation magnitude per step is not diminishing too much, the time cost per manipulation could be reduced by about 50%. This, however, requires further work and likely simulations for demonstration.

While the feedback loop does not currently meet the expectations, the developed automatic data extraction shows promise for investigating arbitrary surface-adsorbate systems. For the Ag(111)-DDNB system, we were able to qualitatively discuss the electrostatic and stochastic nature of single-molecule movement and quantify the probabilities of translation and rotation outcomes. The stochasticity revealed by the translation consistency study and the coupling of rotation and translation relative to the tip position appear to be general features, as they were also observed for the Ag(111)-H_2Pc system. Given the completeness of the recorded data, the extraction and analysis can be quickly adapted to any surface-adsorbate system. This accelerates the analysis of small datasets and enables the analysis of large datasets in the first place.

5.2 Object Detection

The second objective of this work was to develop a powerful object detection pipeline that performs well with only a single labelled training image. The current framework can be trained within minutes on new systems and reliably detects and distinguishes instances of objects in STM images, if the image resolution is sufficient and all classes of occurring objects are predefined. Implementation into the reinforcement-learning-based manipulation framework is pending and represents another critical step toward fully autonomous nanofabrication.

A future pathway in autonomous nanofabrication could involve linking positioned molecular assemblies into stable structures by forming bonds between these building blocks. This will necessitate distinguishing different reaction products and configurations, including by-products that emerge during assembly and are not initially defined as semantic object classes. Constant identification of new instances in images with high object-likeness and low similarity to existing object classes, followed by their introduction as new classes, could be valuable for such reactive nanofabrication.

List of Figures

List of Tables

References

[1] GitHub. GitHub Copilot, 2023. Accessed: 2024-06-05.
`https://github.com/features/copilot`

[2] M. Chen, J. Tworek, H. Jun, Q. Yuan, H. P. de Oliveira Pinto, J. Kaplan, H. Edwards, Y. Burda, N. Joseph, G. Brockman, A. Ray, R. Puri, G. Krueger, M. Petrov, H. Khlaaf, G. Sastry, P. Mishkin, B. Chan, S. Gray, N. Ryder, M. Pavlov, A. Power, L. Kaiser, M. Bavarian, C. Winter, P. Tillet, F. P. Such, D. Cummings, M. Plappert, F. Chantzis, E. Barnes, A. Herbert-Voss, W. H. Guss, A. Nichol, A. Paino, N. Tezak, J. Tang, I. Babuschkin, S. Balaji, S. Jain, W. Saunders, C. Hesse, A. N. Carr, J. Leike, J. Achiam, V. Misra, E. Morikawa, A. Radford, M. Knight, M. Brundage, M. Murati, K. Mayer, P. Welinder, B. McGrew, D. Amodei, S. McCandlish, I. Sutskever, W. Zaremba. *Evaluating Large Language Models Trained on Code.* CoRR **abs/2107.03374** (2021).

[3] OpenAI. ChatGPT-3.5, 2023. Accessed: 2024-06-05.
`https://www.openai.com/chatgpt`

[4] M. F. Crommie, C. P. Lutz, D. M. Eigler. *Confinement of Electrons to Quantum Corrals on a Metal Surface.* Science **262** (1993) 218.
doi:10.1126/science.262.5131.218

[5] B. Ramsauer, G. J. Simpson, J. J. Cartus, A. Jeindl, V. García-López, J. M. Tour, L. Grill, O. T. Hofmann. *Autonomous Single-Molecule Manipulation Based on Reinforcement Learning.* J. Phys. Chem. A **127** (2023) 2041.
doi:10.1021/acs.jpca.2c08696

[6] I.-J. Chen, M. Aapro, A. Kipnis, A. Ilin, P. Liljeroth, A. S. Foster. *Precise atom manipulation through deep reinforcement learning.* Nature Communications **13** (2022) 7499.
doi:10.1038/s41467-022-35149-w

[7] G. Binnig, H. Rohrer. *Scanning tunneling microscopy.* Surface Science **126** (1983) 236.
doi:10.1016/0039-6028(83)90716-1

[8] L. Bartels, G. Meyer, K.-H. Rieder. *Controlled vertical manipulation of single CO molecules with the scanning tunneling microscope: A route to chemical contrast.* Applied Physics Letters **71** (1997) 213.
doi:10.1063/1.119503

[9] G. Meyer, L. Bartels, K.-H. Rieder. *Atom manipulation with the STM: nanostructuring, tip functionalization, and femtochemistry.* Computational Materials Science **20** (2001) 443.

doi:10.1016/S0927-0256(00)00205-6. 9th Int. Workshop on Computational Materials Science

[10] D. Civita, M. Kolmer, G. Simpson, A.-P. Li, S. Hecht, L. Grill. *Control of long-distance motion of single molecules on a surface.* Science **370** (2020) 957. doi:10.1126/science.abd0696

[11] H. Bai, S. Wu. *Deep-learning-based nanowire detection in AFM images for automated nanomanipulation.* Nanotechnology and Precision Engineering **4** (2021) 013002.
doi:10.1063/10.0003218

[12] M. Rashidi, R. A. Wolkow. *Autonomous Scanning Probe Microscopy in Situ Tip Conditioning through Machine Learning.* ACS Nano **12** (2018) 5185. doi:10.1021/acsnano.8b02208. PMID: 29790333

[13] G. J. Simpson, V. García-López, P. Petermeier, L. Grill, J. M. Tour. *How to build and race a fast nanocar.* Nature Nanotechnology **12** (2017) 604. doi:10.1038/nnano.2017.137

[14] G. J. Simpson, V. García-López, A. Daniel Boese, J. M. Tour, L. Grill. *How to control single-molecule rotation.* Nature Communications **10** (2019) 4631. doi:10.1038/s41467-019-12605-8

[15] G. J. Simpson, V. García-López, A. D. Boese, J. M. Tour, L. Grill. *Directing and Understanding the Translation of a Single Molecule Dipole.* J. Phys. Chem. Lett. **14** (2023) 2487.
doi:10.1021/acs.jpclett.2c03472

[16] B. Ramsauer. Autonomous Manipulation of Phthalocyanine Molecules on Ag(111) Based on Reinforcement Learning, 2023. Unpublished experimental data.

[17] S. S. A. Zaidi, M. S. Ansari, A. Aslam, N. Kanwal, M. Asghar, B. Lee. *A survey of modern deep learning based object detection models.* Digital Signal Processing **126** (2022) 103514.
doi:doi.org/10.1016/j.dsp.2022.103514

[18] T.-Y. Lin, M. Maire, S. Belongie, J. Hays, P. Perona, D. Ramanan, P. Dollár, C. L. Zitnick. Microsoft COCO: Common Objects in Context. *Computer Vision – ECCV 2014* (herausgegeben von D. Fleet, T. Pajdla, B. Schiele, T. Tuytelaars). Springer International Publishing, Cham, 2014 S. 740–755.

[19] G. Huang, I. Laradji, D. Vazquez, S. Lacoste-Julien, P. Rodriguez. *A Survey of Self-Supervised and Few-Shot Object Detection.* IEEE Transactions on Pattern Analysis and Machine Intelligence **45** (2023) 4071.
doi:10.1109/TPAMI.2022.3199617

[20] C.-Y. Wang, A. Bochkovskiy, H.-Y. M. Liao. YOLOv7: Trainable Bag-of-Freebies Sets New State-of-the-Art for Real-Time Object Detectors. *2023 IEEE/CVF Conference on Computer Vision and Pattern Recognition (CVPR)*, 2023 S. 7464–7475.
doi:10.1109/CVPR52729.2023.00721

[21] G. Jocher, A. Chaurasia, J. Qiu. Ultralytics YOLO, 2023.
`https://github.com/ultralytics/ultralytics`

[22] YOLOv8—Ultralytics YOLOv8 Documentation. `https://docs.ultralytics.com/models/yolov8/`, 2023. Accessed: 2024-04-02.

[23] N. Otsu. *A Threshold Selection Method from Gray-Level Histograms.* IEEE Transactions on Systems, Man, and Cybernetics **9** (1979) 62.
doi:10.1109/TSMC.1979.4310076