Quantitative Image Simulation and Analysis of Nanoparticles

Madsen, Jacob; Schiøtz, Jakob; Hansen, Thomas Willum

Publication date:
2017

Document Version
Publisher's PDF, also known as Version of record

Quantitative Image Simulation and Analysis of Nanoparticles

Jacob Madsen

A thesis presented for the degree of
Doctor of Philosophy

Department of Physics
Technical University of Denmark
Denmark
November 2017

Preface

This thesis is submitted in candidacy of the Ph.D. degree at the Technical University of Denmark (DTU). The work has been carried out from September 2014 to November 2017 at the Center for Atomic-Scale Materials Design (CAMD) at the Department of Physics with Professor Jakob Schiøtz as main supervisor and Senior Researcher Thomas Willum Hansen as co-supervisor.

I would, first of all, like to thank my supervisor Jakob for always being enthusiastic and supportive about my work, especially during some difficult periods. I was privileged to have a great amount of freedom to pursue my own ideas, while still having enough guidance to keep me on track. I am also grateful to my colleagues at CAMD for creating a friendly and inspiring learning atmosphere. A special thanks to my office mates Ulrik Grønbjerg Vej-Hansen and Peter Mahler Larsen who were always helpful.

The work behind this thesis was carried out in close collaboration with the Center for Electron Nanoscopy (CEN). From CEN I would especially like to thank Thomas Willum Hansen, Jakob Birkedal Wagner and Pei Liu. I learned something new from each one of our weekly meetings and they serve as a concrete proof, that having a constant relationship with experimentalists can be healthy. In particular, I would like to thank Pei for letting me in on her scientific problems, which inspired large parts of this thesis.

For hosting me during a excellent external research stay, I would like to thank Professor Christoph Koch and the members of his research group at Humboldt-Universität zu Berlin. During my highly productive stay in Berlin, the Python-based image simulation program PyQSTEM was developed, and during my conversations with Christoph and Wouter van den Broek, I was further encouraged to pursue the neural network based method for structural analysis.

Marianne Ærsøe, Hanne Sørensen and Helle Ingeby Jørgen deserves much thanks for their patience, as do Ole Holm Nielsen for his help on supercomputing-related matters. I would also like to thank Henning Bo Nicolajsen for help with administrative matters and Jane Hvolbæk Nielsen for guidance.

Lastly, I gratefully acknowledge funding through Grant 1335-00027B from the Danish Council for Independent Research.

Kgs. Lyngby, November 2017
Jacob Madsen

Abstract

Materials science increasingly relies on powerful microscopes to study the relationship between a material property and its underlying structure. Understanding this relationship is critical for catalysis, due to the importance of structure at the nanoscale. High-Resolution Transmission Electron Microscopy (HRTEM) has become a routine analysis tool for structural characterization at atomic resolution, and with the recent development of in-situ TEMs, it is now possible to study catalytic nanoparticles under reaction conditions. However, the connection between an experimental image, and the underlying physical phenomena or structure is not always straightforward. The aim of this thesis is to use image simulation to better understand observations from HRTEM images.

Surface strain is known to be important for the performance of nanoparticles. Using simulation, we estimate of the precision and accuracy of strain measurements from TEM images, and investigate the stability of these measurements to microscope parameters.

This is followed by our efforts toward simulating metal nanoparticles on a metal-oxide support using the Charge Optimized Many Body (COMB) interatomic potential. The simulated interface structures are used as input for image simulations, to understand how support-induced strain influences a HRTEM image.

This thesis also introduces two novel analysis tools for atomic-resolution images. The first tool is an automatic method for calculating strain from HRTEM images with several advantages over previous methods. The second tool, is a neural network based algorithm for recognition of the local structure in images. The neural network was trained entirely from image simulations, but is capable of making correct predictions on experimental images.

Resumé

Materialevidenskaben afhænger i stigende grad af kraftfulde mikroskoper til at undersøge forholdet mellem en materiale-egenskab og den underlæggende struktur. At forstå dette forhold er særligt vigtigt indenfor katalyse, hvor strukturen på nanoskala er ekstremt vigtig. 'High-resolution Electron Microscopy' (HRTEM) er en meget udbredt mikroskopi-metode med atomar opløsning, og med den nye udvikling indenfor in-situ TEM er det muligt at undersøge nanopartiklerne under reaktionstilstande. Forbindelsen mellem et eksperimentelt billede og den underlæggende struktur er dog ikke altid klar. Målet med denne afhandling er derfor at bruge simulerede billeder til bedre at forstå eksperimentelle observationer.

Det er kendt, at tøjning (på engelsk 'strain') ved overfladen er vigtigt for ydeevnen af nanopartikler. Ved hjælp af simuleringer estimerer vi præcisionen og nøjagtigheden af målinger af tøjning og undersøger, hvor stabile disse målinger er overfor ændringer af mikroskopets parametre.

Dette bliver efterfulgt af vores arbejde henimod simulering af metal-nanopartikler på et substrat af metaloxid ved hjælp af interatomare potentialer. De simulerede strukturer bliver brugt som input i en billedsimulering for at forstå, hvordan de strukturer, som opstår ved grænsefladen mellem partiklen og oxidet, påvirker billedet.

Denne afhandling introducerer også to nye værktøjer til at analysere billeder med atomar opløsning. Det første værktøj er en automatisk metode til at beregne tøjning fra HRTEM-billeder med flere fordele i forhold til tidligere metoder. Det andet værktøj er en algoritme baseret på et neuralt netværk, der automatisk kan genkende lokale strukturer i billeder med atomar opløsning. Det neurale netværk er trænet udelukkende med simuleret data, men kan lave korrekte forudsigelser på eksperimentelle billeder.

Contents

Chapter 1

Introduction

With dwindling oil reserves worldwide and increasing environmental impact of burning fossil fuels, it is imperative that we find new technological solutions. There is no single replacement for fossil fuels, however catalysis is likely to play a major role. Catalysis is already a big workhorse in the chemical industry, where it is involved in nearly 90 % of all chemical processes: including the production of fuels, plastics and fertilizers [1]. Developing materials to act as catalysts with improved performance have been a major area of many researchers over the past decade. With a limited supply of the precious metals often used in these catalysts, their effective use becomes increasingly important.

In heterogeneous catalysis, solids are used to catalyze reactions of molecules in gaseous or liquid form. The reactants adsorb on the solid surface, their bonds are broken and new ones formed, the products then desorb from the surface. The catalytic process exploits the weakening of the internal bonds of a molecule, as it interacts with the solid surface. Metal catalysts normally consists of nanoparticles, deposited on a high surface area substrate in order to maximize the ratio between the surface area of the catalyst and the amount of metal used.

Modern materials science relies on studying the fundamental relationships between structure and property, for example the relationship between catalyst surface area and increase in rate of a chemical reaction (the catalyst activity). Understanding how different physical processes and phenomena impact the surface area of a nanoparticle, provides valuable input towards establishing new synthesis methods for improved nanostructures.

The surface area depends not only on the particle size, but also on their shape and morphology, the latter being influenced by the substrate and chemical environment [2]. The morphology does not stay constant during use of the catalyst, it changes both in response to the instantaneous environment and over time, where sintering leads to increased particle size. The surface area is not the only structural parameter that determines activity. For most catalysts the chemical reaction occurs only on sites with a specific atomic structure [3] and in addition, dramatic improvements of the activity can be gained by even relatively small surface strains [4].

Electron microscopes allow us to study materials at the atomic scale, and directly observe the physical structures and phenomena that gives rise to desireable or undesireable material properties. The capability to introduce gases into the microscope, now opens up the possibility to study changes in particle morphology with high temporal and spatial resolution under reaction conditions [5].
However, the connection between an experimental image and an underlying physical phenomena or structure is not always straightforward. Effective materials research relies on both experimental and theoretical techniques. A synergistic feedback loop has developed where experimental measurements are used as input to computer simulations, which in turn provide a deeper understanding of the underlying physical phenomena.
The association between experiment and theoretical calculations is especially strong, when the theoretical calculations directly simulate the experiment. Simulation of electron microscopy images allows us to start from an assumed structure and obtain a corresponding image approximating the experiment. This allows researchers to strengthen or reject a hypothesis, or go back and modify the assumptions of the theoretical calculations until agreement is reached. Image simulation can also act as the link between experiment and theoretical calculations that predicts the structure and properties of materials using quantum mechanics. Lastly, since the true structure is defined, image simulation also facilitates easy prototyping and benchmarking of analysis methods. Although, this thesis stays close to concrete experimental measurements, it is firmly to the theoretical side.

1.1 Outline

The rest of this thesis is ordered as follows:
Chapter 2 describes the electron microscope and how to simulate images. We also present the simulation program PyQSTEM.
Chapter 3 gives an introduction to measuring small structural changes (i.e. strain) from electron microscopy images. We also introduce a novel method for strain measurements from images.
Chapter 4 describes our results on the precision and accuracy of strain measurements from nanoparticles.
Chapter 5 details our efforts towards simulating supported nanoparticles. In addition, we show results for measuring the strain in supported nanoparticles.
Chapter 6 presents a method based on deep learning for automatically extracting quantitative structural information from atomic resolution images.

Chapter 2

Modeling the Transmission Electron Microscope and PyQSTEM

High-resolution transmission electron microscopes (HRTEM) are a type of microscope, which forms the image by transmitting electrons through the specimen. The tiny de Broglie wavelength of high-energy electrons and strong electron-matter interaction, enables the instrument to capture images of extreme resolution - even resolve individual atoms. The TEM has, since its introduction in the 1930s, evolved into a standard instrument for characterizing materials at the nanoscale [6]. The image formation in HRTEM relies on phase contrast. Since phase is not directly observable, this means that we observe the interference in the image plane of the electron beam with itself. In phase-contrast imaging, contrast is not necessarily intuitively interpretable, principally due to the large influence of aberrations resulting from imperfect imaging lenses. Hence, simulation is an important tool for relating an image to the specimen structure [7].
In this chapter, we detail the process of simulating electron microscopy images. We also introduce the Python-based program for image simulation, called PyQSTEM, which was developed during this project.

2.1 The instrument

A schematic cross section of a modern HRTEM instrument is shown in Fig. 2.1(a). From the top down, the first component is the electron source. Modern instruments has a field emission gun instead of a thermionic source for improved brightness and coherence [6]. A system of electromagnetic condenser lenses transfer the accelerated electrons onto the specimen as a plane wave (ideally). The the exit wave emerges at the other side of the specimen, after interacting with the electrostatic potential of the nucleii and electrons in the specimen through elastic scattering (ideally).
The objective lens focus the exit wave onto the back-focal plane, after which projector lenses are used to expand the beam onto an imaging device. This can be a layer of photographic film, or a

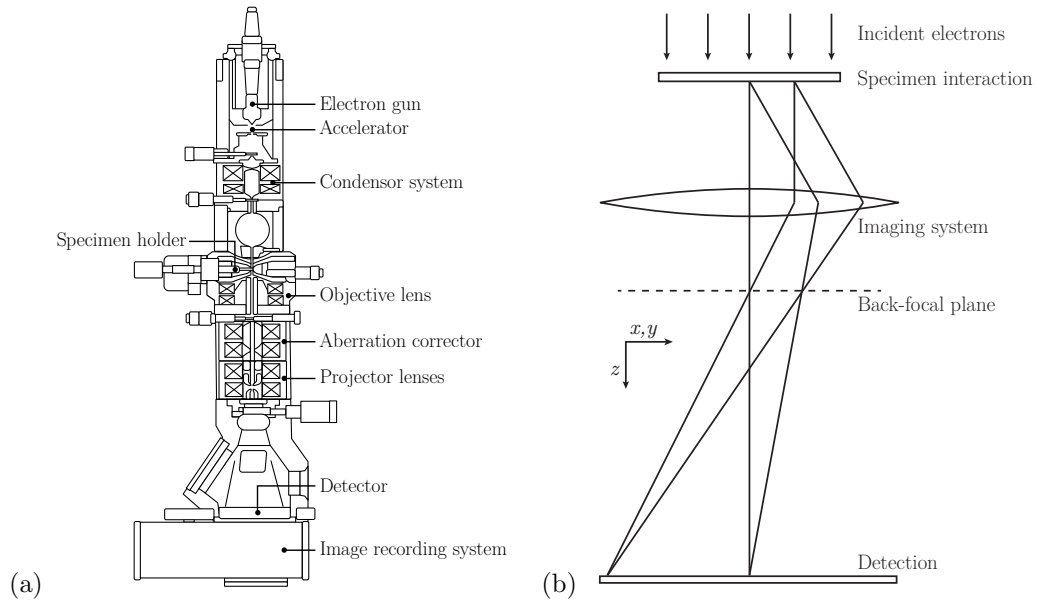


Figure 2.1: **(a)** Cross section of a generic HRTEM. Adapted from Wikimedia Commons. **(b)** Simplified model (not to scale) of a HRTEM. The condenser lenses (above the top of the drawing) and projector lenses have been ignored.

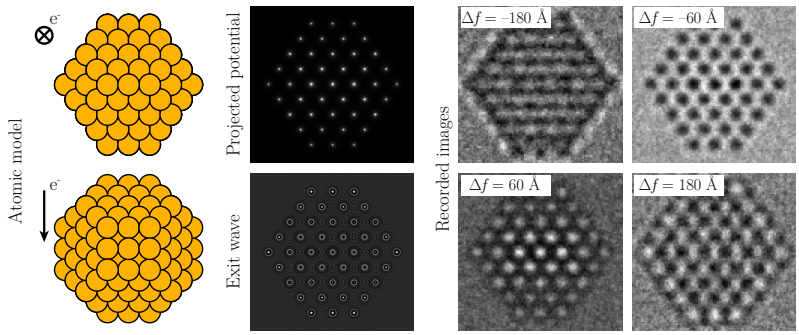


Figure 2.2: The atomic coordinates and species defined by the input atomic model are used to generate the electrostatic potential. The multislice algorithm is used to propagate the input plane wave through the specimen to produce the exit wave function. The aberrations of the objective lens are applied to the exit wave function through the contrast transfer function, defining for example the defocus. Lastly, the detected intensity is calculated from the intensity of distorted exit wave, taking into account the point spread function of the detector and finite electron dose.

sensor such as a charge-coupled device (CCD). The wave function is magnified by an amount equal to the ratio of distances from the objective lens to the specimen and specimen to the image plane. For the rotationally symmetric lens design, positive spherical aberrations are unavoidable [7]. In modern microscopes, this is corrected with an aberration corrector. This have resulted in dramatic improvements to the instrumental resolution over the last decade [6].

As a whole transmission electron microscopes are incredibly complicated instruments. However, much of this complexity can be ignored when simulating images. For example, the condenser system, once aligned, can be reduced to the properties of the illuminating rays (i.e. coherence and angular distribution), and since any defects in the objective lens are greatly magnified, the projector lenses have comparatively little effect on the final image resolution.

The image formation process can be simplified to the model described by Fig. 2.1(b). The model consists of three parts: Interaction of the incident plane wave with the specimen, aberrations introduced by the objective lens and finally the detection of the wave function intensity. An overview of the steps involved in the modelling the electron microscope is illustrated in Fig. 2.2. Each of the steps will be treated in the following sections.

2.2 Specimen potential

The electron beam interacts with the specimen through the Coulomb potential of the electrons and nucleii in the specimen. The electron charge distribution of an atom can be calculated from first-

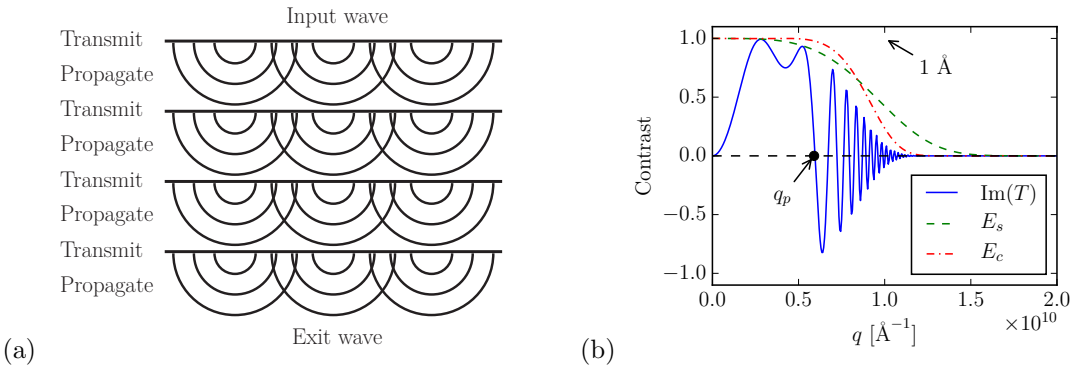


Figure 2.3: **(a)** The multislice method successively transmits the electron wave through each slice of the potential and then propagates it to the next slice. **(b)** The contrast transfer function of a HRTEM operated at 300 kV with $C_s = 0.6$ mm and $\Delta f = -41.25$ nm (Scherzer defocus). The point resolution q_p corresponds to the first crossing of zero for the imaginary part of the CTF. At Scherzer defocus q_p is maximized. The partial coherence envelopes dampens the high frequency components of the CTF leading to a delocalization in real space.

principles electronic structure calculation, while an atomic nucleus is a point charge at the resolution of a TEM. Given a charge distribution, the potential can be obtained via Poisson's equation. Simulation codes generally include a parametrization of the atomic potentials, with a table of parameters for each element fitted to Hartree-Fock calculations. The full specimen potential, $V(\boldsymbol{r})$, is then obtained as a linear superposition of the atomic potentials

$$V(\boldsymbol{r}) = \sum_i V_i(\boldsymbol{r} - \boldsymbol{r}_i) \quad , \tag{2.1}$$

where $V_i(\boldsymbol{r})$ is the atomic potential of the i'th atom. This model obviously neglects any effect due to bonding and charge transfer. However, since the nucleus and core electrons constitutes most of the charge in an atom, this is a reasonable approximation, that gets progressively better as the atomic number increases. Nonetheless, due to recent improvements in microscopes, interest in going beyond this approximation have started. For example, it have been shown that simulated images based on potentials from DFT calculations can provide a better match with experimental data for light elements [8].

2.3 Multislice algorithm

The energy of the incident plane wave (100 - 1000 keV) is much greater than the specimen potential, which provides only minor perturbations on the forward motion of the electrons. Hence, it is useful

to write the wave function, ψ, of the propagating electrons as a slowly varying plane wave along the optical axis, z, with an amplitude modulation

$$\psi(\boldsymbol{r}) = \phi(\boldsymbol{r}) \exp(2\pi i z/\lambda) \quad , \tag{2.2}$$

where $\boldsymbol{r} = (x, y, z)$ and λ is the de Broglie wavelength of the electrons. Substituting this into the Schrödinger equation [9] we obtain

$$-\frac{\hbar^2}{2m}\left[\nabla_{xy}^2 + \frac{\partial^2}{\partial z^2} + \frac{4\pi i}{\lambda}\frac{\partial}{\partial z} + \frac{2meV(\boldsymbol{r})}{\hbar^2}\right]\phi(\boldsymbol{r}) = 0 \qquad \nabla_{xy}^2 = \frac{\partial^2}{\partial x^2} + \frac{\partial^2}{\partial y^2} \quad . \tag{2.3}$$

In the high energy approximation, we assume that the wavefunction varies slowly in the z-direction compared to the potential and that the wavelength is small, thus

$$\left|\frac{\partial^2 \phi}{\partial z^2}\right| \ll \left|\frac{1}{\lambda}\frac{\partial \phi}{\partial z}\right| \tag{2.4}$$

Hence, Eq. (2.3) can be written as a first order differential equation in z

$$\frac{\partial \phi(\boldsymbol{r})}{\partial z} = \left[\frac{i\lambda}{4\pi}\nabla_{xy}^2 + i\sigma V(\boldsymbol{r})\right]\phi(\boldsymbol{r}) \quad , \tag{2.5}$$

where $\sigma = 2\pi me\lambda/h^2$ is the interaction parameter. This equation is integrated numerically by slicing the potential into thin slices, such that the influence of each slice can be approximated as a simple phase shift of the wave function. The wave function is propagated between slices as a small angle outgoing wave (Fresnel diffraction), see Fig. 2.3(b). The transmission and propagation across a single slice can be written

$$\phi(x, y, z + \Delta z) = p(x, y, \Delta z) * [t(\boldsymbol{r})\psi(\boldsymbol{r})] + \mathcal{O}(\Delta z^2) \quad , \tag{2.6}$$

where $*$ represents a convolution. The transmission function, $t(\boldsymbol{r})$, for the portion of the potential between z and $z + \Delta z$ is

$$t(\boldsymbol{r}) = \exp\left[i\sigma \int_z^{z+\Delta z} V(x, y, z)dz'\right] \quad , \tag{2.7}$$

and the Fresnel propagator $p(x, y, \Delta z)$ is

$$p(x, y, \Delta z) = \frac{1}{i\lambda z} \exp\left[\frac{i\pi}{\lambda \Delta z}(x^2 + y^2)\right] . \tag{2.8}$$

The wave at the exit plane of the specimen is obtained by sequentially applying Eq. (2.6) starting with an assumed input wave. A full derivation of the above may be found in Kirklands book [7]. The convolution in Eq. (2.6) can be performed efficiently by utilizing the Fast Fourier Transform (FFT). The implemented form of Eq. (2.6) is

$$\phi_{z+\Delta z}(x, y) = \mathcal{F}^{-1}\{P(k_x, k_y, \Delta z)\mathcal{F}[t(\boldsymbol{r})\phi(\boldsymbol{r})]\} \quad , \tag{2.9}$$

where P is the Fourier transform of Eq. (2.8). The computational cost for the FFT scales as $N\log(N)$ with the number of samples N. Typically, partial coherence is neglected in the interaction with the specimen, since the influences of energy spread and angular spread are insignificant.

2.4 Imaging system

Ideally a lens forms a spherical wave converging on or emerging from a single point. In practice aberrations cause the wave to deviate from a spherical surface. This deviation can be represented as a phase error χ. The specimen is always very near the optical axis so positional errors can be ignored leaving only the angular deviations α. It is customary to write the phase error as a power series expansion in α, or equivalently the spatial frequencies

$$\chi(q_x, q_y) = \frac{2\pi}{\lambda}\sum_{n,m} C_{n,m}\alpha_x^n\alpha_y^m = \frac{2\pi}{\lambda}\sum_{n,m} C_{n,m}\lambda^2 q_x^n q_y^m \quad , \tag{2.10}$$

where (q_x, q_y) is the spatial frequencies and the relation $\alpha = \lambda q$ was used.
If the microscope is well aligned then off-axis aberrations (astigmatisms) are small and the phase error is dominated by the first two isotropic terms

$$\chi(q) \approx \frac{2\pi}{\lambda}\left(\frac{\lambda^2 q^2}{2}\Delta f + \frac{\lambda^4 q^4}{2}C_s\right) \quad , \tag{2.11}$$

where $\Delta f = C_{1,1}$ is the defocus and $C_s = C_{3,3}$ is the third order spherical aberration. The effect of a coherent imaging system on an exit wavefunction, ψ_0, can thus be modelled by a Fourier space multiplication

$$\psi(\boldsymbol{q}) = \psi_0(\boldsymbol{q})\exp[-i\chi(\boldsymbol{q})] = \psi_0(\boldsymbol{q})T(\boldsymbol{q}) \quad , \tag{2.12}$$

where T is called the contrast transfer function (CTF). Choosing the optimum defocus is crucial to fully exploit the capabilities of a HRTEM. If one sets the defocus to zero, the sample is in focus, and the contrast is localized (no information overlap from other parts of the specimen). Unfortunately, the CTF now becomes a function that oscillates quickly with $C_s q^4$. This means that the contrast is reversed for certain spatial frequencies, and the image becomes difficult to interpret. In Scherzer defocus, one aims to counter the term in q^4 with the parabolic term $\Delta f q^2$. Thus by choosing the right defocus value Δf one flattens $\chi(q)$ and creates a wide band, where low spatial frequencies q are transferred into image intensity with a similar phase, see Fig. 2.3(b).
In modern aberration corrected microscopes, the C_s value can be very small. This means that more aberrations have to taken into accoundt in Eq. (2.11), in particular residual astigmatism ($C_{1,2}$ and $C_{2,1}$) and fifth order spherical aberration ($C_{5,5}$).

2.4.1 Partial coherence

Partial spatial and temporal coherence cause a delocalization or blurring of the image. The electron beam illuminating the specimen, will always have a small distribution of angles. These are quantified by the convergence semi-angle β which is subtended at the sample by the condenser aperture.
In principle each angle is incoherent with other angles, so the images due to each angle should be summed incoherently by adding intensities $|\psi|^2$ and not amplitudes ψ. However, in modern microscopes, particularly microscopes equipped with a field emission gun (FEG), the distribution of angles are so narrow that the wave functions can be summed coherently (the quasi-coherent approximation) [7, 10]. If $p(k_\beta)$ represents the (probability) distribution of illumination angles, then the summed wave function can be written

$$\psi(\boldsymbol{q}) = \int \psi_0(\boldsymbol{q}) T(\boldsymbol{q}+\boldsymbol{q}_\beta) p(\boldsymbol{q}_\beta) d\boldsymbol{q}_\beta \quad , \tag{2.13}$$

where $\boldsymbol{q}_\beta$ defines a particular direction of propagation, with the probability

$$p(\boldsymbol{q}_\beta) = \frac{1}{q_s^2 \pi} \exp\left(-\frac{q_\beta^2}{q_s^2}\right) \quad , \tag{2.14}$$

and $q_s = \beta/\lambda$ is the $1/e$ width of the Gaussian distribution. Note that in Eq. (2.13), the distribution of incident directions are incorporated into $T(\boldsymbol{q})$ rather than $\psi_0(\boldsymbol{q})$, an approximation which is valid for quasi-coherent illumination [10]. Evaluation of Eq. (2.13) gives

$$\psi(\boldsymbol{q}) = \psi_0(\boldsymbol{q}) T(\boldsymbol{q}) E_s(\boldsymbol{q}) \quad , \tag{2.15}$$

where E_s is the spatial coherence envelope function

$$E_s(\boldsymbol{q}) = \exp\left(-\frac{\beta}{4\lambda^2}\left|\frac{\partial \chi(\boldsymbol{q})}{\partial \boldsymbol{q}}\right|^2\right) \quad . \tag{2.16}$$

A small spread in energy, ΔE, of the incident electrons is equivalent to a small spread in defocus, due to the chromatic aberration of the objective lens. Fluctuations in the focusing currents, ΔI, in the objective lens also produce an incoherent spread in defocus. Combining these effects, the $1/e$ width of the distribution of defocus values (the focal spread) can be written

$$\Delta = C_c \sqrt{4\Delta I^2 + \Delta E^2} \quad . \tag{2.17}$$

As in our treatment of spatial coherence above, we assume that Δ is small. Then, it can be shown that focal spread can be approximated as a simple envelope function

$$E_t(\boldsymbol{q}) = \exp\left(-\frac{\delta^2}{4}\left|\frac{\partial \chi(\boldsymbol{q})}{\partial \Delta f}\right|^2\right) \quad . \tag{2.18}$$

Blurring can also be caused by all kinds of noise leading to a random deflection of the image relative to the detector, such as vibrations, drift of the stage, and time-dependent magnetic noise fields resulting from eddy currents in the material of the lenses [11]. Assuming that the image deflection is small and follows a Gaussian distribution, this can be included as an additional envelope

$$E_d(\boldsymbol{q}) = \exp\left(-\frac{\sigma^2 q^2}{2}\right) \tag{2.19}$$

where σ is the $1/e$ amplitude of the deflection. Thus, in the quasi-coherent approximation the wave function at the image plane, given the exit wave ψ_0, can be written

$$\psi(\boldsymbol{q}) = \psi_0(\boldsymbol{q})T(\boldsymbol{q})E_s(\boldsymbol{q})E_t(\boldsymbol{q})E_d(\boldsymbol{q}) \quad . \tag{2.20}$$

2.5 Temperature

In the creation of the specimen potential the atoms were treated as completely stationary. At room temperature, at which most electron microscopy is conducted, the atomic vibrations are small compared to a typical interatomic distance, so the effect of temperature is expected to subtle. In particular, the thermal vibrations lead to a diffuse background intensity, which lowers the contrast of the images [12]. A general theory of imaging and diffraction in the presence of thermal vibrations can be rather involved. However, there exists a simple, if somewhat brute force approach to numerically simulate, the effects of thermal vibrations in the specimen, this is known as the frozen phonon approximation. It is a semi-classical model based on the assumption that a single high-energy electron passing through the specimen, at about half the speed of light, can only probe a single frozen "snapshot" of the vibrating crystal. The image is produced by averaging incoherently over many snapshots, where the atoms are slightly displaced from their equilibrium positions. The frozen phonon model has been shown to be numerically equivalent to the full quantum-mechanical treatment of the inelastic phonon scattering process [12]. The displacements can be calculated accurately using MD, however most simulation codes use the Einstein approximation, where the atomic vibrations are assumed to be independent [13] and the vibrational magnitudes are tabulated. Temperature can also be modelled with the less accurate, but much faster method, of using a time-averaged potential [14].

2.6 Detection

TEM images are typically recorded using charge-coupled detectors (CCD) with a fiber-optics coupled scintillator. In the ideal case, each electron is only detected by one of the detector pixels. In practice, a single imaging electron can cause signals in more than one pixel because of multiple scattering within the scintillator material and the creation of an excitation volume [15]. This effect

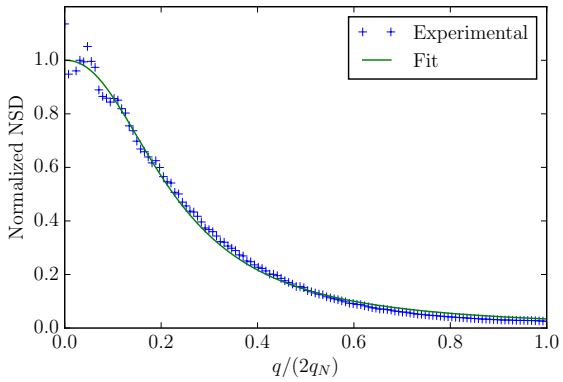


Figure 2.4: The normalized noise spectral density of an experimental image and a fit of the form in Eq. (2.21).

can be expressed by the point-spread function (PSF), describing the response of a detector to a point source or point object (i.e. the impulse response). The effect of the PSF, or equivalently its Fourier transform the modulation transfer function (MTF), are typically implemented through a parametrization of its influence. The easiest method of measuring the MTF is through the spectral density of a noisy image without a signal, i.e. the noise spectral density (NSD) [15]. A more accurate method for measuring the MTF exists, however this requires a specialized measurement [16]. In our simulations, we use the following parametrization for the MTF

$$\mathrm{MTF}(q) = \frac{1 - a_1}{1 + \left(\frac{q}{2a_2 q_N}\right)^{a_3}} + a_1 \quad , \tag{2.21}$$

where q_N is the Nyquist frequency related to the sampling δ by $\delta = 1/2q_N$. A common alternative choice for parametrizing the MTF is a mix of an exponential function and a Gaussian.
In HRTEM we can usually assume that the noise is dominated by shot noise, hence the measured electron count in each pixel can be modelled by a Poisson distribution [17]. The average number of electrons N collected by the i'th detector pixel is given by

$$N_i = D\delta^2 I_i \tag{2.22}$$

The signal-to-noise ratio of the whole image is given by [18]

$$\mathrm{SNR} = \frac{\bar{N}}{\sigma(N)}\,, \tag{2.23}$$

where $\bar{N}$ is the average number of electrons per pixel and $\sigma(N)$ is the standard deviation of the number of electrons collected by each pixel. In the limit of low dose this can be reduced to [19]

$$\mathrm{SNR} = \sqrt{\bar{N}} = \sqrt{DI}\delta\,, \tag{2.24}$$

whereas in the limit of high dose other sources of noise are dominant (e.g. thermal noise) and the SNR becomes constant. We are only including shot noise in the simulations.

2.7 PyQSTEM

There exists a vast number of commercial and open source programs for simulating the electron microscope. Most of these programs are based on a graphical user interface. This useful for some applications, however it is impractical for purposes that require a large numbers of automatically generated simulations such as statistical analysis, optimization and machine learning. In addition, many simulation programs are rather rigid with respect to modifying individual aspects of a simulation, such as supplying custom potentials from DFT. These packages are typically also stand-alone requiring import/export to other programs for further analysis. PyQSTEM was developed during this project, in order to provide a single scripting environment, for doing everything related to image simulation from model building to analysis.

PyQSTEM is a Python based interface and extension to the well-established multislice simulation program QSTEM based on C++ and the FFTW library. It was created in collaboration with Christoph Koch the original creator of QSTEM.

Python has become a leading language for scientific applications. This is especially due to the large number of free and open-source numerical libraries. Packages such as numpy, scipy and scikit-image provide easy access to a multitude of common tools for data analysis, including many tools which are common in the analysis of HRTEM images. Additionally, Python is dominant in machine learning due to libraries such as scikit-learn and tensorflow.

In PyQSTEM, the task of setting up the atomic model is solved with the Atomic Simulation Environment (ASE), a very popular tool in the computational materials community [20]. ASE has modules for defining a wide range of different structures including nanotubes, bulk lattices, surfaces, and nanoparticles. A large degree of manual control is provided, and import capability for most atomic structure files. Lastly, ASE makes it easier to integrate results from atomistic simulations into image simulations, including DFT potentials and phonons from MD simulations.

PyQSTEM and all its dependencies are open source under the GNU license. It is available on all platforms from the github repository [1].

2.7.1 HRTEM simulation with PyQSTEM

We exemplify the use of PyQSTEM through the process of simulating the image of the gold nanoparticle in Fig. 2.2. More advanced examples are available in the online repository. The atomic model

[1] https://github.com/jacobjma/PyQSTEM

is defined by specifying 6 layers in the (100) directions, 9 in the (110) directions and 5 in the (111) directions.

```
>>> from ase.cluster.cubic import FaceCenteredCubic
>>> surfaces = [(1, 0, 0), (1, 1, 0), (1, 1, 1)] # Miller indices
>>> layers = [6, 9, 5] # Number of layers
>>> nanoparticle = FaceCenteredCubic('Au', surfaces, layers)
>>> nanoparticle.rotate('y', 0.7853981) # Rotate by pi/4
>>> nanoparticle.center(vacuum = 2)
```

Above `nanoparticle` holds an ASE `Atoms` object, representing a collection of atoms of a given chemical species with given Cartesian positions. At the center of PyQSTEM is the `PyQSTEM` object. It connects to the QSTEM backend and handles memory allocation and cleanup. The main motivation behind this object is to minimize communication of large arrays between Python and FFTW, while maintaining the ability to work interactively. To run a multislice calculation it is necessary to attach a wave function and a specimen potential to the `PyQSTEM` object. Below we attach a plane wave at a sampling of 0.05 Å/pixel and an energy of 300 keV. We also calculate and attach the potential given the atomic model created above.

```
>>> from pyqstem import PyQSTEM
>>> qstem = PyQSTEM('TEM')
>>> qstem.build_wave(sampling=0.05, energy=300, type='plane')
>>> qstem.build_potential(nanoparticle, slice_thickness=0.5)
>>> qstem.run()
```

The `.run()` method passes the currently attached wave function once through the potential using the multislice algorithm, and the attached wave function is now the exit wave function. To continue we retrieve the wave function from the QSTEM backend into our Python session. The retrieved `Wave` object wraps a complex wave function (numpy array) with the energy and a sampling rate. We simulate the aberrations of the objective lens as well as partial coherence by applying a `CTF` object to the wave function (this essentially implements Eq. (2.20)).

```
>>> from pyqstem.imaging import CTF
>>> exitwave = qstem.get_wave()
>>> ctf = CTF(Cs=0.6*10**7, defocus='Scherzer',
              focal_spread=30, abberations={'a40' : 5*10**7})
>>> imagewave = wave.apply_ctf(ctf)
```

At this point the image is just the absolute square of the wave function. However, the `.detect` does several additional operations. In the following order; the image is downsampled to the detector resolution, shot noise is simulated as a Poisson process and a MTF is applied.

```
>>> from pyqstem.detection import MTF
>>> mtf = MTF({'a1' : 1, 'a2' : 0, 'a3' : .5, 'a4' : 2.3})
>>> image = imagewave.detect(resample=0.18, dose=10**4, mtf=mtf)
```

By default the MTF utilize the parametrization in Eq. (2.21), however a custom function may be used.

Other features

PyQSTEM includes many features not high-lighted above, these features are best explored in the online repository, however we have listed some of them below.

STEM Scanning Transmission Electron Microscopy is implemented, see tutorial online.

CBED Convergent Beam Electron Diffraction is implemented, see tutorial online.

Inline holography Any custom wave function can be set using the `.set_wave` method of the `PyQSTEM` object. Example applications includes inline holography simulations and phase plate exit wave reconstruction.

DFT potentials Any custom potential can be set using the `.set_potential` method of the `PyQSTEM` object. A guide for STEM and TEM simulations using a potential derived from DFT can be found in the repository. PyQSTEM also includes a solver for calculating the electrostatic potential given the all-electron density from DFT (and nucleii positions/charges).

MD frozen phonons Instead of supplying a single `Atoms` object to PyQSTEM a list of `Atoms` can be supplied. The main application is to simulate accurate thermal vibrations in the frozen phonon approximation using a list of snapshots from a molecular dynamics simulation. When a list of `Atoms` objects are supplied to PyQSTEM, a wave function is simulated for each, these are collected in a special `WaveBundle` object, which provides most of the same methods as the `Wave` object.

MTF estimation A tool for estimating the noise spectral density and converting this to a parametrized MTF.

Large-scale simulations The full 3d potential for very large atomic models are too memory intensive, hence PyQSTEM includes the possibility of dividing the unit cell in the direction of propagation and calculating the required division on the fly.

Exit wave reconstruction The iterative exit wave reconstruction algorithm by Allen et al. is implemented [10]. See tutorial online.

Jupyter integration PyQSTEM is designed to work well with Jupyter Notebooks [21]. This allows the user to create documents that can contain live code, equations, visualizations and explanatory text. The scripted workflow in Jupyter Notebooks provide a great framework for sharing simulations, since each step is documented and can be rerun, possibly with small customizations.

Parallelization A large number of tasks related to simulating the TEM can be trivially parallelized. For example, integration over thermal vibrations or each pixel in a STEM image. PyQSTEM uses Message Passing Interface (MPI) to parallelize these tasks. For easy maintenance and cross-platform compatibility, parallelization is handled in Python.

Chapter 3

Strain Analysis and Structural Template Matching

Strain is a key parameter for understanding many physical phenomena at the nanoscale. The mechanical and electronic properties of materials are directly related to the strain, and the response to an applied strain is fundamental to materials engineering. The surface strain is of particular importance for nanoparticles, as a possible tunable parameter that can be used to optimize the adsorption energies of surfaces for a particular catalytic reaction [4]. Platinum-based oxygen reduction catalysis is improved by weakening the binding of adsorbed oxygen intermediates by 0.1 eV, this can be achieved by a ~ 2 % compressive strain [22]. Strain in nanoparticles can be generated by a variety of sources: particle size, shape, twinning, by the lattice mismatch between metals in multimetallic core–shell nanoparticles or it can be induced by the supporting substrate [23]. Characterizing the influence of these effects requires a technique capable of measuring structural information at atomic resolution. X-ray diffraction and Raman spectroscopy are two widespread techniques that offer high strain precision (respectively 10^{-5} [24] and 10^{-4} [25]) but limited spatial resolution $\delta \approx 500$ nm which makes them unsuitable for characterizing individual nanoparticles.

In TEM, there are several methods that can be used to measure local strain including; transmission electron microscopy (TEM) [26, 27], convergent-beam [28] and nano-beam [29] electron diffraction (CBED, NBED), dark-field electron holography [30] (DFEH), and annular dark-field scanning transmission electron microscopy (ADFSTEM). Reviews and comparisons of the methods are given elsewhere [31, 32, 33], but notably, each of them have their unique issues. In CBED strain measurements can require significant sample tilt to achieve high-precision and can be vulnerable to crystal plane bending, while DFEH and NBED techniques each require additional hardware (a bi-prism, an additional condenser lens, and precession coils respectively) and more involved off-line data processing. Lastly, in STEM the serial nature of the image means that environmental effects can distort the image fidelity [34, 35] and the necessary atomic resolution generally requires aberration correction.

HRTEM have probably been the most widely used technique for strain mapping in nanoparticles, because of its availability and the the possibility of 2d strain mapping. However, careful choice of

imaging parameters must be chosen to avoid image artefacts or contrast inversions from sample-tilt, sample-thickness, defocus or aberration changes. Recently, there have been increased interest in measuring strain from ADF-STEM images due to the incoherent nature of the images. Both of these techniques rely on high resolution images that resolve the crystal lattice.

3.1 Strain analysis from images

There are several different approaches for extracting strain from a HRTEM image. These can be broadly classified into two different types: direct measurement of interatomic distances in real space or analysis in Fourier space. Although they were originally applied to conventional high resolution TEM images, the same methods are increasingly being applied to ADFSTEM images [35].

3.1.1 Geometric Phase Analysis

The most widely used technique for calculating strain from an atomic resolution image, is the Fourier space method known as Geometric Phase Analysis [36]. The technique have several advantages. No real space peak detection is required, hence GPA typically requires less user intervention. This is particularly an advantage when no clear individual peaks are available. GPA also does inherent noise filtering and spatial averaging, which leads to superior performance over real space methods, where these issues are not treated appropriately.
An image with intensities that vary as a function of position, $I(\boldsymbol{r})$, can be expressed by a Fourier series, summed over all reciprocal lattice vectors

$$I(\boldsymbol{r}) = \sum_{\boldsymbol{g}} H_g(\boldsymbol{r}) \exp(2\pi \boldsymbol{g} \cdot \boldsymbol{r}) \quad , \tag{3.1}$$

where $\boldsymbol{g}$ is a reciprocal lattice vector and the Fourier components, $H_g(r)$, are given by

$$H_g(\boldsymbol{r}) = A_g(\boldsymbol{r}) \exp(iP_g) \quad , \tag{3.2}$$

where $A_g(\boldsymbol{r})$ is the amplitude and $P_g(\boldsymbol{r})$ is the geometric phase. It can be shown [36] that the geometric phase is related to the displacement field as

$$P_g = -2\pi \boldsymbol{g} \cdot \boldsymbol{u} \quad . \tag{3.3}$$

In practice, the Fourier components $\tilde{H}_{\boldsymbol{g}}(\boldsymbol{k})$ are extracted in Fourier space by application of a mask (usually Gaussian) around the reciprocal lattice point $\boldsymbol{g}$. This is sometimes called Bragg filtering. An inverse Fourier transform yields $H_{\boldsymbol{g}}(\boldsymbol{r})$ and it is then trivial to calculate the displacement $\boldsymbol{u}$ in the direction of bmg. By repeating this process for a non-collinear $\boldsymbol{g}$, the full displacement field can be measured, which is then differentiated to get the strain.

Unfortunately, we do not think that GPA is a technique, that should be used for calculating strain in nanoparticles. And in particular, it should not be used for determining surface strain. An implicit assumption in GPA is that the change in geometric phase for the lattice fringes $\boldsymbol{g}$ is due to the displacement of the atoms in the material. This is almost always correct, prior to aberration correction, when the resolution of the image only is sufficient to resolve the crystal lattice. However, at higher resolutions the shape of the peaks are influenced by changes in the local structure can also contribute to the geometric phase. This leads to issues when large thickness variations are present, or if the structure changes from one area to another in the image (e.g. across an interface). The change in this additional phase will appear as a strain, even if there is no lattice strain [37].
Spatial averaging is implicit in GPA, this can enhance the precision of the technique, however it can become a problem in the vicinity defects. An example of where GPA fails, is for determining the strain at surfaces, where spatial averaging including the vacuum is nonsensical. This is demonstrated later in Fig. 3.6.

3.1.2 Real space analysis

Real space lattice analysis is based on locating peaks associated with the atomic columns that are recorded in an image. While conceptually simpler than GPA, it is significantly more complex in practice, requiring several steps.

1. As opposed to GPA, noise filtering is not inherent and have to be applied separately, common choices include Wiener filtering [38] and Gaussian filtering [39].

2. The positions the atomic columns have to be detected. This can be done automatically in some cases by employing simple metrics such as local intensity, separation and width of the intensity distribution.

3. The rough positions have to be refined to sub-pixel accuracy to produce the final measurements of the positions of the atomic columns. This step is crucial for accuracy and will be described in detail below.

4. When the atomic structure is represented as a set of discrete lattice points, the strain is calculated by, comparing the measured positions to an ideal lattice.

Peak refinement

There are two unambiguous ways of measuring the position of an atomic column: from the position of the intensity extrema or from the center of mass. There is no good theoretical argument for choosing the center of mass over the peak or vice versa, and generally the methods perform similarly. The maxima is very sensitive to small irregularities in the shape of the peak, however such

irregularities can also change the integration region which in turn has an influence on the center of mass. We find that the precision of the two methods are similar, as shown in the supplementary information of the paper included in A.1.
Since using the intensity extrema is simpler we use this. Determining the intensity extrema with sub-pixel accuracy is done by fitting a polynomial function, to a region around the detected extrema. We model the intensity of the extrema as a polynomial function [40]

$$I(x,y) = c_0 + c_1 x + c_2 y + c_3 x^2 + c_4 y^2 + c_5 xy \quad , \tag{3.4}$$

where (x, y) is a pixel coordinate and c_i $i = 0, \ldots, 5$ are parameters to be fitted. The peak position is then found by setting the derivatives. Alternatively, the intensity can be fitted as an elliptical Gaussian [41].

Strain calculation

Strain is a concept from continuum physics, and there seems to be no single agreed upon method for calculating strain from a distorted set of lattice points. The electron microscopy literature provides two main different methods for calculating strain using the real space method: by direct comparison to an ideal reference lattice [42, 43], or using Peak Pairs Analysis (PPA) [40].
In the first method, an ideal lattice corresponding to the measured lattice is defined using 6 parameters: two base vectors, $\boldsymbol{a}$, $\boldsymbol{b}$, and the lattice origin $\boldsymbol{x}_{00}$. These parameters are obtained by minimizing the distance between the measured and the reference lattice using the least-squares difference

$$r = \sum_{h \neq 0, k \neq 0} \| \boldsymbol{x}_{hk} - h\boldsymbol{a} - k\boldsymbol{b} - \boldsymbol{x}_{00} \|_2^2 \quad , \tag{3.5}$$

where the sum is over the lattice points in a designated reference region. Each lattice point is indexed using two integers of h and k. The displacement at each lattice point is then obtained as

$$\boldsymbol{u}(h,k) = \boldsymbol{x}_{hk} - h\boldsymbol{a} - k\boldsymbol{b} - \boldsymbol{x}_{00} \quad , \tag{3.6}$$

and the strain may be found as the derivative of this, possibly after transforming the displacements to a cartesian coordinate system. Our main issue with this method is that the difference in the numerical derivatives are calculated with respect to just two adjacent lattice points. It is often preferable, when noise is present, to take additional lattice points into account, in order to reduce statistical errors. In GPA this is accomplished by reducing the radius of the mask when the image is Bragg filtered. Another issue is that this method requires the task of assigning indices to each peak, for which no robust automatic algorithm have been proposed. This is especially problematic when strain have to be calculated for multiple lattices or when defects are present.

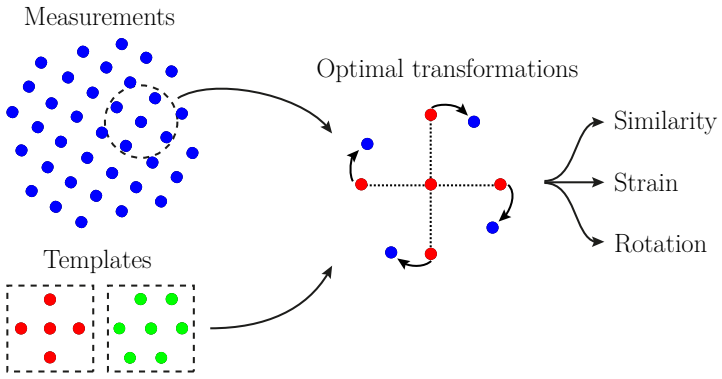


Figure 3.1: The algorithm for calculating the strain and structural similarity from a set of 2d atomic positions: The measured local lattices are compared to a set of one or more templates, using a similarity measure that is invariant to rotation and scale. The most similar template is used as comparison for calculating the strain.

PPA solves the last of these issues to some extent. In PPA the ideal lattice vectors are extracted in a fashion similar to above, the next step is to connect each measured lattice point to two adjacent lattice points by the most probable distorted lattice vectors. The strain tensor at each lattice vector is the one that transforms the two ideal lattice vectors into the distorted ones. The main advantage of this method, is that it is automatic. However, this method still have the issue of using just two of the available lattice vectors.

3.2 Structural template matching

In this section, we describe our proposed method for calculating the strain from a set of measured points. An overview of the method is given in Fig. 3.1. Strain can only be defined in terms of the relative positions of neighbouring points. Hence, the first step is to define the *local lattice* associated with every point. Each separated local lattice is compared to a set of ideal template lattices and a similarity measure is extracted for each. Once the best matching template is identified, the strain is calculated by determining the optimal affine transformation between the template and the measured local lattice. Since the template generally should consist of more than three points, the optimal transformation is found in a least-squares sense. An important additional feature of the method is the ability to categorize the local structure using the similarity measure. This similarity measure can also be used in a machine learning context, to automatically identify and categorize local lattices in a set of points. Peter Mahler Larsen's work was a large inspiration for this method.

3.2.1 Separating the local lattice

Since the local lattice is an inherently ambiguous concept, we should be clear in how we choose to define it. As we are considering a set of discrete points, this comes down to how we separate the points inside the local lattice from those outside. The local lattice will be defined for every measured point, and will consist of the central point itself and its neighbours. One way to choose the neighbours is to include all the points within some radius from the central point. However, this means that the number of points in the local lattice is sensitive to small distortions. Instead we will define the local lattice for some point, as the point itself and its N nearest neighbours. To ensure that the local lattice extents approximately equidistantly in all directions, N should be chosen to fill a neighbour shell. For example, when defining the local structure of graphene N can be $3, 9, 12$, etc.

This definition works well for an ideal or slightly perturbed ideal lattice, however in the presence of defects, the extent from the central point to the N nearest neighbours becomes irregular and sensitive to small distortions. Later, we discuss how we treat defects and we introduce an optional mechanism for locally lowering N.

3.2.2 Similarity

Given two structures described by two sets of N discrete points, $\boldsymbol{V} = \{\boldsymbol{v}_0, \boldsymbol{v}_1, \ldots, \boldsymbol{v}_N\}$ and $\boldsymbol{W} = \{\boldsymbol{w}_0, \boldsymbol{w}_1, \ldots, \boldsymbol{w}_N\}$. In this context, one set is the local lattice and the other set is a template. We assume that the first point in both sets are the central point, for which the local lattice is defined. We also assume that, we have already established a correct point-to-point correspondence between the two sets, i.e. $\boldsymbol{V}$ and $\boldsymbol{W}$ are ordered optimally. The process of finding the optimal point-to-point correspondence is described in the next section. We use the Root-Mean-Square-Deviation as our measure of the similarity between the two sets of points

$$\mathrm{RMSD}(\boldsymbol{V}, \boldsymbol{W}) = \sqrt{\frac{1}{N}\sum_{i=1}^{N} \|(\boldsymbol{v}_i - \boldsymbol{v}_0) - (\boldsymbol{w}_i - \boldsymbol{w}_0)\|_2^2} \quad , \tag{3.7}$$

where the central point of both sets have been made to coincide with the origin. The RMSD can be made invariant to rotation by finding the optimal rotation between the two sets of points

$$\mathrm{RMSD}(\boldsymbol{V}, \boldsymbol{W}) = \min_{\boldsymbol{U}} \sqrt{\frac{1}{N}\sum_{i=1}^{N} \|(\boldsymbol{v}_i - \boldsymbol{v}_0) - \boldsymbol{U}(\boldsymbol{w}_i - \boldsymbol{w}_0)^T\|_2^2} \quad , \tag{3.8}$$

where $\boldsymbol{U}$ is a right-handed orthogonal matrix. Often we also want to make the RMSD invariant to both absolute and relative scale. We can do this by scaling $\boldsymbol{W}$ such that the mean distance of each

point from the origin is 1, and finding the optimal scaling s between the two sets of points

$$\mathrm{RMSD}(\boldsymbol{V},\boldsymbol{W}) == \min_{\boldsymbol{U},s} \sqrt{\frac{1}{N}\sum_{i=1}^{N} \|s(\boldsymbol{v}_i - \boldsymbol{v}_0) - \frac{1}{S(\boldsymbol{W})}\boldsymbol{U}(\boldsymbol{w}_i - \boldsymbol{w}_0)^T\|_2^2} \quad , \tag{3.9}$$

where

$$S(\boldsymbol{W}) = \frac{1}{N}\sum_{i=1}^{N} \|\boldsymbol{w}_i - \boldsymbol{w}_0\| \quad . \tag{3.10}$$

A solution for finding s is given by Horn [44], who also show that $\boldsymbol{U}$ is independent of s. The scale-invariant RMSD serves two purposes; it avoids preferential weighting of smaller templates and avoids the need for selecting bond lengths. Finding $\boldsymbol{U}$ is a well studied problem with many different solution methods. We use the quaternion characteristic polynomial (QCP) method of D.L. Theobald [45]. A function call from Python of our C-implementation of the QCP method requires on the order 0.2 ms on a laptop with an Intel Core I5 processor.

3.2.3 Template matching

The task is to find the permutation of the points in the template that provides the lowest RMSD with respect to a particular local lattice. Checking all possible permutations for a set of N points, requires $N!$ function calls of the QCP method, hence a brute force approach quickly becomes infeasible. We solve this in one of two ways: If the structure is not distorted too much from the ideal, it is often possible to sort the points into a few symmetry-equivalent orderings. If this fails, we fall back on a robust branch and bound search method. The latter requires significantly more function calls, but guarantees the optimal permutation. Once the optimal permutation and corresponding RMSD for each template is found, the best match is chosen as the one that has the lowest RMSD.

Sorting

When calculating similarity with respect to a defined ideal template structure, we can usually take advantage of rotational symmetry. For example, when matching the 4 nearest neighbours for a square lattice (e.g. fcc in the in the $\langle 100 \rangle$ zone axis), all neighbours are equivalent by rotational symmetry. Hence, we can simply order the neighbours by their azimuthal angle around the central point, see Fig. 3.2.
If we also want to include the second nearest neighbours, there are several non-equivalent ways of ordering the points by angle. However, we can order the points uniquely by using lexicographic ordering of the neighbour shell and the angle

$$(s_i, \alpha_i) \geq (s_j, \alpha_j) \quad \forall i < j \quad ,$$

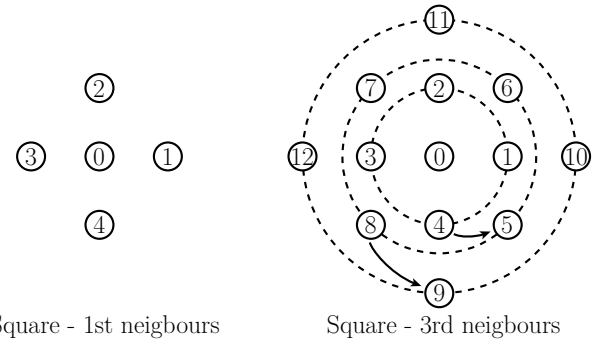


Figure 3.2: It is often possible to sort the template points and measured local lattice uniquely by neighbour shell and azimuthal angle. The ordering of the points in templates for a square lattice is indicated by the numbering. The dashed circles indicates the neighbour shells.

where s is the shell number and α is the azimuthal angle, see Fig. 3.2. The catch is that the sorting of the local lattice should be robust to small distortions. This is the reason, that we use the neighbour shell rather than simply the distance from the central point, where the neighbour shell is defined with respect to the template. In the local lattice the neighbour shell is identified by the ordering with regard to the distance to the central point, i.e. for a cubic lattice, $s = 0$ for the first four closest points and $s = 1$ for the next four closest points etc.

The template constructed from the cubic lattice can be ordered, without any regard for the choice of the first point. This is because the first shell has the same number of points as the rotational symmetry order. If the order of symmetry is lower than the number of points in the first shell, the choice of the first point matters. In our implementation this is solved by calculating the RMSD for all orderings with non-equivalent starting points. We start ordering from the shell with the smallest number of points and wrap around after the outermost shell is reached. The number of templates that have to be tested is then the smallest number of points in a shell divided by the rotational symmetry order.

Sorting the points is not guaranteed to provide the optimal RMSD. If the measured lattice is distorted too much away from the template, a neighbour shell or azimuthal angle can overlap, see Fig. 3.3(a,b). The severity this problem is highly dependent on the particular lattice. An especially difficult example is the $\langle 110 \rangle$ zone axis for an fcc lattice due to the small separation of first and second neighbour shells. The shells can be separated better using the length of the bounding polygons of the Voronoi cell of the central point, see Fig. 3.3(c). Given a collection of 2d points $\boldsymbol{V} = \{\boldsymbol{v}_1, \boldsymbol{v}_2, \ldots, \boldsymbol{v}_N\}$, the Voronoi cell of a point $\boldsymbol{v}_i$ consists of all points which are at least as close to $\boldsymbol{v}_i$ as to any other $\boldsymbol{v}_j$. The boundary of a Voronoi cell can be defined by a set of polygons, each of which defines the interface to an adjacent Voronoi cell.

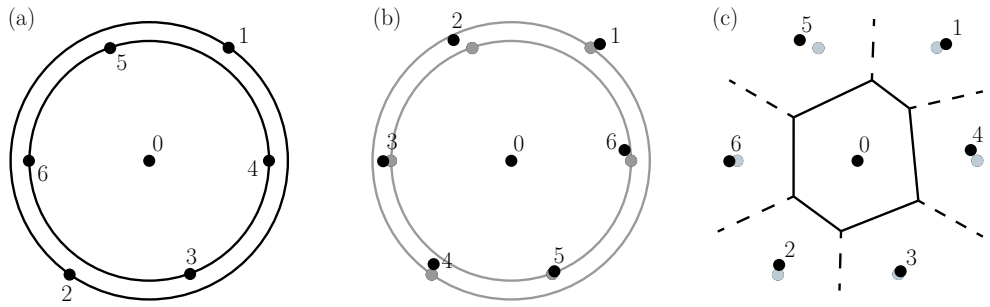


Figure 3.3: **(a)** The ordering of a template for the $\langle 110 \rangle$ zone axis of an fcc lattice. By ordering starting from the outer shell, we obtain a single rotationally invariant ordering. **(b)** Under small perturbations from the ideal structure, the neighbour shells starts to overlap. Hence, the same ordering as the template is not retained. **(c)** Ordering by the interface between Voronoi cells often keeps the neighbour shell separation, better than ordering by Euclidean distance.

Branch and Bound

We use branch and bound (BnB) search for finding the optimal permutation between the template and the measured local lattice. A comparison of other methods is given by Griffiths et. al. in the context of matching 3d crystal structures [46]. We use this method since the number of points to match is small, typically on the order of $4 - 12$, and the two sets of points typically are distorted versions of each other. Additionally, BnB allows us to match partial templates at no additional cost. Our method is similar to that of Hong et. al. [47], it differs somewhat since we use a scale invariant RMSD, whereas they use a translationally invariant RMSD with a fixed scale.
There are two main ideas in BnB. First, it uses branching to tackle a large problem effectively. Instead of solving one huge problem, BnB solves many smaller sub problems. Second, it uses bounding. Unlike naive approaches that explores every feasible solution, BnB attempts to save computation by progressively increasing the lower bound of the optimal solution of a subproblem and compare this bound to a known upper bound. In our case, the upper bound is calculated using the permutation given by the sorting algorithm above. If sorting the points indeed leads to the optimal solution, the BnB search will recover this permutation quickly. The algorithm relies on subproblems always having a lower or equal similarity measure, than a larger version of the problem, hence we use the Root-Sum-Square-Deviation during the search

$$\text{RSSD} = \sqrt{N}\ \text{RMSD} \quad . \tag{3.11}$$

We will exemplify the method by application to a local lattice consisting of four neighbours (and the central point). This is illustrated in Fig. 3.4. Let the local lattice be given by $\boldsymbol{W} = \{\boldsymbol{w}_i\}$ and the template by $\boldsymbol{V} = \{\boldsymbol{v}_i\}$, with $i \in \{1, 2, 3, 4\}$. For this example, the first step is to branch the

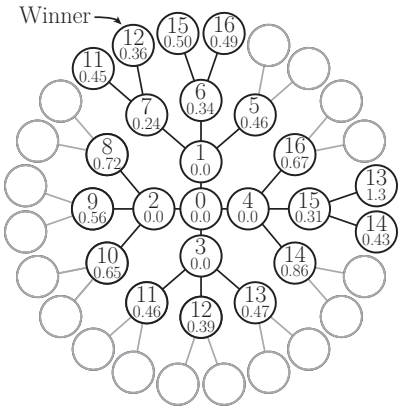


Figure 3.4: The search tree for our BnB method for matching a template to a local lattice consisting of four neighbours. The upper number of each node indicates the order with which the node is visited, and the lower number shows the RSSD value for the permutation the node represents. The outer rim contains $4! = 24$, whereas the BnB algorithm required 18 RSSD evaluations in this case.

zero'th node into four nodes matching the first point in the local lattice to the each point in the template, i.e. matching $\boldsymbol{W}' = \{\boldsymbol{w}_1\}$ to the partial templates $\boldsymbol{V}' = \{\boldsymbol{v}_i\}$. Due to scale invariance the RSSD is zero, and hence the lower bound is still zero. Each of the four new nodes are expanded to match $\boldsymbol{V}' = \{\boldsymbol{v}_1, \boldsymbol{v}_2\}$ to $\boldsymbol{W}' = \{\boldsymbol{w}_i, \boldsymbol{w}_j\}$ where $j \neq i$. This costs 12 evaluations of the RSSD. The smallest RSSD is set as the lower bound and the algorithm traverse down the corresponding node, expanding two new nodes for each. The partial ordering is extended to include an additional point, which also fixes the last point. The algorithm always branches the node with the lowest RSSD, unless it is at the final level and no branching is possible. It is terminated as soon as the lower bound of any node belongs to a node at the final level. Since the RSSD never decreases along any branch, this lower bound is guaranteed to correspond to the best possible match.

Partial matching

Using the BnB algorithm, it is also possible to supply a maximum RMSD value, r_{max}. If this is surpassed, the node cannot be branched, saving a significant number of evaluations. If it is found that r_{max} is surpassed for all nodes, the algorithm can either return no match ($r = \infty$) or a partial match. If partial matches are desired, the returned match will correspond to the node at the highest level for which $r < r_{max}$. If the partial match is at a user defined too low level, no match can be returned instead. It should be stressed that the partial match is not in general the best match of a subset of points in the template and the measured local lattice. The partial match eliminates points starting from the last point, hence the order of the local lattice is significant for partial matches,

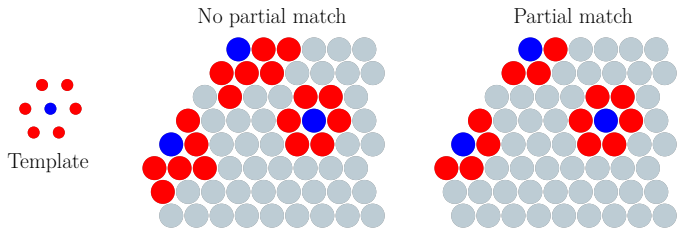


Figure 3.5: Partial matches become relevant for e.g. calculating surface strain, since the N nearest neighbours does not match the template well, and it becomes arbitrary which N points are included in the local lattice. The dots indicate measured lattice points, for which we match the indicated template. The blue dots are the central points, with their neighbours in red. Without partial matches the RMSD for the points at the surface, is huge due to the obvious incompatibility with the template. Using partial matches it is possible to define a smaller template with an equivalent RMSD to the bulk points, and for which a sensible strain calculation can be computed.

we will use euclidean ordering. The use of partial matches is illustrated in Fig. 3.5.

3.2.4 Strain

The strain is easily obtained once the optimal permutation of the template with respect to the measured points is found. In general, we calculate the strain for the partial template, $\boldsymbol{V}'$, and local lattice, $\boldsymbol{W}'$, with the number of matched points N'. First we find the optimal affine transformation, $\boldsymbol{A}$, between the two sets of points using a least squares fit:

$$r = \min_{\boldsymbol{A}} \left\| \sum_{i=0}^{N'} \boldsymbol{w}_i \boldsymbol{A}^T - \boldsymbol{v}_i \right\|_2 \quad , \tag{3.12}$$

where r is the residual term. The minimization problem in Eq. (3.12) is implemented efficiently by expressing the set of points as a matrix. The orientation and elastic strain matrices are obtained via a left-sided polar decomposition of the transformation matrix:

$$\boldsymbol{P}\boldsymbol{U} = \boldsymbol{A} \quad , \tag{3.13}$$

where $\boldsymbol{U}$ is an orthogonal right-handed matrix (the rotation matrix), and $\boldsymbol{P}$ is the symmetric deformation gradient tensor. The choice of a left-sided polar decomposition is arbitrary, but we find the elastic strain in the same frame of reference preferable for comparison of strains across different lattices. In the case where $\boldsymbol{P}$ is not the identity matrix, $\boldsymbol{U}$ is not the same rotation found by minimizing the RMSD, since the addition of strain means, we no longer have a rigid-body transformation. The residual term in Eq. (3.12) could be used to determine the local structure

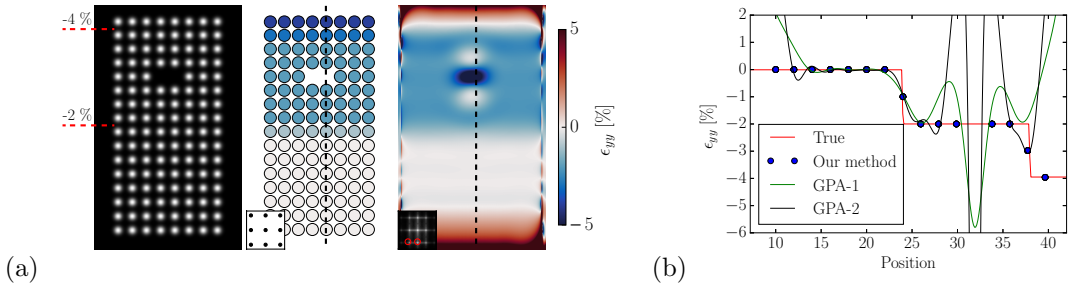


Figure 3.6: **(a)** An array of Gaussians is used for calculating strain using our method and GPA. The centers of the Gaussians are displaced to obtain regions with compressive strain. Given this We show the strain caluclated using our method and using GPA. In the lower left corner, we show the template used for calculating strain with our method, and indicate reciprocal lattice vectors used for GPA. **(b)** The strain along a vertical slice through the strain maps in (a). We show the strain calculated using GPA with both a small mask (GPA-1) and large mask (GPA-2), used during Bragg filtering.

instead of the RMSD, however, the elastic strains are typically on the order of 5 %, and the extra degrees of freedom provided by the strain matrix often results in highly-strained erroneous structural identifications. Setting the scale of the template was not required during template matching, however calculating the strain tensor obviously requires us to set a scale. The scale dependent strain tensor can be calculated as

$$\boldsymbol{\epsilon} = \begin{pmatrix} \epsilon_{xx} & \epsilon_{xy} \\ \epsilon_{yx} & \epsilon_{yy} \end{pmatrix} = \begin{pmatrix} \frac{S(\boldsymbol{V})P_{0,0}}{s} - 1 & P_{0,1} \\ P_{1,0} & \frac{S(\boldsymbol{V})P_{1,1}}{s} - 1 \end{pmatrix} \quad , \tag{3.14}$$

where s is the mean distance from the origin to the points in the *full* template, accounting for pixel size. We note that this is the only point where setting the scale is required. Contrary to GPA or PPA we do not require a patch of undeformed lattice as reference. However, it is possible to set the scale by requiring the mean planar strain to be zero in a specified region.

3.2.5 Comparison to GPA

GPA is simpler to apply, than our method, both computationally and from a user standpoint. The main difficulty of using our technique is, as with all real space techniques, related to detecting the lattice points. On the other hand, GPA can be used even when only a single lattice fringe is visible. However, the ease of applying GPA have resulted in some situations, where we do not think it is appropriate, such as for surface strain [48] and under certain circumstances for nanoparticles [49]. Fig. 3.6 shows a simple test comparing our technique to GPA. The test case is cubic lattice of Gaussians, with step function changes in the strain, both in the bulk and at the surface, additionally

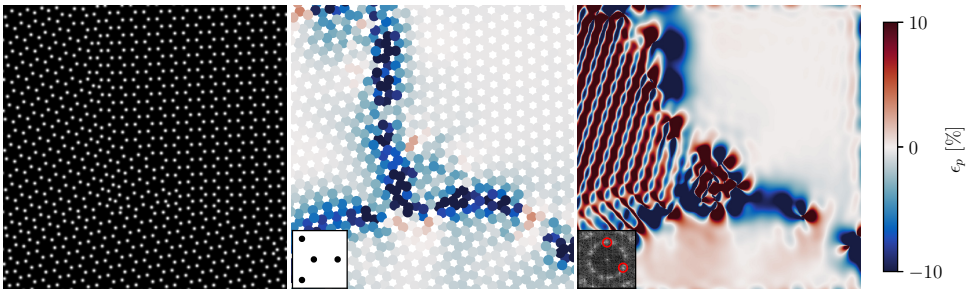


Figure 3.7: **(a)** Image consiting of a sum of Gaussians placed at the atomic positions from a simulation of polycrystalline graphene. Given this image we calculated the the strain using **(b)** our method and **(c)** GPA. In the lower left corner, we show the template used for calculating strain with our method, and indicate reciprocal lattice vectors used for GPA.

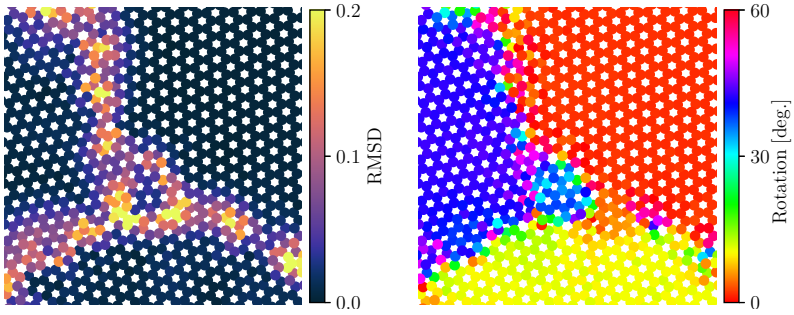


Figure 3.8: The RMSD and rotation determined using our method given the image in Fig. 3.7(a).

a vacancy is included. Analysing the strain, we see that GPA agrees perfectly with with our method in the bulk, however close to the defect and the surfaces, the results differs significantly. We use partial matching, hence spatial averaging is directional in the vicinity defects and surfaces. The spatial averaging is constant in GPA, hence the results becomes non-sensical across defects. The strain calculated using GPA is frequently used to highlight defects [50], which is a valid use, however care should be taken when interpreting that strain.

With our technique it is possible to separate the strain in the near-periodic lattice from the defects. It is still possible to highlight defects using either the RMSD given the full template or the number of points in the partial template N'. This gives our method an increased amount of freedom with respect to how surfaces and defects are treated in the analysis.

Another case where our method shines, is when multiple different sublattices are present. Fig. 3.7 shows the application of our technique to a lattice consisting of three graphene grains, the positions of the lattice points was obtained as described in section 6.4.1. Since there are three

differently oriented sublattices, it is not possible to pick a set of two reciprocal lattice vectors, that will be optimal for all grains. This severely affects the performance of GPA. Our technique, due to rotational invariance is capable of calculating the strain equally well in each grain, without requiring any additional user input. It is also possible to obtain the rotation of each grain as shown in Fig. 3.8.

Lastly, we compare the performance of our method to GPA in the presence of noise, see Fig. 3.9. An image of a cubic lattice of Gaussians with an amplitude of one was created, the centers of which was displaced according, to the strain given by the test functions

$$\epsilon_{xx} = (x+y)\exp(a(-x^2-y^2)) \quad \text{and} \quad \epsilon_{yy} = (x-y)\exp(a(-x^2-y^2)) \quad . \tag{3.15}$$

Gaussian noise with a variance of 0.05 was added independently to each pixel. When our method was applied the noise was removed with a Wiener filter and the positions of the peaks were refined according to Eq. (3.4). From the results shown in Fig. 3.9, we conclude that our method is neither more nor less stable to noise than GPA.

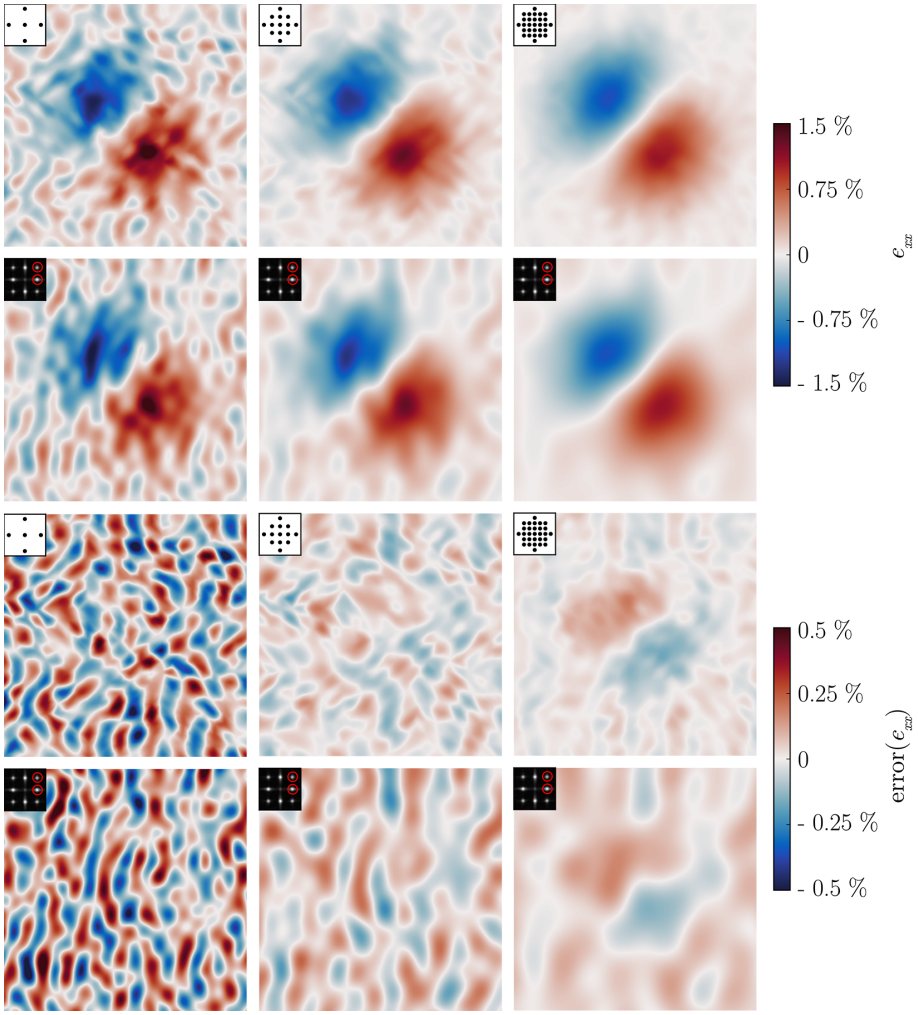


Figure 3.9: The first two rows of images shows the strain calculated using our method and using GPA. We interpolated the strain calculated with our method to a uniform grid with the same resolution as the input image for easier visual comparison to GPA. The two last rows shows a comparison of the strain error, i.e. the difference between the true strain and the measured strain. We show results for increasing spatial averaging: For our method this means including more points in the template, whereas in GPA this means decreasing the size of the Bragg filter mask. In the lower left corner, we show the template used for calculating strain with our method, and indicate reciprocal lattice vectors used for GPA.

Chapter 4

Accuracy of Strain Measurements in Nanoparticles from HRTEM Images

As described in the introduction of the previous chapter, strain on the order of a few percent can improve the catalytic performance of nanoparticles dramatically. In this chapter, we analyze the precision and accuracy of strain measurements from simulated HRTEM images of nanoparticles. HRTEM images relies on phase-contrast and is extremely sensitive to aberrations of the objective lens. As a consequence, considerable care have to be shown when extracting quantitative information from an image. We consider to what extent it is possible to measure surface relaxations, taking into account the influence defocus, nanoparticle size, tilt and noise.

4.1 Related work

A first investigation to determine the accuracy with which surface strain could be determined was undertaken by Marks [51]. Image simulations were used to compare actual relaxations, in the input atomic models, with the apparent relaxations, measured from the corresponding simulated images. They found that there was a linear relationship, between apparent and real strain, with a constant outward shift of about 5 %. They also demonstrated that the true positions of the atomic columns at the surface could be determined to within 0.2 Å, corresponding to 5 % of the lattice parameter of gold. This investigation was done before the invention of hardware aberration correction.
Newer investigations have primarily focused on the accuracy of strain measurements in the vicinity of interfaces in heterostructures [52, 53, 54, 55]. The estimated error has been found to be as low as 0.1 % [56, 57]. In these studies the specimen thickness was uniform and the lattice was (nearly) periodic except across the interface. The situation is different for nanoparticles, where thickness variations and surfaces break the local symmetry. Moreover, in these studies the strain distributions were fundamentally 2D, i.e. the atomic columns were displaced in the plane perpendicular to the zone axis. This is different from images of nanoparticles where the true 3D strain is projected on

a 2D plane.
There is no shortage of experimental studies using HRTEM to measure strain in nanoparticles or in the vicinity of surfaces. These measurements are often backed by comparison with a simulation approximating the experimental structure and microscope conditions [58, 59, 60, 61, 62, 63]. The general conclusion seems to be that the measurement error is small enough, that meaningful conclusions regarding surface strain can be based on HRTEM images. However, the studies generally lack a systematic analysis of the sensitivity to experimental variables.

4.2 Methods

4.2.1 Atomic models

Our main investigation considers gold nanoparticles. The overall shape of the model clusters were determined from a Wulff construction, see Fig. 4.1. Finite temperature was modelled in the frozen phonon approximation [13]. The snapshots were random steps from a constant temperature MD simulation using Langevin dynamics at 300 K [64]. We only use steps after the initial equilibration and the simulation was run for long enough to properly represent the thermal distribution of the atomic positions.
The difference between the thermally averaged atomic positions and relaxed positions is mainly a thermal expansion. Hence, simulating temperature in this way, the atomic models are in effect relaxed. We used a slightly reparametrized version of the COMB potential for gold presented in section 5.3. The potential was slightly adjusted to predict surface relaxations and other structural properties of gold, at the cost of some fidelity for e.g. surface energies. Experimental studies have demonstrated that the surface layer of many clean transition metals relaxes inward [65], while expansion of the top layer have been found for some surfaces of noble metals [66], including the {111} facets of gold. For an infinitely extended {111} surface, the COMB potential predicts a 1.2 % surface expansion of the top layer, which is close to the experimental value of 1.3 % [67]. For the {100} surface an inward relaxation of 1.1 % is predicted. It was not possible to find a corresponding experimental value, however the prediction is close to 1.2 % [68] and 1.51 % [69] calculated with DFT.

4.2.2 Image simulation

The microscope conditions were modelled after an aberration corrected FEI Titan microscope operated at 300 kV. Unless otherwise stated the third order spherical aberration were set to $C_s = -10$ µm and all other aberrations except for defocus was set to zero. We have tested the stability of our results to inclusion of additional aberrations, in particular two-fold astigmatism on the order of 5-10 nm and 5th order spherical aberrations on the order of 2.5 mm, and found no significant

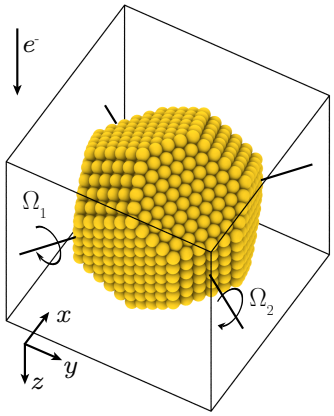


Figure 4.1: Atomic model of a gold nanoparticle. The two rotation axes Ω_1 and Ω_2 are indicated.

impact on the conclusions of this chapter. The focal spread was $\Delta = 2.9$ nm and the convergence angle was set to 15 mrad. We used the parametrization of the MTF given in section 2.21, and noise were simulated as a Poisson process. The sampling used for the simulations were at least 0.05 Å/pixel, and the simulated images were downsampled to experimental resolution using bilinear interpolation.

4.2.3 Strain measurements

We benchmark the accuracy of strain measurements by comparing the strain calculated directly from input atomic models to the strain measured from simulated HRTEM images. We calculate the strain using structural template matching, with the nearest and the second nearest neighbours, hence the template is equivalent to that shown in Fig. 3.3. Generally, the positions of the atomic columns are determined from the corresponding intensity extrema, however we show that the center of mass provides similar precision and accuracy. We will usually just show the planar strain, ϵ_p, calculated as the average of the normal strains in the x- and y-direction

$$\epsilon_p = \frac{1}{2}(\epsilon_{xx} + \epsilon_{yy})\,. \tag{4.1}$$

We define surface relaxations as the strain at the outermost atomic positions in the direction perpendicular to the same surface. Hence, the surface relaxation associated with an atom on a surface perpendicular to the unit vector $\hat{\boldsymbol{n}}$ is found as

$$\epsilon_{\hat{\boldsymbol{n}}} = \hat{\boldsymbol{n}}^T \boldsymbol{\epsilon} \hat{\boldsymbol{n}}\,. \tag{4.2}$$

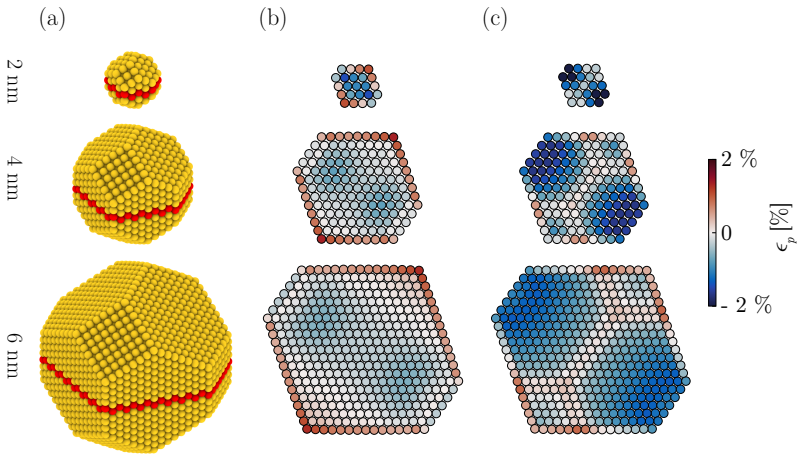


Figure 4.2: **(a)** Three differently sized nanoparticles, with the approximate diameter is given. **(b)** The distribution of strain given the atomic positions from the slices indicated with red in (a). **(c)** The distribution of strain given the average projected positions.

4.3 What is measured?

Given a single image of a crystal in a low index zone-axis, we can only measure the average position of the atomic columns. The actual strain in nanoparticles is 3D, projected onto a 2D image, this have some significant consequences, as illustrated in Fig. 4.2.

The *true* average column position, or projected position, of each column is calculated as the average of the (x, y)-coordinate of all the atoms belonging to that column. The projected positions exhibit a significant compressive strain in the bulk of the particle. This strain disappears when we look at a subset of the atoms in the center of the nanopartiple. Hence, the apparent compressive strain in the projected positions, is actually due to relaxations close to the front and back surface of the nanoparticle. Another noteworthy difference is the variation of the surface strain in the projected positions due to thickness differences along the optical axis. As the size of the particle increase the discrepancy between the two strains diminishes, since the relative importance of surfaces and edges is smaller.

In contrast to diffraction-based techniques, an HRTEM image does not measure the average strain: The average is over positional coordinates, and not strain. In principle this means that the measured quantity, given a single view is somewhat ambiguous, however in practice this strain is fairly well correlated with the true average 3D strain. Even if the strain is not the true average, it can be used as a measure to indicate how environmental factors influence the nanoparticle.

With the above reservation in mind. The next issue is the accuracy with which the strain given the true projected positions can be measured, and what errors are introduced by the imaging. We will

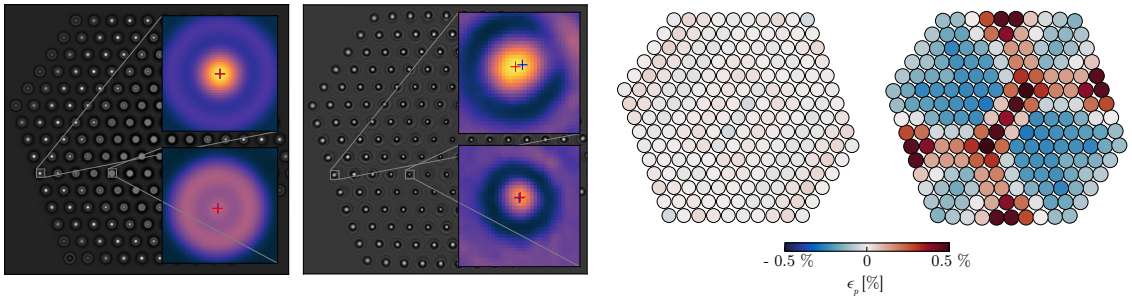


Figure 4.3: **(a)** The intensity of the exit wave function given a 4 nm relaxed gold cluster. The average column position and the measured intensity extrema is indicated by the and the red dots respectively. **(b)** The planar strain error given the positions of the intensity extrema in (a).

define the error

$$\text{error}(\epsilon) = \epsilon_{measured} - \epsilon_{true} \quad , \tag{4.3}$$

where $\epsilon_{measured}$ is the measured strain given a HRTEM image and ϵ_{true} is the strain given the true projected positions.

Fig. 4.3(a,b) shows the exit wave function, given an ideal and a relaxed model of a 4 nm gold cluster. The zoomed insets also indicate the projected position of an atomic column and the corresponding measured intensity extrema. For the ideal exit wave function the measured position corresponds exactly (within numerical errors) to the true projected position. On the other hand, a discrepancy is clearly visible for the corresponding relaxed model. This discrepancy corresponds to a difference in the strain of up to half a percent, and the error approximately follows the thickness gradient of the nanoparticle. Hence, even a measurement without any optical aberrations, will have some ambiguity due to the 3D nature of the strain. Moreover, it is common to use a model of an ideal nanoparticle to estimate errors in the strain measured from HRTEM images [58, 59, 60, 61]. The above result shows that this practice can be problematic.

4.3.1 Origin of imaging errors

The above errors are only due to the 3D displacement of the atoms, the thickness variation is just the secondary cause. This can be shown on the basis of the phenomenon of channelling [70, 71]. For gold and a specimen thickness of < 10 nm, each column can to a very good approximation be thought of as independent. The physical reason is that the strong positive electrostatic potential of the atomic cores acts as a guide or channel within which electrons can scatter without leaving the column. The only essential assumption for channelling is a certain spatial separation of adjacent

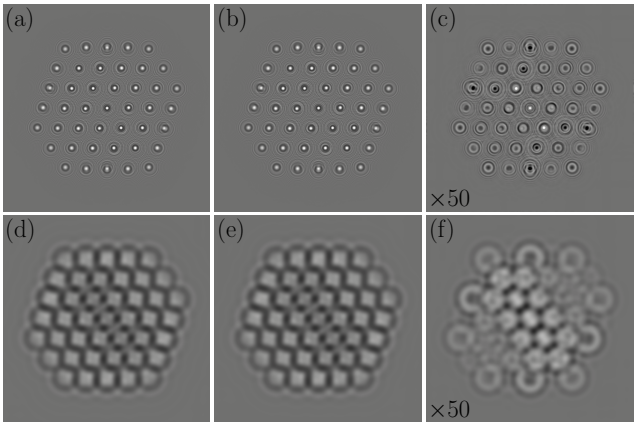


Figure 4.4: **(a)** The intensity of the exit wave function given the full specimen potential. **(b)** The intensity of the exit wave function obtained by summing the scattered wave due to each column independently. **(c)** 50 times the difference between the images in (a) and (b). **(d,e,f)** Same as (a,b,c), however the scattered waves due to each column are summed after applying imaging aberrations.

atomic columns, a separation that will typically be reached for propagation along a high symmetry zone axis. Channelling also persists for small tilts or in the the vicinity of surfaces [72].
As a demonstration Fig. 4.4 shows two different simulations of a gold cluster: One is simulated normally using the full specimen potential. The other is produced by summing the scattered wave due to each column independently. To make the situations comparable, we define the scattered wave function for each column

$$\psi_s = \psi - \psi_0 \quad , \tag{4.4}$$

where ψ_0 is the initial plane wave and ψ is the exit wave function. The following approximation then holds

$$\psi_{full} \approx \psi_0 + \sum_i \psi_{s,i} \quad , \tag{4.5}$$

where ψ_{full} is the exit wave simulated using the full potential, $\psi_{s,i}$ is the scattered wave for an identical simulation where only the i'th column is kept and the sum is over all atomic columns. The approximation also holds when summing the wave function at the image plane, where the overlap between adjacent columns is significant.
Fundamentally, the strain measured from the detected image, can be expected have additional errors, if the scattered wave due to adjacent atomic columns begins overlapping. Hence, it is

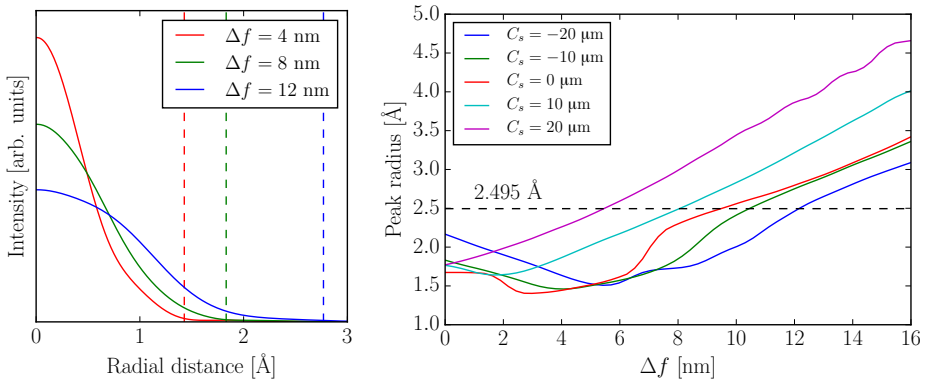


Figure 4.5: **(a)** The profile of the intensity distribution due to an individual atomic column with a height of 8 atoms a and a separation between the atoms along the direction of propagation, equivalent to a view along the ⟨110⟩ zone axis. **(b)** The width of the intensity profile due to an individual atom as a function of defocus.

interesting to look at how imaging parameters influence the delocalization of the image of an atomic column. To quantify this we (arbitrarily) define the radius of influence, as the radial distance where the amplitude of the intensity profile is equal to 1 % of its maximum value, see Fig. 4.5(a).
In Fig. 4.5(b) we show this radius as a function of defocus for five different C_s-values. At $C_s = -10$ µm graph showing the radius is relatively flat area from $\Delta f = 0$ to $\Delta f = 8$ nm, after this point the radius increase approximately linearly. The curves are rather sensitive to changes in C_s

4.4 Accuracy and precision of strain measurements

4.4.1 Influence of defocus

The top row of Fig. 4.6 shows simulated images at different defocus and the bottom row show the error in the planar strain measured from these images. The smallest defocus shown is 4.5 nm since contrast inversion begins to take effect for a smaller defocus. We present results for only a positive defocus, hence the positions of the atomic columns corresponds to the intensity maxima.
For Δf up to 8.5 nm the strain errors does not increase much beyond those of the exit wave function, i.e. the error is on the order of 0.5-1 %. This is expected since the scattered wave due to adjacent columns only overlap very little at this defocus (see Fig. 4.5).
As the defocus increases to $\Delta f = 12.5$ nm the intensity distributions of adjacent columns starts to overlap. This affects the error at the surface in particular, due to the asymmetric environment of surface columns. Due to the sign and location of these errors, they could easily be mistaken for

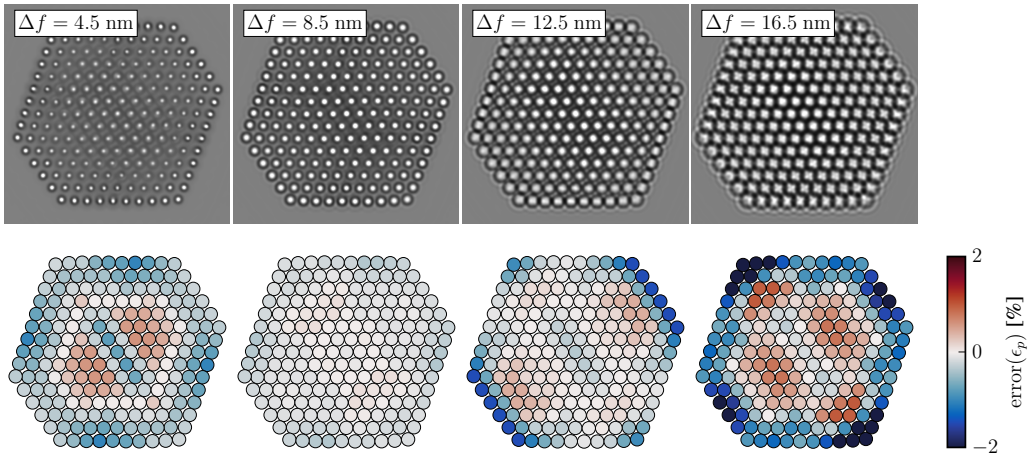


Figure 4.6: The top row shows simulated images for a nanoparticle with a diameter of 4 nm. The bottom row shows the corresponding distribution of errors in the planar strain. The defocus is different in each column, as indicated in the figure.

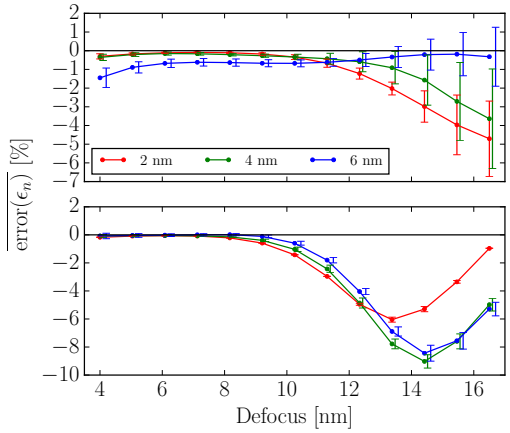


Figure 4.7: The error in the measured surface relaxations averaged across the facets as a function of defocus, for the three particle sizes given in the legend, for: (a) the {111} facets and (b) the {100} facets. The bars indicate the standard deviation of the errors in the surface relaxation error across bthe facets. The bars are shifted slightly from the points for visual clarity.

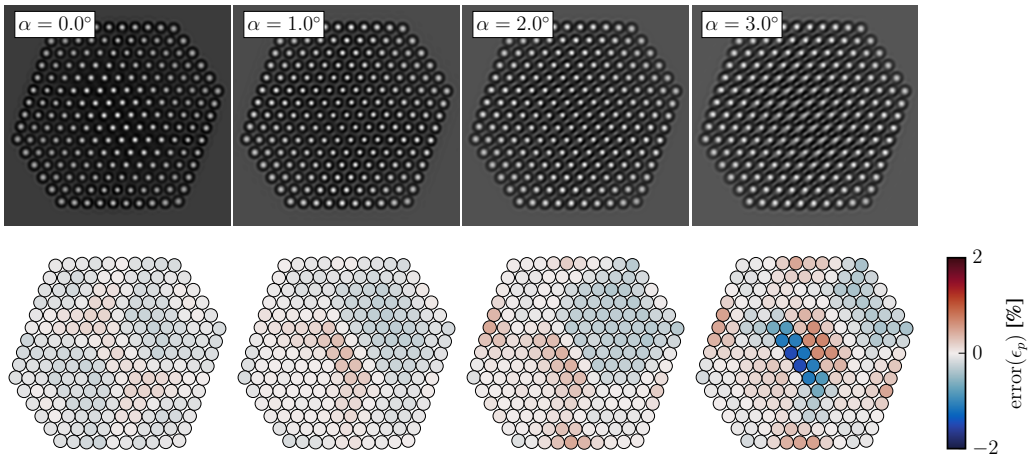


Figure 4.8: The top row shows simulated images at increasing tilt around the Ω_1-axis for a defocus $\Delta f = 8.5$ nm. The bottom row show the error in the planar strain at each lattice point measured from these images.

real surface relaxations.

The error in the measured surface relaxations averaged across the facets for the uppermost atomic layers is shown as a function of defocus in Fig. 4.7. Since this error can vary quite a bit across the {111} facets, we also show the corresponding standard deviation. For the {100} facet the error is small up to a defocus of 8.5 nm. For both facets and all sizes the errors stay below 2 % up to a defocus of ~ 11 nm, where the mean error increases sharply at the {100} facets. The mean error does not increase as drastically for the {111} facets. On the other hand the standard deviation does increase, which is due to the thickness variation along these facets.

4.4.2 Influence of tilt

It is unavoidable that the specimen will be tilted relative to the ideal high symmetry zone axis. Fig. 4.8 show the distribution of errors in the planar strain for increasing tilt, α, around the Ω_1-axis. For tilt $\alpha = 1.0°$, the errors have changed very little compared to the untilted crystal. The appearance of the image have changed in the central part of particle, this is due to an effective diminishing of the projected potential, as have been reported elsewhere [72]. The errors stay small up to a tilt $\alpha = 2.0°$, but increase sharply in the center of the nanoparticle between $\alpha = 2.0°$ and $\alpha = 3.0°$. The error introduced by tilt is very dependent on the height of the atomic columns, since the length of the footprint of the projection of a tilted column increases linearly with its height.

Fig. 4.9 show the effects of tilt on the errors in the measured surface relaxations for a defocus

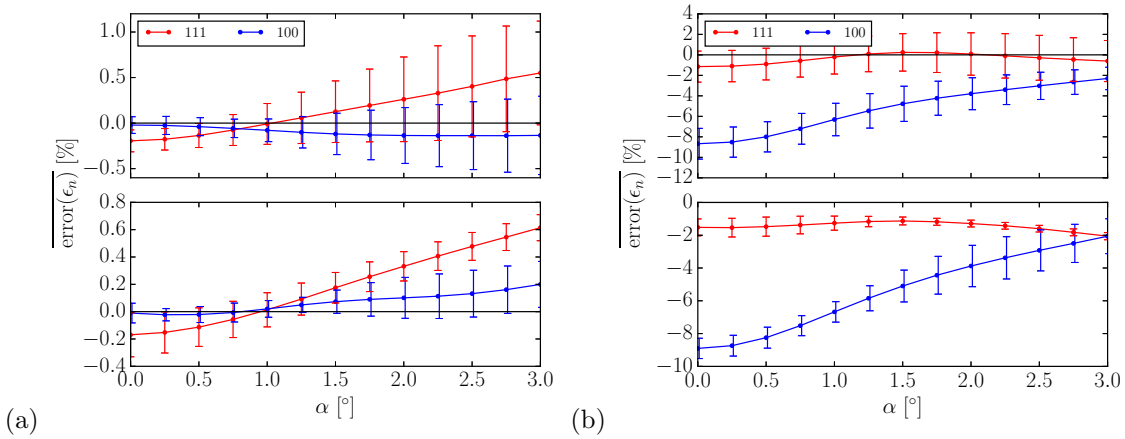


Figure 4.9: The error in the measured surface relaxations averaged across the facets as a function of tilt, around the axes (a) Ω_1 and (b) Ω_2. The defocus was $\Delta f = 8.5$ nm. The curves are for the {100} and {111} facets, as indicated by the legend.

$\Delta f = 8.5$ nm. The tilt has a somewhat limited impact on the measured surface relaxations. The mean and standard deviation of error changes by at most 1 % over the entire tilt range. The effects of tilt on the strain measurements are also very dependent on defocus. For example at a defocus $\Delta f = 14.5$ nm, the mean surface relaxation error changes by more than 6 % at the {100} facets.

4.4.3 Influence of noise

The evolution of the nanoparticle visibility with respect to the sampling and dose is shown in Fig. 4.10. At a dose of 10^2 $e^-/\text{Å}^2$ the nanoparticle is barely visible, while the images are essentially unaffected by noise at 10^5 $e^-/\text{Å}^2$.

To determine the influence of dose on the errors in strain measurements, we simulate a statistically representative ensemble of images, $K = 300$, with different distributions of noise. The error due to noise is quantified using the mean absolute error, MAE, over the ensemble of images for each lattice point

$$\text{MAE}_i = \frac{1}{K}\sum_{k=0}^{K} |\tilde{\epsilon}_{k,i} - \epsilon_i| \ , \tag{4.6}$$

where $\tilde{\epsilon}_{k,i}$ is the planar strain at the i'th lattice point measured from the k'th noisy image and ϵ_i is the corresponding measured strain without noise. Since the automatic polynomial fitting can fail at low doses, extreme outliers have been removed before taking the average. Fig. 4.11(a) shows the distribution of the MAE across a nanoparticle, there is a fairly large difference between the MAE for different lattice points, varying by a factor of three between the center of the particle

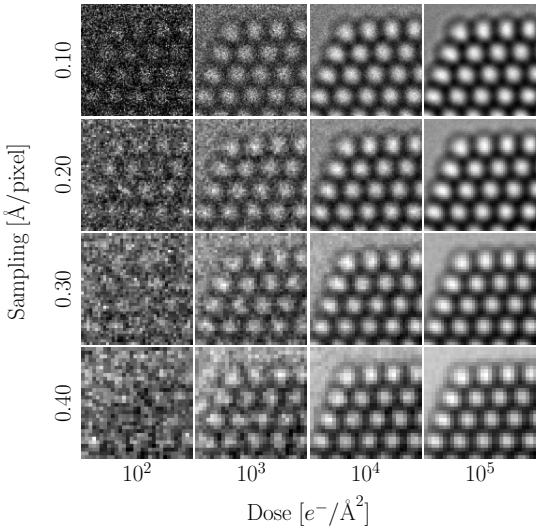


Figure 4.10: (a) Sections of the simulated images at a defocus of $\Delta f = 8.5$ nm for different doses and samplings. All images are mapped onto the same range of gray levels. (b) The same images after applying a Wiener filter.

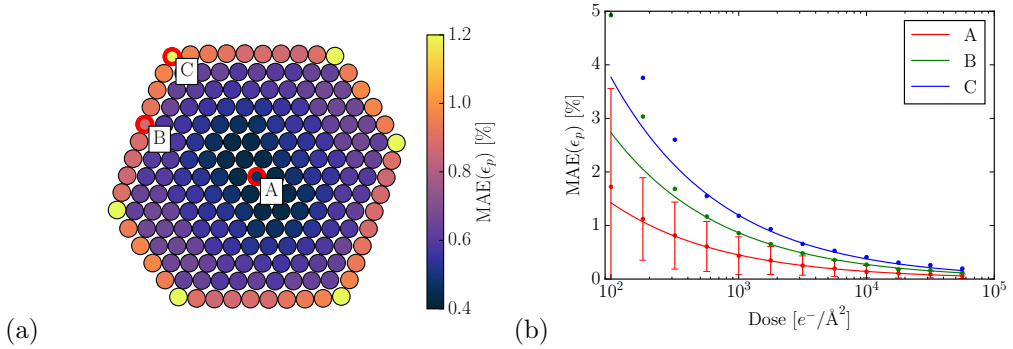


Figure 4.11: (a) The MAE of the planar strain due to noise at each lattice point for a Wiener filtered noisy image at a sampling of 0.2 Å/pixel, a dose of 10^3 e^-/Å^2 and a defocus $\Delta f = 8.5$ nm. (b) The MAE as a function of the dose for the three lattice points, A, B and C, as indicated in Fig. 4.11. The dots show the MAE calculated from the simulated images and the full lines are curves of the form given by Eq. (4.7), where the constant of proportionality has been fitted to the dots. The bars indicate the standard deviations, which for visual clarity are shown only for lattice point B, proportionally the standard deviations are similar for the other lattice points.

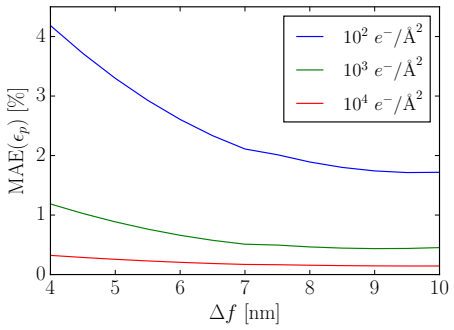


Figure 4.12: The MAE as a function of defocus for three doses at a sampling of 0.2 Å/pixel at the lattice point B.

and a corner. The reason for this is mainly that the strain at surfaces is determined on the basis of fewer surrounding lattice points. The strain at corner atoms is determined on the basis of just three neighbours, while the measurements in the center relies on twice that number of neighbours. The MAE at three chosen lattice sites as a function of dose is shown in Fig. 4.11. We find a simple approximate empirical relationship, assuming constant sampling, between the MAE and the dose

$$\text{MAE} \propto \frac{1}{\sqrt{D}} \propto \frac{1}{\text{SNR}}\,, \tag{4.7}$$

where the constant of proportionality is determined by the number of neighbours, local image contrast and sampling. The second approximate proportionality assumes low dose and is due to Eq. (2.24).

In the previous sections, we saw that the defocus should be kept small to obtain strain measurements that are relatively unaffected by aberrations. The disadvantage of this is that phase contrast imaging relies on the additional phase added by the objective lens, and hence a too small defocus will negatively impact the image contrast. This is quantified in Fig. 4.12. At a low dose the errors grow very large when the defocus is small, but even at a higher dose, errors due to noise increases, when the defocus is too small. When the defocus is increased the MAE becomes smaller, however saturation is reached relatively quickly, additional defocus beyond $\Delta f = 8$ nm does not further improve the MAE.

4.5 Summary

We looked at the accuracy of surface strain measurements from abberation corrected HRTEM images of nanoparticles.

It was shown that given the exact exit wave function, a strain error of $0.5-1$ % should be expected

due to the 3D nature of the strain. We also pointed out that even if this error was not present, some care have to be exercised when interpreting the measured 2D strain distribution. We also showed that the practice of using simulations based on ideal sample models to calibrate strain measurements is problematic, since the predicted errors from such simulations, does not in general reflect the errors for an identical model that includes relaxations.

In general the impact of the interaction between tilt, thickness, defocus and other aberrations on the final strain measurement is both complicated and sensitive to small changes. However, we see that if the defocus can be kept in a certain C_s-dependent range, the overlap between the scattered wave of adjacent atomic columns is negligible, and the additional error due to post-sample optics are small. Additionally, it is found that the error due to tilts away from the ideal $\{110\}$ zone axis are small up to the point where the tilt becomes visible.

When all the errors are added it is obvious that the total error is significantly larger than the $0.1 - 0.5$ % reported for strain measurements from near periodic images [56]. For a dose of 10^3 $e^-/\text{Å}^2$, the optimal defocus for the gold nanoparticles is somewhere around 8.5 nm (given $C_s = -10$ µm), at this defocus the precision and accuracy are 1 % or more. This error is significant compared to the 1-2 % true surface relaxations in gold.

Chapter 5

Interatomic Potentials for Modelling Oxide-Supported Metal Nanoparticles

Details of the interactions between metal nanoparticles and substrates are poorly understood. However, it is known that this interaction is important for the stability of the resulting material. A strong interaction can pin the nanoparticles effectively in their position, preventing particle mobility and subsequent growth due to sintering. Thus, characterizing the local atomic structure at or adjacent to the interface becomes of vital importance as it controls the resulting properties. Although massive investigations of supported nanoparticles have been performed using HRTEM, an accurate and quantitative description on the local atomic structure at the interface is difficult to elucidate. A single edge on image of the interface can establish the orientation relationship, but hardly provides the full story. Simulation of atomic structures can provide a complementary route to understanding the interfaces.
The systems we investigate are Pt on ZrO_2, Pt on TiO_2 and Au on TiO_2. These were originally chosen because it was deemed realistic to investigate them both experimentally and theoretically. However, since the experimental part ended up going in different directions, this part of my thesis work, ended being significantly reduced compared to the original intention. More work is currently needed to bring the contents of this chapter to a publishable state.

5.1 Modelling large-scale supported nanoparticles

There are a range of methods available for theoretical modelling of oxide-supported nanoparticles. All of these rely on mapping from the set of atomic coordinates to the total energy of the system, a relationship typically termed the potential energy surface (PES). First principles methods provide an accurate and general framework for calculating the energy of a system of atoms and density functional theory (DFT) in particular have had huge success, due to its reasonable computational cost for many real systems. However, the practical use of DFT is limited to periodic or small non-

periodic systems. For this reason, theoretical work has typically focused on gas-phase clusters, or on individual ideal surface facets and coherent interfaces [73, 74, 75]. Research on supported clusters have generally focused on sub-nanometric nanoparticles, consisting of on the order 10 metal atoms [76, 77]. Recently, investigations of larger nanoparticles have been conducted with DFT [78, 79], however these systems are still smaller and structurally simpler than experimental model systems. A less expensive computational approach is mandatory, when the supported nanoparticle consists of over a few hundred atoms, since the cubic scaling of standard DFT, with the number of particles, makes even single geometry optimizations prohibitively expensive. The use of interatomic potentials provides a much faster method for evaluating the PES, that scales linearly with system size. The downside is the loss of accuracy, particularly for the complicated bonding environments present in mixed systems, such as metal/oxide interfaces.

5.1.1 Interatomic potentials for metal/oxide systems

Interatomic potentials refers to mathematical models, describing the interaction between the particles of a given system. Given a collection of N particles, with the set of positions $\{\boldsymbol{r}\}$, the prediction for the total energy, is written as an expansion of functional terms that depend on the position of one, two, three, etc. atoms at a time

$$V = \sum_i V_1(\boldsymbol{r}_i) + \sum_{i<j} V_2(\boldsymbol{r}_i, \boldsymbol{r}_j) + \sum_{i<j} V_2(\boldsymbol{r}_i, \boldsymbol{r}_j, \boldsymbol{r}_k) + \dots \quad , \tag{5.1}$$

where V_1 is the one-body term, V_2 is the two-body term and V_3 is the three body term.
If there is no external potential, the total energy only depends on the relative positions of the atoms

$$V = \sum_{i,j}^{N} V_2(r_{ij}) + \sum_{i,j,k}^{N} V_3(r_{ij}, r_{ik}, \theta_{ijk}) + \dots \quad , \tag{5.2}$$

where r_{ij} is the distance between particle i and j (the bond length), and θ_{ijk} is the angle between the vectors from the position of particle i to the positions of particle j and k (the bond angle). Simple pair potentials provides a reasonable description of oxides [64]. For example the Coulomb-Buckingham potential, where the energy for each pair of particles is written

$$V(r_{ij}) = c_0 \exp\left(-c_1 r_{ij}\right) - \frac{c_2}{r_{ij}^6} + \frac{q_i q_j}{4\pi\varepsilon_0 r_{ij}} \quad , \tag{5.3}$$

where c_k, $k = 0, 1, 2$ are fitted parameters, q_i and q_j are the charge of the i'th and j'th atom and ϵ_0 is the vacuum permitivity.
However, pair potentials have some significant limitations, such as the inherent inability to describe all 3 elastic constants of cubic metals. Describing metals requires the inclusion of many-body effects,

such as in the Embedded Atom Model (EAM)

$$V = \sum_{i}^{N} F_i \left(\sum_{j} \rho(r_{ij}) \right) + \frac{1}{2} \sum_{i,j}^{N} V_2(r_{ij}) \quad , \tag{5.4}$$

where F_i is a so-called embedding function, which is a function of the sum contributions to the electron density of neighbouring atoms. EAM is motivated from density functional theory as the energy needed to "embed" an atom into the electron density and can not in general be applied to covalent insulators or ionic materials, even though extended EAM potentials have been suggested. While many well-established potentials exists for oxides and metals individually, much fewer have been created for mixed systems. This is a natural consequence of the combinatorics, but also due to the challenge of creating accurate models. In general, the potential have to capture multiple-bonding events, both covalent, metallic and ionic and correctly describe their relative strength.
In our review of the litterature, we found three well-developed frameworks for simulating metallic nanoparticles on oxides using interatomic potentials.
The Vervisch–Mottet–Goniakowski (VMG) potential [80, 81, 82, 83] and similar approaches [84] have been applied mainly to non-reactive metal/oxide interfaces, in particular the metal/MgO system. Due to the rigidity and simplicity of the compact crystal structure of MgO, a description of the oxide is neglected, and instead they just include the intermetallic interactions and the interactions between metal and the oxide. Many-body effects are included as a smooth function the coordination number with respect to the atoms in the metallic phase. The parameters of the potential are fitted to small DFT calculations of metal/MgO systems. This approach relies on the weakness of the interaction at the interface of non-reactive metal/oxide interfaces [85] and the compactness the MgO substrate.
In particular the metal/TiO_2 is known to be reactive [86, 87, 88], and interfacial charge redistribution have been found to be a significant contributor to interfacial bonding [75]. Additionally, TiO_2 is not close-packed, and the assumption that the there is no reciprocal influence of the metal on the oxide is unrealistic. This requires a mechanism to adjust the interfacial atoms to the mixed bonding environment at the interface. This is sometimes implemented by directly modifying the parameters of the first metal-layer [89], however a more elegant approach is to use potentials based on self-consistent charge equilibration and bond order.
The ReaxFF (reactive force field) method and the charge-optimized many-body (COMB) potentials are two well-documented variable-charge reactive potentials that are able to simultaneously treat mixed systems. Both formalisms are based on the very general bond-order concept. This type of potential have a form that resembles a pair potential, with an attractive part, V^A, and a repulsive part, V^R,

$$V_{ij}(r_{ij}) = V^R(r_{ij}) + b_{ijk} V^A(r_{ij}) \quad , \tag{5.5}$$

however, the strength of V^A is modified by the environment of particle i via the b_{ijk} term. If implemented without an explicit angular dependence, this can be shown to be mathematically equivalent to Eq. (5.4).

In particular the COMB model, have been demonstrated for a range of metal-oxide systems. This includes many systems with a metal and its own oxide Zr/ZrO_2 [90], Cu/Cu_2O [91], Al/Al_2O_3 [92], and Ti/TiO_2 [93]. There are some examples of applications to three-element interfacial systems, most of these have used copper as the metallic phase, some examples include Cu/ZnO [94], Cu/TiO_2 [93] and Cu/SiO_2 [95]. The COMB potential is developed for transferability, and hence the interactions are generally not fitted to particular metal/metal-oxide systems. The metal-metal interactions are fitted to the properties of bulk metals and alloys both experimental and from first principles. Equivalently the metal-oxygen interactions are fitted to properties of oxides.

Recently, two new potential types have been introduced that could be a remedy some the restrictions physically motivated potential: Gaussian Approximation Potentials (GAP) [96] and potentials based on artificial Neural Networks (NN) [97]. Both approaches are purely mathematically motivated, and allow the interpolation of the PES using a flexible, non-linear functional form based on a set of reference structures. The number studies using these approaches for metal/metal-oxide systems is limited. At present the only study on metal oxide systems is the development and application to a Cu/ZnO system [98]. However, this is likely to change in the near future.

5.1.2 COMB potential formalism

The COMB potential is based on the Abell-Tersoff potential [99, 100]. This means that the potential is based on the bond order concept, while also taking the directionality of bonds into account. Coupled to this the COMB potential uses dynamic charge equilibration, following the principle of electronegativity equalization by following the formalism of Rappe-Goodard [101]. We will not fully describe every term in the COMB potential, as it is rather extensive and have been done elsewhere [95], however we will describe some of the core concepts (i.e. the charge-dependent bond order potential and the charge equilibriation).

In the COMB potential formalism, the total energy for a collection of atoms with the set relative positions $\{r\}$ and charges $\{q\}$ is given by

$$E^{tot}[\{q\},\{r\}] = E^{es}[\{q\},\{r\}] + E^{short}[\{q\},\{r\}] + E^{vdw}[\{q\},\{r\}] + E^{corr}[\{q\},\{r\}] \quad , \tag{5.6}$$

where E^{es} is the electrostatic energy, E^{short} takes into account short range interactions (bond order term), E^{vdW} considers long range van der Waals interactions via a Lennard Jones potential [64] and the fourth term, E^{corr}, is a correction term.

The electrostatic contribution can be expanded as

$$E^{es}[\{q\},\{r\}] = E^{self}[\{q\},\{r\}] + E^{qq}[\{q\},\{r\}] + E^{qZ}[\{q\},\{r\}] + E^{polar}[\{q\},\{r\}] \quad , \tag{5.7}$$

where E^{self} is the energy required to form a charge on an atom, E^{qq} is the charge-charge interaction, E^{qZ} is charge-nuclei interaction and E^{polar} is the energy associated with atomic polarizeability. Using the Mulliken definition of electronegativity, the self energy is the sum of the ionization energy (or electron affinity energy) of an isolated atom. A correction function, termed the field effect, reflects environmental changes

$$E_{ij}^{self}[\{q\},\{r\}] = E_i^{ion}(q_i) + E_{ij}^{field}(r_{ij}, q_j) \quad . \tag{5.8}$$

The ionization energy is expressed as a Taylor series expansion with respect to its charge, keeping the first four terms

$$E_i^{ion}(q_i) = \chi_i q_i + J_i q_i^2 + K_i q_i^3 + L_i q_i^4 \quad , \tag{5.9}$$

where the parameter χ is identified as the electronegativity, while the higher order terms are associated with chemical hardness or self-Coulombic interaction. The field energy is associated with specific bonds and is given by

$$E_{ij}^{field}(r_{ij}, q_j) = \frac{1}{4\pi\epsilon_0} \sum_{j\neq i}^{NN} \left(\frac{P_{ij}^{\chi} q_j}{r_{ij}^3} + \frac{P_{ij}^{J} q_j}{r_{ij}^3} \right) . \tag{5.10}$$

The charges are equilibrated by considering the derivative of the potential with respect to the charge assigned to each atom. This derivative can be seen as a chemical potential, which at equilibrium should be equal at all atoms.
The term for the short range interactions is given by

$$E^{short} = \frac{1}{2} \sum_i \sum_{j\neq i} F_c(r_{ij}) \left[V^R(r_{ij}, q_i, q_j) - \frac{b_{ij} + b_{ji}}{2} V^A(r_{ij}, q_i, q_j) \right] \quad , \tag{5.11}$$

where F_c is the Tersoff cutoff function [100], V^A is an exponentially decaying attractive term, V^R is an exponentially decaying repulsive term and bij and bji are the three-body bond order terms. We note that in general $b_{ij} \neq b_{ji}$. V^A and V^R are given by respectively,

$$V^R(r_{ij}, q_i, q_j) = A_{ij} \exp\left[-\lambda_{ij} r_{ij} + \frac{1}{2}(\lambda_{ii} D_i(q_i) + \lambda_{jj} D_j(q_j)) \right] \tag{5.12}$$

and,

$$V^A(r_{ij}, q_i, q_j) = B_{ij} B_{ij}^*(q_i, q_j) \exp\left[-\lambda_{ij} r_{ij} + \frac{1}{2}(\lambda_{ii} D_i(q_i) + \lambda_{jj} D_j(q_j)) \right] \quad . \tag{5.13}$$

The functions $D_i(q)$ and $B_{ij}^*(q_i, q_j)$ modifies the short range interactions according to the local charge, and the bond order term b_{ij} modifies the short-range attraction based on bond angles, local symmetry and number of nearest neighbors.

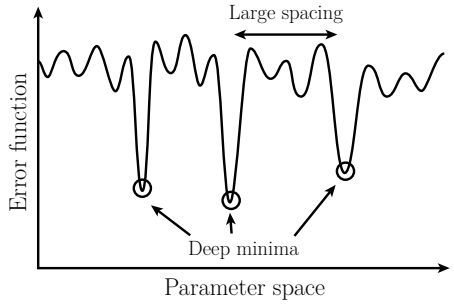


Figure 5.1: Figurative illustration of the error function in parameter space. The deep minima are separated with higher lying minima in between.

5.2 Fitting interatomic potentials

5.2.1 Fitting algorithm

Fitting an interatomic potential requires a description of the accuracy with which the potential predicts a set of material properties. This is described by the error function [102]. Let $\tilde{p}_n$ be the "true" value of a material property and $p_n(\Theta)$ be the prediction of an interatomic potential, then we write the error function as

$$E(\Theta) = \sum_n \left(\frac{p_n(\Theta) - \tilde{p}_n}{\delta p_n} \right)^2 \quad , \tag{5.14}$$

where δp_n is the target relative accuracy and Θ is the set of parameters defining the potential. While an interatomic potential generally has the capability to describe, with some level of precision, many properties of a given system, it cannot describe all of the properties of a system with high fidelity. It is therefore necessary to decide early in the development of the error function which properties are the most important and choose the appropriate relative accuracies. An additional, consideration is the accuracy with which the properties are known: Fitting a material property more accurately than the uncertainty of that value does not make sense. Hence, the target accuracy of a particular value should be no better than its uncertainty.

When a database of material properties is constructed, the challenge is to find the set of parameters that minimizes the error function. The error function is highly non-linear and non-convex, and finding the global minima is generally not possible. We will search for good minima heuristically, using random walk with local minimization at each step. Local minimization is required, since the minima are deep and well separated, whereas the jumps during the random walk, lets the optimization escape poor local minima. This is illustrated in Fig 5.1.

The optimization progresses as follows: At the n'th step of the random walk, given a set of parameters Θ_n, a new set of parameters Θ'_{n+1} is constructed by applying a random perturbation to each

parameter based on a normal distribution. We use a normal distribution, rather than a uniform distribution, since we want to search for sets of parameters close to previous set, while retaining the ability to possibly make large jumps. The new set of parameters is adjusted by local minimization using the Nelder-Mead downhill simplex algorithm [103]. We have to resort to this method, rather than faster gradient descent methods, since the derivative of the error function is unavailable. Once the local minima is found, the new set of parameters Θ_{n+1} is accepted based on the probability

$$P(\Theta'_{n+1}|\Theta_n) = \min\left\{1, \exp\left(-\frac{E(\Theta_{n+1}) - E(\Theta_n)}{E(\Theta_n)T}\right)\right\} \quad , \tag{5.15}$$

where T is a fictitious temperature used to tune the probability for acceptance. If T is low, only sets of parameters improving on the previous set is likely to be accepted, while if T is large most new sets will be accepted. If accepted Θ_{n+1} is set to Θ'_{n+1} and the algorithm makes a new jump, until a specified number of steps is reached.
We will generally run multiple parallel searches; one node uses the initial input parameters as starting point, whereas all the others uses randomly generated parameters based on a normal distribution around the initial guess. The randomly generated potentials can occasionally result in errors during evaluation of the error function. This is handled by catching all errors and assigning a very high value to the error function.

5.2.2 DFT

Wherever possible we fit to materials properties from experimental data, when this is not available we use first-principles data from the literature, and when this is not available, we perform our own DFT calculations. All DFT calculations are performed with the GPAW code [104, 105]. The electron exchange and correlation are described with the generalized gradient approximation (GGA) of Perdew, Burke, and Ernzerhof (PBE) [106], using the atomic setups supplied by GPAW with the projected augmented wave (PAW) method [107]. The calculations utilize plane-wave basis sets with a 500 eV energy cutoff. We use a Monkhort-Pack k-point mesh [108] with at least 1000 k-points per reciprocal atom (i.e. the number of k-points times the number of atoms is 1000 or more).

5.2.3 Material properties

Lattice parameters

The crystalline structure of a material can be described by a basis and a set of lattice parameters [109]. In many cases the basis is constrained by symmetry, hence the lattice parameter is the only structural parameter describing the bulk material. Additionally, lattice parameters are well documented experimentally or easy to obtain from an electronic structure calculation. For a cubic crystal there is only one degree of freedom; uniform scaling of the unit cell. Hence, the lattice

constant can be found by calculating the energy for a range of scales spanning 2 % on each side of the true equilibrium. The optimum lattice parameter is then found by fitting a third order polynomial to the volume. The same principle can be applied when there are multiple degrees of freedom. However, we find that this method is generally outperformed by energy minimization with respect to the lattice parameters using the the simplex algorithm.

Formation energies

The formation energy is the energy required to form a structure given a set of atoms in a reference structure. The cohesive energy of an elemental crystal is the energy needed to isolate its atoms

$$E_{coh} = E_{free} - E_{bulk} \quad , \tag{5.16}$$

where E_{bulk} is the energy per atom of the bulk material and E_{free} is the energy of a free neutral atom.

The heat of formation of a compound, such as an oxide or an alloy, is the energy needed to separate the constituents in their ground state reference structure. For the compound C of two different elements, A and B, the formation energy per atom can be written

$$E_f = \frac{E^C_{bulk} - n_A E^A_{ref} - n_B E^B_{ref}}{n_A + n_B} \quad , \tag{5.17}$$

where E^C_{bulk} is the energy of the compound, E_{ref} is the energy per atom of the reference structure and n is the number of atoms in the reference structure of a given element. For example, the formation energy per atom of gold oxide is calculated as

$$E_f = \frac{5E^{\mathrm{Au2O3}}_{bulk} - 2E^{\mathrm{Au}}_{bulk} - \frac{3}{2}E^{\mathrm{O2}}_{gas}}{5} \, , \tag{5.18}$$

where $E^{\mathrm{Au2O3}}_{bulk}$, E^{Au}_{bulk} and E^{O2}_{gas} are the energy per atom of gold oxide, bulk gold and the oxygen molecule.

The calculation of adsorbtion energies follows a similar vein

$$E_{ad} = -\frac{1}{N_{\mathrm{O}}}\left[E^{\mathrm{O/Au}} - E^{\mathrm{Au}} - N^{\mathrm{O}}E^{\mathrm{O}}\right] \quad , \tag{5.19}$$

where $E^{\mathrm{O/Au}}$, E^{Au}, and E^{O} represent the total energy of the adsorbate-substrate system, the clean surface, and the free oxygen atom, respectively, and N^{O} is the number of oxygen atoms in the unit cell.

Elastic constants

Elastic constants describes the elastic properties of a material, in particular the energy it takes to deform the crystal. Assuming small strain, the relationship between stress and strain is described

by Hooke's law

$$\sigma = C\epsilon \quad , \tag{5.20}$$

where σ is the stress tensor, ϵ is the strain tensor and $\boldsymbol{C}$ is the elastic stiffness tensor. Enforcing zero torque, there are six independent stress and strain components, and hence 36 eleastic stiff constants. This can be reduced to three using crystal symmetry

$$\begin{pmatrix} \sigma_{xx} \\ \sigma_{yy} \\ \sigma_{zz} \\ \sigma_{yz} \\ \sigma_{zx} \\ \sigma_{xy} \end{pmatrix} = \begin{pmatrix} C_{11} & C_{12} & C_{12} & 0 & 0 & 0 \\ C_{12} & C_{11} & C_{12} & 0 & 0 & 0 \\ C_{12} & C_{12} & C_{11} & 0 & 0 & 0 \\ 0 & 0 & 0 & C_{44} & 0 & 0 \\ 0 & 0 & 0 & 0 & C_{44} & 0 \\ 0 & 0 & 0 & 0 & 0 & C_{44} \end{pmatrix} \begin{pmatrix} \epsilon_{xx} \\ \epsilon_{yy} \\ \epsilon_{zz} \\ \epsilon_{yz} \\ \epsilon_{zx} \\ \epsilon_{xy} \end{pmatrix} \tag{5.21}$$

. The bulk modulus describes the stiffness of a crystal to a uniform dilation, i.e. a uniform increase of volume caused by a deformation in all three principal directions. The bulk modulus is related to the elastic stiffness tensor as

$$B = \frac{1}{3}(C_{11} + 2C_{12}) \quad . \tag{5.22}$$

The elastic stiff constants can be calculated by applying a series of strains in different directions. For example, C_{11} is found by by applying a series of strains in the x-direction, and calculating the resulting stresses σ_{xx}, from which C_{11} can be found through a linear fit.

Surface energies

The surface energy is the energy per unit area associated with cleaving a crystal along a given crystal plane. The equilibrium shape of mesoscopic nanoparticles is determined by the surface energy, hence it is important that the metal potentials describes the surface energies well.
In principle, the surface energy, γ, can be obtained by calculating the energy of a slab with N layers and subtract the contribution due to the bulk

$$\gamma = \lim_{N\to\infty} \frac{1}{2}(E^N_{slab} - NE_{bulk}) \quad , \tag{5.23}$$

where E^N_{slab} is the total energy of the slab and E_{bulk} is the bulk energy of one layer in the slab. However, the above expression diverge, if there is a tiny difference in energy between the bulk energy and the change in energy between a slab with N and $N+1$ layers [110].
Instead, we use that, for large N, Eq. (5.23) can be written as [111]

$$E_N \approx 2\gamma + NE_{bulk} \quad , \tag{5.24}$$

from which, given a series of calculations for increasing N, the surface energy can be obtained from a linear fit, without reference to the bulk energy. It is difficult to measure the surface energy of solid metals and generally experimental surface energies are only available for liquid metals, which can be extrapolated to 0 K. Hence, to describe the anisotropy of the surface energy for different facets, the surface energy have to be calculated from first-principles calculations. There is a huge spread in surface energies predicted with DFT depending on the exchange-correlation functional. The PBE functional generally predicts a too low surface energy, this is especially true for the noble metals, including gold and platinum. Hence, we use energies based on the PBEsol functional [112], which have been shown to reproduce surface energies well.

5.3 Fitting the COMB potentials

The complexity of the COMB potential necessitates, that it is fitted in stages. The fitting process starts with the simplest systems, then slowly adding increasingly more complex systems with more complex physics, bringing into play additional terms and their associated parameters. We will optimize the parameters in three stages. We start by fitting the pure metals, then the single-element charge dependent parameters and lastly we fit the binary parameters for the alloys and oxides. The process of fitting the COMB potential is well-documented by the original authors [102]. Parameters for O, Zr, and Ti have been taken from existing potentials [93, 90].

The training database for the pure metals, Au and Pt, consists of fourteen properties. The cohesive energy and bond length of the dimer. The cohesive energy, lattice constant and three elastic constants of the face-centered cubic phase. The phase change energy of the hexagonal close packed and body centered cubic phase. Lastly, we included the surface energies of the three low index facets ((111), (100) and (110)). The properties are given in Tables 5.1 and 5.2 for Pt and Au respectively, along with a comparison of our COMB values to two other potentials.

We compare to the EAM potential because it is known to perform well for metals, and a large number of predicted properties are available. We also compare with the ReaxFF potential, due to its similarity to COMB and since a potential for the corresponding oxides exists. After fitting the metal potentials, a set of parameters for these interactions, were published by the original authors of the COMB potential [113].

During optimization, the parameters for both metals, were initialized to the published COMB copper parameters. We used the random walk algorithm to search for minima with a total 400 steps, using jumps of 5 % and a temperature of 0.014. The number of minima within 0.5 % of the best were 2.4 %.

The training databases for the mixed systems, Pt-Ti, Pt-Zr, Au-Ti, Pt-O and Au-O, mainly consisted of the lattice constants and formation energies of eight different crystal structures for each interaction. The crystal structures were chosen as known stable structures and some simple hypo-

thetic phases. The structures were chosen such that different coordinations and bond angles were probed. In addition to the crystal structures, we included the adsorbtion energy of oxygen for the metal-oxygen interactions.

Since, we were not interested in the properties of any of the specific oxides or alloys, we have weighed the different structures fairly equally in the training database, with a moderate bias towards predicting the known stable structures well. The included crystal structures and predictions for the minimized energies of the included crystal structures is given in Tables . The structures are defined using the prototype structure as given in the Inorganic Crystal Structure Database (ICSD) [114].

The parameters for the mixed systems were initialized using the mixing rules given in [95]. As for the pure metals, we used the random walk algorithm to search for minimas with a total ~ 100 steps, using jumps of 5 % and a temperature of 0.014. The fewer steps were due to the increased computational cost of calculating the properties of the mixed systems.

We will not go through an evaluation of each interaction here. This may be done elsewhere, once the interactions have been further validated for the metal/metal-oxide systems. The final parameter sheets are likely to be published elsewhere after further validation and are available upon request.

5.4 Application

We were, due to the project ending, unable to really explore the fitted potentials. In some of our preliminary simulations with Pt and Au on rutile TiO_2, we found a strong interaction, which during thermal equilibration immediately led to significant rearrangements of the atoms at the interface. From experimental results this is not unrelastic, as this interface is known to be reactive, even leading to full encapsulation of the nanoparticle by a TiO_x overlayer under certain conditions [86, 87, 88]. However, further validation, using DFT simulations, have to be performed before we can confidently use the potentials.

We instead present an application of the potentials, where this issue is less important. The simulations presented here are preliminary, and serve as a demonstration of using thermal ensembles from molecular dynamics, as input models for simulations of HRTEM images. Our goal was to better understand, how the interface influences the strain distribution in a supported nanoparticle, and what can be inferred about this, from a profile-view HRTEM image. The influence of oxidizing and reducing the support on the strain distribution of nanoparticles was recently investigated using HRTEM by López-Haro et al.

We investigate two systems, chosen mainly due to their large difference: Pt supported on cubic ZrO_2 and Pt on anatase TiO_2. The lattice mismatch for the Pt(100)‖anatase(100) interface is just ~ 3.4 %, whereas the lattice mismatch for the Pt(111)‖ ZrO_2(111)-interface is ~ 24 %. The systems are shown in Fig. 5.2.

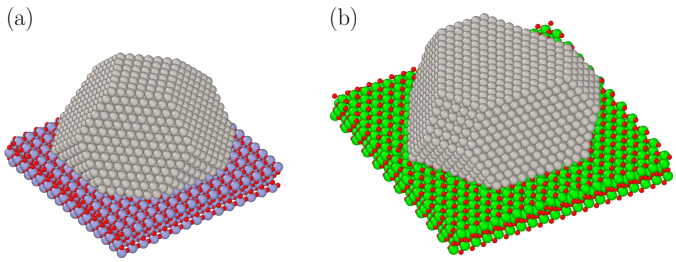


Figure 5.2: **(a)** Pt nanoparticle on anatase TiO_2. **(b)** Pt nanoparticle on cubic ZrO_2.

5.4.1 Pt/c-ZrO_2

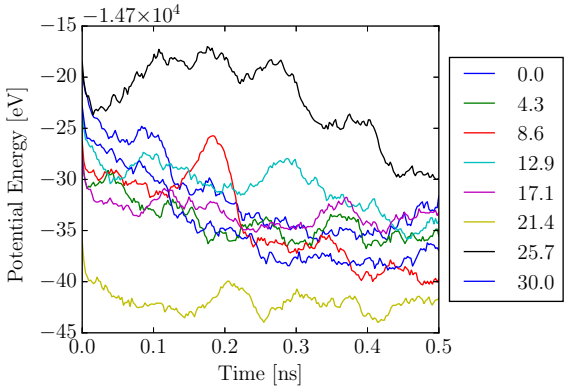
The system is initialized by placing the top half of a Wulff constructed nanoparticle, oriented with the {111}-facet along the interface, on an oxygen terminated ZrO_2(111)-slab. The cubic phase of ZrO_2 is only stable at high temperature, or can be stable at room temperature by adding dopants. We use the cubic phase rather than the monoclinic groundstate for simplicity.

The dimensions of the ZrO_2 surface slab is $66 \times 68 \times 8$ and consists of 3,240 atoms, whereas the nanoparticle consists of 3,091 atoms. A thicker slab may be needed for accurately representing the oxide, however the thinner slab was used to save computational time. We fix the bottom layer of the oxide. The only optimization we do is thermal equilibration, with a slightly elevated temperature, this ensures that the structure can escape shallow local minima. However, it is not sufficient for searching for global minima. We use Langevin dynamics to bring the system into thermal equilibrium. This entails coupling all atoms of the system to a thermal bath at a given temperature, ensuring that all atoms of the system exhibits ensemble behaviour [64].

During equilibration the system is gradually heated to 800 K over 0.25 ns, then cooled to 300 K over 0.25 ns. The time step was 0.5 fs, for a total of 1 million steps. Due to the large lattice mismatch at the interface, it is not clear, how the surfaces should be rotated or translated with respect to each other. Hence, we initiated the simulations at various different rotations and translations of the nanoparticle. Fig. 5.3(a) shows the total potential energy as a function of time during equilibration.

The energy separation between the different rotations is on the order of ~ 20 eV, corresponding to the bulk cohesive energy of just 4-6 Pt atoms. Our interpretation is that the lattice mismatch is so large, that a relative rotation only slightly modifies the adhesion.

In Fig. 5.4, we show the strain distribution of a nanoparticle, which have been brought into thermal equilibrium as above. The input atomic positions have been averaged over the last 0.05 ns of the thermal equilibration. The strain is calculated using the method of Larsen et al. [115]. The strain distribution is seen to be disordered. This is not due to using the thermal average, since approximately the same strain distribution is found when fully relaxing the nanoparticle.



(a)

Figure 5.3: **(a)** Total potential energy as a function of time during thermal equlibration for a system a Pt nanoparticle and a TiO_2 substrate. We show a curve for a nanoparticle initialized with the natural lattice constant of the Pt potential (3.93 Å), and with the lattice constant of the substrate (3.78 Å).

Even though the strain is generally disordered, there is two features we will highlight for later comparison: The extended area with expansive strain in the ϵ_{xx}-map and in the ϵ_{yy}-map the multiple small areas with compressive strain, with a spacing of approximately four times the lattice spacing of platinum. The local strain generated by the interface, although quite large, only penetrates approximately 3-4 layers up from the interface into the nanoparticle.

Fig. 5.6 shows the strain in the average column positions for a projection along the $\langle 110 \rangle$. The average projected column positions were defined in section 4.3. Comparing this to the actual strain from the 3d models. We see that both of the highlighted strain features can be recognized to some extent. The next question is whether this carries over in a strain measurement from a HRTEM image. Fig. 5.5 shows a simulated HRTEM image, calculated from an average of 40 structures from the last 0.05 ns of the thermal equilibration. The microscope parameters were the same as those used in chapter 4, with a defocus $\Delta f = 10$ nm. The electron dose is assumed to be infinite.

The strain error is generally ~ 2 % or less in the upper part of the nanoparticle, but it is significantly larger along the interface. Hence, when interpreting the measured strain we should not look at these layers. Nonetheless, both of the highlighted strain features can be observed to some extent in the measured strain.

5.4.2 Pt/anatase-TiO_2

The system is initialized by placing the top half of Wulff constructed nanoparticle oriented with the {100}-facet parallel to the interface, on an oxygen terminated anatase TiO_2(111)-slab. The

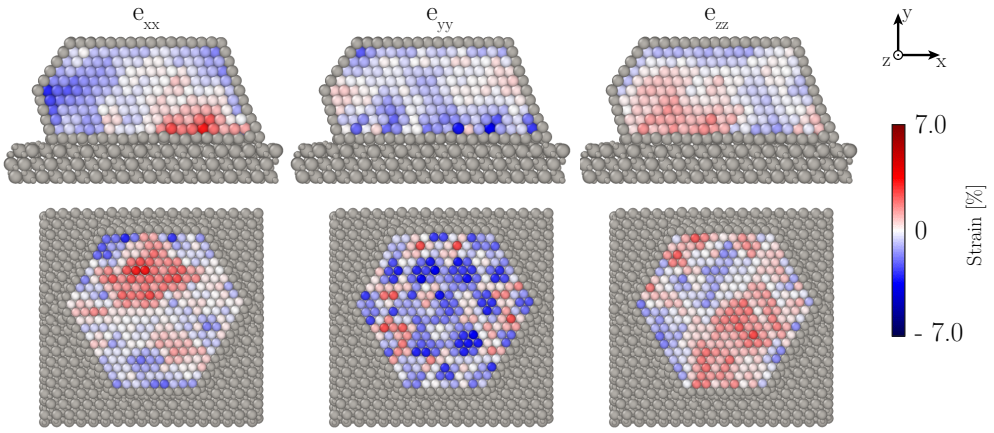


Figure 5.4: The 3d strain distributions in the three principal directions at slices through a ZrO_2 supported Pt particle. In the upper row the slice is perpendicular to the subtrate through the center of the nanoparticle. In lower row slice is parallel to the interface, and the strain is shown for the second metal layer from the interface. The coordinate system is indicated in the upper right corner.

Figure 5.5: Simulated HRTEM image of a platinum nanoparticle on cubic ZrO_2. The simulation parameters follows those used in section 4 with a defocus $\Delta f = 8$ nm.

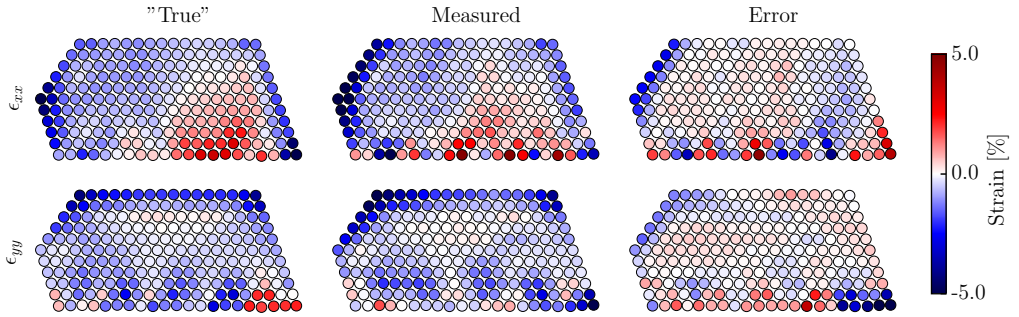


Figure 5.6: The first row shows the "true" 2d strain given the averaged column positions. The second row shows the strain measured from the electron microscope image in Fig. 5.5. We also show error, i.e. the difference between the first true and measured strains. The strain was measured using structural template matching, with partial matching for the surface and edge atoms.

dimensions of the TiO_2 surface slab are $62 \times 62 \times 6.3$ and it consists of 2,700 atoms, whereas the nanoparticle consists of 2,155 atoms. The Pt(100)‖anatase(100)-interface is essentially coherent, and there is two feasible relative positions of the platinum with respect to the TiO_2, Pt on top of Ti or Pt on top of O. We found that the most stable configuration was, Pt on top of O, which is in agreement with DFT [75]. When the system was initialized with Pt on top of Ti, it became extremely unstable due to a large amount of energy released by atomic rearrangements during the thermal equilibration.

The system is equilibrated at 300 K for 0.5 ns, with a timestep of 0.5 fs. Fig. 5.7 shows the total potential energy as a function of time during equilibration. The system oscillates violently initially, but settles without any major rearrangements. We tried initializing the lattice constant of the platinum to both the actual lattice constant for the Pt potential, and the lattice constant of the substrate, and found that this resulted in the same final structure.

We found that, after initial equilibration, the nanoparticle completely adapted to the substrate, in one of the two principal directions parallel to the surface. Since the x and z directions are symmetrically equivalent, there is no obvious reason for this. Running the same equilibration multiple times, we saw that the direction that adapts to the support, seems to be randomly chosen. Hence, the reason could be that adapting to the substrate in both directions, raises the strain energy too much compared to the energy gained by adapting to the substrate.

In Fig. 5.8, we show the strain distributions, given the thermally averaged atomic positions. This strain is compared to that of a free nanoparticle, demonstrating the influence of the support. We observe a large normal strain in the x-direction, as it is compressed to match the support. Conversely, there is almost no difference between ϵ_{zz} for the supported and the free nanoparticle.

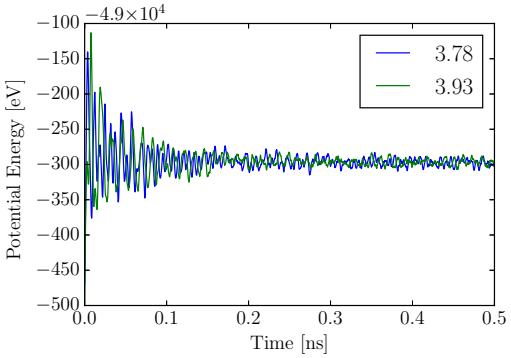


Figure 5.7: Total potential energy as a function of time during thermal equlibration for a system a Pt nanoparticle and a ZrO_2 substrate. We show curves for initializing the nanoparticle with its natural lattice constant and with the lattice constant of the substrate.

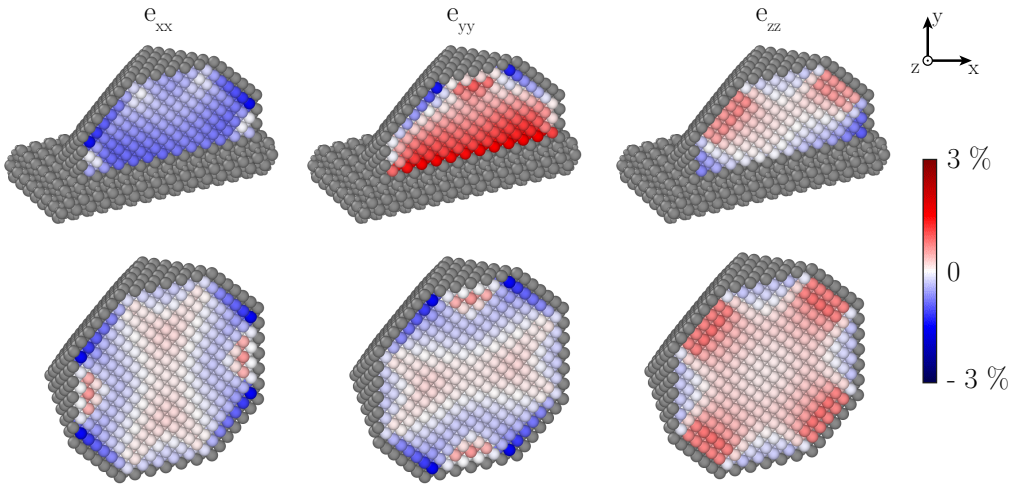


Figure 5.8: The 3d strain distributions in the three principal directions at slices through a TiO_2 supported Pt particle and a free particle of the same shape. The coordinate system is indicated in the upper right corner.

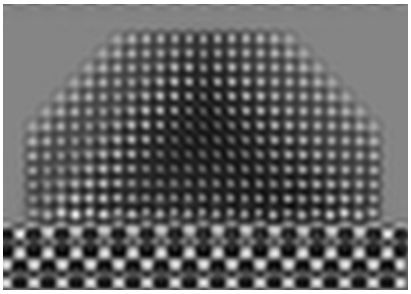


Figure 5.9: Simulated HRTEM image of a platinum nanoparticle on cubic TiO_2. The simulation parameters follows those used in chapter 4 with a defocus $\Delta f = 8$ nm.

Due to the large Poisson ratio of Pt, the nanoparticle is expanded in the y-direction, to compensate the compression. While the strain is smaller close to the interface for the TiO_2-support compared to ZrO_2, the strain penetrates much further into the nanoparticle.
Fig. 5.10 compares the strain from the average column positions, for a projection along the $\langle 100 \rangle$ zone axis to the strain calculated from the simulated HRTEM image in Fig. 5.9. The image was simulated in the manner described in the section above.
Due to the large thickness variations in the $\langle 100 \rangle$ zone axis, a significant error is introduced, in particular in vicinity of the surfaces. While the strain measurement does correctly show the overall strain, in both directions, the asymmetric structure in the measured strain is entirely wrong.

5.5 Summary

Interatomic potentials for Pt/TiO_2-, Pt/ZrO_2- and Au/TiO_2-systems were developed, however more work is needed to validate the potentials, before they can be applied to predict complex interface structures and energies with confidence. We did preliminary simulations of Pt nanoparticles on TiO_2 and ZrO_2, two systems with very different interfaces. The structural simulations were used as input for TEM image simulations, from which the strain was measured and compared to the true 3D strain. We saw that the strain gave a reasonable indication of the support-induced strain. However, significant errors also means, that it is necessary to be very careful when interpreting the strain. Nonetheless, by correlating measured strain from TEM, with simulations it may be possible to obtain a better understanding of interfaces.

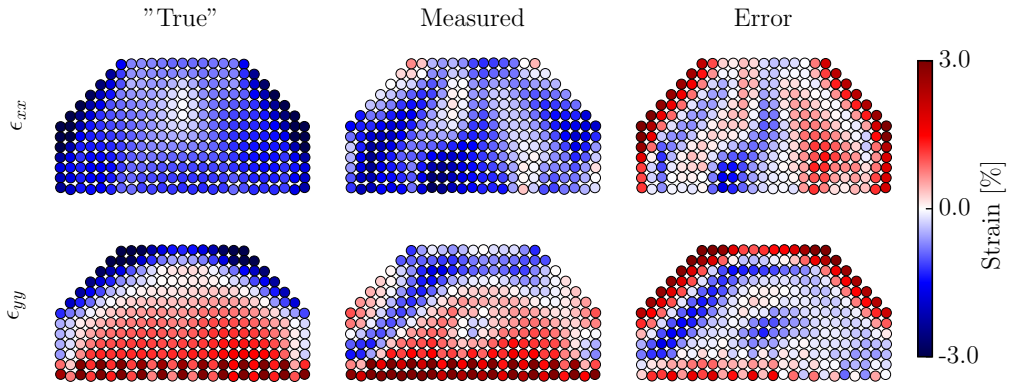


Figure 5.10: The first row shows the "true" 2d strain given the averaged column positions. The second row shows the strain measured from the electron microscope image in Fig. 5.9. We also show error, i.e. the difference between the first true and measured strains. The strain was measured using structural template matching, with partial matching for the surface and edge atoms.

5.6 Tables of material properties

	Exp. or DFT	EAM [116]	ReaxFF [117]	COMB3
fcc properties				
a (0 K) [Å]*	4.06[a]	4.06	4.06	4.038
a (300 K) [Å]	4.078[b]	4.078	-	4.058
E_0 [eV]*	-3.81[c]	-3.93	-3.91	-3.852
B [GPa]	180.3[c]	178	177	172
C_1 [GPa]*	201.6[c]	193	249	212
C_{12} [GPa]*	169.7[c]	165	142	152
C_{44} [GPa]*	45.4[c]	45	147	48.4
melting point [K]	1337[b]	1281	-	$\sim$ 1200
phase transitions				
ΔE (hcp-fcc) [eV]*	0.002[g], 0.0054[h]	0.007	0.0002	0.007
ΔE (bcc-fcc) [eV]*	0.04[d], 0.026[h]	0.04	0.14	0.033
ΔE (diamond-fcc) [eV]	1.01[h]	0.95	1.17	1.00
ΔE (sc-fcc) [eV]	0.28[h]	0.43	0.86	0.54
planar defects				
$\gamma(111)$ [J/m^2]*	0.74[i], 0.98[j], 1.5[e]	1.15	1.07	0.868
$\gamma(100)/\gamma(111)$*	1.16[i], 1.22[j]	1.06	1.13	1.19
$\gamma(110)/\gamma(111)$	1.31[j]	1.11	-	1.25
γ (Intrinsic) $\langle 12\bar{1}\rangle$ [J/m^2]	27[k]	30	-	32
γ (Unstable) $\langle 12\bar{1}\rangle$ [J/m^2]	94[k]	-	-	170
point defects				
E_f (V) [eV]	0.89[f]	0.97	1.07	1.15
E_f (octahedral) [eV]	-	3.57	-	3.58
E_f ([111]-dumbbell) [eV]	-	4.20	-	4.35
E_f ([110]-dumbbell) [eV]	-	3.88	-	3.93
E_f ([100]-dumbbell) [eV]	-	3.75	-	3.44
clusters				
E_f (Au_2) [eV]*	-1.17[b]	-	-1.2	-1.19
bond length (Au_2) [Å]	2.47[b]	-	2.52	2.38

[a] Exp. [118], [b] Exp. [119], [c] Exp. [109], [d] Exp. [120], [e] Exp. [121], [f] Exp. [122], [g] DFT-PW91 [123], [h] DFT-LDA [117], [i] DFT-PBE [124], [j] DFT-PBEsol [125], [k] DFT-PBE [126] .

Table 5.1: Material properties for fitting the COMB Au potential. The star indicates a property included in the error function.

	Exp. or DFT	EAM [116]	ReaxFF [127]	COMB3
fcc properties				
a (0 K) [Å]*	3.91[a]	3.91	3.95	3.93
a (300 K) [Å]	3.92[b]	4.078	-	3.94
E_0 [eV]*	-5.84[c]	-5.77	-5.77	-5.87
B [GPa]	288.4[c]	282	-	279
C_1 [GPa]*	347[c]	347	-	351
C_{12} [GPa]*	251[c]	253	-	243
C_{44} [GPa]*	77[c]	78	-	77
melting point [K]	2041[b]	1890	2047	$\sim$ 1900
phase transitions				
ΔE (hcp-fcc) [eV]*	0.03[d]	0.03	0	0.04
ΔE (bcc-fcc) [eV]*	0.16[d]	0.16	0.12	0.18
ΔE (diamond-fcc) [eV]	1.01[h]	1.16	1.8	1.86
ΔE (sc-fcc) [eV]	0.28[h]	0.37	1.24	0.74
planar defects				
$\gamma(111)$ [J/m^2]*	0.74[i], 1.91[j], 1.5[e]	1.69	1.68	1.77
$\gamma(100)/\gamma(111)$*	1.16[i], 1.17[j]	1.05	1.13	1.20
$\gamma(110)/\gamma(111)$	1.22[j]	1.14	1.2	1.24
γ (Intrinsic) $\langle 12\bar{1}\rangle$ [J/m^2]	-	121	-	126
γ (Unstable) $\langle 12\bar{1}\rangle$ [J/m^2]	-	320	-	557
point defects				
E_f (V) [eV]	0.89[f]	1.5	-	1.4
E_f (octahedral) [eV]	-	5.0	-	6.9
E_f ([111]-dumbbell) [eV]	-	6.1	-	8.8
E_f ([110]-dumbbell) [eV]	-	5.4	-	7.8
E_f ([100]-dumbbell) [eV]	-	5.1	-	6.3
clusters				
E_f (Pt_2) [eV]*	-1.65[b]	-	-1.2	-1.64
bond length (Au_2) [Å]	2.4[b]	-	2.52	2.32

[a] Exp. [118], [b] Exp. [119], [c] Exp. [109], [d] Exp. [120], [e] Exp. [121], [f] Exp. [122], [g] DFT-PBE [127], [h] DFT-LDA [117], [i] DFT-PBE [124], [j] DFT-PBEsol [125] .

Table 5.2: Material properties for fitting the COMB Pt potential. The star indicates a property included in the error function.

	Exp. or DFT	ReaxFF [117]	COMB3
formation energies			
$E_f(A_2O_3)$ [eV]*	0.042[a], -0.027[b], -0.104[c]	-0.095	-0.036
E_f(AuO-Cu_2O) [eV]*	0.16[a]	0.12	0.11
E_f(AuO$_2$-CaF_2) [eV]*	0.61[a]	-	0.62
E_f(Au_2O-CaF_2) [eV]*	0.72[a]	-	0.57
E_f(AuO-NaCl) [eV]*	0.54[a]	-	0.24
E_f(AuO$_2$-$CaCl_2$) [eV]*	0.26[a]	-	0.033
E_f(Au_2O-Ag_2F) [eV]*	0.47[a]	-	0.26
E_f(Au_3O_4-Pt_3O_4) [eV]	0.08[a]	-	0.08
E_f(AuO-CsCl) [eV]	0.79[a]	-	0.74
adsorbtion			
0.25 ML (111-fcc) [eV]	-3.25[d]		-3.4
0.5 ML (111-fcc) [eV]*	-2.91[d]		-2.7
0.75 ML (111-fcc) [eV]	-2.45[d]		-2.2
1 ML (111-fcc) [eV]	-1.99[d]		-1.9

[a] This work DFT-PBE, [b] Experimental [128], [c] DFT [129], [d] DFT[130]

Table 5.3: Material properties for fitting the COMB Au/O potential. The star indicates a property included in the error function. The adsorption energies are calculated with respect to a free oxygen atom.

	Exp. or DFT	ReaxFF [131]	COMB3
formation energies			
E_f(PtO_2-$BaSi_2$) [eV]*	-0.43[a], -0.46[b]	-0.43	-0.38
E_f(PtO_2-$CaCl_2$) [eV]*	-0.47[a], -0.46[b]	-0.27	-0.41
E_f(PtO-PdO) [eV]*	-0.26[a]	-0.25	-0.22
E_f(Pt_2O-Cu_2O) [eV]*	0.14[a]		0.11
E_f(PtO-NaCl) [eV]*	0.57[a]		-0.32
E_f(PtO_2-CaF_2) [eV]*	-0.066[a]		-0.07
E_f(PtO-CsCl) [eV]*	0.79[a]		0.84
E_f(PtO-rutile) [eV]*	-0.38[a]		-0.47
adsorbtion			
0.25 ML (111-fcc) [eV]*	-1.19[c]	-1.17	-1.13
0.5 ML (111-fcc) [eV]	-0.91[c]	-0.81	-0.65
0.75 ML (111-fcc) [eV]*	-0.52[c]	-	-0.35
1 ML (111-fcc) [eV]	-0.15[c]	-	-0.15

[a] This work DFT, [b] Experimental [128], [c] DFT [132]

Table 5.4: Material properties for fitting the COMB Pt/O interactions. The star indicates a property included in the error function. The adsorption energies are calculated with respect to the oxygen molecule.

	DFT	COMB3
formation energies		
E_f(PtTi2-CaF2) [eV]*	-0.24	-0.43
E_f(PtTi-CsCl) [eV]*	-0.75	-1.0
E_f(TiPt3-AuCu3) [eV]*	-0.84	-0.56
E_f(PtTi-NaCl) [eV]*	-0.48	-0.49
E_f(PtTi-AuCd) [eV]*	-0.88	-1.0
E_f(Pt8Ti-Pt8Ti) [eV]*	-0.42	-0.49
E_f(PtTi3-AuCu3) [eV]*	-0.48	-0.61
E_f(TiPt2-CaF2) [eV]	-0.61	-0.57

Table 5.5: Material properties for fitting the COMB Pt/Ti interactions. The star indicates a property included in the error function. All target values are calculated using DFT from this work.

	DFT	COMB3
formation energies		
E_f(PtZr2-CaF2) [eV]*	-0.51	-0.51
E_f(PtZr-CsCl) [eV]*	-1.15	-0.89
E_f(ZrPt3-AuCu3) [eV]*	-0.84	-0.51
E_f(PtZr-NaCl) [eV]*	-0.73	-0.76
E_f(Pt3Zr-Ni3Ti) [eV]*	-1.11	-1.02
E_f(PtZr-TiI) [eV]*	-1.00	-1.07
E_f(PtZr3-AuCu3) [eV]*	-0.48	-0.42
E_f(Zr5Pt3-Mn5Pt3) [eV]	-0.78	-0.82

Table 5.6: Material properties for fitting the COMB Pt/Zr interactions. The star indicates a property included in the error function. All target values are calculated using DFT from this work.

	DFT	COMB3
formation energies		
E_f(Au2Ti-Au2Ti) [eV]*	-0.42	-0.30
E_f(AuTi-AuCd) [eV]*	-0.38	-0.46
E_f(AuTi-CsCl) [eV]*	-0.31	-0.46
E_f(AuTi-CuTi) [eV]*	-0.27	-0.28
E_f(AuTi2-CaF2) [eV]*	0.10	0.02
E_f(TiAu3-AuCu3) [eV]*	-0.093	-0.092
E_f(AuTi-NaCl) [eV]*	0.005	0.005
E_f(Ti3Au-Cr3Si) [eV]*	-0.37	-0.42
E_f(AuTi3-AuCu3) [eV]	-0.24	-0.51

Table 5.7: Material properties for fitting the COMB Au/Ti interactions. The star indicates a property included in the error function. All target values are calculated using DFT from this work.

Chapter 6

Structural Recognition using Convolutional Neural Networks

TEM has become a routine analysis tool for characterizing the structure of materials at the atomic scale, and with the recent development of a wide range of in-situ TEMs and dedicated environmental TEM, it has become increasingly interesting to capture the dynamic behaviour of materials as video [5, 133]. This progress is supported by improved detectors, with increased sensitivity and faster readout rate, in particular with the introduction of the direct detection cameras [134].

As the process of acquiring large amounts of data becomes easier, our analysis tools also have to become more efficient. In many applications, accurate identification and classification of local structure is a crucial first step in deriving useful information from atomic-resolution images and video. Examples include characterizing the distribution of dopants [135] and defects [136], in situ imaging of phase transformations [137], structural reordering during materials growth [138, 139] and dynamic surface phenomena [140].

Crystallography is the traditional tool used to identify and classify local structures in atomic resolution images. However, the restrictive assumptions of symmetry and periodicity leaves some to be desired. Nevertheless, methods such as Geometric Phase Analysis (GPA) [141], which still adopt many of the assumptions of traditional crystallography have been very successful at extracting local structural information, including defects, strain and phase boundaries [142].

Real space approaches typically rely on localization of certain structures such as individual atoms, atomic columns or small aggregates thereof by their shape in the image. In some cases this can be done automatically by employing simple metrics such as the width and intensity of local intensity extrema. Traditional computer vision techniques such as template matching have also had some success here [43]. Generally, the performance of these automatic approaches are significantly worse than a human expert. The difficulties arise in some part from the phase contrast nature of TEM, making the image extremely sensitive to the distance between the sample and the objective lens. This becomes especially problematic for the analysis of video sequences, since small rotations and

vibrations can modify the appearance of the structures, thus necessitating user involvement at every frame. In addition, delocalization due to optical aberrations and partial coherence cause the appearance of a local structure to be influenced by the surrounding structure. These difficulties are compounded by low signal-to-noise ratios, in cases where the electron dose have to be limited, to reduce radiation damage to the sample [143].

In chapter 3, we introduced a method that once the positions of the structures are extracted can classify them using their geometric relationships. Another recently introduced method accomplishes a similar task [144]. However, these methods are inherently limited since crucial intensity information goes unused at the classification stage. In a recent paper this was approached using unsupervised machine learning. The Laplacian of Gaussian operator was used to extract keypoint features. Then the scale-invariant feature transform (SIFT) was used to extract descriptors of the local region around each keypoint, and agglomorative clustering was used to classify the descriptors [145].

Recently, deep learning has gained much attention, in large part due to the excellent performance of convolutional neural networks (CNNs) in visual recognition tasks. Deep learning have only been used to a very limited extent in the context of atomic resolution microscopy. In 2000, Kirchner et al. published a method for predicting defocus and sample thickness from HRTEM images [146]. Neural networks have also been used for reconstructing the wave function from off-axis electron holograms [147]. Recently, the back-propagation algorithm from neural networks have been used to calculate the partial derivatives, required to optimize the full 3d atomic potential from either a tilt series of TEM images or CBED images [148, 149].

The work highlighted above is either not relevant to our problem, or does not use any of the recent breakthroughs in the application of CNNs to image recognition. Hence, the main influences for the work here are recent applications of CNNs in computer vision. Especially recent applications to biological microscopy imaging, including particle detection [150] and automatic segmentation of brain images [151] from cryo-electron microscopy images. In this chapter, we describe a CNN based method that can classify and localize structures in atomic resolution images. The method is presented for HRTEM, but could in principle be applied to images acquired through other atomic scale imaging methods. In particular STEM and scanning tunnelling microscopy (STM).

6.1 Classification and detection

The goal in computer vision is to infer estimates of meaningful information that relates the input image to the scene it captures and our understanding the scene. In many cases, this can be reduced to classifying the objects present, by predicting class labels along with a description of the locations of the objects. Some examples of computer vision tasks are shown in Fig. 6.1. Today CNNs are not only progressing classification of whole images [152], they are also improving on object detection

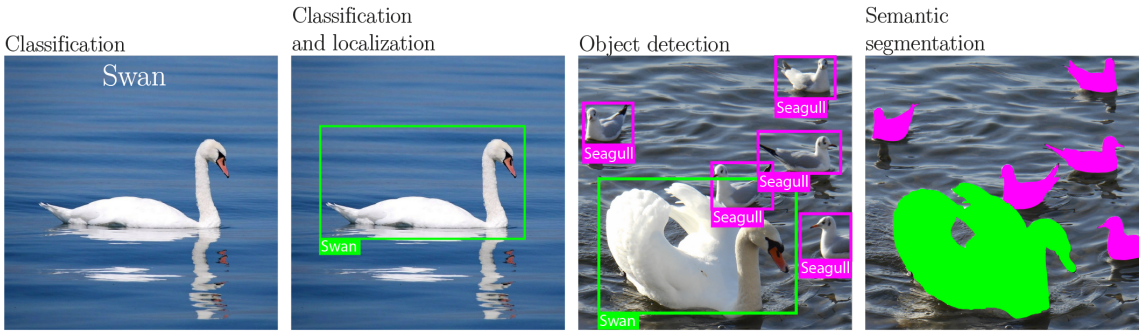


Figure 6.1: Examples of basic tasks in computer vision. Images from pixabay.com.

[153] and pixel level image segmentation [154].

In the simplest case with only one object in the image, the localization can be solved by directly predicting its bounding box coordinates through regression, i.e. a mapping from the image to four numbers. Alternatively, this can be recast as a classification task. By using a classifier on subregions of the full image, it is possible to obtain region-dependent classifications. The classification probabilities in each sub-region may be combined to obtain a prediction of the bounding box coordinate [155].

Classifying and localizing *multiple* objects in an image is referred to as detection. Since the number of objects belonging to each class is generally unknown, it is not possible to build a model for directly regressing the bounding box coordinates. It is however still possible to use methods based on sub-region classification, and recent advances in object detection are driven by the success of region-based CNNs (RCNNs) [155]. This kind of detection has proven to be effective in a wide variety of related vision tasks, such as real-time localization from photographs [153] or cell detection in medical images [150, 156].

Semantic segmentation is the task of partitioning the image into semantically meaningful parts, and to classify each part into one of the pre-determined classes. In pixel-wise semantic segmentation each pixel of the image is assigned a class. The state-of-the-art in segmentation is based on CNNs, which have had success in tasks such as road scene parsing [157] and segmentation of brain scan images [158].

6.2 Feed forward neural networks

Before describing the CNN, we begin with a description of the basic feed-forward neural network. A neural network is based on a collection of connected units called artificial neurons. The connections transmit a signal from one neuron to another, and each neuron transforms the incoming signals

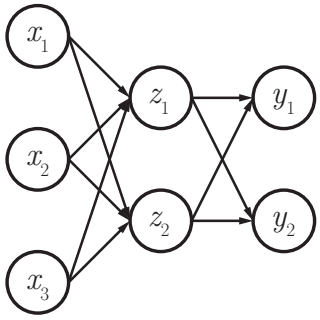


Figure 6.2: A simple feed-forward neural network.

to a single output. The neurons in a network are generally organized into a sequence of layers. The example network in Fig. 6.2 consists of three layers, an input, a hidden and an output layer. Here the nodes of adjacent layers are only connected in the forward direction, this is true for all feed-forward neural networks (but not all neural networks). The connectivity pattern of a neural network is referred to as the network's architecture.

We can express the mathematical operation of a single neuron as follows. Let $\boldsymbol{v}$ denote an input vector to a neuron, then, the neuron computes the activation

$$a = f\left(\sum_{i=1}^{N} w_i x_i + b\right) \quad , \tag{6.1}$$

where w_i is the weight associated with the i'th input connection, b is a bias associated with the neuron and f is a non-linear function (the *activation function*). Some common activation functions include the sigmoid and the hyperbolic tangent.

For the network in Fig. 6.2, we can write the output of a unit y_k as a composition function as follows

$$\hat{y}_k(\boldsymbol{x}; \Theta) = f^{(2)}\left(\sum_{j=1}^{2} W_{kj}^{(2)} f^{(1)}\left(\sum_{i=1}^{3} W_{kj}^{(2)} x_i + b_j^{(1)}\right) + b_k^{(1)}\right) \quad , \tag{6.2}$$

where the superscipt denotes a layer index and $\Theta = \{\boldsymbol{W}^{(1)}, \boldsymbol{W}^{(2)}, \boldsymbol{b}^{(1)}, \boldsymbol{b}^{(2)}\}$ is the set of trained parameters. This process where information propagates through the network is called the forward-propagation step.

A sufficiently large neural network with a single hidden layer can approximate any continuous function on a bounded domain [159]. However, overwhelming empirical evidence have shown that increasing the number of hidden layers is important, this is referred to as as increasing the depth of the neural network. Deep networks tend to be better at making complex predictions, which

seem hard to capture using shallow architectures, and deep networks often lead to better practical performance [160, 161].

The model parameters Θ have to be learned from data. In practice this means optimizing the parameters with respect to a loss function L. For example the mean squared difference between the predicted label $\hat{y}_k(\boldsymbol{x};\Theta)$ and the true label y, given a training set $\{\boldsymbol{x}^{(i)}, \boldsymbol{y}^{(i)}\}$, i.e.

$$L = \sum_i \|\hat{y}_k(\boldsymbol{x}^{(i)};\Theta) - \boldsymbol{y}^{(i)}\|_2^2 \quad . \tag{6.3}$$

The loss function is highly non-linear and non-convex. Hence, we have to resort to an iterative gradient descent algorithm. To efficiently update the parameters throughout the network, requires a method for computing the partial derivatives $\partial L/\partial\theta$ for each parameter $\theta \in \{\boldsymbol{W}^{(1)}, \boldsymbol{W}^{(2)}, \boldsymbol{b}^{(1)}, \boldsymbol{b}^{(2)}\}$. For a feed-forward neural network, this can be calculated efficiently using the error backpropagation algorithm [162]. Once the partial derivatives are obtained the weights can be updated as follows

$$\theta_i = \theta_i - \eta\frac{\partial L}{\partial\theta} \quad , \tag{6.4}$$

where η is the learning rate and i denotes an iteration index. The update process repeats until convergence or until reaching the predefined number of iterations. Typically, each update in Eq. (6.4), utilize a small randomly selected subset (a mini-batch) of training samples (stochastic gradient descent).

6.2.1 Convolutional neural network

Fundamentally, a CNN is simply a multilayer, feed-forward neural network, with three key distinguishing factors: local receptive fields, weight sharing and layers for subsampling (or supersampling).

In fully connected layers, such as those in Fig. 6.2, each neuron is connected to all of the neurons in the following layer. In tasks involving structured input, such as images, the pixels that are close together are strongly correlated while pixels that are far apart are weakly correlated. This is exploited in most standard computer vision algorithms by extracting local features. In the CNN architecture the same principle is used by constraining each neuron to depend only on a local subset of the features in the previous layer (so-called local receptive fields).

Weight sharing means that trained weights are shared across neurons in the hidden layers. Each neuron is computing a weighted linear combination of its input. This is equivalent to filtering the input values by a linear filter. When weights are shared, this means that the same filter is applied to the entire image, which amounts to performing a convolution of the image with the filter, see Fig. 6.3(a). Hence, the CNN effectively learns a set of linear filters applied in a predefined set of sequences to the images. Weight sharing forces the CNN to learn a spatially invariant representation

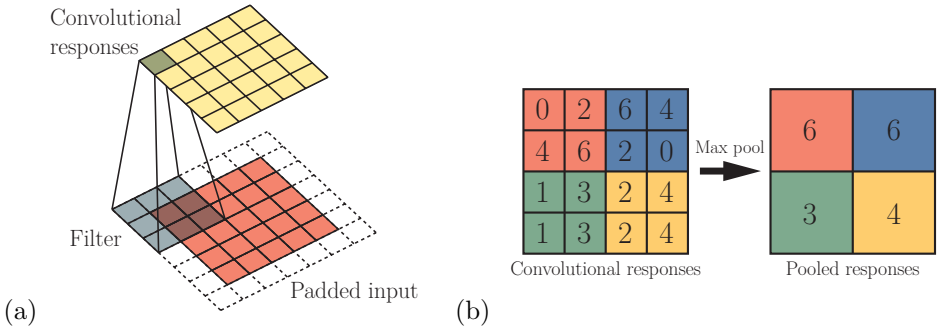


Figure 6.3: **(a)** Convolving a 3×3 filter over a 5×5 input with padding to maintain the same shape in the convolutional responses. The convolutional responses are linear combinations of 9 pixels in the previous layer, weighted by the filter. **(b)** Max pooling in a convolution neural network. Here the max pool is over blocks of size 2×2 on the 4×4 feature map. The max pool is the maximum in each block, yielding the 2×2 pooled response map shown on the right.

of the image, and have the additional significant upside of reducing the number of free parameters, making the network faster to train.

The basic convolutional layers are typically intermixed with pooling layers. The motivation here is twofold: to aggregate multiple low-level features to gain spatial invariance and reduce the dimensionality of the convolutional maps to improve computational efficiency. In pooling, the convolutional response map is first divided into a set of blocks over which a pooling function is evaluated yielding a set of smaller response maps. In the case of max pooling, the response for each block is taken to be the maximum value over the block responses, see Fig. 6.3(b).

A typical CNN, have multiple hidden layers, alternating between convolution and pooling. For example, a convolution-pooling layer can be stacked on top of the outputs of the first convolution-pooling layer. In this case, the outputs of the first set of convolution-pooling layers are simply treated as the input to the second set of layers. In this way, a multilayered or deep architecture can be constructed. Intuitively, the first convolutional layers, can be thought of as providing a low-level encoding of the input data. In the case of image data, this may consist of simple edge detection. Moving to higher layers, the lower level features are combined to produce increasingly complex and abstract features. Neural networks for whole-image classification generally connects the final high-level features directly to a fully connected layers to perform classification.

The use of pooling layers results in very coarse feature maps, hence tasks like semantic segmentation requires a way of upsampling the output. One way to achieve this is interpolation. For example, simple bilinear interpolation computes each output from the nearest four inputs by a mapping, depending only on the relative positions of the input and output pixels. In neural networks interpolation can be a learned operation. Usually termed a transpose convolution, this works by swapping

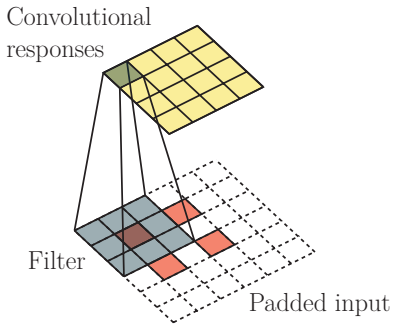


Figure 6.4: Upsampling by a factor of 2 a can be accomplished by convolving a filter over a modified input, obtained by inserting a zero between each pixel. Note that this is equivalent to a transpose convolution, but does not represent the actual operation as it is implemented.

the forward and backward passes of a convolution [154]. This operation is illustrated in Fig. 6.4.

6.3 Methods

Given a set of 2d atomically resolved images, our task is to classify and localize each instance, n, of a set of atomic structures by assigning a set of 2d Cartesian coordinates $\{(x_n, y_n)\}$ and class labels $\{c_n\}$ to each. The class label can be chosen from a set of N_c predefined labels $\mathcal{L} = \{l_1, \ldots, l_{N_c}\}$. As an intermediate result, each instance is assigned a probability distribution, $p_n(l_k)$, given our recognition model, over the set of class labels.

The structure class might be an atom, a column of atoms or a specific configuration of atoms, it might also be a vacancy, i.e. the absence of an atom. Choosing how to categorize the structures is problem specific. The choice should depend on how the researcher derives meaning from the image, and not necessarily rely on the physics.

The outlined problem is an object detection task, for which we mentioned that, recent advances are driven by the success of region proposal methods. However, classifiers based on region-proposals, generally rely on each class being visually distinct regardless of context. This is a problem for atomic resolution images. An atomic structure typically appear as a blob with very little internal distinguishing structure, and the appearance may change in response to the neighbourhood due to delocalization. Hence, successful detection requires inclusion of contextual knowledge.

Similar considerations is behind the development of the segDeepM network [163], which utilizes segmentation to improve object detection. Their approach enabled more accurate bounding boxes in cases where strong contextual cues were available. Other methods utilize probability (or density) maps for each class, representing the probability for finding an instance of a particular structure

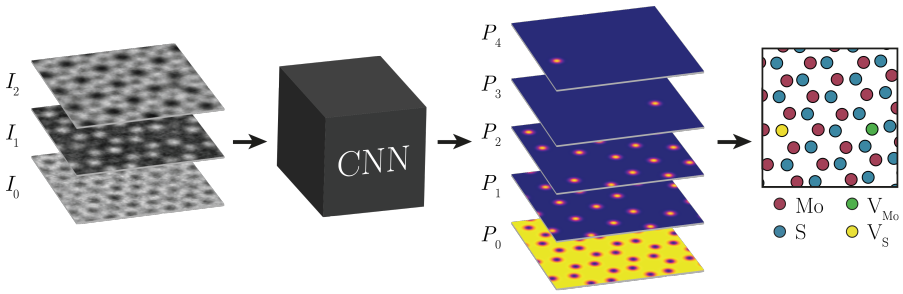


Figure 6.5: Our method for structural recognition exemplified by its application to a single sheet of MoS_2. The input images are converted to a set of probability maps. There is one map for each of the four structure labels $\mathcal{L}$ ={"Mo", "S", "V_{Mo}" and "V_S"} and one for the background class. Finally, the probability maps are interpreted to obtain discrete instances defined by a coordinate and a class labels.

at a given location in the image. This have been successfully used for e.g. human pose estimation [164] and mitosis detection [165]. Similar density maps have been successful in object counting tasks, especially in cases where the distribution of objects is dense [166, 167].

6.3.1 Image-to-image CNN

Our approach is based on generating a set of probability maps using an image-to-image CNN. The approach is illustrated for the case of MoS_2 in Fig. 6.5. The CNN maps from N_f images of size $N_x \times N_y$ to N_c images of the same spatial dimensions. We represent the input as a 3D array of shape $N_x \times N_y \times N_f$. Each component of the input image is denoted, I_{ijk}, and each image is denoted I_k. The input may just be a single gray scale image ($N_f = 1$), and if multiple input images are used, it is assumed that they are resolved over the same spatial domain, for example a focal series. The output is also a 3D array, with the components P_{ijk}, with the same spatial dimensions as the input and a depth equal to the number of class labels plus a background class, $N_x \times N_y \times (N_c + 1)$. The inclusion of a background class is standard in semantic segmentation, it usually takes the same form as the models for the classes of interest, but is supervised by negative instances [154]. The background class ensures competition between the classes by allowing us to impose

$$\sum_k P_k = 1 \quad . \tag{6.5}$$

If two instances overlap spatially, we introduce a new class for the combined structure. For example, in order to classify atomic columns of different heights, we would create a separate class for each height. In approaches for object counting, the same task is accomplished by predicting a density proportional to the number of objects in a subregion [167]. The advantage of our model is that the

uncertainty with respect to, e.g. two very different column heights, can be properly expressed.

6.3.2 Supervised learning

The main drawback of using a deep neural network is that it requires a huge amount of data for training. Hence, in some fields, researchers have collected databases containing tens of thousands of hand labelled images from hundreds of categories [168]. Unfortunately, such databases does not exist for atomic resolution images. Rather than attempt the expensive process of creating a hand-labelled dataset, we create training data exclusively from simulation. We show that a CNN trained on simulation is able to generalize from simulation to experiment, without ever being shown a real image. The idea of leveraging simulation for inexpensive training of neural nets have also been applied successfully in fields such as robotics [169].

We simulate a new training image for every training iteration, hence the main way the network is regularized is by ensuring that the data have a large degree of variability. Nonetheless, we found that a moderate weight decay improves the performance on experimental data. By decaying all weights, any weights that are not being used by the network to produce meaningful output will become negligible, rather than persist in the network for no reason. While these weights may not affect performance on the simulated training data, they may produce errors if the experimental data deviates slightly from the simulations.

The CNN is trained using the mean squared difference with a penalty on the size of the L_2 norm of the weights

$$L = \sum_{ijk} \|\tilde{P}_{ijk} - P_{ijk}\|^2 + \frac{1}{2}\lambda \sum_i W_i^2 \quad , \tag{6.6}$$

where $\tilde{P}$ is the output and P is the ground truth. To evaluate the above during training, we generate the ground truth probability maps from dot labels created from the center of positions of the input atomic structures. Each probability map is a 2D representation of the belief that a particular structure occurs at each pixel location. Ideally, if a single instance occurs in the image, a single peak should exist in the corresponding probability map; if multiple instances occur, there should be a peak corresponding to each instance. We generate individual probability maps P_k for each class label l_k. Let $(x_n^{(k)}, y_n^{(k)})$ be the ground truth position of the instance n with the class label $c_n = l_k$. The value at the position (x_{ij}, y_{ij}) of the i, j'th pixel is defined as the following superposition of Gaussians

$$P_{ijk} = \sum_n G\left(x_i, y_j | x_n^{(k)}, y_n^{(k)}, w^2\right) \quad k > 0 \quad , \tag{6.7}$$

where

$$G(x_i, y_j | x_0, y_0, \sigma^2) = \exp\left(-\frac{(x_{ij} - x_0)^2 + (y_{ij} - y_0)^2}{2\sigma^2}\right) \quad , \tag{6.8}$$

where σ^2 controls the width of the Gaussian. The ground truth for the background, P_0, is created to fulfill Eq. (6.5), hence

$$P_0 = 1 - \sum_k P_{ijk} \quad . \tag{6.9}$$

We will assume that the overlap between any pair of Gaussians is negligible. If this can not be assumed a new class combining the classes of overlapping Gaussians should be introduced. The width of the Gaussian influences the penalty of wrongly assigning a region of the inferred probability map to the background class. We found that if σ^2 were very small, the neural network would be very cautious: Any region in the images that was difficult to assign to a specific class (e.g. due to noise), would be assigned to the background class. We also found that a common local minima at training, was to assign everything to the background. Both issues could be mitigated by increasing σ^2. A better way to control this would be to directly equalize the weight of the different classes by modifying the loss function, however we have not explored this. We found that a width $\sigma^2 = 1$ $\mathring{A}$ (or ~ 10 pixels at $0.12\mathring{A}$/pixel) worked well for the cases we have considered.
The Adam algorithm is used to update the parameters during training [170], this is a stochastic gradient descent algorithm with an adaptive learning rate. We used a base learning rate of 10^{-4}, this is lower than the standard 10^{-3} for Adam. We found that a smaller learning rate improved convergence and helped avoid a bad local minima resulting in checkerboard artifacts during the upsampling [171].

6.3.3 Preprocessing

We normalize the input at both training and inference to ensure that the experimental images are within the training domain. An experimental image may locally contain structures, that are not part of the training set. To avoid skewing the normalization in the valid regions, we use local contrast normalization. An added benefit of this is to eliminate the need for simulating uneven illumination. The subtractive normalization operation for each component computes

$$I'_{ijk} = I_{ijk} - \frac{1}{N_f} \sum_{pqk} \frac{1}{2\pi\sigma_N^2} G(x_p, y_q | 0, 0, \sigma_N^2) I_{i+p,j+q,k} \quad , \tag{6.10}$$

where G is defined in Eq. (6.8). We want to preserve the information present in the relative intensities of the stack of images, hence the background is averaged across the feature dimension (k-dimension). To keep the loss of spatial information low, the effective radius of the weighting should be significantly larger than the size of any individual classified structure and its immediate neighborhood. To normalize the variance we compute the following divisive normalization

$$I''_{ijk} = \frac{I'_{ijk}}{\frac{1}{N_f}\sqrt{\sum_{pqk} \frac{1}{2\pi\sigma_N^2} G(x_p, y_q | 0, 0, \sigma_N^2)(I''_{i+p,j+q,k})^2}} \tag{6.11}$$

The denominator is the weighted standard deviation over a spatial neighborhood, averaged over the feature dimension to preserve relative variance locally.

6.3.4 Network architecture

The network architecture is based on fully convolutional networks for pixel-wise segmentation [154]. Several variants of fully convolutional networks have been proposed since their introduction and applied in different contexts. One such context is in segmentation of electron microscopy brain images. Chen et al. proposed a network combining multi-level upsampling layers from several scales for final segmentation [172]. Ronneberger et al. introduced skip connections for concatenating feature maps at different scales in their U-net architecture [173]. Quan et al. used additive skips and residual blocks as a solution to vanishing gradient in their FusionNet allowing for training of deeper neural networks [151]. Multi-level upsampling and skip connections accomplish the same task of combining global abstract information from deep coarse paths and local spatially resolved information from shallow paths. Our design is based on the FusionNet architecture. A pictoral description is given in Fig. 6.6.

The design is motivated by the need to capture information across multiple scales. Identifying individual peaks and valleys is essential for recognizing atomic structures, however estimating what an atomic structure should look like requires contextual information. Whether an atomic column appears dark or bright, relative to the background, can only be answered from the context, hence the network must have some mechanism to effectively process and combine features across scales. The architecture we use have a single pipeline with additive skip connections to preserve spatial information at each resolution. The network reaches its lowest resolution at one eighth the input resolution allowing smaller spatial filters to be applied that compare features across large areas. The shape of the network is symmetric, so for every layer present on the way down there is a corresponding layer going up.

Each step starts and ends with a convolutional layer, with a residual block in between [174]. The residual block consists of three convolutional layers and a skip connection that merges the input using elementwise addition. It is difficult to optimize the depth of a deep neural network. In theory deeper networks always have more predictive power, than a shallower network, however this is not true in practice due to the problem of vanishing gradients [175] and curse of dimensionality [161]. Essentially, if the network is too deep, it is difficult to properly back-propagate the errors. The use of residual blocks mitigate this problem in two ways: the error can more easily propagate backwards and in the worst case, the block can learn to be an identity mapping and thus not harm the performance.

Small convolutional kernels of size 3×3 are used everywhere except for the final prediction. We

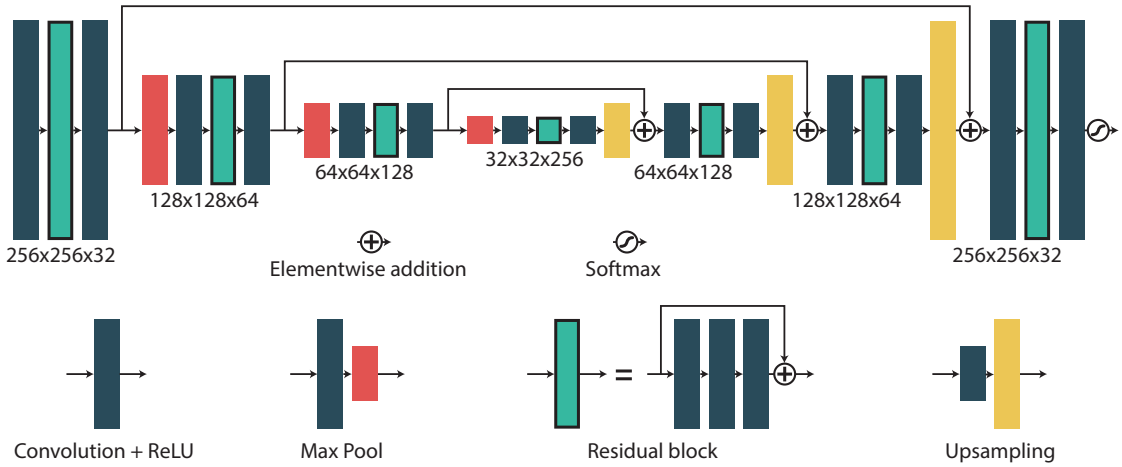


Figure 6.6: The architecture of the neural network. Information flows from left to right. The features are downsampled in an encoding path and upsampled through a decoding path, in addition several skip connections ensure that it is possible to retain fine spatial information.

use element-wise rectified linear activations after each layer that contains parameters to be trained

$$h(x) = \max\{0, x\} \quad . \tag{6.12}$$

These layers are then batch normalized [176]. Batch normalization regularizes the network by normalizing the data throughout the network, avoiding the problem the authors refers to as "internal covariate shift" [177]. The number of feature maps are doubled whenever downsampling is performed using max pooling. Each step in the upsampling path begins with a transpose convolutional layer, this is merged using elementwise addition with output from the same level of the downsampling path by a long skip connection. The transpose convolutional layers are initialized as bilinear interpolation and all other layers use random weight initialization. The final scoring consists of a convolutional layer with 1×1 kernel followed by a softmax non-linearity

$$\sigma(P_k) = \frac{\exp(P_k)}{\sum_{k=1}^{N_f} \exp(P_k)} \, . \tag{6.13}$$

The network is implemented with TensorFlow using the Python API [178]. TensorFlow is a recently open sourced deep learning framework developed at Google. All models are trained and tested on a single NVIDIA GTX 1080 Ti graphics card. Our models and code will be made publicly available.

6.3.5 Interpretating the probability maps

Each probability map will generally contain several peaks associated with different instances of the same class, hence we need to segment the image in order to associate a subset of pixels with

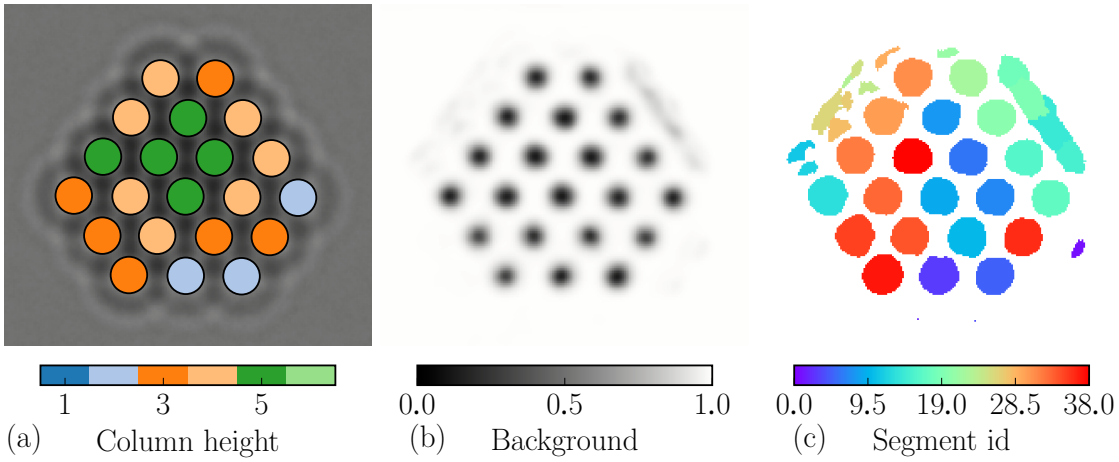


Figure 6.7: **(a)** The position of atomic columns for a small nanoparticle, color coded according to the corresponding height of the atomic column. **(b)** The background probability map predicted by the CNN given a series of images calculated from the nanoparticle shown in (a). **(c)** Watershed segmentation of background probability map, color coded according to segment id.

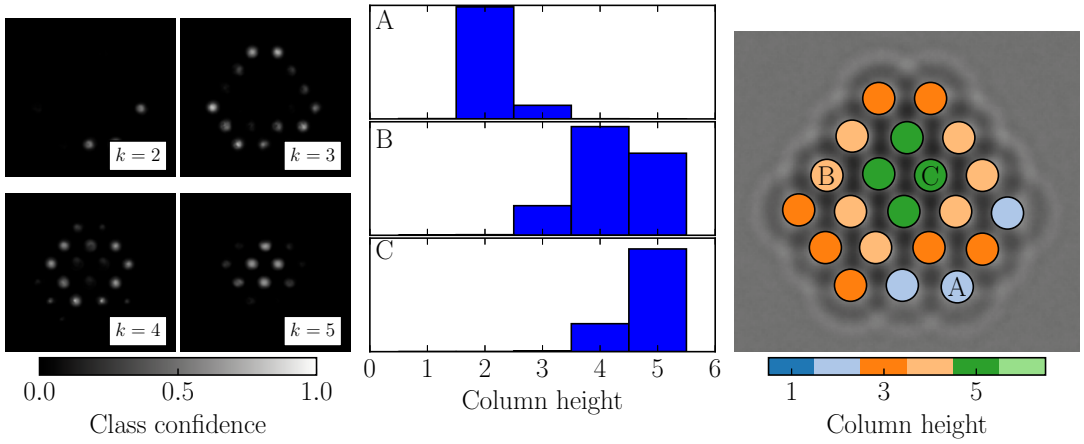


Figure 6.8: **(a)** Probability maps calculated by the CNN given a focal series. For this example the P_k probability map correspond to a column height of k. **(b)** Probability distributions, $p_n(c_k)$, predicted by our neural network for classes corresponding to different column heights, at the three different identified coordinates, A, B and C, shown in (c). **(c)** The most probable column height according to our neural Comparing with Fig. 6.7(a) our method made an error at the B coordinate, however the correct class is the second most likely according to our recognition model.

each instance. We start by finding all local minima in the background probability map. To avoid treatment of regions that are obviously in the background, we discard very shallow minima with a value above a tolerance $\epsilon = 0.995$. The local minima are used as seeds for basins created using the watershed principle with Meyers algorithm [179]. To avoid oversegmentation a hard upper limit for each basin is set to ϵ. Each resulting image segment $\mathcal{S}_n$ is assumed to correspond a candidate for the instance, n, of an atomic structure. We exemplify the segmentation by its application to determining the height (in number of atoms) of the atomic columns from a focal series of images. The example is further evaluated in section 6.4.3. The segmentation of the regressed background probability map is illustrated in Fig. 6.7.
The corresponding probability distribution over the class labels will be calculated as

$$p_n(c_k) = \frac{\sum_{i,j\in\mathcal{S}_n} P_{ijk}}{\sum_k \sum_{i,j\in\mathcal{S}_n} P_{ijk}} \quad k > 0 \quad , \tag{6.14}$$

where the sum is over all pixels belonging to the n'th image segment. The coordinate of the atomic structure is calculated as the center-of-mass of the image segment

$$(x_n, y_n) = \frac{\sum_k \sum_{i,j\in\mathcal{S}_n} (x_i, y_j) P_{ijk}}{\sum_k \sum_{i,j\in\mathcal{S}_n} P_{ijk}} \quad . \tag{6.15}$$

A number of false identifications of instances of atomic structures will generally be found in the presence of noise, hence we will eliminate any candidate instance, n, for which

$$\frac{\sum_{i,j\in\mathcal{S}_n} P_{ij0}}{\sum_k \sum_{i,j\in\mathcal{S}_n} P_{ijk}} < t \quad , \tag{6.16}$$

where t is a threshold to be chosen by the user. Some of the final probability distributions for the example introduced above are shown in Fig. 6.8.

6.3.6 Evaluation

The performance of our method is evaluated in terms of precision and recall. Precision can be interpreted as the probability that an identified positive identification is a true positive, while recall is the probability of identifying any given true positive. In terms of total number of true positives (TP), false positives (FP) and false negatives (FN), precision and recall is defined

$$\text{Precision} = \frac{\text{TP}}{\text{TP} + \text{FP}} \quad , \quad \text{Recall} = \frac{\text{TP}}{\text{TP} + \text{FN}} \tag{6.17}$$

There is natural tradeoff between precision and recall, increasing the threshold t will in general lower the number of positives, both false and true, this will increase the precision, but lower the recall. The relationship between precision, recall and threshold is expressed as a precision-recall curve.

6.3.7 Image simulation

A main challenge of our approach is to generate simulated training sets, that lets the neural network generalize from simulation to reality.
When building the input specimen models, we will not attempt to approximate in any way, a true distribution of the experimental specimens (as this is what we are investigating in the first place). We will instead generate randomized specimen models, the purpose of this is to provide enough simulated variability at training time such that, at inference the model is able to generalize to the actual specimen. The randomized specimen models still have respect the true structure to some extent. This requires case-specific solutions, which will be exemplified in the next sections.
The multislice algorithm have been shown to provide a quantitative match with experiment on an absolute contrast scale [180]. Nonetheless, exactly matching the conditions of a particular real microscope requires difficult measurements of the microscope aberrations. Hence, instead of attempting to simulate a particular microscope, we simulate an ensemble of random microscope conditions. We generate the contrast transfer function (CTF) at each training iteration. The effect of energy spread (i.e. temporal coherence) is included in the quasi-coherent approximation (see section 2.4.1). Temperature effects are included with the less accurate method of blurring the atomic potentials. To ensure that the CNN performs well across a range of sampling rates, the images are resampled randomly. Noise is included by modelling the finite electron dose rate as a Poisson process. Including the modulation transfer function of the detector is extremely important, due to its significant effect on the spectral properties of the noise. The MTF is parametrized according to Eq. (2.21) with randomized values for the parameters. We also slightly adjust the brightness, contrast and gamma of the images, to ensure a that the recognition model is robust. During training, the microscope conditions, including the CTF, MTF, dose and partial coherence, are randomized at every iteration. However, the exit wave functions for the structure models are calculated beforehand.

6.4 Experiments

6.4.1 Graphene

Graphene is considered a promising material, due to its unique electronic properties. The performance of graphene devices depends highly on the atomic structure of individual graphene sheets, including the presence of defects, strain and the configuration of edges. HRTEM have been extensively used to investigate graphene, and several automatic algorithms for extracting quantitative information have been proposed [181, 182]. We will compare our deep learning based technique to a recently introduced technique by Vestergaard et. al. [183]. Their technique combines geometric priors and image intensity in a coherent model to detect the hexagon centers. The centers are refined

parameters	lower bound	upper bound	distribution
defocus (Δf)	-200 Å	200 Å	uniform
3rd order spherical (C_s)	-20 µm	20 µm	uniform
5th order spherical (C_5)	0	5 mm	uniform
1st order astigmatism magnitude	0	100 Å	uniform
1st order astigmatism angle	0	2π	uniform
deflection	0	25 Å	uniform
focal spread	20 Å	40 Å	uniform
dose	10^1 $e^-/\text{Å}^2$	10^4 $e^-/\text{Å}^2$	exponential
c_1 (MTF)	0	0.1	uniform
c_2 (MTF)	0.4	0.6	uniform
c_3 (MTF)	2	3	uniform

Table 6.1: Randomized parameters for generating training examples of graphene for a 80 kV microscope. The focal spread and deflection are defined in Eq. (2.18) and (2.19), respectively.

with high accuracy using a model based on Markov Random Fields (MRFs). The experimental data used for validation in their original paper have been supplied to us by the authors.

Training

We want to train a neural network that is capable correctly localizing the atomic positions, even when they do not adhere to their ideal or almost ideal positions. Thus, the atomic models used to generate the training images cannot simply be ideal sheets of graphene. We might instead generate the atomic positions from a uniform random distribution, however this is incredibly inefficient, as only a tiny fraction of the training images relates to experimentally interesting situations.

We present an algorithm for generating distributions of atomic position that lie somewhere between the two extremes. The algorithm is explained in Fig. 6.9. While there is little theoretical reasoning behind the presented algorithm, the final structures have some nice properties. The "bond lengths" are reasonably narrowly distributed around their mean, and by correctly choosing the number of seed points, the mean bond length, can be chosen to coincide with that of ideal graphene. Additionally, the structures include a large number of distorted n-gons with $n = 5 - 8$. This same structure is also found experimentally along grain boundaries.

We generated 500 random structures with a size of 43.2×43.2 Å, or 360×360 pixels at a sampling rate of 0.12 Å/pixel. All the simulations were done at an acceleration voltage of 80 kV.

Fig. 6.10 shows the training loss as a function of the number training samples. The running standard deviation shows that the variability in the performance is huge. This is a consequence of

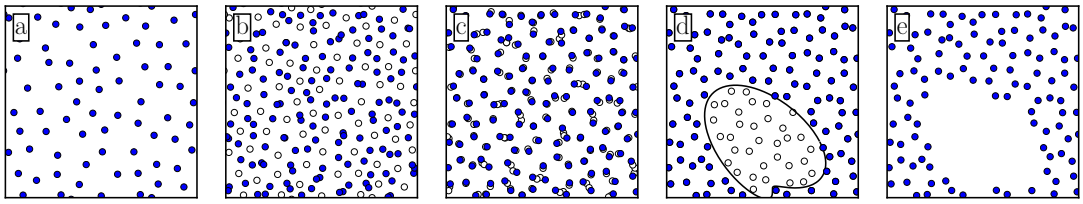


Figure 6.9: Procedure for generating random training for graphene. **(a)** A square in 2d space is filled with randomly distributed seed points under the constraint of a minimum separation in terms of euclidean distance (i.e. a Poisson disc distribution [184]). **(b)** Next, the Voronoi diagram of the seed points are generated, and the vertices of the diagram are used as atomic positions while discarding the original points. **(c)** To avoid overlap the positions of the Voronoi vertices are relaxed using Lloyd's algorithm [185]. **(d)** Lastly, zero to four holes of varying size and shape are introduced in the structure.

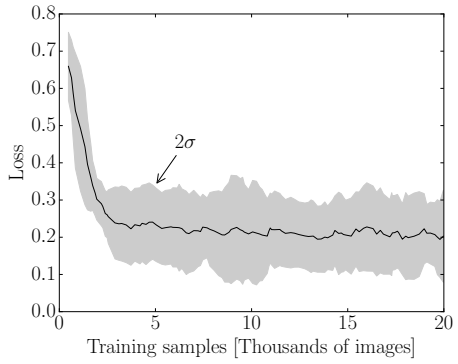


Figure 6.10: The running average loss and standard deviation as a function of the number of training samples. The running average and standard deviation is calculated over $\sim$ 600 training samples using a boxcar window. The training is mostly converged after the first 2500 training iterations.

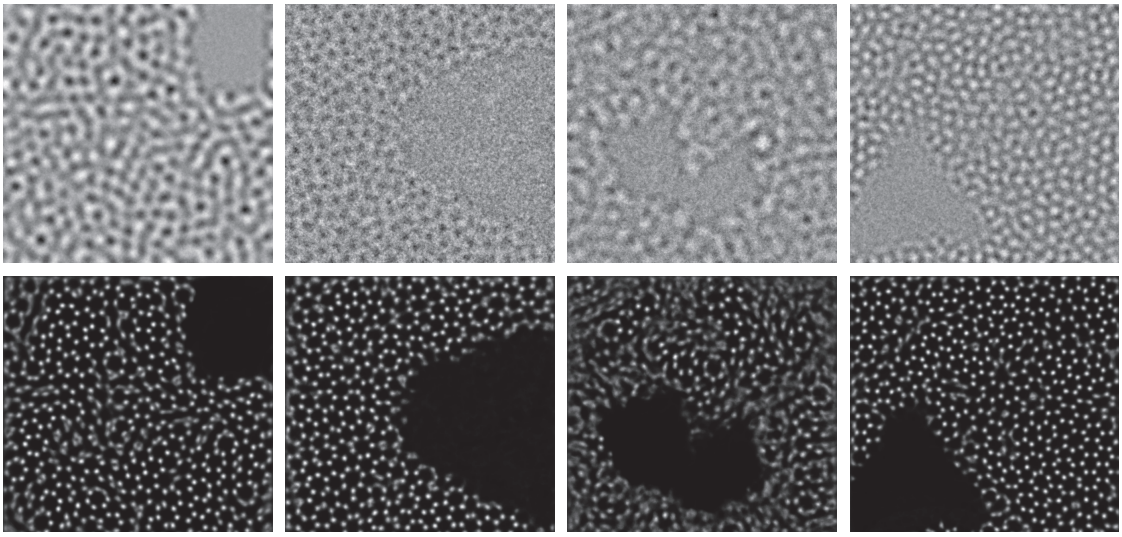


Figure 6.11: Four random training images and the corresponding output of the trained neural network. A significant number of the atomic positions are difficult to interpret, either due to insufficient imaging conditions or due to the difficult structure of training examples.

randomly generating the microscope parameters, a number of the training examples are impossible to interpret, due to a combination of noise and the relationship between the spherical aberration and defocus.

Fig. 6.11 shows training examples and corresponding output after the neural network is trained. Clearly, large regions of the probability maps are far from the ground truth. However, they are probably as good can be reasonably expected given the difficulty of the training samples. The neural network performs much better, when it applied to more reasonable experimental images, as shown in Fig. 6.12.

Application to experimental images

The experimental graphene images were measured using a FEI Titan 80-300 Environmental TEM equipped with a monochromator at the electron gun and spherical aberration (C_s) corrector at the objective lens. The acceleration voltage of the microscope was 80 kV, which is below the knock-on threshold for carbon atoms in pristine graphene. The electron beam energy spread were below 0.3 eV, while the C_s-corrector was aligned to minimize the spherical aberration. The images were recorded using a Gatan US1000 CCD camera with an exposure time of 1 s.

Fig. 6.13(a) shows a HRTEM image of pristine graphene. The negative C_s imaging (with positive defocus) results in images where the carbon atoms are bright spots, with the centers of the hexagons

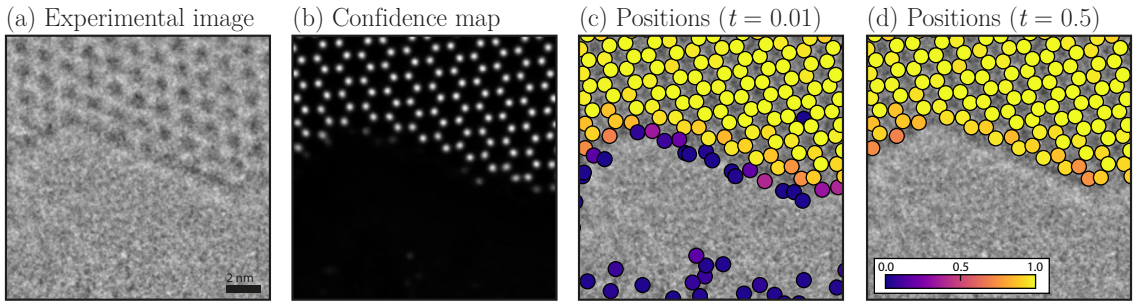


Figure 6.12: A neural network trained exclusively on simulated data is capable of generalizing to experimental images. **(a)** Single suspended graphene sheet with a hole formed under the influence of the electron beam. **(b)** The regressed probability distribution predicted by the neural network for the image in (a). (**(c)**) The local peak positions of the probability map is overlayed on the image. The peaks are color-coded according to their maximum value. Peaks with a maximum value less than 0.01 are excluded. (**d**) A higher tolerance for exclusion is used to remove peaks with a maximum value less than 0.5.

appearing dark. The output of the CNN is shown in 6.13(b). The neural network detects all atomic positions. This is accomplished without imposing any hard geometric constraints. Additionally, the neural network automatically recognizes that the atoms appear bright, which is only the case for about half of the training images. Fig. 6.13(c) shows the strain given the interpreted positions from the neural network. The strain is calculated using structural template matching with the first two neighbour shells (i.e. the 9 nearest neighbours).

Fig. 6.13 shows the strain calculated from a sheet of defective graphene. The strain is small in the regions above and below the hole, however a significant strain is detected to the left and right.

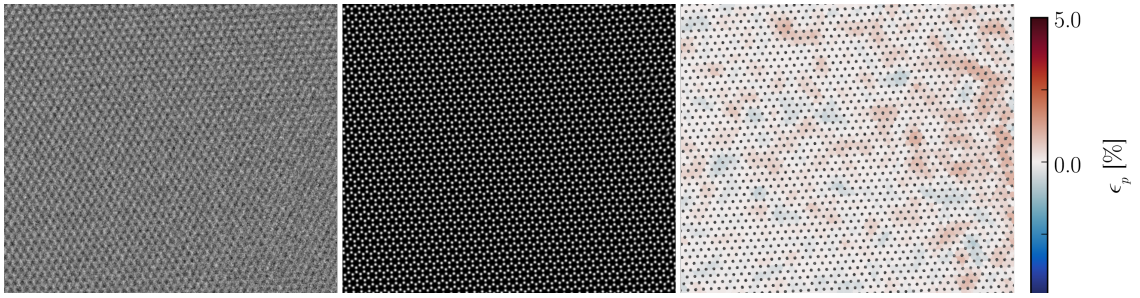


Figure 6.13: **(a)** Experimental image of a pristine sheet of graphene. (b) The output of our neural network given the image in (a). **(c)** The planar strain given the positions interpreted from (b).

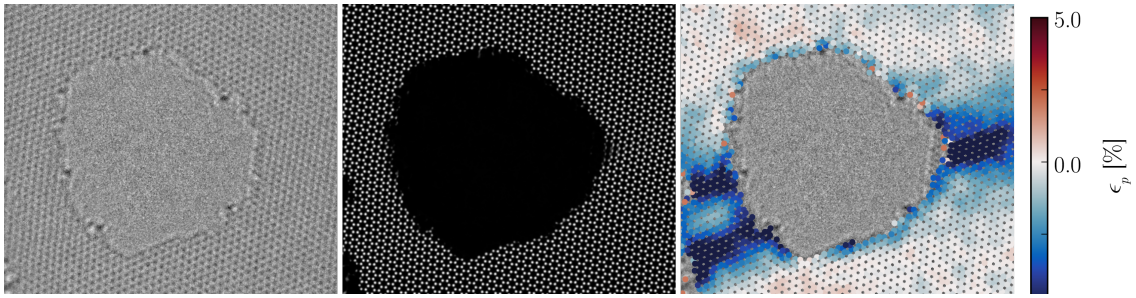


Figure 6.14: **(a)** Experimental image of a pristine sheet of graphene. (b) The output of our neural network given the image in (a). **(c)** The planar strain given the positions interpreted from (b).

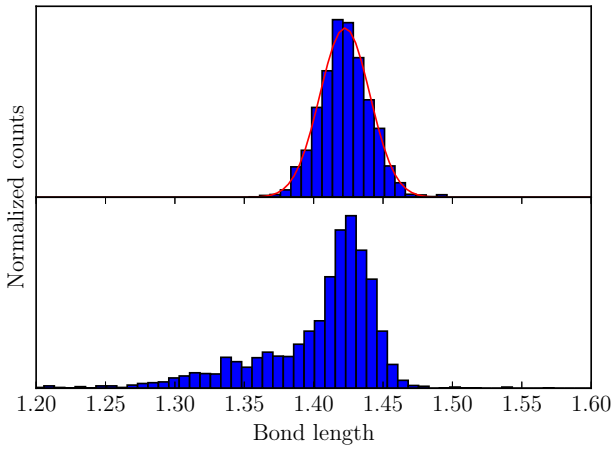


Figure 6.15: Histograms of detected bond lengths **(a)** from Fig. 6.13 and **(b)** from Fig. 6.14. The bond lengths are binned by 0.05 Åsteps and the counts are normalized for easier comparison. The histogram in (a) follows a normal distribution with a mean of 1.42 Å and a standard deviation of 0.18 Å. The histogram in (b) have a tail towards shorter bonds lengths.

This could be explained by out-plane ripples in the graphene sheet, resulting in a projection of the structure that appears strained. If the graphene sheets are assumed to be flat, we find that the maximum strain along any direction is ~ 8 %. This is qualitatively the same strain distribution, that was found by the method of Vestergaard et. al. We also measure a small compressive strain along the edge of the hole, which was not found in their study.
Due to the inherent small signal-to-noise ratio in images of graphene, and the continuously induced damage to the edge structure by the electron beam. Both ours and Vestergaards method have difficulty completely resolving the edge of the hole. Generally, our method finds that the hole is zig-zag terminated, which is in agreement with previous findings. However, we also find a few positions that are clearly ridiculous from an energetic viewpoint. This is not unexpected given that our method, does not make use of any hard geometric constraints or any prior knowledge about the real structure graphene.
Fig. 6.15 shows histograms of the bond lengths from the images in Fig. 6.13 and 6.14. The histogram can be compared to Fig. 3 in an article by Jens Kling et. al. which is based on the same data [186]. We do not expect any strain in a pristine area of graphene, hence we should expect to find a narrow distribution of bond lenghs. We find that the bond lengths are normally distributed around the mean with a standard deviation of 0.18 Å. This standard deviation is comparable or smaller to what was found using the method of Vetergaard et. al. [186], hence demonstrating that the positions determined with our method are resistant to noise.

Precision and recall

To quantitatively characterize the performance of our method, we have simulated realistic polycrystalline graphene sheets. The initial graphene models are generated using the method of Ophus et al. [187]. To equilibrate the samples, we first anneal them at 2000 K for 50 ps after which the system is quenched to 0 K during a 20 ps run. The simulations are carried out with a NPT ensemble [64], a Langevin thermostat and the reactive bond order potential by Stuart et al. [188]. The final models exhibits realistic defects along the grain boundary, long range strain and out-of-plane ripples. We simulate 16 sheets with the dimensions 122.4×122.4 Å, thus resulting in 1024×1024 pixel images at 0.12 Å/pixel.
The images were simulated to match the experimental conditions in the previous section as closely as possible, hence, the performance evaluation should be seen in this context. An improvement of the image conditions would also improve the below performance characteristics. The images were simulated with the following parameters: spherical aberration $C_s = -12$ µm, defocus $\delta = 84$ nm, focal spread $\delta = 35$ Å, deflection $\sigma = 25$ Å and a MTF parametrized by $c_1 = 1$, $c_2 = 0.48$ and $c_3 = 2.45$.
Since atoms along the grain boundary are significantly more difficult to recognize than the ideal lattice, we want to separate the statistics. The atoms are categorized as being locally part of the

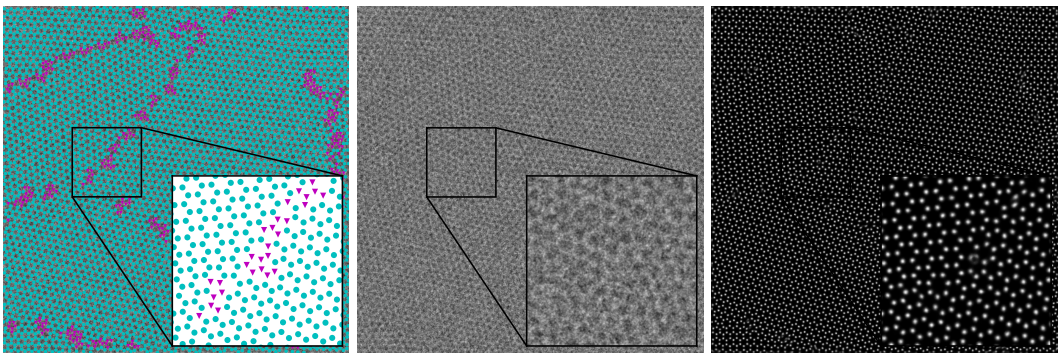


Figure 6.16: Simulation of a poly crystalline sheet of graphene. **(a)** The atomic positions of the graphene sheet is colorcoded according to whether the atom is characterized as part of an ideal hexagonal lattice (green) or not (magenta). **(b)** Simulated HRTEM image given the simulated graphene sheet. **(c)** Output of out neural network given the image in (a).

ideal hexagonal lattice or a part of the grain boundary, see Fig. 6.16. The categorization is based on the RMSD of the 9 nearest neighbours with respect to the ideal lattice. The following performance evaluation is based on $\sim 10^5$ ideal atomic positions and $\sim 10^4$ grain boundary positions.

Fig. 6.17 shows precision-recall curves for different electron doses, obtained by changing the threshold t in Eq. (6.16). At the lowest dose, 18 $e^-/\text{Å}^2$, the results are essentially as bad a random guessing. This is already improved at 10^2 $e^-/\text{Å}^2$. At a dose of 5.6 $e^-/\text{Å}^2$, for the ideal atomic positions, it is possible to obtain a recall of ~ 0.95 while maintaining an almost perfect precision. This is not the case for the grain boundary which requires at least 3.2 $e^-/\text{Å}^2$ to be properly resolved. At a dose 10^5 $e^-/\text{Å}^2$ or above, it is possible to obtain perfect precision and recall for the ideal positions, however even at infinite dose there are around 1 % of the grain boundary positions that cannot be recalled. These positions generally correspond to very distorted hexagons, due to a large local inclination of the graphene sheet. This shape is not well-represented in the training set, hence it may be possible to find these positions by creating a training set with this in mind. However, it may also be the case that the simulated imaging conditions are simply insufficient, and indeed it is possible to obtain perfect recall and precision along the grain boundary by lowering focal spread and deflection.

Fig. 6.18(a,b) shows the precision and recall as a function of the threshold, t, for interpreting a peak as a positive identification. This shows that our method is fairly insensitive to the choice of threshold, unless the structure is very difficult to resolve due to noise.

Fig. 6.19(a) shows the precision and recall as a function of dose. The hexagonal structure is essentially perfectly resolved at a dose of 10^3 $e^-/\text{Å}$, while it requires a dose more than ten times larger to resolve the grain boundary.

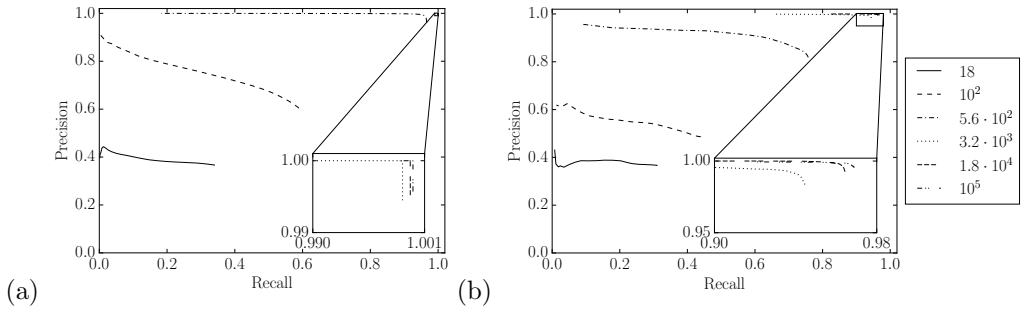


Figure 6.17: Precision-recall curves for simulated images of poly crystalline graphene for **(a)** ideal positions and **(b)** positions along the grain boundary. We show curves for size different electron doses as shown in the legend in $e^-/\text{Å}^2$.

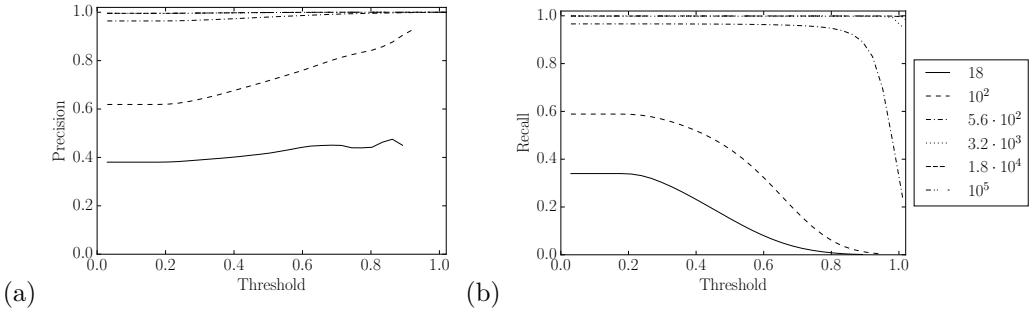


Figure 6.18: Precision and recall as a function of the threshold used for discriminating between whether a peak in the probability map should be recognized as an atomic position.

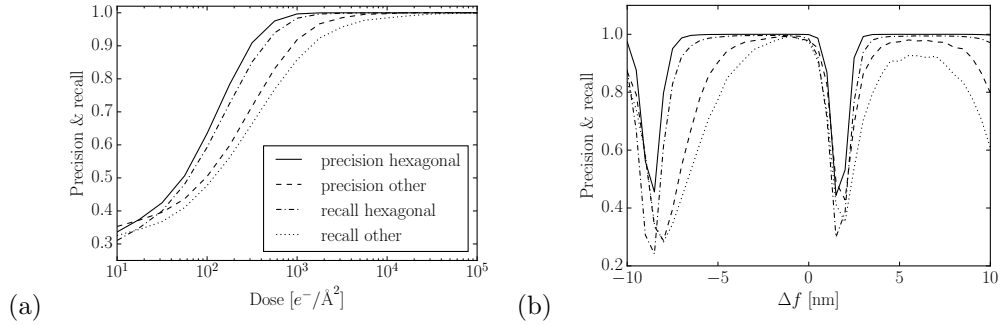


Figure 6.19: Precision and recall at a threshold of 0.5 as a function of (a) dose and (b) defocus. The dose was 2×10^3 $e^-/\text{Å}$.

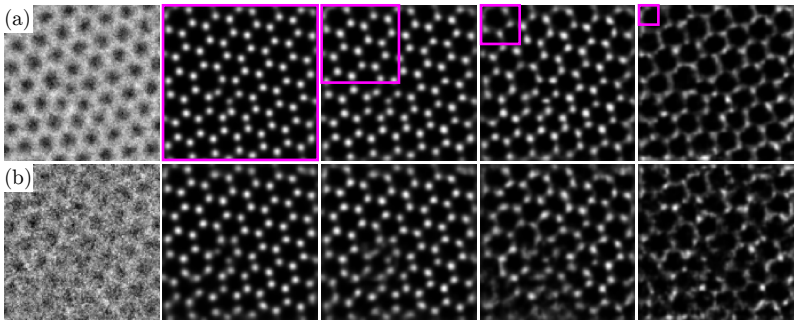


Figure 6.20: The first panel in each row shows the full input image to the CNN, before applying the sliding window. The output of the network for progressively smaller sliding windows is shown in the next panels. The sizes of the sliding windows are indicated by the red square, the size is cut in half, for every panel, from left to right. The rows show results for different electron doses: **(a)** 5×10^4 e^-/Å, **(b)** 5×10^3/Å.

The CNN is in principle trained to perform equally well at all defocus between $\Delta f = -20$ nm to $\Delta f = 20$ nm. However, at a finite electron dose, the signal-to-noise ratio is very dependent on the defocus. HRTEM images of graphene relies entirely on phase contrast, hence the performance dips drastically at the defocus, where the contrast is inversed. This is shown in Fig. 6.19(b). Interestingly, the neural network actually seems to perform slightly better in a limited region, when the atomic positions are transferred as dark spots and the centers of the hexagonal rings are bright.

Occlusion analysis

To investigate how much non-local information the neural network is utilizing, we applied the neural network to partially occluded images, i.e. images where some of the pixel values have been set to zero. The results are shown in Fig. 6.20. Each pixel in the probability map is obtained from an evaluation on the image, where everything outside a box-shaped sliding window is set to zero. The final probability maps are then a compound of an evaluation of the neural network for each pixel. The process is similar to a technique used for localization in networks for whole image classification [189], however in this case the CNN is allowed to see what is inside the sliding window, while everything outside the window is blacked out.

Given the full 15×15 Å image, the CNN is only slightly perturbed by increasing noise. However, as size of the sliding window is decreased, the performance is impacted much more by noise. Hence, it seems that the neural network relies more on non-local information, when the signal-to-noise ratio is low. At low noise, the neural network performs reasonable until the sliding window is smaller than individual hexagons in the graphene structure.

6.4.2 Nanoparticles

Nanoparticle morphology, which decides the number of active sites and catalytic efficiency, is strongly determined by the gases involved in synthesis, treatment, and reaction. Several investigations have been performed to understand the morphological response of gold to the gas environment [190, 191, 192]. Using in situ environmental transmission electron microscopy (ETEM), Uchiyama et al. observed that the overall shape of gold nanoparticles supported on ceria, behaves systematically depending on the atmosphere [193]. They found that, in an oxygen environment the nanoparticles became more rounded due to a fluctuation between {111}, {100}, and {110} facets with time.

We also observed a Au/CeO_2 catalyst in vacuum and oxygen atmospheres by means of ETEM, and were able to resolve the individual atomic columns in the $\langle 110 \rangle$ zone axis. We saw that in the oxygen atmosphere, the columns at the facets could be remarkably mobile. The statistical nature of the surface migration of atoms, requires a significant amount of data to quantify, hence being able to do this automatically is highly desirable. This is especially true, if we want to be able to separate the influence on the dynamics of an individual parameter, such as gas species, partial pressure, support, electron dose and temperature. In this section, we look at the application of our technique to analysing time-resolved HRTEM video of individual nanoparticles.

Training

The neural network should perform well for nanoparticles with both faceted and rough surfaces. To ensure this, we create a training set with both types of surfaces. We start by generating a seed crystal by creating a random number of atomic planes in the low index directions (i.e. the $\langle 100 \rangle$-, $\langle 110 \rangle$- and $\langle 111 \rangle$-direction). To obtain reasonable particles with imperfect surfaces, we add additional atoms to the seed crystal with a probability determined by the coordination number of the candidate site. Let i denote a valid crystal position with respect to the seed crystal with the coordination number N_i. We assume that $N_i > 0$, then the probability that the next atom will be appended to this candidate position is given by

$$p(n_i) = \frac{\exp(N_i/T)}{\sum_{N_i} \exp(N_i/T)} \quad , \tag{6.18}$$

where the sum is over all positions for which $N_i > 0$. The constant T is a fictitious temperature that can be used to tune the roughness of the generated nanoparticles. For $T \to \infty$ each candidate position is equally likely, whereas for $T \to 0$ only the highest coordinated positions are probable. A random number of atoms, scaled by the size of the seed crystal, are appended for each training structure. The particles are rotated either into the $\langle 100 \rangle$- or $\langle 111 \rangle$-axis, then they are rotated a random amount with a rotation axis perpendicular to the zone axis. Lastly, each particle is tilted 0-3° away from the ideal low index zone axis. Every particle is placed in a 50×50 Åunit cell,

and displaced by a random amount. The simulations of the exit waves are done at an acceleration voltage of 300 kV.

Precision and recall

To quantify the performance of our method on nanoparticles, we simulate validation images from a new set atomic models. The shapes of the nanoparticles are based on data from a theoretical on study on nanoparticle morphology, and represents a realistic distribution of particle shapes at room temperature [194]. The size of the nanoparticles used for validation were within the range of those used for training and were rotated in the same manner. The nanoparticles at training were without any relaxations, this may negatively influence the performance, when encountering experimental images where this will not be the case. Hence, we have created validation images of nanoparticles with the ideal crystal structure, and the same nanoparticles relaxed using the COMB potential for gold.

An example application of the CNN for varying electron dose is shown in Fig. 6.22. For the simulated image conditions, it is not possible to resolve atomic columns at the surface with a height of $\sim 1-3$ atoms. As the dose is increased these atomic columns are found. We find that the neural network generally errs on the side of safety, in the sense that the number of false negatives are greater than the number of false positives.

Fig. 6.23(a) shows the precision and recall of our method as a function of the electron dose, for images of both the ideal and relaxed models. We see that including relaxations have little impact on the performance of our method. It should be noted that the surface relaxations are fairly minor structural changes. We expect that the performance of our method, will be impacted significantly when encountering structures that deviates strongly from the training set.

There are two failure modes of our method for detecting a particular atomic column, the signal-to-noise ratio of the column can be too low or the column can be too tilted. The number of failures due to tilt are generally not lowered much by increasing the dose, hence the precision and recall does not converge to one at very high dose. The number of errors due to tilt increases with the column height, since the effective footprint, with respect to the viewing direction increases. Contrarily, the number of errors due to noise decrease with the column height, since the image intensity of the atomic column increases. This is illustrated in Fig. 6.23(b) where we show the recall for the validation set with respect to the column height.

To better understand how the CNN interprets the images of nanoparticles, we have considered the detection of an atomic column at two different placements; placed at a valid lattice position and isolated from the nanoparticle. The results are shown in Fig. 6.24. We see that the CNN only detects the atomic column, when it is placed at the valid lattice position. This can be seen as both an advantage and a potential problem of our method. It is an advantage because, it is unlikely that, the same performance could be obtained, at low dose, without taking the geometry of the

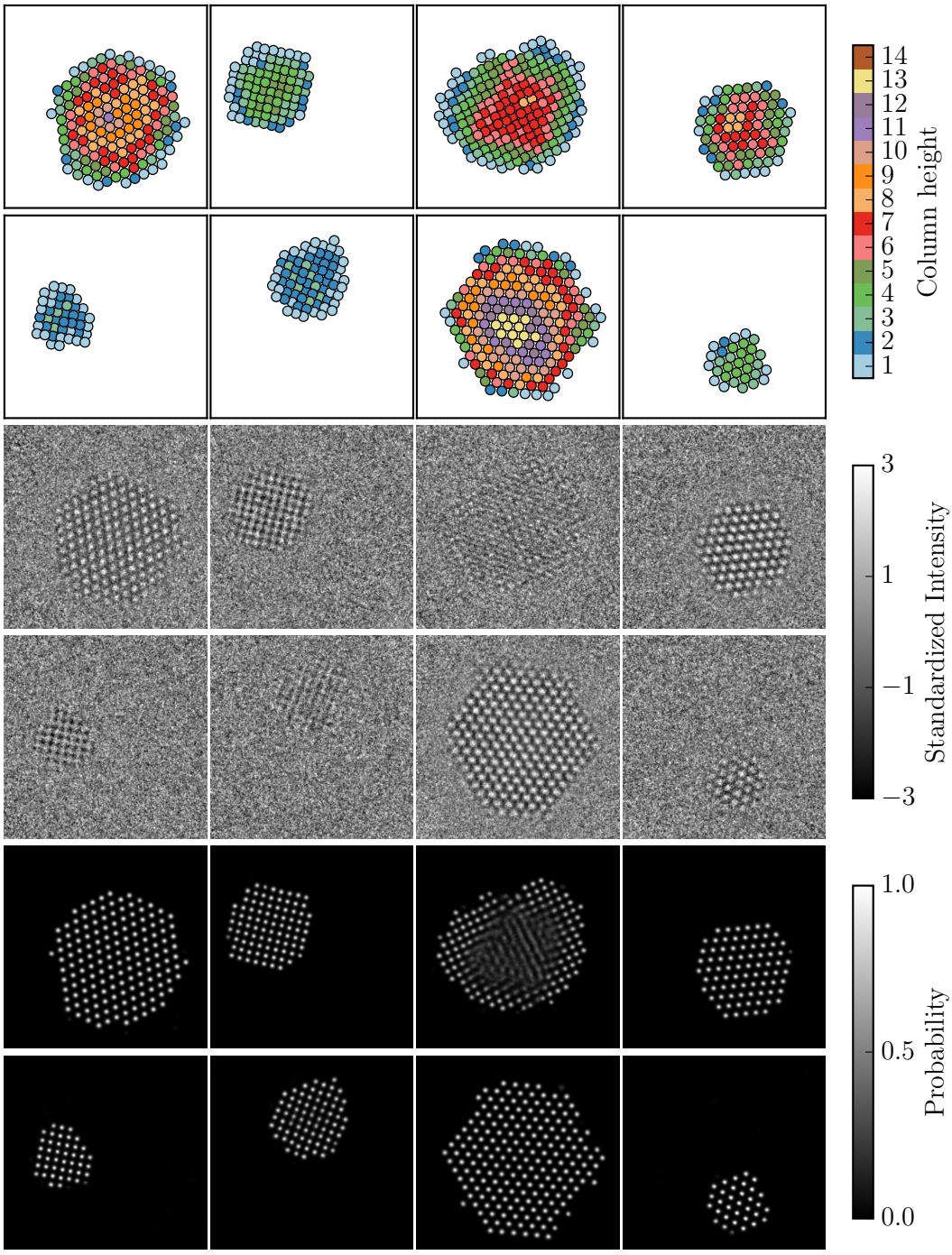


Figure 6.21: **(a)** Examples of nanoparticles generated using the algorithm we have proposed. The height of the atomic columns of the nanoparticles are indicated with a colorcoding. **(b)** Resulting images given the atomic models in (a). **(c)** Output from the CNN given the images in (b).

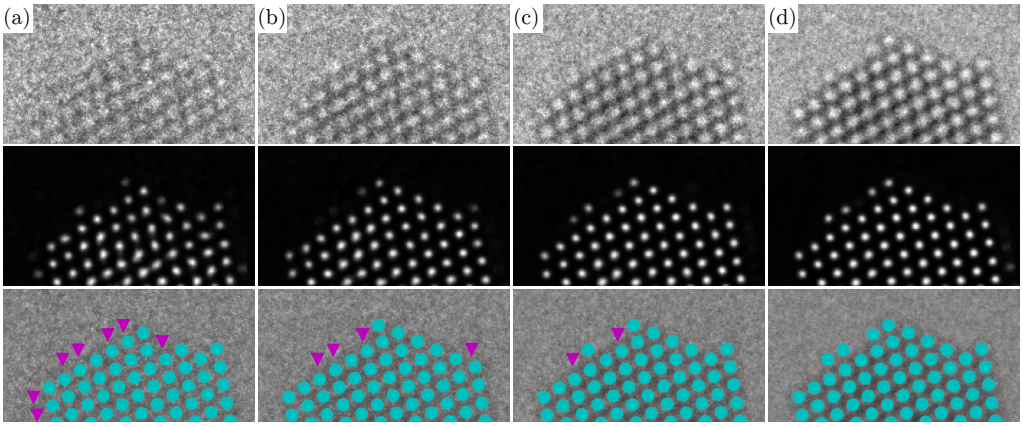


Figure 6.22: From top to bottom. The first row shows simulated electron microscopy images, the second row show the corresponding prediction of the CNN and the last row shows the interpreted positions. A magenta triangle indicate a false negative and a cyan circle indicate a true positive. There are no false positives in this example. The simulated electron doses were (a) 10^2 $e^-/\text{Å}^2$, (b) 2.2×10^2 (c) 464 $e^-/\text{Å}^2$ and (d) 10^3 $e^-/\text{Å}^2$

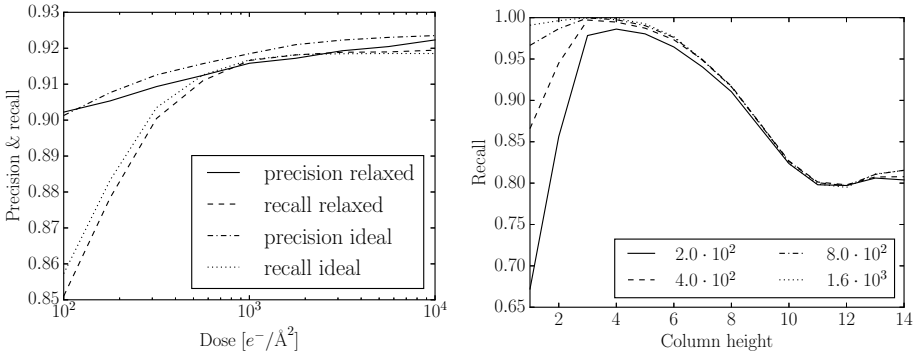


Figure 6.23: **(a)** The precision and recall as a function of dose, for both relaxed and ideal nanoparticles from the validation set. The CNN is trained on models where the atoms are placed on ideal crystal sites. We see that the performance is only decreased slightly when it is applied to models that include realistic surface relaxations. **(b)** Recall at different doses as a function of the column height for the nanoparticles in the validation set. The dose given in the legend is in units of $e^-/\text{Å}^2$.

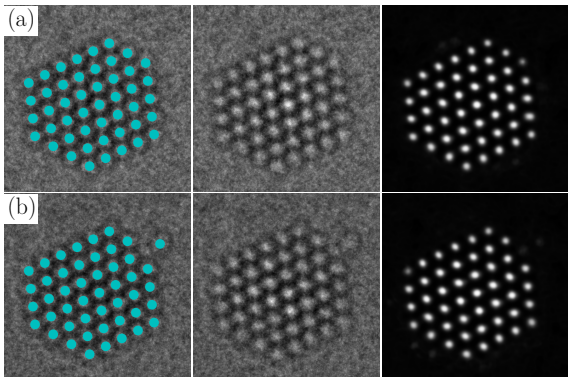


Figure 6.24: The top row shows a simulated image and the resulting output of our neural network for a nanoparticle where all the atoms adhere to the ideal lattice. All the atomic columns are easily found by the CNN. In the bottom row, one of the atomic columns have been displaced. At this level of noise, the CNN does not find the displaced column.

placement into account. However, it can also be a potential issue, since it is difficult to understand how the CNN makes decisions. Using this method, one should be very aware of the potential biases build into the training set. It should be noted that at a higher dose, the CNN detects the atomic column regardless of its placement relative to the nanoparticle.

Application to experimental images

Fig. 6.25 shows a time series of a gold nanoparticle in an oxygen environment. The series was measured in an environmental TEM by Pei Liu. We also show the corresponding output of the CNN. It can be observed that the $\langle 111 \rangle$ facet opposite to the interface changes dynamically. This is in agreement with the output of the CNN. We find that the CNN generally makes identical predictions to human experts, although there are some disagreements at this level of noise. However, we found that the level of disagreement between the CNN and a human expert, were generally not larger than the disagreement between two human experts. We should note that the results shown in Fig. 6.22, indicate that it is not possible to resolve single atoms at this low dose.

In Fig. 6.26 we compare the surface dynamics of the same nanoparticle in vacuum to a particle exposed to oxygen. Clearly, according to our analysis, there is a significant effect on the surface dynamics of the nanoparticle exposed to oxygen. In agreement with previous experimental findings [193]. Interestingly, this only happens on the $\{111\}$-facet opposite the interface. To investigate whether this is just a curious case, requires an investigation on a larger selection of nanoparticles, however the method presented here could make such an investigation easier.

In Fig. 6.27, we look the strain in the nanoparticle in both vacuum and oxygen. The positions are

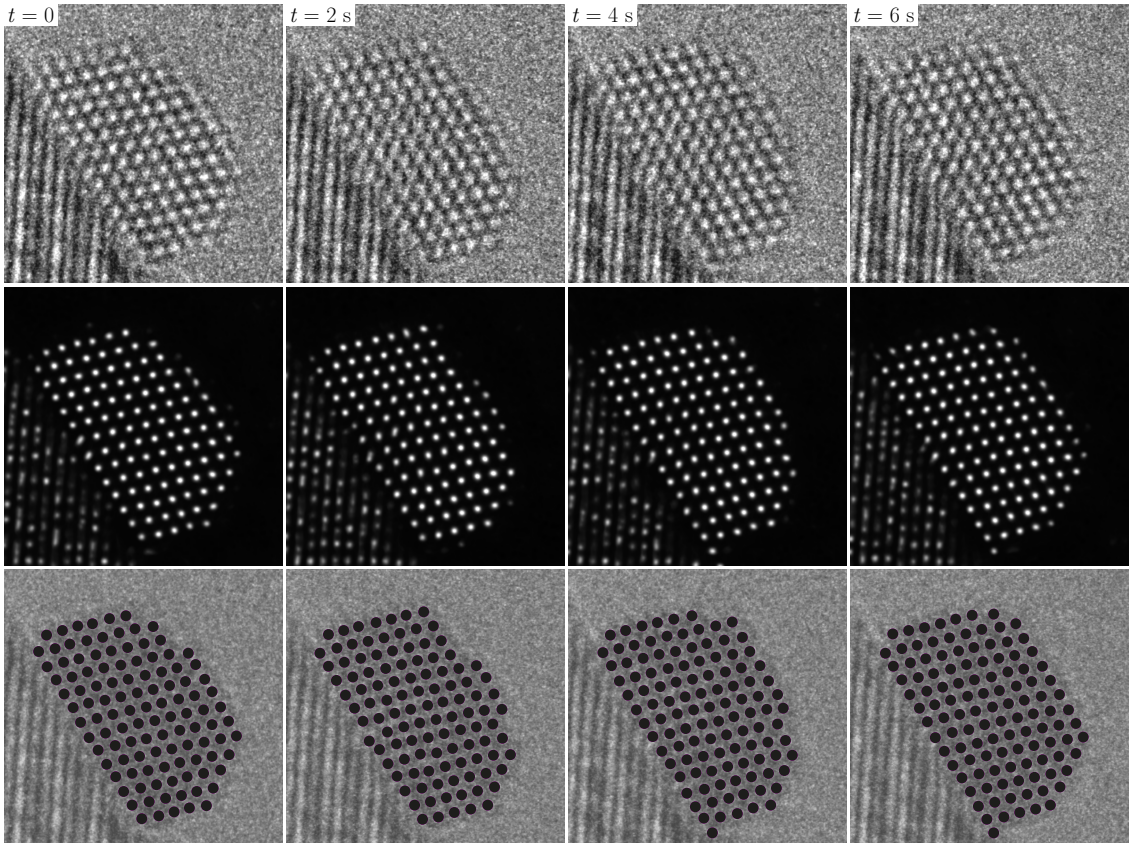


Figure 6.25: Time series of a gold nanoparticle on a ceria subtrate. The dose rate was 1.56×10^3 $e^-/\text{Å}^2/s$ with an exposure time of 0.2 s (dose per image 3.1×10^2 $e^-/\text{Å}^2$). The experimental images were measured using a FEI Titan 80-300 Environmental TEM operated at 300 kV. The oxygen pressure was 4.5 Pa. The second row shows corresponding output of the CNN and the last row shows the interpreted positions. Since the neural network is not trained on CeO2, the predictions for the substrate are unusable, hence these were removed manually.

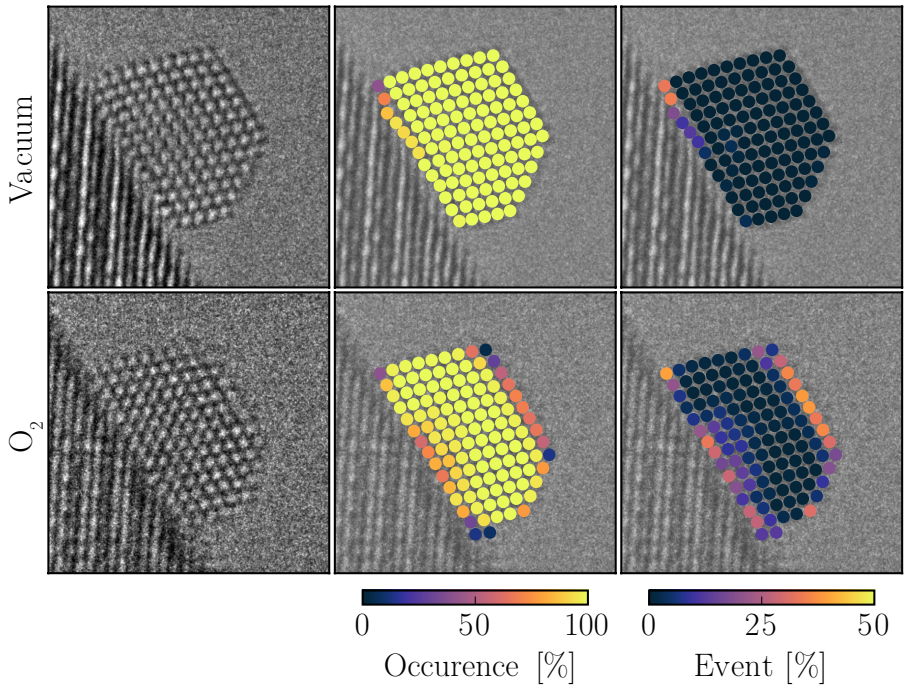


Figure 6.26: The surface dynamics of gold nanoparticles is influenced by the gaseous atmosphere. The occurrence is the percentage of frames where the neural network identified an atomic column at a possible site. The event percentage is the percentage of frames where a site was previously occupied, but is unoccupied in the frame immediately after, or conversely unoccupied then occupied immediately afterwards. The events along the interface are due to the difficulty of detecting the atoms along the interface and due to tilt. The maps are based on 300 and 320 frames, for the vacuum and oxygen respectively.

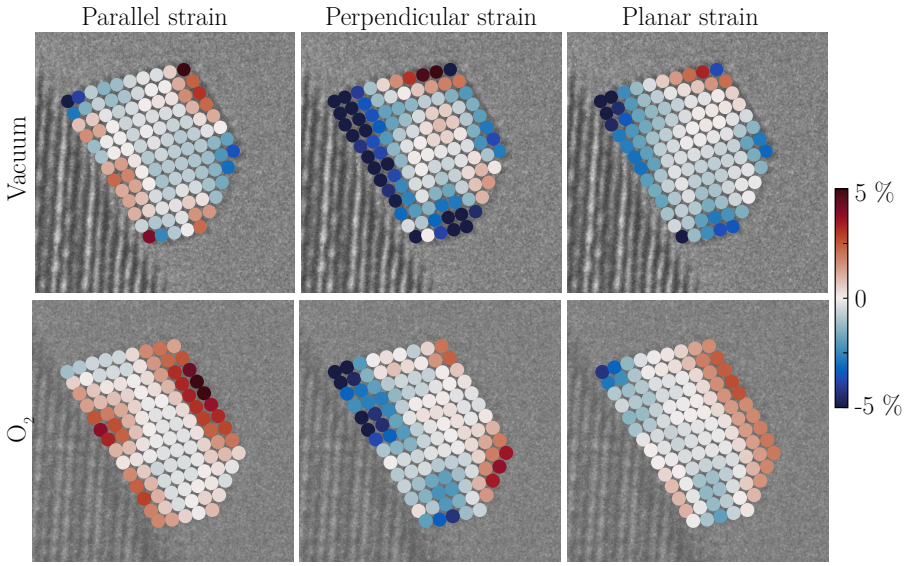


Figure 6.27: The strain in the direction parallel and perpendicular to the interface and the planar strain. The strain is calculated using structural template matching with the size nearest neighbours, and partial matching for the surface atoms.

determined using the average of the output of the CNN for all the frames. The main thing to note is the outward relaxation of the {111}-facet opposite the interface, i.e. the facet that was found to rearrange. This strain could indicate adsorbtion of oxygen on the surface. Theoretical studies have found that adsorbates, can induce changes to surface relaxations. A DFT study on the adsorbtion of atomic oxygen at the (111)-facet of gold, found that the surface relaxation of the top layer was 8.7 % outward at a coverage of 0.5ML, while a coverage of 1ML an *inward* relaxation of 3.88 % was predicted [130].

6.4.3 Atom-counting in nanoparticles

In the previous sections, we only applied our method to the detection of a single class of structure. To investigate how far the method can be pushed, we have applied it to 3D reconstruction of nanoparticles. The results are preliminary, since we have only investigated the performance on simulated data.

Different techniques to recover 3D structure have been suggested in atomic resolution electron microscopy. One of the most promising techniques is electron tomography [195, 196, 197], however this have several limitations, electron tomography requires multiple exposures, which is not always feasible in practice. For example, in situ experiments or the characterization of beam sensitive

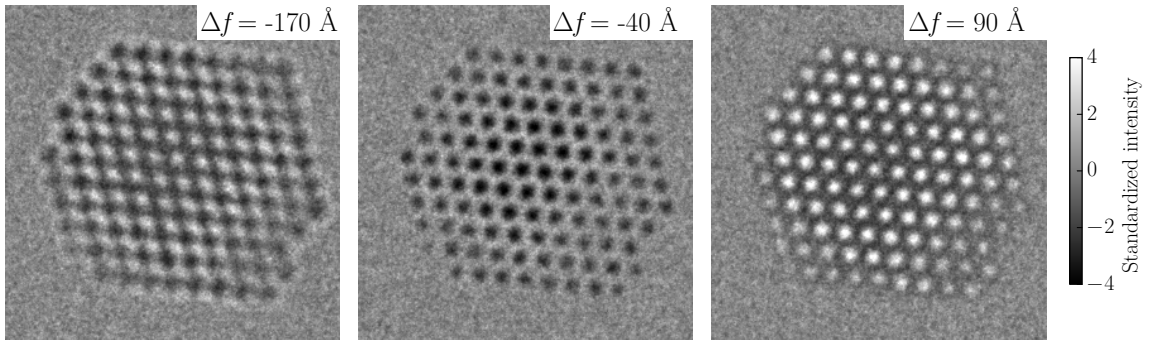


Figure 6.28: Example of the input images to our CNN-based atom-counting technique for a dose 10^3 $e^-/\text{Å}^2$ per image.

materials.

Alternatives to electron tomography for a 3D reconstruction at the atomic scale have been proposed where the 3D atomic structure is reconstructed from a single projection. LeBeau et al. [198] and Jones et al. [199] compared simulated atom column scattering cross-sections with normalised experimentally measured atom column intensities in order to count the number of atoms in an atomic column from STEM images. Information about the column heights, combined with prior knowledge of crystal structure and energy minimization, can be used to reconstruct the 3d structure. Recently, this method was verified using tomography [200].

In TEM, it has been shown that peak intensities, can be used to count the number of atoms in a projected atomic column using negative spherical aberration imaging [201]. However, peak intensities are sensitive to a small tilts of the sample, and the peak intensity is not in general a monotonic function of the peak height. This was discussed in a recent comparison of STEM and TEM based atom counting by Gonnissen et al. [202]. This study also demonstrated, in theory, that a model based on all pixel values of a TEM image, not just the peak intensity, under optimal experimental conditions, obtain a better accuracy for atom counting than STEM based techniques. However, no practical demonstration of such a model exists currently. The model would have to accurately encode the dependence of all image pixel intensities on the sample thickness, and be stable to small variations in parameters such as crystal tilt. Using a CNN could be a possible way of creating such a model.

The number of atomic columns can also be counted using the reconstructed exit wave from a TEM focal series [203] or from inline electron holography [204, 205]. This method uses the column approximation for TEM images [70, 71].

In our approach, we recast the 3D reconstruction as a classification problem, where each atomic column is classified according to height. Similarly to the methods based on focal series reconstruc-

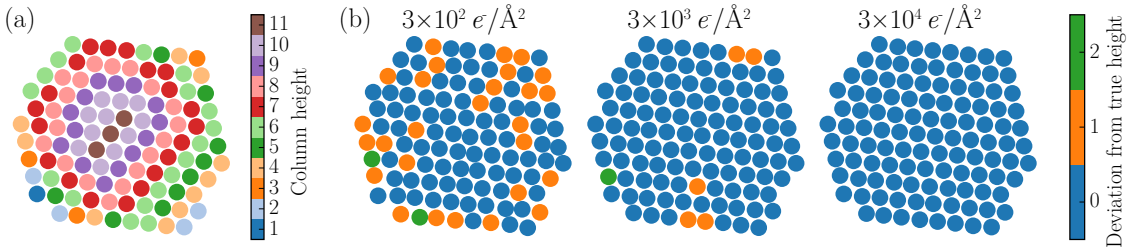


Figure 6.29: **(a)** Map of the true number atoms in each column of the nanoparticle used to generate the images in Fig. 6.28 **(b)** Map of the deviation from the true height in the predictions of our CNN-based atom counting technique.

tion we use three images at different defocus as input. We used a more limited range of simulation parameters and only nanoparticles in the ⟨110⟩-zone axis was considered. The defocus value for input images, each denoted by k, was chosen as

$$\Delta f_k = [-170 + 130k + \mathcal{U}(-20, 20)]\text{Å} \quad , \tag{6.19}$$

where $\mathcal{U}(-20, 20)$ is a uniform random distribution from -20 to 20. We used a range of spherical aberrations between $C_s = -10$ µm to $C_s = -20$ µm. While the other simulation parameters were chosen as described in Table 6.1. An example of a set of images is given in Fig. 6.28.

For validation, we used the same set of relaxed nanoparticles, as described in the previous section. Fig. 6.29 shows the results of applying the trained CNN to the nanoparticle, given different electon doses. For this example, there is no errors at a dose of 3×10^4 $e^-/\text{Å}^2$. At lower doses, the prediction is generally only off by 1 atom. This example was almost exactly in the ideal ⟨110⟩, hence the performance of the CNN, was somewhat better than the average of full ensemble used for validation. The performance on nanoparticles tilted up to 3 ° was significantly worse. In Fig. 6.30, we show the mean fraction of errors over the entire ensemble.

At a dose of 10^4 $e^-/\text{Å}^2$ the probability for making an error, that is off by one from the true height is $\sim$ 10 %, which decreases to $\sim$ 5 % at a dose of 10^6 $e^-/\text{Å}^2$. Generally, when the CNN makes a mistake, the predicted probability for the true height, is only slightly smaller than the false prediction. The number of errors that are off by 2 or more is very low.

Feedback from experiment is required to understand whether the estimation of the errors presented here are realistic. However, the results are interesting enough to warrant further investigation. In any case, the results of this section indicate that our CNN-based method is capable of distinguishing very similar structures.

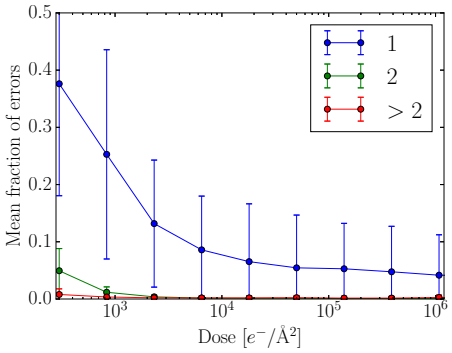


Figure 6.30: The mean fraction of errors over the validation set for predicting the height of an atomic column in a nanoparticle. We show separate curves for errors that are off by 1, 2 or more than 2 from the true height. The dose is total for all three images.

6.5 Summary

A deep learning-based algorithm for recognition of the local structure in TEM images was presented. It was shown to be stable to microscope parameters and noise. The neural network was is trained entirely from simulation, but is capable of making correct predictions on experimental images. The method was applied successfully to exmperimental images of single sheets of defected graphene and to a video sequence of a supported nanoparticle. The method also showed promising but, as of yet, inconclusive results for reconstructing the 3d shape of nanoparticles using atom counting.

Generally the first introduction of an image recognition model is improved by consecutive by the general community. Hence we believe that there is still much potential for improving the method. For example, we have done very little hyperparameter optimization with respect to e.g. weight decay and the depth of the CNN. In the machine learning community, improvement is often driven by creating standardized training and validation sets, and competing to obtain the best performing model [206].

Bibliography

[1] I. (Ib) Chorkendorff and J. W. Niemantsverdriet. *Concepts of modern catalysis and kinetics*. Wiley-VCH, 2007, p. 457. ISBN: 3527316728.

[2] C.V. Ovesen et al. "Kinetic Implications of Dynamical Changes in Catalyst Morphology during Methanol Synthesis over Cu/ZnO Catalysts". In: *J. Catal.* 168.2 (June 1997), pp. 133–142. ISSN: 00219517. DOI: 10.1006/jcat.1997.1629.

[3] K. Honkala et al. "Ammonia Synthesis from First-Principles Calculations." In: *ChemInform* 36.17 (Apr. 2005). ISSN: 0931-7597. DOI: 10.1002/chin.200517025.

[4] M. Mavrikakis, B. Hammer, and J. K. Nørskov. "Effect of strain on the reactivity of metal surfaces". In: *Phys. Rev. Lett.* 81.13 (Sept. 1998), pp. 2819–2822. ISSN: 0031-9007. DOI: 10.1103/PhysRevLett.81.2819.

[5] Jakob B. Wagner et al. "Exploring the environmental transmission electron microscope". In: *Micron* 43.11 (Nov. 2012), pp. 1169–1175. ISSN: 09684328. DOI: 10.1016/j.micron.2012.02.008.

[6] David J. Smith. "Chapter 1. Characterization of Nanomaterials Using Transmission Electron Microscopy". In: Aug. 2015, pp. 1–29. DOI: 10.1039/9781782621867-00001.

[7] Earl J. Kirkland. *Advanced computing in electron microscopy*. Springer, 2014. ISBN: 1489995099.

[8] Simon Kurasch et al. "Simulation of bonding effects in HRTEM images of light element materials." In: *Beilstein J. Nanotechnol.* 2 (2011), pp. 394–404. ISSN: 2190-4286. DOI: 10.3762/bjnano.2.45.

[9] Leslie E. Ballentine. *Quantum mechanics : a modern development*. 2014, p. 722. ISBN: 9814578584.

[10] L.J. Allen et al. "Exit wave reconstruction at atomic resolution". In: *Ultramicroscopy* 100.1-2 (July 2004), pp. 91–104. ISSN: 03043991. DOI: 10.1016/j.ultramic.2004.01.012.

[11] Stephan Uhlemann et al. "Thermal Magnetic Field Noise Limits Resolution in Transmission Electron Microscopy". In: *Phys. Rev. Lett.* 111.4 (July 2013), p. 046101. ISSN: 0031-9007. DOI: 10.1103/PhysRevLett.111.046101.

[12] D. Van Dyck. "Is the frozen phonon model adequate to describe inelastic phonon scattering?" In: *Ultramicroscopy* 109.6 (May 2009), pp. 677–682. ISSN: 03043991. DOI: 10.1016/j.ultramic.2009.01.001.

[13] R. F. Loane, P. Xu, and J. Silcox. "Thermal vibrations in convergent-beam electron diffraction". In: *Acta Crystallogr. Sect. A Found. Crystallogr.* 47.3 (May 1991), pp. 267–278. ISSN: 0108-7673. DOI: 10.1107/S0108767391000375.

[14] C. R. Hall and P. B. Hirsch. "Effect of Thermal Diffuse Scattering on Propagation of High Energy Electrons Through Crystals". In: *Proc. R. Soc. A Math. Phys. Eng. Sci.* 286.1405 (June 1965), pp. 158–177. ISSN: 1364-5021. DOI: 10.1098/rspa.1965.0136.

[15] Rüdiger R. Meyer and Angus I. Kirkland. "Characterisation of the signal and noise transfer of CCD cameras for electron detection". In: *Microsc. Res. Tech.* 49.3 (May 2000), pp. 269–280.

[16] Rachel S Ruskin, Zhiheng Yu, and Nikolaus Grigorieff. "Quantitative characterization of electron detectors for transmission electron microscopy." In: *J. Struct. Biol.* 184.3 (Dec. 2013), pp. 385–93. ISSN: 1095-8657. DOI: 10.1016/j.jsb.2013.10.016.

[17] Z Lee et al. "Optimum HRTEM image contrast at 20 kV and 80 kV—Exemplified by graphene". In: *Ultramicroscopy* 112 (2011), pp. 39–46. DOI: 10.1016/j.ultramic.2011.10.009.

[18] Marc. De Graef. *Introduction to conventional transmission electron microscopy*. Cambridge: Cambridge University Press, 2003, p. 718. ISBN: 9780521629959.

[19] Z Lee et al. "Electron dose dependence of signal-to-noise ratio, atom contrast and resolution in transmission electron microscope images". In: *Ultramicroscopy* 145 (2014), pp. 3–12. DOI: 10.1016/j.ultramic.2014.01.010.

[20] Ask Hjorth Larsen et al. "The atomic simulation environment—a Python library for working with atoms". In: *J. Phys. Condens. Matter* 29.27 (July 2017), p. 273002. ISSN: 0953-8984. DOI: 10.1088/1361-648X/aa680e.

[21] Thomas Kluyver. "Jupyter Notebooks – a publishing format for reproducible computational workflows". In: (2016).

[22] Patricia Hernandez-Fernandez et al. "Mass-selected nanoparticles of PtxY as model catalysts for oxygen electroreduction". In: *Nat. Chem.* 6.8 (July 2014), p. 732. ISSN: 1755-4330. DOI: 10.1038/nchem.2001.

[23] Brian T. Sneed et al. "Building up strain in colloidal metal nanoparticle catalysts". In: *Nanoscale* 7.29 (2015), pp. 12248–12265. ISSN: 2040-3364. DOI: 10.1039/C5NR02529J.

[24] S. Di Fonzo et al. "Non-destructive determination of local strain with 100-nanometre spatial resolution". In: *Nature* 403.6770 (Feb. 2000), pp. 638–640. ISSN: 0028-0836. DOI: 10.1038/35001035.

[25] Vincent Senez et al. "Strain determination in silicon microstructures by combined convergent beam electron diffraction, process simulation, and micro-Raman spectroscopy". In: *J. Appl. Phys.* 94.9 (Nov. 2003), pp. 5574–5583. ISSN: 0021-8979. DOI: 10.1063/1.1611287.

[26] H. Seitz et al. "Quantitative strain mapping using high-resolution electron microscopy". In: *Phys. Status Solidi* 150.2 (Aug. 1995), pp. 625–634. ISSN: 00318965. DOI: 10.1002/pssa.2211500206.

[27] M.J. Hÿtch, E. Snoeck, and R. Kilaas. "Quantitative measurement of displacement and strain fields from HREM micrographs". In: *Ultramicroscopy* 74.3 (Aug. 1998), pp. 131–146. ISSN: 03043991. DOI: 10.1016/S0304-3991(98)00035-7.

[28] Peng Zhang et al. "Direct strain measurement in a 65nm node strained silicon transistor by convergent-beam electron diffraction". In: *Appl. Phys. Lett.* 89.16 (Oct. 2006), p. 161907. ISSN: 0003-6951. DOI: 10.1063/1.2362978.

[29] V. B. Ozdol et al. "Strain mapping at nanometer resolution using advanced nano-beam electron diffraction". In: *Appl. Phys. Lett.* 106.25 (June 2015), p. 253107. ISSN: 0003-6951. DOI: 10.1063/1.4922994.

[30] A. Béché et al. "Dark field electron holography for strain measurement". In: *Ultramicroscopy* 111.3 (Feb. 2011), pp. 227–238. ISSN: 03043991. DOI: 10.1016/j.ultramic.2010.11.030.

[31] A. Béché et al. "Strain measurement at the nanoscale: Comparison between convergent beam electron diffraction, nano-beam electron diffraction, high resolution imaging and dark field electron holography". In: *Ultramicroscopy* 131 (Aug. 2013), pp. 10–23. ISSN: 03043991. DOI: 10.1016/j.ultramic.2013.03.014.

[32] Martin J. Hÿtch and Andrew M. Minor. "Observing and measuring strain in nanostructures and devices with transmission electron microscopy". In: *MRS Bull.* 39.02 (Feb. 2014), pp. 138–146. ISSN: 0883-7694. DOI: 10.1557/mrs.2014.4.

[33] David Cooper et al. "Strain mapping of semiconductor specimens with nm-scale resolution in a transmission electron microscope". In: *Micron* 80 (Jan. 2016), pp. 145–165. ISSN: 09684328. DOI: 10.1016/j.micron.2015.09.001.

[34] Martin Couillard, Guillaume Radtke, and Gianluigi A. Botton. "Strain fields around dislocation arrays in a Σ9 silicon bicrystal measured by scanning transmission electron microscopy". In: *Philos. Mag.* 93.10-12 (Apr. 2013), pp. 1250–1267. ISSN: 1478-6435. DOI: 10.1080/14786435.2013.778428.

[35] Lewys Jones et al. “Optimising multi-frame ADF-STEM for high-precision atomic-resolution strain mapping”. In: *Ultramicroscopy* 179 (Aug. 2017), pp. 57–62. ISSN: 03043991. DOI: 10.1016/j.ultramic.2017.04.007.

[36] M.J. Hÿtch and W.M. Stobbs. “Quantitative comparison of high resolution TEM images with image simulations”. In: *Ultramicroscopy* 53.3 (Mar. 1994), pp. 191–203. ISSN: 03043991. DOI: 10.1016/0304-3991(94)90034-5.

[37] Jonathan J. P. Petersa et al. “Artefacts in geometric phase analysis of compound materials”. In: (Apr. 2015). arXiv: 1504.06089.

[38] L.D. Marks. “Wiener-filter enhancement of noisy HREM images”. In: *Ultramicroscopy* 62.1-2 (Jan. 1996), pp. 43–52. ISSN: 03043991. DOI: 10.1016/0304-3991(95)00085-2.

[39] Ondrej L. Krivanek et al. “Atom-by-atom structural and chemical analysis by annular dark-field electron microscopy”. In: *Nature* 464.7288 (Mar. 2010), pp. 571–574. ISSN: 0028-0836. DOI: 10.1038/nature08879.

[40] Pedro L Galindo et al. “The Peak Pairs algorithm for strain mapping from HRTEM images.” In: *Ultramicroscopy* 107.12 (Nov. 2007), pp. 1186–93. ISSN: 0304-3991. DOI: 10.1016/j.ultramic.2007.01.019.

[41] Magnus Nord et al. “Atomap: a new software tool for the automated analysis of atomic resolution images using two-dimensional Gaussian fitting”. In: *Adv. Struct. Chem. Imaging* 3.1 (Dec. 2017), p. 9. ISSN: 2198-0926. DOI: 10.1186/s40679-017-0042-5.

[42] K Du. “Lattice distortion analysis directly from high resolution transmission electron microscopy images - the LADIA program package”. In: *J. Mater. Sci. Technol.* 18.2 (2002). ISSN: 10050302.

[43] Jian-Min Zuo et al. “Lattice and strain analysis of atomic resolution Z-contrast images based on template matching”. In: *Ultramicroscopy* 136 (2014), pp. 50–60. ISSN: 03043991. DOI: 10.1016/j.ultramic.2013.07.018.

[44] Berthold K. P. Horn. “Closed-form solution of absolute orientation using unit quaternions”. In: *J. Opt. Soc. Am. A* 4.4 (Apr. 1987), p. 629. ISSN: 1084-7529. DOI: 10.1364/JOSAA.4.000629.

[45] Douglas L. Theobald. “Rapid calculation of RMSDs using a quaternion-based characteristic polynomial”. In: *Acta Crystallogr. Sect. A Found. Crystallogr.* 61.4 (July 2005), pp. 478–480. ISSN: 0108-7673. DOI: 10.1107/S0108767305015266.

[46] Matthew Griffiths, Samuel P. Niblett, and David J. Wales. “Optimal Alignment of Structures for Finite and Periodic Systems”. In: *J. Chem. Theory Comput.* 13.10 (Oct. 2017), pp. 4914–4931. ISSN: 1549-9618. DOI: 10.1021/acs.jctc.7b00543.

[47] E.-J. Hong, K.-H. Lee, and W. Wenzel. "RMSD computation for clusters of identical particles". In: *Proc. 4th WSEAS Conf. Math. Biol* (2008), p. 46.

[48] Takeshi Fujita et al. "Atomic origins of the high catalytic activity of nanoporous gold". In: *Nat. Mater.* 11.9 (Aug. 2012), pp. 775–780. ISSN: 1476-1122. DOI: 10.1038/nmat3391.

[49] Takeshi Daio et al. "Lattice Strain Mapping of Platinum Nanoparticles on Carbon and SnO2 Supports". In: *Sci. Rep.* 5.1 (Oct. 2015), p. 13126. ISSN: 2045-2322. DOI: 10.1038/srep13126.

[50] Mingjian Wu et al. "Interface structure and strain state of InAs nano-clusters embedded in silicon". In: *Acta Mater.* 90 (May 2015), pp. 133–139. ISSN: 13596454. DOI: 10.1016/j.actamat.2015.02.042.

[51] L D Marks. "Direct atomic imaging of solid surfaces - 1. Image simulations and interpretation". In: *Surf. Sci.* 139.1 (1984), pp. 281–298.

[52] K. Du and F. Phillipp. "On the accuracy of lattice-distortion analysis directly from high-resolution transmission electron micrographs". In: *J. Microsc.* 221.1 (Jan. 2006), pp. 63–71. ISSN: 0022-2720. DOI: 10.1111/j.1365-2818.2006.01536.x.

[53] Elisa Guerrero et al. "Error quantification in strain mapping methods". In: *Microsc. Microanal.* 13.05 (Oct. 2007), pp. 320–328. ISSN: 1431-9276. DOI: 10.1017/S1431927607070407.

[54] K Tillmann et al. "Impact of column bending in high-resolution transmission electron microscopy on the strain evaluation of GaAs/InAs/GaAs heterostructures". In: *Ultramicroscopy* 83.1-2 (May 2000), pp. 111–128. ISSN: 03043991. DOI: 10.1016/S0304-3991(99)00175-8.

[55] A. Rosenauer, D. Gerthsen, and V. Potin. "Strain state analysis of InGaN/GaN - sources of error and optimized imaging conditions". In: *Phys. status solidi* 203.1 (Jan. 2006), pp. 176–184. ISSN: 18626300. DOI: 10.1002/pssa.200563519.

[56] Jayhoon Chung and Lew Rabenberg. "Effects of strain gradients on strain measurements using geometrical phase analysis in the transmission electron microscope". In: *Ultramicroscopy* 108.12 (Nov. 2008), pp. 1595–1602. ISSN: 03043991. DOI: 10.1016/j.ultramic.2008.05.010.

[57] M.J. Hÿtch and T. Plamann. "Imaging conditions for reliable measurement of displacement and strain in high-resolution electron microscopy". In: *Ultramicroscopy* 87.4 (May 2001), pp. 199–212. ISSN: 03043991. DOI: 10.1016/S0304-3991(00)00099-1.

[58] M. José Yacamán, J.A. Ascencio, and G. Canizal. "Observation of surface relaxation surface steps and surface reconstruction in gold nanorods". 2001.

[59] Takeshi Fujita et al. "Atomic origins of the high catalytic activity of nanoporous gold". In: *Nat. Mater.* 11.9 (Aug. 2012), pp. 775–780. ISSN: 1476-1122. DOI: 10.1038/nmat3391.

[60] Miguel López-Haro et al. “Strain field in ultrasmall gold nanoparticles supported on cerium-based mixed oxides. Key influence of the support redox state”. In: (2016).

[61] K. Du et al. “Expansion of interatomic distances in platinum catalyst nanoparticles”. In: *Acta Mater.* 58.3 (2010), pp. 836–845. ISSN: 13596454. DOI: 10.1016/j.actamat.2009.09.061.

[62] Marianne Monteforte et al. “Quantitative two-dimensional strain mapping of small core–shell FePt@Fe ¡sub¿3¡/sub¿ O ¡sub¿4¡/sub¿ nanoparticles”. In: *New J. Phys.* 18.3 (Mar. 2016), p. 033016. ISSN: 1367-2630. DOI: 10.1088/1367-2630/18/3/033016.

[63] Lin Gan et al. “Lattice Strain Distributions in Individual Dealloyed Pt–Fe Catalyst Nanoparticles”. In: (2012). DOI: 10.1021/JZ300192B.

[64] Ellad B. Tadmor and Ronald E. Miller. *Modeling Materials*. Vol. 9780521856. Cambridge: Cambridge University Press, 2011. ISBN: 9781139003582. DOI: 10.1017/CBO9781139003582.

[65] W. Martienssen and Hans Warlimont. *Springer handbook of condensed matter and materials data*. Ed. by Werner Warlimont and Martienssen Hans. 1st ed. Berlin: Springer, 2005, p. 1119. ISBN: 9783540304371.

[66] R.J Nichols et al. “Surface relaxation and surface stress of Au(111)”. In: *Surf. Sci.* 513.2 (2002), pp. 263–271. ISSN: 00396028. DOI: 10.1016/S0039-6028(02)01510-8.

[67] R.W. Vook, S. Ouyang, and M.A. Otooni. “X-ray measurement of the (111) surface relaxation in gold”. In: *Surf. Sci.* 29.1 (Jan. 1972), pp. 277–290. ISSN: 00396028. DOI: 10.1016/0039-6028(72)90084-2.

[68] Byung Deok Yu and Matthias Scheffler. “Physical origin of exchange diffusion on fcc(100) metal surfaces”. In: *Phys. Rev. B* 56.24 (Dec. 1997), R15569–R15572. ISSN: 0163-1829. DOI: 10.1103/PhysRevB.56.R15569.

[69] Nicholas E. Singh-Miller and Nicola Marzari. “Surface energies, work functions, and surface relaxations of low-index metallic surfaces from first principles”. In: *Phys. Rev. B* 80.23 (Dec. 2009), p. 235407. ISSN: 1098-0121. DOI: 10.1103/PhysRevB.80.235407.

[70] D. Van Dyck and M. Op de Beeck. “A simple intuitive theory for electron diffraction”. In: *Ultramicroscopy* 64.1 (1996), pp. 99–107. ISSN: 03043991. DOI: 10.1016/0304-3991(96)00008-3.

[71] P. Geuens and D. Van Dyck. “The S-state model: a work horse for HRTEM”. In: *Ultramicroscopy* 93.3 (2002), pp. 179–198. ISSN: 03043991. DOI: 10.1016/S0304-3991(02)00276-0.

[72] D Van Dyck. “Does crystal tilt enhance the electron interaction?” In: *Microsc. Microanal.* 4.4 (1998). ISSN: 1431-9276.

[73] S E Kul’kova et al. “Theoretical study of adhesion at the metal-zirconium dioxide interfaces”. English. In: *Tech. Phys.* 58.3 (2013), pp. 325–334. DOI: 10.1134/s106378421303016x.

[74] J W Feng, W Q Zhang, and W Jiang. "Ab initio study of Ag/Al2O3 and Au/Al2O3 interfaces". English. In: *Phys. Rev. B* 72.11 (2005), p. 11. DOI: 10.1103/PhysRevB.72.115423.

[75] Hungru Chen et al. "Bonding and Electron Energy-Level Alignment at Metal/TiO ¡sub¿2¡/sub¿ Interfaces: A Density Functional Theory Study". In: *J. Phys. Chem. C* 120.10 (Mar. 2016), pp. 5549–5556. ISSN: 1932-7447. DOI: 10.1021/acs.jpcc.5b12681.

[76] Giovanni Barcaro, Edoardo Aprà, and Alessandro Fortunelli. "Structure of Ag clusters Grown on Fs-Defect Sites of an MgO(1 0 0) surface." In: *Chemistry* 13.22 (Jan. 2007), pp. 6408–18. ISSN: 0947-6539. DOI: 10.1002/chem.200601796.

[77] Jakob Nordheim Riedel et al. "H ¡sub¿2¡/sub¿ /D ¡sub¿2¡/sub¿ exchange reaction on monodisperse Pt clusters: enhanced activity from minute O ¡sub¿2¡/sub¿ concentrations". In: *Catal. Sci. Technol.* 6.18 (2016), pp. 6893–6900. ISSN: 2044-4753. DOI: 10.1039/C6CY00756B.

[78] Sergey M Kozlov et al. "Effect of MgO(100) support on structure and properties of Pd and Pt nanoparticles with 49-155 atoms." In: *J. Chem. Phys.* 139.8 (Aug. 2013), p. 084701. ISSN: 1089-7690. DOI: 10.1063/1.4817948.

[79] Ashkan Moradabadi, Shideh Ahmadi, and Payam Kaghazchi. "Evidence of a strong effect of defect-free metal oxide supports on Pt nanoparticles". In: *Nanoscale* 9.13 (Mar. 2017), pp. 4478–4485. ISSN: 2040-3364. DOI: 10.1039/C6NR07816H.

[80] C. Mottet and J. Goniakowski. "Melting and freezing of Pd nanoclusters: effect of the MgO(100) substrate". In: *Surf. Sci.* 566-568 (Sept. 2004), pp. 443–450. ISSN: 00396028. DOI: 10.1016/j.susc.2004.05.083.

[81] Riccardo Ferrando et al. "Structures of metal nanoparticles adsorbed on MgO(001). I. Ag and Au". In: *J. Chem. Phys.* 130.17 (May 2009), p. 174702. ISSN: 0021-9606. DOI: 10.1063/1.3077300.

[82] J Goniakowski et al. "Structures of metal nanoparticles adsorbed on MgO(001). II. Pt and Pd". English. In: *J. Chem. Phys.* 130.17 (2009), p. 9. DOI: 10.1063/1.3121307.

[83] Davide Bochicchio et al. "Structures and segregation patterns of Ag–Cu and Ag–Ni nanoalloys adsorbed on MgO(0 0 1)". In: *J. Phys. Condens. Matter* 28.6 (Feb. 2016), p. 064005. ISSN: 0953-8984. DOI: 10.1088/0953-8984/28/6/064005.

[84] David Mora-Fonz et al. "Development of Interatomic Potentials for Supported Nanoparticles: The Cu/ZnO Case". In: *J. Phys. Chem. C* 121.31 (Aug. 2017), pp. 16831–16844. ISSN: 1932-7447. DOI: 10.1021/acs.jpcc.7b04502.

[85] Jacek Goniakowski, Christine Mottet, and Claudine Noguera. "Non-reactive metal/oxide interfaces: from model calculations towards realistic simulations". In: *Phys. status solidi* 243.11 (Sept. 2006), pp. 2516–2532. ISSN: 0370-1972. DOI: 10.1002/pssb.200541456.

[86] H. Iddir et al. “Atomic scale characterization of the Pt/TiO2 interface”. In: *Micron* 36.3 (Apr. 2005), pp. 233–241. ISSN: 09684328. DOI: 10.1016/j.micron.2004.12.002.

[87] Shuyi Zhang et al. “Dynamical Observation and Detailed Description of Catalysts under Strong Metal–Support Interaction”. In: *Nano Lett.* 16.7 (July 2016), pp. 4528–4534. ISSN: 1530-6984. DOI: 10.1021/acs.nanolett.6b01769.

[88] Hailian Tang et al. “Classical strong metal-support interactions between gold nanoparticles and titanium dioxide.” In: *Sci. Adv.* 3.10 (Oct. 2017), e1700231. ISSN: 2375-2548. DOI: 10.1126/sciadv.1700231.

[89] Sergey V. Dmitriev et al. “Atomistic structure of the Cu(111)/α-Al2O3(0001) interface in terms of interatomic potentials fitted to ab initio results”. In: *Acta Mater.* 52.7 (Apr. 2004), pp. 1959–1970. ISSN: 13596454. DOI: 10.1016/j.actamat.2003.12.037.

[90] Zizhe Lu et al. “Nanoindentation of ZrO 2 and ZrO 2 /Zr systems by molecular dynamics simulation”. In: *J. Nucl. Mater.* 486 (Apr. 2017), pp. 250–266. ISSN: 00223115. DOI: 10.1016/j.jnucmat.2017.01.022.

[91] Tao Liang et al. “Variable Charge Reactive Potential for Hydrocarbons to Simulate Organic-Copper Interactions”. In: *J. Phys. Chem. A* 116.30 (Aug. 2012), pp. 7976–7991. ISSN: 1089-5639. DOI: 10.1021/jp212083t.

[92] Kamal Choudhary et al. “Charge optimized many-body (COMB) potential for Al ¡sub¿2¡/sub¿ O ¡sub¿3¡/sub¿ materials, interfaces, and nanostructures”. In: *J. Phys. Condens. Matter* 27.30 (Aug. 2015), p. 305004. ISSN: 0953-8984. DOI: 10.1088/0953-8984/27/30/305004.

[93] Yu-Ting Cheng et al. “A charge optimized many-body (comb) potential for titanium and titania”. In: *J. Phys. Condens. Matter* 26.31 (Aug. 2014), p. 315007. ISSN: 0953-8984. DOI: 10.1088/0953-8984/26/31/315007.

[94] Yu-Ting Cheng et al. “Atomistic simulations of the adsorption and migration barriers of Cu adatoms on ZnO surfaces using COMB potentials”. In: *Surf. Sci.* 606.15-16 (Aug. 2012), pp. 1280–1288. ISSN: 00396028. DOI: 10.1016/j.susc.2012.04.007.

[95] Tao Liang et al. “Classical atomistic simulations of surfaces and heterogeneous interfaces with the charge-optimized many body (COMB) potentials”. In: *Mater. Sci. Eng. R Reports* 74.9 (Sept. 2013), pp. 255–279. ISSN: 0927796X. DOI: 10.1016/j.mser.2013.07.001.

[96] Albert P. Bartók et al. “Gaussian Approximation Potentials: The Accuracy of Quantum Mechanics, without the Electrons”. In: *Phys. Rev. Lett.* 104.13 (Apr. 2010), p. 136403. ISSN: 0031-9007. DOI: 10.1103/PhysRevLett.104.136403.

[97] Jörg Behler. “Neural network potential-energy surfaces in chemistry: a tool for large-scale simulations.” In: *Phys. Chem. Chem. Phys.* 13.40 (Oct. 2011), pp. 17930–55. ISSN: 1463-9084. DOI: 10.1039/c1cp21668f.

[98] Nongnuch Artrith, Björn Hiller, and Jörg Behler. "Neural network potentials for metals and oxides - First applications to copper clusters at zinc oxide". In: *Phys. status solidi* 250.6 (June 2013), pp. 1191–1203. ISSN: 03701972. DOI: 10.1002/pssb.201248370.

[99] G. C. Abell. "Empirical chemical pseudopotential theory of molecular and metallic bonding". In: *Phys. Rev. B* 31.10 (May 1985), pp. 6184–6196. ISSN: 0163-1829. DOI: 10.1103/PhysRevB.31.6184.

[100] J. Tersoff. "New empirical approach for the structure and energy of covalent systems". In: *Phys. Rev. B* 37.12 (Apr. 1988), pp. 6991–7000. ISSN: 0163-1829. DOI: 10.1103/PhysRevB.37.6991.

[101] Anthony K. Rappe and William A. Goddard. "Charge equilibration for molecular dynamics simulations". In: *J. Phys. Chem.* 95.8 (Apr. 1991), pp. 3358–3363. ISSN: 0022-3654. DOI: 10.1021/j100161a070.

[102] Jackelyn A. Martinez et al. "Fitting empirical potentials: Challenges and methodologies". In: *Curr. Opin. Solid State Mater. Sci.* 17.6 (Dec. 2013), pp. 263–270. ISSN: 13590286. DOI: 10.1016/j.cossms.2013.09.001.

[103] J. A. Nelder and R. Mead. "A Simplex Method for Function Minimization". In: *Comput. J.* 7.4 (Jan. 1965), pp. 308–313. ISSN: 0010-4620. DOI: 10.1093/comjnl/7.4.308.

[104] J. J. Mortensen, L. B. Hansen, and K. W. Jacobsen. "Real-space grid implementation of the projector augmented wave method". In: *Phys. Rev. B* 71.3 (Jan. 2005), p. 035109. ISSN: 1098-0121. DOI: 10.1103/PhysRevB.71.035109.

[105] J Enkovaara et al. "Electronic structure calculations with GPAW: a real-space implementation of the projector augmented-wave method". In: *J. Phys. Condens. Matter* 22.25 (June 2010), p. 253202. ISSN: 0953-8984. DOI: 10.1088/0953-8984/22/25/253202.

[106] John P. Perdew, Kieron Burke, and Matthias Ernzerhof. "Generalized Gradient Approximation Made Simple". In: *Phys. Rev. Lett.* 77.18 (Oct. 1996), pp. 3865–3868. ISSN: 0031-9007. DOI: 10.1103/PhysRevLett.77.3865.

[107] P. E. Blöchl. "Projector augmented-wave method". In: *Phys. Rev. B* 50.24 (Dec. 1994), pp. 17953–17979. ISSN: 0163-1829. DOI: 10.1103/PhysRevB.50.17953.

[108] Hendrik J. Monkhorst and James D. Pack. "Special points for Brillouin-zone integrations". In: *Phys. Rev. B* 13.12 (June 1976), pp. 5188–5192. ISSN: 0556-2805. DOI: 10.1103/PhysRevB.13.5188.

[109] Charles. Kittel. *Introduction to solid state physics*. Wiley, 2005, p. 680. ISBN: 047141526X.

[110] J. C. Boettger. "Nonconvergence of surface energies obtained from thin-film calculations". In: *Phys. Rev. B* 49.23 (June 1994), pp. 16798–16800. ISSN: 0163-1829. DOI: 10.1103/PhysRevB.49.16798.

[111] Vincenzo Fiorentini and M Methfessel. "Extracting convergent surface energies from slab calculations". In: *J. Phys. Condens. Matter* 8.36 (Sept. 1996), pp. 6525–6529. ISSN: 0953-8984. DOI: 10.1088/0953-8984/8/36/005.

[112] John P. Perdew et al. "Restoring the Density-Gradient Expansion for Exchange in Solids and Surfaces". In: *Phys. Rev. Lett.* 100.13 (Apr. 2008), p. 136406. ISSN: 0031-9007. DOI: 10.1103/PhysRevLett.100.136406.

[113] A C Antony et al. "Charge optimized many body (COMB) potentials for Pt and Au." In: *J. Phys. Condens. Matter* 29.22 (June 2017), p. 225901. ISSN: 1361-648X. DOI: 10.1088/1361-648X/aa6d43.

[114] Mariette Hellenbrandt. "The Inorganic Crystal Structure Database (ICSD)—Present and Future". In: *Crystallogr. Rev.* 10.1 (Jan. 2004), pp. 17–22. ISSN: 0889-311X. DOI: 10.1080/08893110410001664882.

[115] Peter Mahler Larsen, Søren Schmidt, and Jakob Schiøtz. "Robust structural identification via polyhedral template matching". In: *Model. Simul. Mater. Sci. Eng.* 24.5 (June 2016), p. 055007. ISSN: 0965-0393. DOI: 10.1088/0965-0393/24/5/055007.

[116] H. W. Sheng et al. "Highly optimized embedded-atom-method potentials for fourteen fcc metals". In: *Phys. Rev. B* 83.13 (Apr. 2011), p. 134118. ISSN: 1098-0121. DOI: 10.1103/PhysRevB.83.134118.

[117] T. T. Järvi et al. "Development of a ReaxFF description for gold". In: *Eur. Phys. J. B* 66.1 (Nov. 2008), pp. 75–79. ISSN: 1434-6028. DOI: 10.1140/epjb/e2008-00378-3.

[118] Y. S. (Yerem Sarkis) Touloukian. *Thermal expansion : metallic elements and alloys*. IFI/Plenum, 1975, p. 1440. ISBN: 0306670321.

[119] William M. Haynes. *Handbook of chemistry and physics : a ready-reference book of chemical and physical data.* 2014. ISBN: 1482208679.

[120] A.T. Dinsdale. "SGTE data for pure elements". In: *Calphad* 15.4 (Oct. 1991), pp. 317–425. ISSN: 03645916. DOI: 10.1016/0364-5916(91)90030-N.

[121] F. R. de (Frank R.) Boer. *Cohesion in metals : transition metal alloys*. North-Holland, 1988, p. 758. ISBN: 0444870989.

[122] H.-E. Schaefer. "Investigation of Thermal Equilibrium Vacancies in Metals by Positron Annihilation". In: *Phys. Status Solidi* 102.1 (July 1987), pp. 47–65. ISSN: 00318965. DOI: 10.1002/pssa.2211020104.

[123] Cheng Wang et al. "Generalized-stacking-fault energy and twin-boundary energy of hexagonal close-packed Au: A first-principles calculation". In: *Sci. Rep.* 5.1 (Sept. 2015), p. 10213. ISSN: 2045-2322. DOI: 10.1038/srep10213.

[124] Richard Tran et al. "Surface energies of elemental crystals". In: *Sci. Data* 3 (Sept. 2016), p. 160080. ISSN: 2052-4463. DOI: 10.1038/sdata.2016.80.

[125] Simon Hedegaard Brodersen, Jakob Schiøtz, and Jane Hvolbæk Nielsen. *Theoretical modelling of nanoparticles with applications to catalysis and sustainable energy*. 2014.

[126] Xiao-Zhi Wu et al. "Ab initio calculations of generalized-stacking-fault energy surfaces and surface energies for FCC metals". In: *Appl. Surf. Sci.* 256.21 (Aug. 2010), pp. 6345–6349. ISSN: 01694332. DOI: 10.1016/j.apsusc.2010.04.014.

[127] Donato Fantauzzi et al. "Development of a ReaxFF potential for Pt–O systems describing the energetics and dynamics of Pt-oxide formation". In: *Phys. Chem. Chem. Phys.* 16.42 (2014), pp. 23118–23133. ISSN: 1463-9076. DOI: 10.1039/C4CP03111C.

[128] G. V. Samsonov, ed. *The Oxide Handbook*. Boston, MA: Springer US, 1973. ISBN: 978-1-4615-9599-1. DOI: 10.1007/978-1-4615-9597-7.

[129] Hongqing Shi, Ryoji Asahi, and Catherine Stampfl. "Properties of the gold oxides Au 2 O 3 and Au 2 O : First-principles investigation". In: *Phys. Rev. B* 75.20 (May 2007), p. 205125. ISSN: 1098-0121. DOI: 10.1103/PhysRevB.75.205125.

[130] Hongqing Shi and Catherine Stampfl. "First-principles investigations of the structure and stability of oxygen adsorption and surface oxide formation at Au(111)". In: *Phys. Rev. B* 76.7 (Aug. 2007), p. 075327. ISSN: 1098-0121. DOI: 10.1103/PhysRevB.76.075327.

[131] Donato Fantauzzi et al. "Development of a ReaxFF potential for Pt–O systems describing the energetics and dynamics of Pt-oxide formation". In: *Phys. Chem. Chem. Phys.* 16.42 (2014), pp. 23118–23133. ISSN: 1463-9076. DOI: 10.1039/C4CP03111C.

[132] Jeffery M. Hawkins, Jason F. Weaver, and Aravind Asthagiri. "Density functional theory study of the initial oxidation of the Pt(111) surface". In: *Phys. Rev. B* 79.12 (Mar. 2009), p. 125434. DOI: 10.1103/PhysRevB.79.125434.

[133] J R Jinschek. "Advances in the environmental transmission electron microscope (ETEM) for nanoscale in situ studies of gas-solid interactions." In: *Chem. Commun. (Camb).* 50.21 (Mar. 2014), pp. 2696–706. ISSN: 1364-548X. DOI: 10.1039/c3cc49092k.

[134] G. McMullan et al. "Comparison of optimal performance at 300keV of three direct electron detectors for use in low dose electron microscopy". In: *Ultramicroscopy* 147 (Dec. 2014), pp. 156–163. ISSN: 03043991. DOI: 10.1016/j.ultramic.2014.08.002.

[135] Jannik C Meyer et al. "Experimental analysis of charge redistribution due to chemical bonding by high-resolution transmission electron microscopy". In: *Nat. Mater.* 10 (2011). DOI: 10.1038/NMAT2941.

[136] Jannik C. Meyer et al. "Direct Imaging of Lattice Atoms and Topological Defects in Graphene Membranes". In: *Nano Lett.* 8.11 (Nov. 2008), pp. 3582–3586. ISSN: 1530-6984. DOI: 10.1021/nl801386m.

[137] Xinfeng He et al. "In Situ Atom Scale Visualization of Domain Wall Dynamics in VO2 Insulator-Metal Phase Transition". In: *Sci. Rep.* 4.1 (May 2015), p. 6544. ISSN: 2045-2322. DOI: 10.1038/srep06544.

[138] Keisuke Nagao et al. "Experimental Observation of Quasicrystal Growth". In: *Phys. Rev. Lett.* 115.7 (Aug. 2015), p. 075501. ISSN: 0031-9007. DOI: 10.1103/PhysRevLett.115.075501.

[139] Xing Li et al. "Direct Observation of the Layer-by-Layer Growth of ZnO Nanopillar by In situ High Resolution Transmission Electron Microscopy". In: *Sci. Rep.* 7 (Jan. 2017), p. 40911. ISSN: 2045-2322. DOI: 10.1038/srep40911.

[140] S. Schneider et al. "Atomic surface diffusion on Pt nanoparticles quantified by high-resolution transmission electron microscopy". In: *Micron* 63.Sp. Iss. SI (Aug. 2014), pp. 52–56. ISSN: 09684328. DOI: 10.1016/j.micron.2013.12.011.

[141] M.J. Hÿtch, E. Snoeck, and R. Kilaas. "Quantitative measurement of displacement and strain fields from HREM micrographs". In: *Ultramicroscopy* 74.3 (1998), pp. 131–146. ISSN: 03043991. DOI: 10.1016/S0304-3991(98)00035-7.

[142] Yuanyuan Zhu et al. "Interface lattice displacement measurement to 1pm by geometric phase analysis on aberration-corrected HAADF STEM images". In: *Acta Mater.* 61.15 (Sept. 2013), pp. 5646–5663. ISSN: 13596454. DOI: 10.1016/j.actamat.2013.06.006.

[143] Julio C. Azcárate, Mariano H. Fonticelli, and Eugenia Zelaya. "Radiation Damage Mechanisms of Monolayer-Protected Nanoparticles via TEM Analysis". In: *J. Phys. Chem. C* (Nov. 2017), acs.jpcc.7b08525. ISSN: 1932-7447. DOI: 10.1021/acs.jpcc.7b08525.

[144] Alex Belianinov et al. "Identification of phases, symmetries and defects through local crystallography". In: *Nat. Commun.* 6 (July 2015), p. 7801. ISSN: 2041-1723. DOI: 10.1038/ncomms8801.

[145] Nouamane Laanait et al. "Identifying local structural states in atomic imaging by computer vision". In: *Adv. Struct. Chem. Imaging* 2.1 (Jan. 2017), p. 14. ISSN: 2198-0926. DOI: 10.1186/s40679-016-0028-8.

[146] Holger Kirschner and Reinald Hillebrand. "Neural networks for HREM image analysis". In: *Inf. Sci. (Ny).* 129.1-4 (Nov. 2000), pp. 31–44. ISSN: 00200255. DOI: 10.1016/S0020-0255(00)00067-0.

[147] R.R. Meyer and E. Heindl. "Reconstruction of off-axis electron holograms using a neural net". In: *J. Microsc.* 191.1 (1998), pp. 52–59. ISSN: 00222720.

[148] Wouter Van den Broek and Christoph T. Koch. "Method for Retrieval of the Three-Dimensional Object Potential by Inversion of Dynamical Electron Scattering". In: *Phys. Rev. Lett.* 109.24 (Dec. 2012), p. 245502. ISSN: 0031-9007. DOI: 10.1103/PhysRevLett.109.245502.

[149] Robert S. Pennington, Wouter Van den Broek, and Christoph T. Koch. "Third-dimension information retrieval from a single convergent-beam transmission electron diffraction pattern using an artificial neural network". In: *Phys. Rev. B* 89.20 (May 2014), p. 205409. ISSN: 1098-0121. DOI: 10.1103/PhysRevB.89.205409.

[150] Yanan Zhu, Qi Ouyang, and Youdong Mao. "A deep convolutional neural network approach to single-particle recognition in cryo-electron microscopy." In: *BMC Bioinformatics* 18.1 (July 2017), p. 348. ISSN: 1471-2105. DOI: 10.1186/s12859-017-1757-y.

[151] Tran Minh Quan, David G. C. Hildebrand, and Won-Ki Jeong. "FusionNet: A deep fully residual convolutional neural network for image segmentation in connectomics". In: (Dec. 2016). arXiv: 1612.05360.

[152] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E. Hinton. *ImageNet Classification with Deep Convolutional Neural Networks*. 2012.

[153] Ross Girshick. "Fast R-CNN". In: (Apr. 2015). arXiv: 1504.08083.

[154] Evan Shelhamer, Jonathan Long, and Trevor Darrell. "Fully Convolutional Networks for Semantic Segmentation". In: (May 2016). arXiv: 1605.06211.

[155] Ross Girshick et al. "Rich feature hierarchies for accurate object detection and semantic segmentation". In: (Nov. 2013). arXiv: 1311.2524.

[156] Malte Stær Nissen et al. "Convolutional neural networks for segmentation and object detection of human semen". In: (Apr. 2017). arXiv: 1704.00498.

[157] Hengshuang Zhao et al. "Pyramid Scene Parsing Network". In: (Dec. 2016). arXiv: 1612.01105.

[158] Ali Işın, Cem Direkoğlu, and Melike Şah. "Review of MRI-based Brain Tumor Image Segmentation Using Deep Learning Methods". In: *Procedia Comput. Sci.* 102 (2016), pp. 317–324. ISSN: 18770509. DOI: 10.1016/j.procs.2016.09.407.

[159] G. Cybenko. "Approximation by superpositions of a sigmoidal function". In: *Math. Control. Signals, Syst.* 2.4 (Dec. 1989), pp. 303–314. ISSN: 0932-4194. DOI: 10.1007/BF02551274.

[160] Ronen Eldan and Ohad Shamir. "The Power of Depth for Feedforward Neural Networks". In: (Dec. 2015). arXiv: 1512.03965.

[161] Tomaso Poggio et al. "Why and When Can Deep – but Not Shallow – Networks Avoid the Curse of Dimensionality: a Review". In: (Nov. 2016). arXiv: 1611.00740.

[162] Christopher M. Bishop. *Pattern recognition and machine learning*. Springer, 2006, p. 738. ISBN: 9780387310732.

[163] Yukun Zhu et al. "segDeepM: Exploiting Segmentation and Context in Deep Neural Networks for Object Detection". In: (Feb. 2015). arXiv: 1502.04275.

[164] Zhe Cao et al. "Realtime Multi-Person 2D Pose Estimation using Part Affinity Fields". In: (Nov. 2016). arXiv: 1611.08050.

[165] Dan C Cireşan et al. "Mitosis detection in breast cancer histology images with deep neural networks." In: *Med. Image Comput. Comput. Assist. Interv.* 16.Pt 2 (2013), pp. 411–8.

[166] Joseph Paul Cohen et al. "Count-ception: Counting by Fully Convolutional Redundant Counting". In: (Mar. 2017). arXiv: 1703.08710.

[167] Weidi Xie, J. Alison Noble, and Andrew Zisserman. "Microscopy cell counting and detection with fully convolutional regression networks". In: *Comput. Methods Biomech. Biomed. Eng. Imaging Vis.* (May 2016), pp. 1–10. ISSN: 2168-1163. DOI: 10.1080/21681163.2016.1149104.

[168] Jia Deng et al. "Imagenet: A large-scale hierarchical image database". In: *CVPR* (2009).

[169] Josh Tobin et al. "Domain Randomization for Transferring Deep Neural Networks from Simulation to the Real World". In: (Mar. 2017). arXiv: 1703.06907.

[170] Diederik P. Kingma and Jimmy Ba. "Adam: A Method for Stochastic Optimization". In: (Dec. 2014). arXiv: 1412.6980.

[171] Wenpei Gao, Zachary D. Hood, and Miaofang Chi. "Interfaces in Heterogeneous Catalysts: Advancing Mechanistic Understanding through Atomic-Scale Measurements". In: *Acc. Chem. Res.* 50.4 (Apr. 2017), pp. 787–795. ISSN: 0001-4842. DOI: 10.1021/acs.accounts.6b00596.

[172] Hao Chen et al. "Deep Contextual Networks for Neuronal Structure Segmentation". In: *Thirtieth AAAI Conf. Artif. Intell.* (Feb. 2016).

[173] Olaf Ronneberger, Philipp Fischer, and Thomas Brox. "U-Net: Convolutional Networks for Biomedical Image Segmentation". In: (May 2015). arXiv: 1505.04597.

[174] Kaiming He et al. "Deep Residual Learning for Image Recognition". In: *2016 IEEE Conf. Comput. Vis. Pattern Recognit.* IEEE, June 2016, pp. 770–778. ISBN: 978-1-4673-8851-1. DOI: 10.1109/CVPR.2016.90.

[175] Kaiming He et al. "Deep Residual Learning for Image Recognition". In: (Dec. 2015). arXiv: 1512.03385.

[176] Sergey Ioffe and Christian Szegedy. "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift". In: (Feb. 2015). arXiv: 1502.03167.

[177] Sergey Ioffe and Christian Szegedy. "Batch Normalization: Accelerating Deep Network Training by Reducing Internal Covariate Shift". In: (Feb. 2015). arXiv: 1502.03167.

[178] Martín Abadi et al. "TensorFlow: Large-Scale Machine Learning on Heterogeneous Distributed Systems". In: (Mar. 2016). arXiv: 1603.04467.

[179] Edward R. Dougherty. *Mathematical morphology in image processing*. M. Dekker, 1993, p. 530. ISBN: 0824787242.

[180] A. Thust. "High-Resolution Transmission Electron Microscopy on an Absolute Contrast Scale". In: *Phys. Rev. Lett.* 102.22 (June 2009), p. 220801. DOI: 10.1103/PhysRevLett.102.220801.

[181] C. Kramberger and J.C. Meyer. "Progress in structure recovery from low dose exposures: Mixed molecular adsorption, exploitation of symmetry and reconstruction from the minimum signal level". In: *Ultramicroscopy* 170 (Nov. 2016), pp. 60–68. ISSN: 03043991. DOI: 10.1016/j.ultramic.2016.08.009.

[182] Andreas Mittelberger et al. "Automated Image Acquisition for Low-Dose STEM at Atomic Resolution". In: *Microsc. Microanal.* 23.04 (Aug. 2017), pp. 809–817. ISSN: 1431-9276. DOI: 10.1017/S1431927617000575.

[183] Jacob S. Vestergaard et al. "Structure Identification in High-Resolution Transmission Electron Microscopic Images: An Example on Graphene". In: *Microsc. Microanal.* 20.06 (Dec. 2014), pp. 1772–1781. ISSN: 1431-9276. DOI: 10.1017/S1431927614013464.

[184] Guanghui Liang et al. "Poisson disk sampling through disk packing". In: *Comput. Vis. Media* 1.1 (Mar. 2015), pp. 17–26. ISSN: 2096-0433. DOI: 10.1007/s41095-015-0003-7.

[185] Stuart Lloyd. "LEAST SQUARES QUANTIZATION IN PCM". In: *Ieee Trans. Inf. Theory* IT-28.2 (1982). ISSN: 00189448.

[186] Jens Kling et al. "Pattern recognition approach to quantify the atomic structure of graphene". In: *Carbon N. Y.* 74 (Aug. 2014), pp. 363–366. ISSN: 00086223. DOI: 10.1016/j.carbon.2014.03.013.

[187] Colin Ophus et al. "Large-scale experimental and theoretical study of graphene grain boundary structures". In: *Phys. Rev. B* 92.20 (Nov. 2015), p. 205402. ISSN: 1098-0121. DOI: 10.1103/PhysRevB.92.205402.

[188] Steven J. Stuart, Alan B. Tutein, and Judith A. Harrison. "A reactive potential for hydrocarbons with intermolecular interactions". In: *J. Chem. Phys.* 112.14 (Apr. 2000), pp. 6472–6486. ISSN: 0021-9606. DOI: 10.1063/1.481208.

[189] Hedi Harzallah, Frederic Jurie, and Cordelia Schmid. "Combining efficient object localization and image classification". In: *2009 IEEE 12th Int. Conf. Comput. Vis.* IEEE, Sept. 2009, pp. 237–244. ISBN: 978-1-4244-4420-5. DOI: 10.1109/ICCV.2009.5459257.

[190] Yasufumi Kuwauchi et al. "Intrinsic Catalytic Structure of Gold Nanoparticles Supported on TiO ¡sub¿2¡/sub¿". In: *Angew. Chemie Int. Ed.* 51.31 (July 2012), pp. 7729–7733. ISSN: 14337851. DOI: 10.1002/anie.201201283.

[191] H. Yoshida et al. "Visualizing Gas Molecules Interacting with Supported Nanoparticulate Catalysts at Reaction Conditions". In: *Science (80-.).* 335.6066 (Jan. 2012), pp. 317–319. ISSN: 0036-8075. DOI: 10.1126/science.1213194.

[192] Seiji Takeda and Hideto Yoshida. "Atomic-resolution environmental TEM for quantitative in-situ microscopy in materials science". In: *Microscopy* 62.1 (Feb. 2013), pp. 193–203. ISSN: 2050-5698. DOI: 10.1093/jmicro/dfs096.

[193] Tetsuya Uchiyama et al. "Systematic Morphology Changes of Gold Nanoparticles Supported on CeO2 during CO Oxidation". In: *Angew. Chemie Int. Ed.* 50.43 (Oct. 2011), pp. 10157–10160. ISSN: 14337851. DOI: 10.1002/anie.201102487.

[194] Simon H. Brodersen et al. "Understanding the catalytic activity of gold nanoparticles through multi-scale simulations". In: *J. Catal.* 284.1 (2011), pp. 34–41. ISSN: 00219517. DOI: 10.1016/j.jcat.2011.08.016.

[195] J. (Joachim) Frank. *Electron tomography : three-dimensional imaging with the transmission electron microscope.* Plenum Press, 1992, p. 399. ISBN: 0306439956.

[196] Paul A. Midgley and Rafal E. Dunin-Borkowski. "Electron tomography and holography in materials science". In: *Nat. Mater.* 8.4 (Apr. 2009), pp. 271–280. ISSN: 1476-1122. DOI: 10.1038/nmat2406.

[197] Yongsoo Yang et al. "Deciphering chemical order/disorder and material properties at the single-atom level". In: *Nature* 542.7639 (Feb. 2017), pp. 75–79. ISSN: 0028-0836. DOI: 10.1038/nature21042.

[198] James M. LeBeau et al. "Standardless Atom Counting in Scanning Transmission Electron Microscopy". In: *Nano Lett.* 10.11 (Nov. 2010), pp. 4405–4408. ISSN: 1530-6984. DOI: 10.1021/nl102025s.

[199] Lewys Jones et al. "Rapid Estimation of Catalyst Nanoparticle Morphology and Atomic-Coordination by High-Resolution Z-Contrast Electron Microscopy". In: *Nano Lett.* 14.11 (Nov. 2014), pp. 6336–6341. ISSN: 1530-6984. DOI: 10.1021/nl502762m.

[200] A. De Backer et al. "Three-dimensional atomic models from a single projection using Z-contrast imaging: verification by electron tomography and opportunities". In: *Nanoscale* 9.25 (June 2017), pp. 8791–8798. ISSN: 2040-3364. DOI: 10.1039/C7NR02656K.

[201] C. L. Jia et al. "Determination of the 3D shape of a nanoscale crystal with atomic resolution from a single image". In: *Nat. Mater.* 13.11 (Sept. 2014), pp. 1044–1049. ISSN: 1476-1122. DOI: 10.1038/nmat4087.

[202] J. Gonnissen et al. "Atom-counting in High Resolution Electron Microscopy:TEM or STEM – That's the question". In: *Ultramicroscopy* 174 (Mar. 2017), pp. 112–120. ISSN: 03043991. DOI: 10.1016/j.ultramic.2016.10.011.

[203] F.-R. Chen, C. Kisielowski, and D. Van Dyck. "3D reconstruction of nanocrystalline particles from a single projection". In: *Micron* 68 (Jan. 2015), pp. 59–65. ISSN: 09684328. DOI: 10.1016/j.micron.2014.08.009.

[204] F.-R. Chen, D. Van Dyck, and C. Kisielowski. "In-line three-dimensional holography of nanocrystalline objects at atomic resolution". In: *Nat. Commun.* 7 (Feb. 2016), p. 10603. ISSN: 2041-1723. DOI: 10.1038/ncomms10603.

[205] F. -R. Chen, C. Kisielowski, and D. Van Dyck. "Prospects for atomic resolution in-line holography for a 3D determination of atomic structures from single projections". In: *Adv. Struct. Chem. Imaging* 3.1 (Dec. 2017), p. 8. ISSN: 2198-0926. DOI: 10.1186/s40679-017-0041-6.

[206] Olga Russakovsky et al. "ImageNet Large Scale Visual Recognition Challenge". In: *Int. J. Comput. Vis.* 115.3 (Dec. 2015), pp. 211–252. ISSN: 0920-5691. DOI: 10.1007/s11263-015-0816-y.

Appendix A

Included papers

A.1 Paper 1

Accuracy of surface strain measurements from transmission electron microscopy images of nanoparticles

Jacob Madsen, Pei Liu, Jakob B. Wagner, Thomas W. Hansen, and Jakob Schiøtz

Adv Struct Chem Imaging. 2017; 3(1): 14.

Madsen *et al. Adv Struct Chem Imag (2017) 3:14*
DOI 10.1186/s40679-017-0047-0

Advanced Structural and Chemical Imaging

RESEARCH Open Access

Accuracy of surface strain measurements from transmission electron microscopy images of nanoparticles

Jacob Madsen[1*], Pei Liu[2], Jakob B. Wagner[2], Thomas W. Hansen[2] and Jakob Schiøtz[1]

Abstract

Strain analysis from high-resolution transmission electron microscopy (HRTEM) images offers a convenient tool for measuring strain in materials at the atomic scale. In this paper we present a theoretical study of the precision and accuracy of surface strain measurements directly from aberration-corrected HRTEM images. We examine the influence of defocus, crystal tilt and noise, and find that absolute errors of at least 1–2% strain should be expected. The model structures include surface relaxations determined using molecular dynamics, and we show that this is important for correctly evaluating the errors introduced by image aberrations.

Keywords: High-resolution transmission electron microscopy, Strain mapping, Nanoparticles, Surface strain

Background

The surface lattice strain in nanostructures as a topic of research has gained increased interest in recent years due to its significant impact on many material properties. As an example, surface strain is a possible tunable parameter that can be used to optimize the adsorption energies of surfaces for a particular catalytic reaction [1]. Platinum-based oxygen reduction catalysis is improved by weakening the binding of adsorbed oxygen intermediates by 0.1 eV, this can be achieved by a 2% compressive strain [2]. Strain in nanoparticles can be generated by a variety of sources: particle size, shape, twinning, by the lattice mismatch between metals in multimetallic core–shell nanoparticles or it can be induced by the supporting substrate [3]. Characterizing the influence of these effects requires a technique capable of measuring structural information at atomic resolution.

High-resolution transmission electron microscopy (HRTEM) has become a routine tool for determining the structure of materials at an atomic scale [4]. TEM is particularly attractive due to the ability to map local strain. However, TEM images are the result of a complex diffraction and aberration-limited imaging process, and hence considerable care needs to be shown when extracting quantitative information.

An approach to overcome this is to iteratively compare experimental images with simulations [5, 6]; imaging parameters and model structure of the sample are refined until the simulated and experimental image match. This method has been successfully applied to determine various structures including surfaces. Another solution is to reconstruct the exit wave from a focal series, to eliminate the effect of aberrations [7]. However, the additional complexity added by such methods has limited their use. Instead an often used approach is to obtain the atomic positions directly from the experimental images. The positions of the intensity extrema within the image depend on imaging conditions, orientation and sample thickness, hence they do not necessarily coincide with the atomic positions. However, in the periodic part of a solid, a constant spatial relationship can still be assumed between the image and the atomic positions. This assumption breaks in areas with thickness variations, defects and in particular in the vicinity of surfaces and interfaces [8] and thus a systematic assessment of the accuracy is needed for these cases.

*Correspondence: jamad@fysik.dtu.dk
[1] Department of Physics, Technical University of Denmark, Fysikvej, Building 311, 2800 Kongens Lyngby, Denmark
Full list of author information is available at the end of the article

A first investigation to determine the accuracy with which surface strain could be determined was undertaken by Marks [9]. Image simulations were used to compare actual relaxations, in the input structural models, with apparent relaxations, measured from the corresponding simulated images. He found that there was a linear relationship between apparent and real strain, with a constant outward shift of about 5%. He also demonstrated that the true positions of atomic columns at the surface could be determined within 0.2 Å, corresponding to 5% of the lattice parameter of gold. This investigation was done before the invention of the spherical aberration corrector, which today has made it feasible to measure surface relaxations on the order of a few percent.

Newer investigations on the accuracy of strain analysis directly from HRTEM images have focused on interfaces in heterostructures [8, 10–12]. The error in such cases was found to be as low as 0.5% [13, 14]. Using a new technique based on Fourier transforming several overlapping sliding windows, it has been demonstrated that picometric precision and accuracy of interatomic distances can be achieved for measurements inside periodic solids [15]. However, these studies do not investigate surfaces and generally assume a uniform thickness. Moreover, in all these cases the strain distributions were fundamentally 2D, i.e. the atomic columns were mainly displaced in the plane perpendicular to the zone axis. This is different from nanoparticles where the true 3D strain is projected as a 2D image.

The literature has several examples of studies using aberration-corrected microscopy that includes measurements of strain in nanoparticles, and in the vicinity of surfaces, these measurements are often backed by comparison with a simulation that approximates the experimental structure and microscope conditions [16–19]. The general conclusion is that the erroneous surface strain due to imaging aberrations is much smaller in aberration-corrected images than the 5% found by Marks. However, these studies lack a systematic analysis of the sensitivity to experimental variables.

In the present work, we evaluate the accuracy of strain analysis directly from simulations of aberration-corrected HRTEM images focusing on surfaces of nanoparticles. The simulated objects are gold nanoparticles, which in addition to being a topic of research in their own right, provides a model structure that has different exposed surfaces and a linear thickness gradient. We examine the influence of four different effects: defocus, particle size, crystal tilt and noise, and we investigate what accuracy can be expected under which imaging and sample conditions.

Methods

Image simulation

Model and temperature effects

The overall shape of the model clusters was determined using Wulff constructions. The models were placed in a computational cell with 5 Å vacuum on all sides of the particle, see Fig. 1. Real metal surfaces are not simply ideally truncated crystals; experimental studies have demonstrated that the surface layer of many clean transition metals relaxes inward [20], while expansion of the top layer has been found for some surfaces of noble metals [21], including the {111} facets of gold. It has been proposed that expansive surface strains in small decahedral gold nanoparticles are a contributor to their catalytic activity [22].

In this study the ideal crystals were relaxed using molecular dynamics (MD) with an empirical potential. The interactions between the atoms were calculated with the charge-optimized many body (COMB) potential [23]. The potential parameters were fitted with a high priority for surfaces and nanoparticles, and hence reproduce the experimental surface relaxations of gold quite well. For an infinitely extended {111} surface, the potential predicts a 1.2% surface expansion of the top layer, which is close to the experimental value of 1.3% [24]. For {100} surfaces an

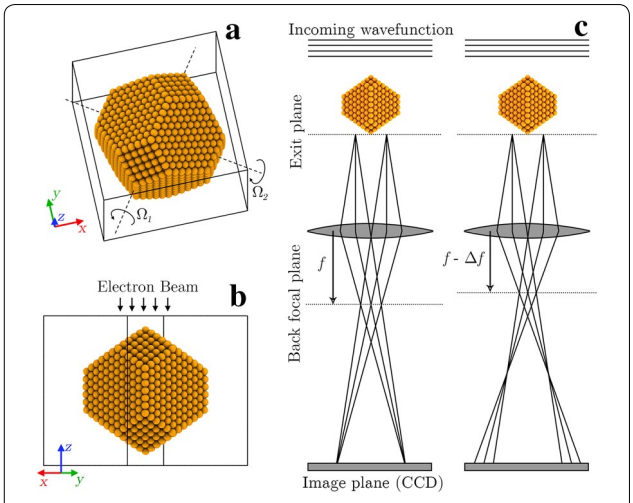


Fig. 1 **a**, **b** Model gold nanoparticle containing 1925 atoms with a diameter of 4 nm. The electron beam travels in the negative *z*-direction. The full lines indicate the computational cell and the dashed lines indicate the rotation axes denoted Ω_1 and Ω_2. **c** The HRTEM images are simulated by propagating the incoming plane wave through the sample potential using the multislice algorithm. The resulting wave at the exit plane is transferred through the objective lens to the detector using the CTF. Defocus is given relative to the bottom of the nanoparticle, with a positive defocus referring to propagation toward the detector

inward relaxation of 1.1% is predicted. There is no corresponding experimental value; however, the prediction is close to 1.2% [25] and 1.51% [26] calculated with density functional theory.

The effect of finite temperature is included using the frozen phonon approximation [27]. This is a semi-classical model based on the assumption that a single high-energy electron passing through the specimen at about half the speed of light can only probe a single frozen "snapshot" of the vibrating crystal. The image is produced by averaging incoherently over many snapshots where the atoms are slightly displaced from their equilibrium positions. The frozen phonon model has been shown to be numerically equivalent to the full quantum-mechanical treatment of the inelastic phonon scattering process [28]. The snapshots are typically determined using the Einstein approximation; however, we chose to use random steps from a constant temperature MD simulation using Langevin dynamics at 300 K [29]. We only used steps after the initial equilibration and the simulation was run for long enough to properly represent the thermal distribution of the atomic positions. We found that the simulated images are converged when ~ 40 snapshots are included in the averaging.

During a MD simulation the projected atomic positions follow a 2D normal distribution. The standard deviation of this distribution is around 0.05 Å or approximately 2% of the distance between the columns. The standard deviation of the distributions is not identical for all columns. It can be approximately 30–40% larger for some surface and corner atoms (see Additional file 1: Figure S3). We find that the difference between the mean relaxed positions and the mean positions obtained from a thermal average is just a constant thermal expansion of the entire crystal.

Diffraction and objective lens aberrations

The exit waves were simulated with the multislice algorithm [30] using the QSTEM code [31]. This code has been interfaced with Python and utilizes the atomic simulation environment [32] for setting up model structures, providing a single environment for building models, simulating and analysing images. The code is publicly available.[1] We have also made code available for directly recalculating and analysing a selection of the results from this paper.

The electrostatic potential of the sample was generated using the independent atom model with the parametrizations of Rez et al. [33]. The potential was generated on a 3D grid before slicing, allowing for accurate simulations of tilted samples. Aberrations due to the objective lens were included by Fourier space multiplication with the contrast transfer function (CTF). The effect of a finite source size and energy spread (i.e. partial spatial and temporal coherence) was included in the Quasi-coherent approximation where envelopes are applied to the wave function [34]. The imaging process is illustrated in Fig. 1c.

The microscope conditions were modelled after an image aberration-corrected FEI Titan microscope operated at 300 kV. Unless otherwise stated, the third-order spherical aberrations were set to $C_s = -10$ µm and all other aberrations except for defocus were set to zero. Other aberrations are generally not negligible in aberration-corrected microscopy; however, we chose to neglect them in order to keep the degrees of freedom limited. We tested the stability of our results to inclusion of additional aberrations, in particular twofold astigmatism on the order of 5–10 nm and 5th-order spherical aberrations on the order of 2.5 mm. While some results change slightly, we found that inclusion of additional aberrations does not change our conclusions in significant ways.

The focal spread was $\Delta = 2.9$ nm and the convergence angle was set to 15 mrad. The sampling used for the simulations was at least 0.05 Å/pixel, and when needed the large simulated images were downsampled using bilinear interpolation.

MTF and thermal magnetic noise

A single electron can cause a signal in more than one pixel of the CCD due to multiple scattering in the scintillator material. This effect can be described by the modulation-transfer function (MTF). A typical MTF can be parametrized as the sum of a Gaussian and an exponential [35]

$$\mathrm{MTF}(q) = a\exp(-bq) + (1-a)\exp(-c^2q^2), \qquad (1)$$

where q is the spatial frequency and the parameters are taken as $a = 0.58$, $b = 2.5$ Å and $c = 5.9$ Å.

An additional blurring can be caused by all kinds of noise that lead to a random deflection of the image relative to the detector. The origin of these aberrations are vibrations and drift of the stage, time-dependent fields resulting from instabilities of the lens currents and in particular thermal magnetic noise resulting from magnetic fields due to eddy currents in the material of the lenses [36]. The blurring is modelled by a Gaussian envelope on the intensity distribution [37]

$$\mathcal{N}(q) = \exp(-(2\pi\sigma)^2q^2), \qquad (2)$$

where σ denotes the standard deviation, and a value of $\sigma = 0.25$ Å has been assumed. It has been shown that including the MTF and a Gaussian blur can account for the so-called Stobbs factor [38], the ubiquitous contrast

[1] https://github.com/jacobjma/PyQSTEM.

mismatch between experimental and simulated images [39]. Since these effects can drastically reduce the contrast, they are important to include for accurately quantifying the influence of noise.

Finite electron dose

We assume that the noise is dominated by shot noise, and hence the measured electron count in each pixel can be modelled by a Poisson distribution [40]. The average number of electrons N collected by the ith detector pixel is given by

$$N_i = D\delta^2 I_i, \tag{3}$$

where D is the dose in electrons per area, δ is the sampling and I_i is the probability for an electron hitting the i'th pixel. The signal-to-noise ratio of the whole image is given by [41]

$$\mathrm{SNR} = \frac{\bar{N}}{\sigma(N)}, \tag{4}$$

where $\bar{N}$ is the average number of electrons per pixel and $\sigma(N)$ is the standard deviation of the number of electrons collected by each pixel. In the limit of low dose this can be reduced to [42]

$$\mathrm{SNR} = \sqrt{\bar{N}} = \sqrt{DI}\delta, \tag{5}$$

whereas in the limit of high dose other sources of noise are dominant (e.g. thermal noise) and the SNR becomes constant. We are only including shot noise in the simulations.

Strain analysis

There are several different approaches for obtaining strain directly from HRTEM images. The methods can broadly be classified into three different types: direct measurement of interatomic distances in real space [43, 44], extraction of the lattice by comparison to a template [45] and analysis in Fourier space [46]. The results of the different approaches are similar inside periodic structures, but can differ in the presence of defects [44]. In this paper the real space method is used, since it has the most straight forward interpretation for surfaces, where the results of Fourier space analysis are very opaque. A comparison between real and Fourier space analysis, using geometric phase analysis (GPA), is provided as supplementary information (see Additional file 1: Figure S4).

The most critical step in the real space approach is to determine the positions of the lattice points. There are several ways of defining these positions. However, the simplest way is to define them as the position of the intensity extrema, assumed to correspond with an atomic column. If the lattice points do not correspond to single intensity peaks, they can instead be found using a cross-correlation of the image with a template motif [8]. The intensity extrema are found at sub-pixel accuracy by fitting a 2D function, usually a polynomial or a Gaussian, to the neighbourhood of each peak and setting the derivatives to zero [44]. It is also possible to define the lattice positions from the centre of mass of the intensity distributions [47]. The methods agree if the intensity distributions are symmetric. However, this is not necessarily the case close to asymmetries in the lattice, such as an interface. A comparison of the two methods of measuring the atomic positions is included as supplementary information. The conclusion is that the methods lead to slightly different errors; however, the magnitude of the errors is essentially the same.

The peak pairs algorithm [44] is the most popular method for finding strain from a set of 2D lattice points from HRTEM images. For the calculation of strain at every lattice point, the peak pairs algorithm uses only two lattice vectors. We have found that an approach using a larger number of lattice vectors is significantly more stable in the presence of noise. For an fcc crystal in the [48] zone axis, this method uses the four nearest and two second nearest neighbours to find the strain at any lattice point in the bulk. Another advantage of this method is that it allows us to determine the strain for lattice points at all surfaces and corners, which is not possible with the standard implementation of the peak pairs algorithm. The routines used for strain analysis, including a rudimentary implementation of GPA, are implemented in Python and made available as open source.[2]

The strain is computed at each lattice point, by comparing the positions of the neighbouring lattice points in an ideal template lattice to the corresponding measured lattice points. In practice, this is done by finding the optimal affine transformation, $\mathbf{A}$, between the two sets of vectors, see Fig. 2. In general finding $\mathbf{A}$ is an overdetermined problem, hence it is found as the best fit to a least-squares fit of the form:

$$r = \min_{\mathbf{A}} \sum_i^N \|\mathbf{A}\mathbf{v}_i - \mathbf{w}_i\|, \tag{6}$$

where r is the residual term, v_i and w_i are vectors containing the ideal and actual lattice vectors, $\mathbf{A}$ is the affine transformation and $\|\cdot\|$ denotes the Euclidean norm. The orientation and elastic strain matrices can be extracted from $\mathbf{A}$ via a left-sided polar decomposition of the deformation gradient

$$\mathbf{PU} = \mathbf{A}, \tag{7}$$

[2] https://github.com/jacobjma/structural-template-mapping.

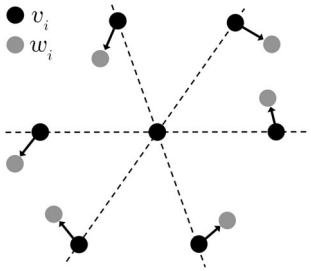


Fig. 2 The black points indicate the ideal lattice for an fcc crystal in the [48] zone axis. The grey points are the positions of slightly displaced lattice points for a strained crystal. The strain at the central lattice point is calculated by finding the optimal affine transformation between the black and grey points, denoted by *v* and *w*, respectively

where **U** is an orthogonal right-handed matrix (the rotation matrix), and **P** is a symmetric matrix (the elastic strain matrix). Finding the correspondence between $\boldsymbol{v}$ and $\boldsymbol{w}$ is done using a branch and bound search method. A similar 3D equivalent of the method is described by Larsen et al. [48].

To limit the amount of results that have to be shown, we will usually just show the planar strain, ϵ_{p}, calculated as the average of the normal strains in the *x*- and *y*-direction

$$\epsilon_{\mathrm{p}} = \frac{1}{2}(\epsilon_{xx} + \epsilon_{yy}). \tag{8}$$

Surface relaxations are the strain at the outermost atoms in the direction perpendicular to the same surface. Hence, the surface relaxation associated with an atom on a surface perpendicular to the unit vector $\hat{\boldsymbol{n}}$ is found as

$$\epsilon_{\hat{\boldsymbol{n}}} = \hat{\boldsymbol{n}}^{\mathrm{T}} \boldsymbol{\epsilon} \hat{\boldsymbol{n}}. \tag{9}$$

We are mainly interested in the strain measurement errors, but to define the errors, we first need to define the true strain. An image provides a single viewpoint of the structure, where each atomic column appears as a dot, hence we can only hope to measure an average column position for the atoms belonging to each column. Defining these averages to be the true column positions, the corresponding planar strain will be denoted as $\epsilon_{\mathrm{p,\,true}}$. The strain calculated from the positions of the maxima in the matching image will be denoted as$\epsilon_{\mathrm{p,\,measured}}$. From these definitions, we define the error of a strain measurement as

$$\mathrm{error}\,(\epsilon_{\mathrm{p}}) = \epsilon_{\mathrm{p,\,measured}} - \epsilon_{\mathrm{p,\,true}}. \tag{10}$$

Results

Influence of relaxations and temperature effects

When image simulations are used to estimate errors due to aberrations, it is a common practice to use a model of an unrelaxed crystal, under the assumption that the errors caused by these aberrations are insensitive to the small difference between the unrelaxed and relaxed crystal [16–19]. Our results demonstrate that this assumption is invalid in general.

The comparison in Fig. 3 shows the difference between results based on an ideal crystal, a relaxed crystal and an average over thermal vibrations. There is a substantial difference between the exit wave intensities. This difference is less obvious in the final images; however, it is large enough to have an impact on the measured strain and more notably on the measurement errors. This means that using the ideal particle to calibrate a strain measurement would lead to wrong conclusions about the measurement errors.

The origin of the errors is deviations from the constant spatial relationship between the image and the underlying projected potential. The peaks are generally more asymmetric for both the image resulting from a relaxed crystal and from a thermal average of crystals, and these small irregularities in the symmetry of adjacent intensity peaks can cause large measurement errors, as illustrated in Fig. 4. All results in the following sections will be based on simulations where temperature effects are included. We also note the $\sim 50\%$ reduction of the image contrast due to thermal vibrations, making the influence of temperature on the image contrast approximately as important as the MTF and thermal magnetic noise.

The strain calculated from the true average projected column positions is shown in Fig. 5a for three different particle diameters. The strain calculated from the projected positions seems to show a significant compressive strain in the bulk of the particle; however, this is misleading. Figure 5b shows the strain calculated directly from the full 3D model for a slice through the centre of the nanoparticle; comparing the strain in the 3D model to the projected strain reveals that the apparent bulk compressive strain is due to relaxations closer to the front and back surface. Hence, even disregarding image aberrations, comparing Fig. 5a, b shows that care has to be taken, when interpreting strain measurements from HRTEM images. The errors in the following sections are calculated with respect to the strain in the projected positions and are thus mainly due to image aberrations.

Influence of defocus

The top row of Fig. 6 shows simulated images at different defocus and the bottom row shows the error in the planar strain measured from these images. The smallest defocus shown is 4.5 nm since contrast inversion begins to take effect for a smaller defocus. We present results for only a positive defocus, which leads to images with bright spots at the positions of the atomic columns. We have

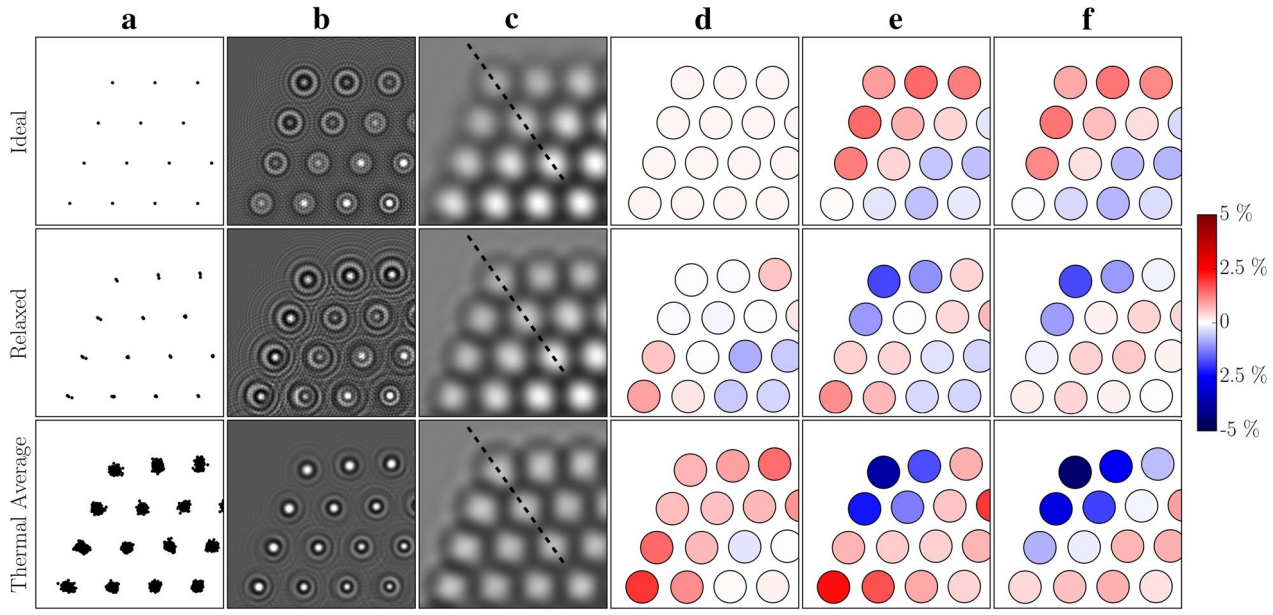


Fig. 3 From top to bottom, the rows contain results relating to an ideal crystal, a relaxed crystal and a thermal average of crystals. Each of the panels show a small section of the corner between two {111} facets. Along the columns we show: **a** the projected positions of the atoms, all the positions used in the thermal average are included. **b** Intensities of the exit waves. **c** Simulated images for a defocus $\Delta f = 14.5$ nm. **d** The true planar strain, $\epsilon_{\mathrm{p,true}}$, i.e. the strain calculated directly from the projected positions of the model crystal. The colour coding shared by all the columns is shown to the right of the figure. **e** The measured planar strain, $\epsilon_{\mathrm{p,measured}}$, i.e. strain calculated from the measured positions of the intensity maxima in simulated images. **f** The measurement error of the planar strain, error(ϵ_p), calculated as the difference between the strain shown in the two preceding columns

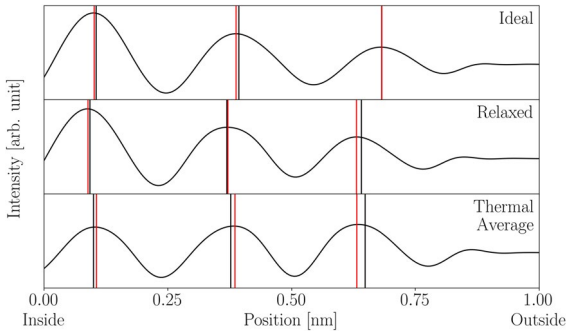


Fig. 4 Slices along the dashed lines in Fig. 3. The black vertical lines indicate the true atomic positions and the red vertical lines indicate the corresponding measured maxima positions. For the ideal model the measured distance between the outermost peaks is too large by 0.006 nm or 2% of the interatomic distance in the slice direction. For the images that include temperature effects, the same measurement is too small by 0.025 nm or 10% of the interatomic distance

obtained results for negative defocus as well, where the atoms appear as black spots on a lighter background. The results are shown in Additional file 1: Figure S5.

A defocus of 8.5 nm results in planar strain errors smaller than 1% everywhere, while a defocus of 12.5 nm causes significant errors at the {100} facets. Due to the sign and location of these errors, they could easily be mistaken for real surface relaxations. The errors generally stay small for columns not at the surface; however, at larger defocus some errors start to appear, generally following the thickness gradient.

The error in the measured surface relaxations averaged across the facets for the uppermost atomic layers is shown as a function of defocus in Fig. 7. Since this error can vary quite a bit across the {111} facets, we also show the corresponding standard deviation. Results for 3 different particle sizes are shown, from a diameter of ~ 2 nm to a diameter of ~ 6 nm.

For the {100} facet the error is almost zero up to a defocus of 8.5 nm, across all three particle sizes. Meanwhile the error for the {111} facet never becomes smaller than 1% for the 4 nm particle, which is approximately the same magnitude as the actual relaxations. For both facets and all sizes, the errors stay below 2% up to a defocus of ~ 11 nm, where the mean error increases sharply at the {100} facets. The mean error does not increase as drastically for the {111} facets. On the other hand, the standard deviation does increase. This is mainly due to the thickness variation along these facets.

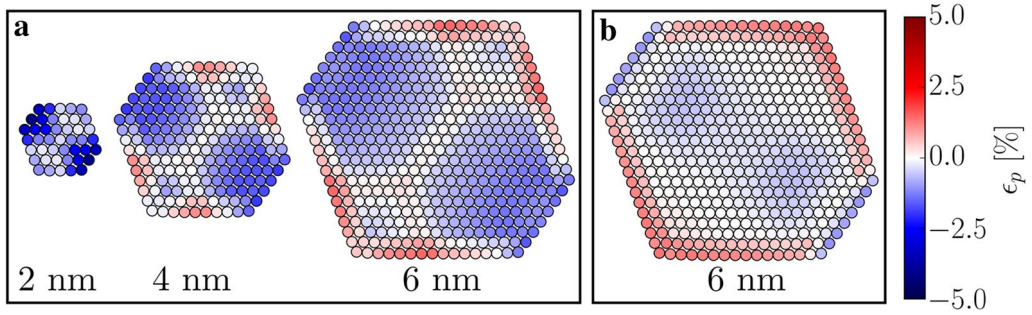


Fig. 5 **a** The "true" planar strain, $\epsilon_{p,\,true}$, calculated from the average projected column positions of the model, for three different nanoparticle diameters. **b** The actual planar strain for a slice through the 3D model

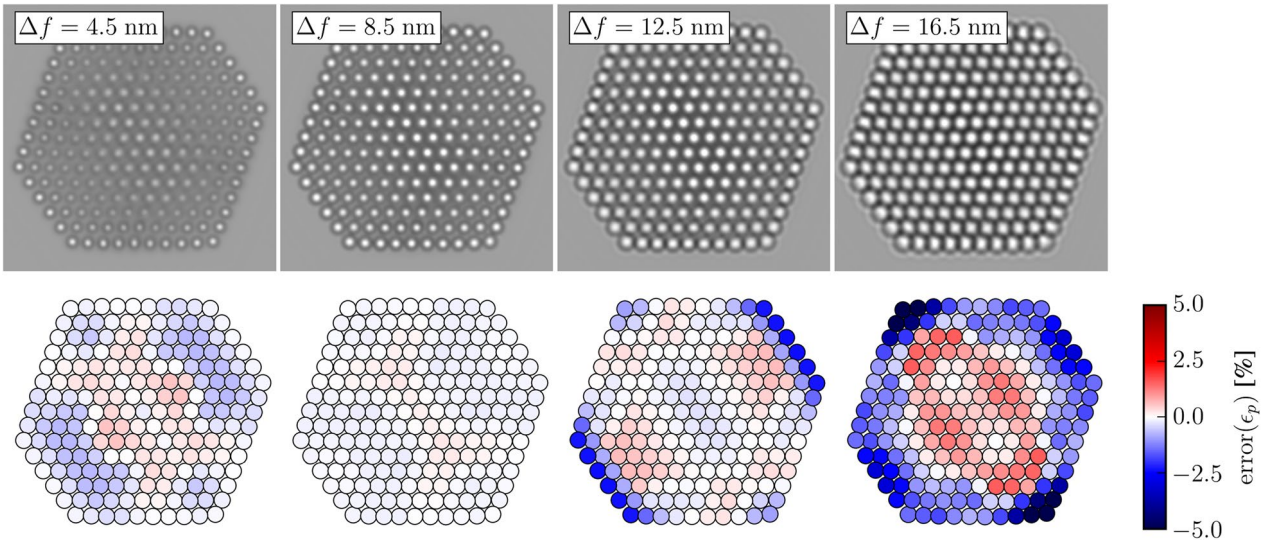


Fig. 6 The top row shows simulated images for a nanoparticle with a diameter of 4 nm. The bottom row shows the corresponding distribution of errors in the planar strain. The defocus is different in each column, as indicated in the figure

Influence of tilt

It is unavoidable that the sample will be slightly tilted relative to the ideal zone axis. Figure 8 shows the distribution of errors in the planar strain for increasing tilt, α, around the Ω_1-axis. At tilt $\alpha = 1.0°$, the errors have changed very little compared to the untilted crystal, though the appearance of the image have changed in the central part of particle, this is due to an effective diminishing of the projected potential, as have been reported elsewhere [49]. The errors stay small up to a tilt $\alpha = 2.0°$, but increase sharply in the centre of the nanoparticle between $\alpha = 2.0°$ and $\alpha = 3.0°$. The error introduced by tilt is very dependent on the height of the atomic columns, since the length of the footprint of the projection of a tilted column increases linearly with its height. Only one direction of tilt is shown; however, the trends are similar for other tilt directions. One other tilt directions is included as Additional file 1: Figure S6.

Figure 9 shows the effects of tilt on the errors in the measured surface relaxations for a defocus $\Delta f = 8.5$ nm. The tilt has a relatively limited impact on the measured surface relaxations. The mean and standard deviation of error changes by at most 1% over the entire tilt range. The effects of tilt on the strain measurements are very dependent on defocus. For example at a defocus $\Delta f = 14.5$ nm, the mean surface relaxation error changes by more than 6% at the {100} facets, a plot showing this is shown in Additional file 1: Figure S7.

Influence of noise

The evolution of the object visibility with respect to the sampling and dose is shown in Fig. 10a. At a dose of 10^2

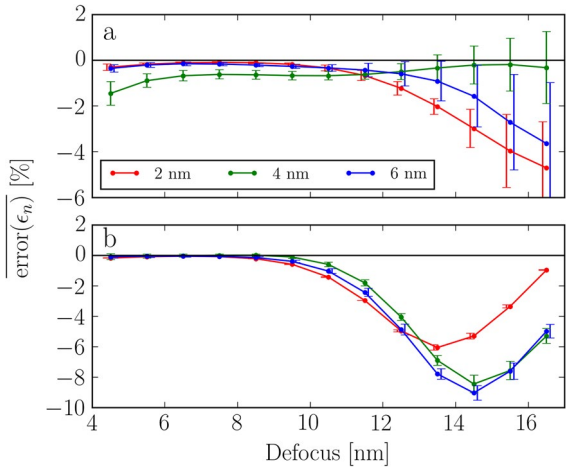


Fig. 7 The error in the measured surface relaxations averaged across the facets as a function of defocus, for the three particle sizes given in the legend, for: **a** the {111} facets and **b** the {100} facets. The bars indicate the standard deviation of the errors in the surface relaxation error across the facets. The bars are shifted slightly from the points for visual clarity

$e^-/\text{Å}^2$ the object is barely visible, while the images are essentially unaffected by noise at $10^5\, e^-/\text{Å}^2$.

Noise removal is essential to obtain the stable polynomial fits necessary for sub-pixel resolution; hence we show the same noisy images after application of a Wiener filter in Fig. 10b [50]. The regularization of the filter was chosen to be optimal for each of the different samplings, but was not changed with the amount of noise.

To determine the influence of dose on the errors in strain measurements, we simulate a statistically representative ensemble of images, $K = 300$, with different distributions of noise. The error due to noise is quantified using the mean absolute error, MAE, over the ensemble of images for each lattice point

$$\text{MAE}_i = \frac{1}{K}\sum_{k=0}^{K} \left|\epsilon_{k,i} - \epsilon_{\infty,i}\right|, \tag{11}$$

where $\epsilon_{k,i}$ is the planar strain at the i'th lattice point measured from the kth noisy image and $\epsilon_{\infty,i}$ is the corresponding measured strain without noise. Since the automatic polynomial fitting can fail at low doses, extreme outliers have been removed before taking the average. Figure 11 shows the distribution of the MAE across a nanoparticle, there is a fairly large difference between the MAE for different lattice points, varying by a factor of three between the centre of the particle and a corner. The reason for this is mainly that the strain at surfaces is determined on the basis of fewer surrounding lattice points. The strain at corner atoms is determined on the basis of just three neighbours, while the measurements in the centre rely on twice that number of neighbours.

The MAE at three chosen lattice sites as a function of dose is shown in Fig. 12. We find a simple approximate empirical relationship, assuming constant sampling, between the MAE and the dose

$$\text{MAE} \propto \frac{1}{\sqrt{D}} \propto \frac{1}{\text{SNR}}, \tag{12}$$

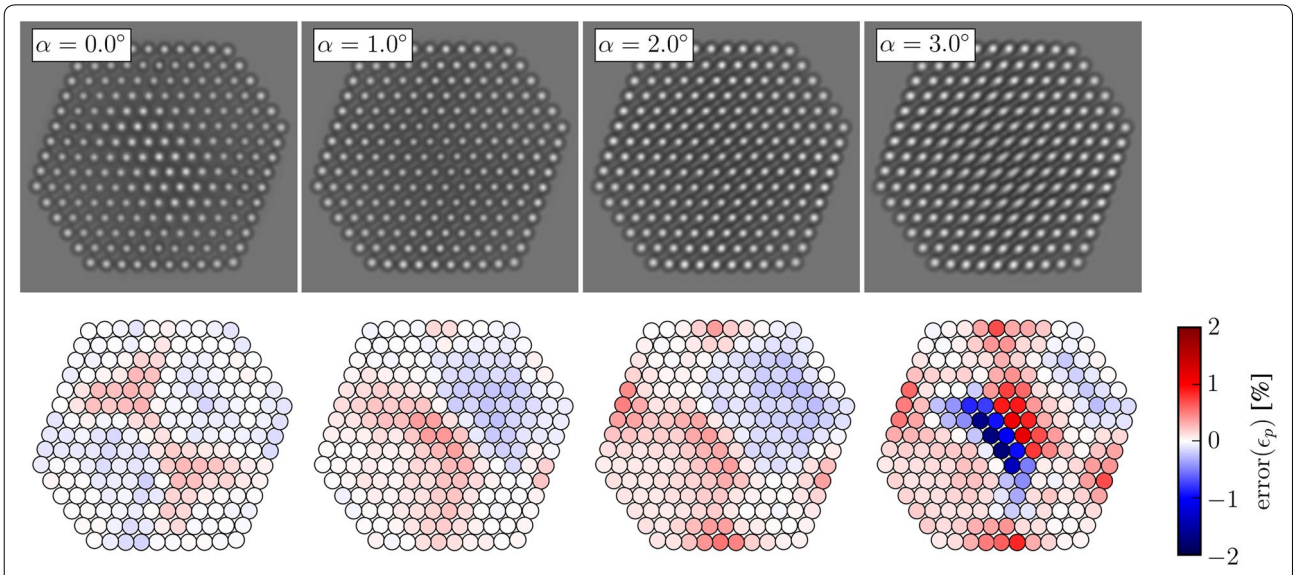


Fig. 8 The top row shows simulated images at increasing tilt around the $\Omega_\top$-axis for a defocus $\Delta f = 8.5$ nm. The bottom row shows the error in the planar strain at each lattice point measured from these images

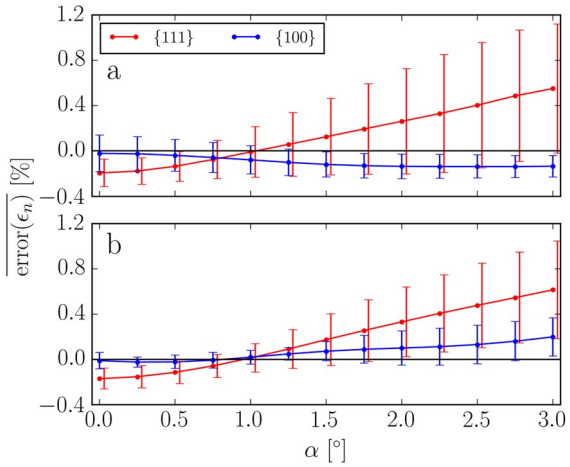


Fig. 9 The error in the measured surface relaxations averaged across the facets as a function of tilt, around the axes (**a**) Ω_1 and (**b**) Ω_2. The defocus was $\Delta f = 8.5$ nm. The curves are for the {100} and {111} facets, as indicated by the legend

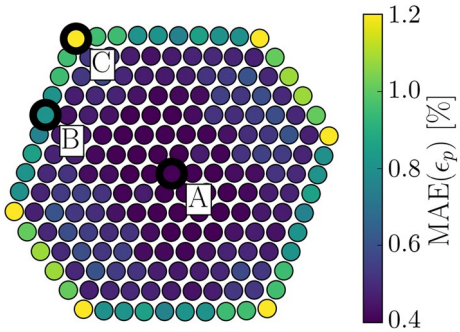


Fig. 11 The MAE of the planar strain due to noise at each lattice point for a Wiener filtered noisy image at a sampling of 0.2 Å/pixel, a dose of $10^3\,e^-/\text{Å}^2$ and a defocus $\Delta f = 8.5$ nm

where the constant of proportionality is determined by the number of neighbours, local image contrast and sampling. The second approximate proportionality assumes low dose and is due to Eq. (5).

Given that the SNR depends linearly on the sampling [see Eq. (5)], the expression above might lead one to expect that coarser sampling would give smaller MAE. This is however not the case, as shown in Fig. 13 where the MAE is plotted as a function of sampling for different doses. The relationship is fairly constant though a sampling of 0.2 Å/pixel is better than both a rougher or a finer sampling. The main reason that there is no decrease in the MAE as the sampling gets coarser is that the better SNR is compensated by a smaller number of pixels across each peak available for polynomial fitting.

In the previous sections, we saw that the defocus should be kept small to obtain strain measurements that are relatively unaffected by aberrations. The disadvantage of this is that phase contrast imaging relies on the additional phase added by the objective lens, and hence a too small defocus will negatively impact the image contrast. This effect is illustrated in Fig. 14 where the change in the visibility of the nanoparticle is shown with respect to

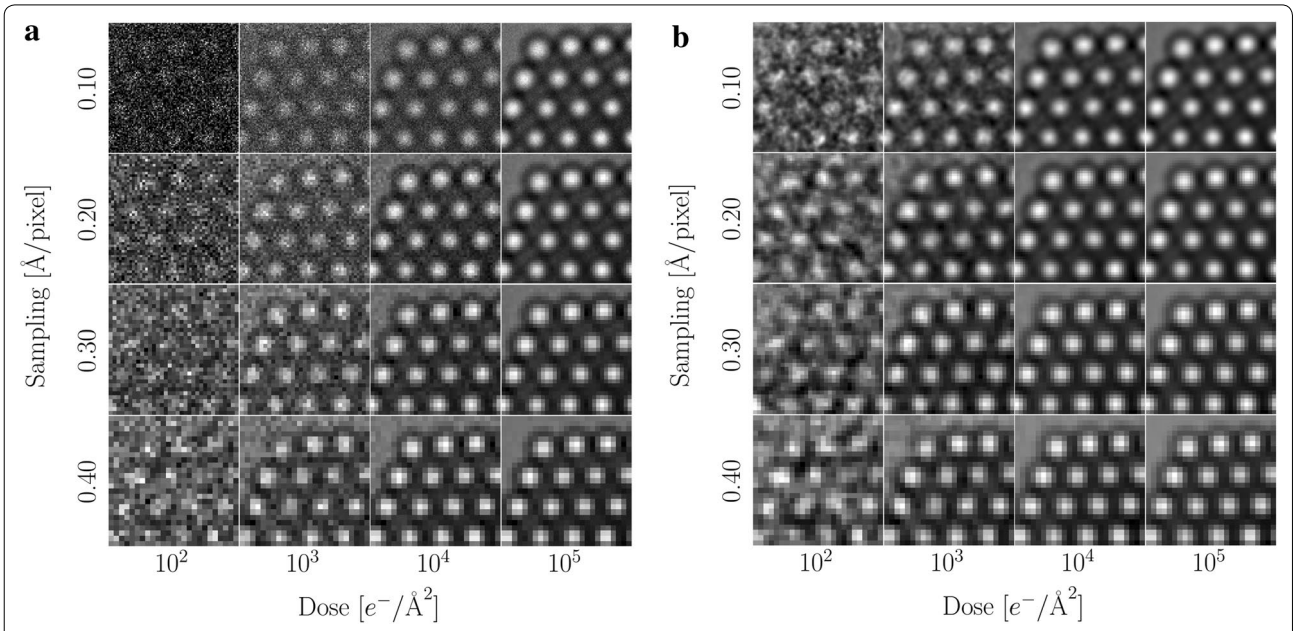


Fig. 10 **a** Sections of the simulated images at a defocus of $\Delta f = 8.5$ nm for different doses and samplings. All images are mapped onto the same range of grey levels. **b** The same images after applying a Wiener filter

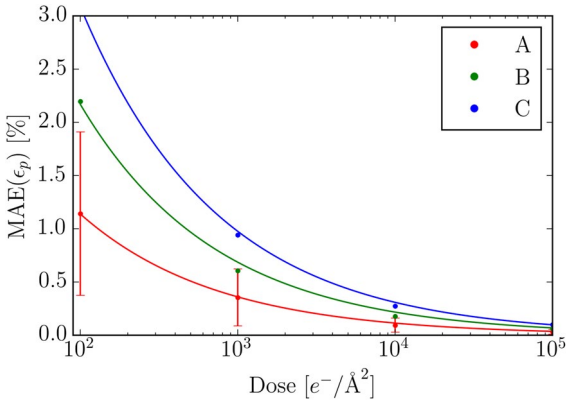


Fig. 12 The MAE as a function of the dose for the three lattice points, *A*, *B* and *C*, as indicated in Fig. 11. The dots show the MAE calculated from the simulated images and the full lines are curves of the form given by Eq. (12), where the constant of proportionality has been fitted to the dots. The bars indicate the standard deviations, which for visual clarity are shown only for lattice point *B*, proportionally the standard deviations are similar for the other lattice points

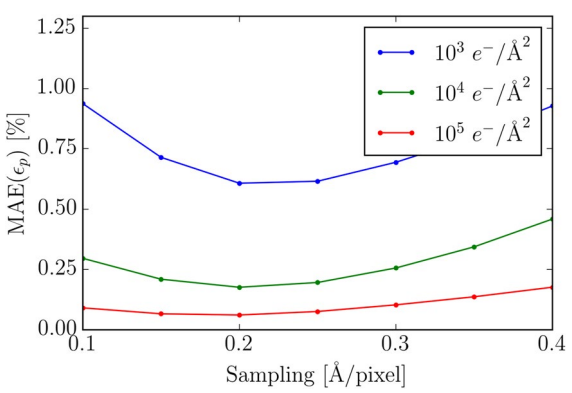


Fig. 13 The MAE as a function of the sampling for five different doses at the *B* lattice point (see Fig. 11). The defocus was $\Delta f = 8.5$ nm and the sampling was 0.2 Å/pixel

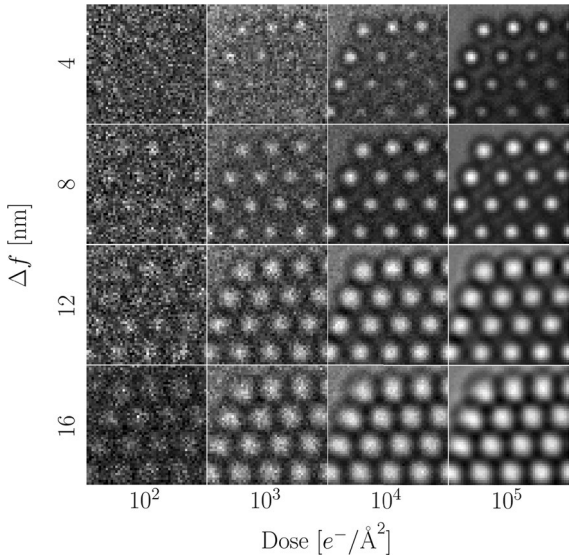


Fig. 14 Sections of simulated HRTEM images for different doses and defocus at a sampling of 0.2 Å/pixel. All images are mapped onto the same range of greys

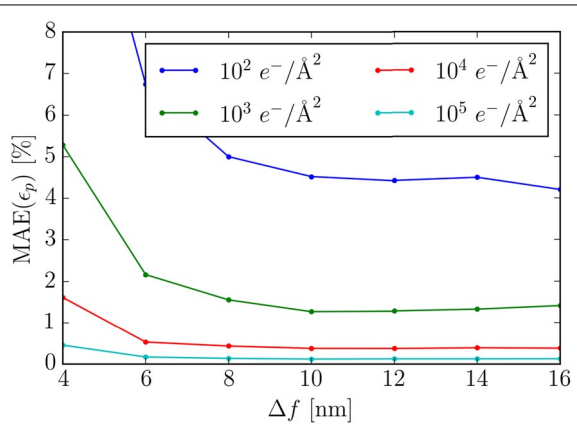


Fig. 15 The MAE as a function of defocus for four different doses at a sampling of 0.2 Å/pixel at the lattice point B

defocus and dose. The corresponding errors are quantified in Fig. 15. At a low dose, the errors grow very large when the defocus is small, but even at a higher dose, errors due to noise become present when the defocus is too small. When the defocus is increased the MAE becomes smaller, however saturation is reached relatively quickly, and additional defocus beyond $\Delta f = 8$ nm does not further improve the MAE.

Conclusion

We looked at the accuracy of surface strain measurements from HRTEM images of nanoparticles. We showed that the practice of using simulations based on ideal sample models to calibrate strain measurements is problematic, since the predicted errors from such simulations do

not in general reflect the errors for an identical model that includes relaxations.

In general, the impact of the interaction between tilt, thickness and defocus on the final strain measurement is very complicated. However, we observe that if the defocus is small enough, the errors in the measured surface relaxations due to image aberrations can be kept at less than 2%, even for visually obvious tilts. This is significantly larger than the 0.5% that have been found for strain measurements inside periodic solids [13]. The main reason for the larger error is the asymmetry in the peaks close to surfaces.

In order to obtain measurements with small errors, the defocus should not be chosen solely to maximize contrast, since this will also cause large errors due to aberrations. The choice of defocus has to balance delocalization and contrast; if the defocus is too small the contrast will suffer, while if defocus is too large the image aberrations will be the main source of error.

For a dose of $10^3\ e^-/\text{Å}^2$, the optimal defocus for the gold nanoparticles is somewhere around 8.5 nm; at this defocus the errors in the surface relaxations are below 2% and the expected noise error is 1.2% with a standard deviation 0.8%.

Additional file

Additional file 1. Section S1. Measuring the center of mass. **Figure S1.** Definition of integration regions for center of mass calculations. **Figure S2.** Comparison of center of mass positions with peak positions. **Figure S3.** Magnitudes of thermal vibrations. **Figure S4.** Comparison of our method with GPA. **Figure S5.** Negative defocus measurements. **Figure S6.** Planar strain errors for increasing tilt. **Figure S7.** Surface strain errors for increasing tilt.

Author's contributions

JM did the simulations. The data analysis was done by JM with inputs from the other authors. PL performed the experimental work that inspired this article. The manuscript was written by JM with contributions from the other authors. All authors read and approved the final manuscript.

Author details

[1] Department of Physics, Technical University of Denmark, Fysikvej, Building 311, 2800 Kongens Lyngby, Denmark. [2] Center for Electron Nanoscopy, Technical University of Denmark, Fysikvej, Building 311, 2800 Kongens Lyngby, Denmark.

Acknowledgements

Not applicable.

Competing interests

The authors declare that they have no competing interests.

Availability of data and materials section

The source code and algorithms used in this article are available in the repositories https://github.com/jacobjma/PyQSTEM and https://github.com/jacobjma/structural-template-mapping.

Ethics approval and consent to participate

Not applicable.

Funding

We gratefully acknowledge funding through Grant 1335-00027B from the Danish Council for Independent Research.

Publisher's Note

Received: 8 May 2017 Accepted: 5 October 2017
Published online: 25 October 2017

References

1. Mavrikakis, M., Hammer, B., Nørskov, J.K.: Effect of strain on the reactivity of metal surfaces. Phys. Rev. Lett. **81**(13), 2819–2822 (1998). doi:10.1103/PhysRevLett.81.2819
2. Hernandez-Fernandez, P., Masini, F., McCarthy, D.N., Strebel, C.E., Friebel, D., Deiana, D., Malacrida, P., Nierhoff, A., Bodin, A., Wise, A.M., Nielsen, J.H., Hansen, T.W., Nilsson, A., Stephens, I.E.L., Chorkendorff, I.: Mass-selected nanoparticles of PtxY as model catalysts for oxygen electroreduction. Nat. Chem. **6**(8), 732 (2014). doi:10.1038/nchem.2001
3. Sneed, B.T., Young, A.P., Tsung, C.-K., Romanov, A.E., Shalkovskii, A.G., Wang, J., Ma, Y., Xia, Y., Mao, S.X., Lee, B., Shevchenko, E.V., Liu, Z., Salmeron, M., Somorjai, G.A., Somorjai, G.A., Yang, P., Stamenkovic, V.R.: Building up strain in colloidal metal nanoparticle catalysts. Nanoscale **7**(29), 12248–12265 (2015). doi:10.1039/C5NR02529J
4. Smith, D.J.: Characterization of nanomaterials using transmission electron microscopy. In: Kirkland, A.I., Hutchison, J. (eds.) Nanocharacterization, 2nd edn, pp. 1–29. Royal Society of Chemistry, London (2015). doi:10.1039/9781782621867-00001
5. Möbus, G., Schweinfest, R., Gemming, T., Wagner, T., Ruhle, M.: Iterative structure retrieval techniques in HREM: a comparative study and a modular program package. J. Microsc. **190**(1–2), 109–130 (1998). doi:10.1046/j.1365-2818.1998.3120865.x
6. Van Aert, S., De Backer, A., Martinez, G.T., den Dekker, A.J., Van Dyck, D., Bals, S., Van Tendeloo, G.: Advanced electron crystallography through model-based imaging. IUCrJ **3**(Pt 1), 71–83 (2016). doi:10.1107/S2052252515019727
7. Bals, S., Aert, S.V., Tendeloo, G.V., Ávila-Brande, D.: Statistical estimation of atomic positions from exit wave reconstruction with a precision in the picometer range. Phys. Rev. Lett. **96**(9), 096106 (2006). doi:10.1103/PhysRevLett.96.096106
8. Du, K., Phillipp, F.: On the accuracy of lattice-distortion analysis directly from high-resolution transmission electron micrographs. J. Microsc. **221**(1), 63–71 (2006). doi:10.1111/j.1365-2818.2006.01536.x
9. Marks, L.D.: Direct atomic imaging of solid surfaces - 1. Image simulations and interpretation. Surf. Sci. **139**(1), 281–298 (1984)
10. Guerrero, E., Galindo, P., Yáñez, A., Ben, T., Molina, S.I.: Error quantification in strain mapping methods. Microsc. Microanal. **13**(05), 320–328 (2007). doi:10.1017/S1431927607070407
11. Tillmann, K., Lentzen, M., Rosenfeld, R., Peralta, P., McCartney, M.R., Drucker, J., Picraux, S.T.: Impact of column bending in high-resolution transmission electron microscopy on the strain evaluation of GaAs/InAs/GaAs heterostructures. Ultramicroscopy **83**(1–2), 111–128 (2000). doi:10.1016/S0304-3991(99)00175-8
12. Rosenauer, A., Gerthsen, D., Potin, V.: Strain state analysis of InGaN/GaN—sources of error and optimized imaging conditions. Phys. Status Solidi **203**(1), 176–184 (2006). doi:10.1002/pssa.200563519
13. Chung, J., Rabenberg, L.: Effects of strain gradients on strain measurements using geometrical phase analysis in the transmission electron microscope. Ultramicroscopy **108**(12), 1595–1602 (2008). doi:10.1016/j.ultramic.2008.05.010
14. Hÿtch, M.J., Plamann, T.: Imaging conditions for reliable measurement of displacement and strain in high-resolution electron microscopy. Ultramicroscopy **87**(4), 199–212 (2001). doi:10.1016/S0304-3991(00)00099-1

15. Klinger, M., Polívka, L., Jäger, A., Tyunina, M.: Quantitative analysis of structural inhomogeneity in nanomaterials using transmission electron microscopy. J. Appl. Crystallogr. **49**(3), 762–770 (2016). doi:10.1107/S1600576716003800
16. José Yacamán, M., Ascencio, J.A., Canizal, G.: Observation of surface relaxation surface steps and surface reconstruction in gold nanorods. Surf. Sci. **486**(1), L449–L453 (2001). doi:10.1016/S0039-6028(01)01048-2
17. Fujita, T., Guan, P., McKenna, K., Lang, X., Hirata, A., Zhang, L., Tokunaga, T., Arai, S., Yamamoto, Y., Tanaka, N., Ishikawa, Y., Asao, N., Yamamoto, Y., Erlebacher, J., Chen, M.: Atomic origins of the high catalytic activity of nanoporous gold. Nat. Mater. **11**(9), 775–780 (2012). doi:10.1038/nmat3391
18. López-Haro, M., Yoshida, K., del Río, E., Pérez-Omil, J.A., Boyes, E.D., Trasobares, S., Zuo, J.M., Gai, P.L., Calvino, J.J.: Strain field in ultrasmall gold nanoparticles supported on cerium-based mixed oxides. Key influence of the support redox state. Langmuir **32**(17), 4313–4322 (2016)
19. Du, K., Ernst, F., Pelsozy, M.C., Barthel, J., Tillmann, K.: Expansion of interatomic distances in platinum catalyst nanoparticles. Acta Mater. **58**(3), 836–845 (2010). doi:10.1016/j.actamat.2009.09.061
20. Martienssen, W., Warlimont, H.: Springer Handbook of Condensed Matter and Materials Data, 1st edn, p. 1119. Springer, Berlin (2005)
21. Nichols, R.J., Nouar, T., Lucas, C.A., Haiss, W., Hofer, W.A.: Surface relaxation and surface stress of Au(111). Surf. Sci. **513**(2), 263–271 (2002). doi:10.1016/S0039-6028(02)01510-8
22. Walsh, M.J., Yoshida, K., Kuwabara, A., Pay, M.L., Gai, P.L., Boyes, E.D.: On the structural origin of the catalytic properties of inherently strained ultrasmall decahedral gold nanoparticles. Nano Lett. **12**(4), 2027–2031 (2012). doi:10.1021/nl300067q
23. Liang, T., Shan, T.-R., Cheng, Y.-T., Devine, B.D., Noordhoek, M., Li, Y., Lu, Z., Phillpot, S.R., Sinnott, S.B.: Classical atomistic simulations of surfaces and heterogeneous interfaces with the charge-optimized many body (COMB) potentials. Mater. Sci. Eng. R Rep. **74**(9), 255–279 (2013). doi:10.1016/j.mser.2013.07.001
24. Vook, R.W., Ouyang, S., Otooni, M.A.: X-ray measurement of the (111) surface relaxation in gold. Surf. Sci. **29**(1), 277–290 (1972). doi:10.1016/0039-6028(72)90084-2
25. Yu, B.D., Scheffler, M.: Physical origin of exchange diffusion on fcc(100) metal surfaces. Phys. Rev. B **56**(24), 15569–15572 (1997). doi:10.1103/PhysRevB.56.R15569
26. Singh-Miller, N.E., Marzari, N.: Surface energies, work functions, and surface relaxations of low-index metallic surfaces from first principles. Phys. Rev. B **80**(23), 235407 (2009). doi:10.1103/PhysRevB.80.235407
27. Loane, R.F., Xu, P., Silcox, J.: IUCr: Thermal vibrations in convergent-beam electron diffraction. Acta Crystallogr. Sect. A Found. Crystallogr. **47**(3), 267–278 (1991). doi:10.1107/S0108767391000375
28. Van Dyck, D.: Is the frozen phonon model adequate to describe inelastic phonon scattering? Ultramicroscopy **109**(6), 677–682 (2009). doi:10.1016/j.ultramic.2009.01.001
29. Tadmor, E.B., Miller, R.E.: Modeling Materials : Continuum, Atomistic, and Multiscale Techniques, p. 759. Cambridge University Press, Cambridge (2011)
30. Goodman, P., Moodie, A.F.: Numerical evaluations of N-beam wave functions in electron scattering by the multi-slice method. Acta Crystallogr. Sect. A **30**(2), 280–290 (1974). doi:10.1107/S056773947400057X
31. Koch, C.: Determination of Core Structure Periodicity and Point Defect Density Along Dislocations. Arizona State University, Phd (2002)
32. Bahn, S.R., Jacobsen, K.W.: An object-oriented scripting interface to a legacy electronic structure code. Comput. Sci. Eng. **4**(3), 56–66 (2002). doi:10.1109/5992.998641
33. Rez, D., Rez, P., Grant, I.: IUCr: Dirac-Fock calculations of X-ray scattering factors and contributions to the mean inner potential for electron scattering. Acta Crystallogr. Sect. A Found. Crystallogr. **50**(4), 481–497 (1994). doi:10.1107/S0108767393013200
34. Kirkland, E.J.: Advanced Computing in Electron Microscopy. Springer, Boston (2010). doi:10.1007/978-1-4419-6533-2
35. Van den Broek, W., Van Aert, S., Van Dyck, D.: Fully automated measurement of the modulation transfer function of charge-coupled devices above the nyquist frequency. Microsc. Microanal. **18**(02), 336–342 (2012). doi:10.1017/S1431927611012633
36. Uhlemann, S., Müller, H., Hartel, P., Zach, J., Haider, M.: Thermal magnetic field noise limits resolution in transmission electron microscopy. Phys. Rev. Lett. **111**(4), 046101 (2013). doi:10.1103/PhysRevLett.111.046101
37. Haider, M., Hartel, P., Müller, H., Uhlemann, S., Zach, J.: Information transfer in a TEM corrected for spherical and chromatic aberration. Microsc. Microanal. **16**(04), 393–408 (2010). doi:10.1017/S1431927610013498
38. Hÿtch, M.J., Stobbs, W.M.: Quantitative comparison of high resolution TEM images with image simulations. Ultramicroscopy **53**(3), 191–203 (1994). doi:10.1016/0304-3991(94)90034-5
39. Thust, A.: High-resolution transmission electron microscopy on an absolute contrast scale. Phys. Rev. Lett. **102**(22), 220801 (2009). doi:10.1103/PhysRevLett.102.220801
40. Lee, Z., Meyer, J.C., Rose, H., Kaiser, U.: Optimum HRTEM image contrast at 20 kV and 80 kV-Exemplified by graphene. Ultramicroscopy **112**, 39–46 (2011). doi:10.1016/j.ultramic.2011.10.009
41. De Graef, M.: Introduction to Conventional Transmission Electron Microscopy, p. 718. Cambridge University Press, Cambridge (2003)
42. Lee, Z., Rose, H., Lehtinen, O., Biskupek, J., Kaiser, U.: Electron dose dependence of signal-to-noise ratio, atom contrast and resolution in transmission electron microscope images. Ultramicroscopy **145**, 3–12 (2014). doi:10.1016/j.ultramic.2014.01.010
43. Bierwolf, R., Hohenstein, M., Phillipp, F., Brandt, O., Crook, G.E., Ploog, K.: Direct measurement of local lattice distortions in strained layer structures by HREM. Ultramicroscopy **49**(1), 273–285 (1993). doi:10.1016/0304-3991(93)90234-O
44. Galindo, P.L., Kret, S., Sanchez, A.M., Laval, J.-Y., Yáñez, A., Pizarro, J., Guerrero, E., Ben, T., Molina, S.I.: The Peak Pairs algorithm for strain mapping from HRTEM images. Ultramicroscopy **107**(12), 1186–93 (2007). doi:10.1016/j.ultramic.2007.01.019
45. Zuo, J.-M., Shah, A.B., Kim, H., Meng, Y., Gao, W., Rouviére, J.-L.: Lattice and strain analysis of atomic resolution Z-contrast images based on template matching. Ultramicroscopy **136**, 50–60 (2014). doi:10.1016/j.ultramic.2013.07.018
46. Hÿtch, M.J., Snoeck, E., Kilaas, R.: Quantitative measurement of displacement and strain fields from HREM micrographs. Ultramicroscopy **74**(3), 131–146 (1998). doi:10.1016/S0304-3991(98)00035-7
47. Huang, P.Y., Kurasch, S., Alden, J.S., Shekhawat, A., Alemi, A.A., McEuen, P.L., Sethna, J.P., Kaiser, U., Muller, D.A.: Imaging atomic rearrangements in two-dimensional silica glass: watching silica's dance. Science **342**(6155), 224–247 (2013)
48. Larsen, P.M., Schmidt, S., Schiøtz, J.: Robust structural identification via polyhedral template matching. Model. Simul. Mater. Sci. Eng. **24**(5), 055007 (2016). doi:10.1088/0965-0393/24/5/055007
49. Van Dyck, D.: Does crystal tilt enhance the electron interaction? Microsc. Microanal. **4**(4), 428–434 (1998). doi:10.1017/S1431927698980412
50. Gonzalez, R.C., Woods, R.E.R.E., Eddins, S.L.: Digital Image Processing Using MATLAB , 2nd edn. Prentice Hall, New Jersey (2009)

A.2 Paper 2

A deep learning approach to identify local structures in atomic-resolution transmission electron microscopy images

Jacob Madsen, Pei Liu, Jens Kling, Jakob Birkedal Wagner, Thomas Willum Hansen, Ole Winther and Jakob Schiøtz

In preparation

A deep learning approach to identify local structures in atomic-resolution transmission electron microscopy images

Jacob Madsen,[†] Pei Liu,[‡] Jens Kling,[‡] Jakob Birkedal Wagner,[‡] Thomas Willum Hansen,[‡] Ole Winther,[¶] and Jakob Schiøtz[*,†]

[†]*Center for Atomic-scale Materials Design (CAMD), Department of Physics, Technical University of Denmark, 2800 Kgs. Lyngby, Denmark*
[‡]*Center for Electron Nanoscopy (CEN), Technical University of Denmark, 2800 Kgs. Lyngby, Denmark*
[¶]*Department of Applied Mathematics and Computer Science, Technical University of Denmark, 2800 Kgs. Lyngby, Denmark*

E-mail: schiotz@fysik.dtu.dk

Abstract

Recording atomic-resolution transmission electron microscopy (TEM) images is becoming increasingly routine. A new bottleneck is then analyzing this information, which often involves time-consuming manual structural identification. We have developed a deep learning-based algorithm for recognition of the local structure in TEM images, which is stable to microscope parameters and noise. The neural network is trained entirely from simulation, but is capable of making correct predictions on experimental images. We apply the method to single sheets of defected graphene, and to metallic nanoparticles on an oxide support.

Introduction

With the development in transmission electron microscopes that has occurred over the last decade, it has become increasingly common to record and store large amounts of TEM data, often in the form of TEM videos. This development has been accelerated by the advent of faster and more sensitive detectors such as the Direct Detection Camera;[1] but also by the development of the Environmental TEM, where it becomes possible to study how e.g. nanoparticles respond to reaction gasses in real time.[2]

As large amounts of TEM data becomes available, it becomes increasingly important to have efficient and automated analysis tools. In many applications, accurate identification and classification of local structure is a crucial first step in deriving

useful information from atomic-resolution images and video. Examples include characterizing the distribution of dopants[3] and defects,[4] *in situ* imaging of phase transformations,[5] structural reordering during materials growth[6,7] and dynamic surface phenomena.[8]

Analysis methods such as the Geometric Phase Analysis (GPA)[9] are based on the local symmetry and periodicity, and has been very successful at extracting structural information in many regular structures, including identifying defects, strain and phase boundaries.[10] However, GPA typically has difficulties analyzing e.g. surfaces, where the periodicity changes rapidly.[11]

Real space approaches typically either rely on direct identification of atomic positions by fitting local parts of the image to e.g. Gaussian intensity profiles,[12,13] or on direct comparison with a template.[14] However, these methods are in general not able to compete with a trained human expert. The difficulties arise in part due to the phase contrast nature of TEM, which makes the image extremely sensitive to small changes in the defocus, necessitating human intervention in the image analysis. When analyzing time sequences (video), it may even be necessary to adjust the image analysis tools to each frame, as small rotations, vibrations and thermal drift can modify the appearance from one frame to another. These difficulties are compounded by the low signal-to-noise ratio resulting from using the smallest possible electron dose to minimize beam damage to the sample.

Recently, convolutional neural networks and related deep-learning methods have demonstrated excellent performance in visual recognition tasks, including particle detection[15] and automatic segmentation of brain images from cryo-electron microscopy images.[16] Kirschner and Hillebrand have published a method for predicting defocus and sample thickness,[17] and Meyer and Heindl have used neural networks to reconstruct the exit wave function from off-axis electron holograms.[18]

Deep learning methods have, however, to our knowledge not yet been used to analyze the atomic structure in TEM images. In this article, we describe a CNN based method for classifying atomic structures in TEM, and demonstrate that it can be applied to single layers of graphene, as well as to supported metallic nanoparticles. Under good circumstances, the method can be generalized to identify chemical species and to identify the height of atomic columns.

Methods

The task of identifying atoms in atomically resolved TEM images is a special case of a general problem in image analysis. The task is to identify instances of a set of structures, and assigning class labels $\{c_n\}$ and Cartesian coordinates $\{(x_n, y_n)\}$ to each of them. In the simplest case, there is only one class ("atom"), but the analysis can be extended to identify specific structures of atoms; atom columns of various sizes; vacancies; etc. The neural network will be looking for a predefined set of N_c labels, $C = \{c_0, c_1, \ldots, c_{N_c}\}$ and will initially assign a probability for each possible label. The choice of how to categorize the structures is problem specific, and typically depends on how the researcher derives meaning from the image.

An example is shown in Fig. 1, where one or more images of a structure is mapped onto a set of probability maps, from which

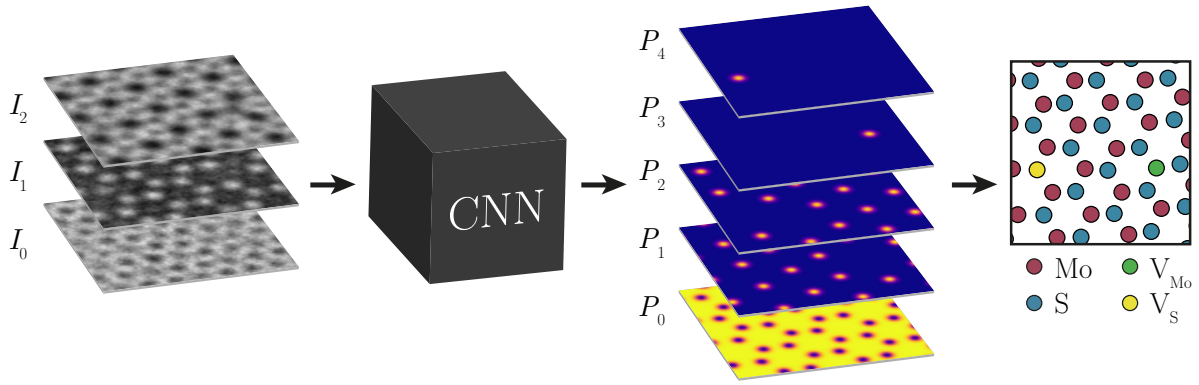


Figure 1: The classification method illustrated on images of single sheets of molybdenum disulphide, MoS_2. The convolutional neural net is fed one or more TEM images of the same sample with varying microscope parameters. The task of the CNN is to classify each pixel as belonging to one of five categories: {background, Mo atom, S atom, Mo vacancy, S vacancy}; where a vacancy is defined as a missing atom where there would be an atom in a perfect lattice. The output of the CNN is thus five probability maps, which are converted into an interpretation of the structure

the interpreted structure can be depicted. The input will typically be a single grey-scale image of size $N_x \times N_y$, but it is possible to use multiple images of the same spatial region, for example a focal series where the microscope focus is varied systematically. Thus in general the neural network maps image data $I_{x,y,k}$ of shape $N_x \times N_y \times N_f$ (where N_f is often 1) to probability maps $P_{x,y,k}$ of shape $N_x \times N_y \times N_c$, where N_c is the number of classes *including a background class*. Including the background class makes it easy to enforce normalization of the probabilities,

$$\forall x, y: \qquad \sum_k P_{x,y,k} = 1. \qquad (1)$$

With such a classification scheme, it is important that structures do not overlap, and overlapping structures should be handled by defining new classes. An example is columns of atoms, which can be handled by making classes for a single atom, a column of two atoms, etc.

Preprocessing

Contrast and illumination may vary significantly across experimentally obtained TEM images, in particular if images contain local structures that are not relevant for the problem being analyzed. This is handled by a combination of subtractive and divisive normalization. First, a local average of the intensity is subtracted from the image

$$G_{ijk} = I_{ijk} - \frac{1}{N_f} \sum_{pqk'} w_{pq} I_{i+p,j+q,k'} \qquad (2)$$

where w_{pq} is a Gaussian weighting window normalized so $\sum_{pq} w_{pq} = 1$. The decay length of the Gaussian weighting window must be chosen to be significantly longer than the length scales of the features the net should detect, to avoid washing them out.

Finally, the contrast is normalized with a divisive normalization using the same Gaussian weighting window

$$H_{ijk} = \frac{G_{ijk}}{\frac{1}{N_f}\sqrt{\sum_{pqk'} w_{pq} G^2_{i+p,j+q,k'}}} \qquad (3)$$

Neural net architecture

The neural network needs to be able to combine information on multiple length scales. Locally, the atoms are identified as local peaks or valleys, but estimating what an atom should look like requires contextual information since it depends on e.g. the defocus of the microscope. In some images the atoms may be bright spots, in other they are dark spots, and the contrast may even invert within different regions of the same image.

The network architecture is based on fully convolutional networks (FCN) for pixel-wise segmentation.[19] Following the FusionNet structure proposed by Quan *et al.*[16] we use additive skips and residual blocks to prevent vanishing gradients and to allow for training of deeper neural nets. Multi-level up-sampling and skip connections combine global abstract information from deep coarse paths with local spatially resolved information from shallow paths.

The network has a single pipeline with additive skip connections to preserve spatial information at each resolution. The lowest resolution is one eighths of the full resolution, this allows for spatial filters to be applied that compare features across the entire image. The shape of the network is chosen to be symmetric, so for every layer present in the part where resolution is reduced, there is a corresponding layer in the part where resolution is increased again. The chosen architecture is shown in Fig. 2

At each resolution on the down-sampling and up-sampling paths, the network consists of five convolutional layers, with a skip connection bypassing the middle three layers using elementwise addition (shown as a *residual block* in Fig. 2). Every convolutional layer except the last employ a 3×3 convolutional kernel, followed by an element-wise rectified linear activation, $h(x) = \max\{0, x\}$, which are then batch normalized following eq. (2–3).[20]

Feature compression is done in the down-sampling path using a max pooling layer down-sampling by a factor two in both spatial direction, while doubling the number of feature maps. Conversely, in the up-sampling path the features are up-sampled using a transpose convolutional block[19] doubling the spatial resolution while halving the number of feature maps, followed by an element-wise addition from same level of the encoding path, forming a long skip connection.

The final scoring consists of a convolutional layer with a 1×1 kernel followed by a softmax non-linearity

$$\sigma(P_k) = \frac{\exp(P_k)}{\sum_{k=1}^{m_f} \exp(P_k)}\,. \qquad (4)$$

The transpose convolutional layers are initialized as bilinear interpolation and all other layers use random weight initialization.

The network is implemented with TensorFlow using the Python API,[21] chosen due to the wide range of functions already made available, as well as the community support. All models are trained and tested with TensorFlow on a single NVIDIA GTX 1080 Ti. Our models and code are publicly available.[22]

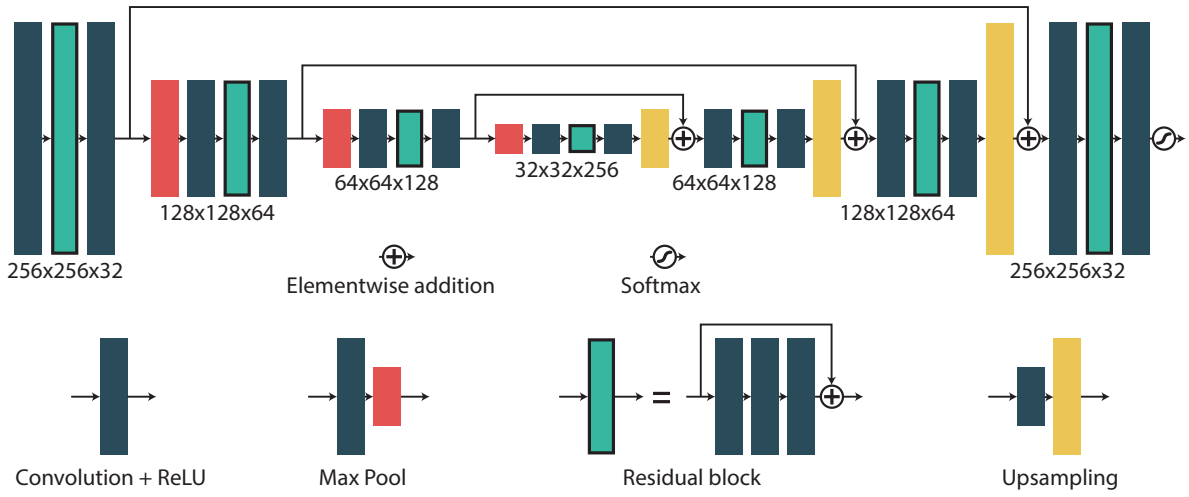


Figure 2: The architecture of the neural network. Information flows from left to right. The features are down-sampled in an encoding path and up-sampled through a decoding path, in addition several skip connections ensure that it is possible to retain fine spatial information.

Generation of training data

A particular challenge is to generate the training data for the neural net, since on one hand these data should include the kind of structures the net should be able to recognize, but on the other hand should not bias the network towards a specific interpretation of the images. This makes it particularly difficult to use real experimental data as training data, since the network would be trained to reproduce any subconscious bias of the scientists generating the interpretations to which the net is trained.

Instead, we train the network to a large set of simulated data. It is important to be aware that this does not preclude biasing the training set, since such a bias will always be present in the selection criteria generating the structures that form the basis for the image simulations, but at least the true positions of all atoms are known for the simulated images.

We try to minimize the bias of the models by generating a training set with a rather large random component while still maintaining realistic atomic positions, but without resorting to e.g. thermodynamical modelling of the systems; this will be discussed further in the sections describing applications of the method.

The training set consists of a collection of computer generated systems (e.g. nanoparticles, if the neural net is to be applied to such), generated using the Atomic Simulation Environment (ASE).[23] Simulated images are generated using the Multislice Algorithm,[24] which has been shown to provide contrast matching with experiment.[25] Simulation is done using the publicly available QSTEM code,[26] through a Python interface to ASE developed by the authors.[11] The exit wave functions for each system in the training set is precomputed, but during training simple symmetry operations (translation, rotation by 90° and mirroring) can easily be applied in each training step.

Table 1: Randomized parameters for generating training examples of graphene for a 80 kV microscope.

parameters	lower bound	upper bound	distribution
defocus (Δf)	-200 Å	200 Å	uniform
3rd order spherical (C_s)	-20 μm	20 μm	uniform
5th order spherical (C_5)	0	5 mm	uniform
1st order astigmatism magnitude	0	100 Å	uniform
1st order astigmatism angle	0	2π	uniform
deflection	0	25 Å	uniform
focal spread	20 Å	40 Å	uniform
dose	10^1 $e^-/$Å^2	10^4 $e^-/$Å^2	exponential
c_1 (MTF)	0	0.1	uniform
c_2 (MTF)	0.4	0.6	uniform
c_3 (MTF)	2	3	uniform

For each training iteration, a Contrast Transfer Function (CTF) is generated with randomly chosen parameters for the electron microscope taken from a distribution; Table 1 shows an example of parameters used for graphene. The CTF is then applied to the precomputed exit wave function. The effect of energy spread (i.e. temporal coherence) is included in the quasi-coherent approximation,[27] and temperature effects are included by blurring the atomic potentials. The images are resampled to a random sampling rate, a technique sometimes referred to as *scale-jittering*. It is essential to include a reasonable model of noise in the images, this is done by modelling the finite electron dose with a Poisson distribution, and including the modulation transfer function (MTF) of the detector in the image simulation. The latter is essential as it has a strong influence on the spectral properties of the noise, and prevents that the network incorrectly is trained to detect atoms by the absence of pure white noise. The MTF is modelled a sum of a Gaussian and an exponential:[28]

$$M(q) = c_1 \exp\left(-c_2 q\right) + \left(1 - c_1\right) \exp\left(-c_3^2 q^2\right) \tag{5}$$

The ground truth for the training data is generated as a superposition of Gaussians with an amplitude of one, centered at the positions of the atoms (or the mean of the positions for atomic columns). The background class is then assigning the remaining probability, such that the sum of probabilities is one; this is possible since the overlap between any pair of Gaussians is negligible. The width, σ, of the Gaussians is an important parameter, since it strongly influences the penalty of wrongly assigning a region of the inferred confidence map to the background. We found that a too small value of σ would lead to a network with a strong tendency to assign any region that is difficult to analyze (e.g. due to noise) to the background class. In a similar way, we found that a common local minimum at training would be to assign anything to the background class, this is also exacerbated by a low value of σ. We found that a width

of $\sigma = 1$Å, corresponding to 8–10 pixels at typical resolutions, worked well for the cases we have considered.

Training

The CNN is trained using a mean squared difference loss function, regularized with a penalty on the size of the l_2 norm of the weights

$$L = \sum_{ijk} \|\tilde{P}_{ijk} - P_{ijk}\|^2 + \frac{1}{2}\lambda \sum_i W_i^2 \quad (6)$$

where $\tilde{P}$ is the output and P is the ground truth. The main way the network is regularized is through the large variability of the training image data, since a new training image is simulated for every training iteration. Nevertheless, we found that performance on actual experimental data is improved by adding moderate l_2 regularization (also known as weight decay), since this causes any weight not being used by the network to produce meaningful output to become negligible rather than to persist for no reason. Such weights may deteriorate performance on actual experimental data although they do not negatively impact the performance on the training data.

Post-processing and interpretation

While the interpretation of the confidence maps is simplest if there is only a single class, we here illustrate how it can be done even in the case of multiple classes.

The first step is finding the regions where a signal is found. This is done by finding all minima in the confidence map for the background class. Only minima that dip below $\varepsilon = 0.995$ on a scale from 0 – 1 are included, this prevents spending time on analyzing regions that are obviously background. The local minima are then used as seeds for basins created using the watershed principle for image segmentation using Meyer's algorithm.[29] We avoid including long tails in segments by setting a hard upper limit for each segment at ε.

Each segment is then assigned a probability for belonging to each non-background class as

$$p_n(c_k) \propto \sum_{i,j \in \mathcal{S}_n} P_{ijk} \quad k > 0 \quad , \quad (7)$$

where the sum is over all pixels belonging to the n'th image segment. The coordinate of the atomic structure is calculated as the center-of-mass of the image segment. Finally, segments are discarded if

$$\sum_{k>0} p_n(c_k) < t p_n(c_0) \quad (8)$$

The value chosen for t is normally uncritical, but values near 0.5 is recommended. It should be noted that in most cases there is only a single class (c_1), "an atom". The process is illustrated in Fig. 3.

Application to graphene

High resolution TEM has been used extensively to study graphene, and several automatic algorithms for extracting quantitative information have been proposed. Of particular interest is the ability to identify defects, both localized (vacancies, dislocations etc) and extended (grain boundaries).

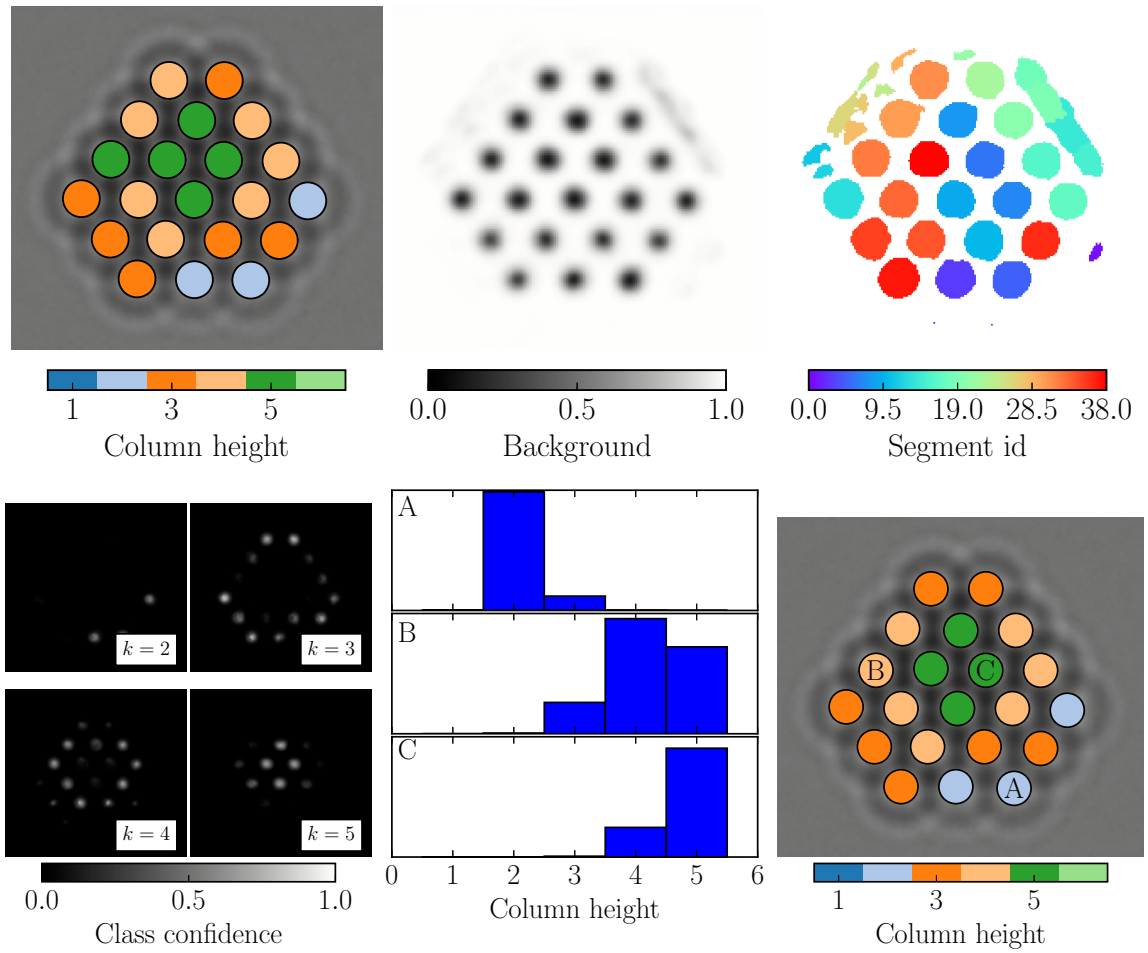


Figure 3: **(a)** The input to the network is three images of a small nanoparticle, recorded (or simulated) with different defocus; here is shown a single simulated image with the ground truth thickness of the individual columns marked with colors. **(b)** The background confidence map; we se that the network correctly identifies that there is something at each atomic column, but also thinks there may be something outside the nanoparticle. **(c)** A segmentation of the background map into several distinct objects. **(d)** Confidence maps for the classes corresponding to columns containing two to five atoms. **(e)** Each object is assigned a probability of belonging to each of the five classes. **(f)** The final classification of the atomic columns. The labels A–C mark the three columns examined in panel (e). In this case, most atomic columns are correctly assigned to their classes. Column B which is has five atoms is incorrectly identified as having only four atoms; however the network is clearly in doubt as seen in the probability distribution. One other column is misassigned, in both cases the network has probably learned that columns at edges and corners are likely to contain fewer atoms, which is not the case for these two columns.

Training

It is an easy task for a CNN to recognize the regular hexagonal lattice of graphene. However, we want the network to be able to correctly localize the atomic positions also in situations where they are not at or near their ideal positions. Thus, the atomic models used to generate the training images cannot simply be ideal sheets of graphene, nor can they be sheets of graphene with added defects.

The opposite extreme, that of generating purely random atomic positions would result in inefficient networks, as the vast majority of the training data would be very different from the experimentally interesting situations. Instead, we generate atomic positions that lie somewhere between these two extremes.

The algorithm is based on the observation that a Voronoi tessellation of a 2D set of points mainly consists of hexagons, and is illustrated in Fig. 4. First, an area is filled with randomly distributed points under the constraint of a minimal distance between the points, i.e. a Poisson disc distribution. These points form the generating centers of a Voronoi tessellation; the vertices of the tessellation will become the carbon atoms. The tessellation is optimized by a few steps of Lloyd's algorithm:[30] the centers of the Voronoi tessellation are moved to the center of mass of their respective Voronoi cell. This makes the Voronoi polyhedra more regular, and in particular it moves closely placed vertices apart, preventing atoms from being placed unrealistically close. Finally, from zero to four holes are cut randomly in the structure.

The resulting structures form a structure which is very suitable for our purpose. The distribution of bond lengths is quite narrow, and the mean can be controlled by choosing the initial number of points. The structure contains a large number of polygons with five to eight sides, similar to what is observed in graphene grain boundaries. A typical training structure is shown in Fig. 5.

We generated 500 random structures with a size of 43.2×43.2 Å, or 360×360 pixels at a sampling rate of 0.12 Å/pixel. All the simulations were done at an acceleration voltage of 80 kV. While the microscope parameters are uniquely generated at each training step, the same structure is utilized multiple times. This have little consequence since most of the variability is in microscope parameters.

Analyzing experimental images

An example of how the network performs on experimental data is given in Fig. 6. When given an experimental TEM image of the edge of a graphene sheet, the neural network has no problems identifying the atoms inside the graphene sheet. At the edge of the sheet, there are positions where the network assigns a small but nonzero probability for the presence of atoms, but using a reasonable cutoff of $t = 0.5$ gives a result in agreement with a manual analysis of the image, and without any high-energy atomic configurations at the edges.

We apply the trained neural net on a number of graphene images that have previously been published by some of us.[31,32] The experimental graphene images were measured using a FEI Titan 80-300 Environmental TEM equipped with a monochromator at the electron gun and spherical aberration (C_s) corrector at the objective lens. The acceleration voltage of the microscope were 80 kV which is

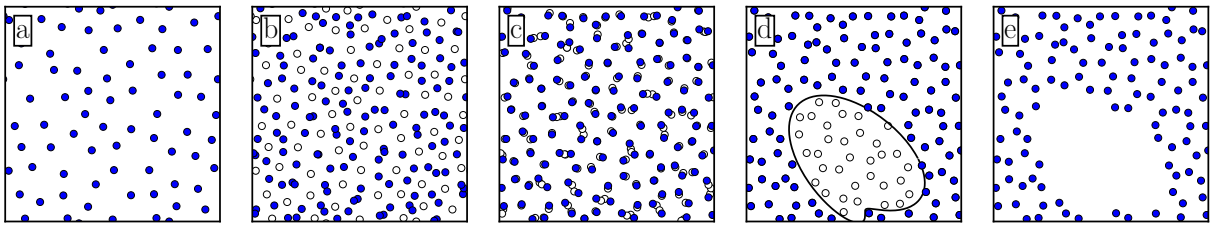


Figure 4: Procedure for generating training structures for graphene. **(a)** A square in 2D space is filled with randomly distributed seed points under the constraint of a minimum separation in terms of euclidean distance. **(b)** Next, the Voronoi tessellation is generated from the seed points, the vertices of the diagram will become atomic positions while the original positions are discarded. **(c)** To avoid overlapping atoms, the positions of the Voronoi vertices are relaxed using Lloyd's algorithm. **(d)** Lastly, zero to four holes of varying size and shape are introduced in the structure.

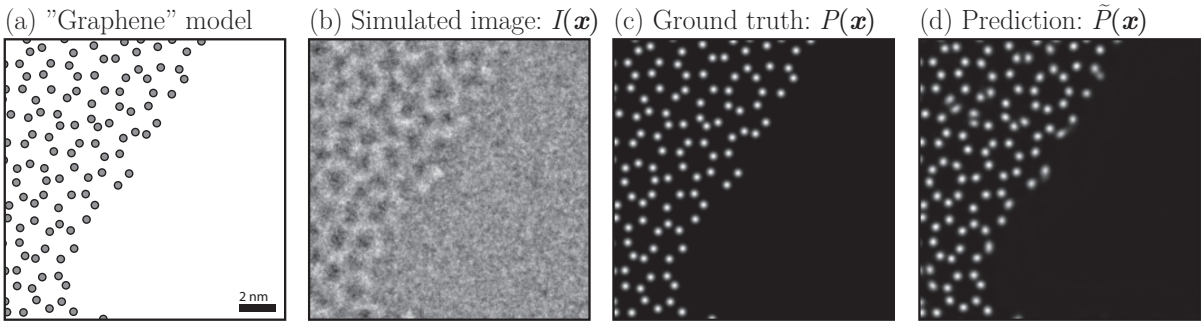


Figure 5: The CNN is trained on simulated images of graphene-like structures, generated by the algorithm in Fig. 4. **(a)** A quasi-random graphene-like structure. **(b)** Simulated image based on the atomic positions. **(c)** The corresponding ground truth calculated from the atomic positions. The network is trained on a series of matching images and ground truth maps. **(d)** The output prediction of the trained neural network given the simulated image in (b).

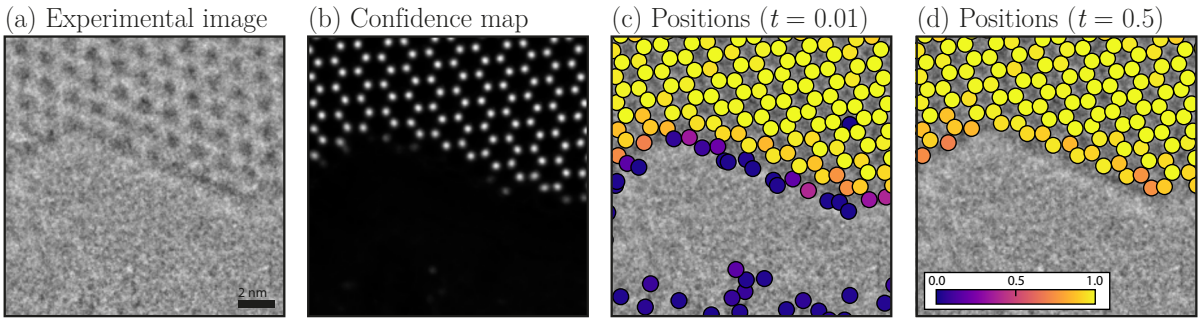


Figure 6: A neural network trained exclusively on simulated data is capable of generalizing to experimental images. **(a)** Single suspended graphene sheet with a hole formed under the influence of the electron beam. **(b)** The regressed probability distribution predicted by the neural network for the image in (a). **(c)** The local peak positions of the probability map is overlayed on the image. The peaks are color-coded according to their maximum value. Peaks with a maximum value less than 0.01 are excluded. (**d**) A higher tolerance for exclusion is used to remove peaks with a maximum value less than 0.5.

below the knock-on threshold for carbon atoms in pristine graphene.[33] The electron beam energy spread was below 0.3 eV, while the C_s-corrector was aligned to minimize the spherical aberration. The images were recorded using a Gatan US1000 CCD camera with an exposure time of 1 s.

Fig. 7 shows a TEM images of pristine graphene, and of graphene with a hole. The negative C_s imaging results in images where the carbon atoms are bright spots, with the centers of the hexagons appearing dark. The output of the neural network is shown in the central column. The neural network detects all atomic positions in the pristine sheet, this is accomplished without having regular hexagonal lattices in the training set. Additionally, the neural network automatically recognizes that the atoms appear bright, which is only the case for half of the training images. Finally, we show the strain calculated from the atomic positions, using a structural template with the two nearest neighbour shells (i.e. the 9 nearest neighbours), as described previously.[11]

Application to metallic nanoparticles

Metallic nanoparticles on oxide support is a very active research topic, mainly due to the applications within heterogeneous catalysis. Often, the detailed atomic structure is important for the catalytic process, as the active size depending on the process may be e.g. step sites,[34] corner atoms[35] or strained facets.[36,37]

For example, although gold is normally chemically inert, nanoparticles of gold have been shown to catalyse the oxidation of CO.[38] It is also a system where significant atomic rearrangement is observed in the presence of gasses, both involving overall shape changes of the nanoparticles[39] and changes in the local surface structure We here use supported gold nanoparticles to illustrate the application of neural nets to the analysis of supported nanoparticles.

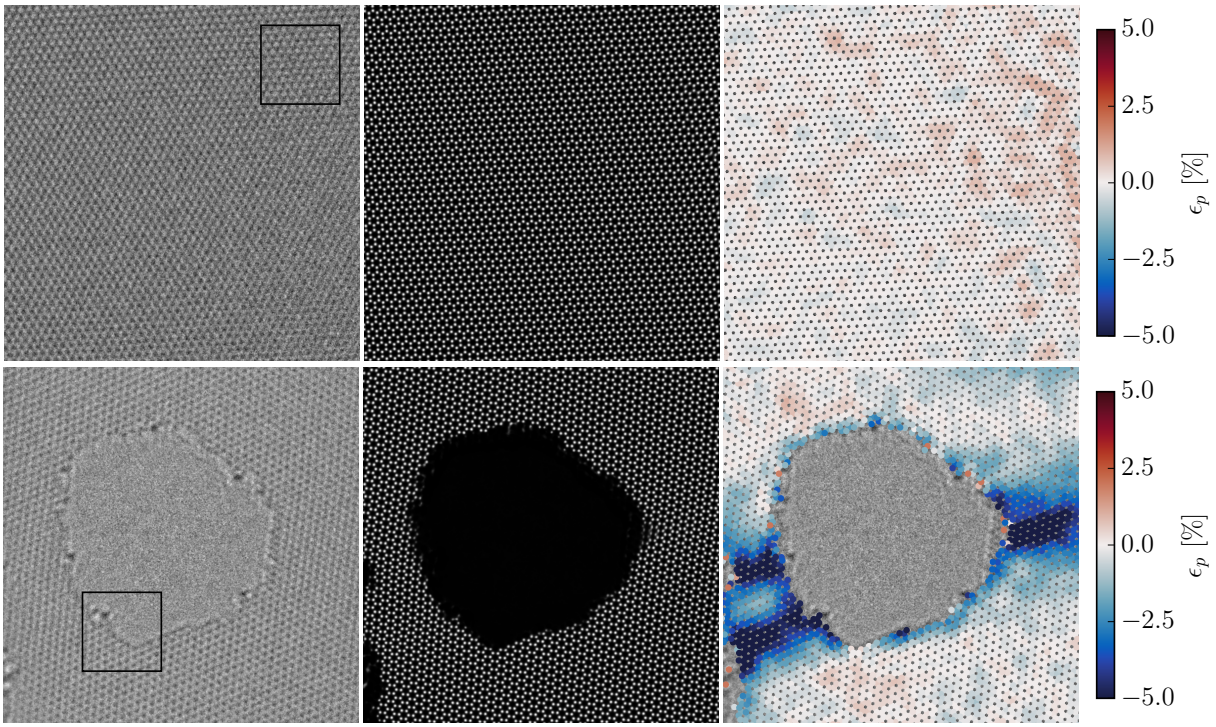


Figure 7: Experimental images of graphene, and their interpretation by the neural net. The first row shows a pristine sheet of graphene, the second a sheet with a hole. The left column shows the original TEM images. The center column shows the output of the neural net. The rightmost column shows the planar strain calculated from the atomic positions, as identified by the neural net.

Training

We have trained the network on simulated gold nanoparticles. As the network should be able to recognize both atomically flat and rough surfaces, the training set includes both kinds of nanoparticles. Initially, nanoparticles are cut from a regular crystal, keeping a random number of layers in directions with low Miller indices (the $\langle 100 \rangle$, $\langle 110 \rangle$ and $\langle 111 \rangle$ directions). To roughen the particles, a random number of additional atoms are added to the particle. The atoms are added at allowed crystal positions at the surface of the nanoparticles, in such a way that highly coordinated surface sites are most likely to be picked. If the coordination number (i.e. the number of occupied neighbor sites) of site i is n_i, then the probability of placing the next atom at site i is chosen as

$$p(n_i) = \frac{\exp(n_i/T)}{\sum_j \exp(n_j/T)} \qquad (9)$$

where the sum is over all sites j where $n_j \geq 1$ and T is a temperature-like parameter that can be chosen differently for each nanoparticle to generate particles with different roughness.

Each particle is then rotated into the $\langle 110 \rangle$ or $\langle 111 \rangle$ zone axis, and is rotated a random amount around the axis. It is finally tilted 0–5° away from the zone axis.

As was the case for the graphene simulations, 500 of nanoparticles were generated, but during the training new microscope parameters were picked for each iteration, and the nanoparticles were randomly translated, mirrored and rotated by a multiple of 90° (operations that can cheaply be performed on the precomputed wavefunctions). Figure 8 shows a sample of generated nanoparticles, and their corresponding images.

Analyzing nanoparticle images and videos

We applied the resulting network to gold nanoparticles on a ceria substrate. Figure 9 shows a TEM image of such a particle, and the corresponding analysis by the neural net. It is seen that the network confidently identifies the atoms in the nanoparticle, but does a much less impressive job in the substrate; this is partly due to the network not being trained on ceria's crystal structure, partly because the substrate is not in a prominent zone-axis orientation.

In the microscope, a video sequence of this nanoparticle was recorded, Fig. 9 shows four snapshots of this video, clearly showing the atomic diffusion processes.

We used the neural network to analyze TEM movies showing surface diffusion on gold nanoparticle in various gasses. Figure 10 shows the same ceria-supported gold nanoparticle in high vacuum and in an oxygen atmosphere. The neural network is applied to each frame in the video sequence, and used to identify the presence (and position) of the atomic columns. Since the network was not trained on substrates, and since atomic resolution is often not obtainable simultaneously in the substrate and the nanoparticle, we only use the network to analyze the metallic nanoparticle and mask out the output of the network corresponding to the substrate.

During the video sequence, atoms at the surfaces and in particular at the corners of the nanoparticle are clearly seen to appear and disappear again, as the surface atoms diffuse on the particle. We illustrate this in the figure in two different ways. In the middle column, atomic columns are colored according to the fraction of time they are present in the image. It is clearly seen that in the presence of oxygen, many of the surface and corner atoms are only present part of the time, indicating surface diffusion. In the rightmost columns, "events" are counted. It is considered an "event" if an atomic column is present in one frame, but absent in the next, or *vice versa*; this cause the diffusing atoms to light up on the figure. Together, this analysis shows that the presence of oxygen clearly enhances the surface diffusion. This will be the topic of a separate publication.

Other applications

We envision that the main application of this technique will be to identify atoms or columns of atoms, as demonstrated in the two afore-mentioned examples. For this kind of applications, it is relatively straight-forward to train a neural net on sets of simulated images, such that the net becomes able to identify the positions of the atoms or atomic columns also in experimentally obtained images.

A far more demanding task is to identify the chemical species of single atoms in 2D materials, or to count the number of atoms in atomic columns in nanoparticles. It does

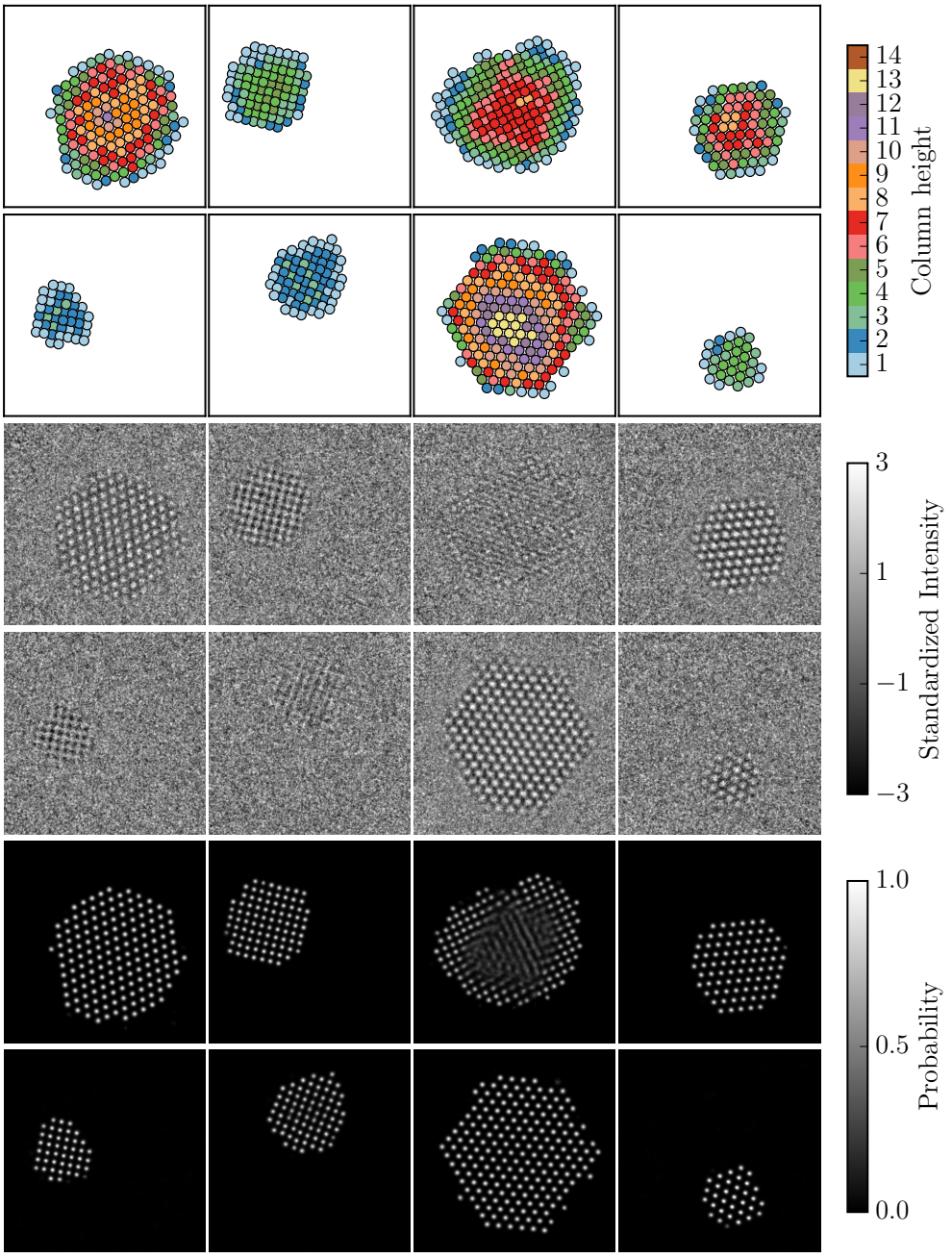


Figure 8: **Top rows:** Examples of nanoparticles generated using the algorithm we have proposed. The height of the atomic columns of the nanoparticles are indicated with a color-coding. **Middle rows:** Simulated images given the atomic models above. **Bottom rows:** Output from our neural network method given the simulated images. As can be seen, in one of the images the network is not able to identify the atomic columns in the thickest part of the nanoparticle, this is due to a combination of low signal-to-noise ratio in the image, and a significant off-axis tilt smearing out the highest atomic columns.

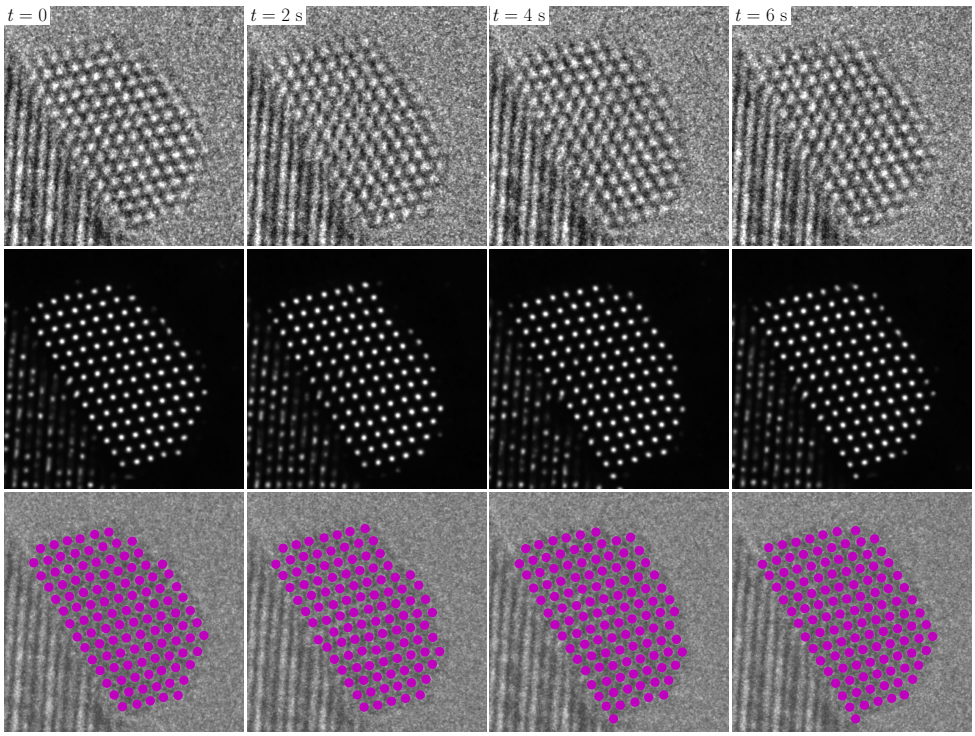


Figure 9: **top row:** Experimental high resolution TEM image of gold on ceria in an oxygen atmosphere. The pressure was 4.5 Pa and the electron dose per image was 3.1×10^2 $e^-/Å^2$ (dose rate 1.56×10^3 $e^-/Å^2/s$ at an exposure time of 0.2 s). **Middle row:** Output from the neural network given the images above. **Bottom row:** The atoms identified by the neural net are marked as purple circles overlaid on the original image. The experimental images were measured using a FEI Titan 80-300 Environmental TEM operated at 300 keV.

not appear to be possible to train a network that solves this kind of tasks based on a single image. However, if one is willing to use a series of images taken with varying focus settings, it appears to be possible to train such networks to identify multiple mutually exclusive atomic objects. The reliability does, however, not seem to be on par with the reliability of identifying a single atom or column. Figures from our attempts to build these more advanced networks have been used to illustrate the general principles (Figures 1 and 3). Since we have not yet been able to demonstrate the useful of such networks on experimental images, we will defer further discussions on the subject to future publications.

Conclusion

We have demonstrated that deep convolutional neural networks can be trained to recognize the local atomic structure in High Resolution Transmission Electron Microscopy images. The network can be trained entirely on simulated data, but is capable of giving interpretations of experimental images that match those of a trained microscopist. We have demonstrated the method both on single layers

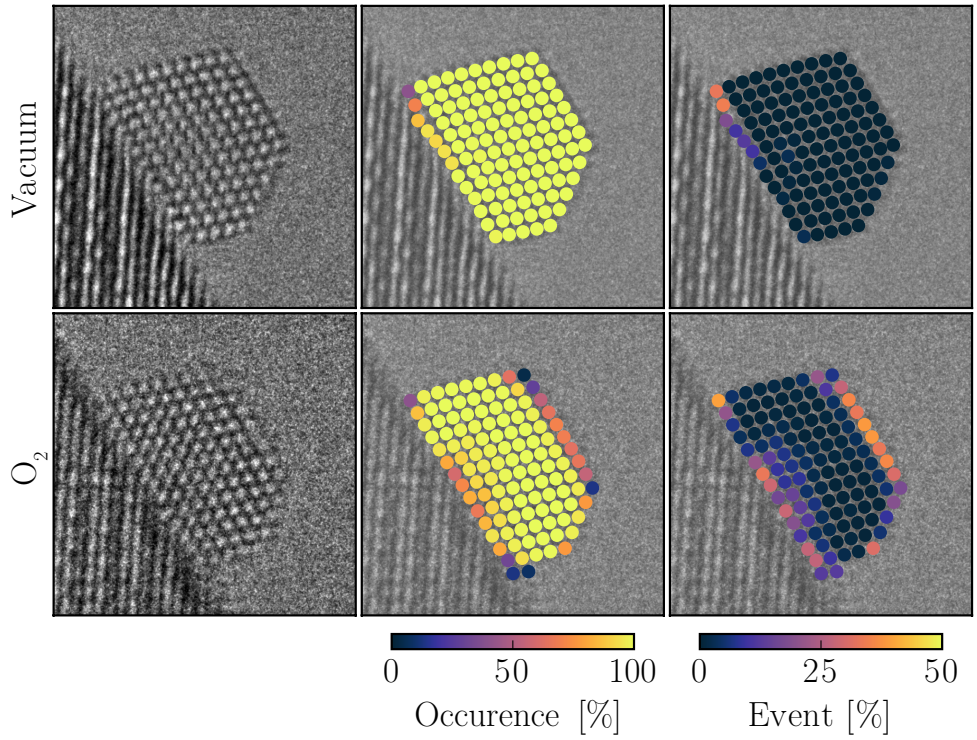


Figure 10: The surface dynamics of gold nanoparticles is influenced by the gaseous atmosphere. The occurence is the percentage of frames where the neural network identified an atomic column at a possible site. And the events are the percentage of frames where a site was previously occupied, but is unoccupied in the frame immediately after.

of defected graphene, and on nanoparticles of gold on a cerium oxide substrate.

References

(1) McMullan, G.; Faruqi, A. R.; Clare, D.; Henderson, R. *Ultramicroscopy* **2014**, *147*, 156–163.

(2) Wagner, J. B.; Cavalca, F.; Damsgaard, C. D.; Duchstein, L. D. L.; Hansen, T. W. *Micron* **2012**, *43*, 1169–1175.

(3) Meyer, J. C.; Kurasch, S.; Park, H. J.; Skakalova, V.; Künzel, D.; Gross, A.; Chuvilin, A.; Algara-Siller, G.; Roth, S.; Iwasaki, T.; Starke, U.; Smet, J. H.; Kaiser, U. *Nat. Mater.* **2011**, *10*, 209–215.

(4) Meyer, J. C.; Kisielowski, C.; Erni, R.; Rossell, M. D.; Crommie, M. F.; Zettl, A. *Nano Lett.* **2008**, *8*, 3582–3586.

(5) He, X.; Xu, T.; Xu, X.; Zeng, Y.; Xu, J.; Sun, L.; Wang, C.; Xing, H.; Wu, B.; Lu, A.; Liu, D.; Chen, X.; Chu, J. *Nature Publishing Group* **2014**, *4*, 6544.

(6) Nagao, K.; Inuzuka, T.; Nishimoto, K.; Edagawa, K. *Phys. Rev. Lett.* **2015**, *115*, 075501.

(7) Li, X.; Cheng, S.; Deng, S.; Wei, X.; Zhu, J.; Chen, Q. *Nature Publishing Group* **2017**, *7*, 40911.

(8) Schneider, S.; Surrey, A.; Pohl, D.; Schultz, L.; Rellinghaus, B. *Micron* **2014**, *63*, 52–56.

(9) Hÿtch, M. J.; Snoeck, E.; Kilaas, R. *Ultramicroscopy* **1998**, *74*, 131–146.

(10) Zhu, Y.; Ophus, C.; Ciston, J.; Wang, H. *Acta Mater.* **2013**, *61*, 5646–5663.

(11) Madsen, J.; Liu, P.; Wagner, J. B.; Hansen, T. W.; Schiøtz, J. *Adv. Struct. Chem. Imag.* **2017**, 1–12.

(12) Bierwolf, R.; Hohenstein, M.; Phillipp, F.; Brandt, O.; Crook, G. E.; Ploog, K. *Ultramicroscopy* **1993**, *49*, 273–285.

(13) Galindo, P. L.; Kret, S.; Sanchez, A. M.; Laval, J.-Y.; Yáñez, A.; Pizarro, J.; Guerrero, E.; Ben, T.; Molina, S. I. *Ultramicroscopy* **2007**, *107*, 1186–1193.

(14) Zuo, J.-M.; Shah, A. B.; Kim, H.; Meng, Y.; Gao, W.; Rouviére, J.-L. *Ultramicroscopy* **2014**, *136*, 50–60.

(15) Zhu, Y.; Ouyang, Q.; Mao, Y. *BMC Bioinformatics* **2017**, *18*, 348.

(16) Quan, T. M.; Hildebrand, D. G. C.; Jeong, W.-K. *arXiv* **2016**, 1612.05360.

(17) Kirschner, H.; Hillebrand, R. *Information Sciences* **2000**, *129*, 31–44.

(18) Meyer,; Heindl, *J. Microsc.* **2008**, *191*, 52–59.

(19) Shelhamer, E.; Long, J.; Darrell, T. *IEEE Trans Pattern Anal Mach Intell* **2017**, *39*, 640–651.

(20) Ioffe, S.; Szegedy, C. *arXiv* **2015**, 1502.03167.

(21) Abadi, M. et al. TensorFlow: A System for Large-Scale Machine Learning. Proceedings of the 12th USENIX Symposium on Operating Systems Design and Implementation (OSDI '16). 2016.

(22) https://github.com/jacobjma.

(23) Hjorth Larsen, A. et al. *J Phys Condens Matter* **2017**, *29*, 273002.

(24) Goodman, P.; Moodie, A. F. *Acta Crystallographica Section A: Crystal Physics* **1974**, *30*, 280–290.

(25) Thust, A. *Phys. Rev. Lett.* **2009**, *102*, 220801.

(26) Koch, C. Determination of core structure periodicity and point defect density along dislocations. Ph.D. thesis, Arizona State University, 2002.

(27) Kirkland, E. J. *Advanced Computing in Electron Microscopy*, 2nd ed.; Springer US: 978-1-4419-6532-5, 2010.

(28) Van den Broek, W.; Van Aert, S.; Van Dyck, D. *Microsc. Microanal.* **2012**, *18*, 336–342.

(29) Dougherty, E. R. *Mathematical morphology in image processing*; M. Dekker, 1993.

(30) Lloyd, S. P. *Ieee Transactions on Information Theory* **1982**, *28*, 129–137.

(31) Vestergaard, J. S.; Kling, J.; Dahl, A. B.; Hansen, T. W.; Wagner, J. B.; Larsen, R. *Microsc. Microanal.* **2014**, *20*, 1772–1781.

(32) Kling, J.; Vestergaard, J. S.; Dahl, A. B.; Stenger, N.; Booth, T. J.; Bøggild, P.; Larsen, R.; Wagner, J. B.; Hansen, T. W. *Carbon* **2014**, *74*, 363–366.

(33) Zobelli, A.; Gloter, A.; Ewels, C. P.; Seifert, G.; Colliex, C. *Phys. Rev. B* **2007**, *75*, 245402.

(34) Honkala, K.; Hellman, A.; Remediakis, I. N.; Logadottir, A.; Carlsson, A.; Dahl, S.; Christensen, C. H.; Norskov, J. K. *Science* **2005**, *307*, 555–558.

(35) Brodersen, S. H.; Grønbjerg, U.; Hvolbæk, B.; Schiøtz, J. *J Catal* **2011**, *284*, 34–41.

(36) Stephens, I. E. L.; Bondarenko, A. S.; Grønbjerg, U.; Rossmeisl, J.; Chorkendorff, I. *Energy Environ. Sci.* **2012**, *5*, 6744.

(37) Escudero-Escribano, M.; Malacrida, P.; Hansen, M. H.; Vej-Hansen, U. G.; Velazquez-Palenzuela, A.; Tripkovic, V.; Schiøtz, J.; Rossmeisl, J.; Stephens, I. E. L.; Chorkendorff, I. *Science* **2016**, *352*, 73–76.

(38) Haruta, M.; Kobayashi, T.; Sano, H.; Yamada, N. *Chemistry Letters* **1987**, *16*, 405–408.

(39) Uchiyama, T.; Yoshida, H.; Kuwauchi, Y.; Ichikawa, S.; Shimada, S.; Haruta, M.; Takeda, S. *Angew. Chem. Int. Ed. Engl.* **2011**, *50*, 10157–10160.

A.3 Paper 3

Surface atom diffusion on Au nanoparticles by time-resolved HRTEM

Pei Liu[1], Jacob Madsen[2], Jakob Schiøtz[2], Jakob Birkedal Wagner[1], Thomas Willum Hansen[1]

[1] Danchip/Cen, Technical University of Denmark, DK-2800 Kongens Lyngby
[2] Department of Physics, Technical University of Denmark, DK-2800 Kongens Lyngby

In preparation

Electron beam effect on the morphology of supported Au nanoparticles was examined under a range of dose rates from 2.7×10^3 to 49×10^3 $e^-/\text{Å}^2/s$. Fluctuations of the surface atom column occupation were measured quantitatively using sequences of high-resolution transmission electron microscopy images. Columns with higher coordination number start hopping under higher dose rates indicating higher internal energy of the particle under the conditions used. The internal energy, which most contributes to the kinetic energy of the diffusing atoms increases with both the dose rates and the total time of exposure. For a given dose rate, hopping column number increase as the exposure time accumulate. For columns with high coordination number, i.e. higher binding energy, hopping is only observed under high dose rates.

A.4 Paper 4

Efficient first principles simulation of electron scattering factors for transmission electron microscopy

Toma Susi[1], Jacob Madsen[2], Ursula Ludacka[1], Jens Jørgen Mortensen[2], Timothy J. Pennycook[3], Zhongbo Lee[3], Bernhard Bayer[1], Jani Kotakoski[1], Ute Kaiser[3] and Jannik C. Meyer[1]

[1] University of Vienna, Faculty of Physics, Boltzmanngasse 5, 1090 Vienna, Austria

[2] Technical University of Denmark, Department of Physics, 2800 Kgs. Lyngby, Denmark

[3] Ulm University, Electron Microscopy Group of Materials Sciences, Albert-Einstein-Allee 11, 89081 Ulm, Germany

In preparation

Electron microscopy is a powerful tool for studying the structure and properties of materials down to the atomic level. However, in many cases, the quantitative interpretation of images requires simulations based on atomistic structure models. These typically use the independent atom approximation that neglects bonding effects, which may, however, be measurable and of great physical interest. Since all electrons and the nuclear cores affect the scattering potential, simulations that go beyond this approximation have relied on computationally highly demanding all-electron calculations. Here, we describe a new method to generate *ab initio* electrostatic potentials when describing the core electrons by projector functions. Combined with an interface to quantitative image simulations written in the same programming language, this implementation enables an easy and fast means to model electron microscopy images and diffraction patterns. In addition to describing the implementation, we compare simulated electron diffraction patterns to high-quality experimental data, showing accuracy equivalent to earlier all-electron calculations at a much lower computational cost.