# StatSTEM
# User Guide
## For StatSTEM v3.0

# Contents

# 1   Getting started

The program can be downloaded from the StatSTEM website. One can choose between running StatSTEM from MATLAB or installing a compiled version (executable).

## 1.1   MATLAB

Once all StatSTEM files are copied to your computer, start MATLAB and go to the folder containing the StatSTEM files. The program can be started by running the main script *StatSTEM.m*.

# 2 A first look

In StatSTEM different options are offered for quantifying your images. In general, the quantification consists of 3 steps:

- Preparation

- Modelling

- Analysis

For each step a tab panel is present on the left hand side. On the right, loaded images and generated results are shown. Multiple images can be loaded. At the bottom left, one can load and store files. On the bottom, messages are shown to inform you when computations are finished, changes are made, or things went wrong. A progress bar indicates the status of the computation. As computations are often iterative, the progress bar cannot be used for estimating how much time computations will take.
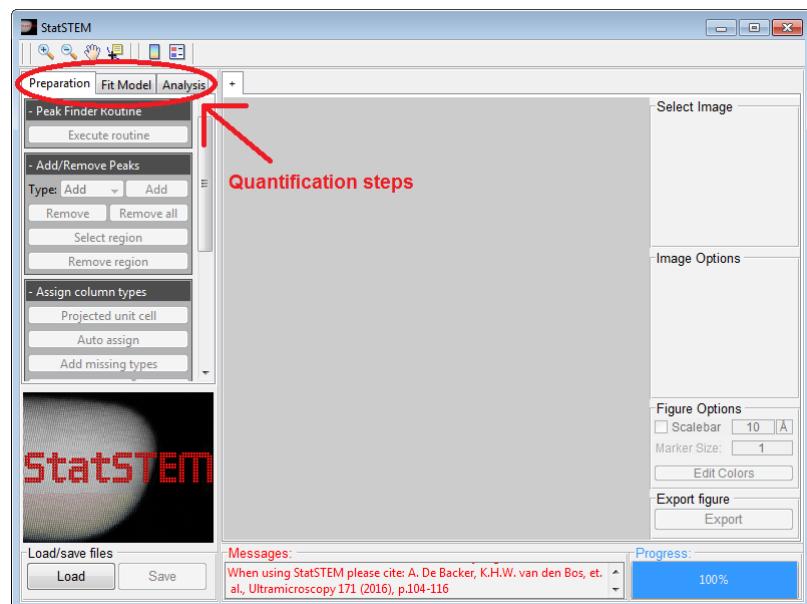


Figure 2.1: A tab panel can be selected for each quantification step

# 3    Load and save files

Once StatSTEM is started, you can load your image into the program by either clicking on the *Load* button below or the addition tab in the right panel. StatSTEM supports files having the MATLAB (*.mat*) and text (*.txt*) format. After loading the image, a dialogue will appear asking for the pixel size. Images and the corresponding parameters can be saved in the MATLAB format by clicking on the *Save* button next to the *Load* button. In the saved file, all variables are stored in as StatSTEM class files. When working with these files, make sure that the StatSTEM folder is always loaded to the path in MATLAB. In Appendix A more details are given on the information that these files hold.
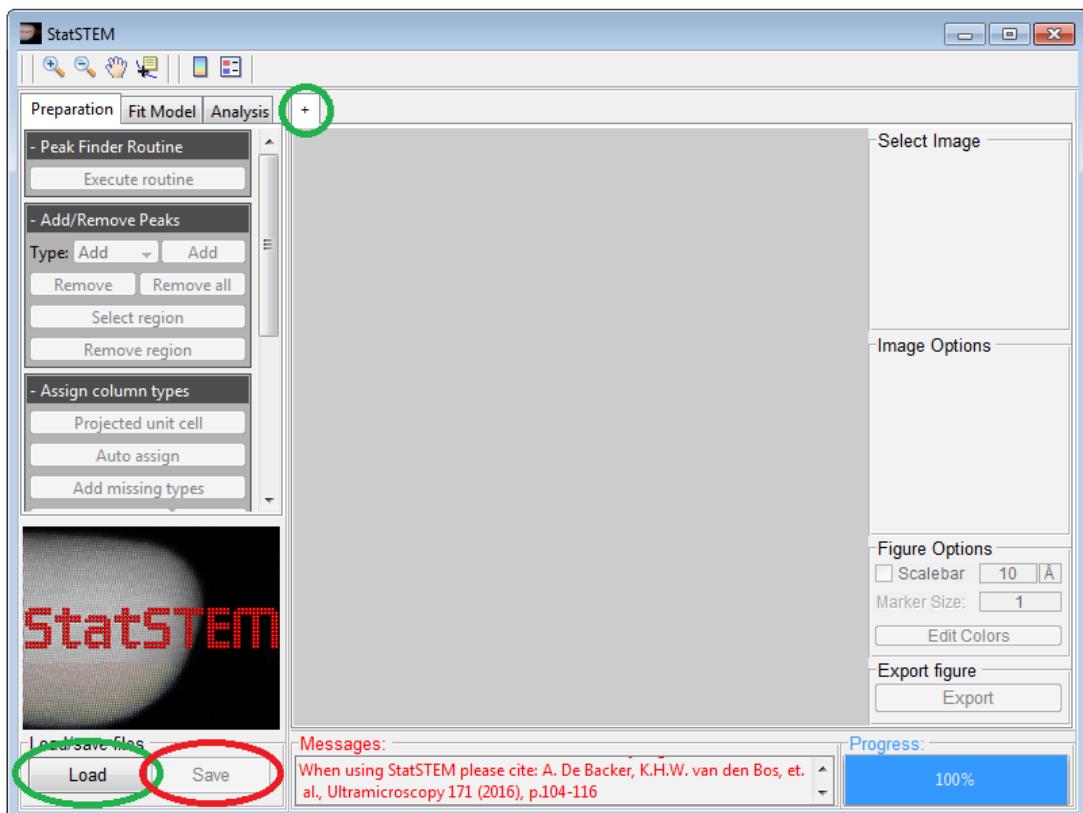


Figure 3.1: Buttons for loading and saving files indicated by respectively the green and red circle(s)

# 4 Figure options

StatSTEM offers different options to show the images and the parameters you want. For each loaded file, the *Select Image* panel allow you to select an image. Different parameters displayed in *Select Options* can be shown in the image. By hitting the *Export* button, the shown image will be opened in a new MATLAB figure which allows one to save the image in the desired format.
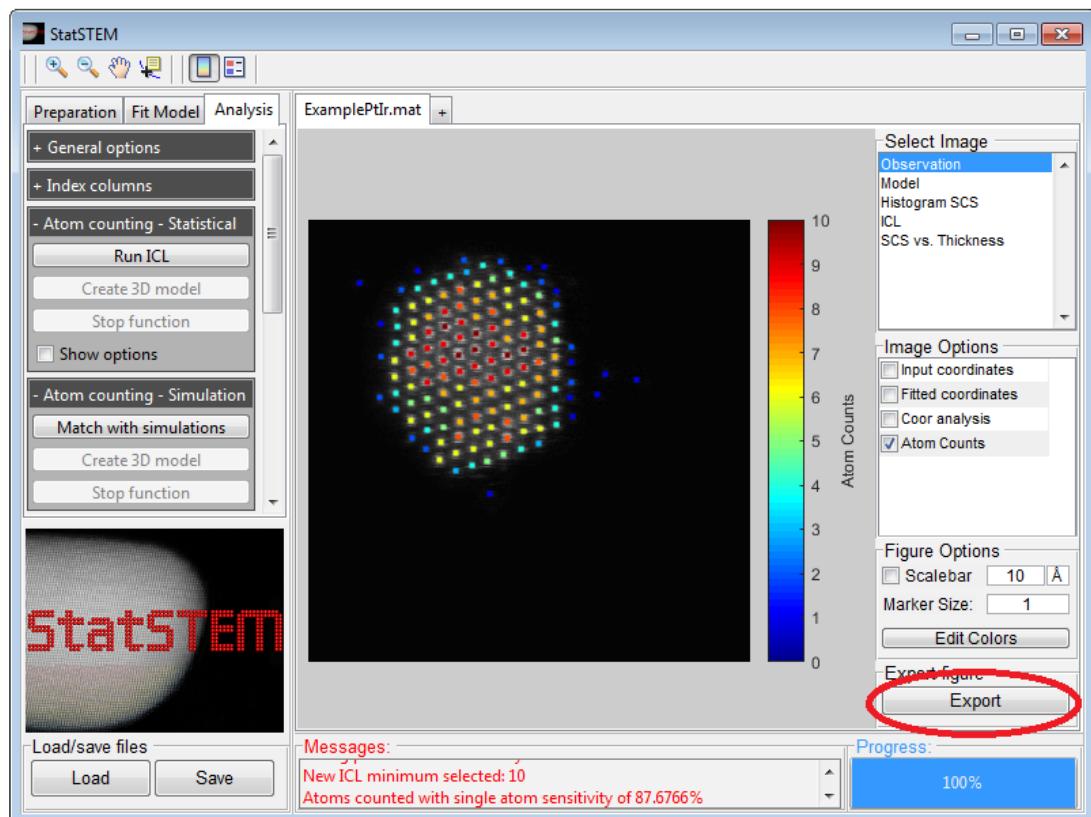


Figure 4.1: On the right hand side of StatSTEM you can select an image with the desired parameters. The export button, indicated by the red circle, can be used to open the image in a new MATLAB figure.

# 5 Preparation

Before fitting a Gaussian model to your loaded experimental images, starting coordinates for the atomic column positions have to be defined (see Fig. 5.3). For this step, different options are available on the preparation panel.

## 5.1 Peak Finder Routine

To quickly find the column positions, a peak finder routine is available which searches for local maxima in the image. When using this option, a new window will open in which the parameters of the peak finder routine can be tuned. As noise is often present in experimental images, filters can be inserted for smoothing the image. In total 3 different filters can be selected, a *gaussian*, *averaging* and a *disk* filer. Furthermore a threshold value can be defined, above which the pixel values should lie. In this window, the routine can be tested for the selected parameters. Once appropriate parameters are selected, the parameters can be used to find and export peak locations to StatSTEM by hitting the *Use values* button.

## 5.2 Add/Remove peaks

In this panel, different routines are available to define, remove or, change starting coordinates manually. By pressing *escape* the routines will be aborted.

### 5.2.1 Type

By using this option, starting coordinates can be labelled. This is particularly useful when your image consists of different column types, for example columns containing different elements. When adding column types, StatSTEM will automatically label different column types by numbers. In the fitting procedure, labelled columns may be given a special treatment. This will be explained in Section 6.1.2. For now it is important to stress out that the fitted parameters are more reliable when identifying different column types in your image.

### 5.2.2 Add

This option allows one to add manually starting coordinates in the image. Each added starting coordinate will have the selected column type label (indicated next to the button). By pressing *escape* or clicking outside the image, the routine will be aborted.

### 5.2.3 Remove

Starting coordinates, of any column type, can be manually removed by this option. By clicking in the image, the starting coordinate closest to the cursor will be removed. By pressing *escape* or clicking outside the image, the routine will be aborted.

### 5.2.4 Remove all

All starting coordinates will be removed.

### 5.2.5 Select region

Select a region in the image in which the starting coordinates should be maintained. Outside the selected region, all starting coordinates will be removed. In this routine, the user defines the corner points of the selected area one by one. A right click will connect the last defined point with the starting point. By pressing *escape*, the routine will be aborted.

### 5.2.6 Remove region

Select a region in the image in which the starting coordinates should be removed. In this routine, the user defines the corner points of the selected area one by one. A right click will connect the last defined point with the starting point. By pressing *escape*, the routine will be aborted.

## 5.3 Assign column types

Here, different options are listed to deal with image where different atomic columns types are present. To obtain the most reliable parameter when fitting, it is important to identify the different column types in your image.

### 5.3.1 Projected unit cell

In StatSTEM, automatic routine are available that can identify the different column types in an image when giving a project unit cell. Here, the relative location of each column in the projected unit cell

8

together with lattice parameters are given. You can use the buttons *New* and *Delete* to add and remove a column, respectively. All columns expect the first one can be remove by using the button *Clear*. You can also give information on the depth location of each atom in a column by using the button *z-information*. The functionality will be explained in 7.2.1. A database is added to StatSTEM for some common materials and viewing directions. Make sure that in StatSTEM you fill in the correct pixel size of the image as the lattice parameters in the projected unit cell should be close to the experimental values.
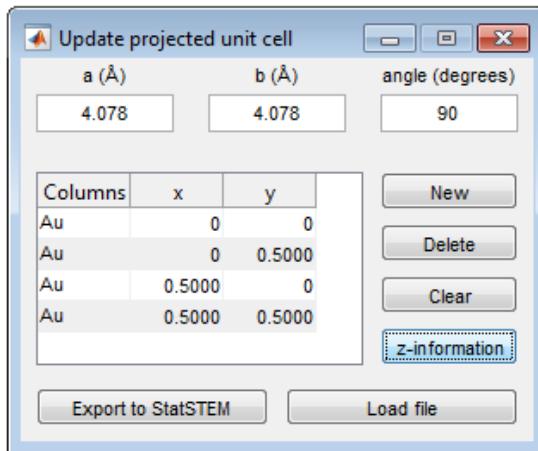


Figure 5.1: The column locations in a projected unit cell of Au viewed along the [100]-direction.

### 5.3.2 Auto assign
By using this option, StatSTEM will try to identify to different column types present in your image. In this procedure, lattice directions are first determined by using the most central coordinate with its neighbouring coordinates. Here, the projected unit cell is used. Then columns are indexed with respect to this central coordinate. In this manner, the position of each column in the project unit cell is identified and the different column types can be assigned.

### 5.3.3 Add missing types
By using this option, StatSTEM will use the projected unit cell to find and add the locations of the missing column types. In this procedure, lattice directions are first determined by using the most central coordinate with its neighbouring coordinates. Here, the projected unit cell is used. Then columns are indexed with respect to

this central coordinate. In this manner, the position of each column in the project unit cell is identified and the missing column types are identified and added.

### 5.3.4   Change type to
By using this option, a region in the image can be selected to change the type of the columns to the selected type label (indicated next to the button). In this routine, the user defines the corner points of the selected area one by one. A right click will connect the last defined point with the starting point. By pressing *escape*, the routine will be aborted.

## 5.4   Image parameters/actions
In this panel, general image parameters can be changed and image operations can be executed

### 5.4.1   Pixel size
Here, the image pixel size can be changed. By changing the pixel size, starting coordinates and fitted parameters will be rescaled.

### 5.4.2   Cut part from image
This option allows one to cut out a rectangular region from the image manually. Define the new corner points in the image.

### 5.4.3   Flip contrast image
This function reverses the image contrast

### 5.4.4   Make image background
By using this function, a part of the image intensities can be made equal to a background intensity. This function is useful to remove image contributions from neighbouring nanoparticles for example. First, you define the background value. This can be done by giving a value, selecting a region in the current image or in another image. Then a region is selected in the current image where the image intensities should be equal to the given background value. Note, that when selecting regions, corner points are defined. To close the region, you can right click on the image. The function can be aborted by pressing *escape*.

### 5.4.5 Normalise image

This function allows one to normalise the image intensities by using the formula:

$$I_{norm} = \frac{I_{image} - I_{vac}}{I_{max} - I_{vac}} \qquad (5.1)$$

If values are unknown, they can be computed from a recorded detector scan/map. An extra option is to convert the normalised image to electron counts, by using the incident electron dose. The experimental dose can be calculated by using the dwell time ($\mu$s) and beam current (pA).
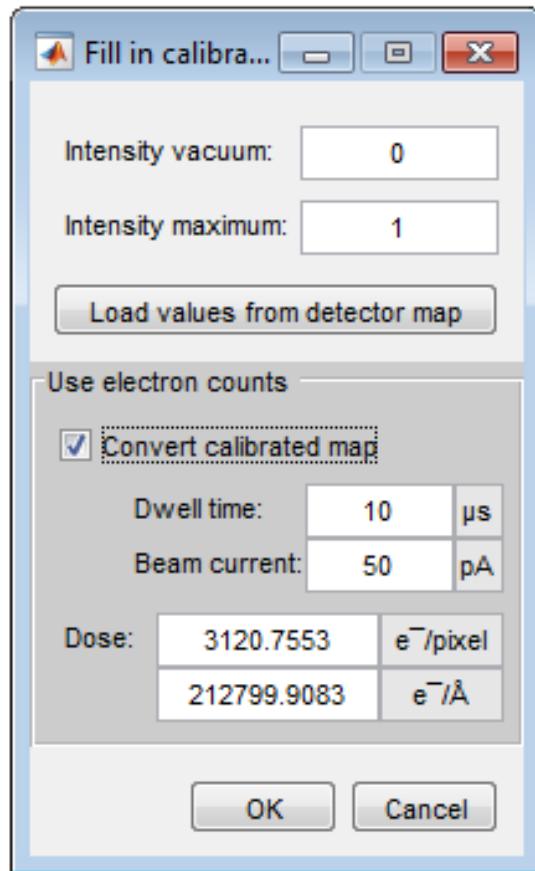
Figure 5.2: Parameters to normalise an image. $I_{vac} = 0$ and $I_{max} = 1$. The option convert calibrated map to electron counts is enabled, to multiple the image intensities with the dose value 3120.7553.

## 5.5   Import peak locations

In this section, a MATLAB (*.mat*) or text (*.txt*) file can be loaded containing starting coordinates. The starting coordinates have to be defined in Ångström in a (n × 2) vector containing the x- and y-coordinates. Furthermore, another StatSTEM file can be loaded to use its starting or fitted coordinates.
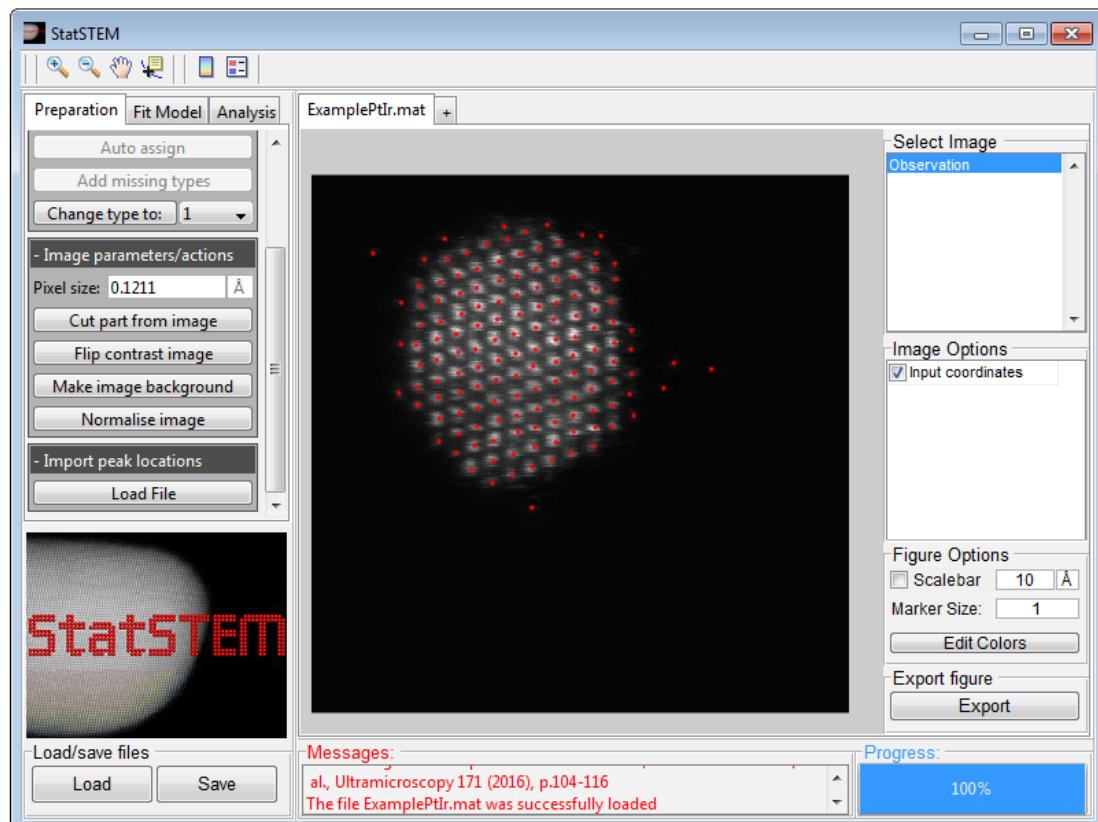


Figure 5.3: An experimental HAADF STEM image of a Pt/Ir sample with defined starting coordinates.

# 6 Fit Model

From this panel, a Gaussian model can be fitted to the image. Here, the standard procedure can be used, when all starting coordinates are defined. Otherwise, the model selection method can be used.

## 6.1 Standard procedure

In this standard procedure, each atomic column will be modelled as a Gaussian peak. Detailed information about the fitting routine can be found in Ref. [1]. The options in this panel are only available if at minimum one starting coordinate is defined. The model can be fitted by hitting the *Run fitting routine* button. When the model is fitted, the fitted Gaussian model with the fitted coordinates will be shown (see Fig. 6.1). The fitting procedure can be aborted by hitting the *Abort fitting routine.* This button may not always work as MATLAB sometimes cancels or postpones functions during a computation. A second (or third) hit usually works. When the procedure will be aborted, StatSTEM shows a message.
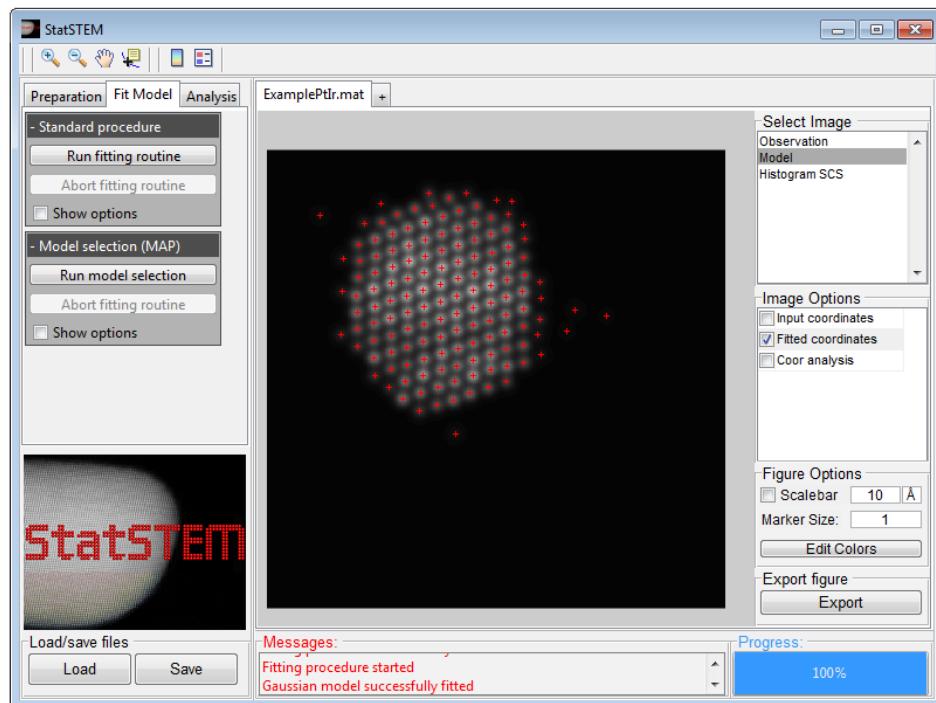


Figure 6.1: The fitted model of an experimental HAADF STEM image of a Pt/Ir sample with the fitted coordinates of the atomic columns.

Once the model is fitted, the fitted coordinates of the atomic columns and the total intensity of electron scattered by the atomic columns, the so-called scattering cross-sections, are calculated. These scattering cross-sections can be seen in the histogram of scattering cross-sections (see Fig. 6.2).
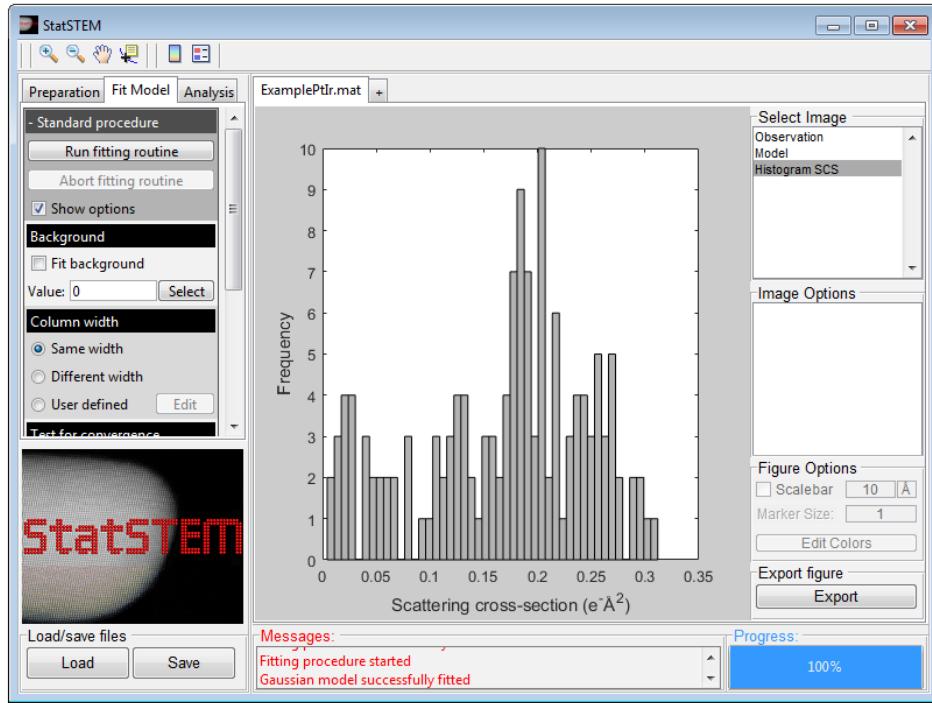


Figure 6.2: The histogram of scattering cross-sections of the atomic columns in an experimental HAADF STEM image of a Pt/Ir sample.

In the panel, different options of the fitting procedure may be defined (by using the *Show options* button). The following sections discuss the available options.

### 6.1.1 Background
In this options, the user can choose to fit a constant background. If no background will be fitted, a constant value may be given. By using the *Select* button, the mean value of a selected region in the image can be determined.

### 6.1.2 Width of atomic columns
In the fitting procedure, one may choose to fit Gaussian peaks to all columns having the same or a different width. In the *Same width* option, the estimated Gaussian peaks will have the same width for columns of the same atom type (as can be labelled in the preparation

step discussed in Chapter 5). In the *Different width* option, a different width is estimated for each Gaussian peak of an atomic column. In the *User defined* option, the user may define a width for each column type which will be used to find the atomic column positions. Usually, the *Same width* option is used as this is computationally less demanding and gives good results. Only when no information is known about the structure under study (the present elements are unknown), the *Different width* option is advisable. More information can be found in Ref. [1].

### 6.1.3 Test for convergence
This option may be used for testing the correctness of the starting coordinates and fitting parameters. The number of iterations will be limited to 4.

### 6.1.3.1 Re-use fitted coordinates
After a test fit is done, the fitted *intermediate* coordinates may be more close the true values than the starting values. Therefore, this option may be used to make the fitted coordinates the new starting coordinates.

### 6.1.4 Parallel computing
For improving computational speed, the fitting procedure uses parallel computing in which the calculations are divided over the different CPU cores of your computer. The number of CPU cores used for parallel computing may be reduced to lower the CPU usage during the fitting procedure (be aware that total calculation time will most likely increase).

## 6.2 Model selection (MAP)

This method can be used to automatically identify the atomic columns and fit a Gaussian model. *More information will be given in an updated manual.*

# 7 Analysis

In this panel, the fitted parameters of the model may be used for further analysis. In version 3.0, options are available to do atom counting, column coordinate based analysis and to create a (simple) 3D model. More options may become available in future versions.

## 7.1 General options

In this panel, general options that influence a further analysis are given.

### 7.1.1 Select new model from MAP

When using the model selection (MAP) method to fit a Gaussian model with the correct number of columns, models a fitted for a range of number of columns. By using this function, you can select a new model (with a specific number of columns) that will be used for further analysis.

### 7.1.2 Show models from MAP

By using this option, you can open a new window where you can see the fitted models by the model selection (MAP) method in function of the number of columns.

**Select columns in image**

Select a region in the image. The columns in this region will be taken into account for further analysis. The columns outside this region are neglected. By pressing *escape*, the routine will be aborted.

**Select columns in histogram**

This option allows one to exclude outliers in the histogram of scattering cross-sections. First the lower limit must be defined, then the upper limit. The scattering cross-sections of the columns that fall outside the limit are neglected in a further analysis. By pressing *escape*, the routine will be aborted.

**Select columns on type**

By using this option, columns can be selected based on the column type labels given in the preparations step (see also Chapter 5).

## 7.2 Index columns

In this panel, you can index the columns in terms of distance from a reference point. This operation should be done before performing a column coordinate based analysis (see section 7.6) or creating a 3D model (see sections 7.3 and 7.4).

### 7.2.1 Projected unit cell

A necessary input to index columns is the location of the different column positions in a projected unit cell, mentioned in section 5.3.1. If one intends to make a 3D model later on, z-information must be given as is shown in the Fig. 7.1. Then it becomes possible to define the lattice parameter in c-direction and per column one can define the depth location of each atom in this column. Atoms in a column can be added and removed by the buttons *New* and *Delete*. All atoms except the first one can be deleted by using the *Clear* button. Each column can be selected from the pop-up window in the right-top corner. A database is added to StatSTEM for some common materials and viewing directions. Make sure that you fill in the correct pixel size of the image as the lattice parameters in the projected unit cell should be close to the experimental values.
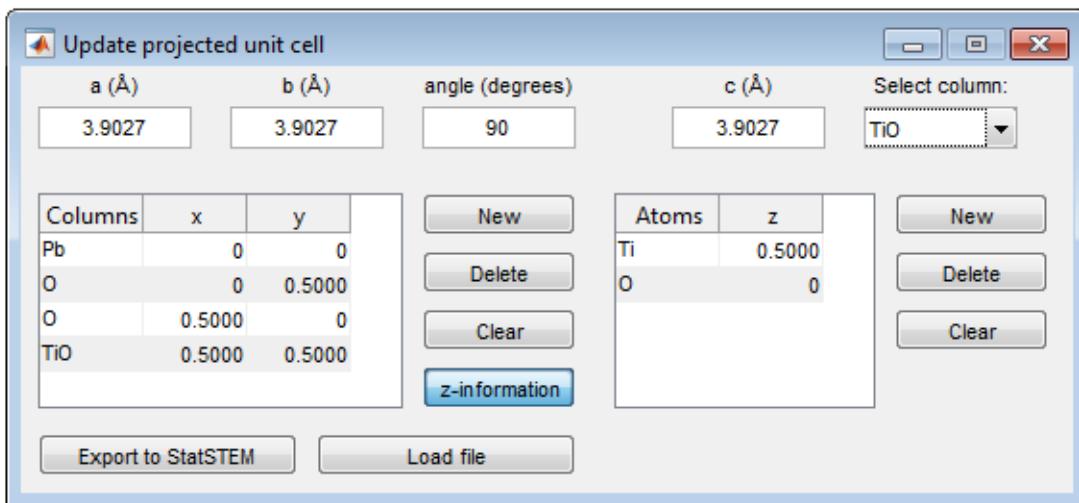


Figure 7.1: A project unit cell can be made by defining the projected lattice constants (a and b) and the relative positions of the columns. Additional information can be given on the depth location of the atoms in the columns.

### 7.2.2 Start indexing

By using this option, all coordinates will be indexed as a function of distance in unit cells from a reference coordinate. For this procedure first a reference coordinate will be chosen, then the lattice directions are found and the indexing procedure starts. If the automatic routines in StatSTEM fail, you can guide StatSTEM by using the advanced options listed below. The details of the automatic routine are described in Appendix B.

**Reference coordinate**

In this section a reference coordinate can be chosen. A displacement map is made from this coordinate. Furthermore, this coordinate gets the index (0,0) during the creation of the strain map. One can choose between different column types for selecting a reference coordinate. Automatically StatSTEM will use the most central coordinate as a reference, this can be change by manually selecting another coordinate.

**Direction a lattice**

For finding the direction of the lattice, an automatic routine can be used or a manual input can be given. The automatic routine used the projected unit cell parameters to identify the lattice direction. Here, the distance of the neighbouring coordinate with respect to the reference coordinate is compared to the given lattice parameters in the projected unit cell. Once the direction is found, the lattice parameters are automatically improved by fitting (this can be disabled). Here, a box of N unit cells (standard 3 UC to each side) around the reference coordinate is used for finding the values of the a (and b) lattice parameter in the image. This option is advised to be used, as the pixel size recorded by an electron microscope is never that accurate. Be however, aware that this option changes the values of a and b and should only be turned off when you are 100% sure about the pixel size.

## 7.3 Atom counting: statistical method

In HAADF STEM imaging, a statistics based method is developed to count the number of atoms based on the scattered cross-sections (the total intensities of electrons scattered by the atomic columns), which increase monotonically with thickness [5]. ***Be aware that this method is only reliable when only one column type is present.*** In this statistics based method, the scattering cross-sections are presented in a histogram. Owing to a combination of experimental noise an residual instabilities, broadened - rather than discrete - components are observed in such a histogram. Therefore, these results cannot directly be interpreted in terms of number of atoms. By evaluation of the so-called integration classification likelihood (ICL) criterion in combination with Gaussian mixture model estimation, the number of components and their respective locations can be found. From the estimated locations of the components, the number of atoms can be quantified. More information can be found in Refs. [2, 3, 4].
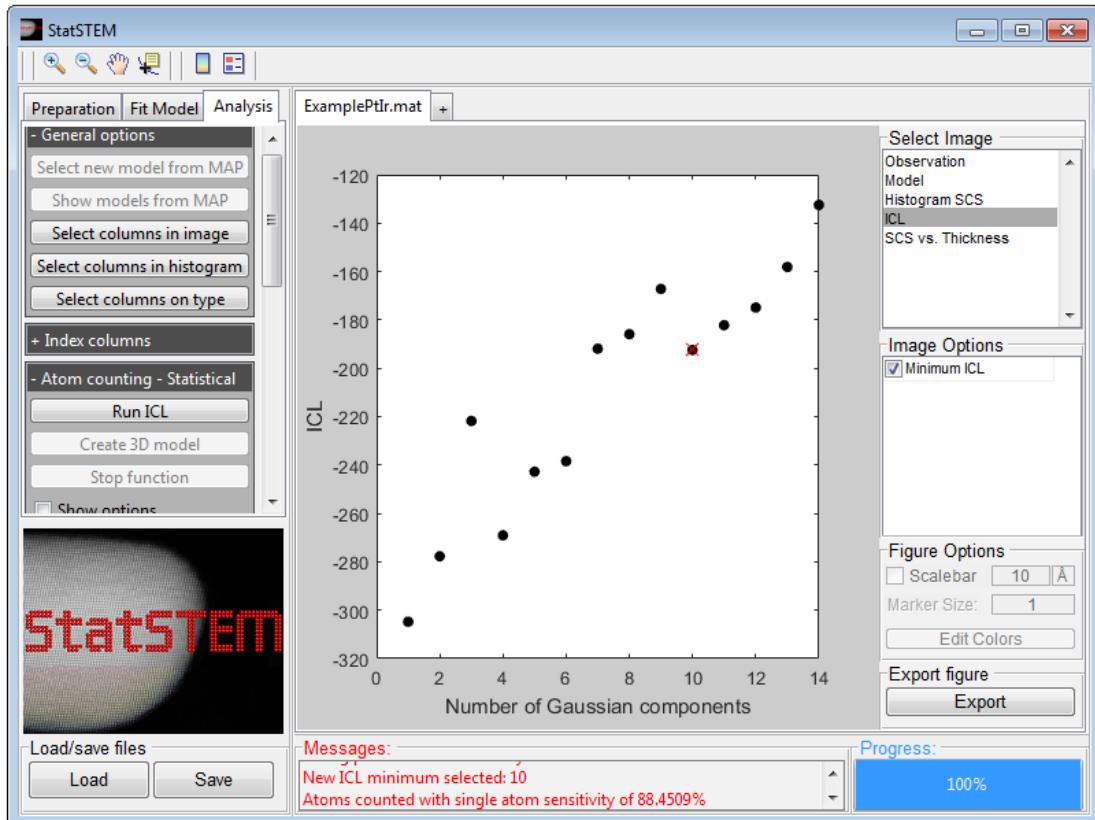


Figure 7.2: The ICL for an experimental image. A local minimum appears at 10.

### 7.3.1 Pre-analysis

**Maximum number of components**

In order to evaluate the ICL, an upper limit on the number of components must be given. Up to a given number of components, Gaussian mixture models are fitted to the histogram of scattering cross-sections and the ICL criterion is determined. A rough estimate for this upper limit can be obtained by using the shape of the particle under study.

### 7.3.2 Post-analysis

After selecting atomic columns and the maximum number of components, the Gaussian mixture model may be fitted to the histogram of scattering cross-sections by using the *Run ICL* button. For each number of components, the ICL-criterion is determined and displayed. For atom counting, one searches for local minima in the ICL-curve (see for example Fig. 7.2). After the computation, StatSTEM asks the user to select a local minimum. After a local minimum is selected, the experimental image with the corresponding atom counts will be shown (see Fig. 7.3). The computation of the ICL values can be aborted by pressing the *Stop function* button. The button may not always work as MATLAB cancels many functions during a com-
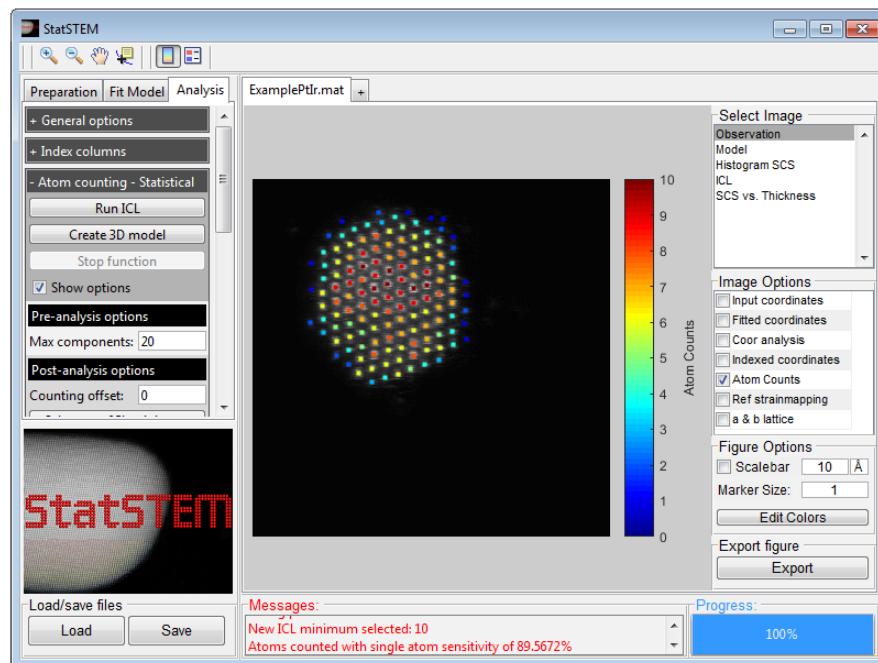


Figure 7.3: The atom counts of the different atomic columns in the experimental HAADF STEM image of a Pt/Ir particle.

putation. A second (or third) hit usually works. Once the procedure will be aborted, StatSTEM will show a message asking whether the user wants to select a local minimum in the current ICL graph.

**Counting offset**

Once a local minimum in the ICL curve is selected, counting results may be rescaled by giving the counts an offset. This is particularly useful for thick particles in which no thin columns are present.

**Select new ICL minimum**

With this option a new local minimum in the ICL curve can be selected.

### 7.3.3 Create 3D model

Once the atom are counted per atomic column and the columns are indexed, a 3D model can be made. Here, the atom are distributed symmetrically along the z-direction. ***Note that this 3D model is only a simple model, to give an idea about the 3D shape of the particle. It can not be used a final result!*** Note that for this function the project unit cell should contain z-information (see section 7.2.1). The 3D model that will shown is color coded, meaning that each atom type will have a different color.
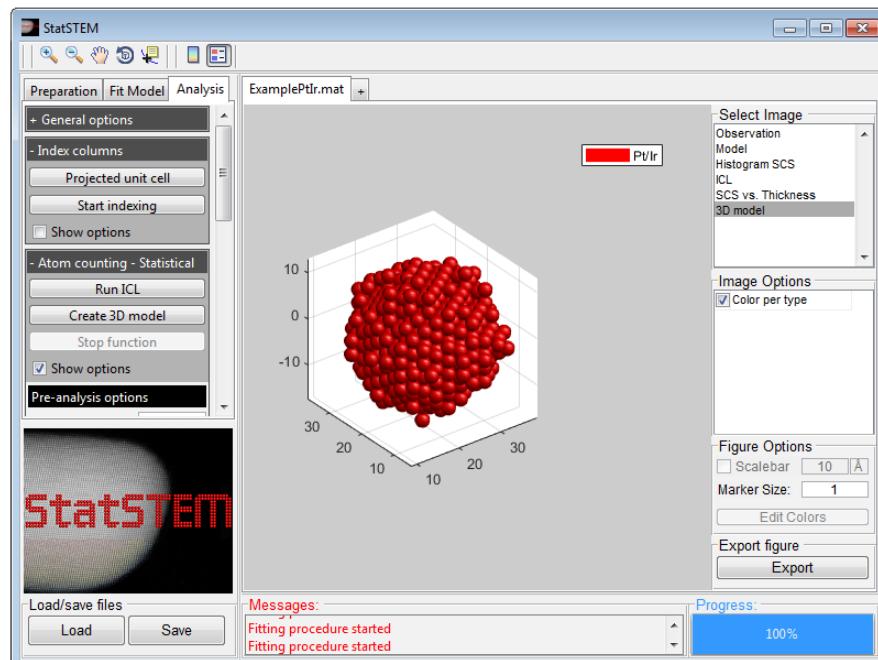


Figure 7.4: A 3D model of a Pt/Ir particle.

## 7.4 Atom counting - Simulation based

Another method for doing atom counting is by comparing simulated library's of scattering cross-sections with the experimentally measured ones.

### 7.4.1 Match with simulations

By clicking on this button, first a library should be loaded. Then, atoms counts are computed by comparing the measured scattering cross-sections from the fitted model to the loaded library of simulated scattering cross-sections. A MATLAB (*.mat*) or text (*.txt*) file can be loaded with simulated values of the scattering cross-sections. The simulated values in function of column thickness must be stored in a (n × 1) vector. In the figure *SCS vs. Thickness*, the scattering cross-sections obtained by the statistics based method can be compared to the simulated values (see Fig. 7.6).
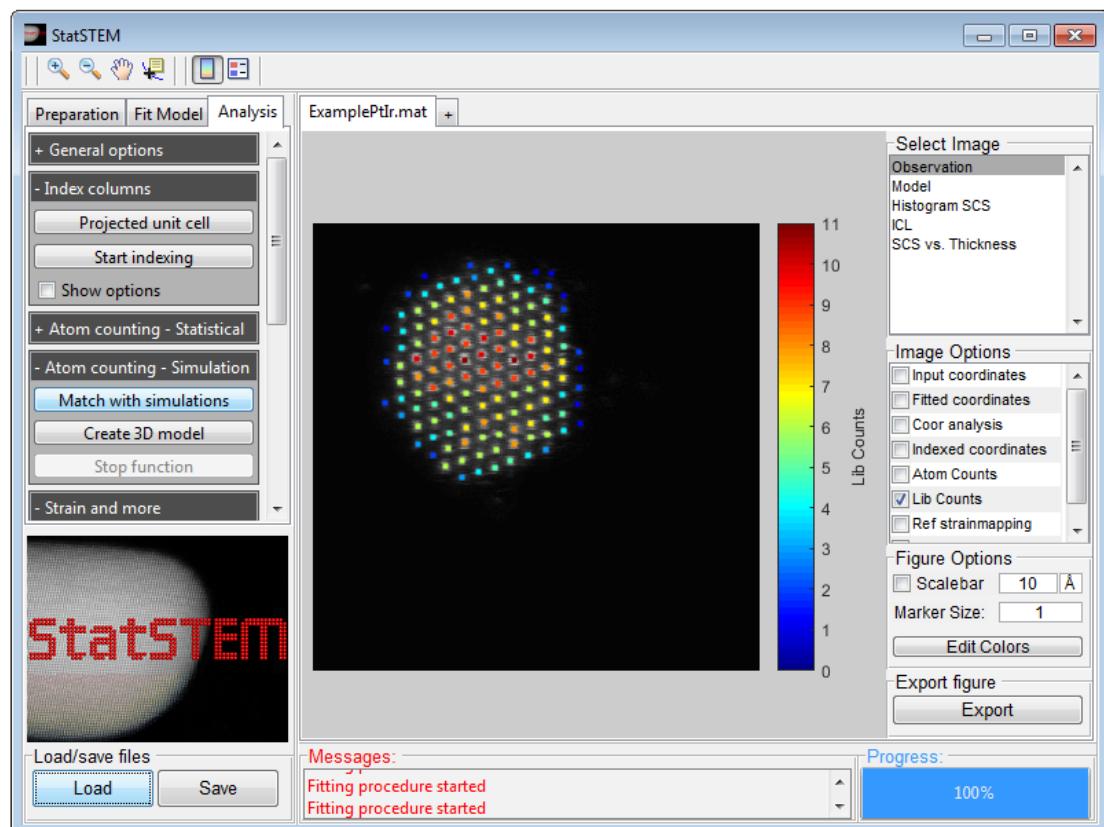


Figure 7.5: The atom counts of the different atomic columns in the experimental HAADF STEM image of a Pt/Ir particle obtained by a comparison with simulated library values of SCS.
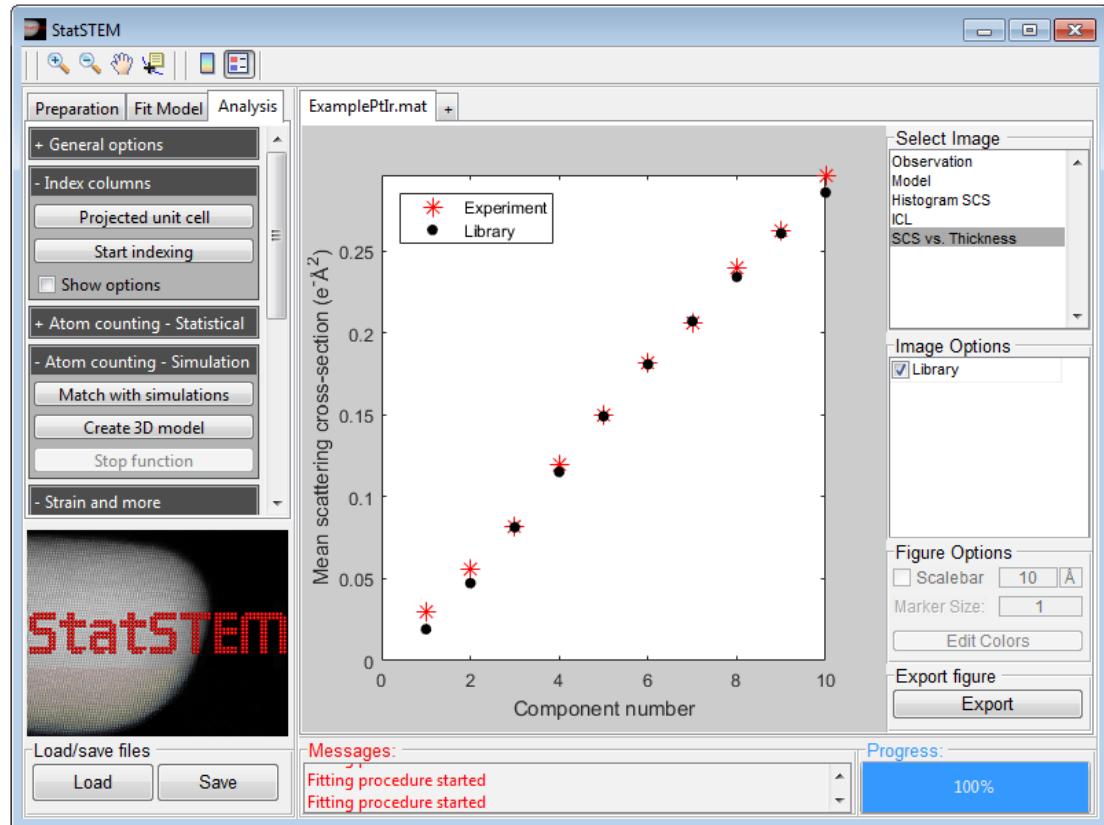
Figure 7.6: A comparison between the scattering cross-sections obtained by the statistics-based method (Experiment) and image simulations (Library).

### 7.4.2 Create 3D model

Once the atom are counted per atomic column and the columns are indexed, a 3D model can be made. Here, the atom are distributed symmetrically along the z-direction. ***Note that this 3D model is only a simple model, to give an idea about the 3D shape of the particle. It can not be used a final result!*** Note that for this function the project unit cell should contain z-information (see section 7.2.1). The 3D model that will shown is color coded, meaning that each atom type will have a different color.

## 7.5 3D model

In sections 7.3.3 and 7.4.2, it is mentioned that a 3D model can be made from atom counting results. Once the model is made, this panel provide options to export the coordinates as an XYZ file or to calculate the coordination number.

### 7.5.1 Save model as XYZ

The 3D model can be saved as an XYZ file that can be loaded into other software packages such as Vesta or VMD.

### 7.5.2 Coordination number

The function can be used to determine the coordination number of each atom in the 3D model. The coordination number is determined by calculation the number of neighbours each atom has within a specific radius. In a fcc material this radius is $r = a/\sqrt{2}$.
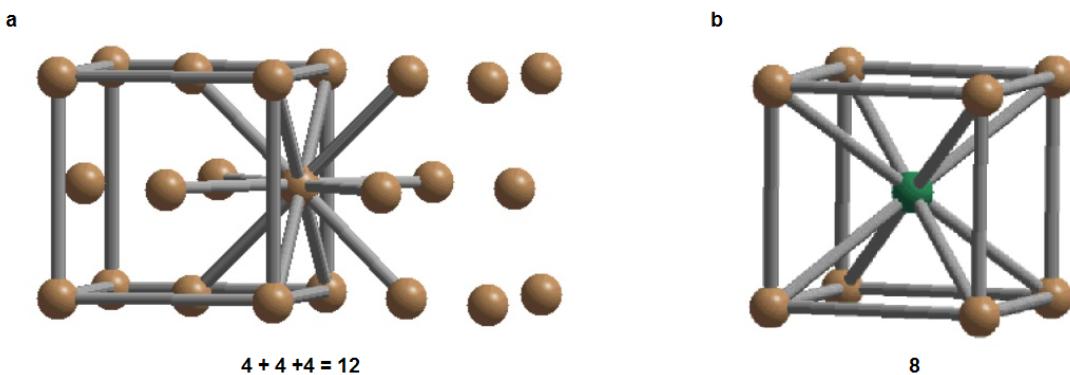


Figure 7.7: The coordination number for (a) an FCC and (b) an BCC crystal can be calculated by searching for neighbouring atoms within a radius of $a/\sqrt{2}$ or $2*a/\sqrt{3}$, respectively.

**Radius**

In StatSTEM, the standard radius that is used is $a*0.8$, which is a little bit larger than $a/\sqrt{2}$ to compensate for small fluctuations in the atom positions.

**Number of atoms**

Standard the coordination number will be determined for all atoms. As this is a demanding calculation, the can leave out the most central atoms for this calculation. These atoms are determined based on the distance from the centre of the particle. In this manner, one can calculate the coordination number only for a fraction of the atoms in the particle.
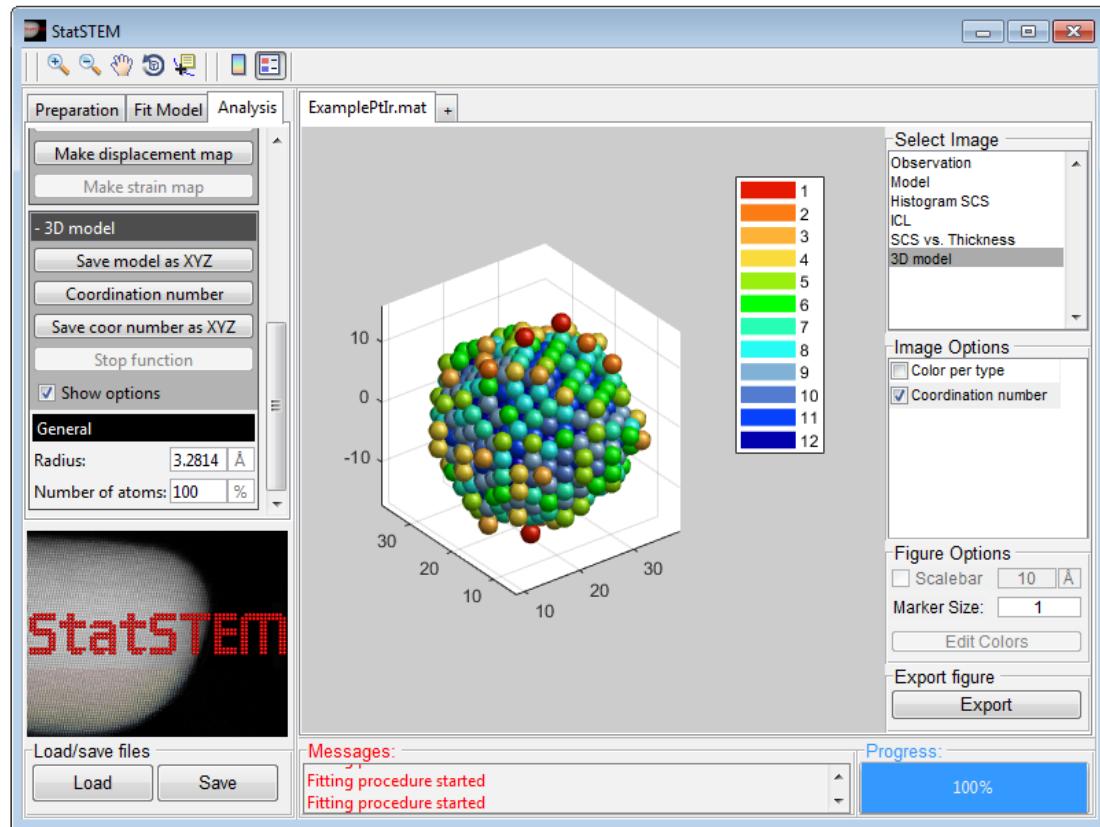
Figure 7.8: A 3D model of a Pt/Ir particle indication the coordination number per atom.

### 7.5.3 Save coor number as XYZ

Once the coordination number is determined, it can be saved as a XYZ file. Here, a specific atom type is given in function of the coordination number. The types as a function of coordination number are listed below in Table 7.1.

| Coordination number | Atom type | Coordination number | Atom type |
|---|---|---|---|
| 1 | V | 7 | Lu |
| 2 | Mg | 8 | Yb |
| 3 | Au | 9 | Al |
| 4 | Na | 10 | Np |
| 5 | Se | 11 | Ho |
| 6 | Zr | 12 | Co |

Table 7.1: Atom type per coordination number that is used when storing the 3D model with coordination numbers as a XYZ file.

26

## 7.6 Strain and more

In this panel, different methods are available to use the fitted coordinates of the model for a further analysis. Here, the lattice parameter, displacement of atoms, octahedral tilt and strain map can be made.

### 7.6.1 Lattice of type

This function determines the lattice parameter per column type. Automatically a coloured plot is made that show the lattice parameter, as is illustrated in Fig. 7.9. When this function is used, one can also look at plots where the lattice parameters as a function of distance from the reference coordinate are shown, both in a- and b-direction.
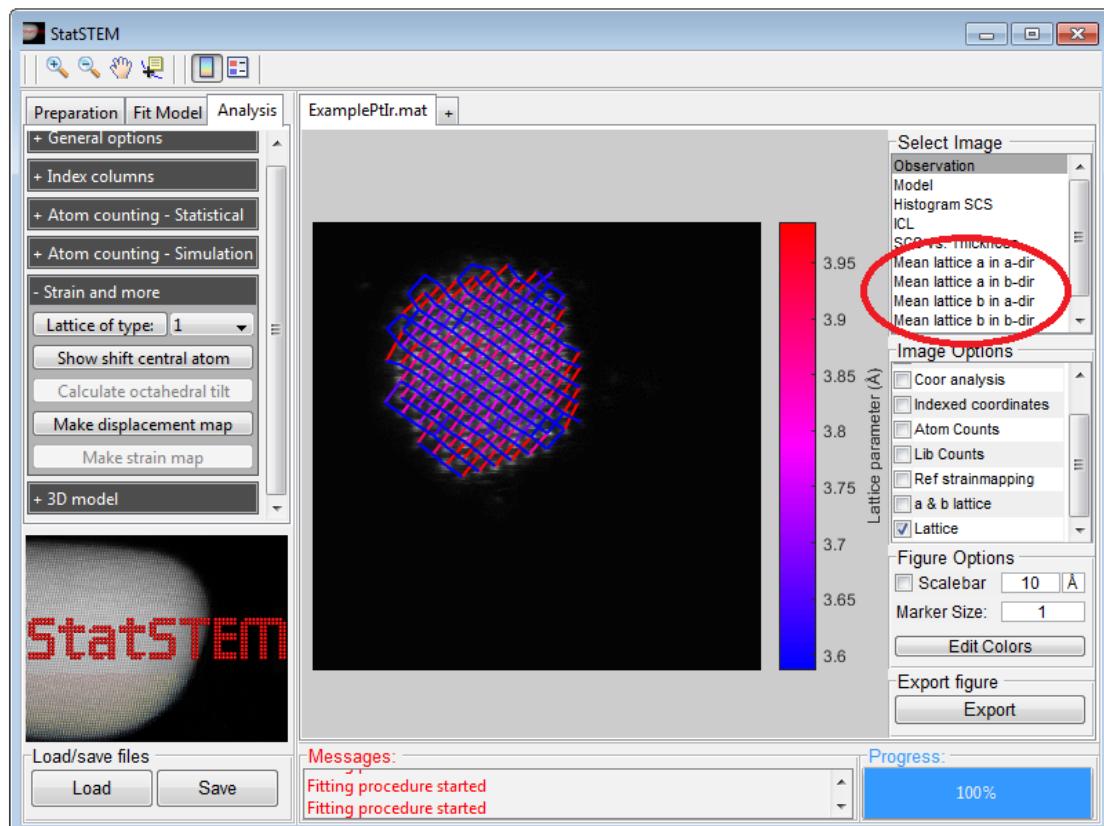


Figure 7.9: The lattice parameters a and b of a Pt/Ir nanoparticle. The options to plot the lattice parameter a or b as a function of distance in the a- or-direction are indicated by the red circle.

### 7.6.2 Show shift central atom

By clicking on this button a displacement map will be made for the central atoms in a unit cell based on the column indexing described in section 7.2. By using the values of the projected unit cell, the expected coordinate are calculated. The displacement map is generated by comparing the expected coordinate with the measured coordinates.
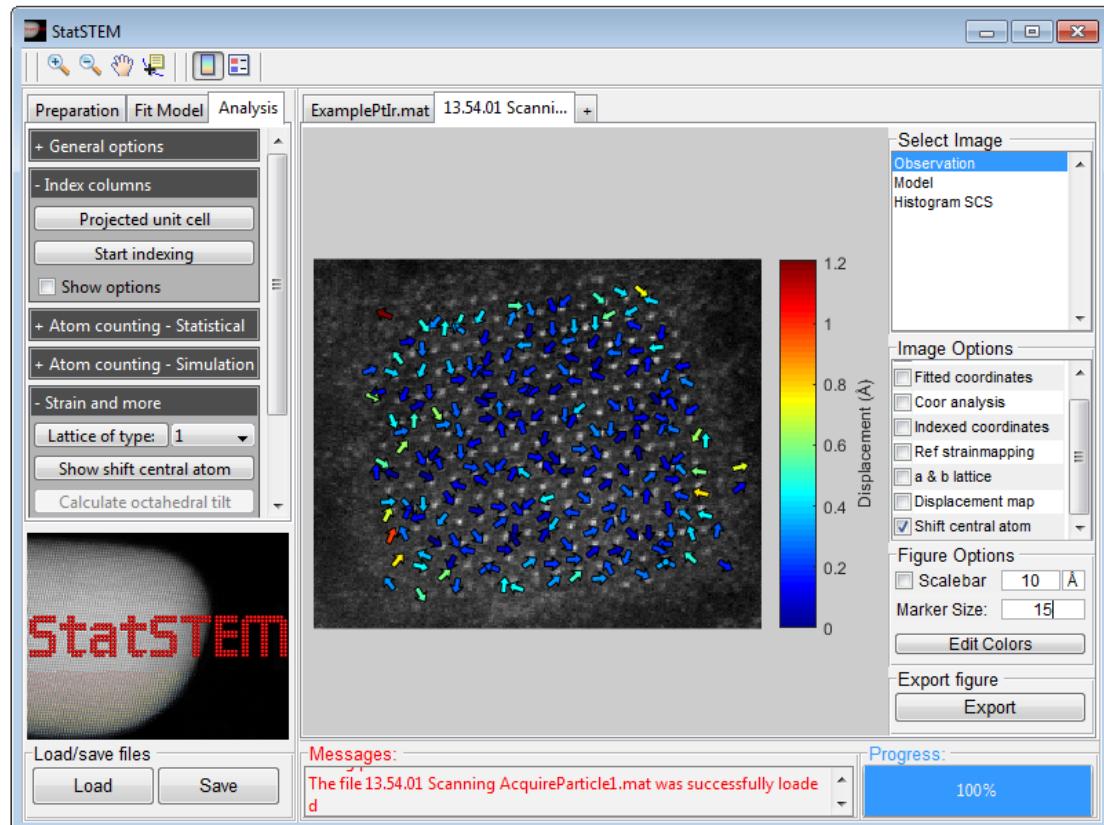


Figure 7.10: The displacement of the central atom in PbCsBr$_3$.

### 7.6.3 Calculate octahedral tilt

In perovskite particles, there is an oxygen octahedra present surrounding the B-cation (or A-cation). Due to internal strain, the oxygen octahedral can rotate. From the indexed columns, octahedral tilt can be determined when there are atoms present at the relative positions in the unit cell: (0,0.5) and (0.5,0). If this condition is satisfied, this function becomes available to determine the octahedral tilt as a function of distance in the a- and b-direction. The distance is measured from the reference coordinate selected when indexing the columns, as described in section 7.2. In the plot, the octahedral tilt is calculate where it is assumed that the octahedral tilt is alternation between a clockwise and a anti-clockwise rotation.
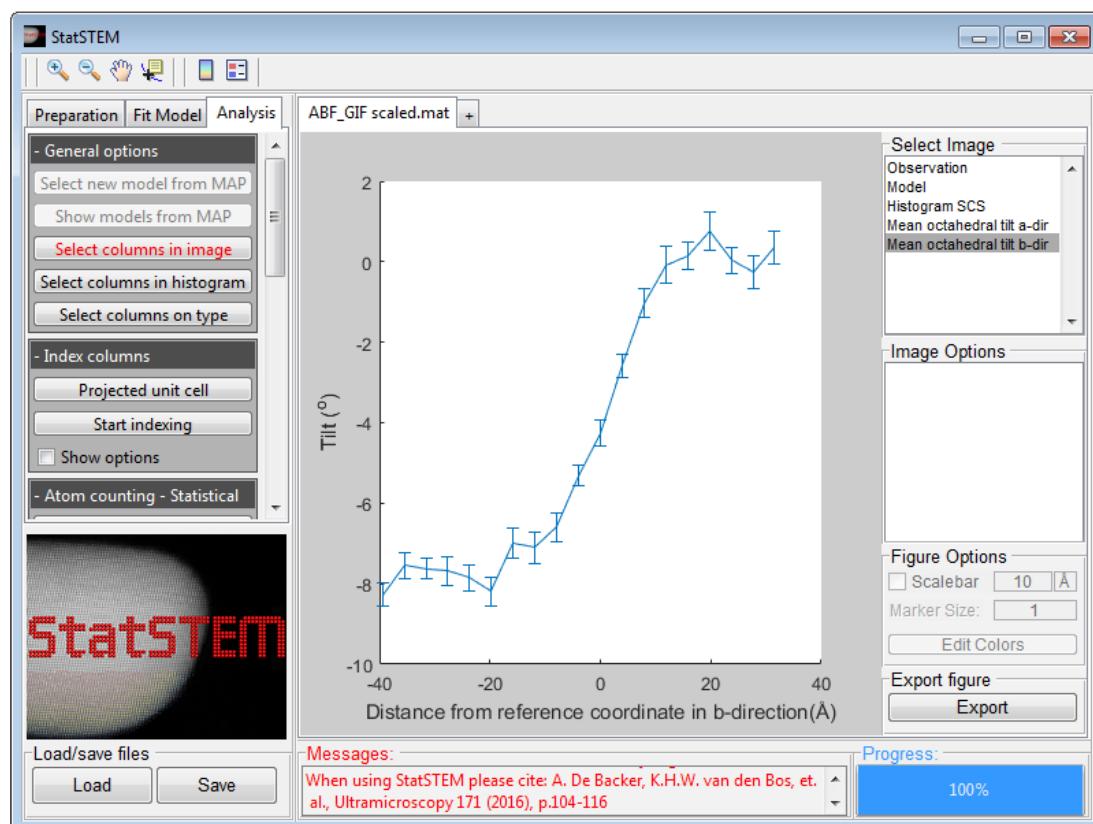


Figure 7.11: The octahedral tilt along the b-direction in $PtTiO_3$.

### 7.6.4 Make displacement map

By clicking on this button a displacement map will be made based on the column indexing described in section 7.2. By using the values of the projected unit cell, the expected coordinate are calculated. The displacement map is generated by comparing the expected coordinate with the measured coordinates.
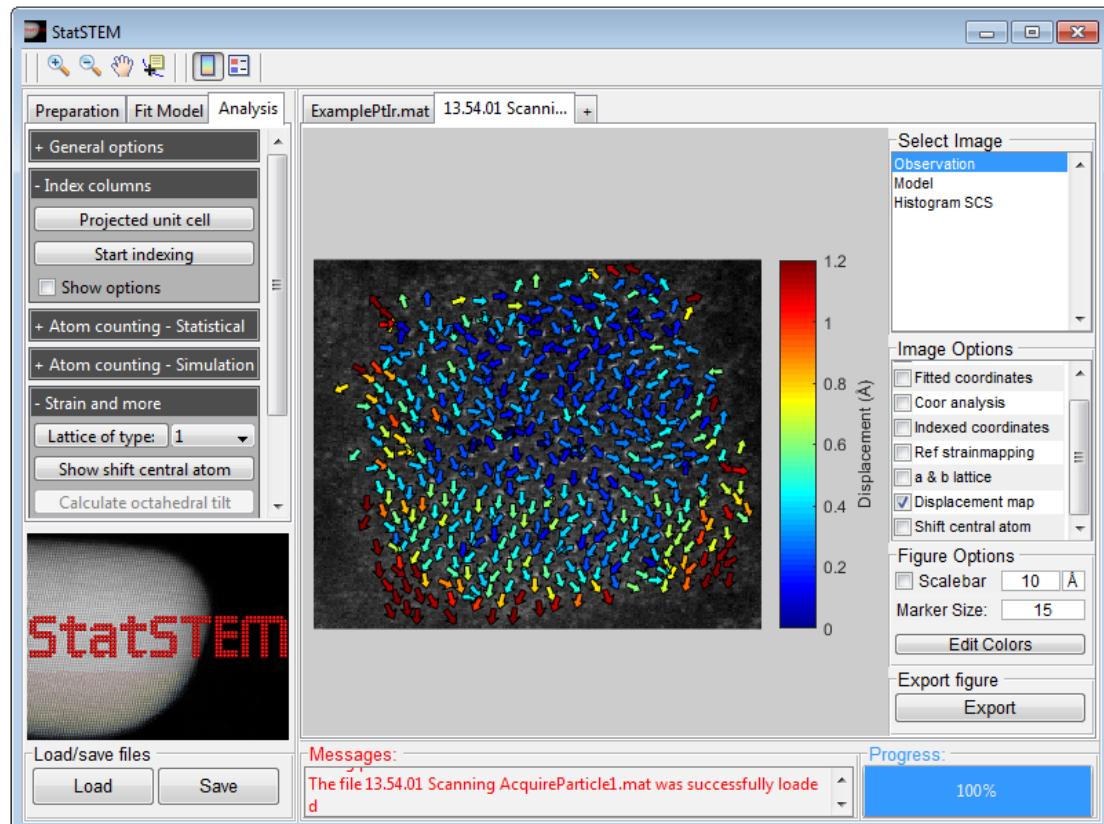


Figure 7.12: A displacement map of a PbCsBr$_3$ particle.

### 7.6.5 Make strain map

By clicking on this button, the $\epsilon_{xx}$, $\epsilon_{xy}$, $\epsilon_y$ and $\omega_{xy}$ strain maps will be generated. Here, the derivative of the displacement map is used [6]. An example is shown in Fig. 7.13.
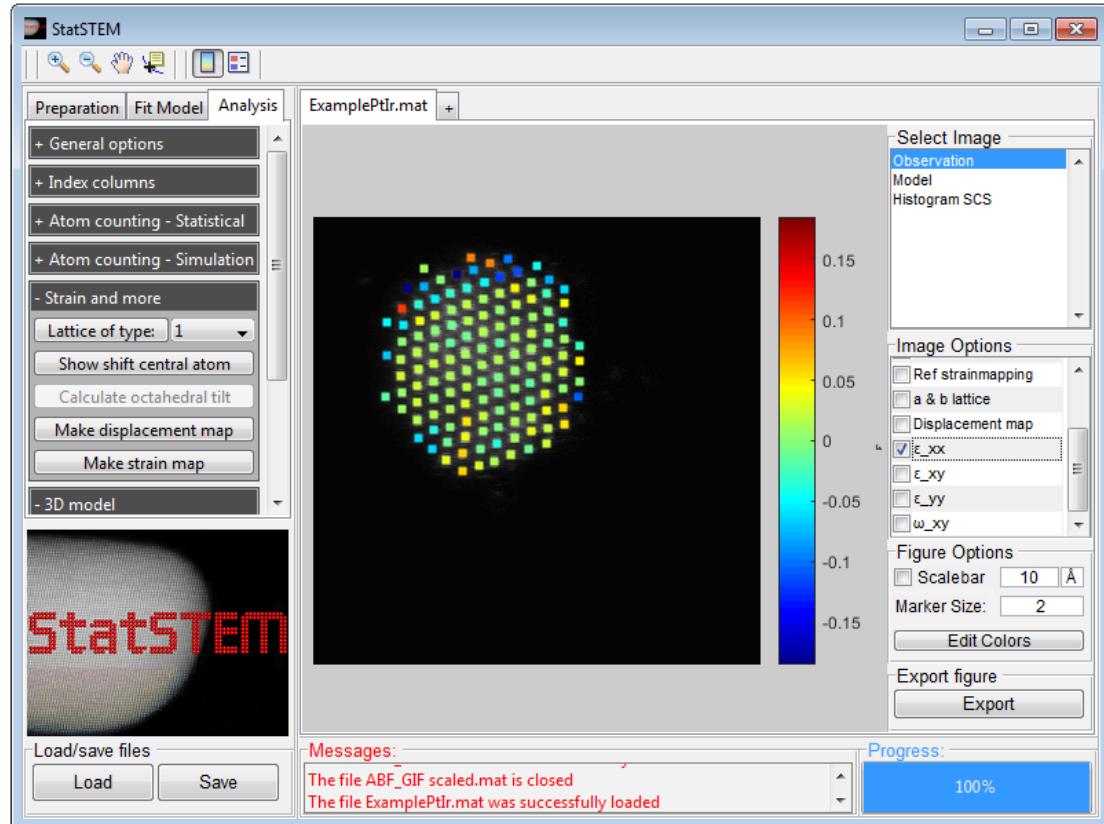


Figure 7.13: An $\epsilon_{xx}$ strain map of a Pt/Ir particle.

# 8   Closing StatSTEM

StatSTEM can be closed by hitting the red cross in the top-right corner. When closing StatSTEM, make sure that all files are saved as no warnings of unsaved files are given.

# 9   Remarks and suggestions

When downloading MATLAB, a folder with examples is included. In this folder MATLAB scripts are present to use StatSTEM without the graphical user interface. Our StatSTEM website offers a forum in case you have any questions, remarks or suggestions. New releases will be announced on this website.

# 10 References

[1] Annick De Backer, Karel H. W. van den Bos, Wouter Van den Broek, Jan Sijbers, and Sandra Van Aert. StatSTEM: An efficient approach for accurate and precise model-based quantification of atomic resolution electron microscopy images. *Ultramicroscopy*, 171:104–116, 2016.

[2] S. Van Aert, K. J. Batenburg, M. D. Rossell, R. Erni, and G. Van Tendeloo. Three-dimensional atomic imaging of crystalline nanoparticles. *Nature*, 470:374–377, 2011.

[3] S. Van Aert, A. De Backer, G. T. Martinez, B. Goris, S. Bals, and G. Van Tendeloo. Procedure to count atoms with trustworthy single-atom sensitivity. *Physical Review B*, 87(064107), 2013.

[4] A. De Backer, G. T. Martinez, A. Rosenauer, and S. Van Aert. Atom counting in HAADF STEM using a statistical model-based approach: methodology, possibilities, and inherent limitations. *Ultramicroscopy*, 134:23–33, 2013.

[5] G. T. Martinez, L. Jones, A. De Backer, A. Bch, J. Verbeeck, S. Van Aert, and P. D. Nellist. Quantitative STEM normalisation: The importance of the electron flux. *Ultramicroscopy*, 159:46–58, 2015.

[6] Pedro L Galindo, Sławomir Kret, Ana M Sanchez, Jean-Yves Laval, Andrés Yáñez, Joaquín Pizarro, Elisa Guerrero, Teresa Ben, and Sergio I Molina. The peak pairs algorithm for strain mapping from hrtem images. *Ultramicroscopy*, 107(12):1186–1193, 2007.

# Appendix A  Structure of saved files

StatSTEM files may be saved in a MATLAB (*.mat*) format. In this file different structures are created in which all variables are stored. A stored files belong to a class named StatSTEMfile. For working with the files, make sure StatSTEM is loaded to the path of MATLAB. For each step or analysis, a structure is created. In the class files, a description is given of the parameters (what the hold and in what the purpose is).

# Appendix B    Indexing/strain mapping

Statistical parameter estimation theory is capable of extracting atomic column positions with high accuracy and precision from (S)TEM images. A direct comparison of the measured column positions with the expected column positions of an ideal crystal lattice gives the displacement of the atomic columns (see Fig. B.1(b)). By using a first derivative, these measured displacement vectors can be used to compute atomically resolved strain maps (see Figs. B.1(c) - B.1(f)). For creating a displacement map, a projected unit cell is used which describes the projected lattice parameters and the relative positions of the different columns. Next, a reference coordinate is given in an unstrained area. Here, the distances of the nearest neighbouring atomic columns are compared to the projected lattice parameters to find the lattice directions in the image (see Fig. B.1(a)). These lattice directions are used to predict the column positions of an ideal structure. Since the pixel size given by the microscope may contain inaccuracies, the dimensions of the lattice parameters are refined by fitting the ideal column positions to the measured columns positions in a selected area of $N \times N$ unit cells around the reference coordinate. Standard, an area of $3 \times 3$ unit cells is used, as is indicated by the blue region in Fig. B.1(a).

After the lattice parameters are found, the column positions of an ideal crystal are predicted. As the crystal structure under study may be heavily strained, measured column positions can be shifted by distances larger than the lattice parameter, hampering a direct identification of the predicted and measured column positions. Therefore, starting from the reference coordinate the neighbouring atomic columns are identified by using the refined lattice parameters. Next, the identified columns serve as new starting points to identify their neighbouring columns. This procedure continues until the boundary of the image or the edge of the particle is reached. During this process, lattice parameters and their directions are continuously updated as strain is usually non-uniform throughout a particle. Here, the lattice parameters $\boldsymbol{a} = (a_x, a_y)$ and $\boldsymbol{b} = (b_x, b_y)$ in the $\boldsymbol{a}$- and $\boldsymbol{b}$-lattice directions are updated by using a scaling factor $\nu$:

$$\boldsymbol{a}^{(new)} = \boldsymbol{a}^{(old)} + \left( \boldsymbol{a}^{(meas)} - \boldsymbol{a}^{(old)} \right) \cdot \nu \qquad \text{(B.1)}$$
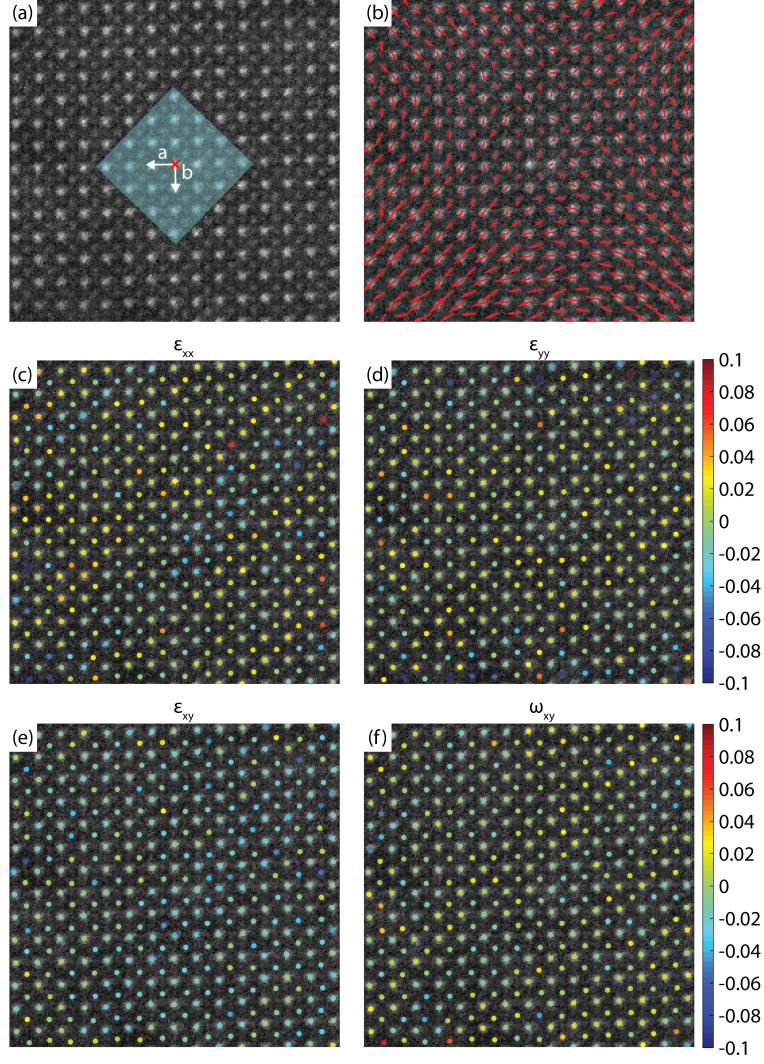
Figure B.1: Strain mapping procedure on an experimental HAADF STEM image of a CsPbBr$_3$ nanoparticle. (a) The procedure start by selecting a reference coordinate, indicated by the red cross. Next, the dimensions of a projected unit cell of the ideal structure are used to find the lattice directions $\boldsymbol{a}$ and $\boldsymbol{b}$. As the pixel size in the image might be inaccurate, a region of $3 \times 3$ unit cells, indicated in blue, is selected around the reference coordinate to refine the lattice parameters by using a fitting procedure. (b) A displacement map, created by comparing the measured column positions to the positions in an ideal unstrained structure. The first derivatives of the displacement map to the different lattice directions are used to determine the (c) $\epsilon_{xx}$, (d) $\epsilon_{yy}$, (e) $\epsilon_{xy}$ strain maps and (f) $\omega_{xy}$ rotation map
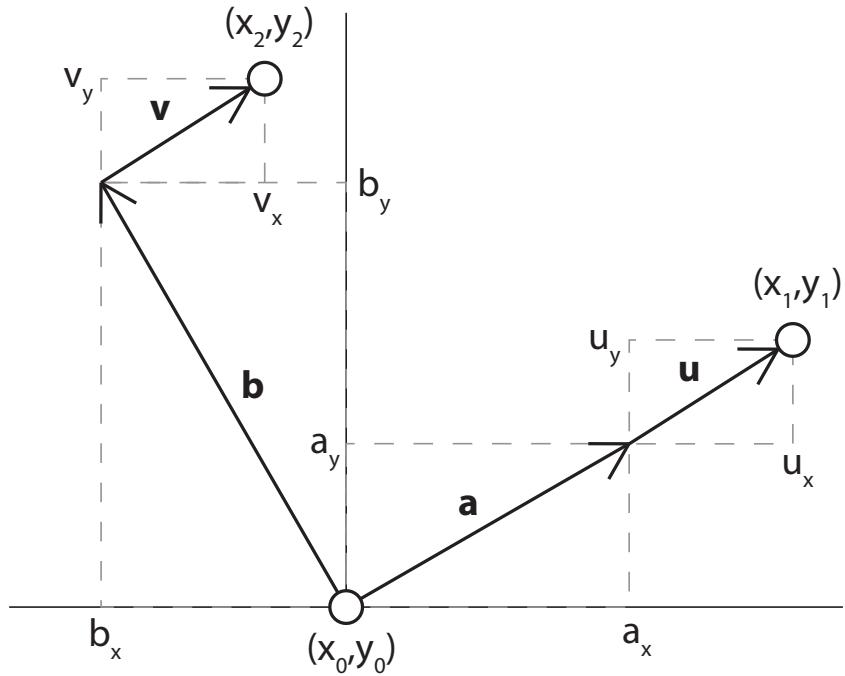
Figure B.2: Diagram of the calculation of displacement vectors $\boldsymbol{u}$ and $\boldsymbol{v}$ from two positions in the $\boldsymbol{a}$ and $\boldsymbol{b}$ lattice direction.

where $\boldsymbol{a}^{(new)}$ is the new lattice parameter, $\boldsymbol{a}^{(old)}$ is the old lattice parameter and $\boldsymbol{a}^{(meas)}$ is the measured lattice parameter. When all the measured atomic columns are identified, the displacement map is generated by determining the distance between the measured column positions from the image and the predicted column positions for an ideal structure (Fig. B.1(b)).

In the next step, the measured displacements of the atomic column positions with respect to an ideal structure are used to calculate strain maps. Here, the first derivatives of the displacement vectors with respect to the lattice parameters give the different strain components [6]:

$$\epsilon_{xx} = \frac{\partial \boldsymbol{u}}{\partial x}, \quad \epsilon_{yy} = \frac{\partial \boldsymbol{v}}{\partial y},$$

$$\epsilon_{xy} = \frac{1}{2}\left(\frac{\partial \boldsymbol{u}}{\partial y} + \frac{\partial \boldsymbol{v}}{\partial x}\right) = \frac{1}{2}\left(\epsilon_1 + \epsilon_2\right), \quad \omega_{xy} = \frac{1}{2}\left(\epsilon_1 - \epsilon_2\right) \tag{B.2}$$

where $\boldsymbol{u} = (u_x, u_y)$ and $\boldsymbol{v} = (v_x, v_y)$ are the displacement vectors of an atomic column in the $\boldsymbol{a}$- and $\boldsymbol{b}$-lattice directions, respectively (Fig B.2). In the StatSTEM software, the displacement vectors per

atomic column are determined by using the average lattice parameters that are obtained by measuring the distance between the selected column and its neighbouring columns. The equations of the different strain components can be rewritten in a set of linear equations [6]:

$$\left.\begin{aligned} u_x &= a_x\epsilon_{xx} + a_y\epsilon_1 \\ u_y &= a_y\epsilon_{yy} + a_x\epsilon_2 \\ v_x &= b_x\epsilon_{xx} + b_y\epsilon_1 \\ v_y &= b_x\epsilon_{xx} + b_y\epsilon_2 \end{aligned}\right\} \qquad \begin{bmatrix} \epsilon_{xx} & \epsilon_2 \\ \epsilon_1 & \epsilon_{yy} \end{bmatrix} = \begin{bmatrix} a_x & a_y \\ b_x & b_y \end{bmatrix}^{-1} \begin{bmatrix} u_x & u_y \\ v_x & v_y \end{bmatrix} \qquad \text{(B.3)}$$

From these linear equations the values of the different strain components at each atomic column position can be computed (Figs. B.1(c)-B.1(f)). Furthermore, by using the standard deviation of the measured column positions, the error on the measured strain values can be computed, which is at maximum 0.008 in the strain maps presented in Figs. B.1(c)-B.1(f).