IOP Institute of Physics

Image-Based Simulation for Industry 2021

IBSim-4i

18-21 October 2021, Institute of Physics, London, UK

Optimisation and Simulation of X-ray images: Automatic registration of surface models on synchrotron microtomography data

Franck P. Vidal, Iwan T. Mitchell, and Jean Michel Létang

PRIFYSGOL BANGOR UNIVERSITY

INSA INSTITUT NATIONAL DES SCIENCES APPLIQUÉES LYON

Acknowledgements

- **Jean-Yves Buffière** and **Ce Xiao** of MATEIS laboratory (Lyon), and **Wolfgang Ludwig** of ESRF (Grenoble) for the new projection and CT data,
- **NVIDIA Corporation** for the donation of the NVIDIA TITAN Xp GPU used in the development and validation of gVirtualXRay,
- **Supercomputing Wales** for the use of its supercomputer,
- **European Commission** for the award of a career integration grant, and
- **All the volunteers** who participated to the user study.

Motivations

- The presence of **strong imaging artefacts** in **microtomographic X-ray** data makes the **CAD modelling process difficult** to carry out.
- A **user study** was conducted to **manually extract geometrical properties** from the CT slice.
- As an alternative to **manual measurements** and **traditional image segmentation techniques**, we proposed to **register CAD models** by deploying a realistic **X-ray simulation on GPU** in an **optimisation** framework.

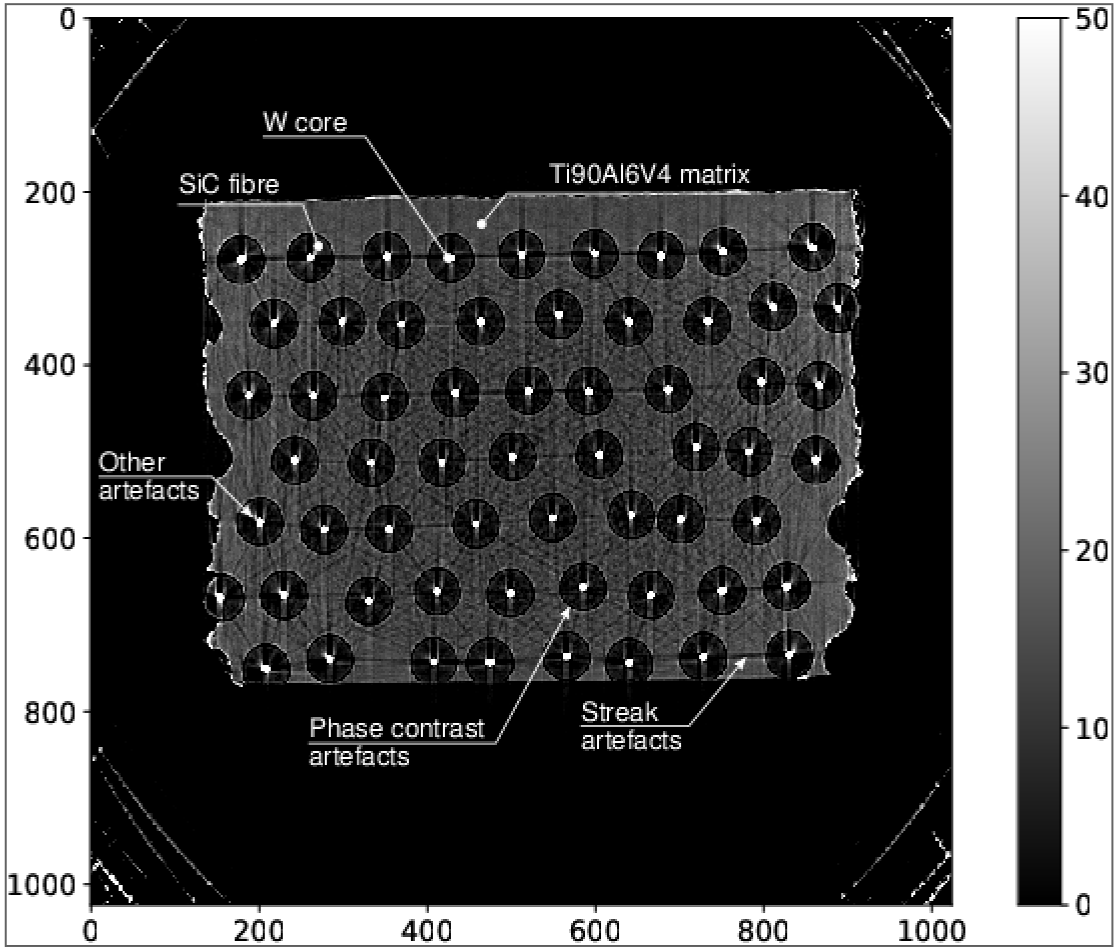
Scanned object

Photograph **Optical microscopy**

- Matrix: mixture composed of titanium (90%), aluminium (6%) and vanadium (4%).
- Fibres: silicon carbide, and
- Cores: tungsten.

Context: Artefacts in CT

- Data acquired at ESRF;
- multilayer monochromator used to make the beam spectrum **almost monochromatic**;
- Selected an energy of **33 keV**.



W core
SiC fibre
Ti90Al6V4 matrix
Other
artefacts
Phase contrast
artefacts
Streak
artefacts
0
200
400
600
800
1000
0
10
20
30
40
50

Linear attenuation coefficients (**µ**) in cm^{-1} from the literature (Theoretical), and from the CT slice of the experiment at ESRF (Experimental)

Structure	Material	Theoretical	Experimental	Error
Core	W	341.61	162.34±21.67	−179.27
Fibre	SiC	2.74	5.61±5.73	+2.87
Matrix	Ti90Al6V4	13.13	12.87±3.57	−0.26

- **W** µ are **underestimated** by a factor of **2**;
- **SiC** coefficients are **overestimated** by a factor of **2**;
- **Ti90Al6V4** coefficients are right.

Manual extraction of geometrical parameters

- Anonymous user-study;
- 13 participants measured
    - the size, position and orientation of the matrix, and
    - the size of the fibres,
    - the size of the cores.
- Maybe some of you responded ;-)

Ti90Al6V4 Matrix

What	Min	Median	Max	Average±stddev	[Max - Min]
Width (µm)	1478	1499	5149	1789±1011	3671
Height (µm)	1056	1068	1501	1113±123	445
Rotation (degrees)	55	91	95	88.5±10.1	40
Centre x-axis (µm)	988	999	1742	1057±206	754
Centre y-axis (µm)	238	921	963	870±191	726

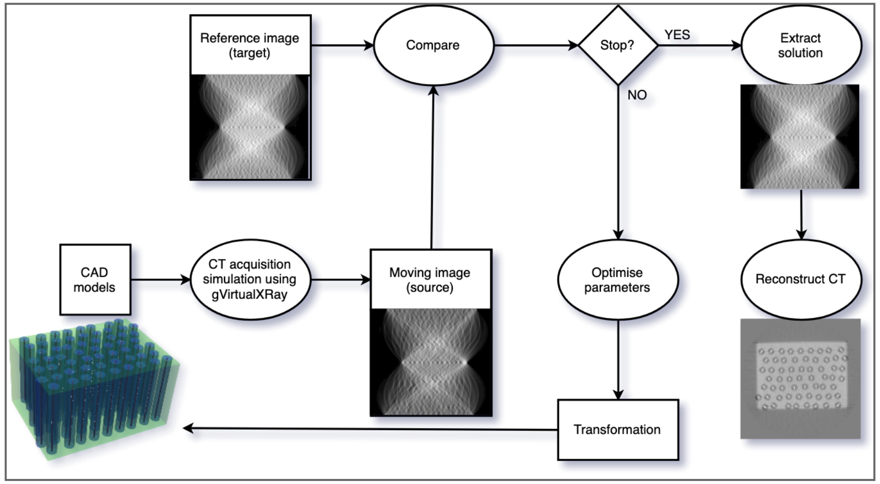
- **One outlier** (which is why the median value may be more relevant than the average);
- **Variability** amongst participants (cf stddev and [Max - Min]);
- *What is the truth?*

Fibres and cores

What	Min	Median	Max	Average±stddev	[Max - Min]
W core diameter (µm)	16.0	20.0	28.0	20.4±2.2	12.0
SiC fibre diameter (µm)	104.0	108.6	116.0	109.6±2.8	12.0

- No outlier; but
- **Still variability** amongst participants,
  - in particular for the **W core diameter**;
- *What is the truth?*

Methodology: Image registration as an optimisation algorithm

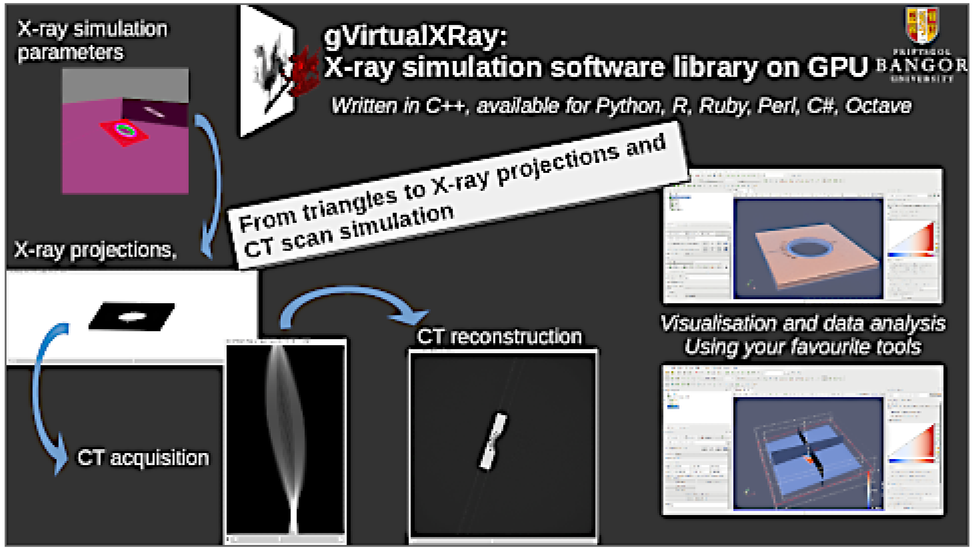


1. CAD models of the scanned object are generated
2. A CT acquisition is simulated to create X-ray projections from the CAD models
3. Simulated X-ray projections are compared with the projections from the real experiments

4. An optimisation algorithm tweaks the parameters of the simulation models (CAD & CT acquisition) until convergence

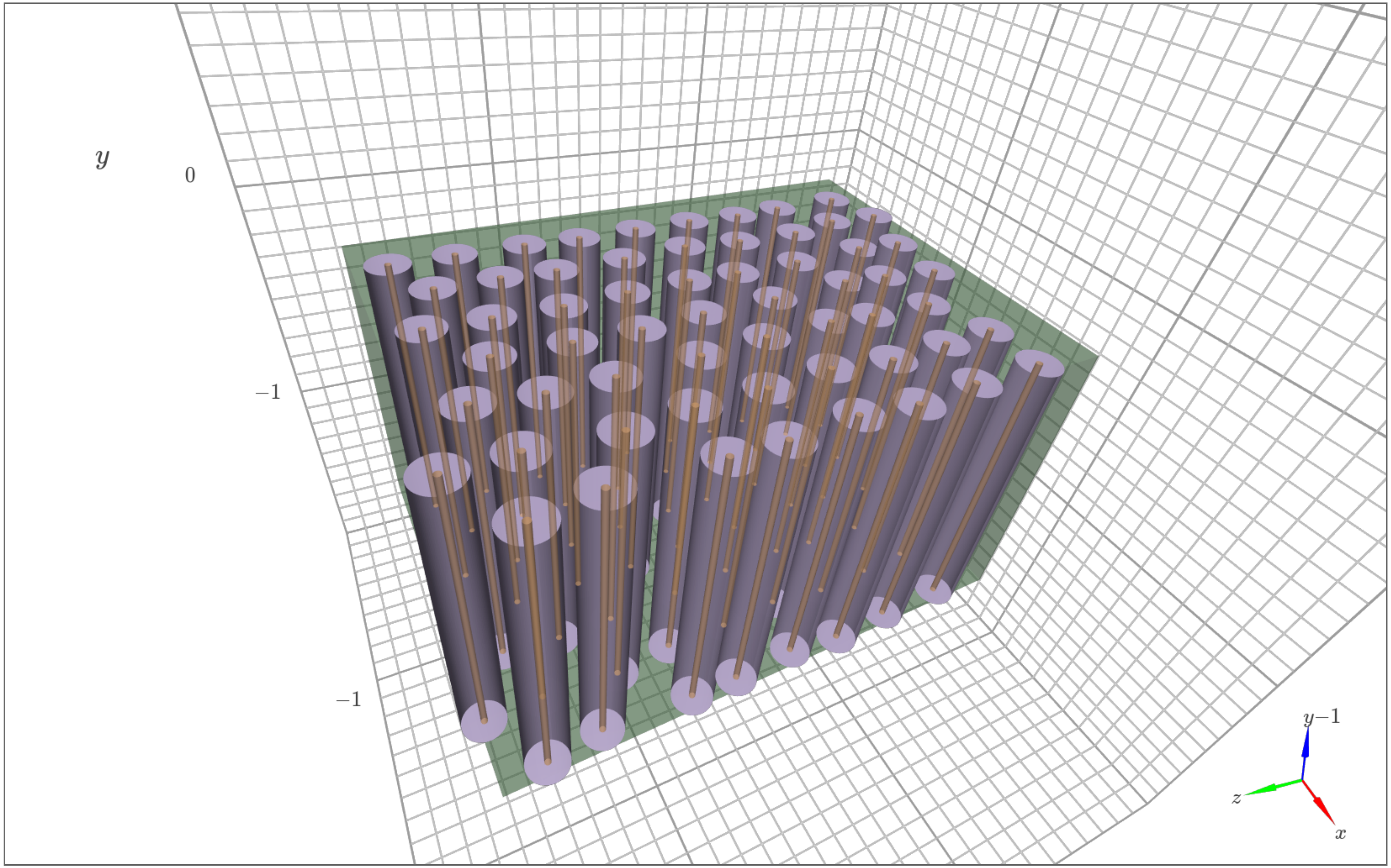
Requirements

- CT scan acquisition simulated in an objective function;
- Objective function repeated numerous times with different parameters;
- Trade off between speed and accuracy:
    - No Monte Carlo simulation;
    - Fast analytical simulation on GPU
    - From surface models.
- Open source software only (image processing, optimisation algorithm, X-ray simulation)
- Prototype in Python


X-ray simulation parameters
gVirtualXRay:
X-ray simulation software library on GPU
Written in C++, available for Python, R, Ruby, Perl, C#, Octave
PRIFYSGOL BANGOR UNIVERSITY
From triangles to X-ray projections and CT scan simulation
X-ray projections,
CT acquisition
CT reconstruction
Visualisation and data analysis
Using your favourite tools

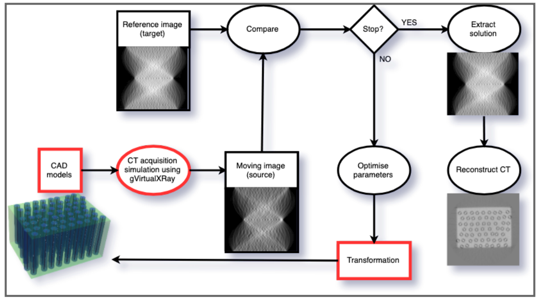
1. CAD models of the scanned object are generated

- A hollowed parallelepiped (matrix)
- Hollowed cylinders (fibres)
- Cylinders (cores)

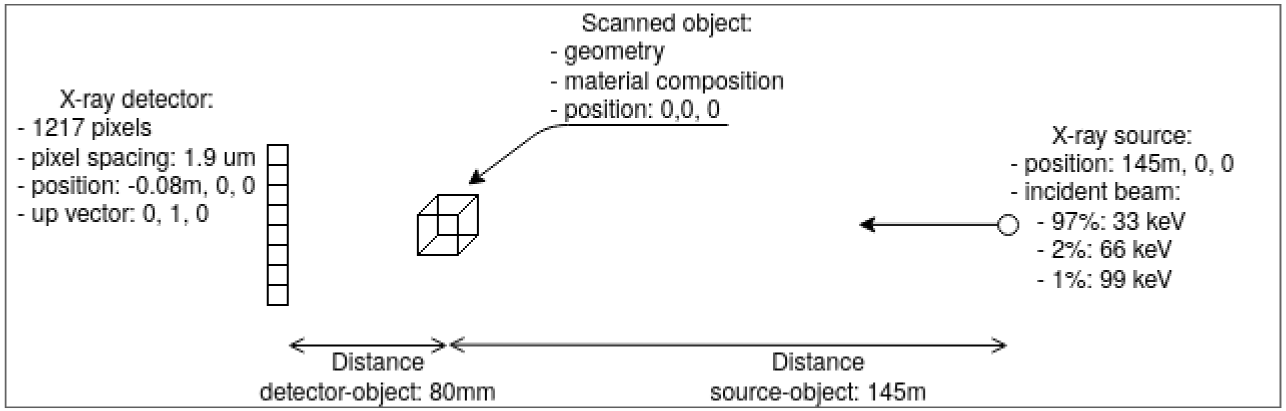


y
0
−1
−1
y−1
z
x

2. A CT acquisition is simulated to create X-ray projections from the CAD models



- Number of angles, angular span
- Geometric properties of the CT acquisition:



Raw projection modelled using the polychromatic version of the Beer-Lambert law:

$$I(x,y) = \sum_i R_i N_i \exp\left(-\sum_j \mu_j(E_i)\, d_j(x,y)\right)$$

- $I(x,y)$ the value of the raw X-ray projection at pixel location (x,y);
- i the i-th energy channel in the beam spectrum;
- E_i the energy in eV;
- R_i and N_i the detector response and the number of photons at that energy;
- j the j-th material being scanned, $\mu_j(E_i)$ its linear attenuation coefficient at energy E_i, and
- $d_j(x,y)$ path length in cm^{-1} of the ray crossing the j-th material from the X-ray source to pixel (x,y).

Flat-field correction is then applied on:

$$Proj = \frac{I - D}{F - D}$$

F (full fields) and D (dark fields) are projection images without sample and acquired with and without the X-ray beam turned on respectively.

Linearisation of the transmission data to get the sinogram:

$$\mathrm{Sino} = -\ln(\mathrm{Proj})$$

Beer-Lambert law enhanced with

- Polychromatism
- Phase
- Impulse response of detector
- Poisson noise

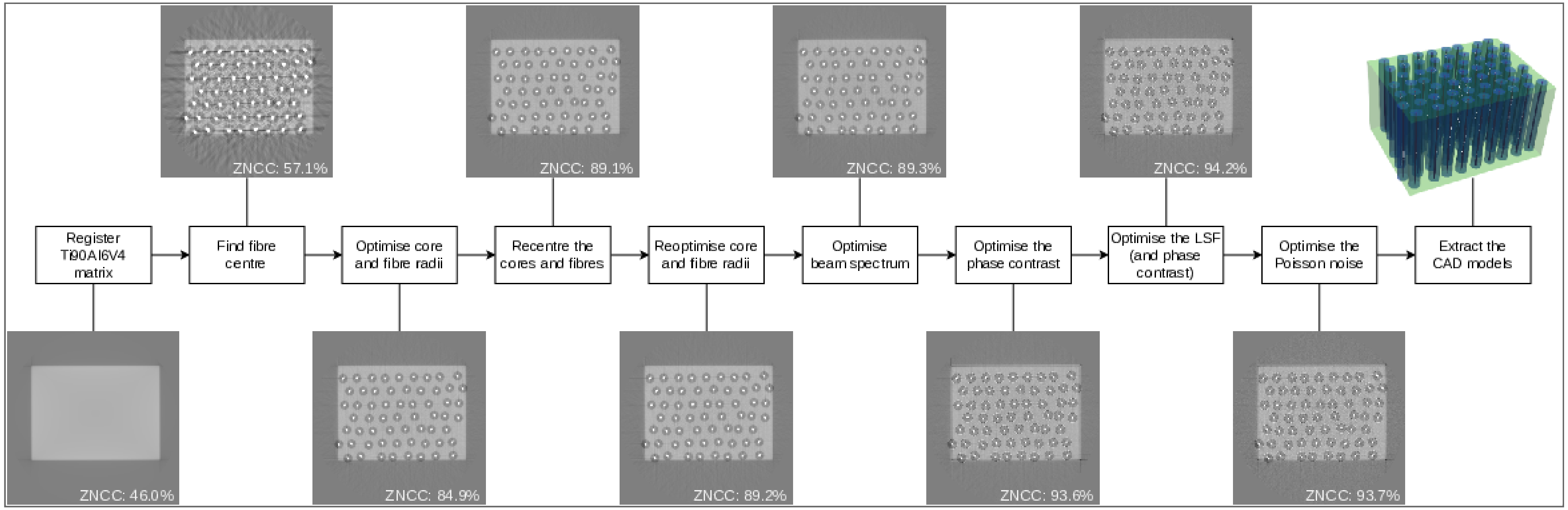
Summary of the parameters that need to be optimised

#	Parameter
1	Position of the matrix along the primary axis (in µm)
2	Position of the matrix along the secondary axis (in µm)
3	Size of the matrix along the primary axis (in µm)
4	Size of the matrix along the secondary axis (in µm)
5	Rotation angle of the matrix (in degrees)
6	Radius of the cores (in µm)
7	Radius of the fibres (in µm)
8	Percentage of 33 keV photons in the beam spectrum
9	Percentage of 66 keV photons in the beam spectrum
10	Percentage of 99 keV photons in the beam spectrum
11	Bias controlling the Poisson noise

#	Parameter
12	Gain controlling the Poisson noise
13	Intensity of the Poisson noise
14	Intensity of the phase contrast for the tungsten core
15	Spread of the phase contrast for the tungsten core
16	Intensity of the phase contrast for the SiC fibres
17	Spread of the phase contrast for the SiC fibres
18	Intensity of the phase contrast for the Ti90Al6V4 matrix
19	Spread of the phase contrast for the Ti90Al6V4 matrix
20-23	Parameters of the LSF

Registration pipeline: Divide and Conquer

- There are too many parameters to optimise everything in one go
- Breaks down the registration problem into sub-problems of the same type
- They become simple enough to be solved directly

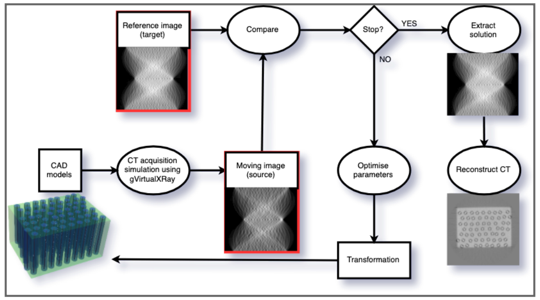


3. Simulated X-ray images are compared with the images from the real experiments

Must choose the objective function carefully:

- which images to compare?
- do we pre-process them and how?
- how to compare them?

Which images to compare?



Candidate	image	Comment
Projections (Projs) after flat-field correction	·	Possible candidate
Sinogram, i.e. $\mathrm{Sino} = -\ln(\mathrm{Projs})$	·	Possible candidate
Reconstructed CT slice	·	Discarded as it involves the CT reconstruction

Do we pre-process them and how?

Do nothing

- It should work in theory

Min-max normalisation

$$m_o = \frac{m - \min(m)}{\max(m) - \min(m)}$$

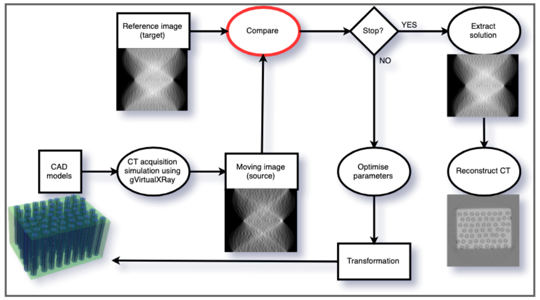
- where m_o is the image after normalisation of Image m
- after normalisation, $\min(m_o) = 0$ and $\max(m_o) = 1$
- discarded as too sensitive to Poisson noise

Zero-mean, unit-variance normalisation (also known as standardisation or Z-score normalisation in machine learning)

$$m_o = \frac{m - \bar{m}}{\sigma_m}$$

- where $\bar{m}$ is the average pixel value of Image m, and σ_m its standard deviation.
- after normalisation, $\bar{m}_o = 0$ and $\sigma_{m_o} = 1$
- why not? It's popular in computer vision and machine learning after all

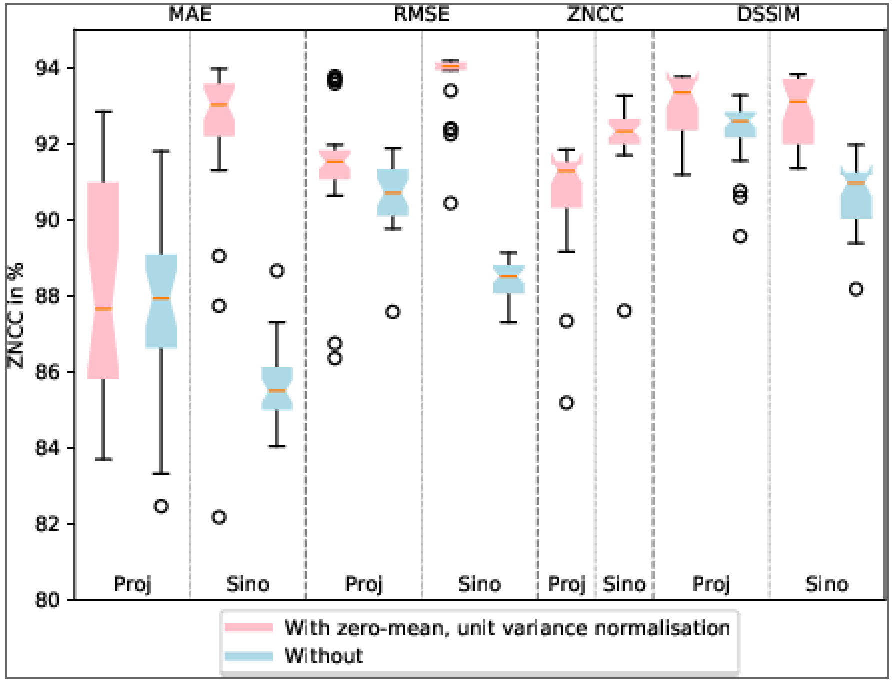
Which image comparison method?



- MAE (super fast, robust to outliers),
- RMSE (quite fast, give a higher weight to large discrepencies (outliers) than MAE),
- SSIM (popular in computer vision),
- ZNCC (popular in computer vision).

Exaustive evaluation on a Supercomputer

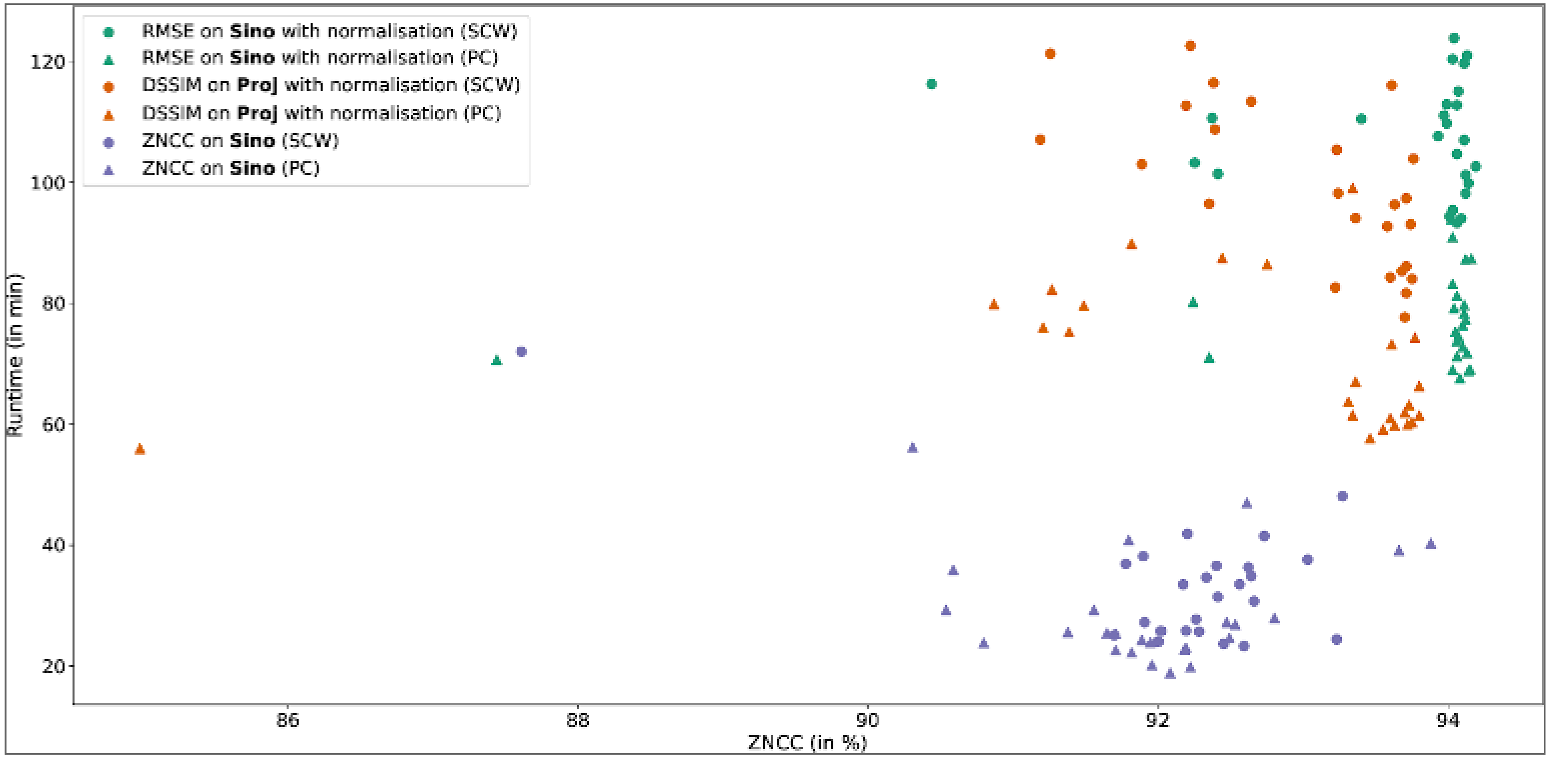
- 14 objective functions in total: every possible combination (image, image metrics, with/without normalisation).
- Run the registration on each objective functions 25 times to gather statistically meaninful data.
- 350 registrations in total (23 days).
- A registration is 1 hour 35 minute (average)
- Best objective: RMSE on **Sino** with zero-mean, unit-variance normalisation



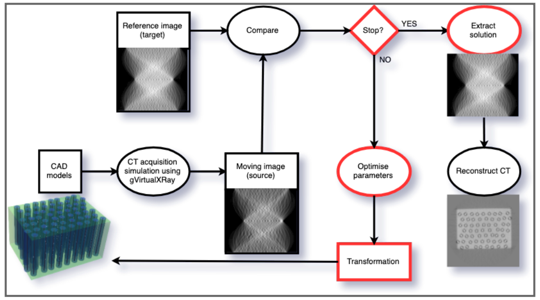
MAE
RMSE
ZNCC
DSSIM
ZNCC in %
94
92
90
88
86
84
82
80
Proj
Sino
Proj
Sino
Proj
Sino
Proj
Sino
With zero-mean, unit variance normalisation
Without

Comparison of the 3 best strategies

- Select the objective functions that works best (in terms of accuracy) on the supercomputer
- Perfrom another 3×25 registrations on a desktop PC
    - i.e. 25 per objective function
    - compare the performance between supercomputer and my office PC


RMSE on **Sino** with normalisation (SCW)
RMSE on **Sino** with normalisation (PC)
DSSIM on **Proj** with normalisation (SCW)
DSSIM on **Proj** with normalisation (PC)
ZNCC on **Sino** (SCW)
ZNCC on **Sino** (PC)
Runtime (in min)
ZNCC (in %)
20
40
60
80
100
120
86
88
90
92
94

4. An optimisation algorithm tweaks the parameters of the simulation models (CAD & CT acquisition) until convergence



- Use one of today's most popular global optimisation algorithm: covariance matrix adaptation evolution strategy (CMA-ES);
- Evolutionary algorithm designed for difficult non-linear non-convex optimisation problems in continuous domain;
- Considered as state-of-the-art in evolutionary computation;
- Does not require a tedious parameter tuning:

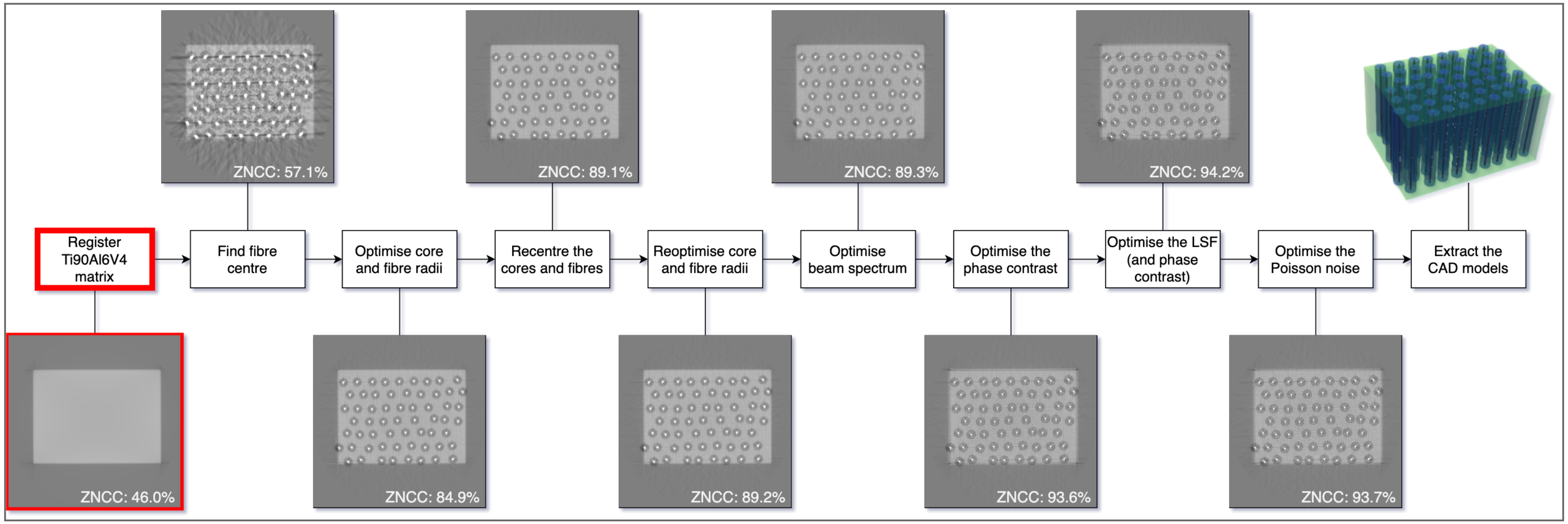
    - finding suitable internal parameters is part of the algorithm design;
- Only an initial solution and an initial standard deviation must be set by the user.
- The default population size is relatively small for fast convergence.

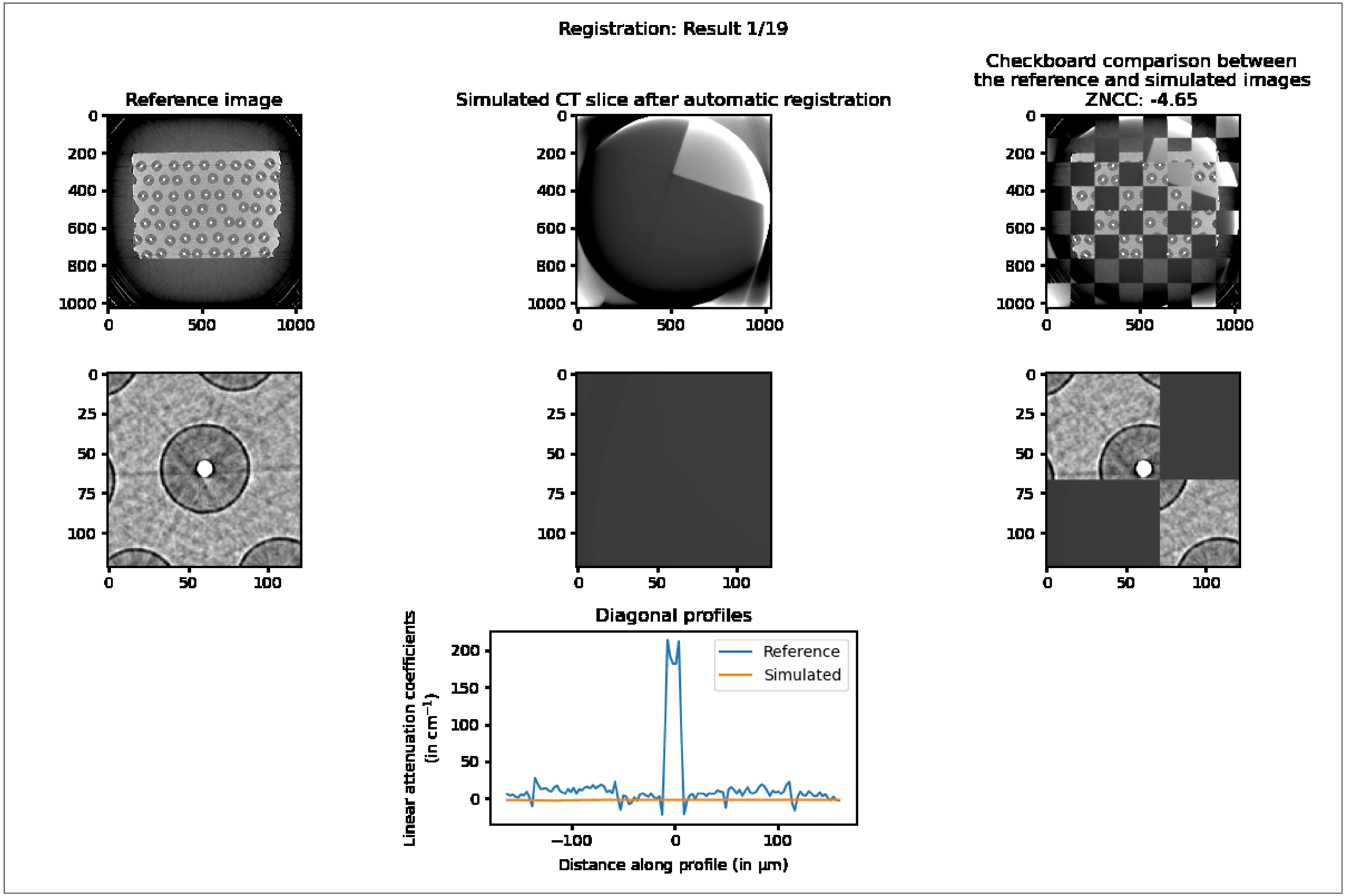
Simulate the CT acquisition

There are 7 successive steps to simulate the XCT data acquisition:

1. Set the cores and fibre geometries and material properties
2. Set the matrix geometry and material properties
3. Simulate the raw projections for each angle:
    - Without phase contrast, or
    - With phase contrast
4. Apply the LSF
5. Apply the flat-field correction
6. Add Poison noise
7. Apply the minus log linearisation to compute the sinogram

Register the Ti90Al6V4 matrix geometry

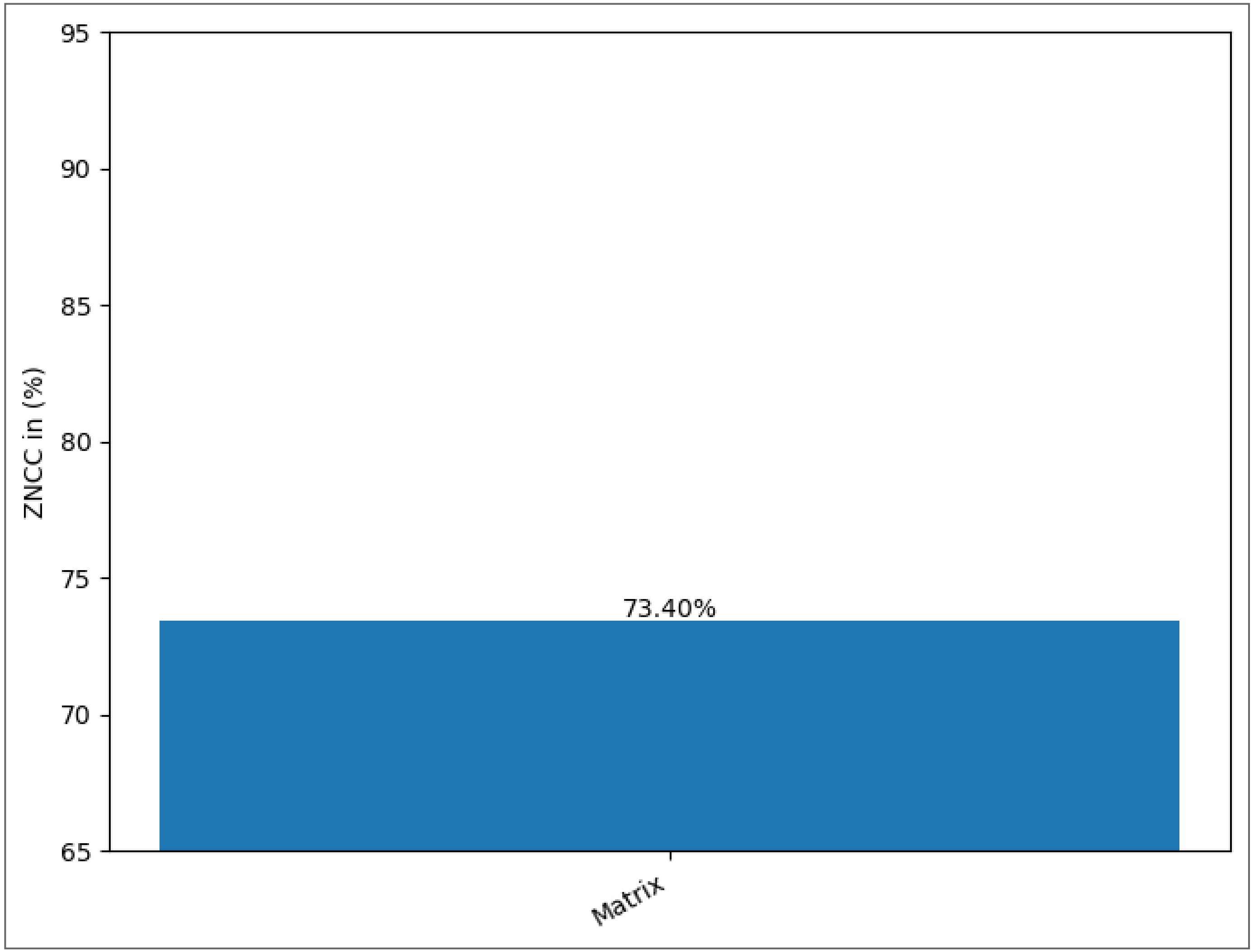




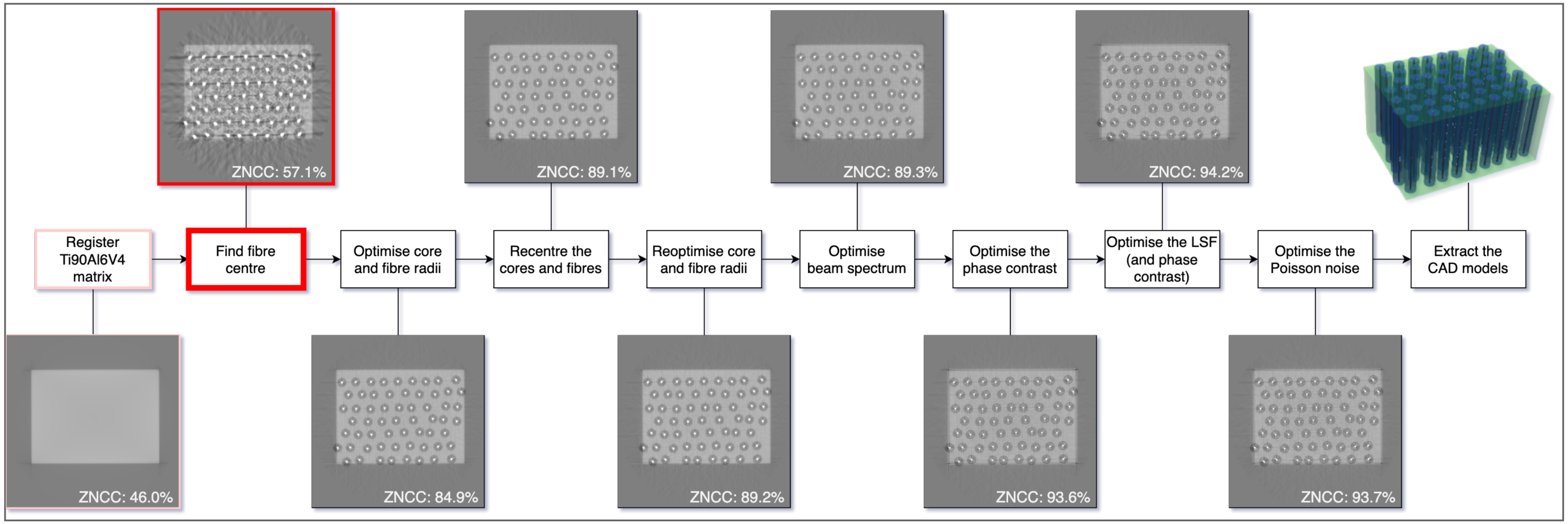
Registration: Result 1/19
Reference image
Simulated CT slice after automatic registration
Checkboard comparison between the reference and simulated images
ZNCC: -4.65
Diagonal profiles
Reference
Simulated
Linear attenuation coefficients (in cm^{-1})
Distance along profile (in μm)

In [39]:

```
plotSetOfZNCC(ZNCC_set, ZNCC_label_set)
```

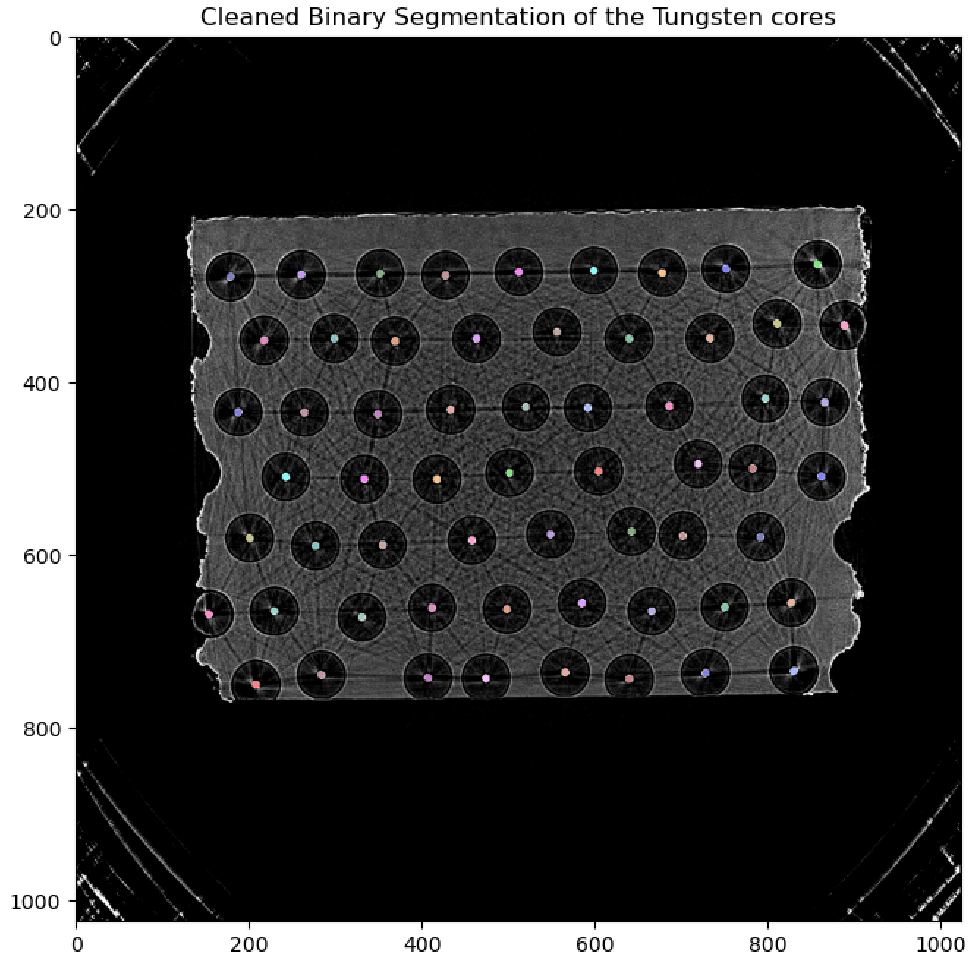


Find the centre of fibres

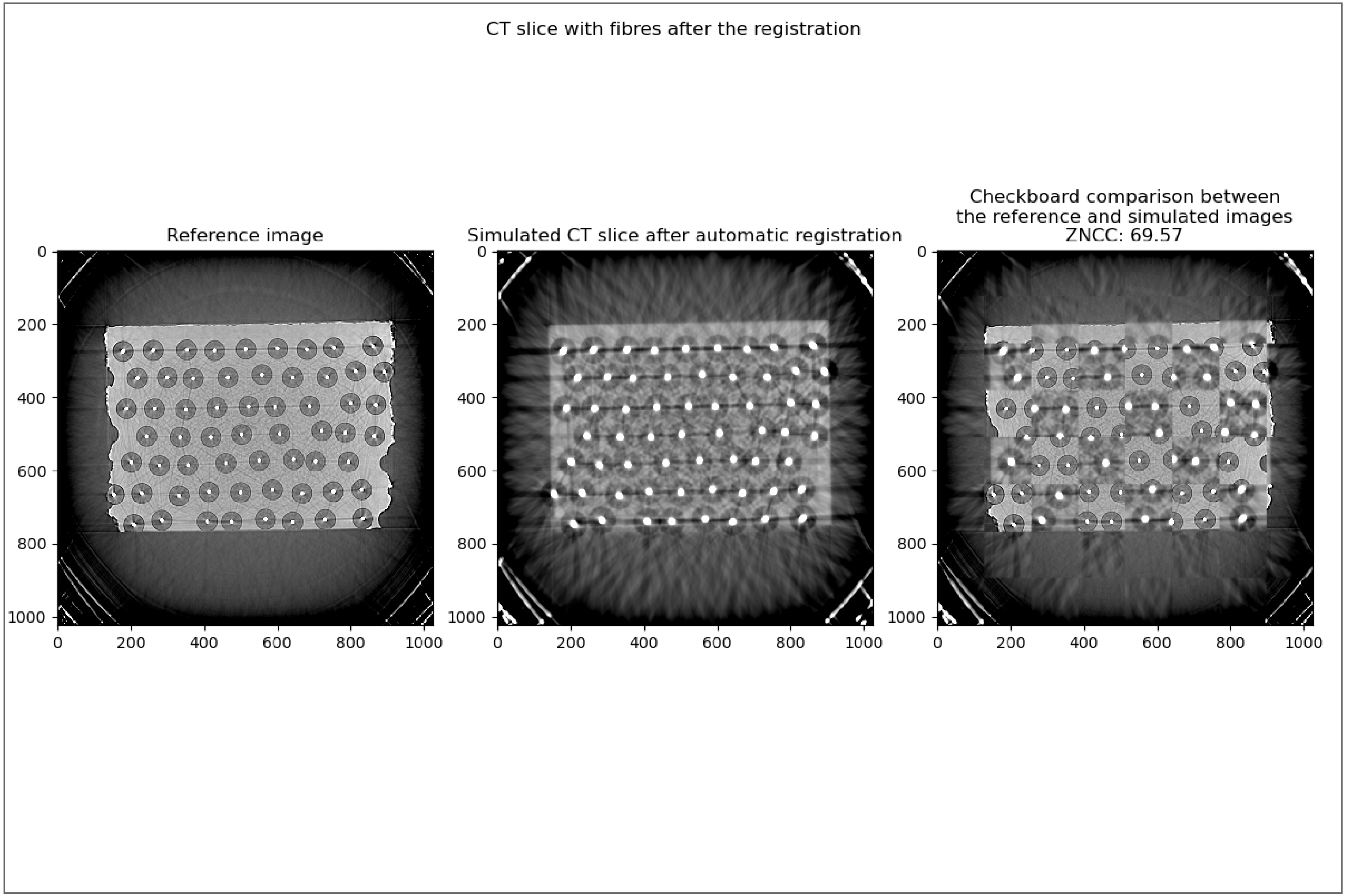


Find circles to identify the centre of fibres

We can use the Hoguh transform to detect where circles are in the image. However, the input image in OpenCV's function must be in UINT. We blur it using a bilateral filter (an edge-preserving smoothing filter).

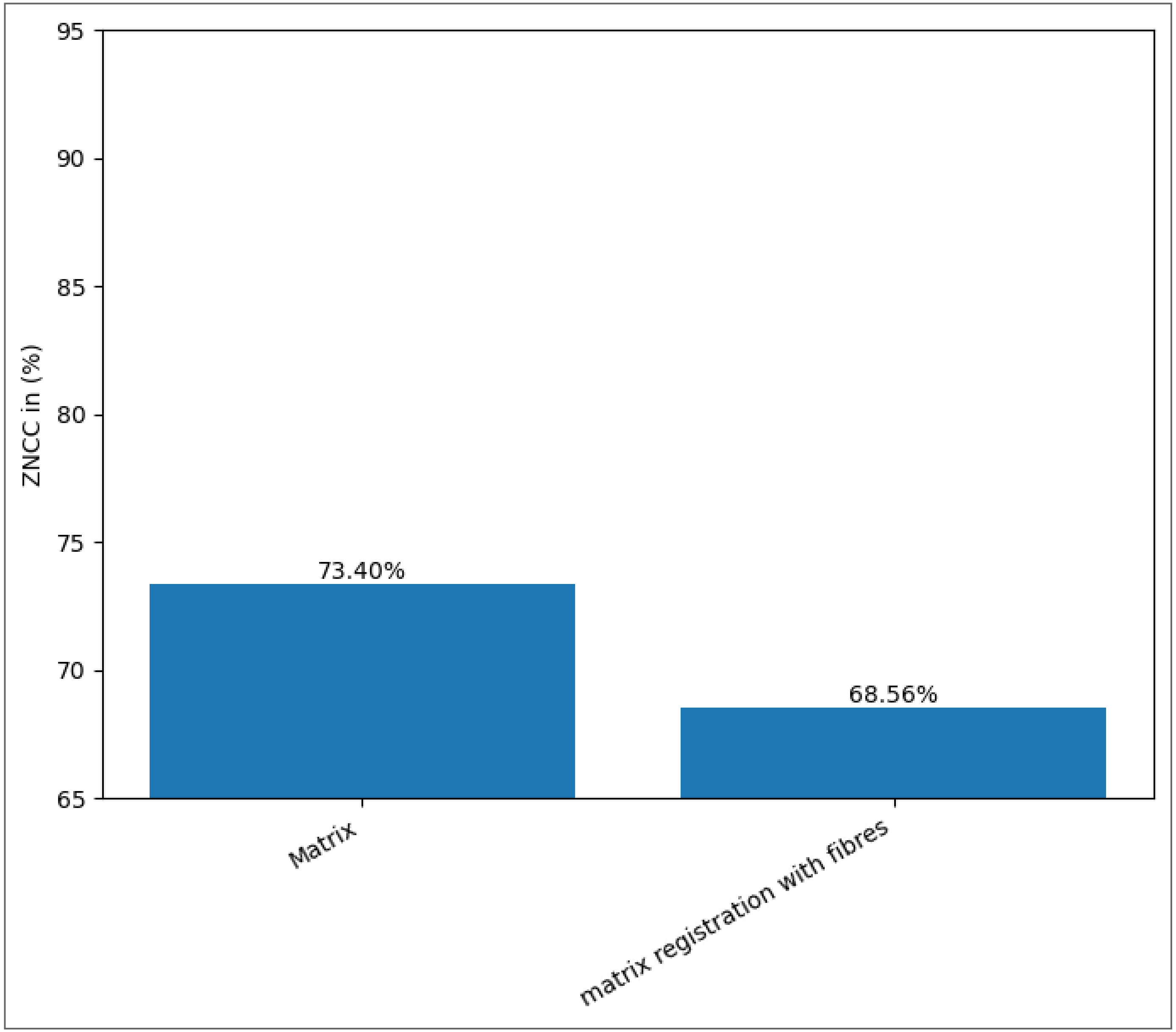


Cleaned Binary Segmentation of the Tungsten cores
0
200
400
600
800
1000
0
200
400
600
800
1000

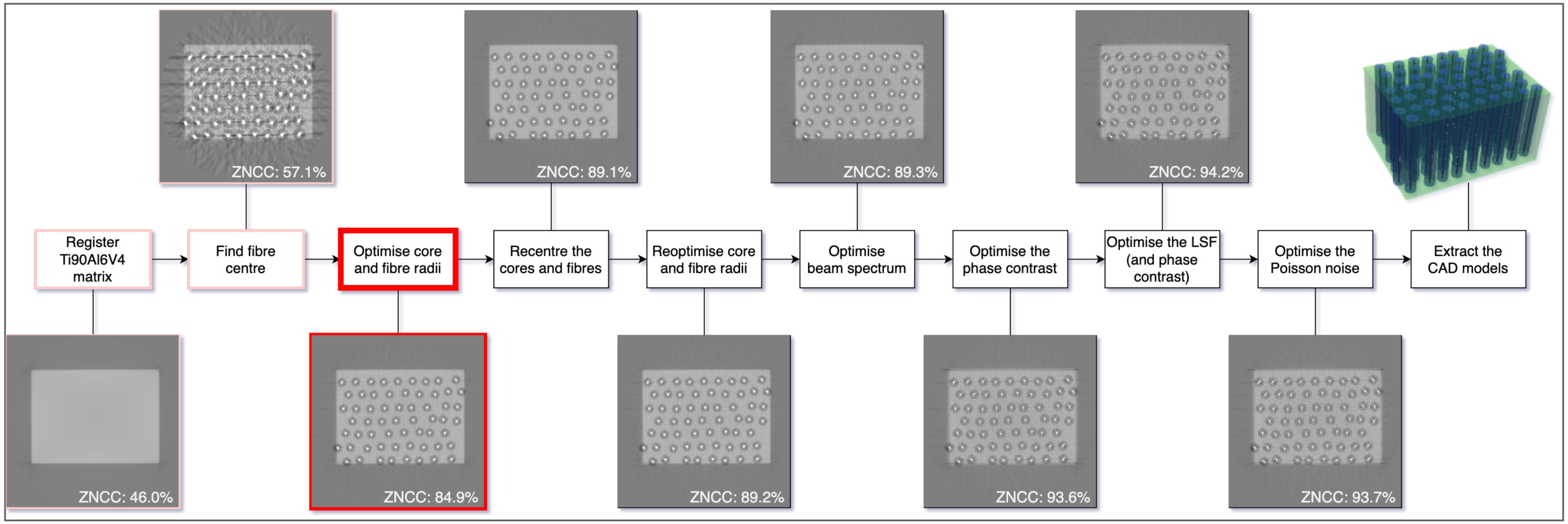

CT slice with fibres after the registration
Reference image
Simulated CT slice after automatic registration
Checkboard comparison between
the reference and simulated images
ZNCC: 69.57
0
200
400
600
800
1000

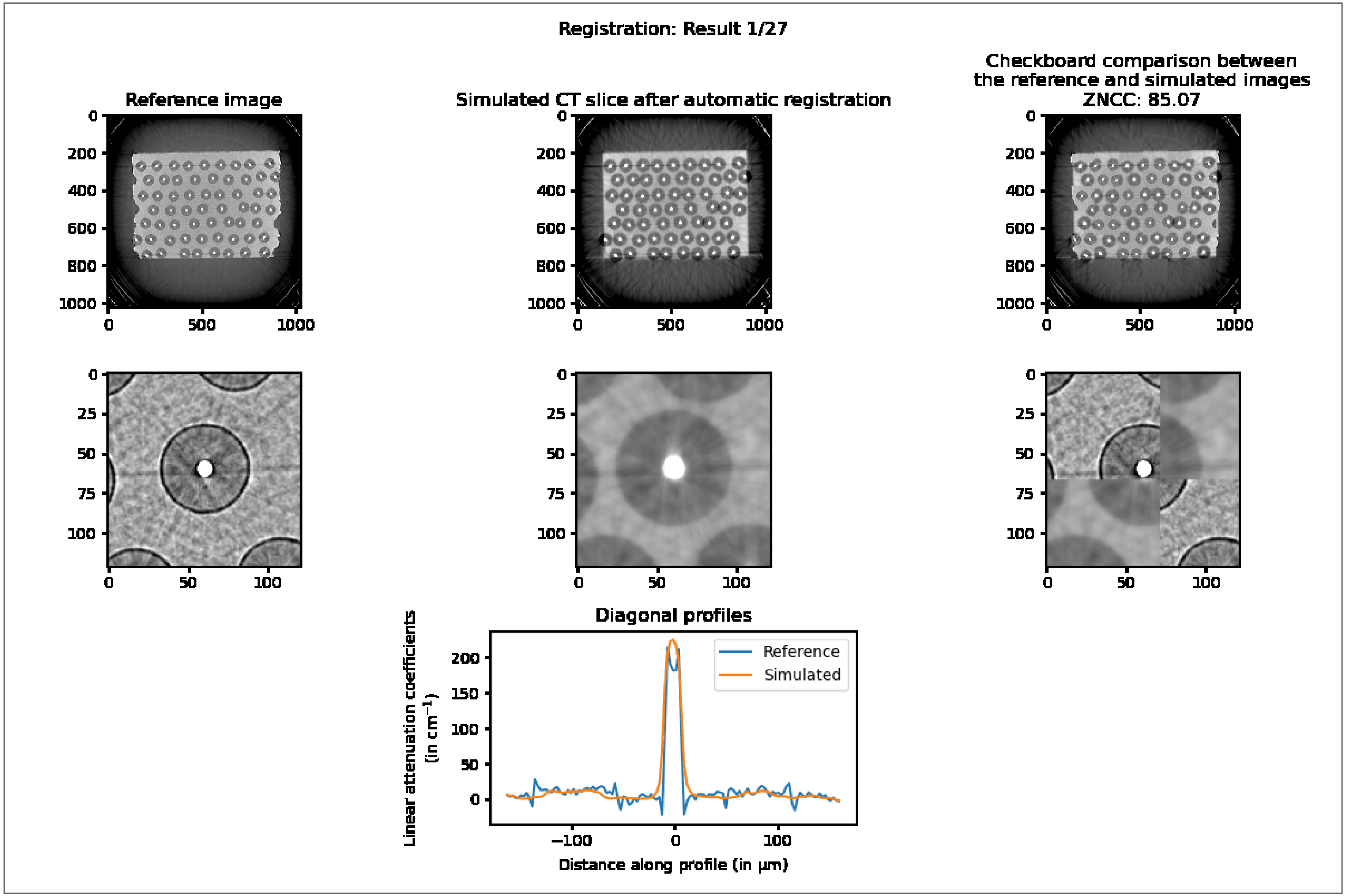
In [61]:

```
plotSetOfZNCC(ZNCC_set, ZNCC_label_set)
```


95
90
85
80
75
70
65
ZNCC in (%)
73.40%
68.56%
Matrix
matrix registration with fibres

Optmise core and fibre radii




Registration: Result 1/27
Reference image
Simulated CT slice after automatic registration
Checkboard comparison between
the reference and simulated images
ZNCC: 85.07
Diagonal profiles
Reference
Simulated
Linear attenuation coefficients
(in cm^{-1})
Distance along profile (in μm)

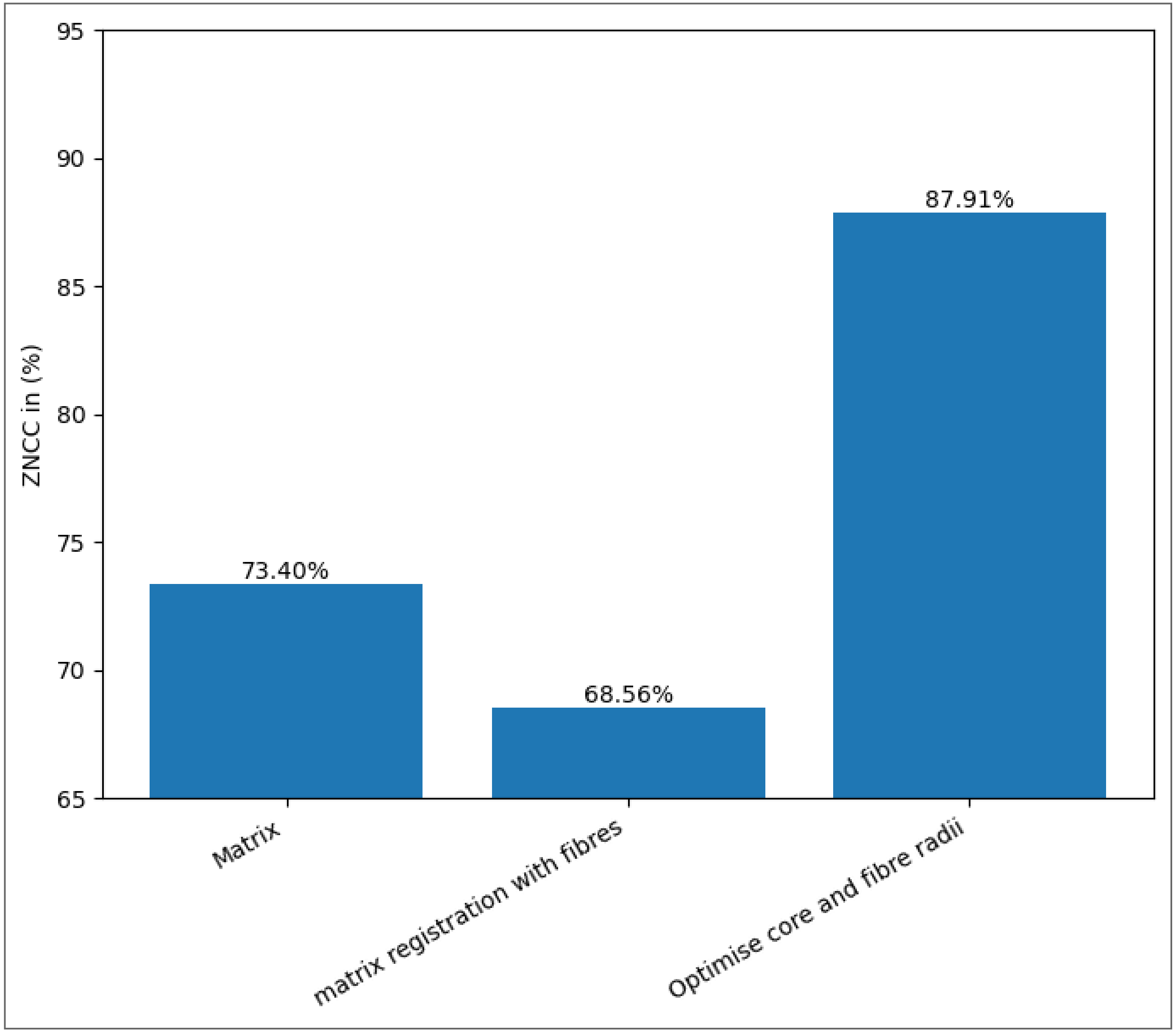
In [66]:

```
print("Core diameter:", round(core_radius * 2), "um");
print("Fibre diameter:", round(fibre_radius * 2), "um");
```

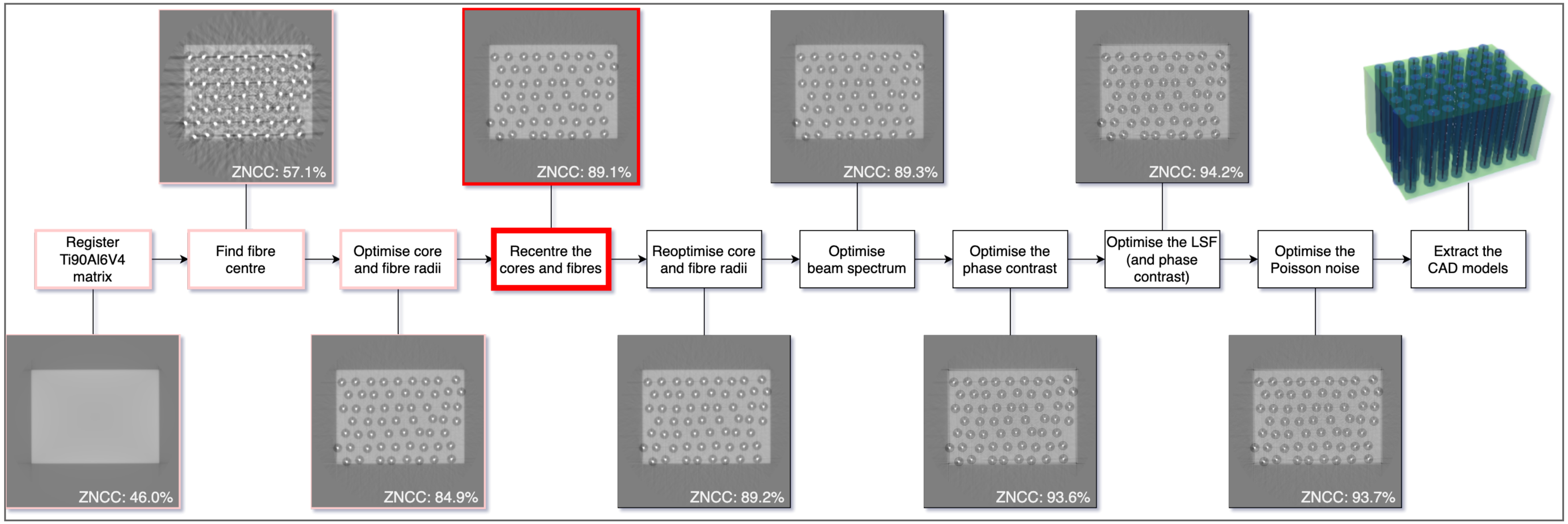
```
Core diameter: 16 um
Fibre diameter: 101 um
```

In [67]:

```
plotSetOfZNCC(ZNCC_set, ZNCC_label_set)
```

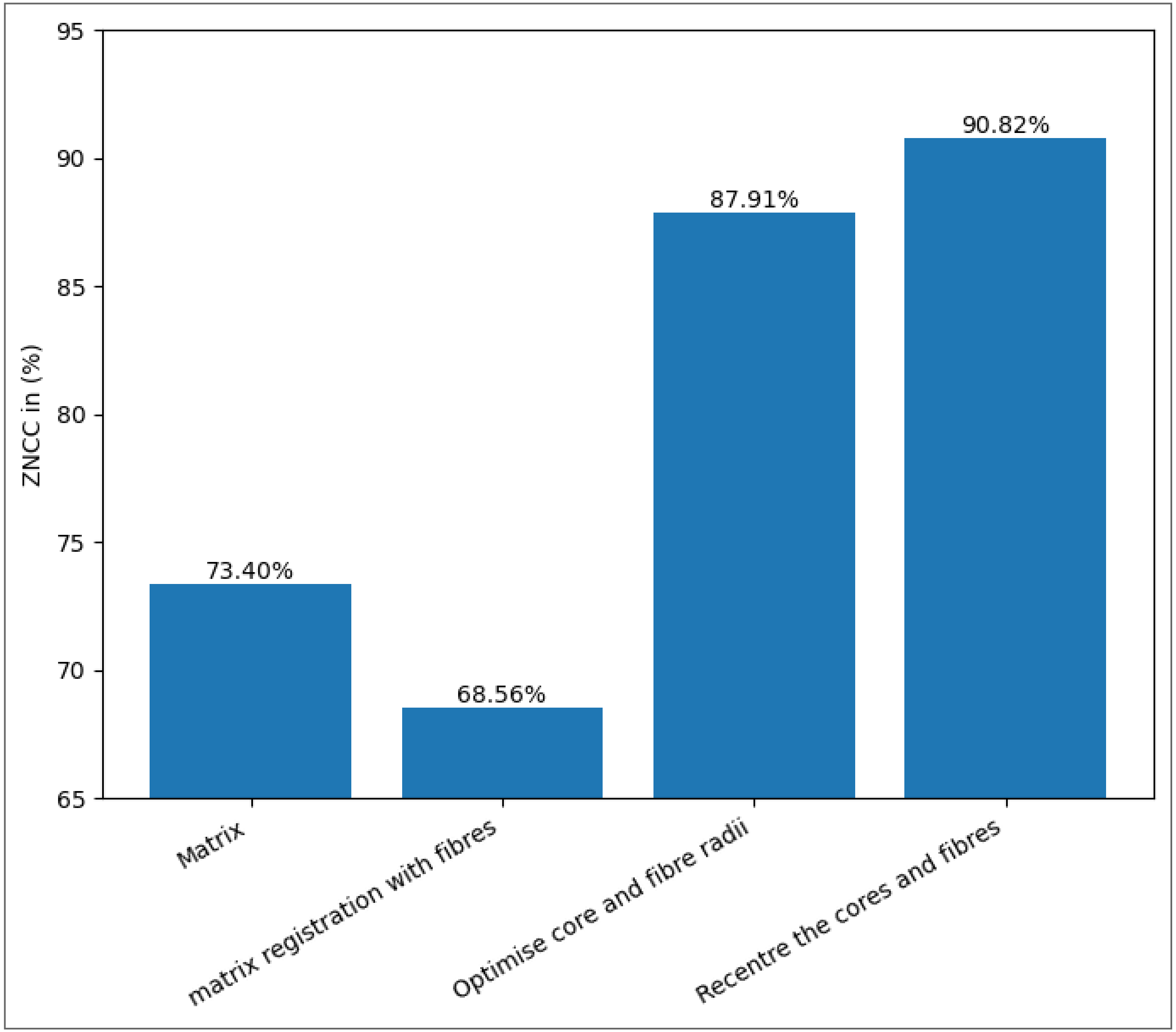

95
90
85
80
75
70
65
ZNCC in (%)
73.40%
68.56%
87.91%
Matrix
matrix registration with fibres
Optimise core and fibre radii

Recentre the cores and fibres

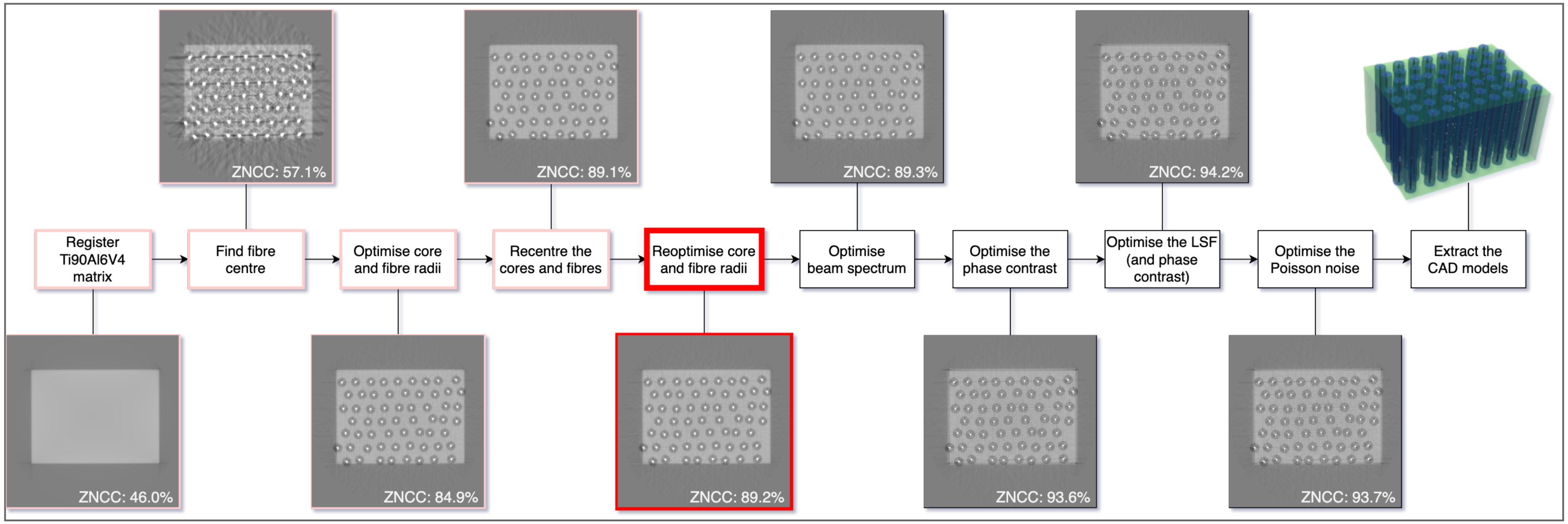


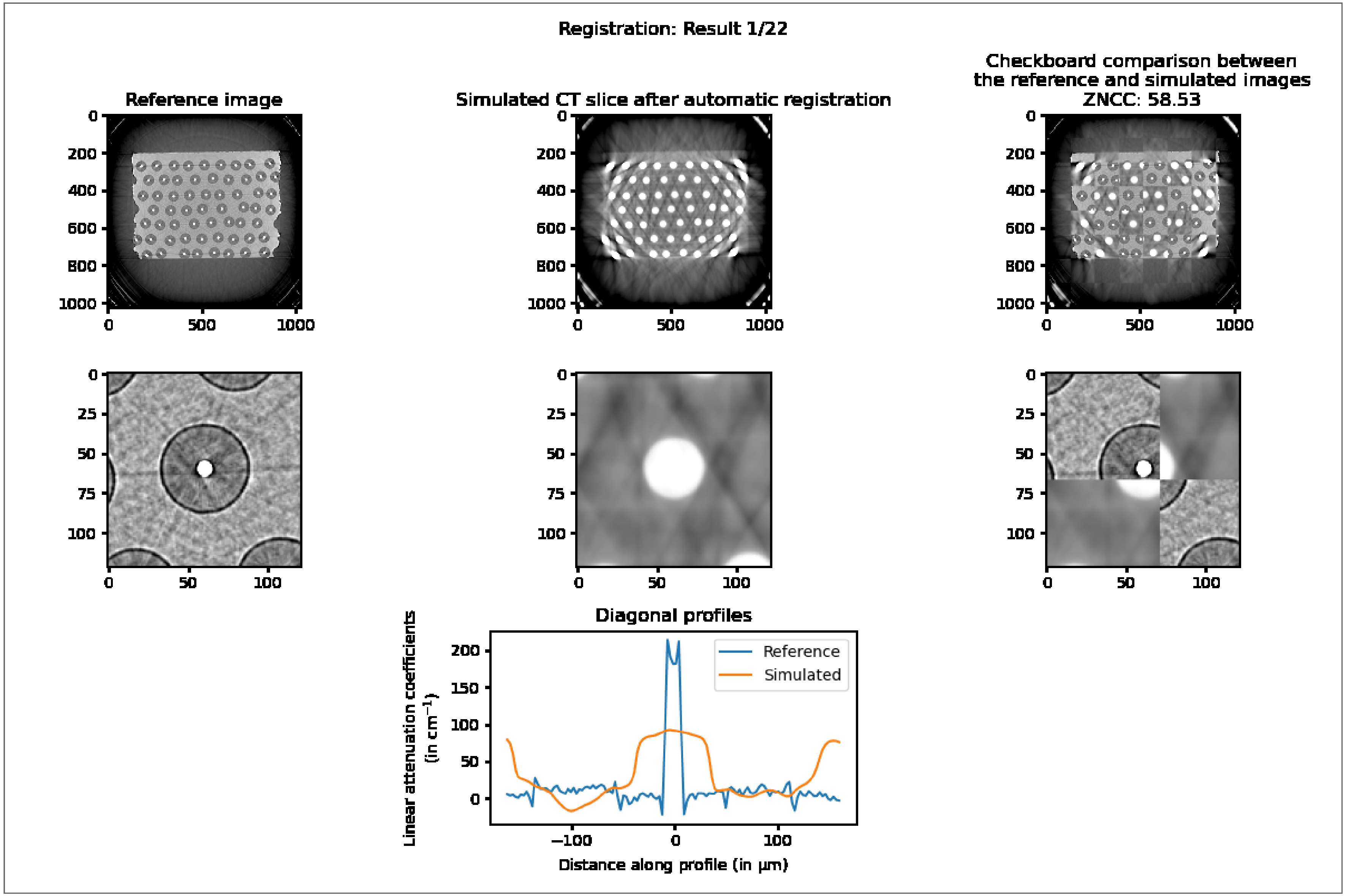
In [71]:

```
plotSetOfZNCC(ZNCC_set, ZNCC_label_set)
```


95
90
85
80
75
70
65
ZNCC in (%)
73.40%
68.56%
87.91%
90.82%
Matrix
matrix registration with fibres
Optimise core and fibre radii
Recentre the cores and fibres

Reoptmise core and fibre radii





Registration: Result 1/22
Reference image
Simulated CT slice after automatic registration
Checkboard comparison between
the reference and simulated images
ZNCC: 58.53
Diagonal profiles
Reference
Simulated
Linear attenuation coefficients
(in cm^{-1})
Distance along profile (in μm)

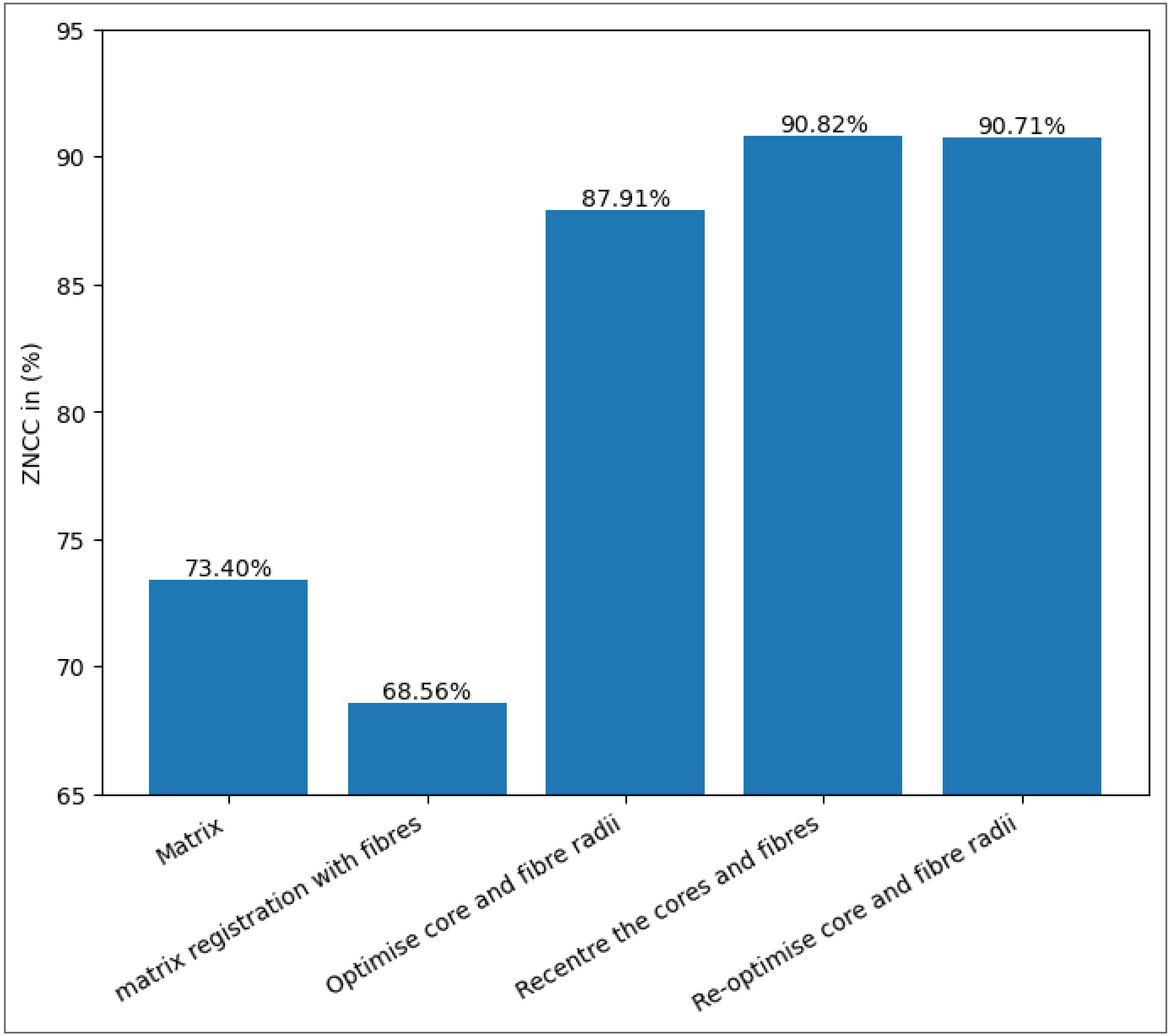
In [75]:

```
print("Core diameter:", round(core_radius * 2), "um");
print("Fibre diameter:", round(fibre_radius * 2), "um");
```

```
Core diameter: 16 um
Fibre diameter: 99 um
```

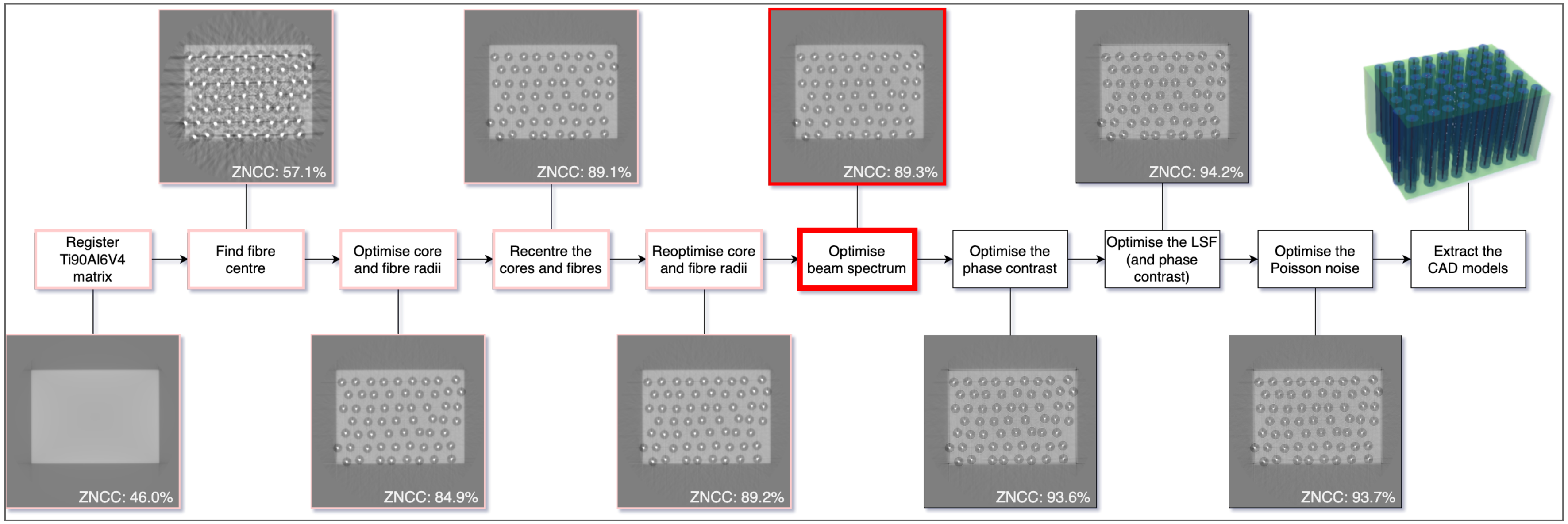
In [77]:

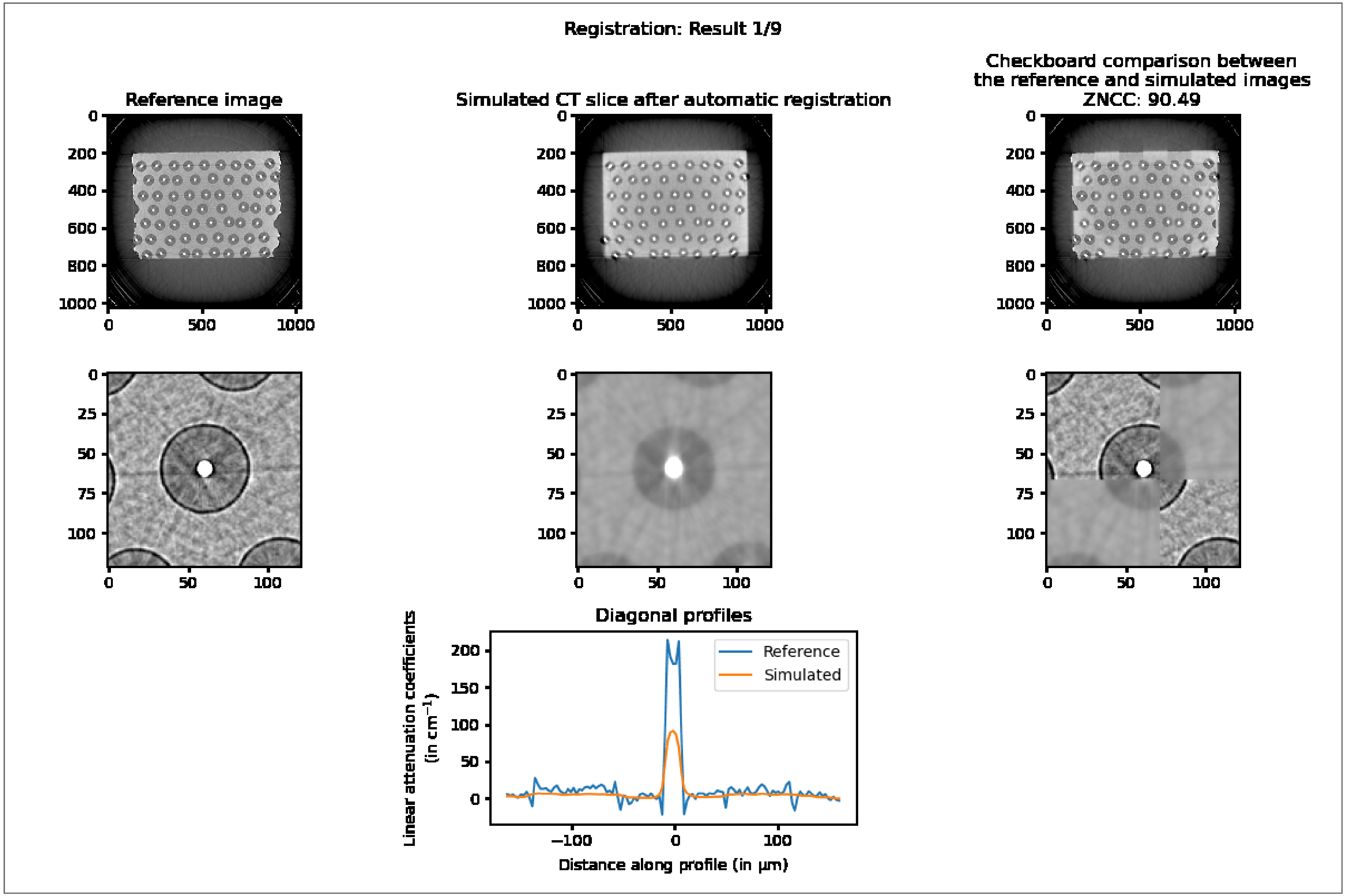
```
plotSetOfZNCC(ZNCC_set, ZNCC_label_set)
```



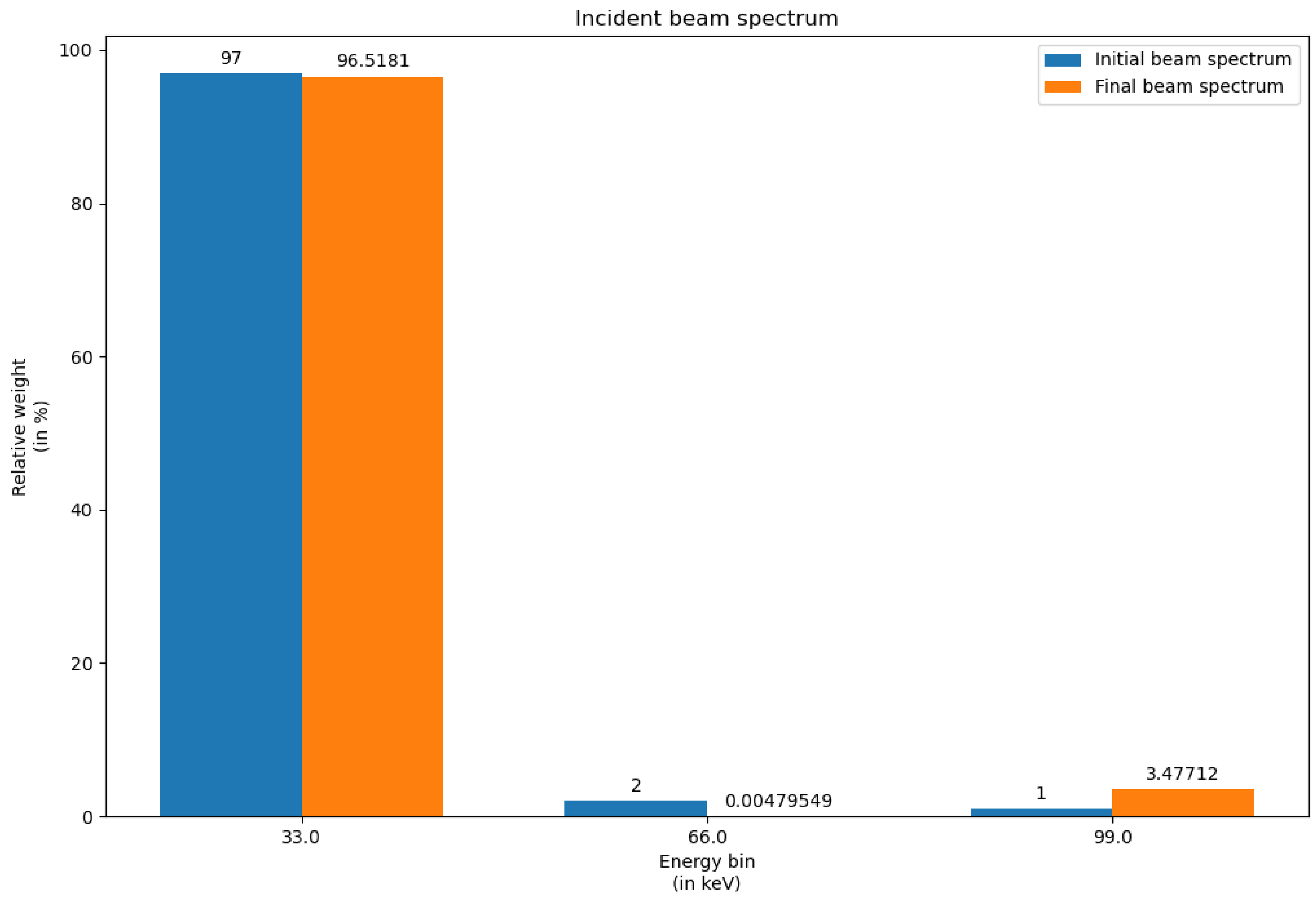
ZNCC in (%)
95
90
85
80
75
70
65
73.40%
68.56%
87.91%
90.82%
90.71%
Matrix
matrix registration with fibres
Optimise core and fibre radii
Recentre the cores and fibres
Re-optimise core and fibre radii

Optmise beam spectrum





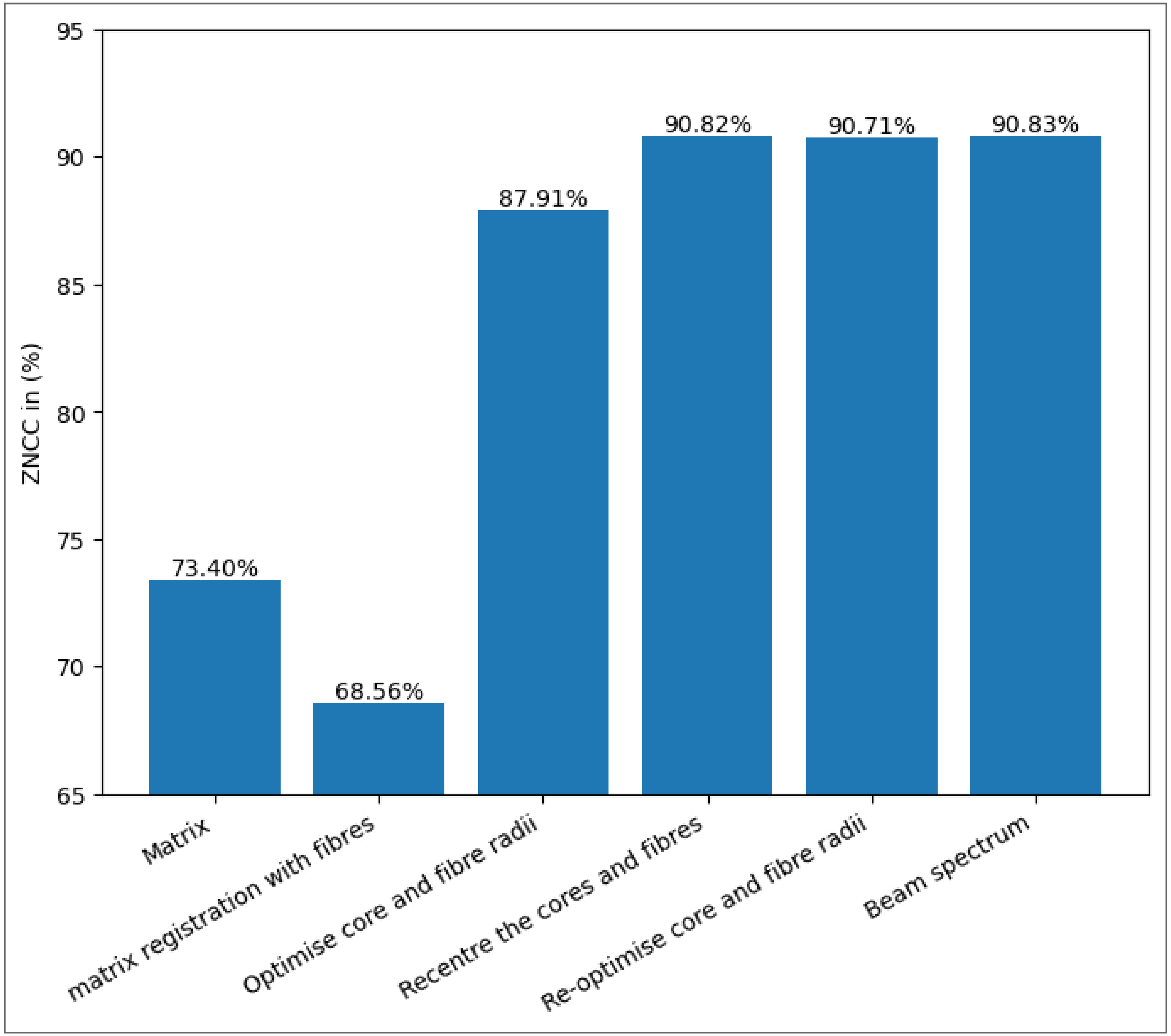
Registration: Result 1/9
Reference image
Simulated CT slice after automatic registration
Checkboard comparison between the reference and simulated images ZNCC: 90.49
Diagonal profiles
Reference
Simulated
Linear attenuation coefficients (in cm^{-1})
Distance along profile (in μm)



Incident beam spectrum
Initial beam spectrum
Final beam spectrum
97
96.5181
2
0.00479549
1
3.47712
100
80
60
40
20
0
Relative weight
(in %)
33.0
66.0
99.0
Energy bin
(in keV)

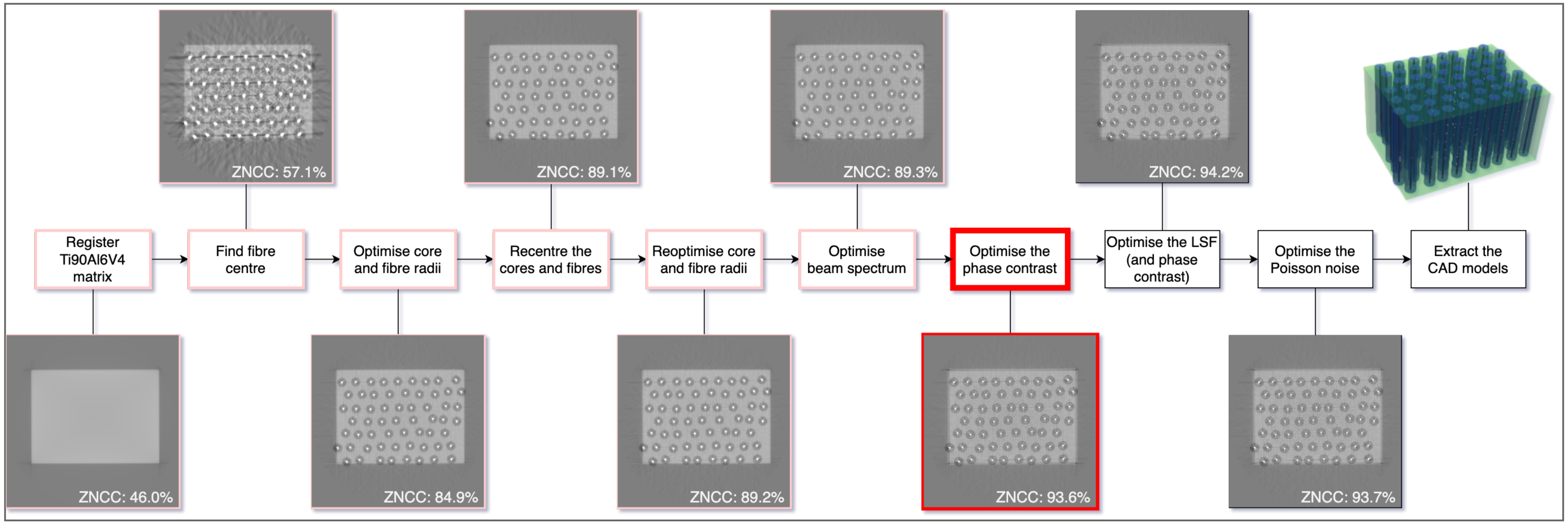
In [85]:

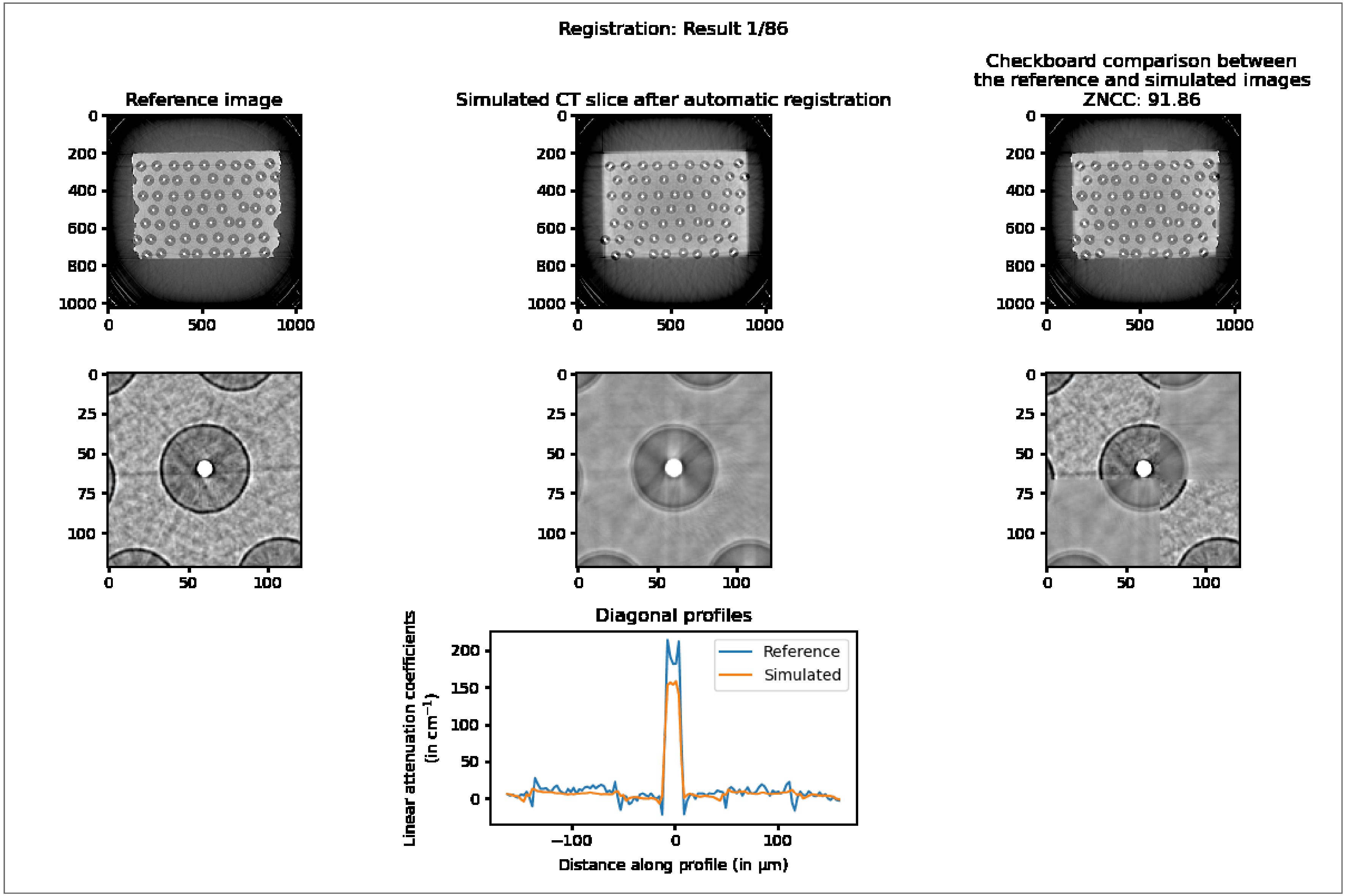
```
plotSetOfZNCC(ZNCC_set, ZNCC_label_set)
```



95
90
85
80
75
70
65
ZNCC in (%)
73.40%
68.56%
87.91%
90.82%
90.71%
90.83%
Matrix
matrix registration with fibres
Optimise core and fibre radii
Recentre the cores and fibres
Re-optimise core and fibre radii
Beam spectrum

Optimisation of the phase contrast and the radii




Registration: Result 1/86
Reference image
Simulated CT slice after automatic registration
Checkboard comparison between
the reference and simulated images
ZNCC: 91.86
Diagonal profiles
Reference
Simulated
Linear attenuation coefficients
(in cm^{-1})
Distance along profile (in μm)

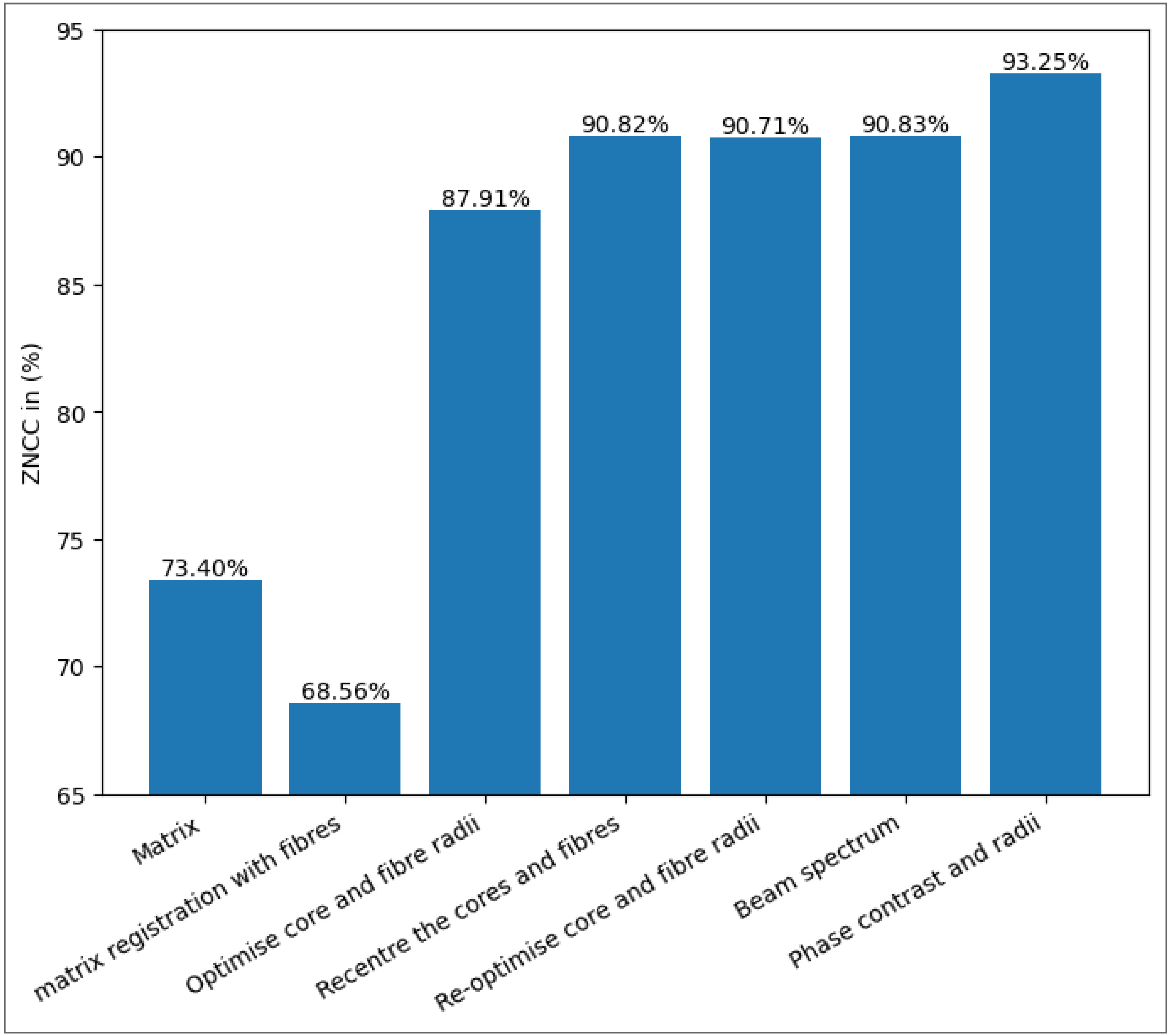
In [93]:

```
print("Core diameter:", round(core_radius * 2), "um");
print("Fibre diameter:", round(fibre_radius * 2), "um");
```

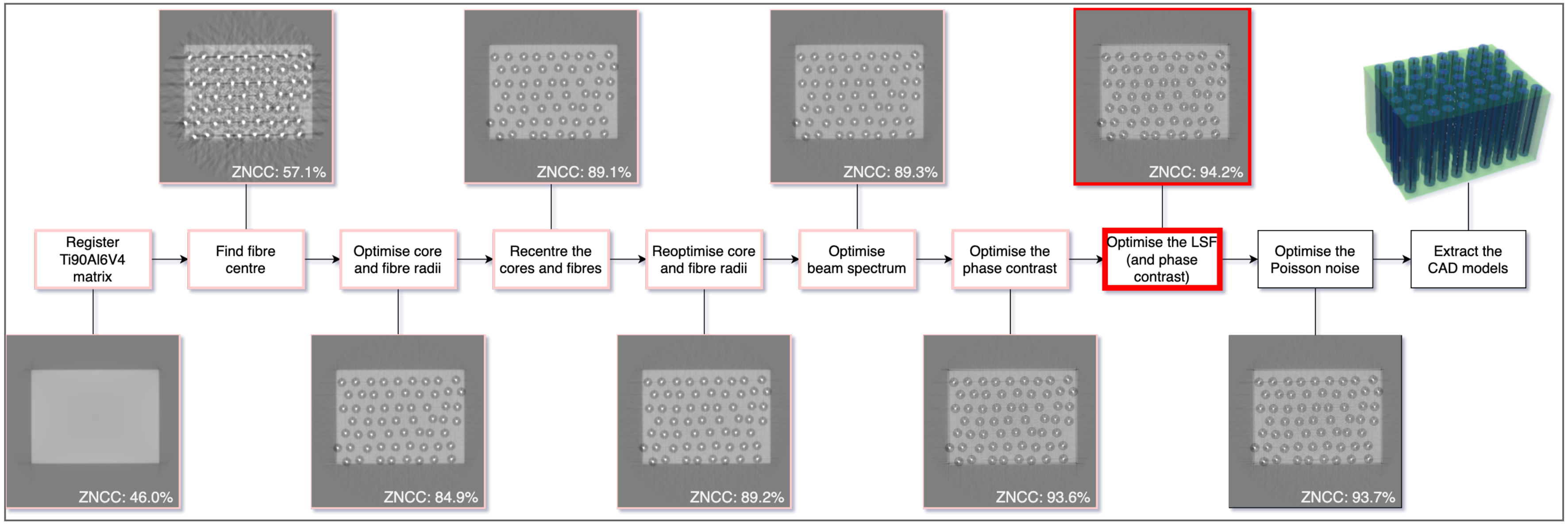
```
Core diameter: 16 um
Fibre diameter: 107 um
```

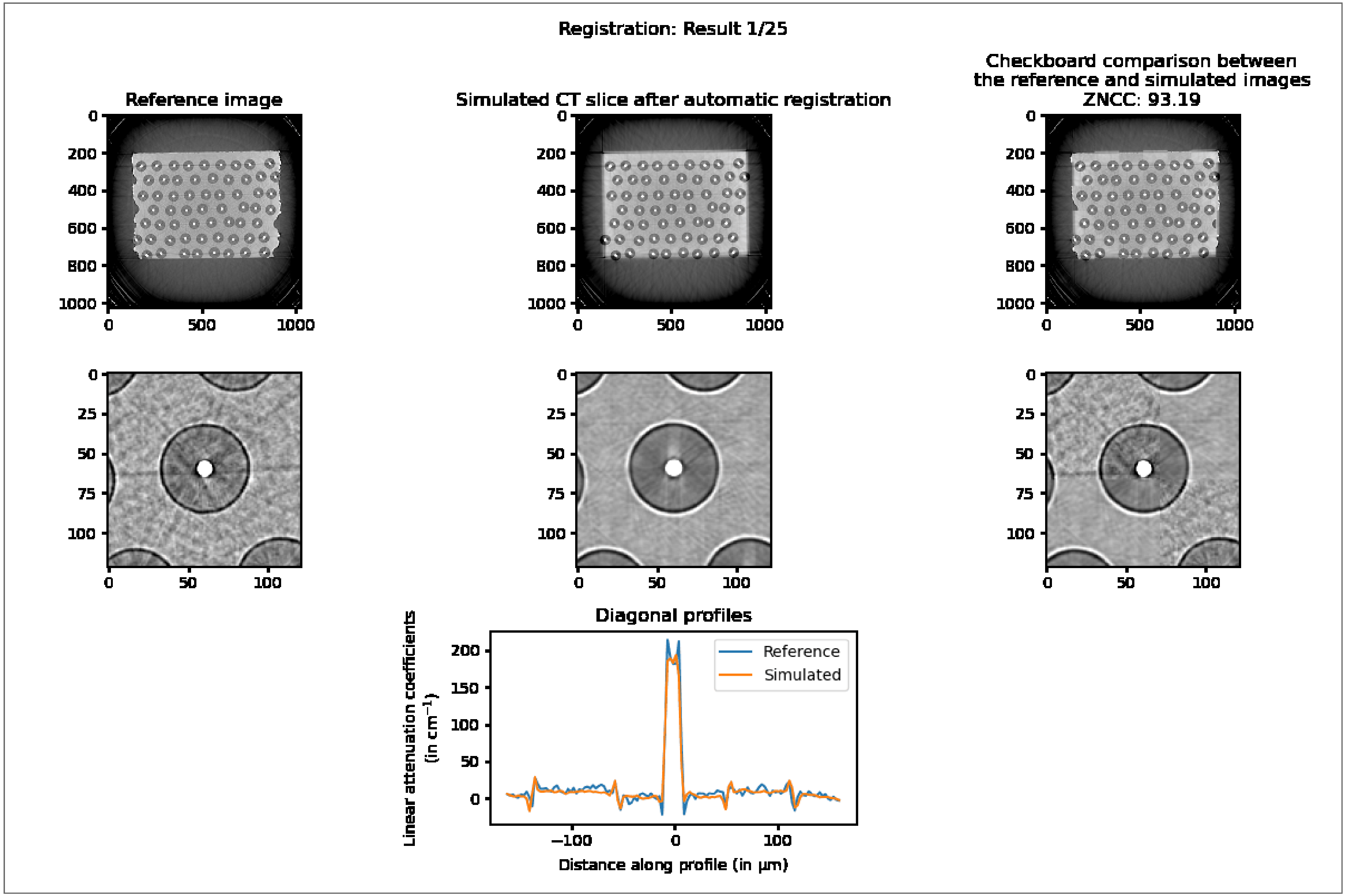
In [95]:

```
plotSetOfZNCC(ZNCC_set, ZNCC_label_set)
```


95
90
85
80
75
70
65
ZNCC in (%)
73.40%
68.56%
87.91%
90.82%
90.71%
90.83%
93.25%
Matrix
matrix registration with fibres
Optimise core and fibre radii
Recentre the cores and fibres
Re-optimise core and fibre radii
Beam spectrum
Phase contrast and radii

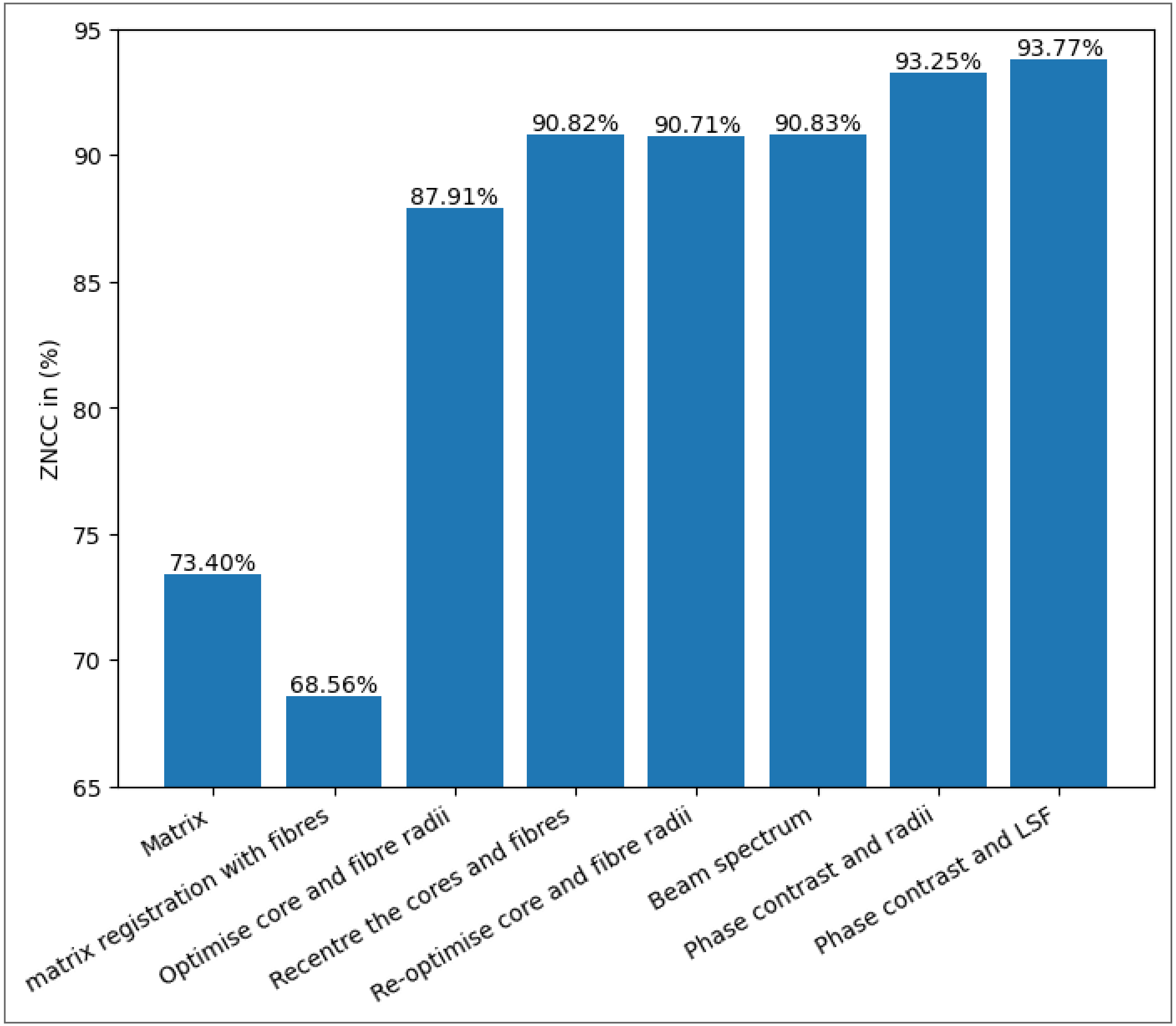
Optimisation of the phase contrast and the LSF

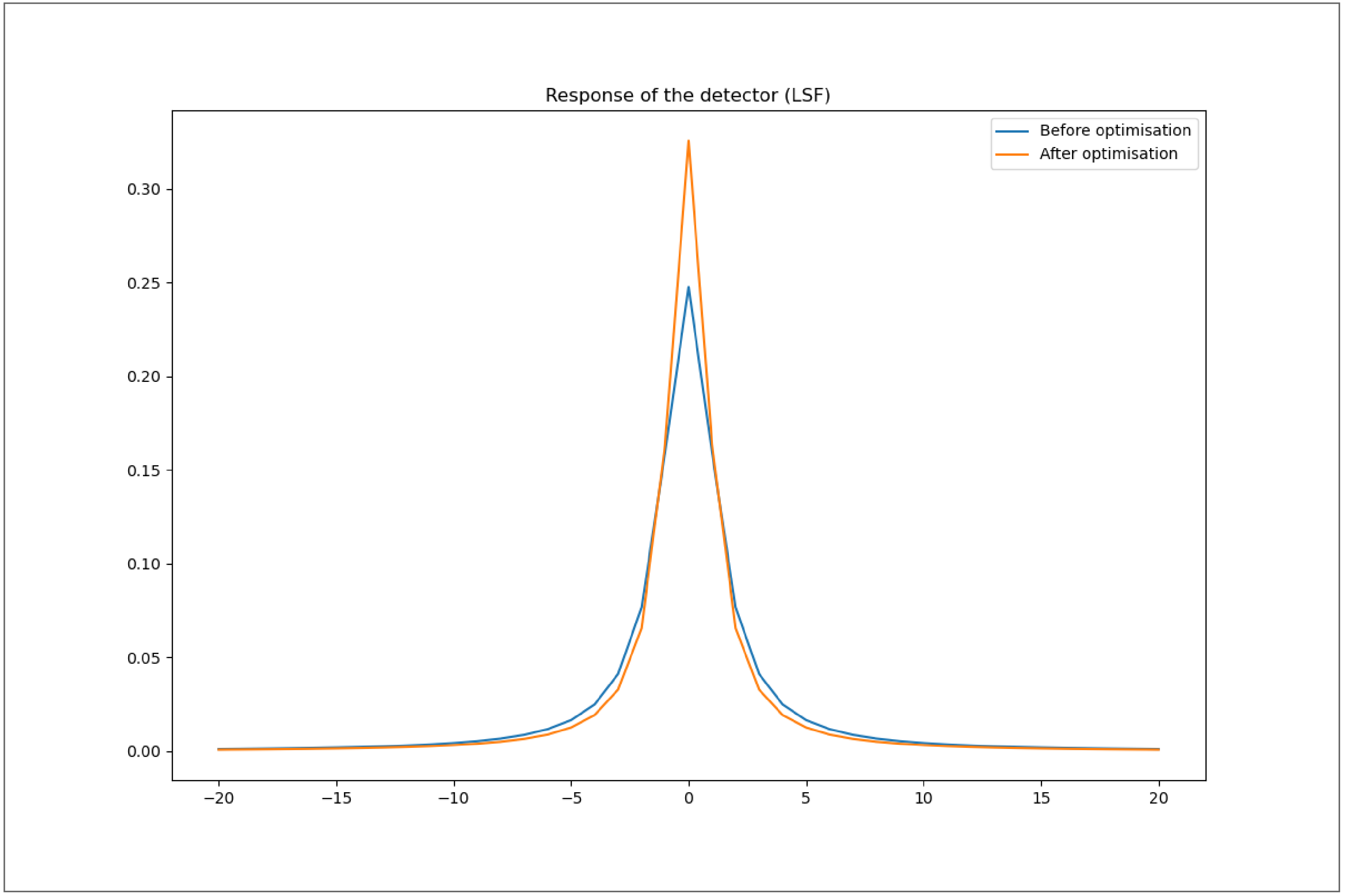



Registration: Result 1/25
Reference image
Simulated CT slice after automatic registration
Checkboard comparison between
the reference and simulated images
ZNCC: 93.19
Diagonal profiles
Reference
Simulated
Linear attenuation coefficients
(in cm^{-1})
Distance along profile (in μm)

In [102]:

```
plotSetOfZNCC(ZNCC_set, ZNCC_label_set)
```


95
90
85
80
75
70
65
ZNCC in (%)
73.40%
68.56%
87.91%
90.82%
90.71%
90.83%
93.25%
93.77%
Matrix
matrix registration with fibres
Optimise core and fibre radii
Recentre the cores and fibres
Re-optimise core and fibre radii
Beam spectrum
Phase contrast and radii
Phase contrast and LSF

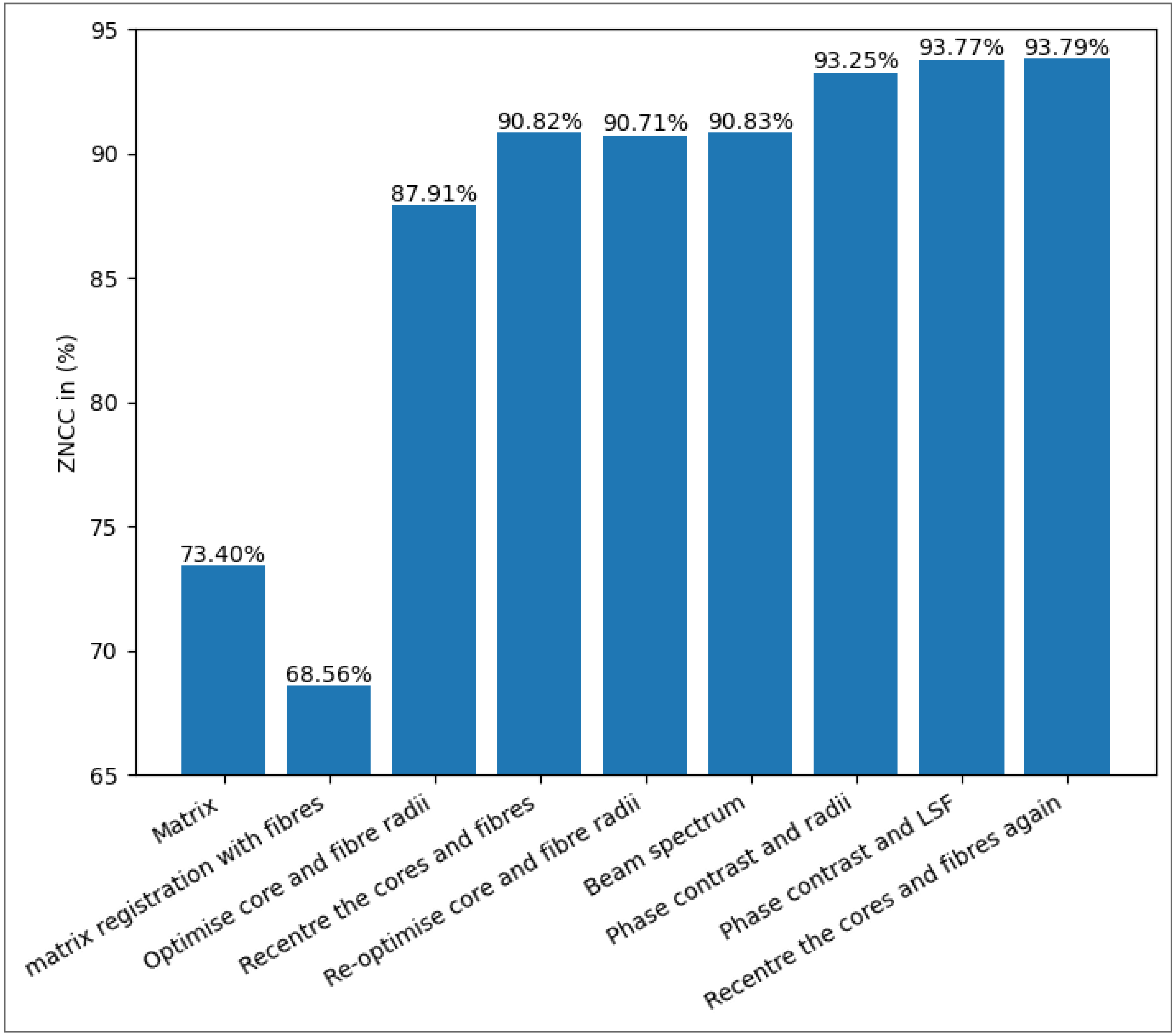


Response of the detector (LSF)
Before optimisation
After optimisation
0.00
0.05
0.10
0.15
0.20
0.25
0.30
−20
−15
−10
−5
0
5
10
15
20

Recentre again

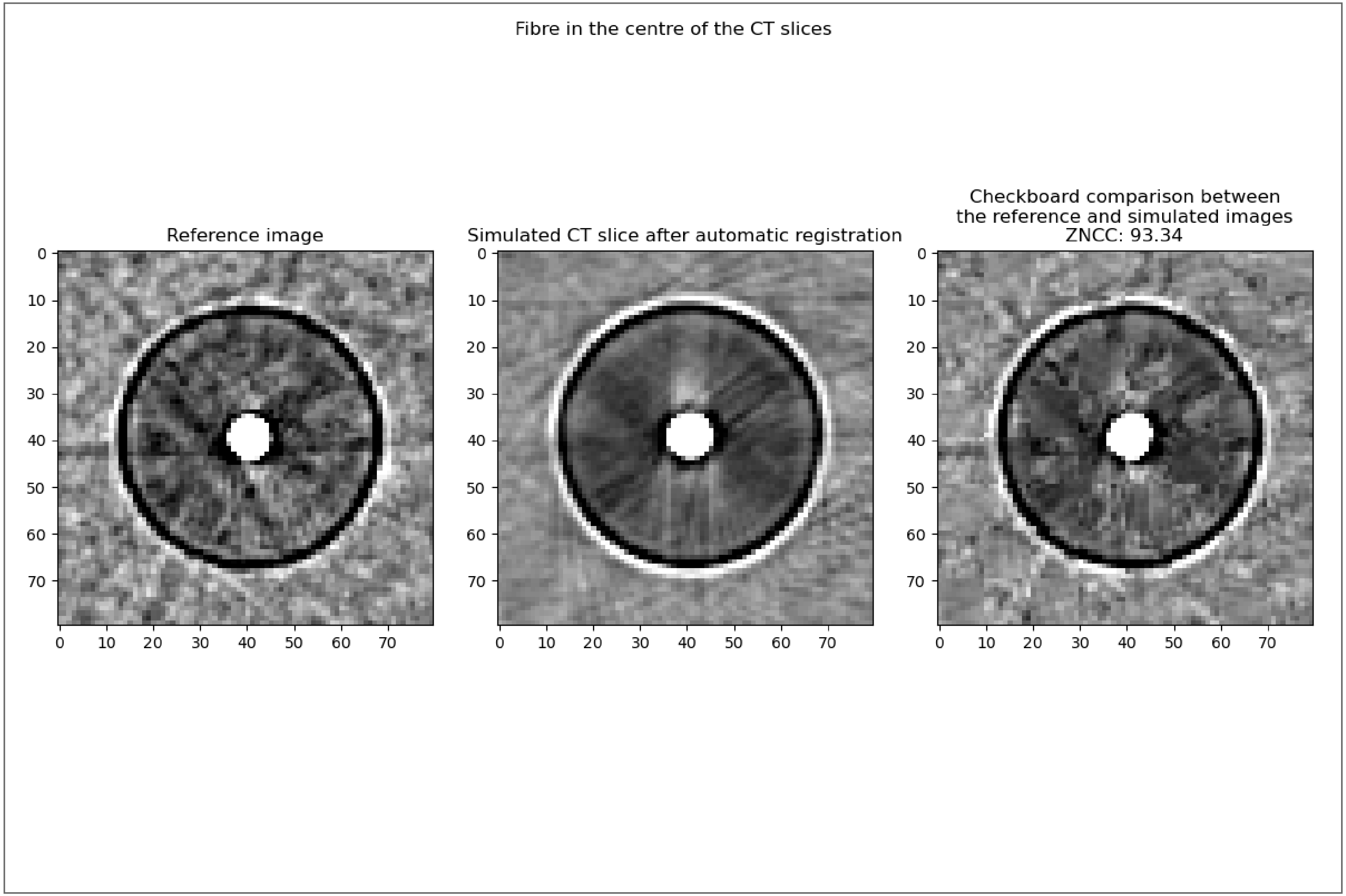
In [105]:

```
plotSetOfZNCC(ZNCC_set, ZNCC_label_set)
```

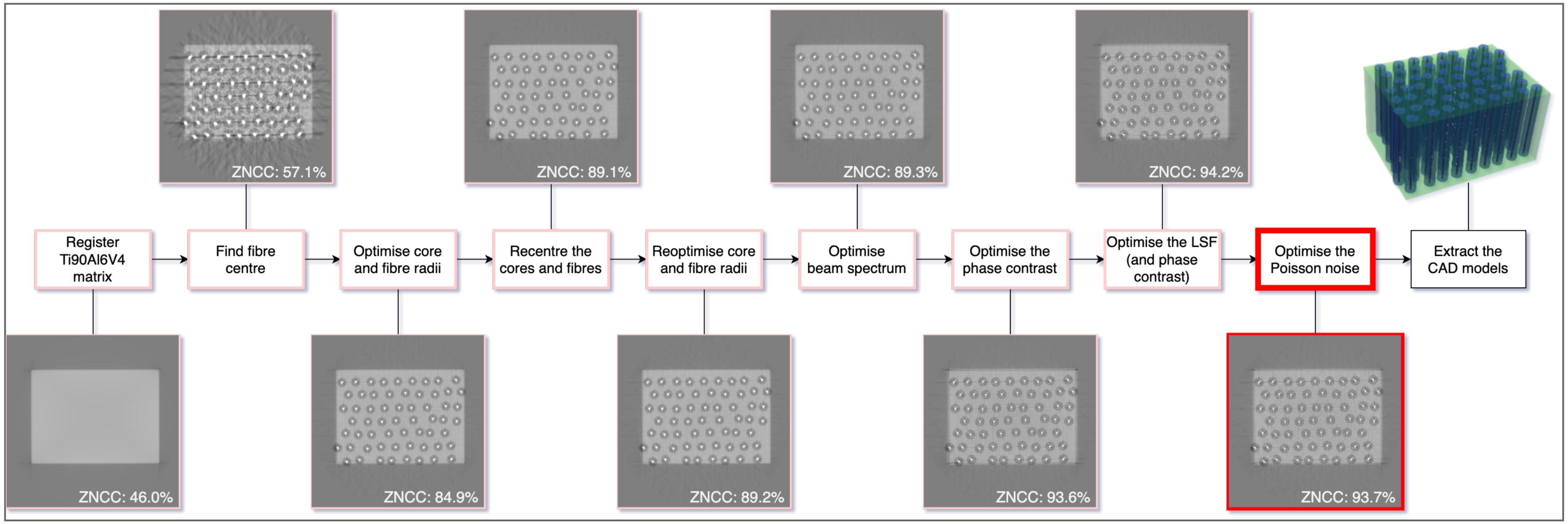


ZNCC in (%)
95
90
85
80
75
70
65
73.40%
68.56%
87.91%
90.82%
90.71%
90.83%
93.25%
93.77%
93.79%
Matrix
matrix registration with fibres
Optimise core and fibre radii
Recentre the cores and fibres
Re-optimise core and fibre radii
Beam spectrum
Phase contrast and radii
Phase contrast and LSF
Recentre the cores and fibres again

Extract the fibre in the centre of the CT slices

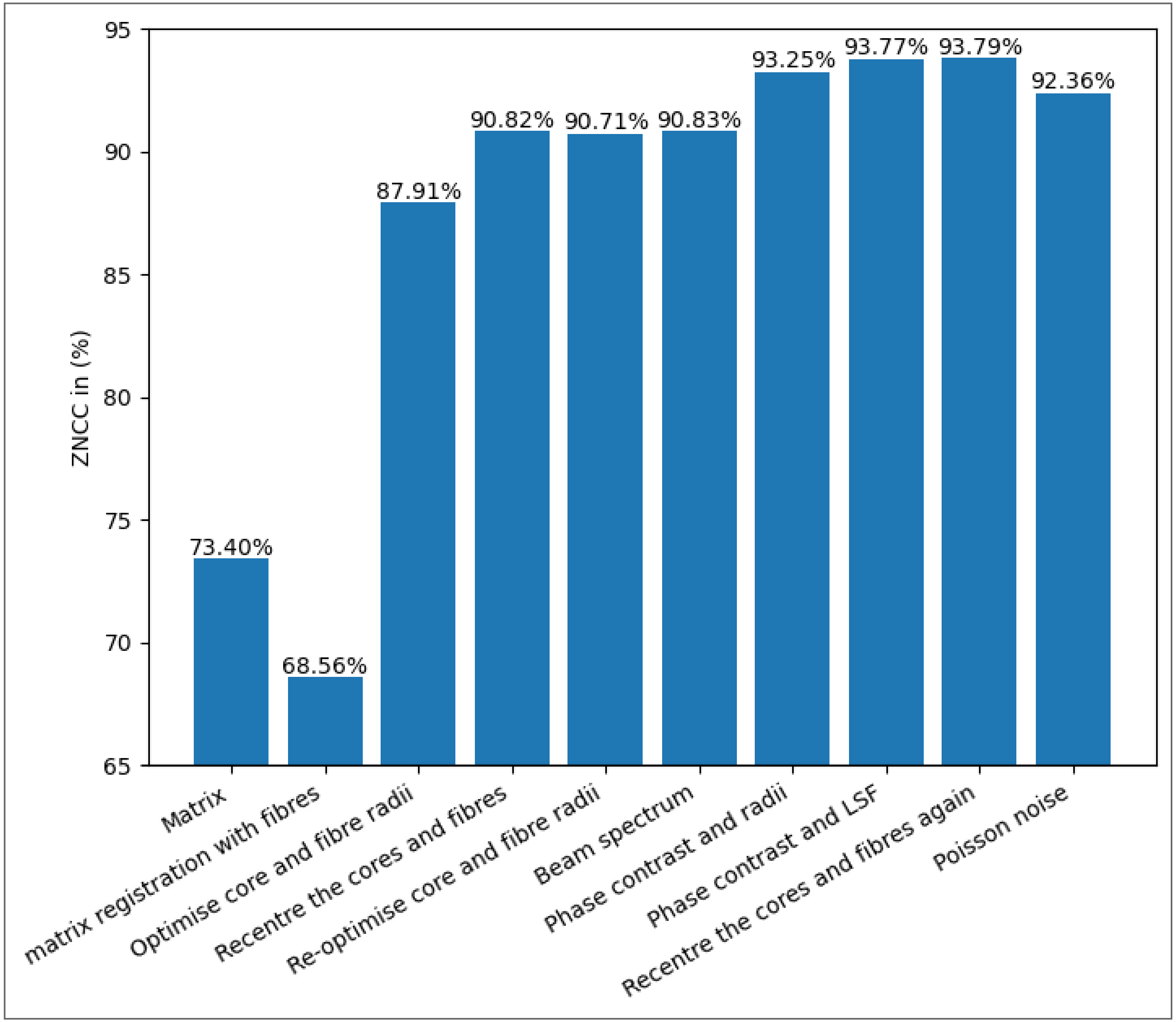

Fibre in the centre of the CT slices
Reference image
Simulated CT slice after automatic registration
Checkboard comparison between
the reference and simulated images
ZNCC: 93.34
0
10
20
30
40
50
60
70

Optimisation of the Poisson noise



In [114]:

```
plotSetOfZNCC(ZNCC_set, ZNCC_label_set)
```


95
90
85
80
75
70
65
ZNCC in (%)
73.40%
68.56%
87.91%
90.82%
90.71%
90.83%
93.25%
93.77%
93.79%
92.36%
Matrix
matrix registration with fibres
Optimise core and fibre radii
Recentre the cores and fibres
Re-optimise core and fibre radii
Beam spectrum
Phase contrast and radii
Phase contrast and LSF
Recentre the cores and fibres again
Poisson noise

Manual measurement vs automatic

Matrix

What	Manual (wihtout outlier)	Registration (without artefacts, wihtout outlier)
Width (in µm)	1452.8±12.1	1454.0±8.0
Height (in µm)	1061.6±9.9	1061.2±7.5
Rotation (in degrees)	90.7±0.4	90.7±0.5

Similar results, once the outliers were discarded.

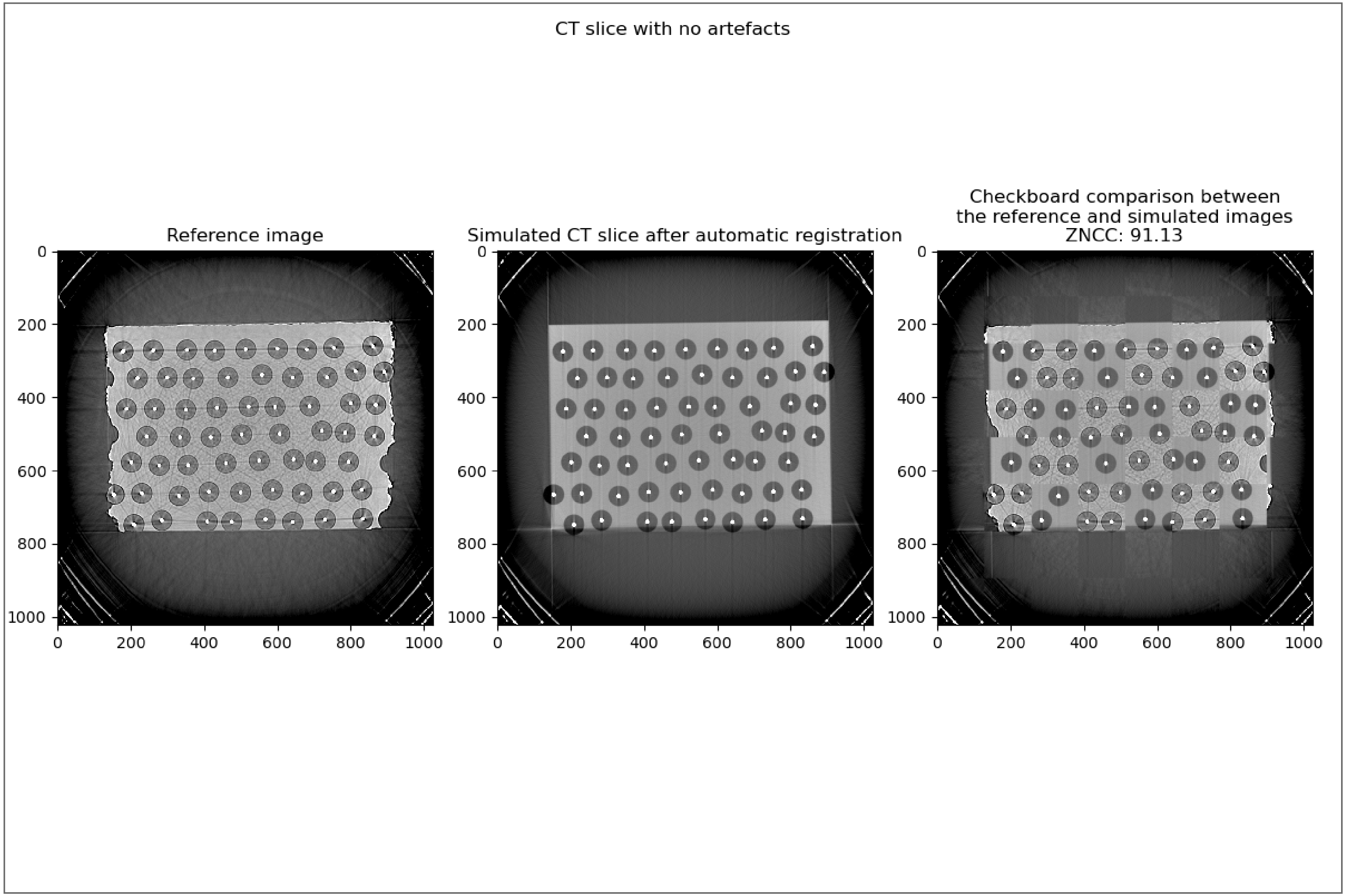
Fibres and Cores

What	Manual (wihtout outlier)	Registration (without artefacts, without outlier)	Registration (with artefacts, wihtout outlier)
Core diameter (in µm)	15.8±0.1	14.8±0.2	15.8±0.1

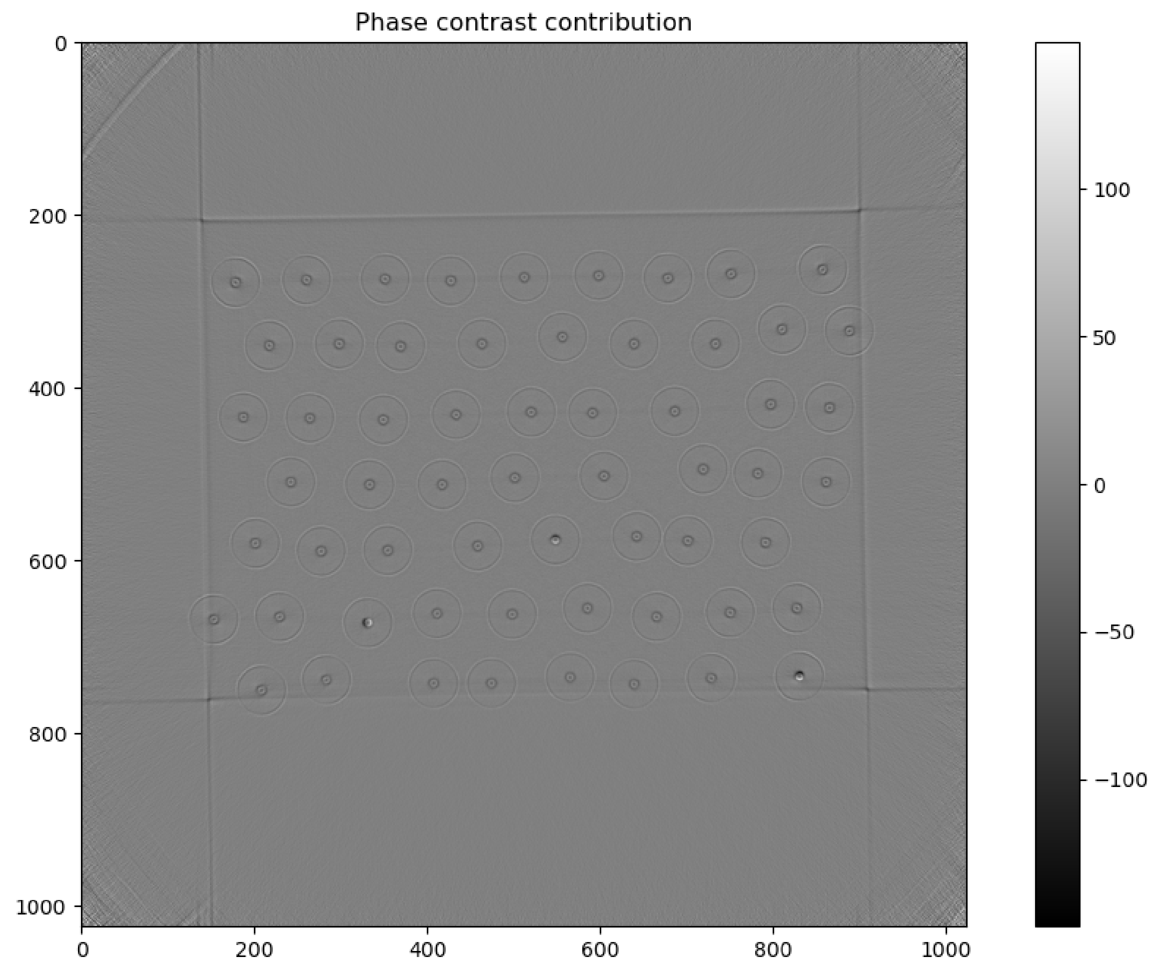
What	Manual (wihtout outlier)	Registration (without artefacts, without outlier)	Registration (with artefacts, wihtout outlier)
Fibre diameter (in µm)	105.0± 7.5	96.8±2.1	107.4±0.2

Similar results for the manual measurements and registration with artefacts for the cores only. The registration without artefacts underestimated the core diameter. The diameter was underestimated by the volunteers and the registration without artefacts due to the phase artefacts.

Improving μ: Subtracting the contribution of each artefact sources

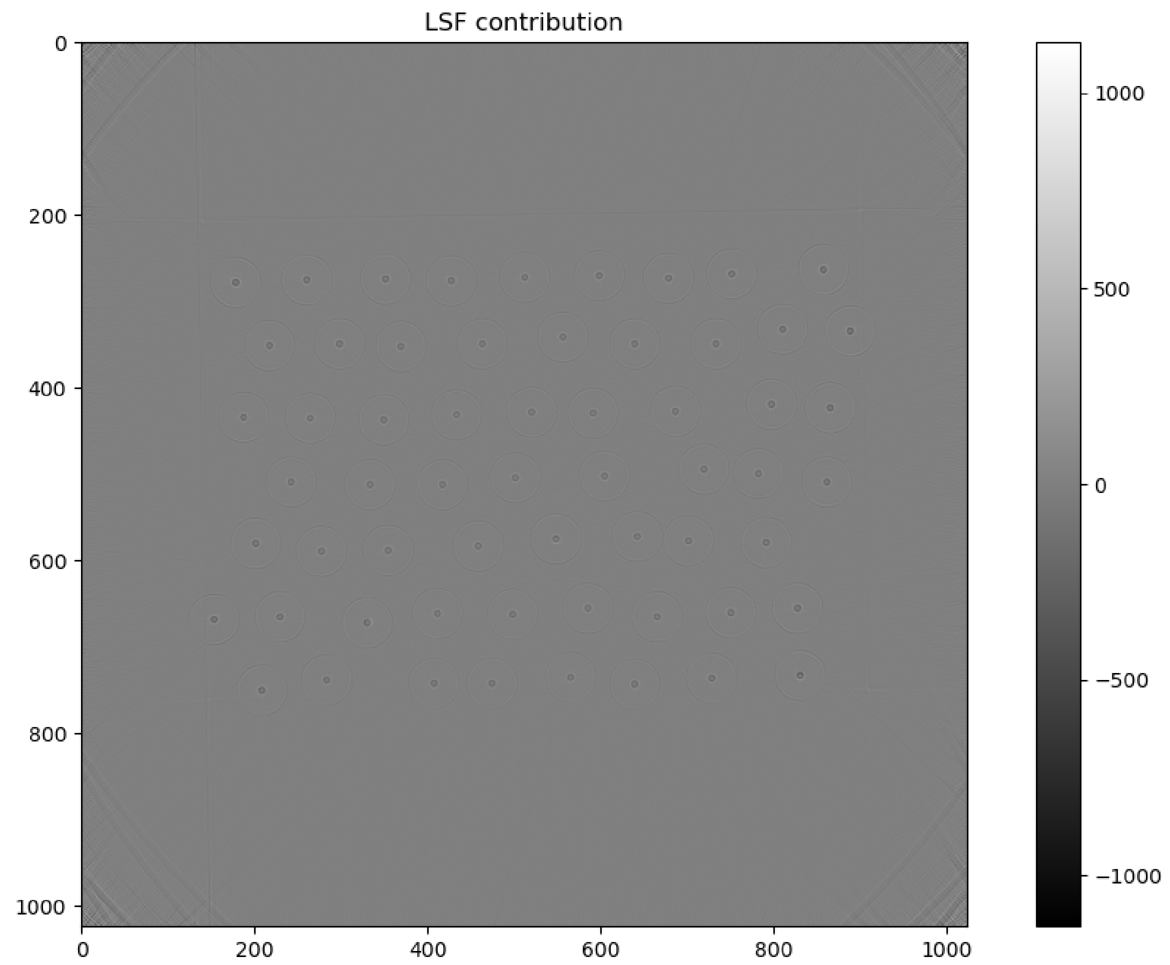

CT slice with no artefacts
Reference image
Simulated CT slice after automatic registration
Checkboard comparison between the reference and simulated images
ZNCC: 91.13
0
200
400
600
800
1000

Phase contribution to artefacts



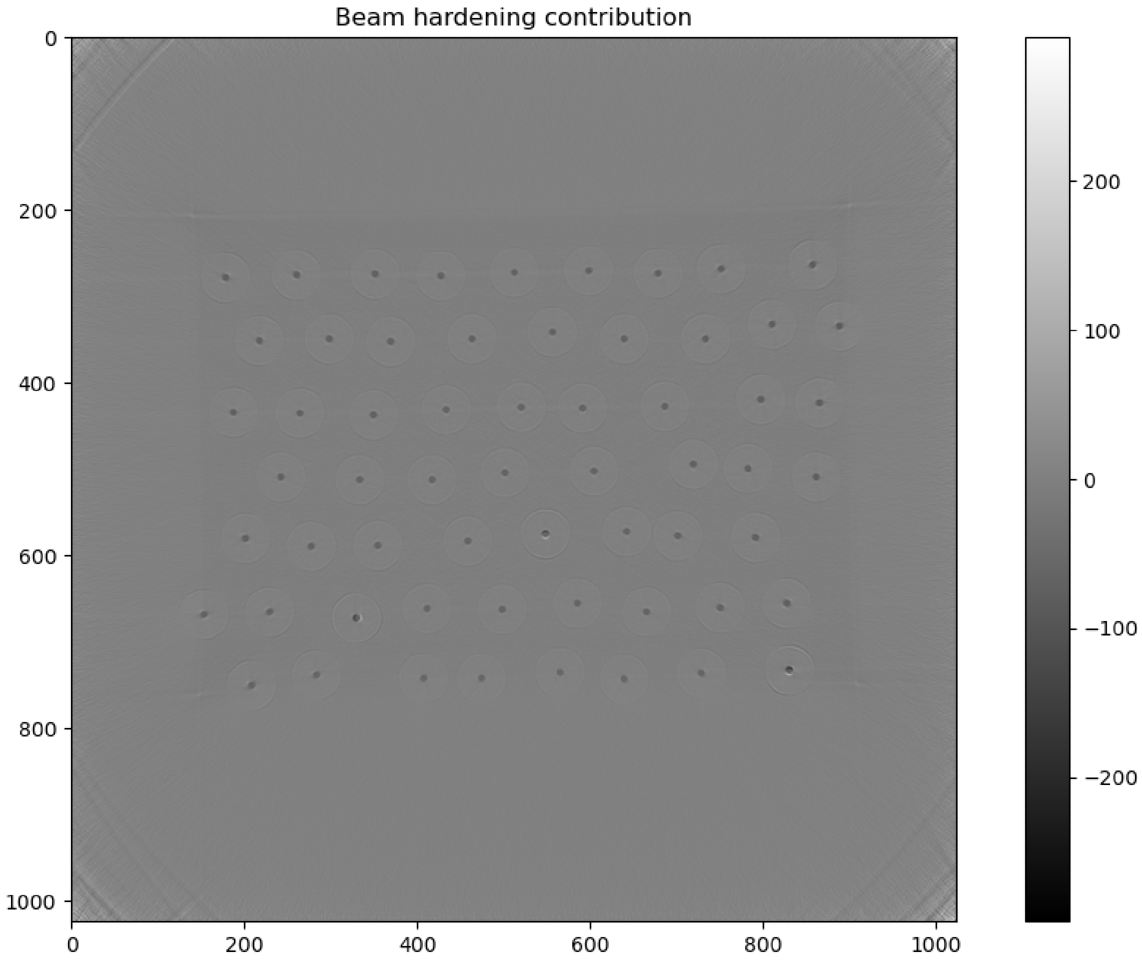
Phase contrast contribution
0
200
400
600
800
1000
100
50
0
−50
−100

LSF contribution to artefacts

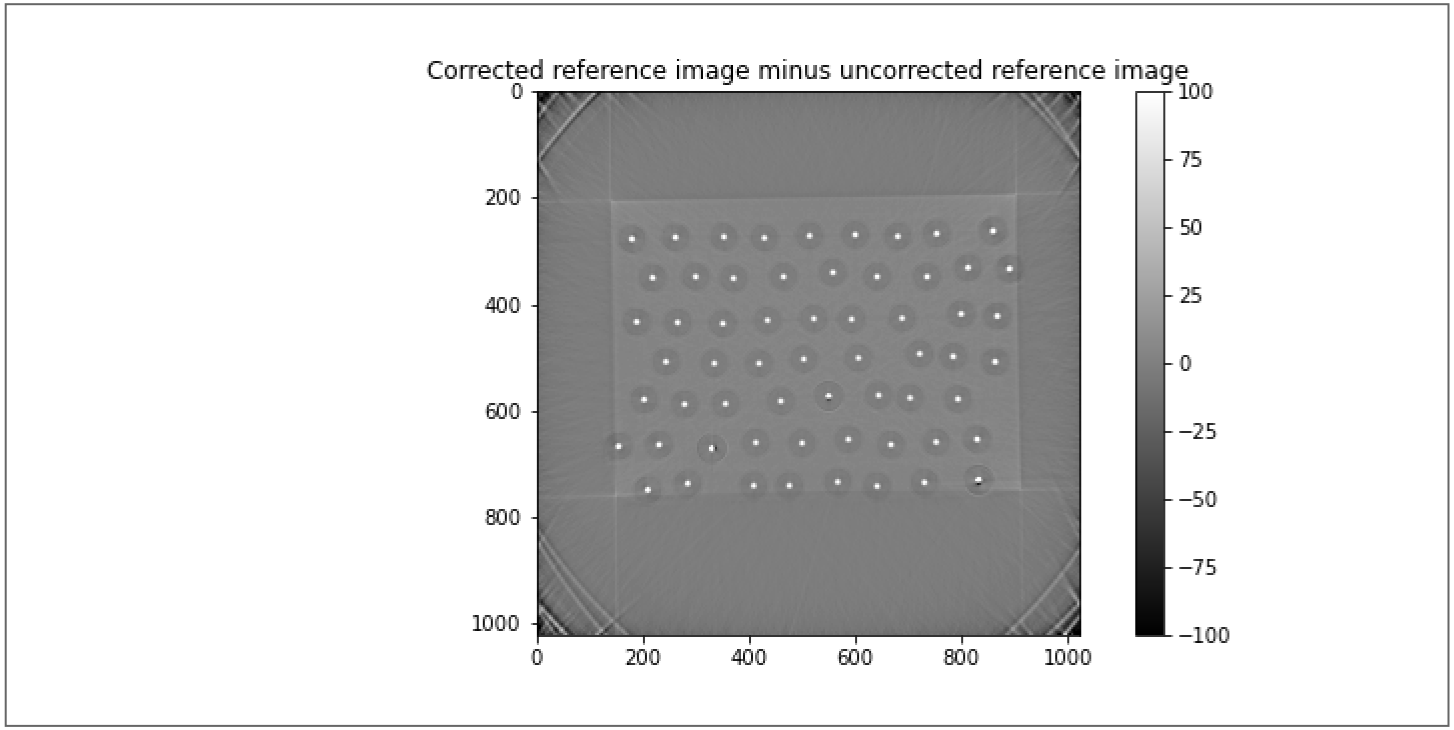


LSF contribution
0
200
400
600
800
1000
0
200
400
600
800
1000
1000
500
0
−500
−1000

Beam hardening contribution to artefacts

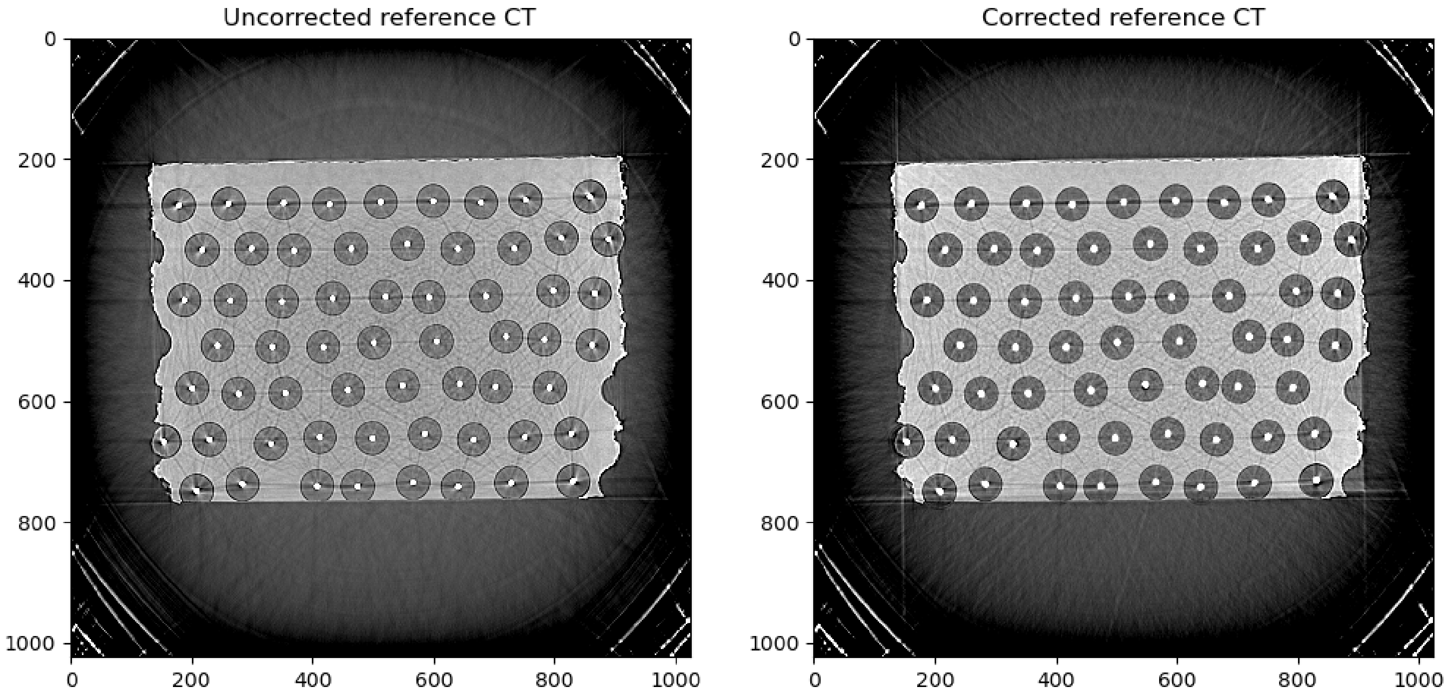


Correction map

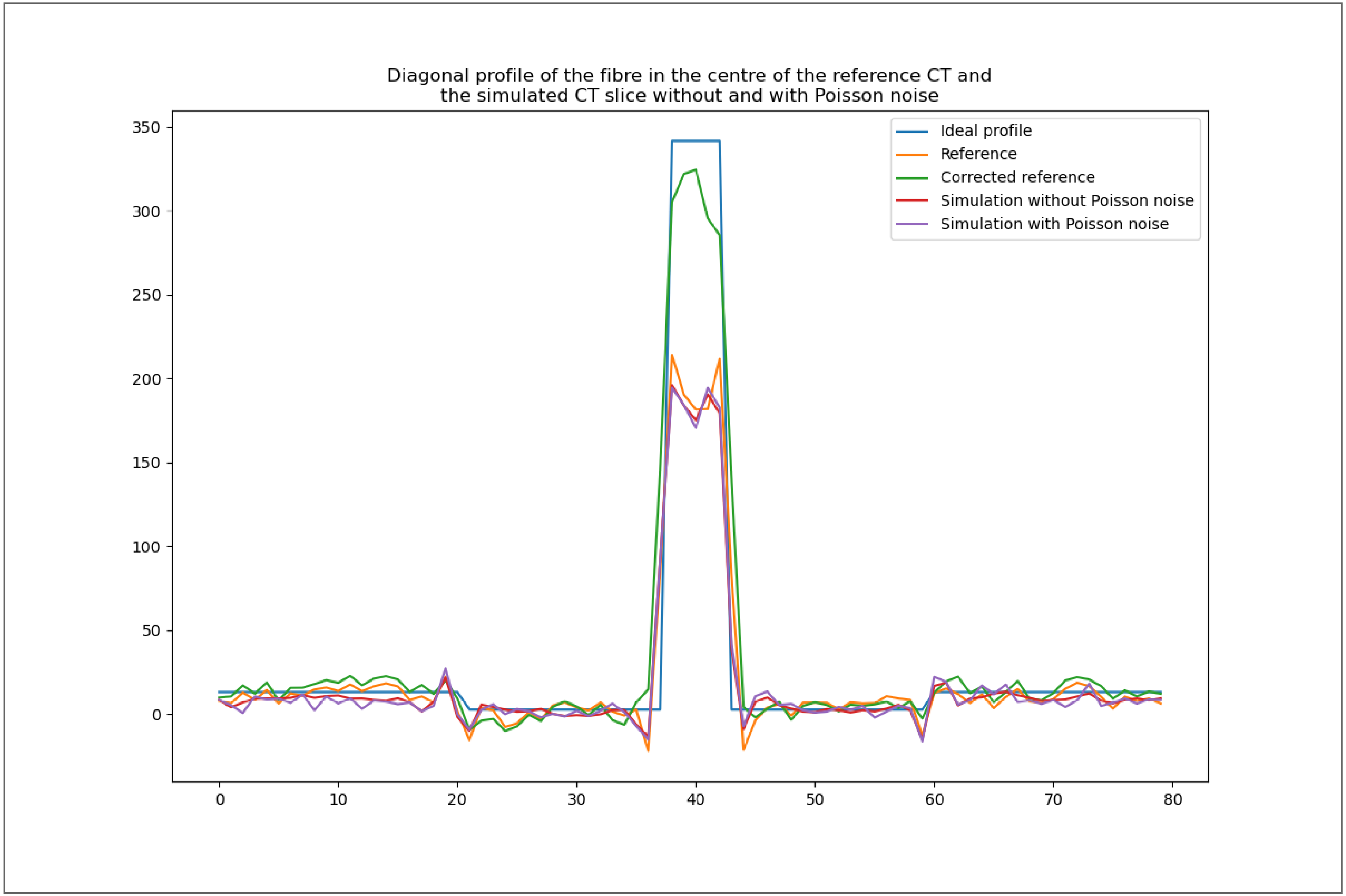

Corrected reference image minus uncorrected reference image
0
200
400
600
800
1000
0
200
400
600
800
1000
100
75
50
25
0
−25
−50
−75
−100

Corrected CT slice

CT slices
(in linear attenuation coefficients, cm^{-1})



Profile through the fibre in the centre of the slice



Diagonal profile of the fibre in the centre of the reference CT and
the simulated CT slice without and with Poisson noise
Ideal profile
Reference
Corrected reference
Simulation without Poisson noise
Simulation with Poisson noise
350
300
250
200
150
100
50
0
0
10
20
30
40
50
60
70
80

Results in terms of linear attenuation coefficients

In [142]:

```
display(df)
```

	CT	Structure	Composition	mean	error
0	Theorical	Core	W	341.610000	N/A
1	Experimental (corrected)			295.885223	45.724777
2	Experimental			193.974380	147.63562
3	Simulation			175.826019	165.783981
4	Theorical	Fibre	SiC	2.736000	N/A
5	Experimental (corrected)			1.274483	1.461517
6	Experimental			3.189285	-0.453285
7	Simulation			2.285063	0.450937

	CT	Structure	Composition	mean	error
8	Theorical	Matrix	Ti90Al6V4	13.127400	N/A
9	Experimental (corrected)			15.013135	-1.885735
10	Experimental			10.639873	2.487527
11	Simulation			9.695848	3.431552

Conclusion

- Manual measurements:
  - may seem to be relatively easy to perform
  - can be prone to bias and unreliable.
- **Realistic projection model required** to automatically produce an accurate CAD models from X-ray tomography **when CT data is so corrupted by artefacts**.
- Our model takes into account geometrical properties, beam hardening, impulse response of the detector, phase contrast, and photon noise.
- The **choice of the objective function** for the optimisation is very important.
- Our method is only suitable when it is possible to generate surface models.
  - Not trivial to model complex shapes.

- All the geometrical properties, e.g. tilting and warping the shapes, must be implemented if needed.
- Taking into account features such as **pores** or **fibre tows**, even if made of basic shapes, is not straightforward.

Future work

- Deformable models to generate complex shapes.
- Model-based iterative reconstruction:
    - Improve qualitative data (remove the artefacts in the image),
    - Improve quantitative data (improve the μ values).
- (if you have a CT scan with simple geometries, we'll be happy to collaborate and test the framework with your data)

Thanks for your attention

(contact details: f.vidal@bangor.ac.uk)