Lecture 7

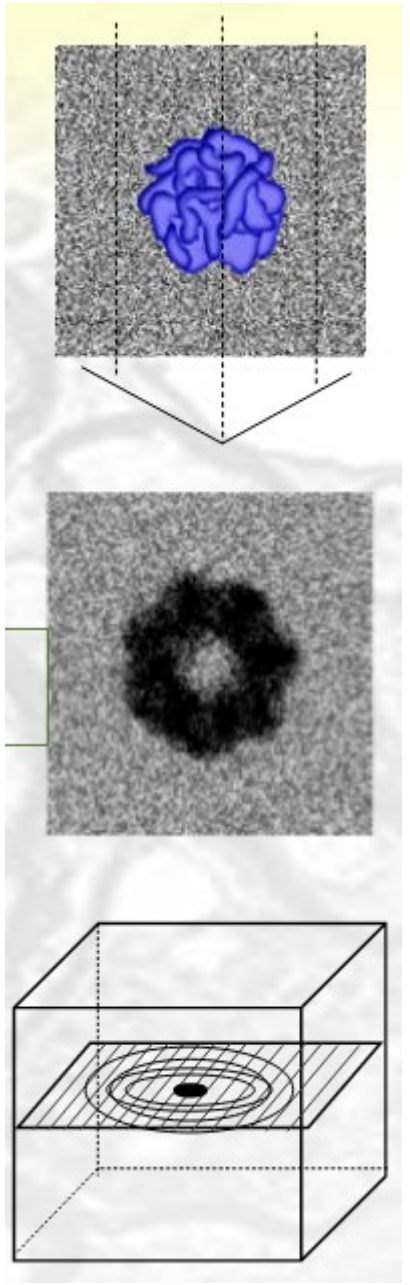
3DEM methods

Single Particle Analysis

Tibor Füzik

Single particle analysis

- Single particle analysis
  - The most well-established method for 3D structure estimation of vitrified samples at molecular resolution
  - “Not single particle at all”
  - Hundreds to million particles averaged together
  - If we know the projection of the object from all possible views, we can create a 3D model of the projected model (application of Projection theorem)
- Specimen definition
  - Pure sample of biological molecules, complexes
  - Sample embedded in vitreous ice
    - thin enough – to avoid high background
    - thick enough to avoid interaction with air-water interface
  - Ideally rigid molecule with low amount of variability and flexibility
  - Assumption that the molecule occupies random orientation in the vitrified specimen
- Data acquisition strategy
  - Cryo-electron microscope equipped with direct electron detector
  - Fast acquisition of 2D projection images (the more particles the better)
  - Balancing the dose and the radiation damage to the sample
  - Balancing the pixel-size vs view-field (defining the Nyquist freq. of the data)
  - Varying applied defocus



https://www.embl-hamburg.de/.../EMBO_cryoEM_2_Bhushan.pdf

Why to acquire in “movie mode”

- Frames vs fractions
  - Internal readout speed of the camera
    - Electron flow hitting the sensor in time set to amount that max 1 e^- will hit 1 px at camera surface
    - Individual count events per frame
    - Image from 1 frame -> Like stars on the sky
  - Several frames summed -> fractions
    - Contain enough information per fraction for further processing
    - Dose fractionation
    - Dose per fraction: ~$1e^-/A^2$
- Recording 1 sec exposure as 400 frames
  - grouped into 100 fractions per 4 summed frames
  - grouped into 10 fractions per 40 summed frames
- Alignment of frames => drift compensation (reduce motion blur)
- Ability to compensate for the beam induced damage
- Movie frames <-> Movie fractions

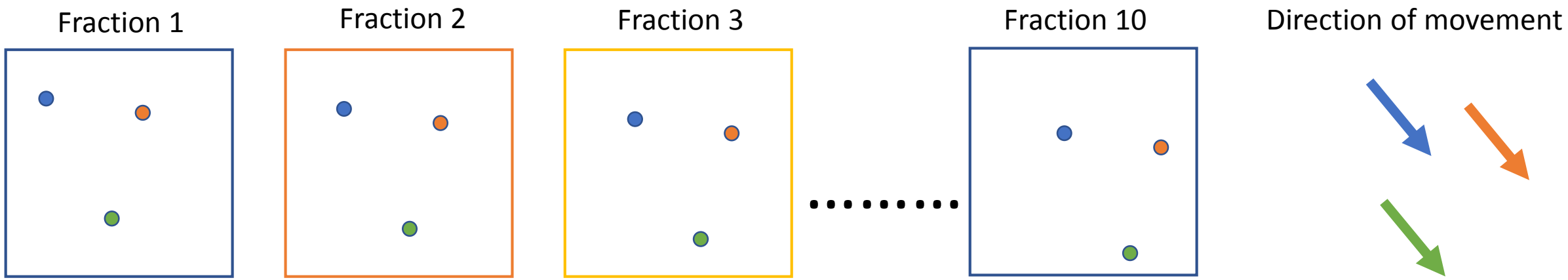
1 sec

Acquired [frames]	Stored [fractions]
1	1
2	
3	
4	
5	2
6	
7	
8	
9	3
10	
11	
12	
.	.
.	.
.	.
399	100
400	

Increasing radiation damage

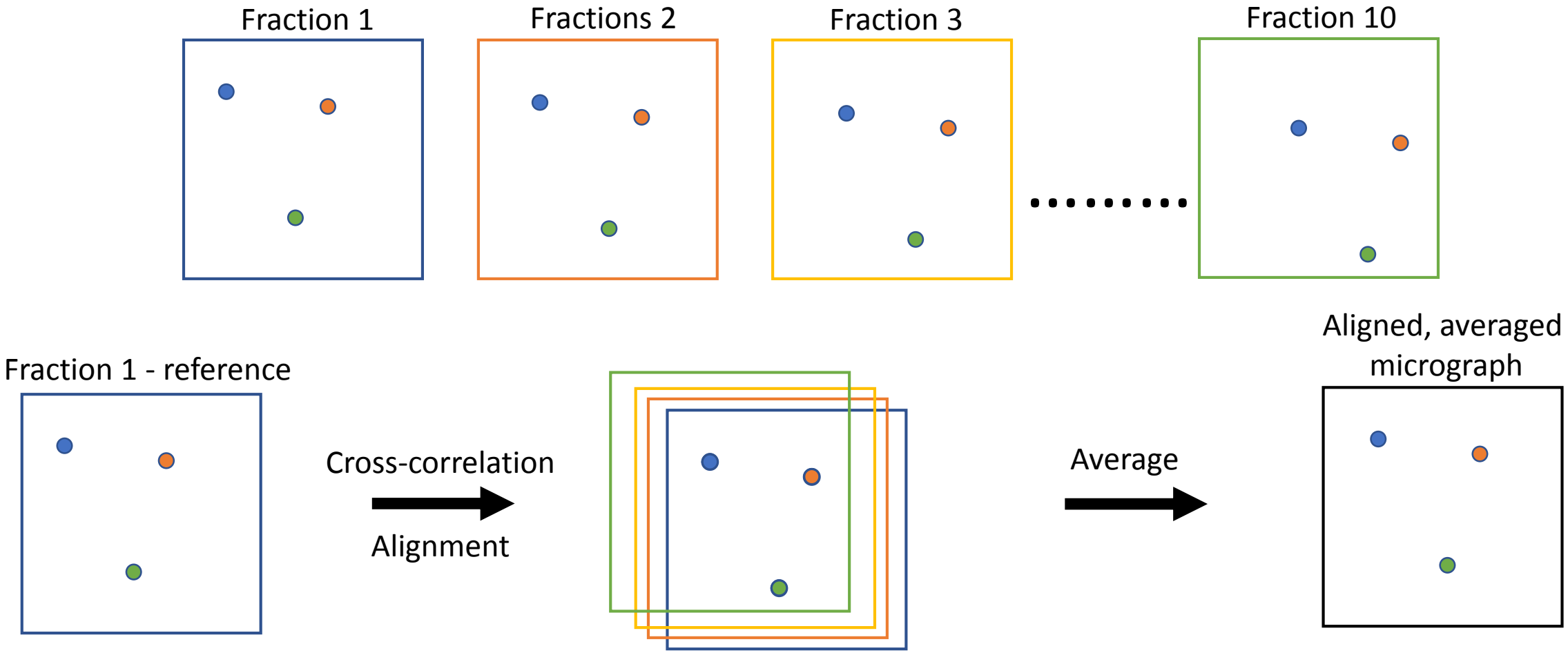
Drift movement of the sample

- Caused by nonstable stage
- Beam induced heating and drifting
- Larger movement
- Uniform movement of the whole image per fraction (direction, speed)
- Speed and direction may vary from fraction to fraction



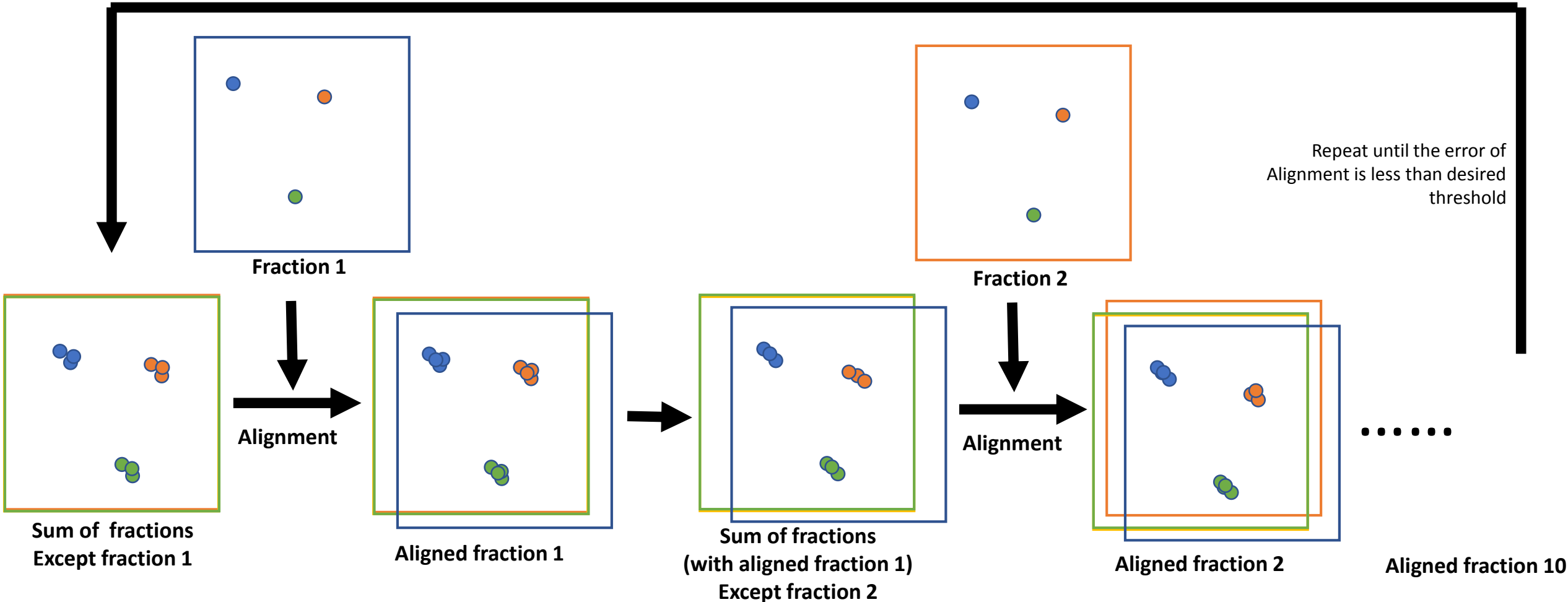
Compensation of drift induced movement

- Correlation based frame averaging
  - Take a reference frame (first / last)
  - Align the rest of frames against this frame according to highest correlation
  - Crop according to the reference

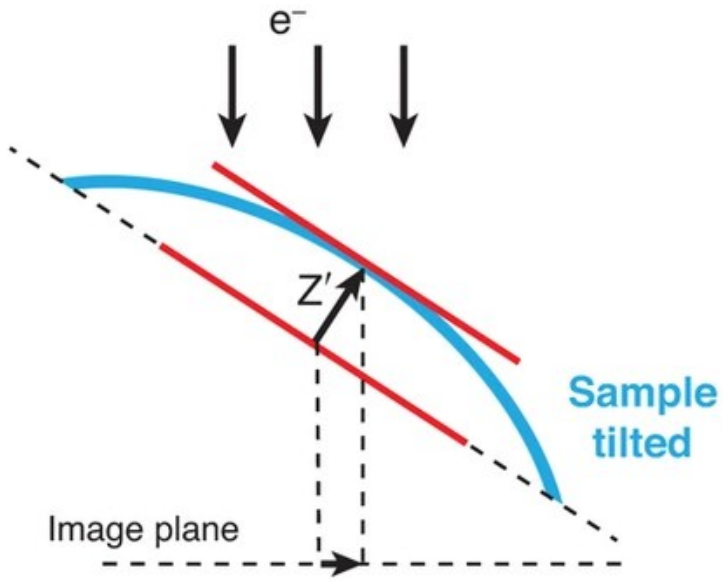


Compensation of drift induced movement

- Iterative correlation-based fraction (frame) averaging
  - Single fraction does not contain enough information to be reference for the rest of the fractions (high noise level)
  - Aligning a single fraction against the sum of the rest fractions
  - Do it iteratively until the X/Y shift error reached a predefined value

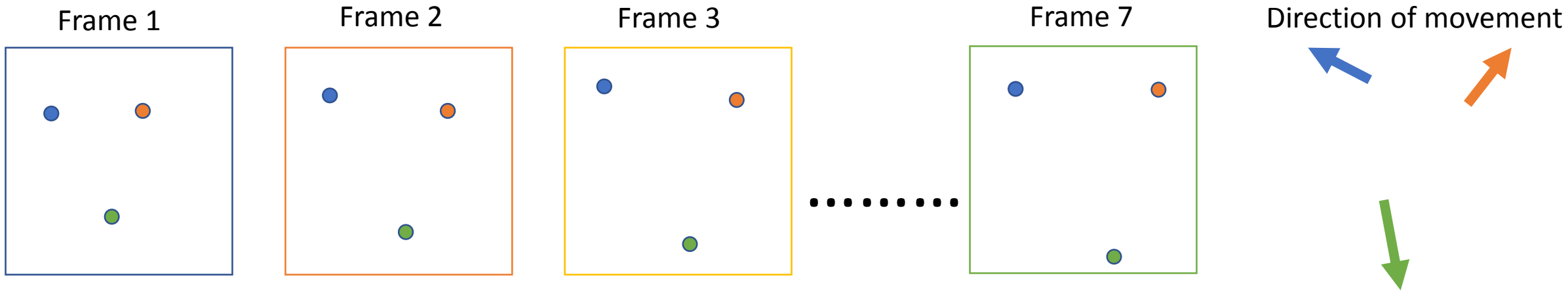


Beam induced movement



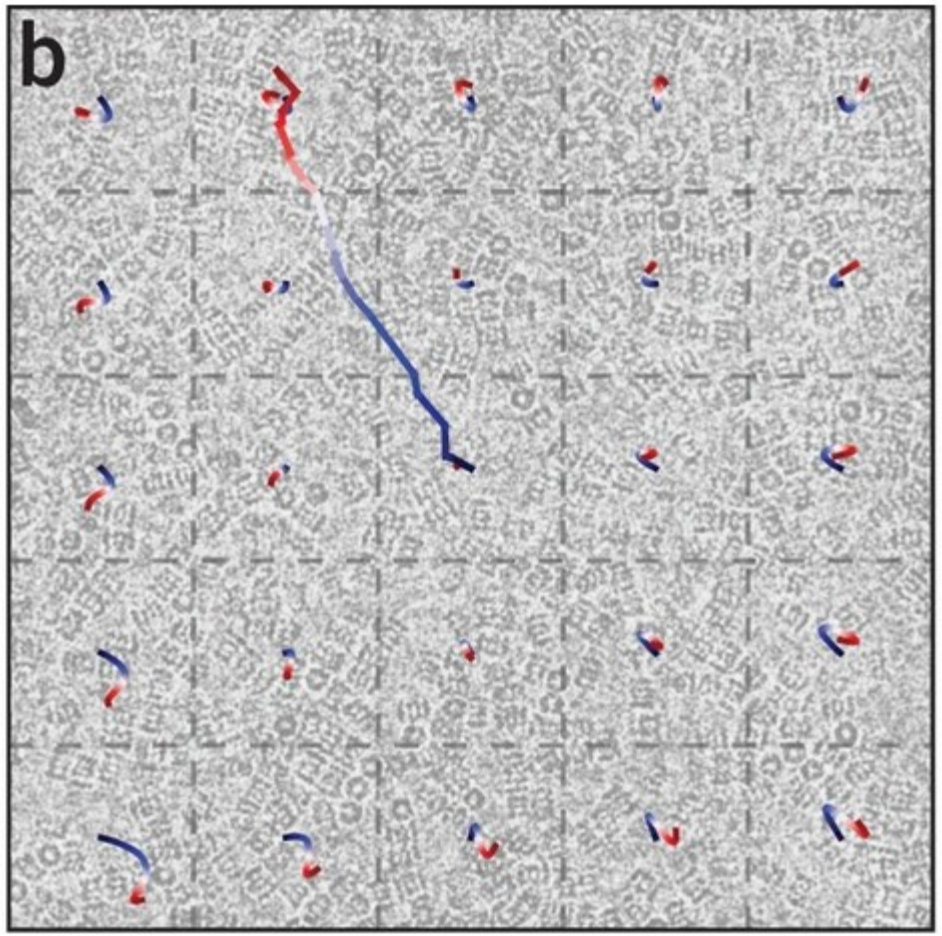
Zheng et al. 2017

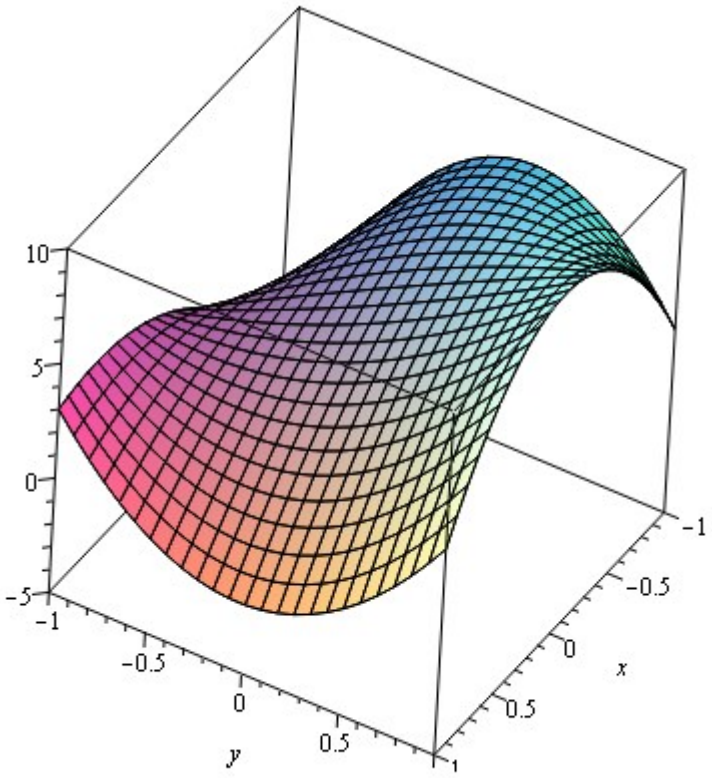
- Caused by beam induced heating/damage
- Smaller movement
- “Random-like” movement -> follows doming effect
- Mainly parts containing “damageable” material move the most



Compensation of beam induced movement

- Performed after global alignment
- Split the images into predefined number of patches (3x3, 5x5)
  - Iterative correlation-based estimation of patch shifts
  - Fit a 2nd order surface on the estimated shifts (every fraction has its own interpolated surface)
  - Deform the fraction image using the 2nd order surface (no checkerboard effect on the patch borders)
  - Sum the deformed (corrected fractions)

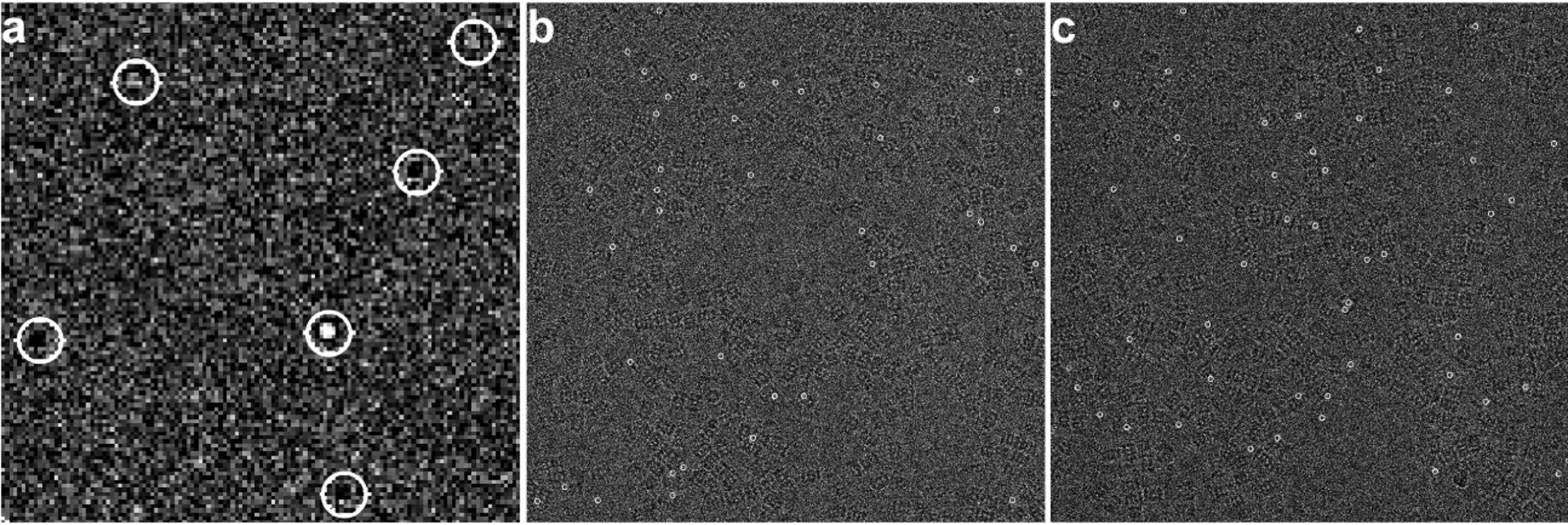




Zheng et al. 2017

Defects (bad pixels) on the detector

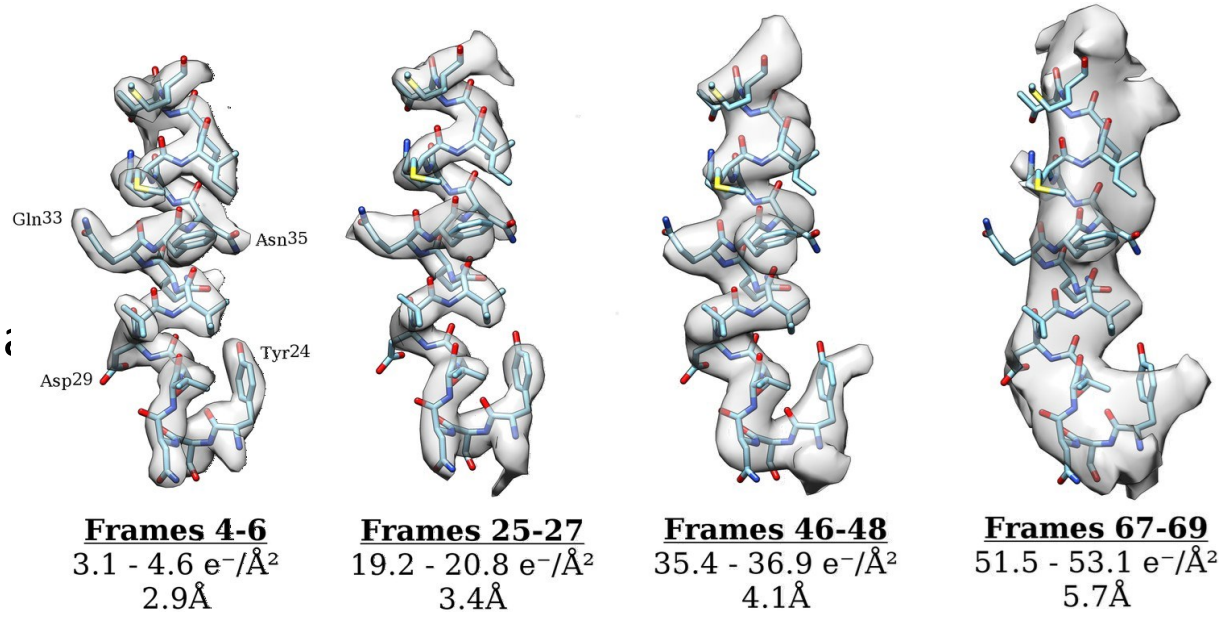
Need to be compensated (give strong false correlation)



Zheng et al. 2017

Dose weighting

- The more radiation the higher SNR – “more visible particles”
- The more radiation the more radiation damage
- Limiting the radiation to achieve high-res reconstructions
- First fractions contain more high-res information tha the last fractions
- Downweighing the hi-res information in the more exposed fractions
- Applying a dose weight filter allows to use higher doses
- Doable without reference (unlike Bayesian polishing)



The less radiation the more hi freq info

Grant & Grigorieff 2015
https://doi.org/10.7554/eLife.06980.001

Spatial frequency

$$q(k, N) = e^{-\frac{N}{2N_e(k)}}$$

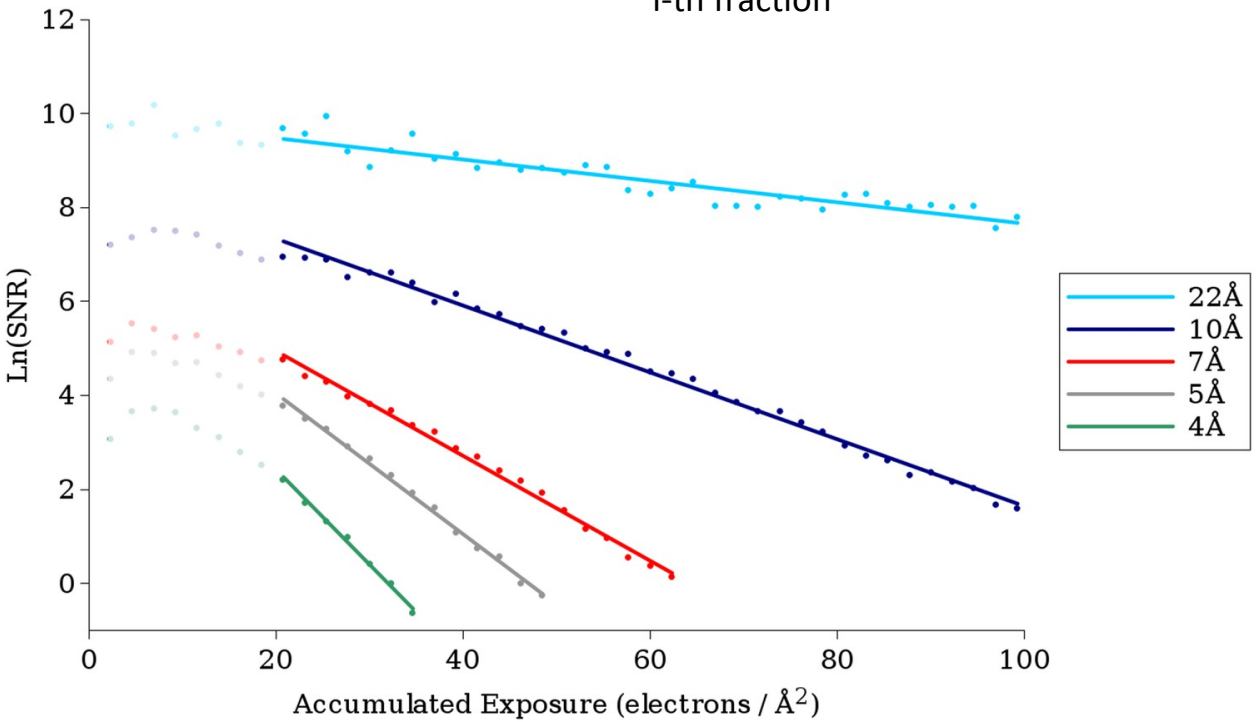
exposure-dependent amplitude attenuation

accumulated exposure of the frame

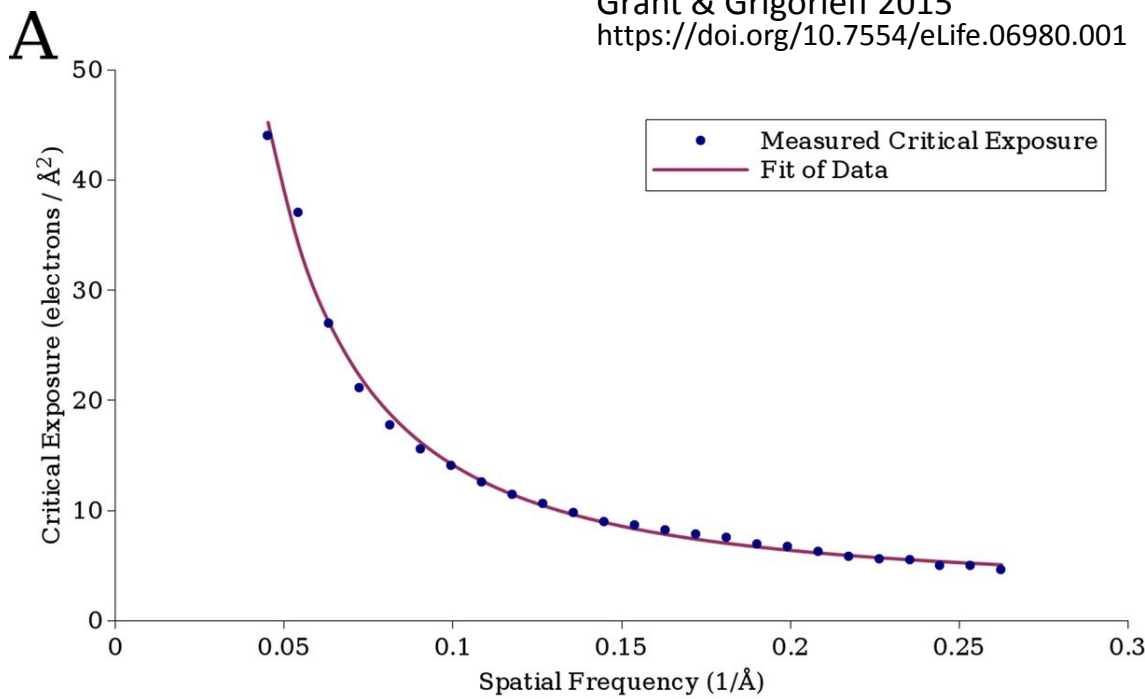
$$\widetilde{F}^{W}(\mathbf{k}) = \frac{\sum_{i=1}^{n} q(k,N_i) F_i(\mathbf{k})}{\sqrt{\sum_{i=1}^{n} q(k,N_i)^2}}$$

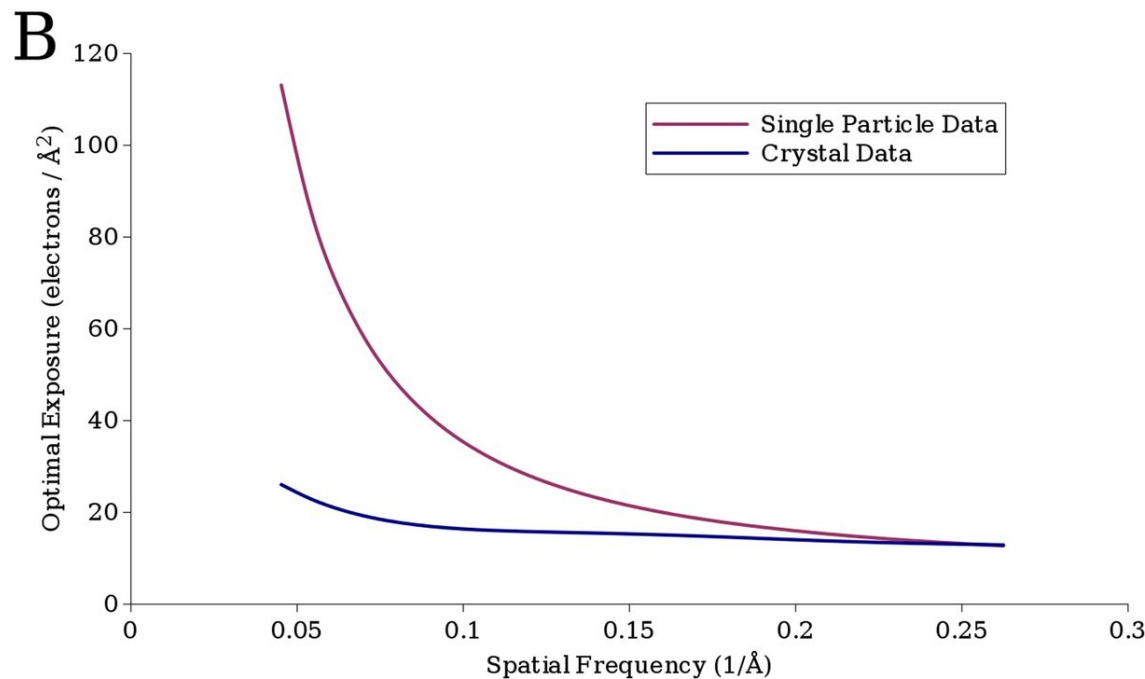
Dose-weighted Fourier transform of Image

i-th fraction



Grant & Grigorieff 2015
https://doi.org/10.7554/eLife.06980.001





Frame (fraction) alignment / dose-weighting

- Done as the first step of processing
- Both use “raw data” – dose-fractionated movies
- Both done in Fourier space
  - image shift => phase shift – subpixel accuracy
  - except image deformation using 2nd order surface (can be combined with anisotropic magnification correction)
- Done usually in a single procedure
- Output:
  - aligned/dose-weighted micrographs
- Optional output:
  - aligned/non-dose-weighted micrographs
  - Power-spectrum of non-dose-weighted micrographs

CTF estimation

$$CTF = sin(-\pi \Delta z \lambda k^2 + \frac{\pi C_s \lambda^3 k^4}{2})$$

Spatial frequency

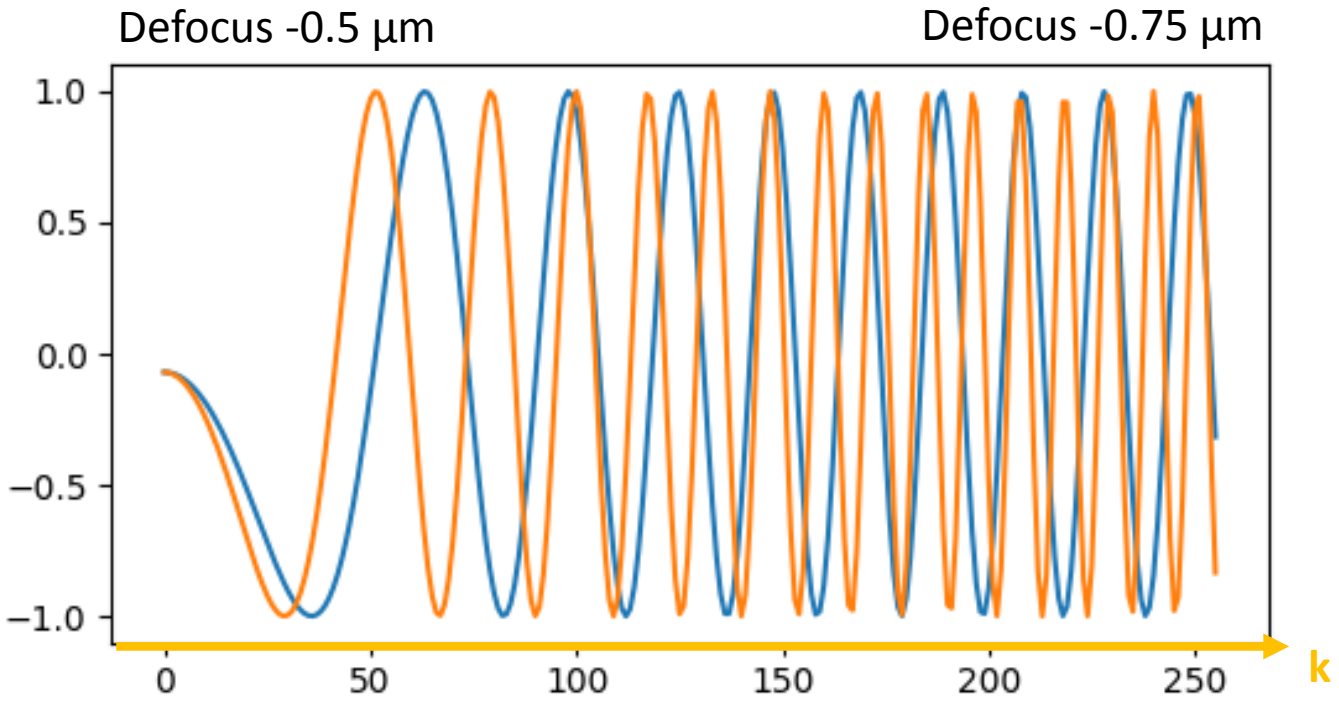
defocus

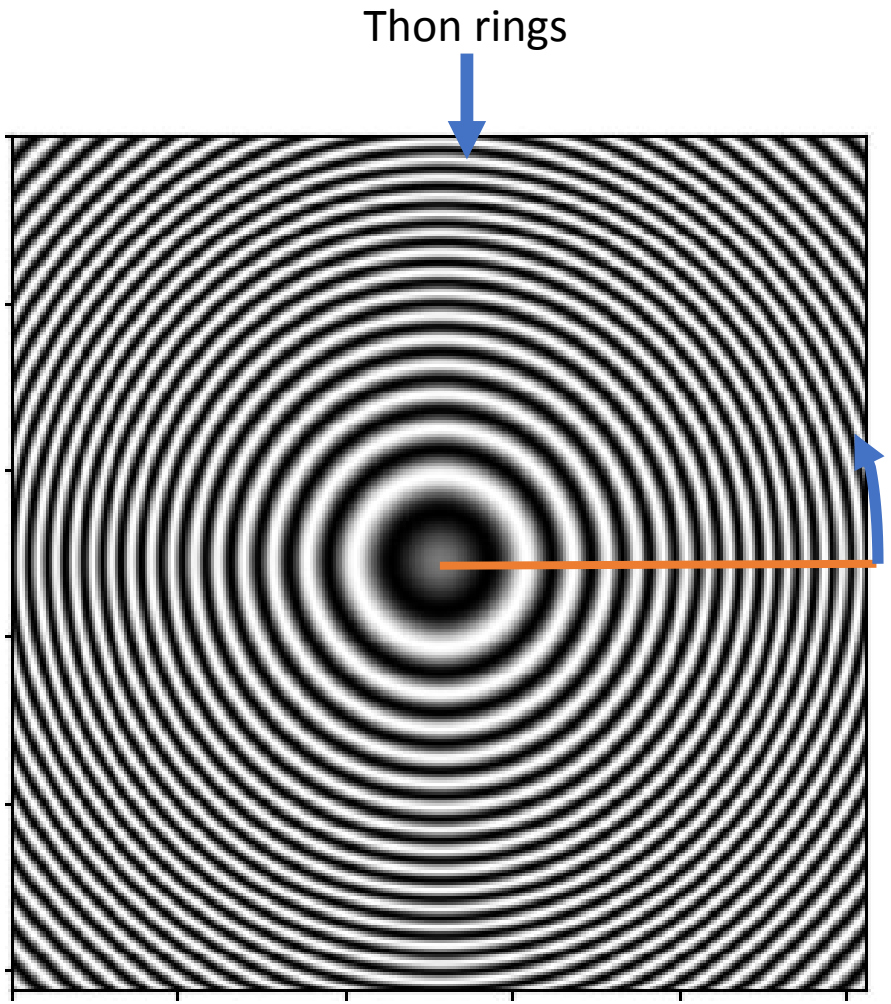
wavelength (e^-)

spherical aberration

- Contrast transfer function
  - Constants
    - Defocus – at a defined defocus the CTF curve has the same shape
    - e^- wavelength – dependent on acceleration voltage
    - Spherical aberration – property of the microscope lens (C_s value)
    - π – 3.1415
  - Variables
    - Spatial frequency
- EM images are convolved by the PSF (multiplied by CTF)
  - we can observe the contrast oscillation in the Fourier transform of the image
- Estimation of astigmatism / angle of astigmatism
- CTF estimation of the dataset only estimates the defocus
  - Does not do any CTF correction (it's done later)

1D / 2D CTF

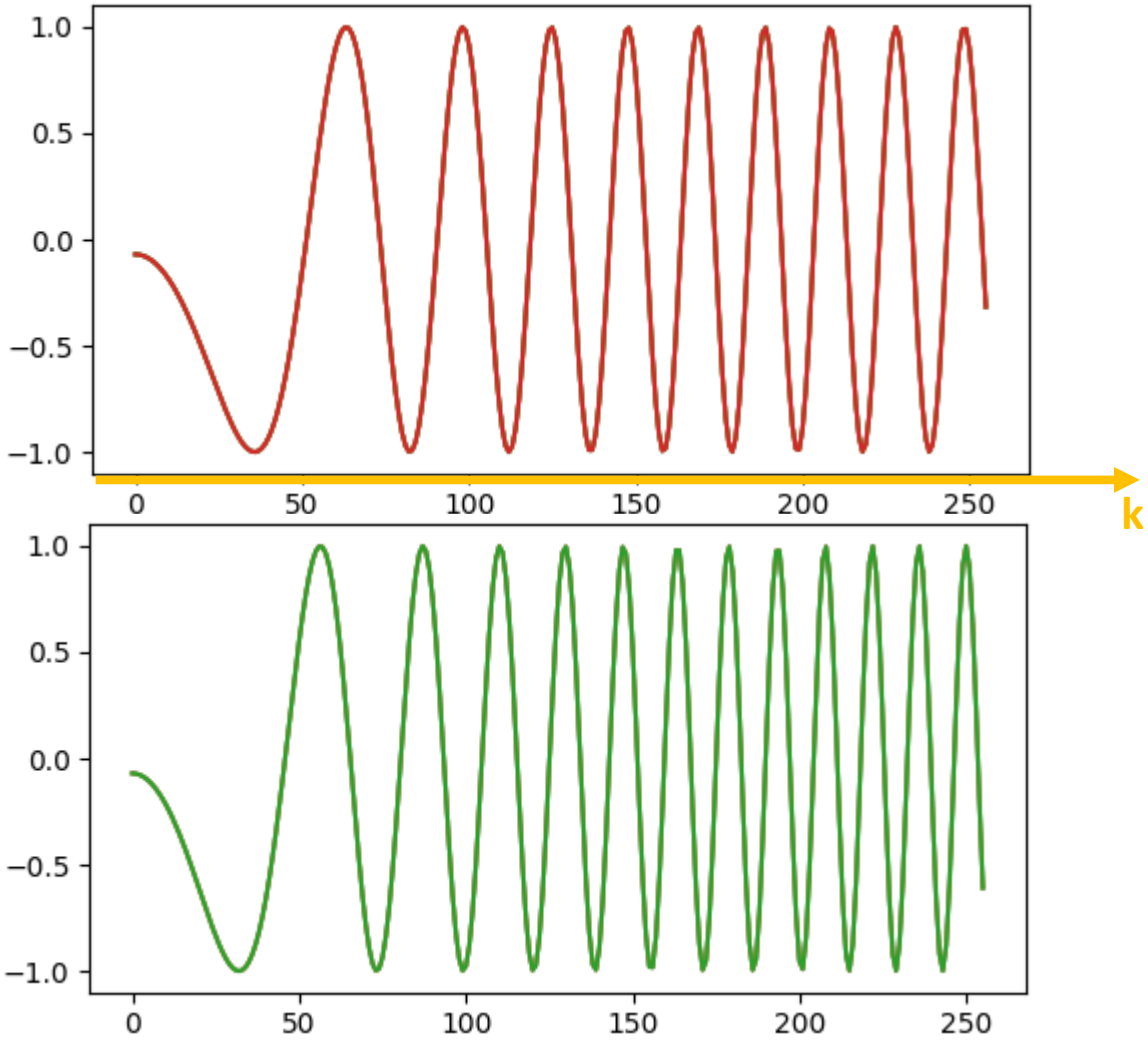




$$CTF = sin(-\pi \Delta z \lambda k^2 + \frac{\pi C_s \lambda^3 k^4}{2})$$

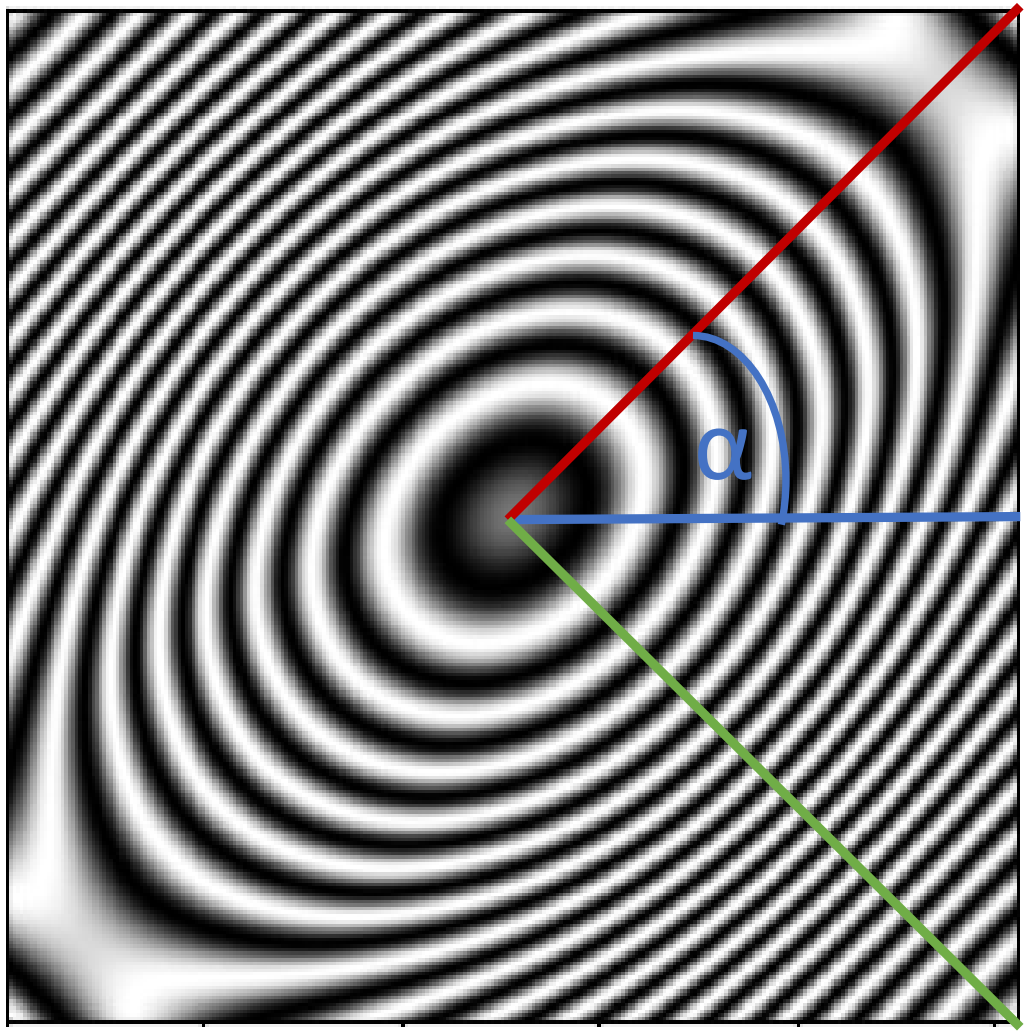
Astigmatism

Image has directional difference in defocus

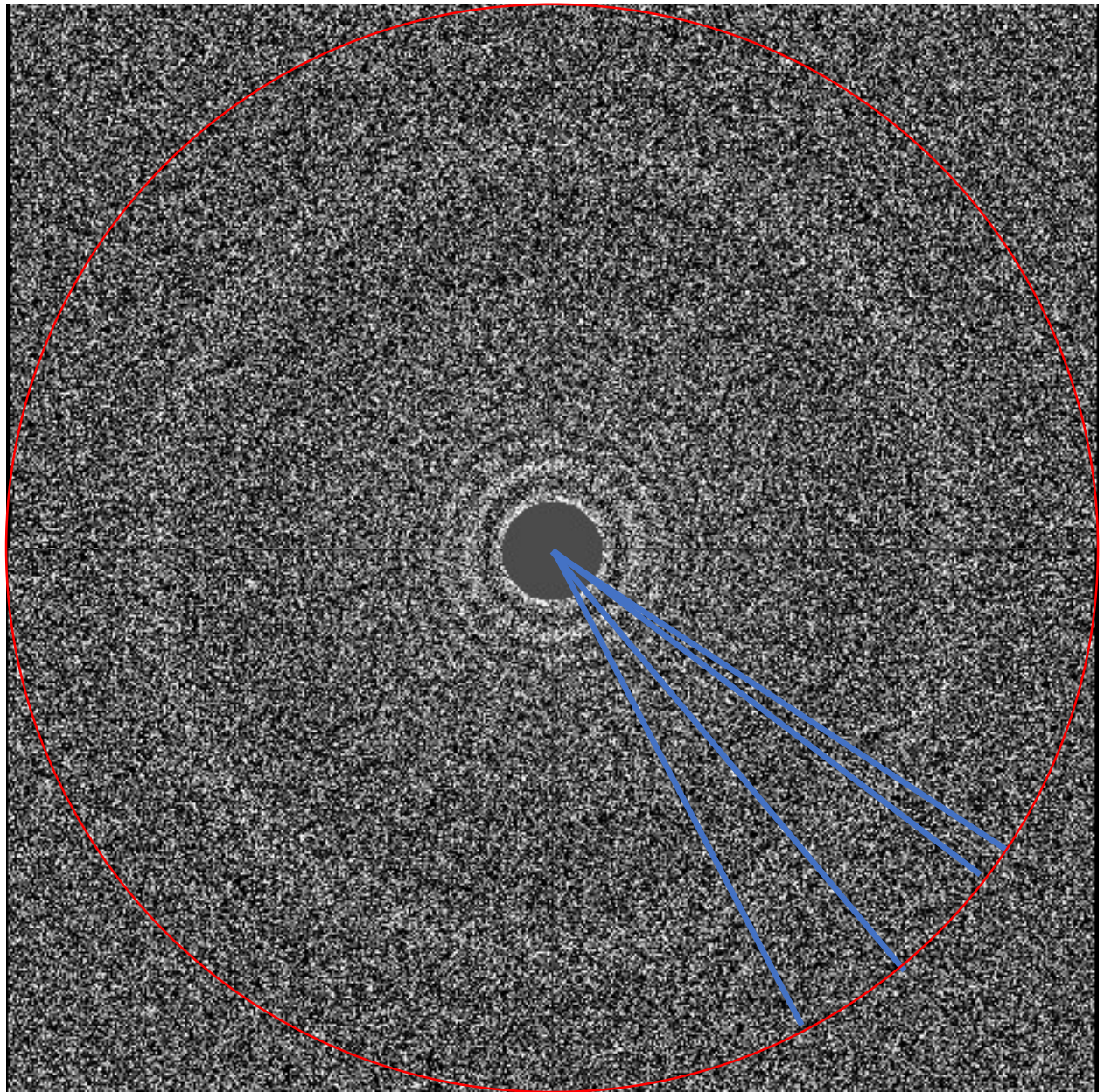


Defined by:

- DefocusU, DefocusV, angle of astigmatism
- Mean defocus, Δdefocus, angle of astigmatism



CTF estimation – DefocusU, DefocusV, Angle of Defocus

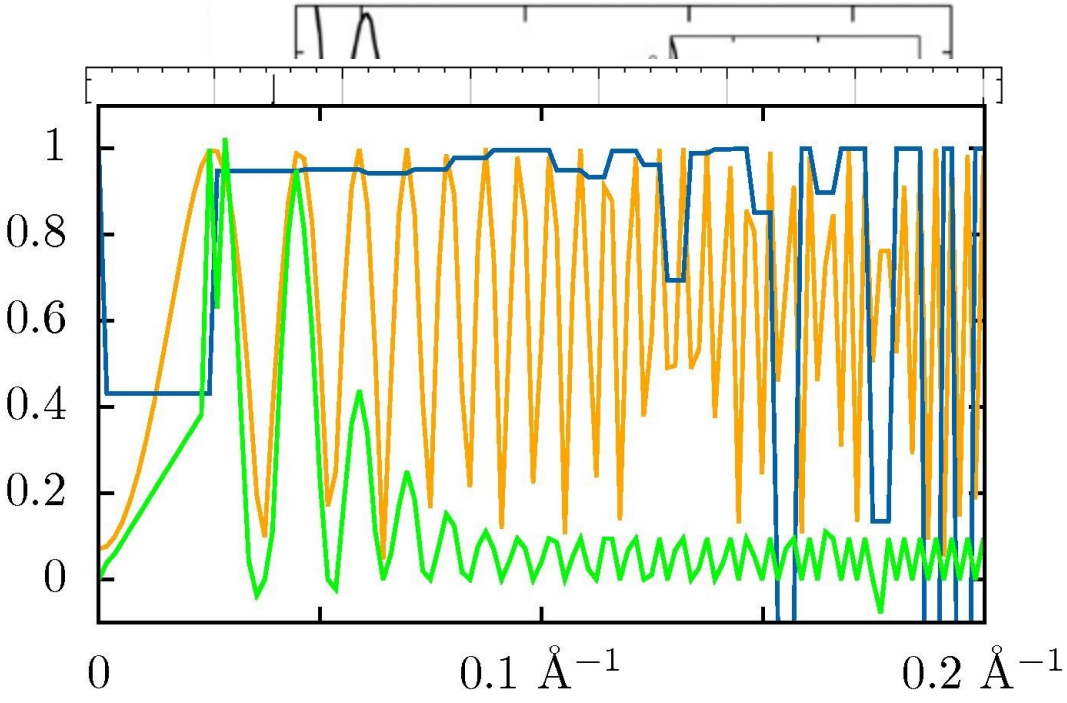


Read micrograph

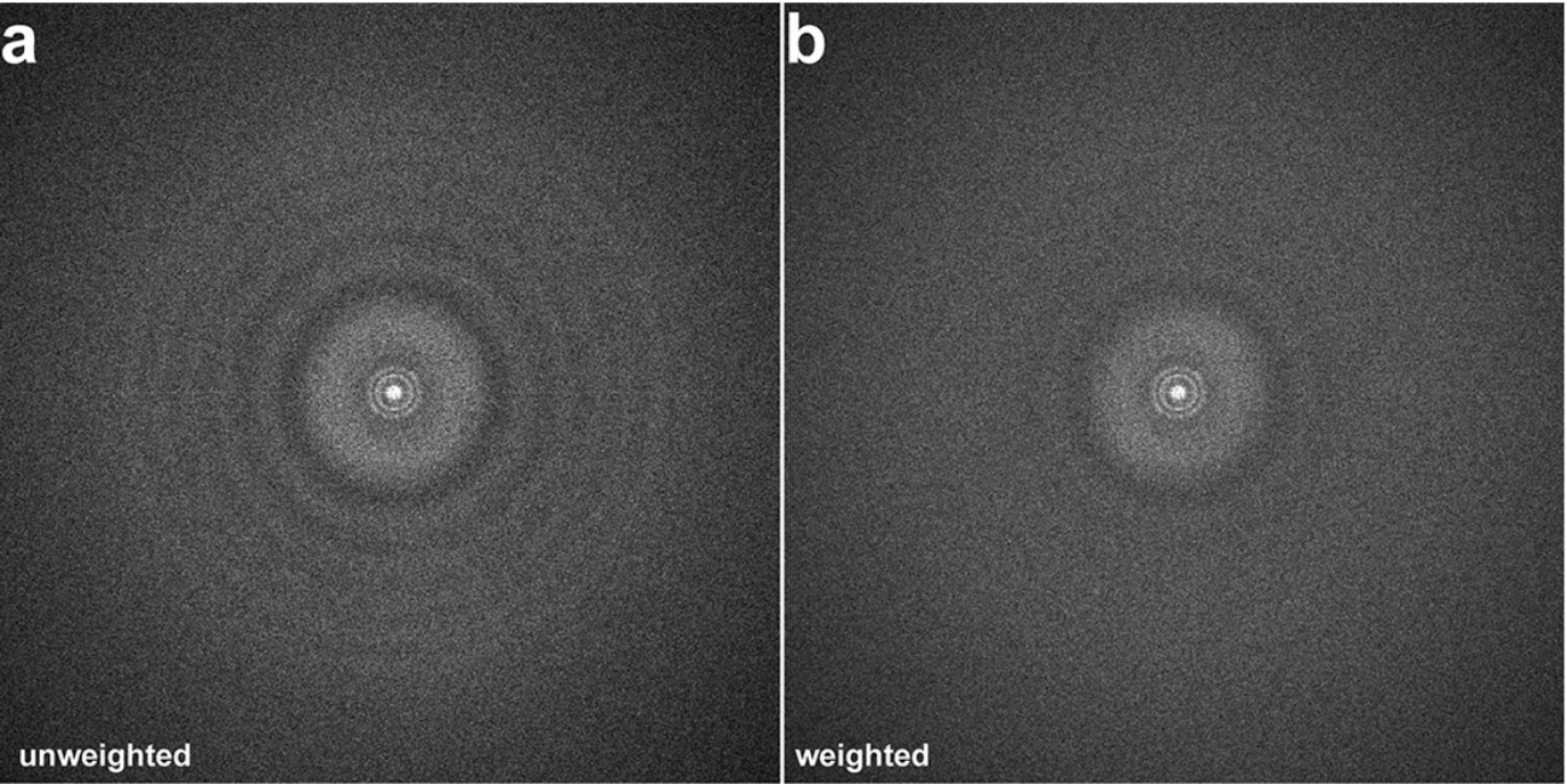
Fast Fourier Transform

Background removal in Fourier space

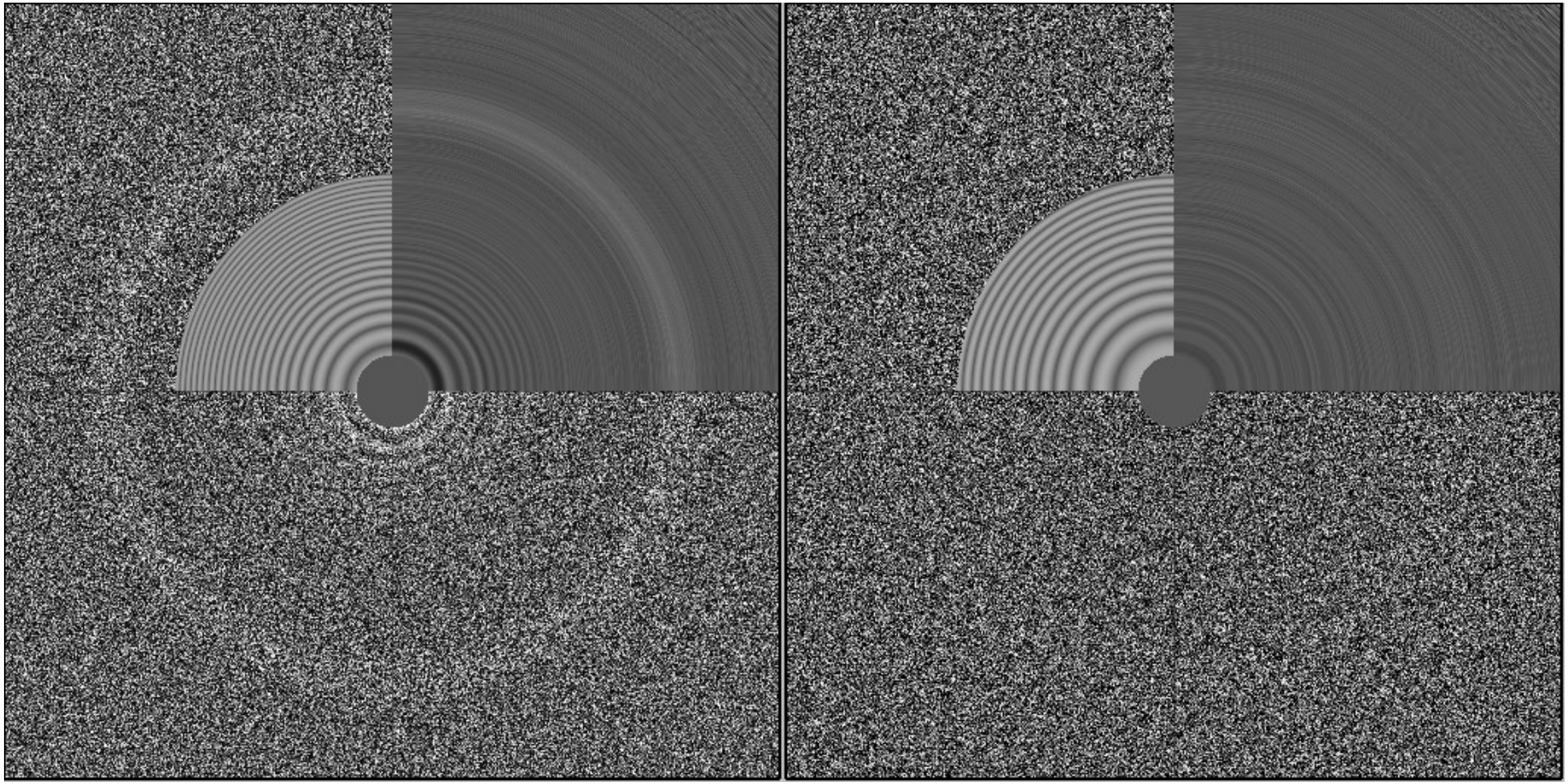
Rotational Average of 2D Spectrum



Dose weighted / non-dose weighted micrographs



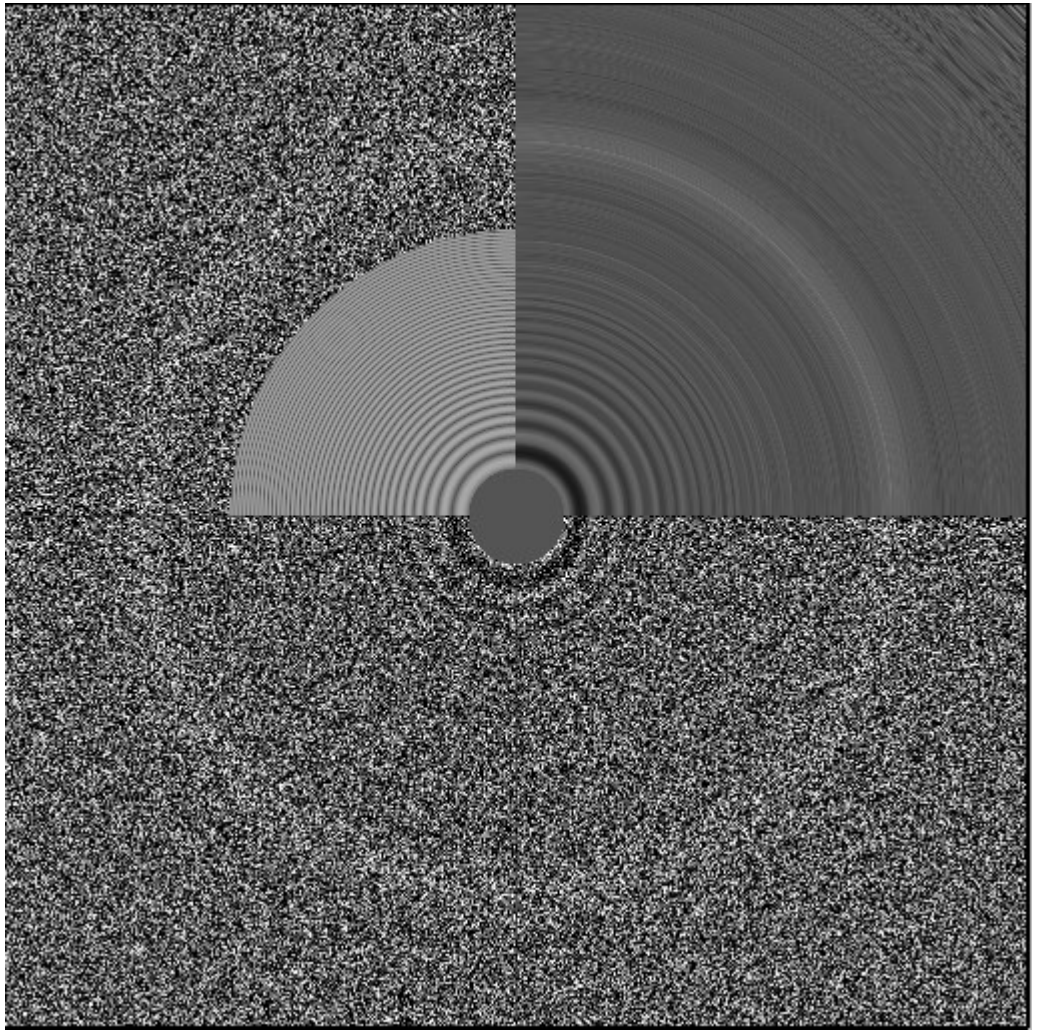
Check the result

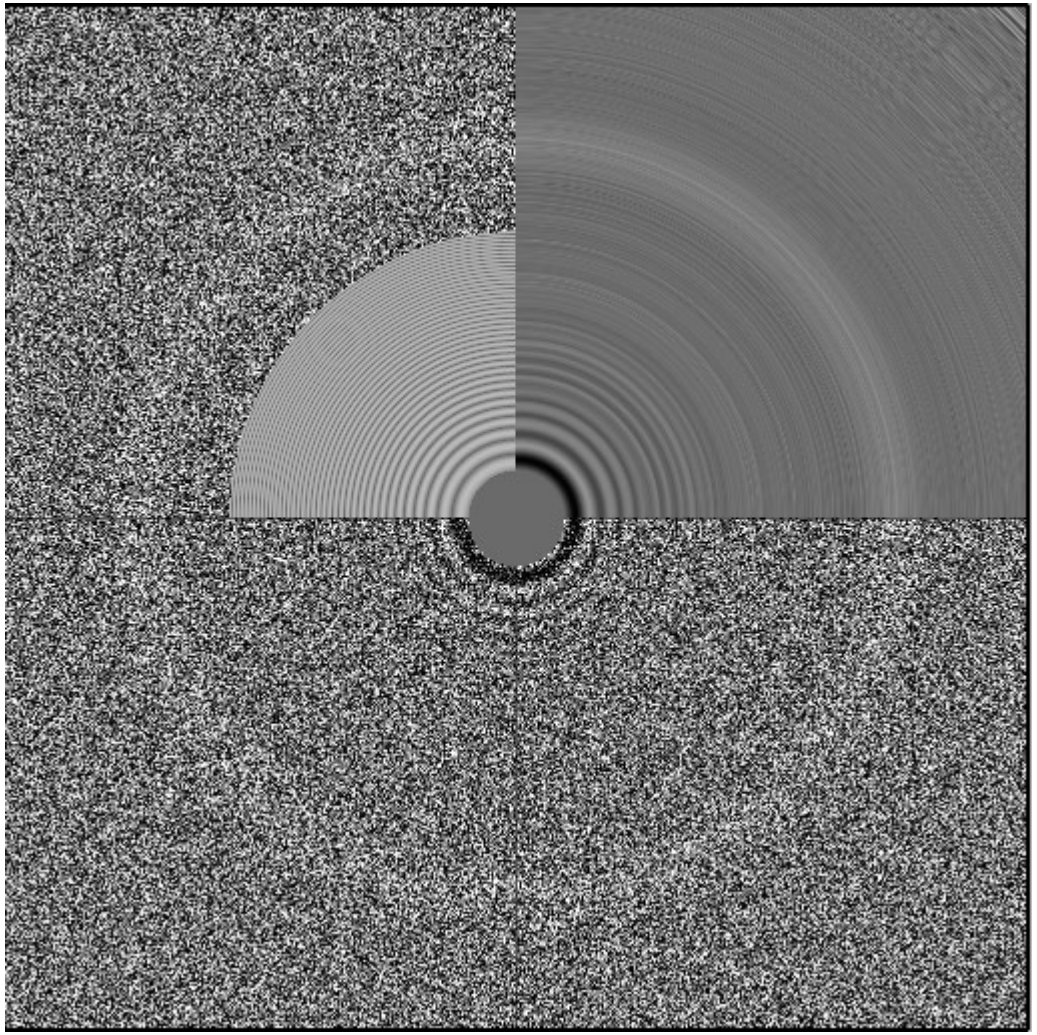


Good fit

Bad Fit (weak Thon rings)

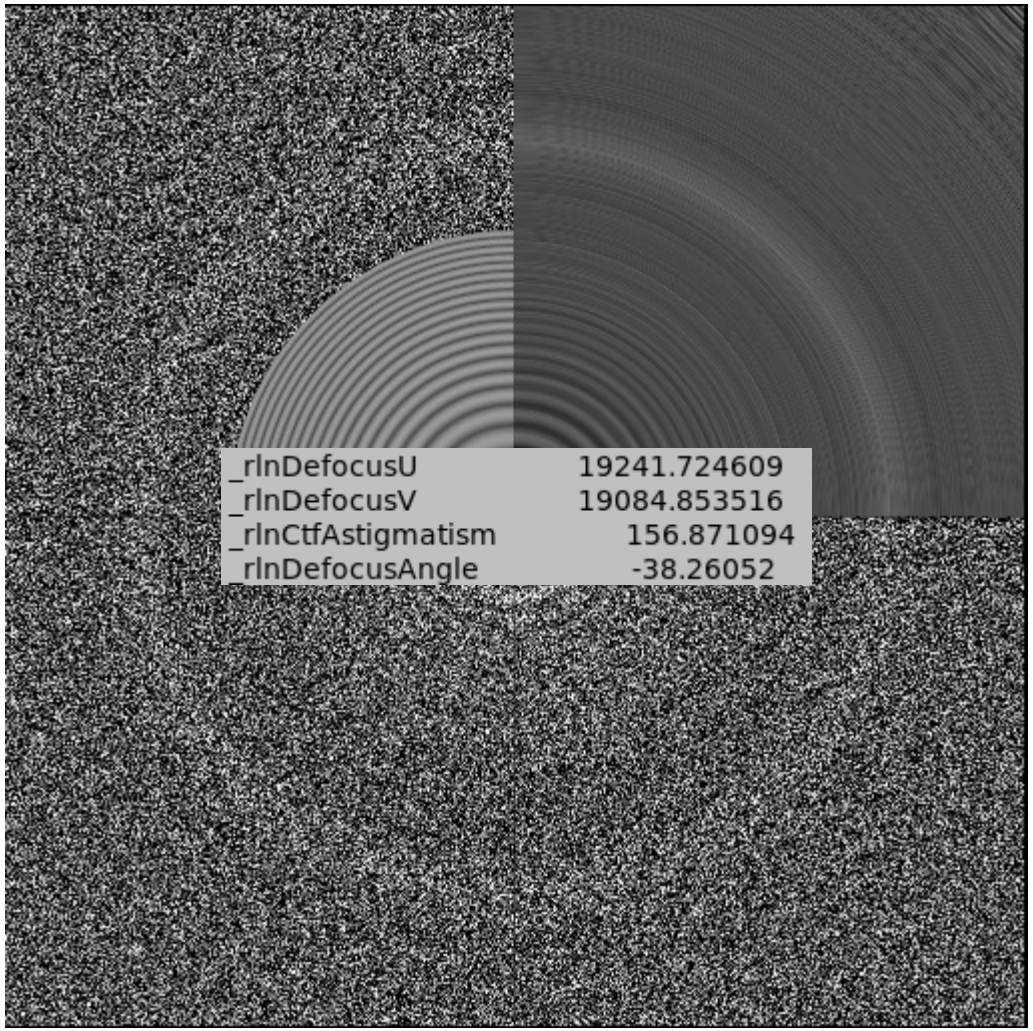
Nice spectrum bad fit



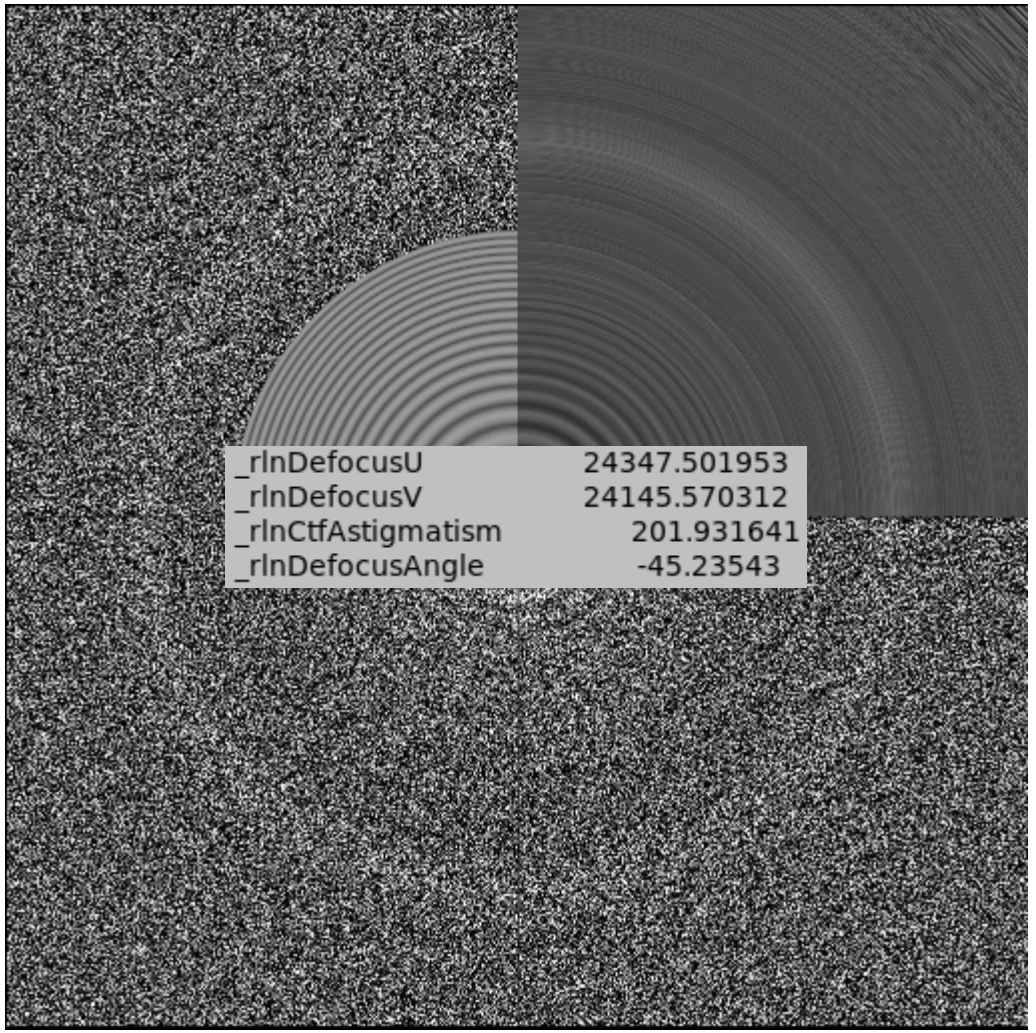


- Check the search range !

Beware of correct microscope parameters



200 kV, Cs 2.7 mm



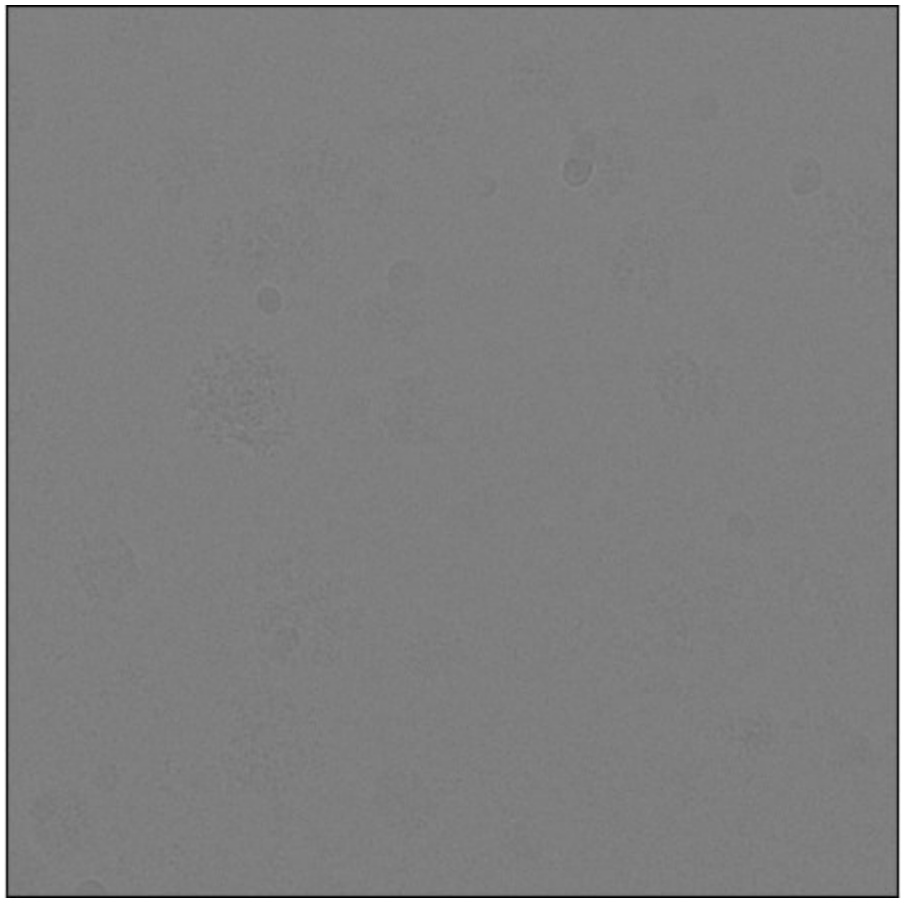
300 kV, Cs 2.7 mm

Particle selection

- Manual picking
  - Hand selected particles
  - Getting overview about the dataset
- Automated selected particles
  - Auto-picking by particle diameter definition
  - Auto-picking by template
  - Auto-picking using trained convolutional neural networks (CNN)
- False positive particles !!!
  - No perfect picking method
  - Aim to minimalize (not to completely avoid)

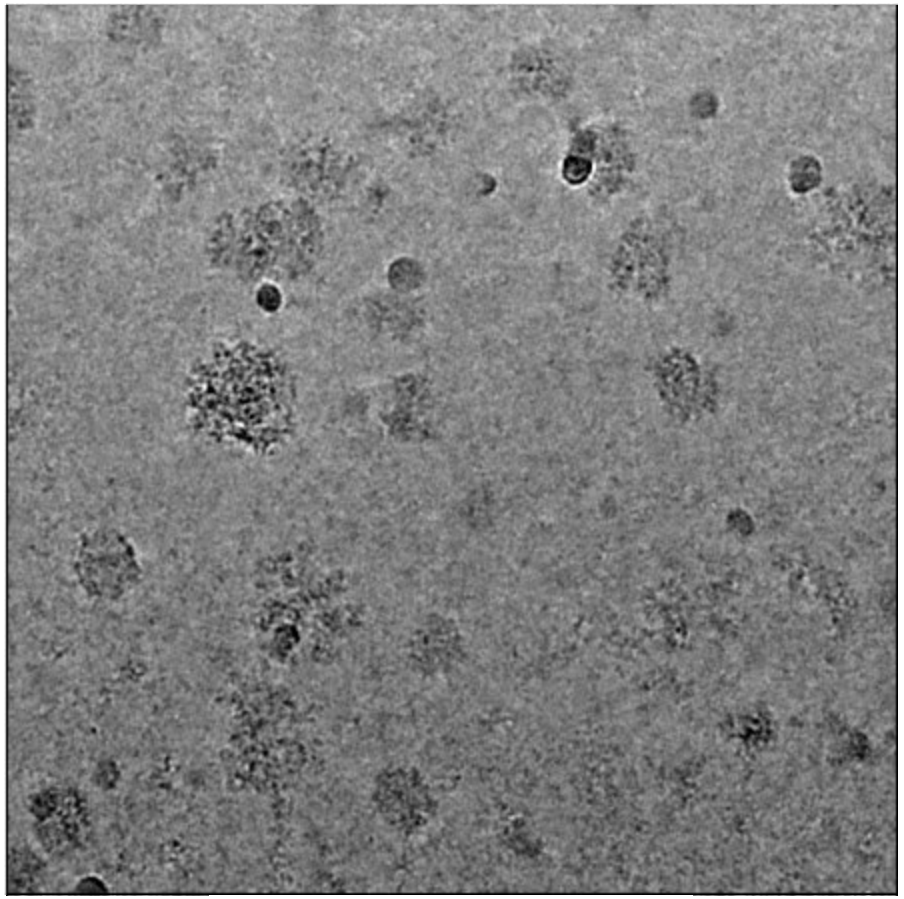
Manual picking

- Precision (not well centered particles)
- Reliability (human factor, possibility of targeted selection)
- Time consuming



Unfiltered

- Input for template-based auto-picking and CNN training
- Visibility of particles (ice thickness, defocus)
- Image filtering (lowpass, denoising)

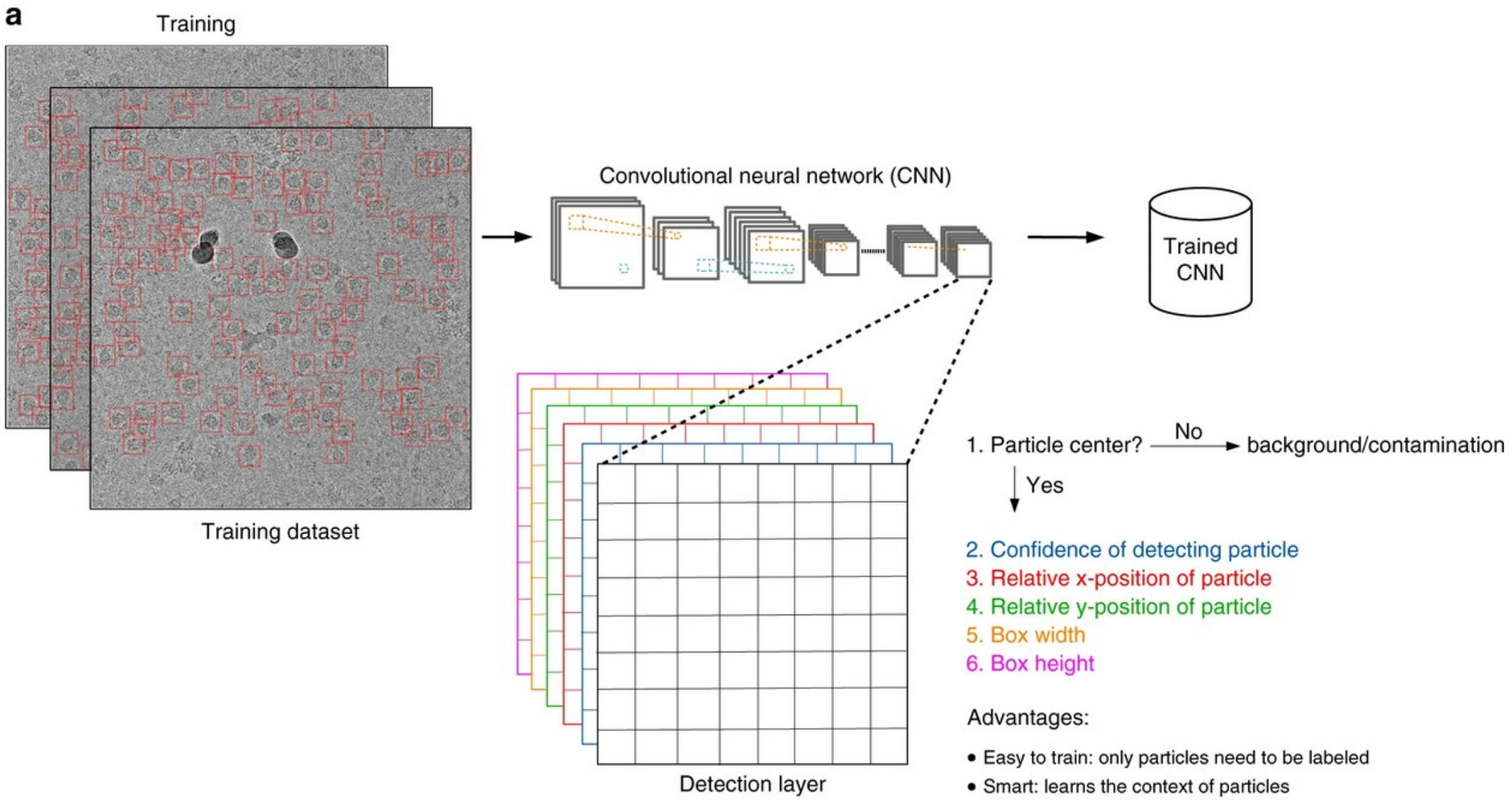


Noise2Noise denoised

Size- and template-based methods

- Size-based methods
  - LoG filter – Laplacian of Gaussian
    - blob detection algorithm
    - defining kernel (particle) size
    - general picking that after 2D classification can produce classes for templates
- Template-based methods
  - Need suitable templates
    - 2D class averages of manually selected particles
    - 2D projections of known 3D structure (e.g. ribosomes, viruses, general shapes)
  - Correlation based methods (real-space, reciprocal-space)
  - Maximum-likelihood version (RELION autopick)
  - Output need to be properly thresholded (many false positives)
  - Computational expensive (correlate the template and its planar rotations)

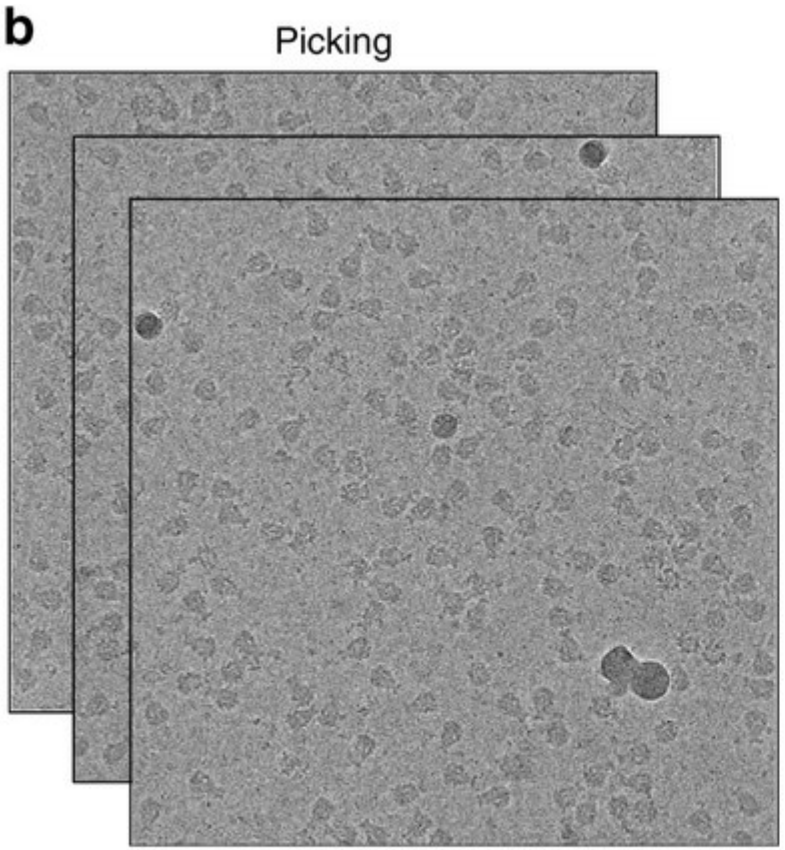
Trained convolutional neural networks



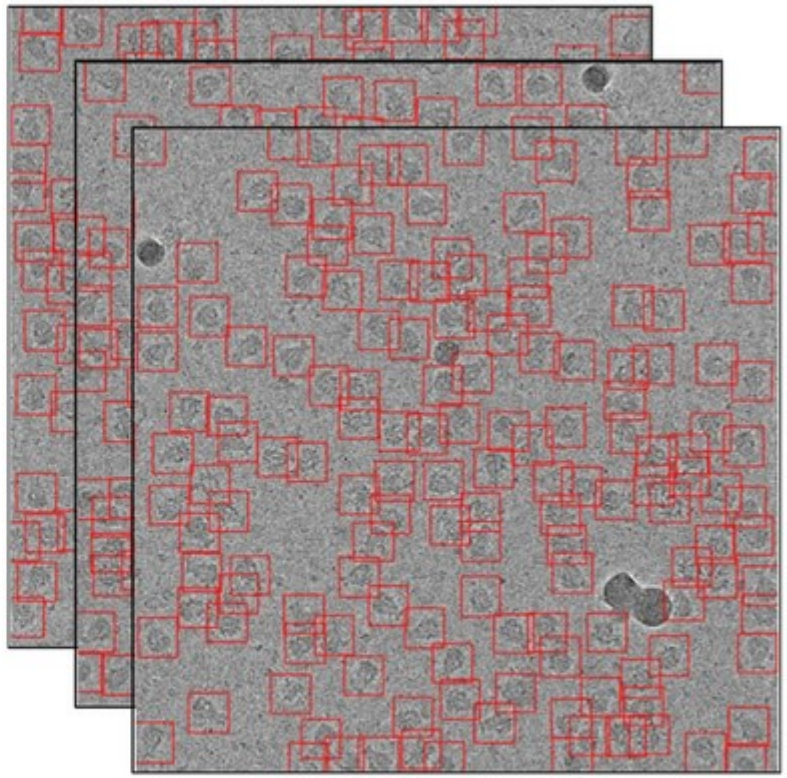
Wagner et al., 2019

https://doi.org/10.1038/s42003-019-0437-z

Particle position prediction by CNN



Availability of generally trained CNN models (out of the box solution)



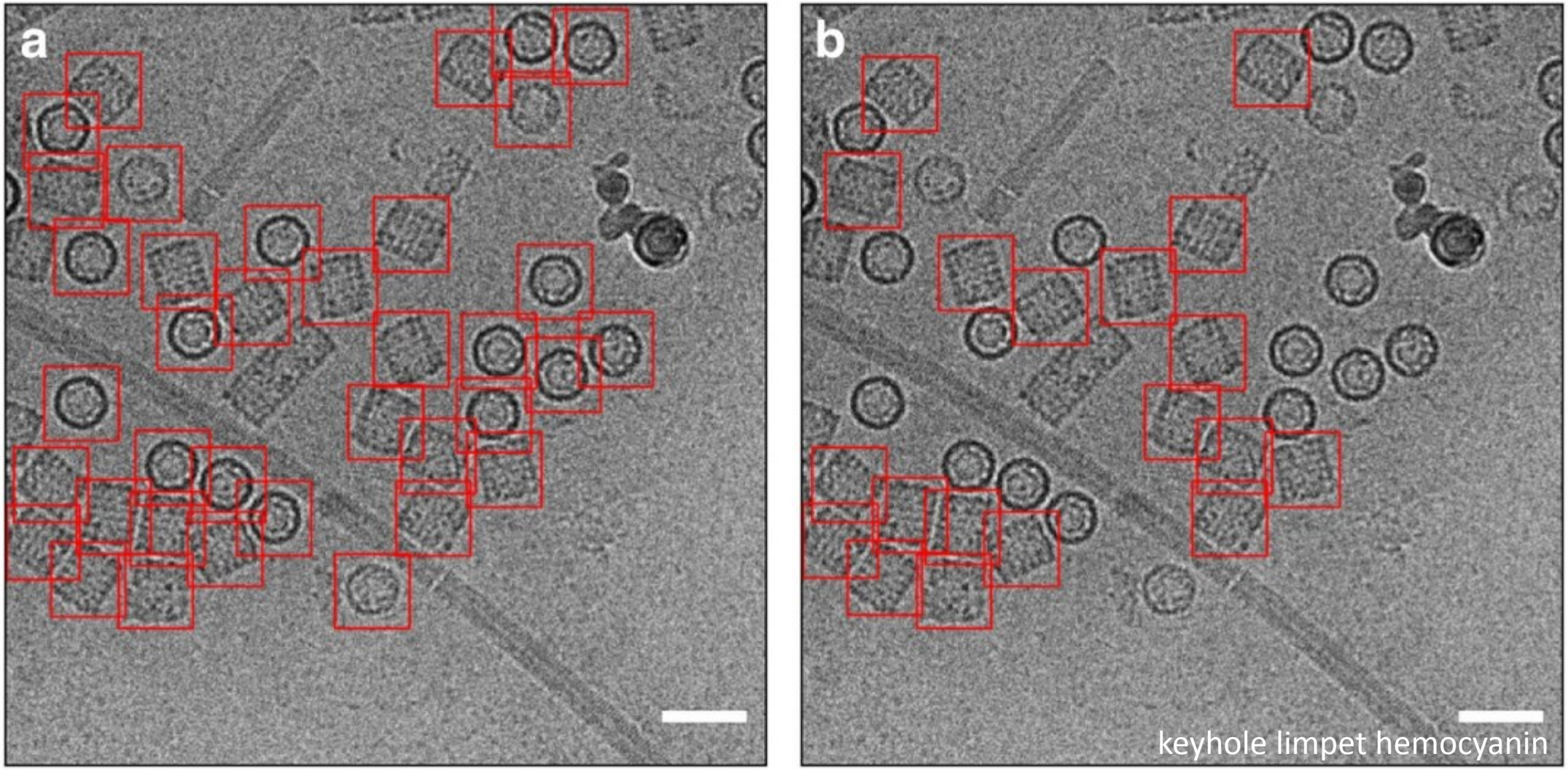
Advantage:

- Very fast: Picks up to 5 micrographs per second
- Outperforms sliding window approach

Wagner et al., 2019

https://doi.org/10.1038/s42003-019-0437-z

CNN can be specifically trained



Trained on top and side views

Trained on side views only

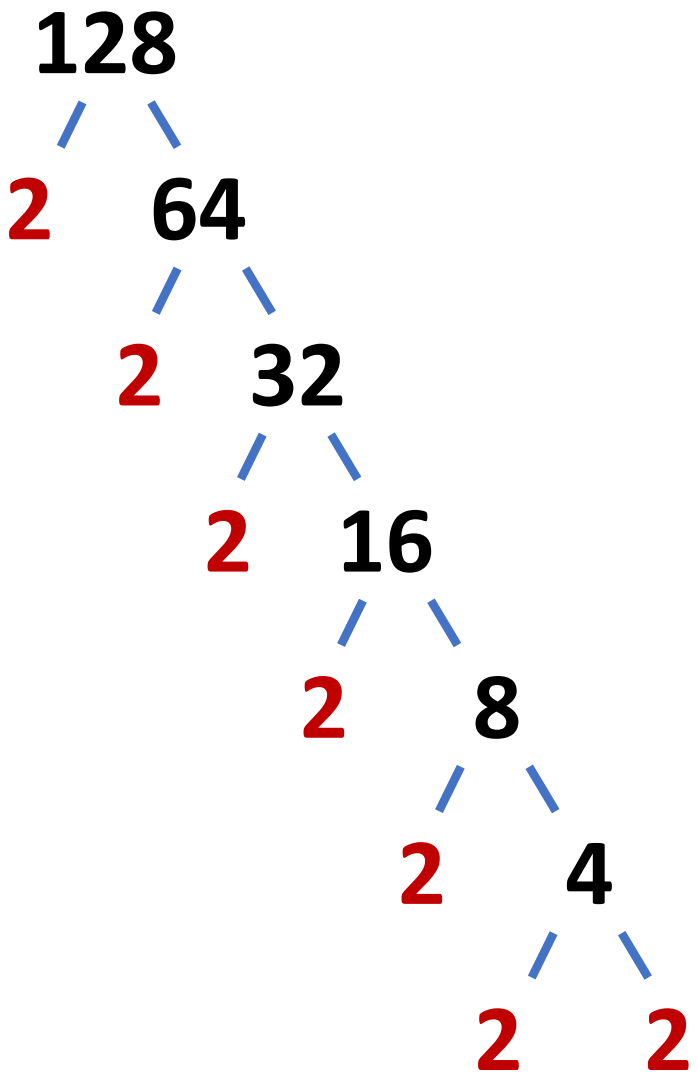
Wagner et al., 2019

https://doi.org/10.1038/s42003-019-0437-z

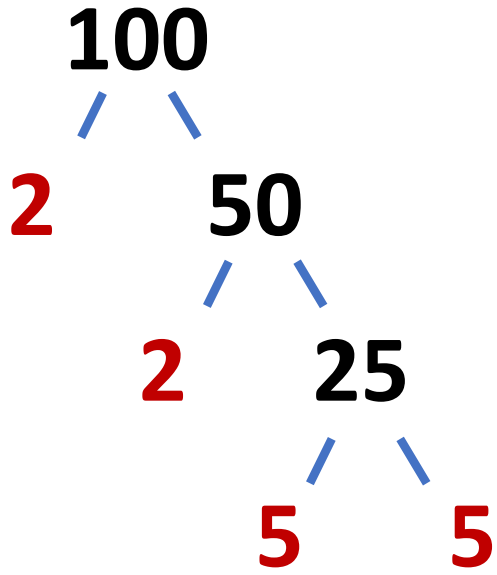
Particle extraction – box size

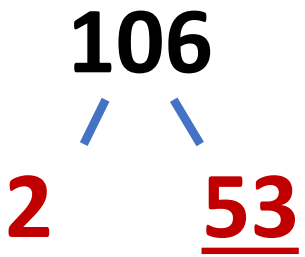
- Box size
  - Affected by particle size
  - Choosing proper value for computation
  - 1.3-2.0x bigger than the particle size
    - Why do we need larger box size ?
    - Background (noise) estimation
    - CTF => signal delocalization
- Inverting contrast
  - White particles on dark background
  - Ensures signal maximalization (not minimalization)
- Normalization
- Scaling/resampling (binning)

Box Size - Prime factorization



Box size must be an even number !





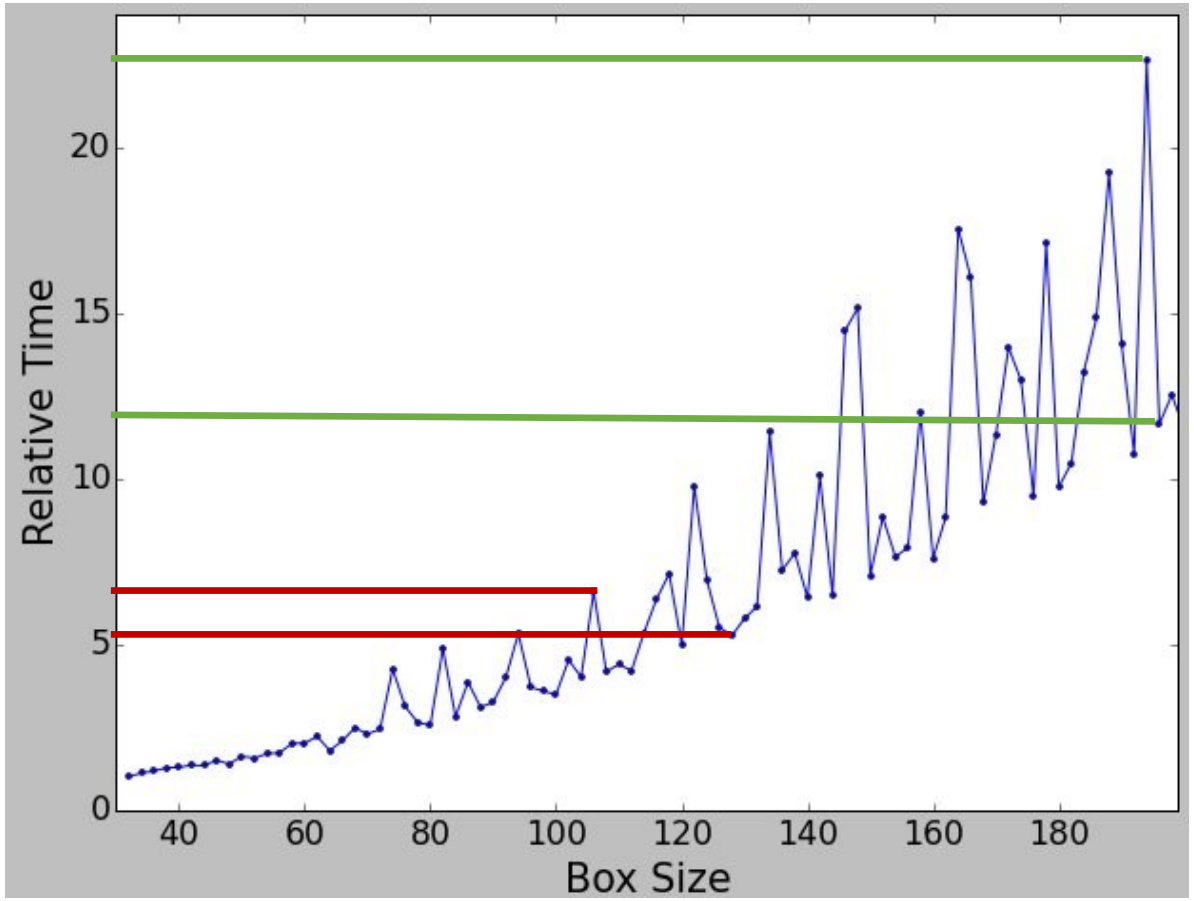
Not all even numbers perform equally well for computation

For **Fast Fourier Transform (FFT)** algorithm:
The **biggest prime factor** should be **as small as possible** !

Good box sizes

[redacted]

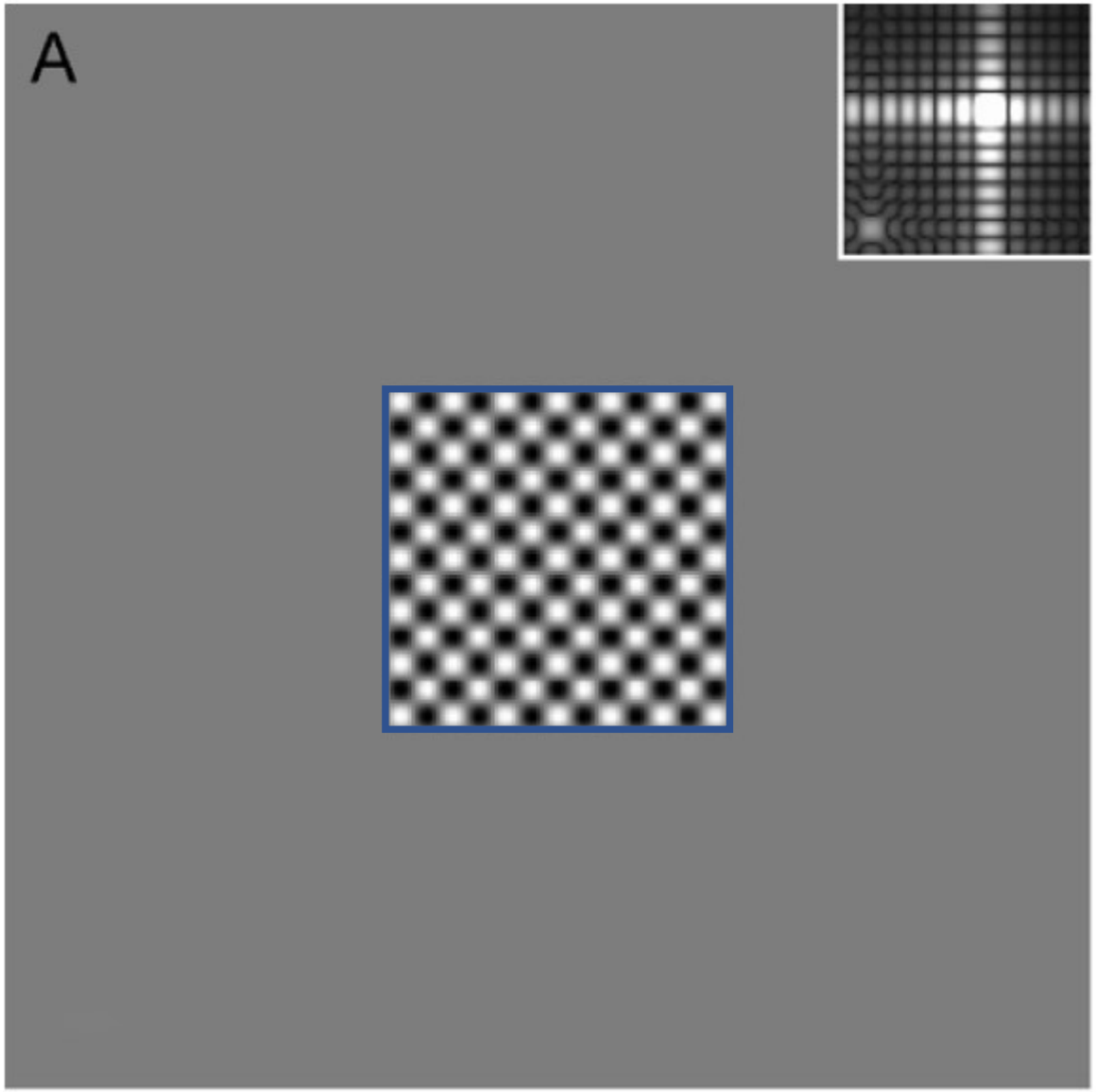
Small difference in size makes big difference in processing time !



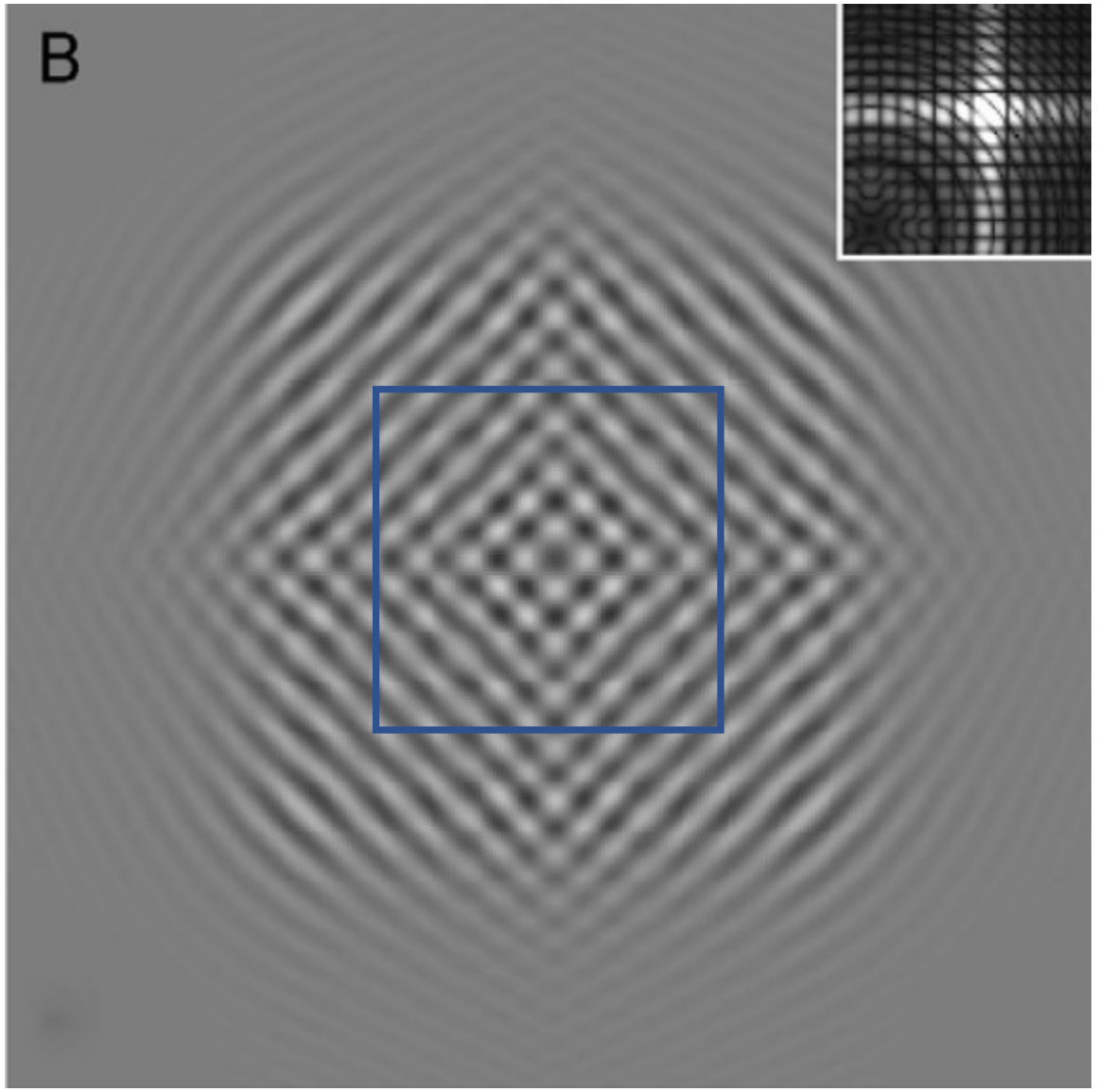
https://blake.bcm.edu/emanwiki/EMAN2/BoxSize

24, 32, 36, 40, 44, 48, 52, 56, 60, 64, 72, 84, 96, 100, 104, 112, 120, 128, 132, 140, 168, 180, 192, 196, 208, 216, 220, 224, 240, 256, 260, 288, 300, 320, 352, 360, 384, 416, 440, 448, 480, 512, 540, 560, 576, 588, 600, 630, 640, 648, 672, 686, 700, 720, 750, 756, 768, 784, 800, 810, 840, 864, 882, 896, 900, 960, 972, 980, 1000, 1008, 1024

CTF signal delocalization



Original image

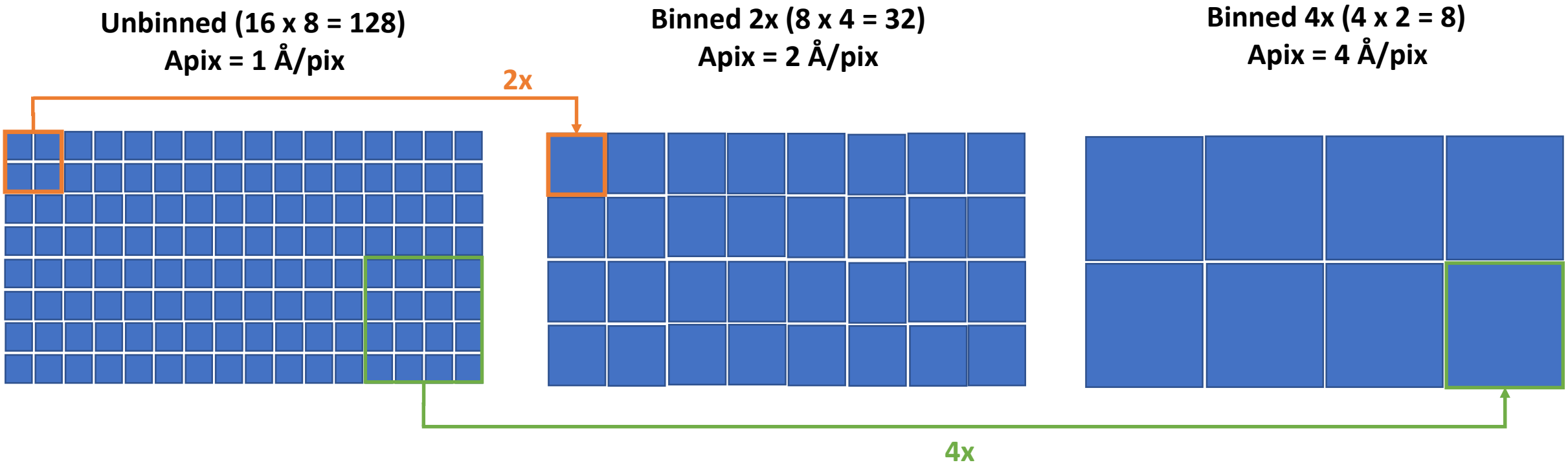


2 000 nm defocus applied

Binning

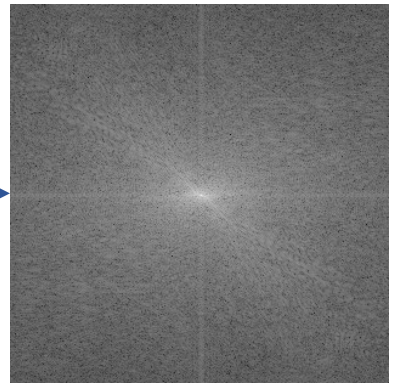
- **Scaling**
  - In **real-space by integer values**
    - Replacing the value of 2x2 (4x4 etc) pixel by their average
    - In image processing “Binnig 2x” means mathematically “Binning 2x2”
  - In **Fourier space by any real**
    - Performed as Fourier space cropping
    - Produce less artifacts

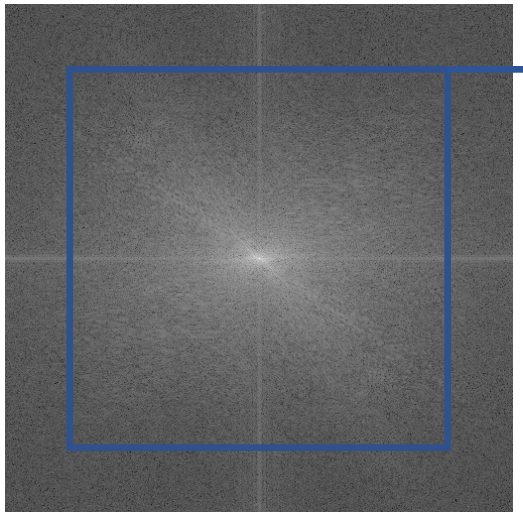
- **Properties of scaled images**
  - Decreasing pixel count
  - Decreasing resolution
  - Decreasing noise / increasing contrast (lowpass filter)
  - Decreasing file size
  - Decreasing computational demand
  - Increasing pixel size



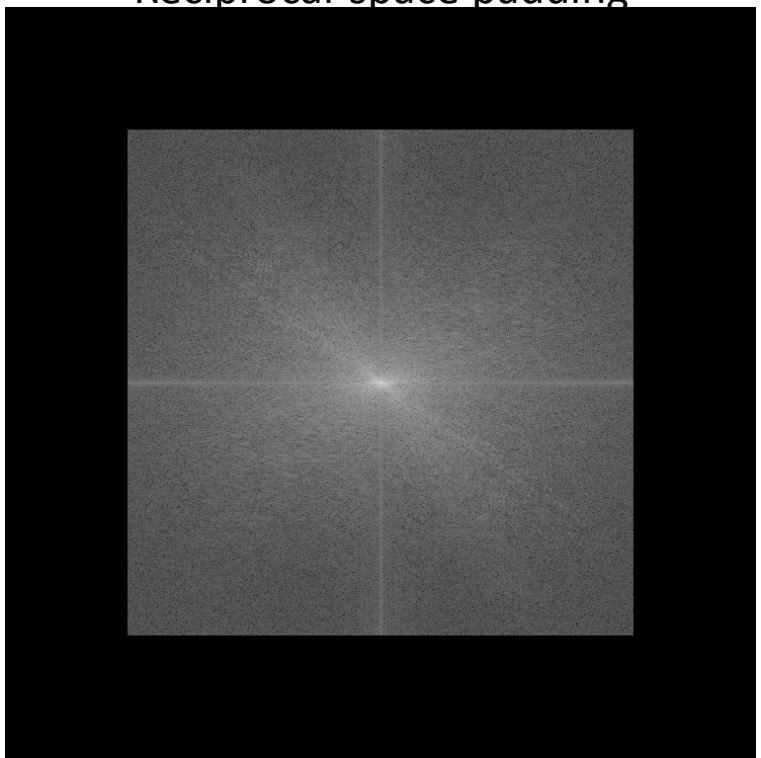
Fourier space cropping, padding

Reciprocal-space cropping





Reciprocal-space padding



Downscaling (~lowpass)

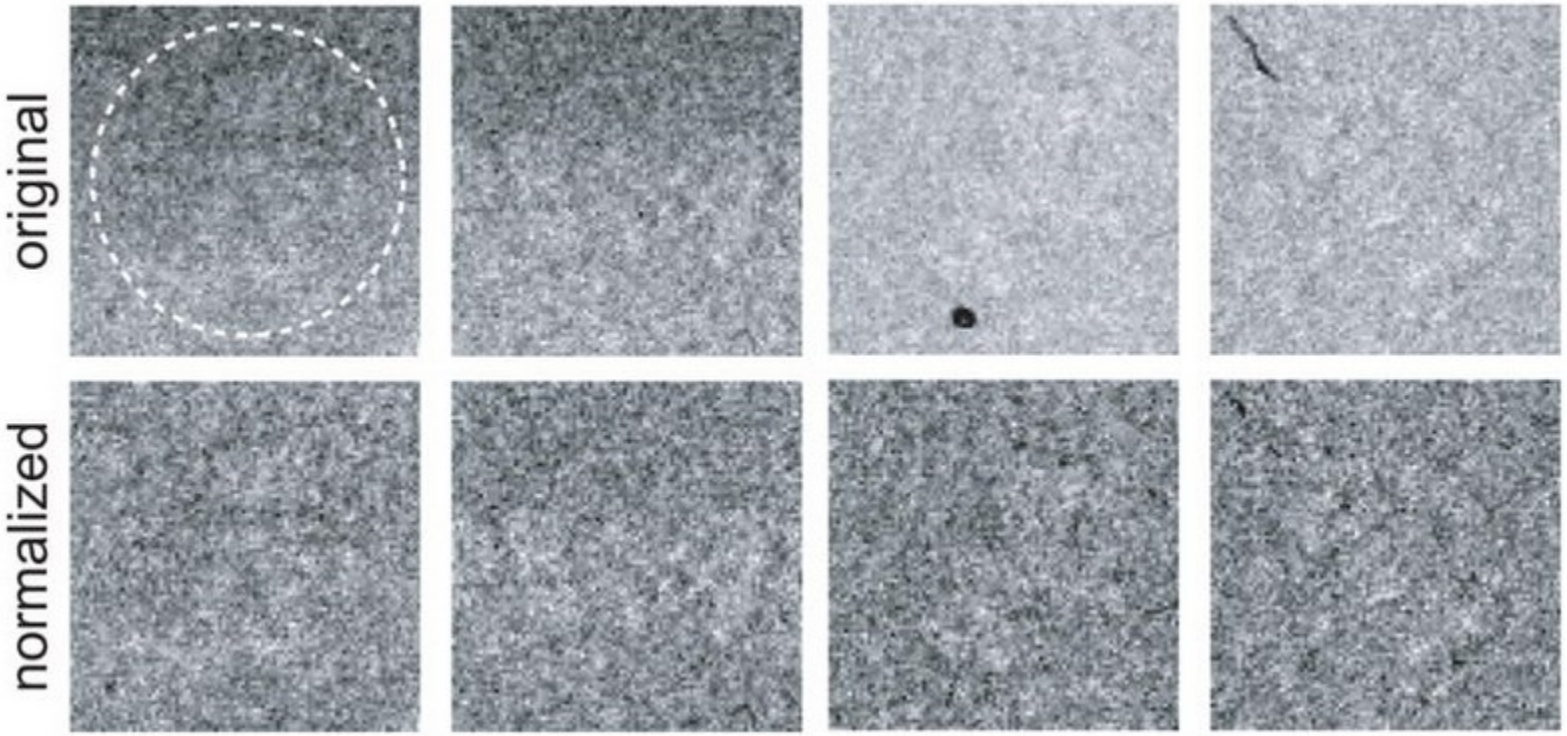
Upscaling (without adding information)

From Lecture 5

Image normalization

- Removing contrast differences between particles
  - Particles located in thin ice, thick ice
  - **Cross-correlation refinement schemes not sensitive to normalization** (normalized cross-correlation coefficients are invariant to additive or multiplicative factors)
  - **Maximum-likelihood techniques sensitive to proper normalization**
- All powers in the noise and in the signal are considered equal
  - Important for next step of particle alignment in Fourier space
- Different normalization formulae
  - Whole image normalization (weak aproach for non-spherical particles)
    - subtract the image means and to divide by the standard deviations => whole image has Avg=0, SD=1
  - Background (noise) normalization
    - area outside from particle mask considered to be background and is normalized as above
  - Fitting planar functions
    - Usually fitted on the background of the particle (good for removing gradients)
  - Applying high-pass filters
    - Introduce artifacts in Fourier space - **avoided** in maximum likelihood approaches
- Reason to have larger box size
  - part of the particle image serves for background (noise) estimation for normalization
- Particle size during extraction (defines the particle vs background)

Image normalization



Scheres 2010

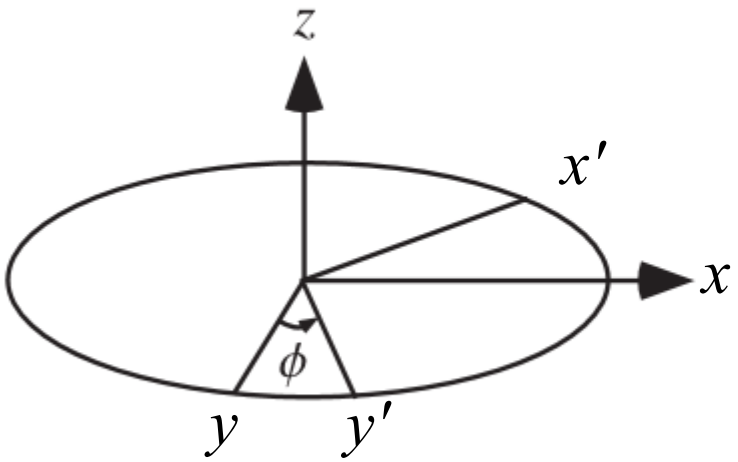
doi: 10.1016/S0076-6879(10)82012-9

Euler angle conventions (ZYZ)

Any rotation in 3D space can be described by three Euler angles:

- The first rotation is called ϕ (phi, rlnAngleRot) and is around the **Z-axis**.
- The second rotation is called θ (theta, rlnAngleTilt) and is around the new **Y-axis**.
- The third rotation is called ψ (psi, rlnAnglePsi) and is around the new **Z axis.**

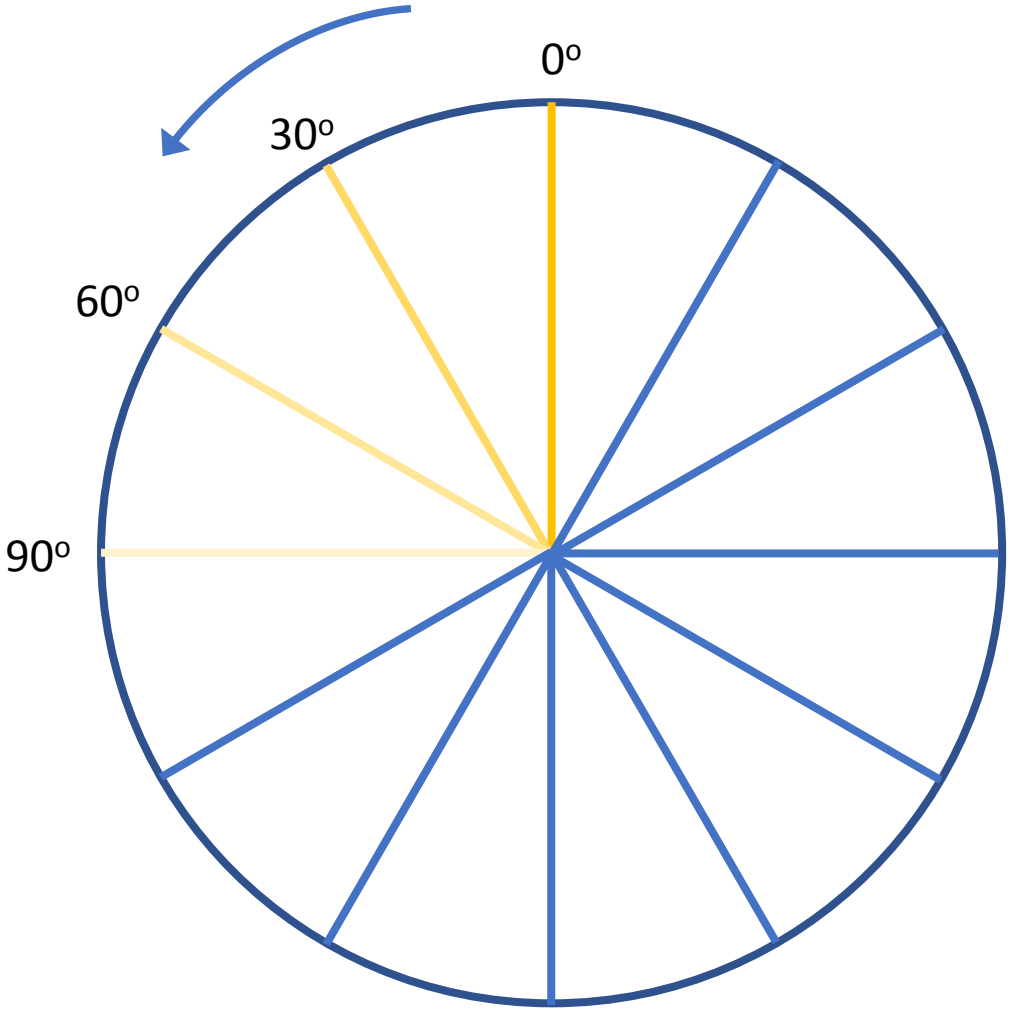
For in-plane rotations (2D), we keep the ϕ and θ zero and rotate the image by ψ angle!

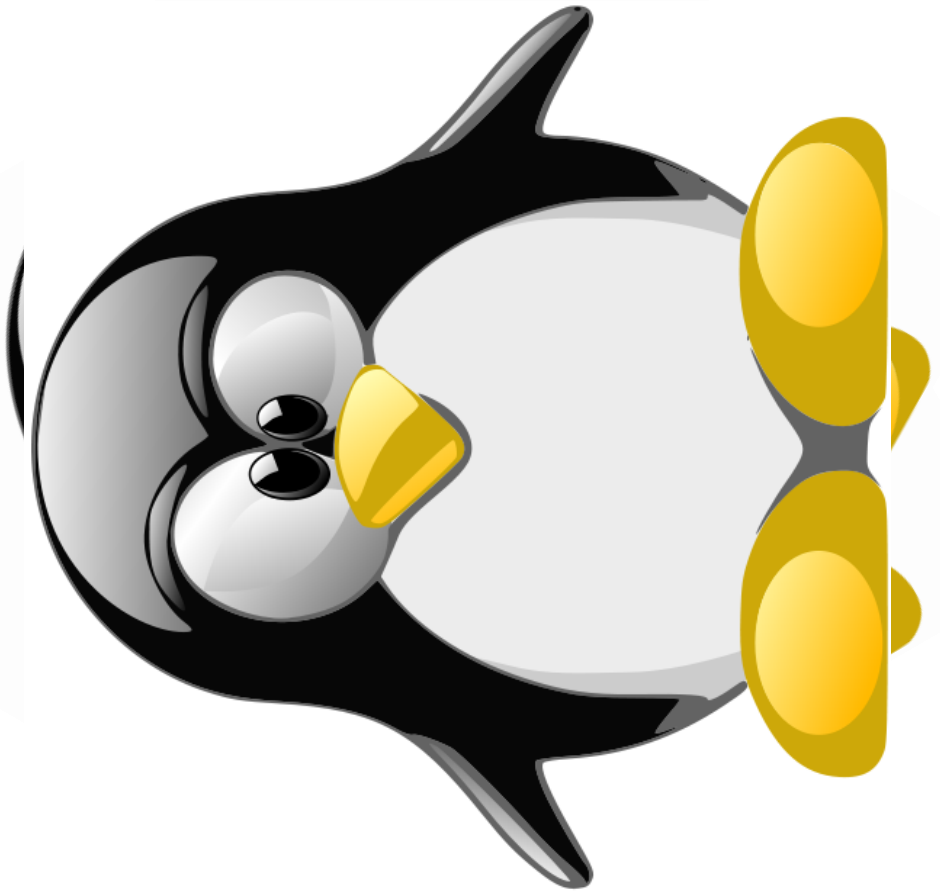


https://mathworld.wolfram.com/EulerAngles.html

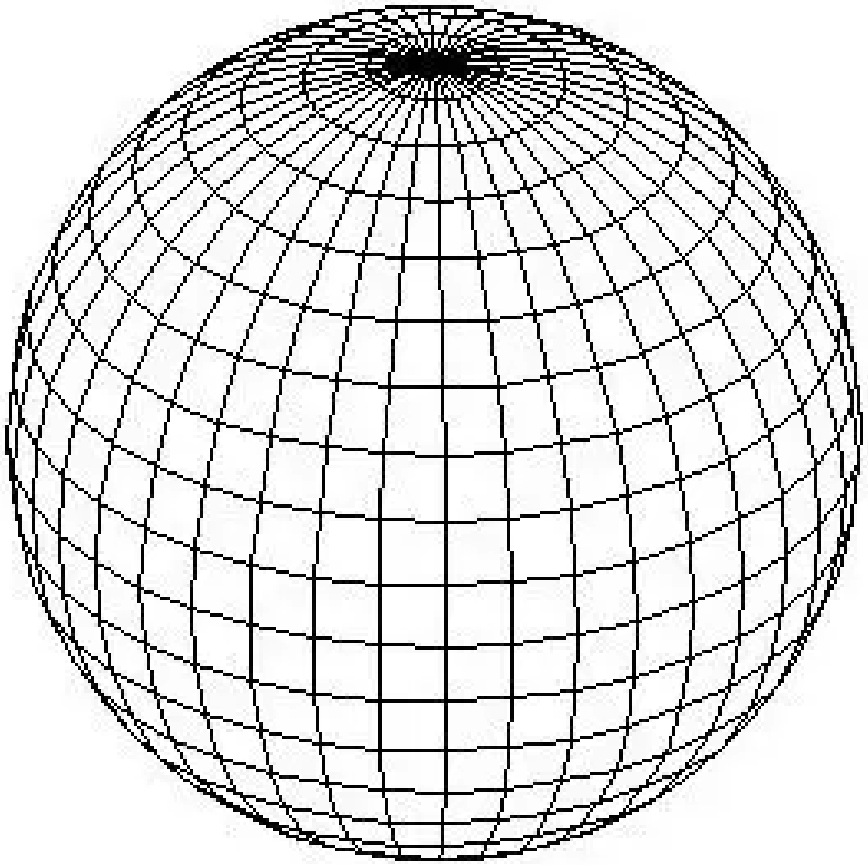
Uniform angular sampling – 2D

Uniform angular sampling in 2D: vary the ψ in <0, 2π> by given sampling rate (ster





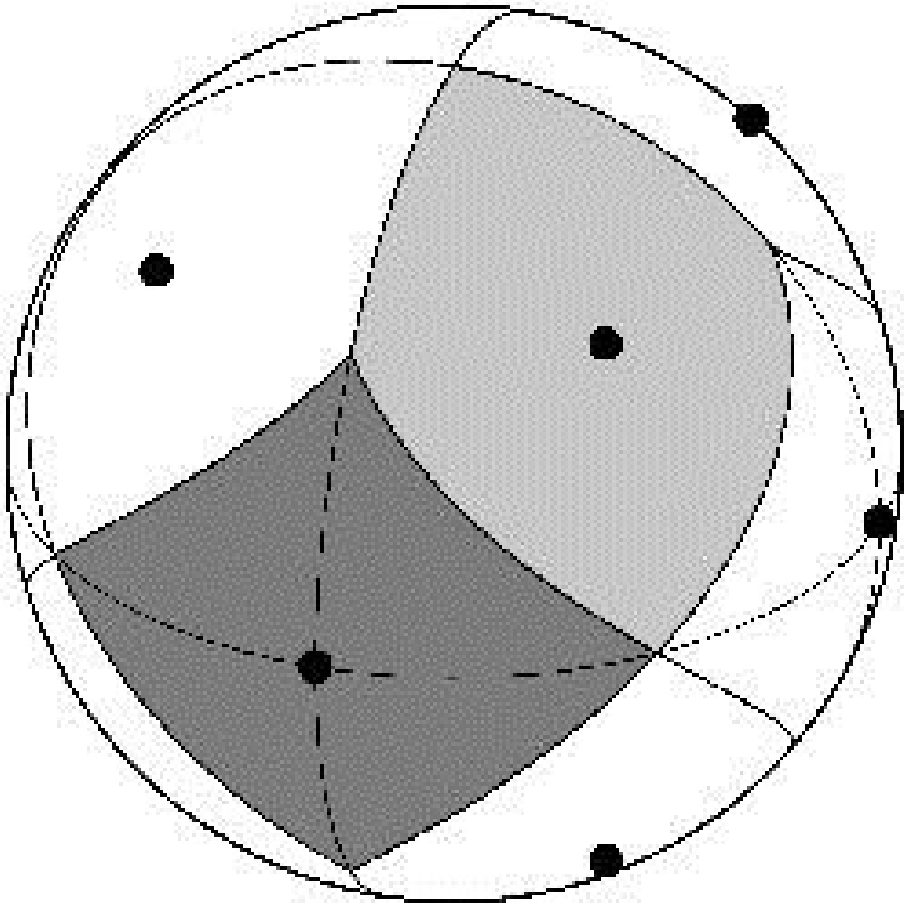
Uniform angular sampling – 3D



Uniform distribution of Euler ϕ, θ angles in range $<0, 2\pi>$

Non-even distribution (clustered towards the poles)

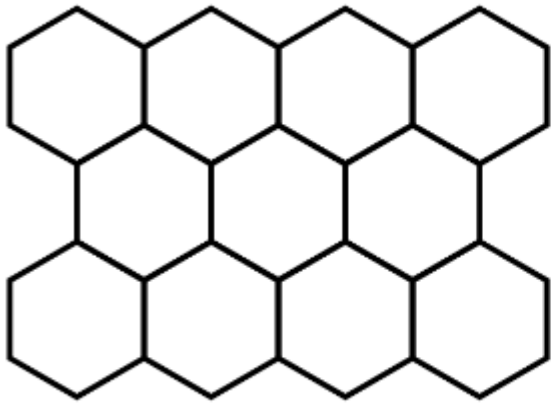
HEALPix

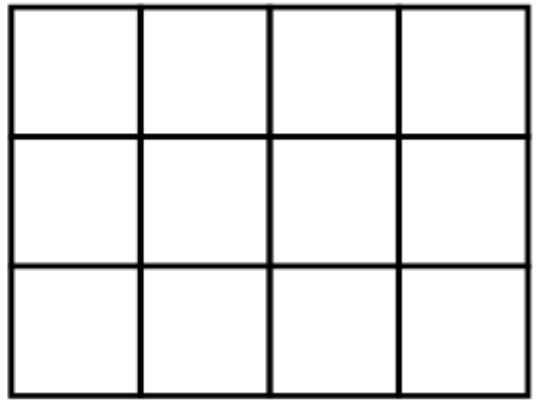


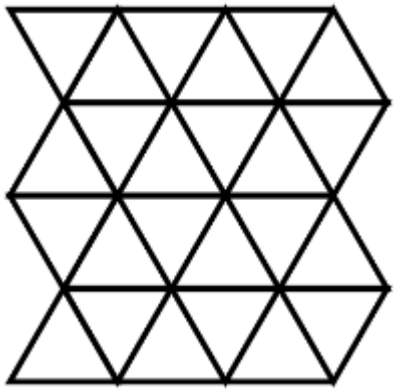
sphere is hierarchically tessellated into curvilinear quadrilaterals

https://healpix.sourceforge.io/

HEALPix - **H**ierarchical **E**qual **A**rea iso**L**atitude **Pix**elation

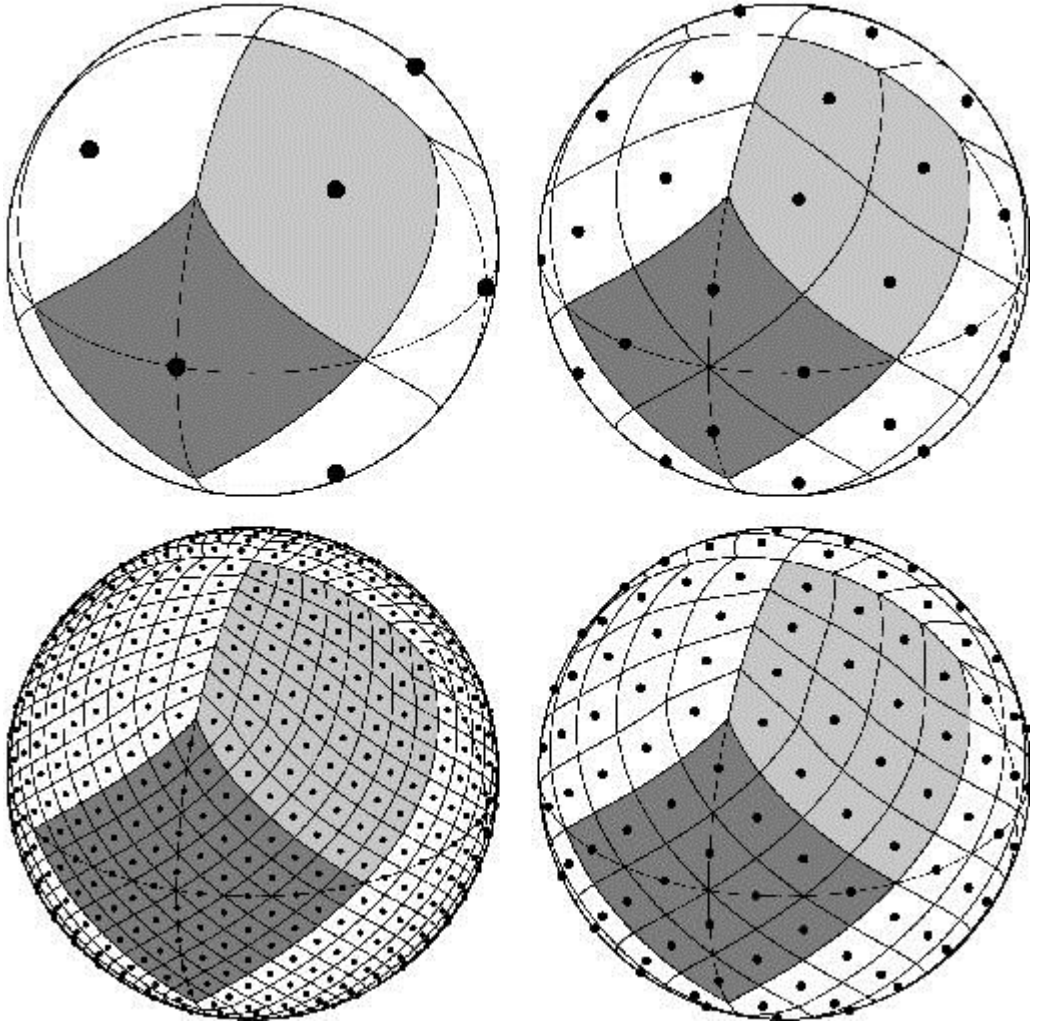






Tessellation = tiling of regular polygons

Euler Angles of the dots represent uniform sampling



Resolution of the tessellation increases by division of each pixel into four new ones.

https://mathworld.wolfram.com/Tessellation.html

Uniform angular sampling – 3D

Angular step = 30^0

```
_rlnPsiStep       30.000000
```

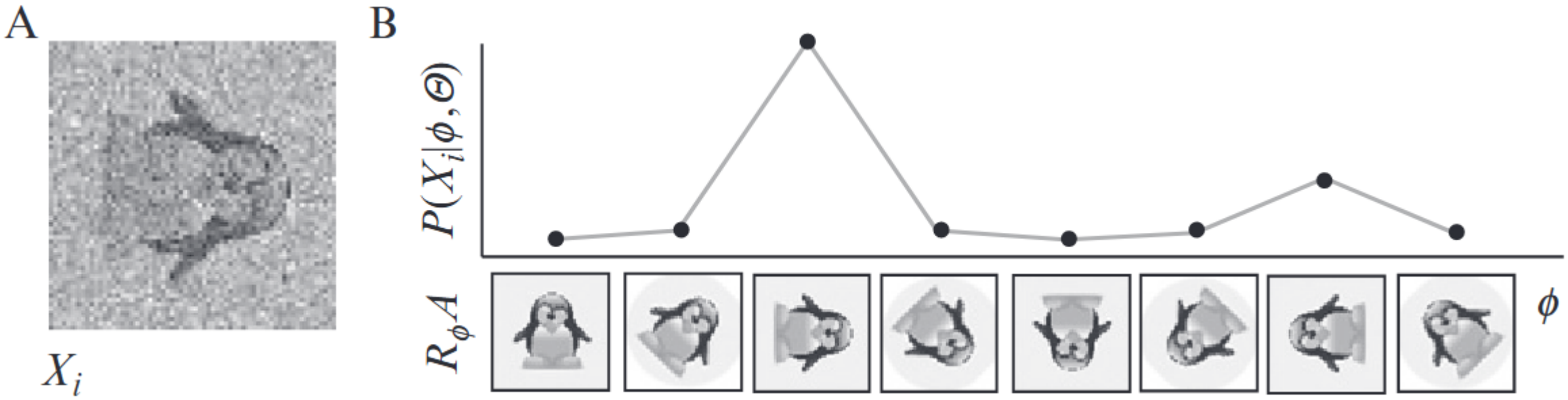
49 - ϕ, θ angle pairs

```
data_sampling_directions

loop_
_rlnAngleRot #1
_rlnAngleTilt #2
   45.000000    70.528779
   67.500000    48.189685
   22.500000    48.189685
   45.000000    23.556464
  135.000000    70.528779
  157.500000    48.189685
  112.500000    48.189685
  135.000000    23.556464
 -135.00000    70.528779
 -112.50000    48.189685
 -157.50000    48.189685
 -135.00000    23.556464
  -45.00000    70.528779
  -22.50000    48.189685
  -67.50000    48.189685
  -45.00000    23.556464
    0.000000   109.471221
   22.500000    90.000000
```

Maximum likelihood

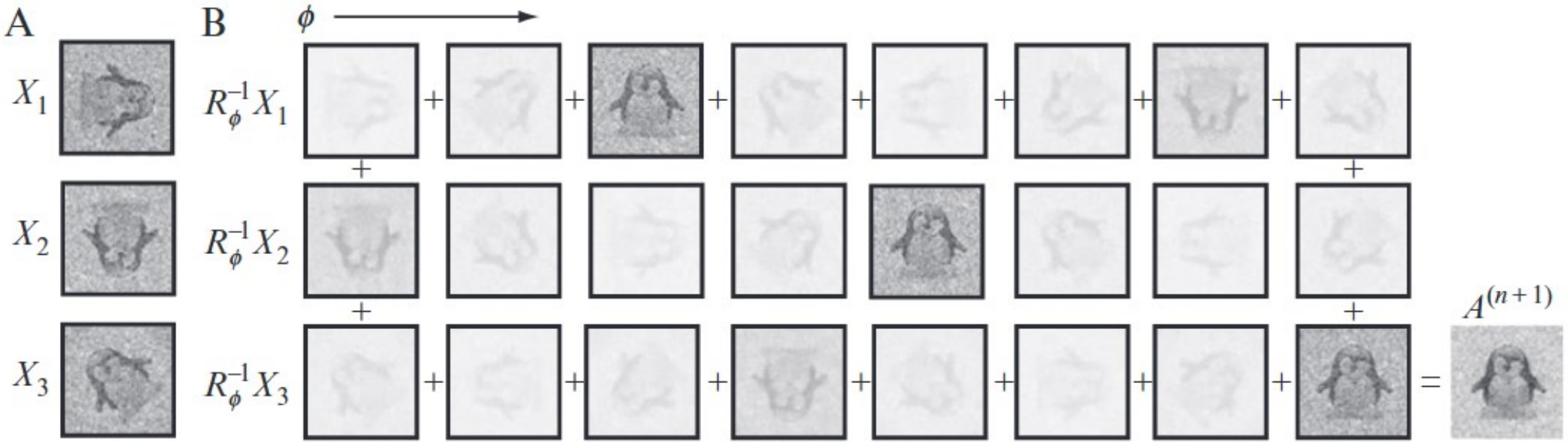
- The reconstruction problem is formulated as finding the model that has the highest probability of being the correct one in the light of both the observed data and available prior information.



Sigworth 2010

https://doi.org/10.1016/S0076-6879(10)82011-7

Summing over all significant probabilities



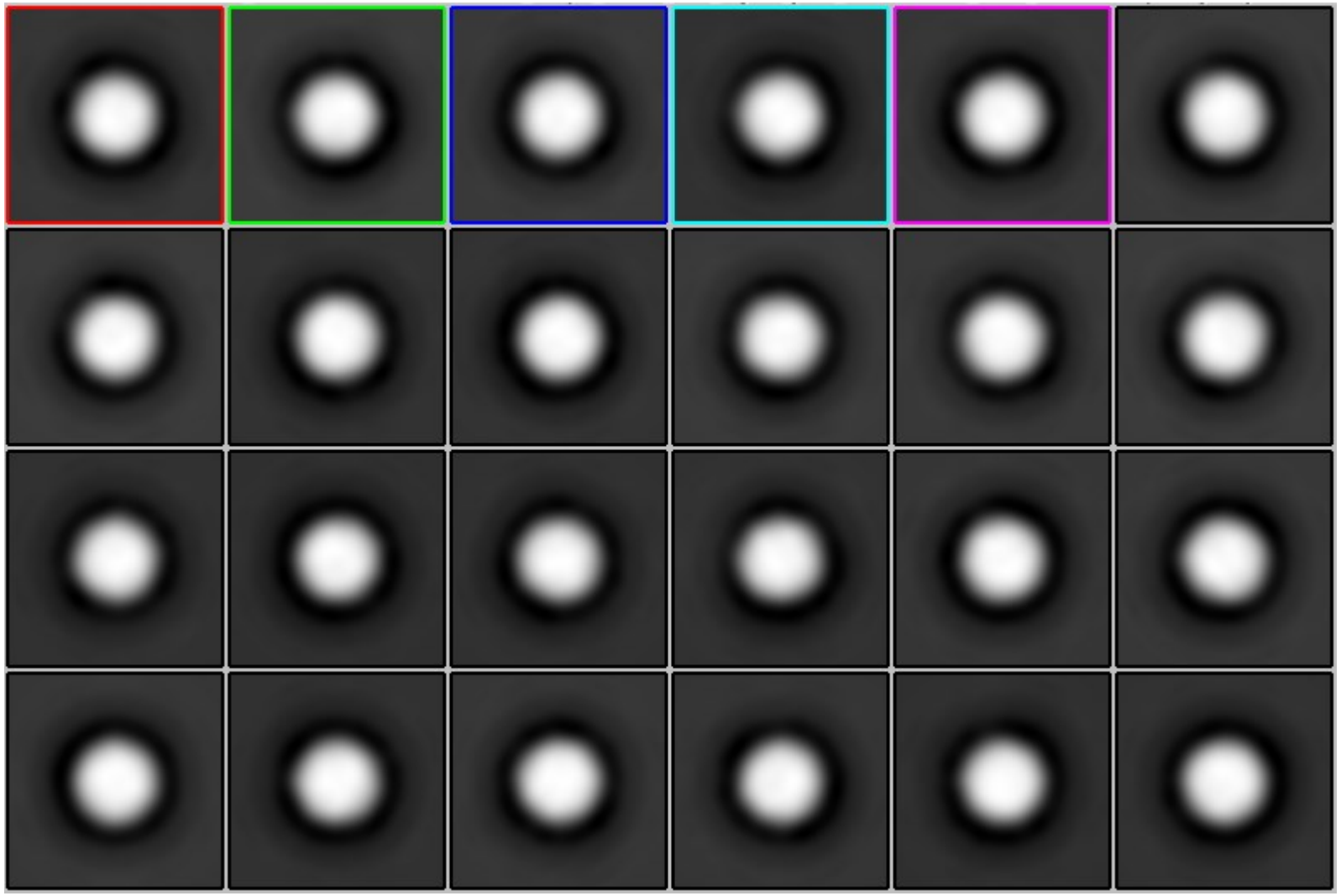
2D classification

- Performed in 2 dimensions: 1 (in plane) rotation angle ; 2 (X, Y) shifts
- Classification of particles into “k” classes
- Reference free method
  - As references randomly oriented particles are randomly distributed among initial classes
- Iterative method
- Correlation-based / Maximum-likelihood based methods of classification
- Real-space / Fourier-space based classification
- Looking for best fit into class when applying a certain shift + rotation psi on the particle
- Coarse evaluation of the dataset
  - Removing junk particles
  - Judging presence of preferred orientations
  - Checking quality of the particles (presence of high resolution details)

2D classification

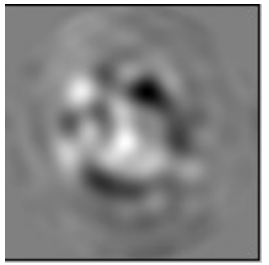
- Performed usually on binned particles
  - No need of high-res information
  - Computational speedup
- Maximum-likelihood method
  - Superior classification in Fourier space
  - Tendency to group “bad particles” into “good classes” after many iterations
    - Creating empty classes
    - Solution:
      - Limit the number of iterations (gradually check the state of classification -> not necessary the last iteration is the best iteration)
      - Output of one 2D classification (selected 2D classes) as input for another round of 2D classification
- Parameters: number of classes; rotational sampling; translational range

2D Classification – iteration 0

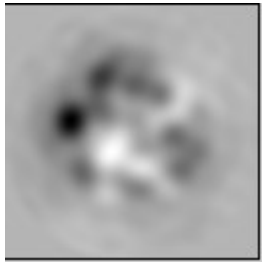


Initialization of the classes – randomly distributing particles in random orientation

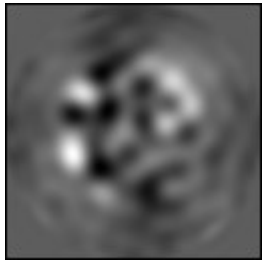
Class average differences



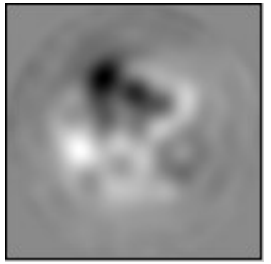
Class 1 - Class 2



Class 1 - Class 3

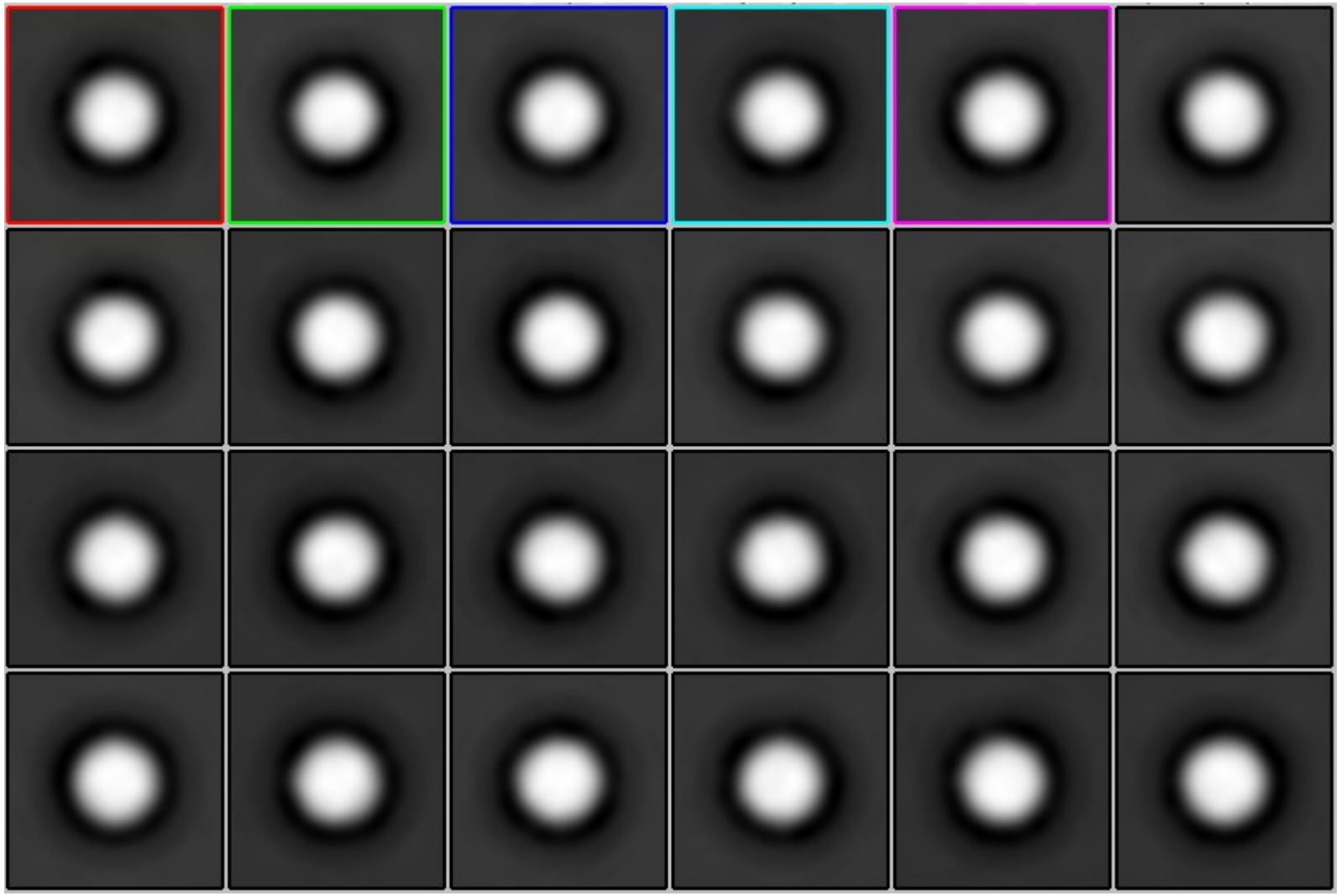


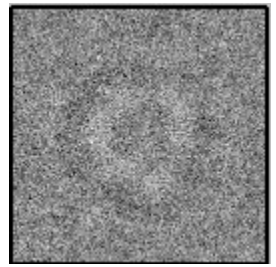
Class 1 - Class 4



Class 1 - Class 5

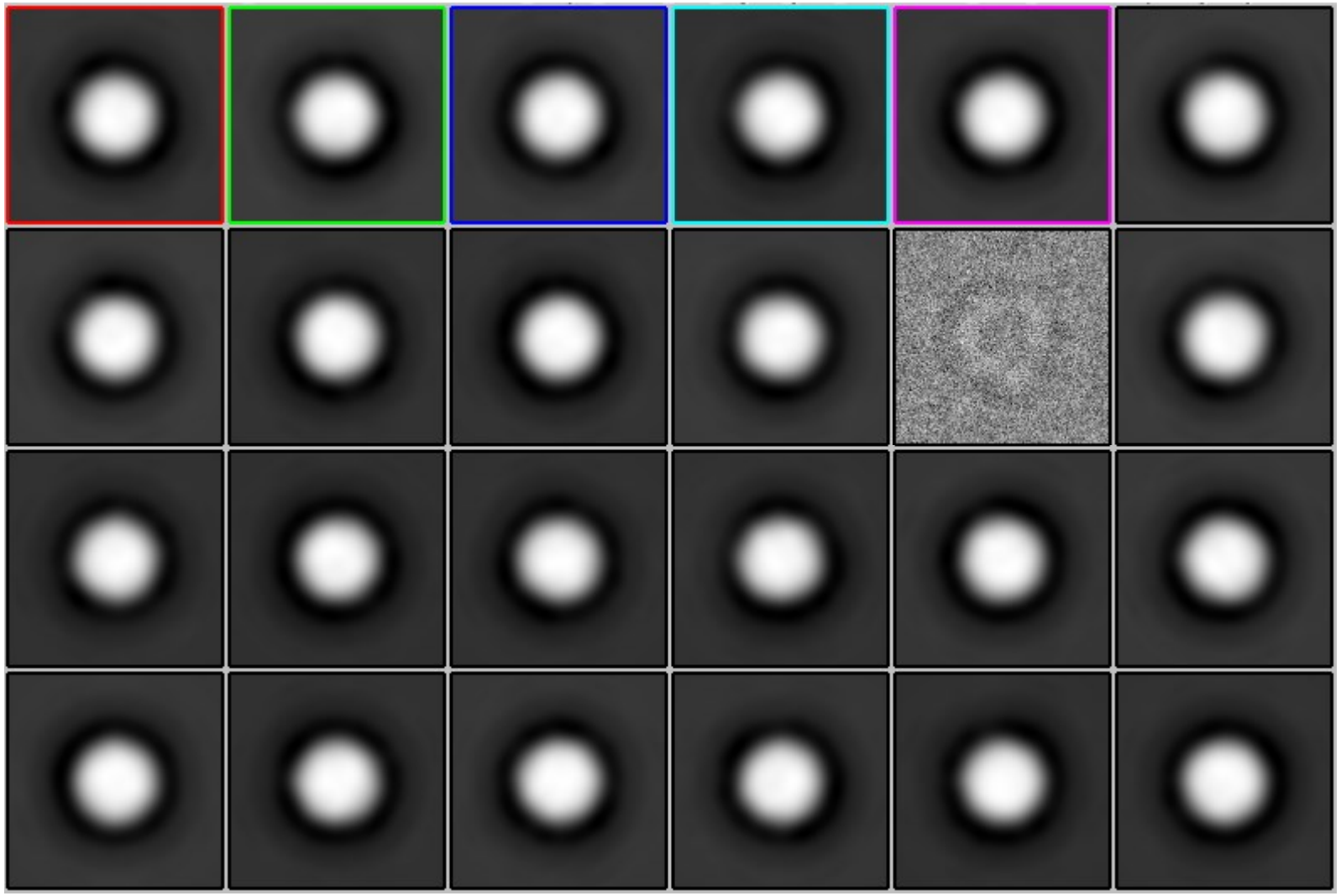
2D Classification – rotational/translational search



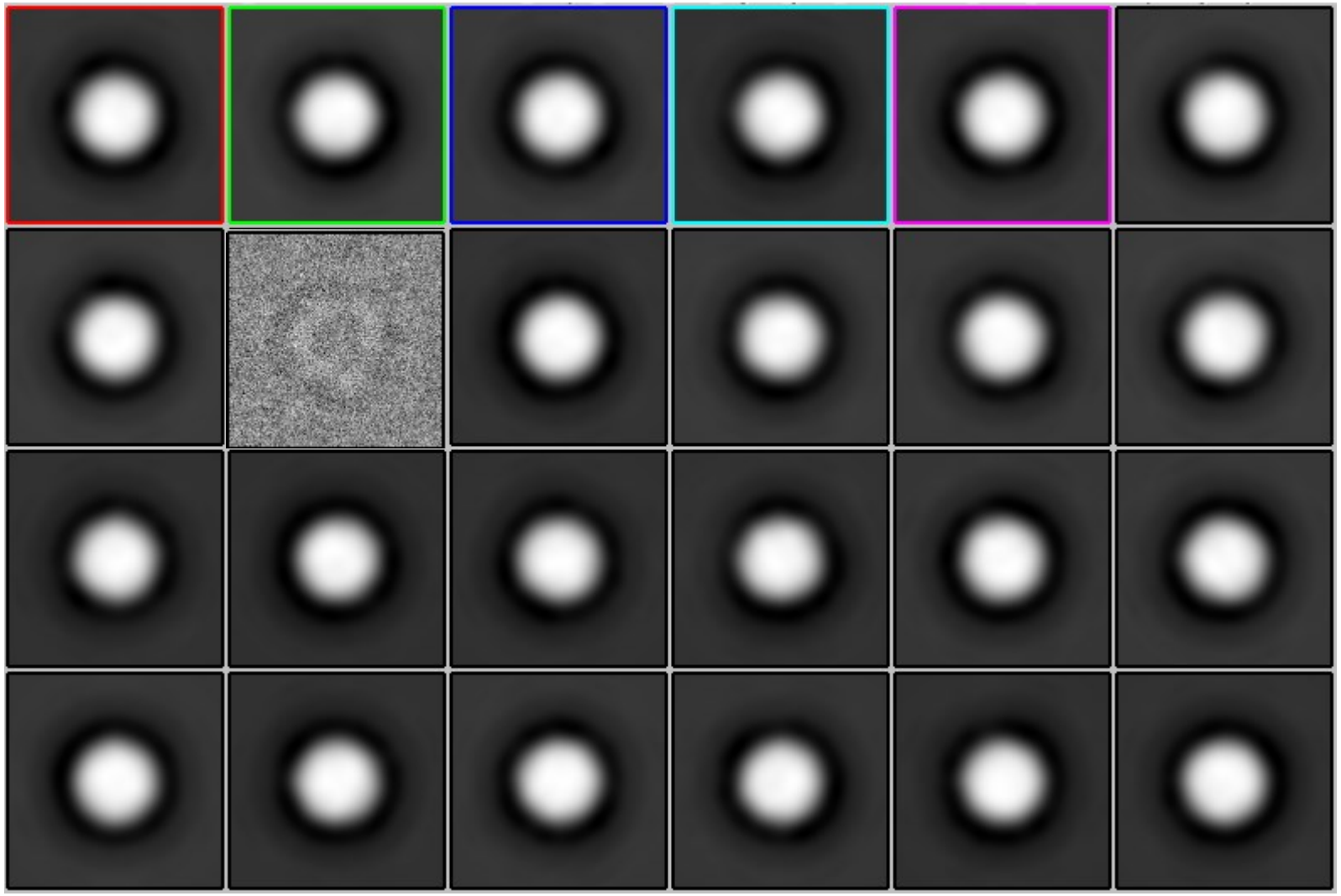


Noisy particle to align and find correct class

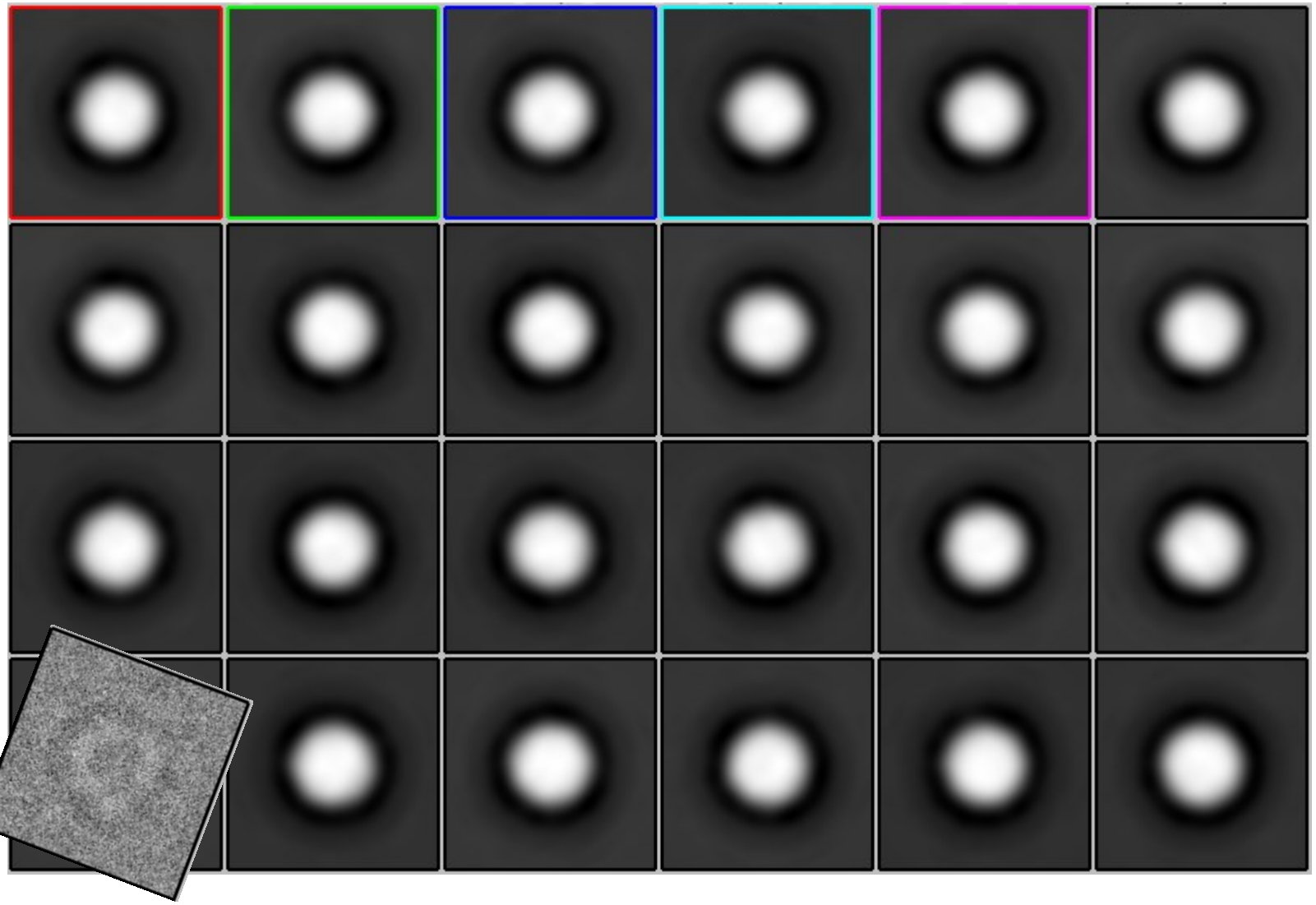
2D Classification – – rotational/translational search



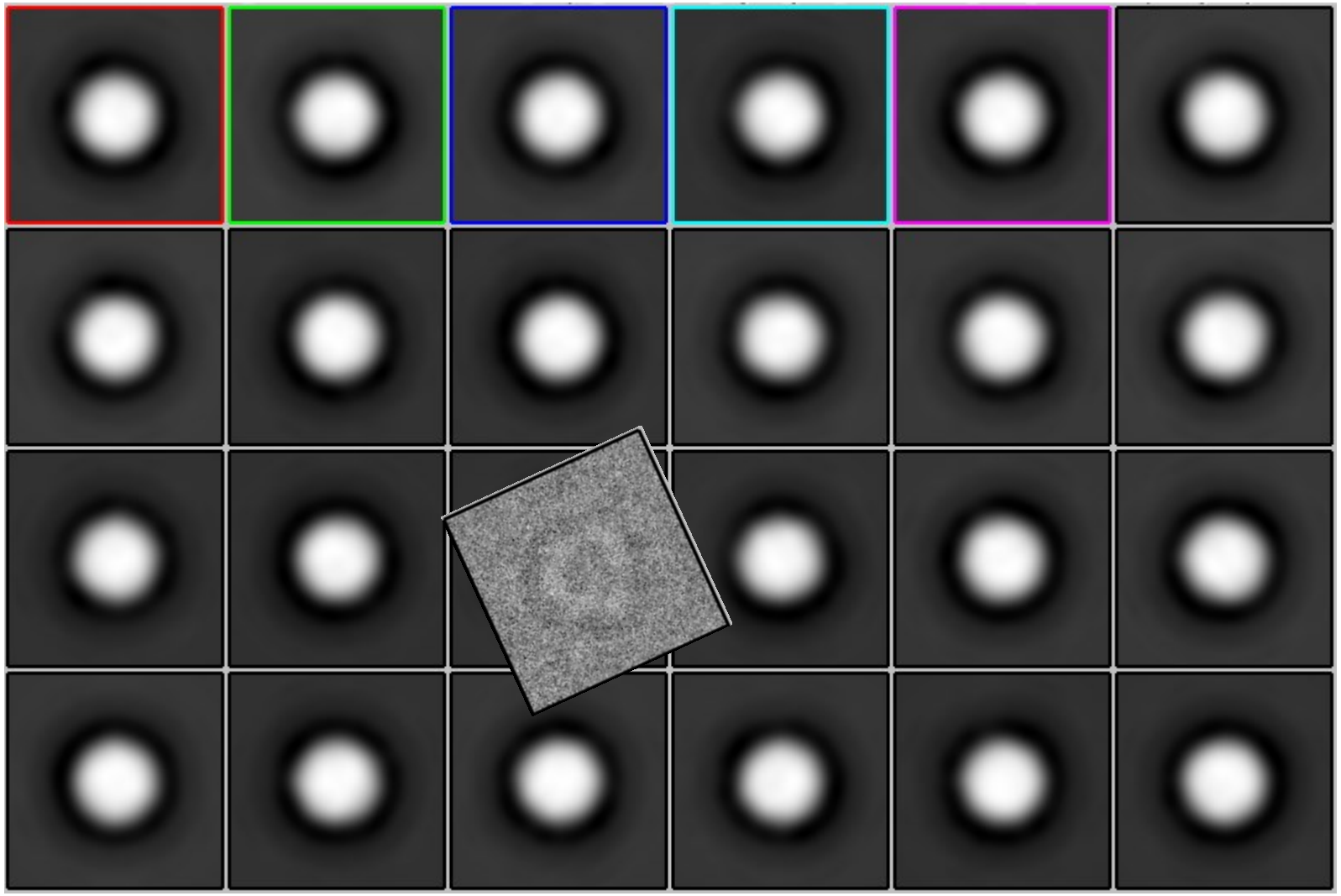
2D Classification – – rotational/translational search



2D Classification – rotational/translational search



2D Classification – rotational/translational search

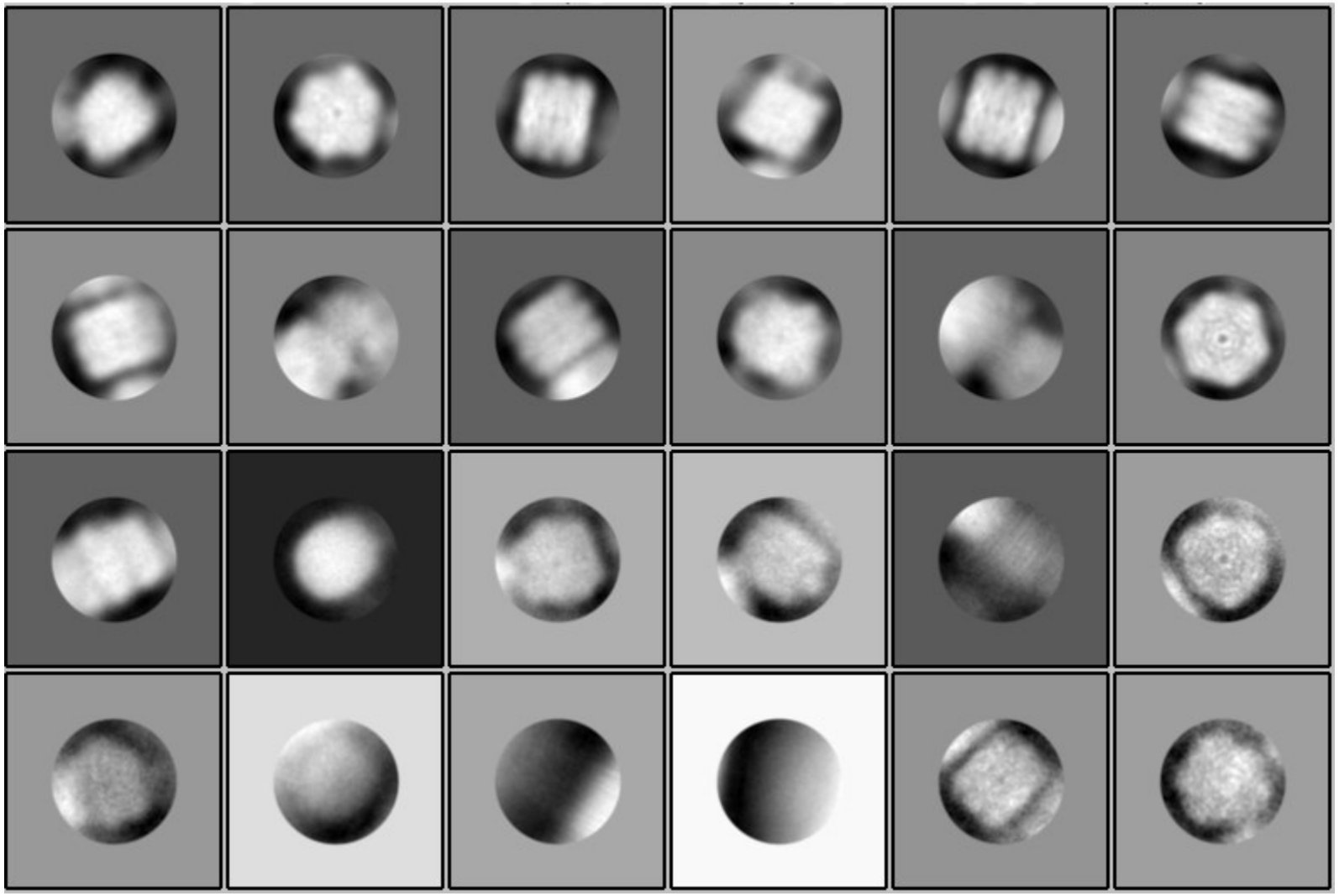


Fitting all particles
in all orientations into all classes

Choosing the best fitting class
 and best fitting orientation

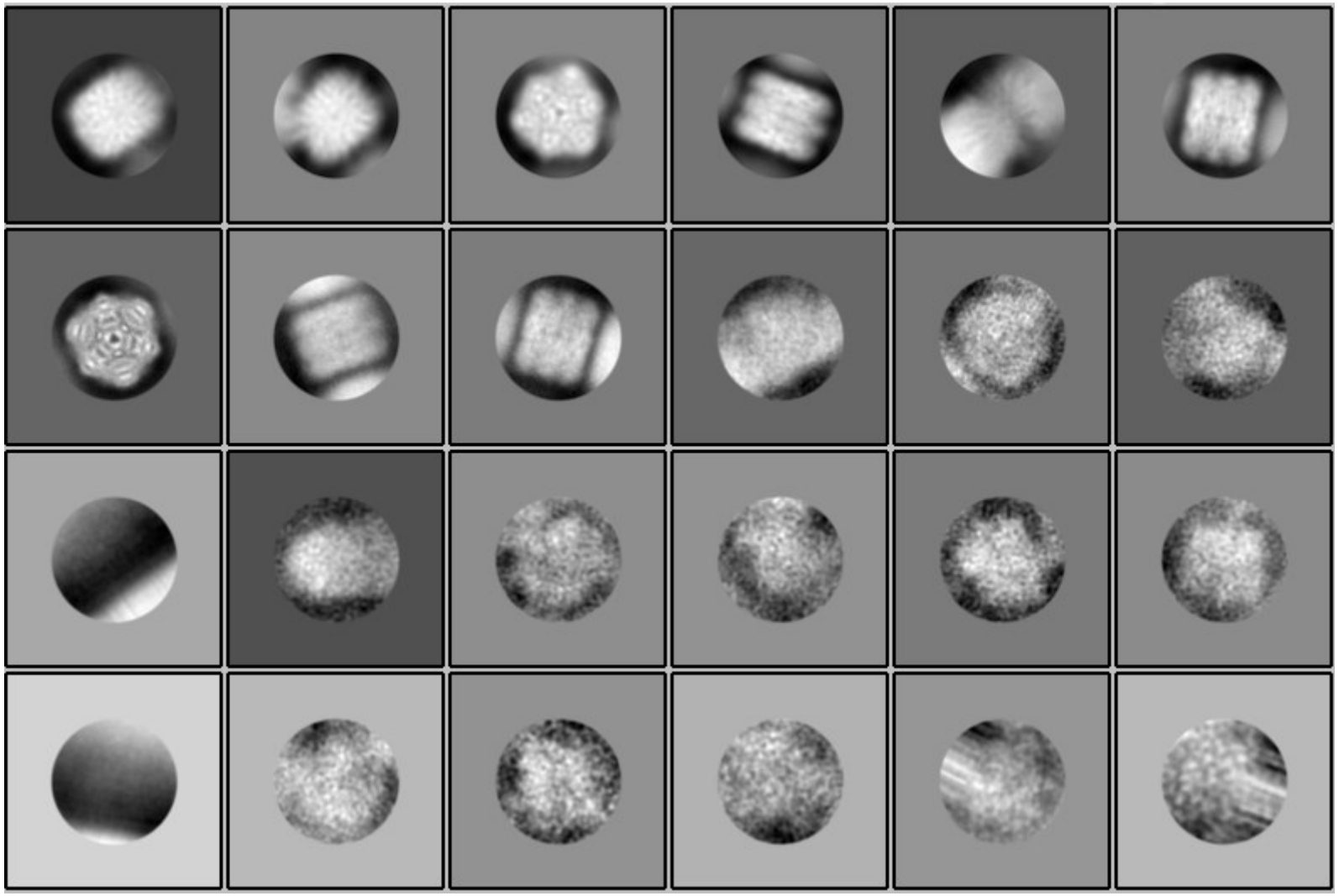
When all particles sorted into classes:
Creating average of the properly oriented
Particles – “2D Class Average”

2D Classification – iteration 5



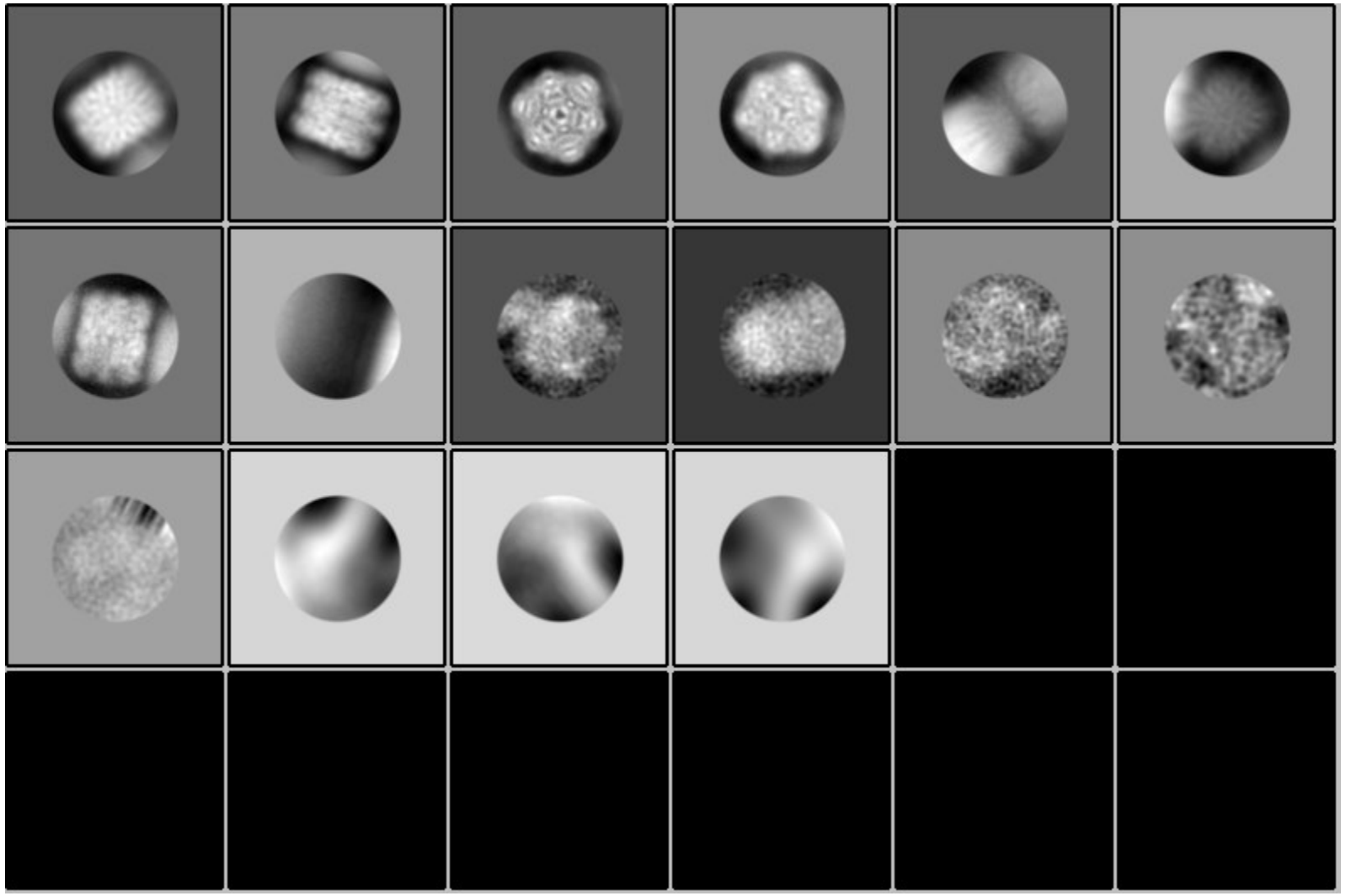
Sorted by most populated classes

2D Classification – iteration 8



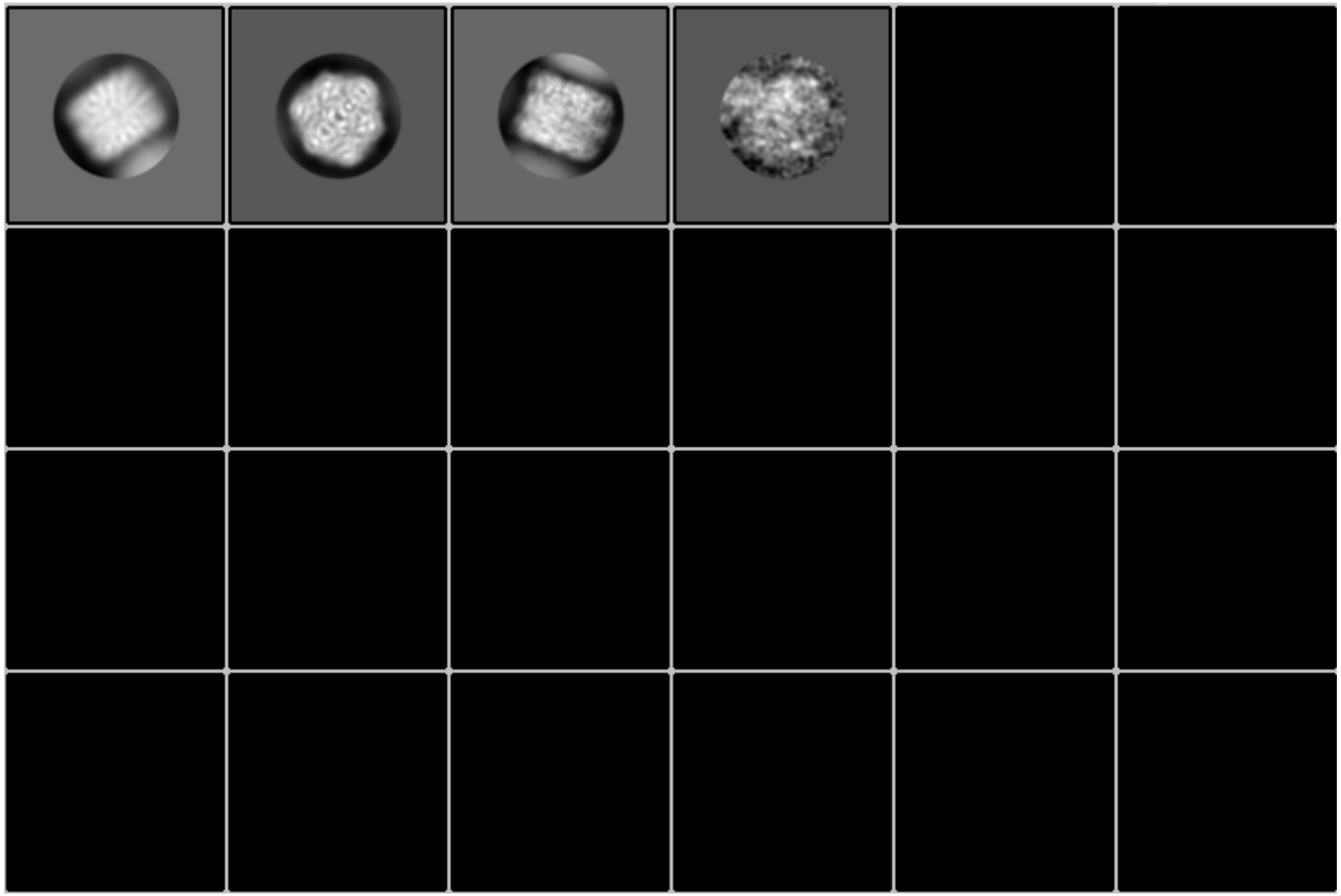
Sorted by most populated classes

2D Classification – iteration 10



Sorted by most populated classes

2D Classification – iteration 15



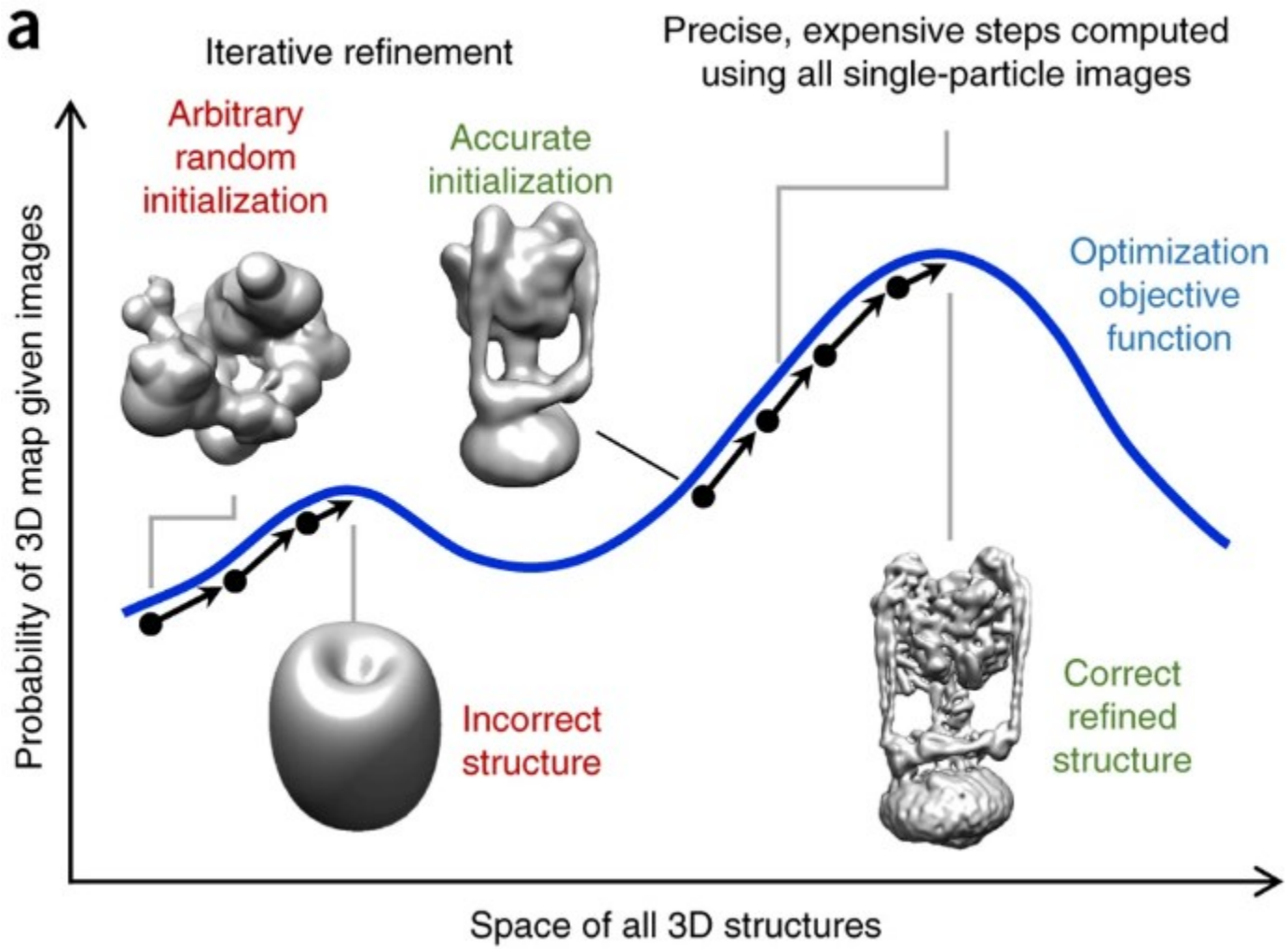
Sorted by most populated classes

3D initial-model (reference) generation

- Alignment problem in 3D is not solvable without prior reference
- Danger of model bias
  - initial models (references) need to be heavily lowpass filtered
- Geometrical models (sphere, cylinder etc…)
- Known structure as initial model (ribosome, icosahedral virus, etc)
  - Heavily lowpass filtered (~40 Å)
- Generation of experimental models
  - Common Lines Method (utilizing common line in Fourier space theorem)
  - Sub-tomogram averaging (need to collect tomograms)
  - Random conical tilts (need to collect tilted dataset)
- *De novo* generation of initial model form the dataset
  - Stochastic gradient descent

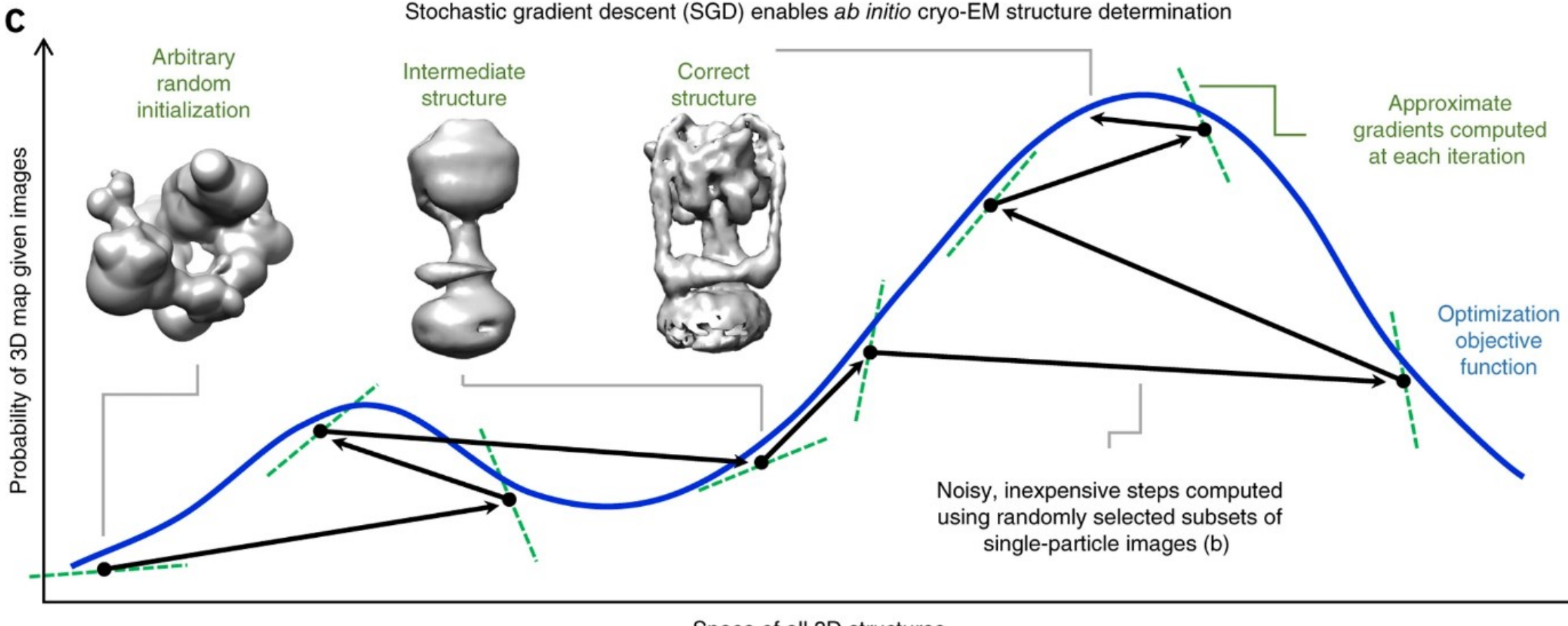
Good vs bad initial model

-> refinement converges to local minima (local maximum probability)



SGD – model generation

- Arbitrary random initialization in 3D
- Random selection of small subsets of particles
  - approximate the true optimization objective (at resolution of current iteration) – find the best orientations of the subsets



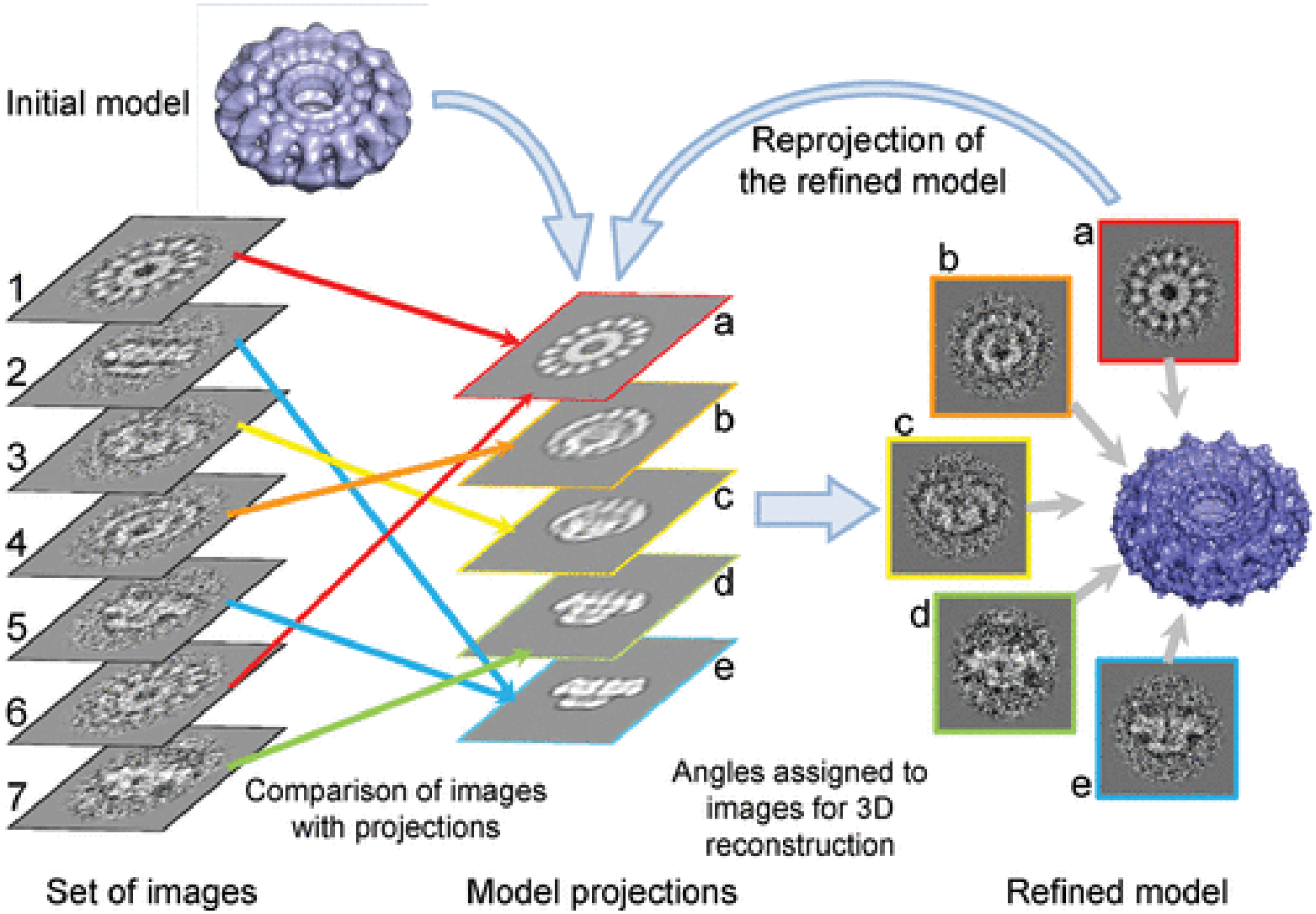
3D refinement

- Automated procedure to 3D align particles against reference
- Every particle represent a projection of the model from certain view (orientation – 3 Euler angles; 2 shifts)
- Iteratively improving the resolution of the reference => iteratively improving the estimated shifts and orientations of the particles
- Gold standard refinement
  - Splitting the dataset into 2 random (independent) halves
  - Both halves has their own set of unique particles and their initial reference
  - During iterative particle alignment the newly generated references for the next iteration steps are filtered and resolution limited to resolution calculated by FSC between the two halves
  - Iterations done until no change in resolution or angular accuracy of the orientations is achieved

Global search vs local search

- Global search
  - Setting a uniform angular sampling at defined sampling rate
  - Using all angles for angular search of particle against the reference
  - The priors of the angles are not transferred to the next iteration
    - Searching again the whole angular sampling space again against the improved reference
    - Avoids stuck of the particles at local minima
  - Global searches are not computationally feasible after reaching very fine sampling rate
- Local search
  - Taking the last best orientation of the particle from previous iteration result as prior
  - Searching (refining angles) only in a defined (narrow) sigma distance at the set sampling rate
  - Previous iterations using global searches should ensure that the particles are not around any incorrect local minima, rather around the global correct solution

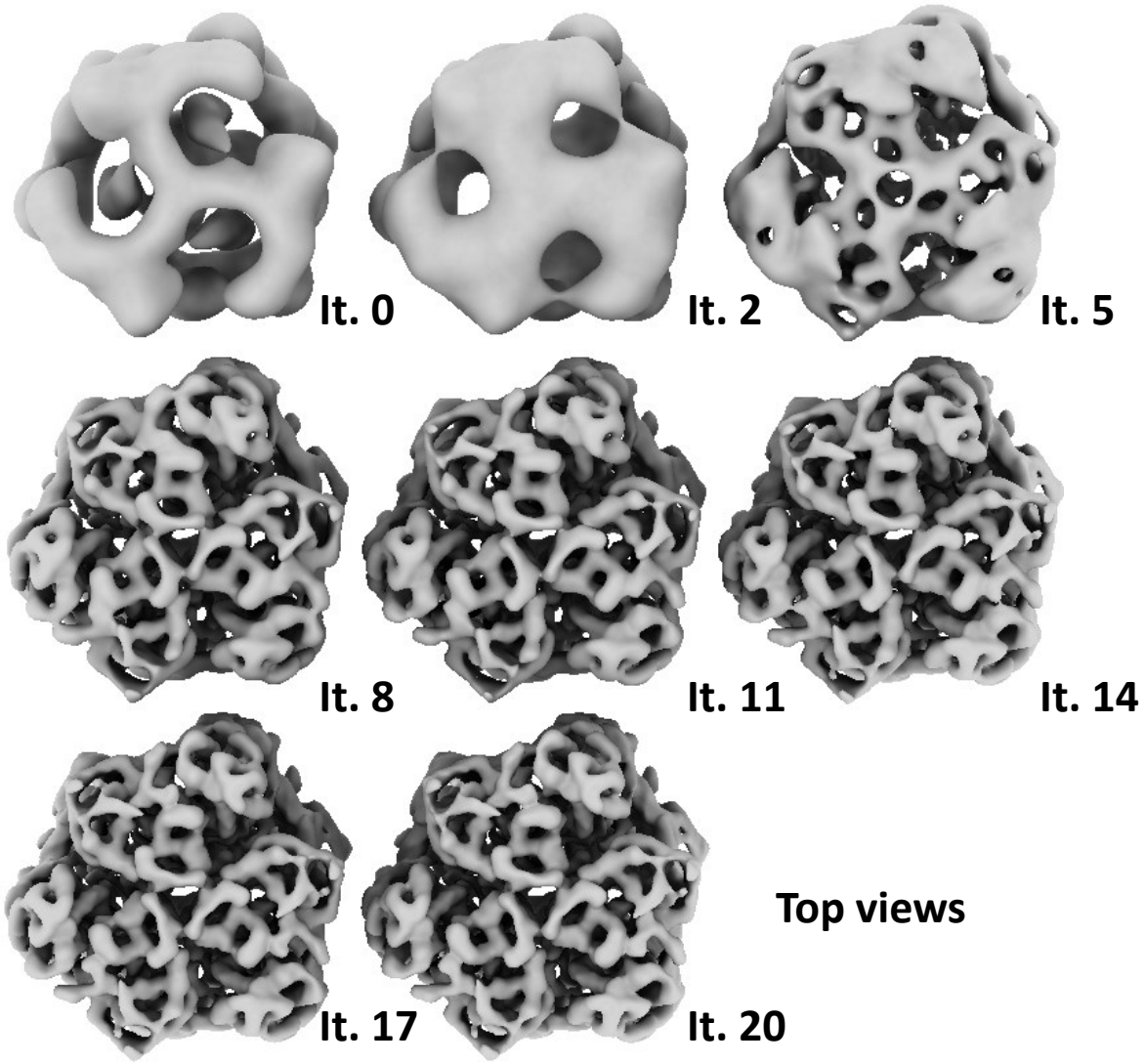
Flow of 3D model refinement

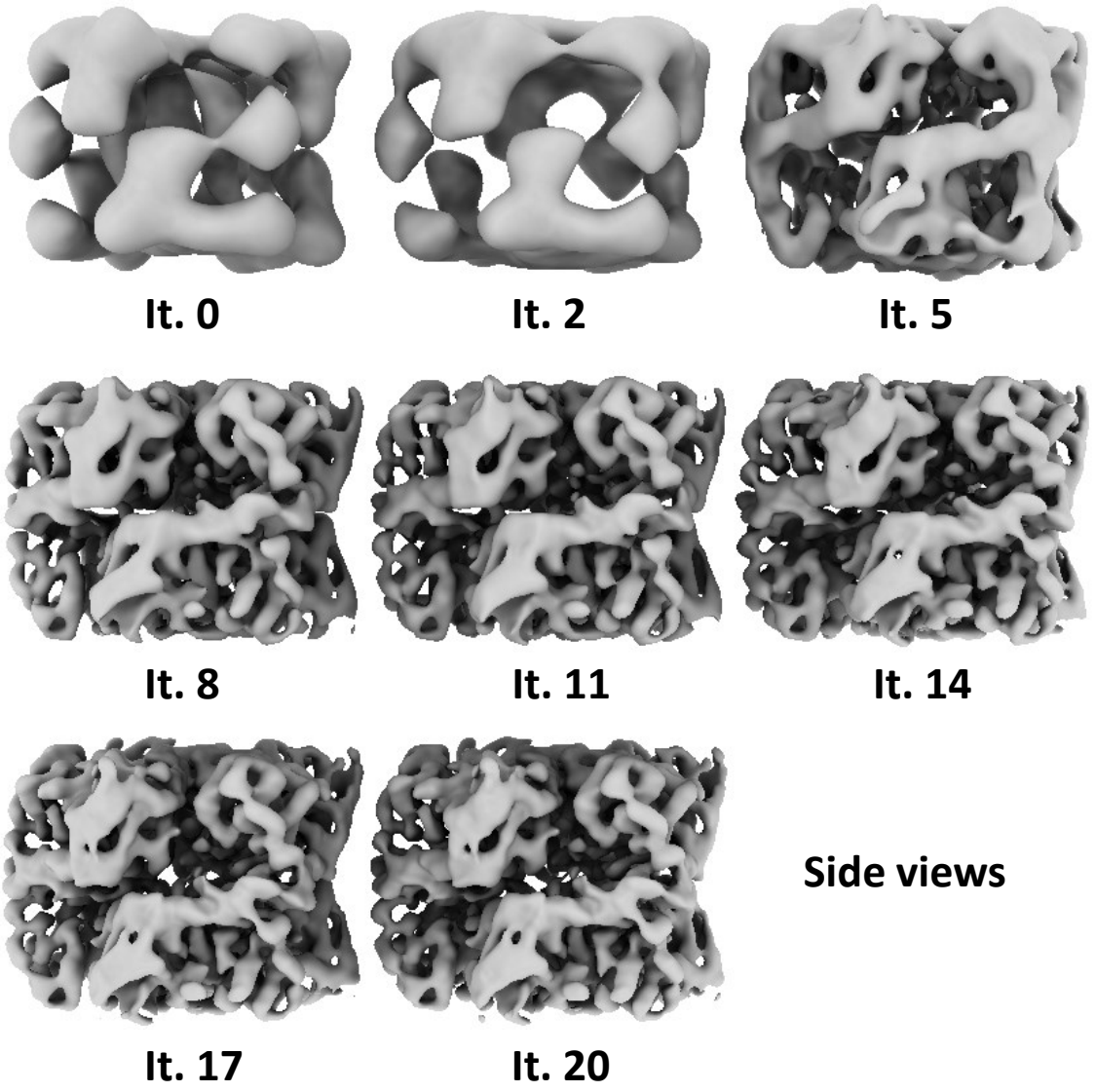


Orlova 2011

https://doi.org/10.1021/cr100353t

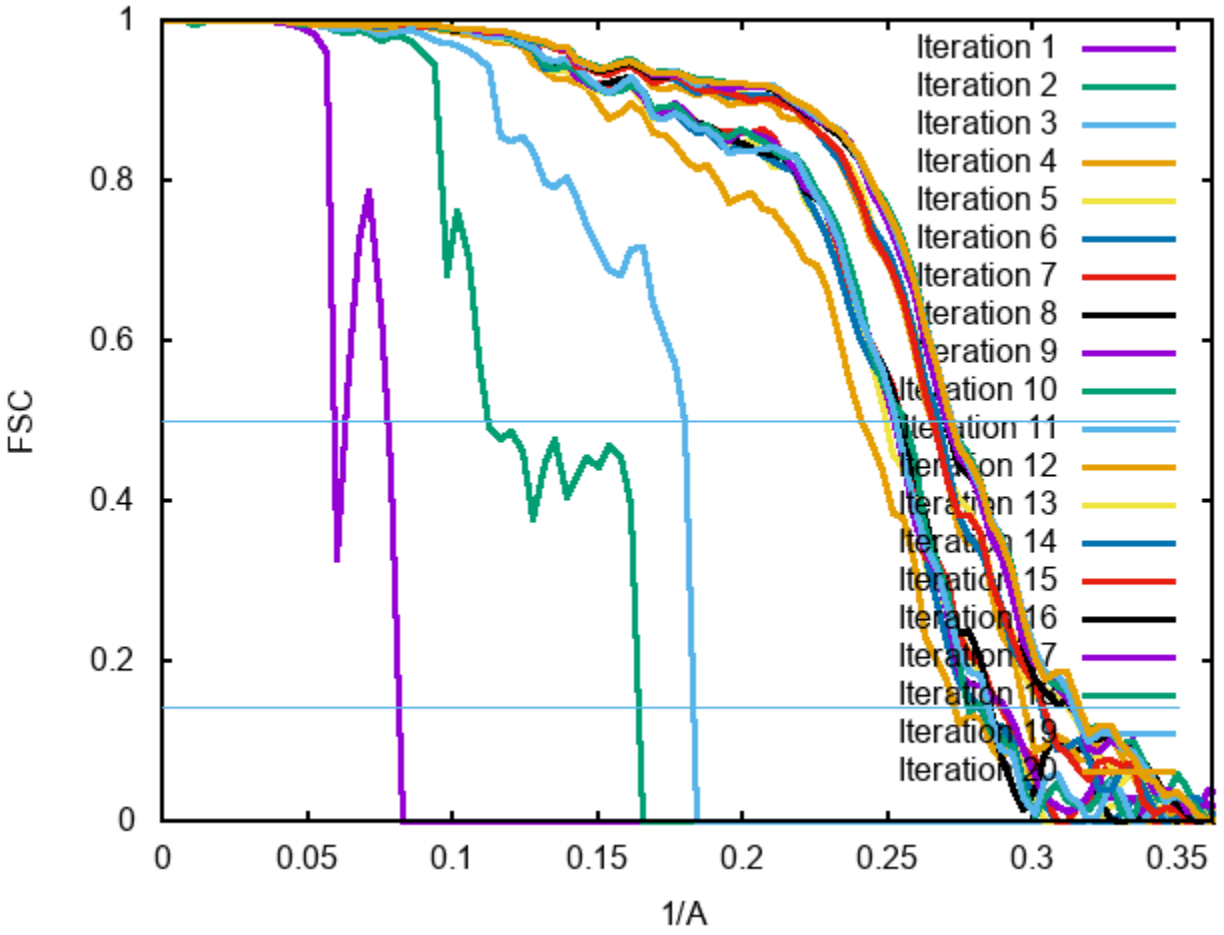
Map resolution improvement during iterations

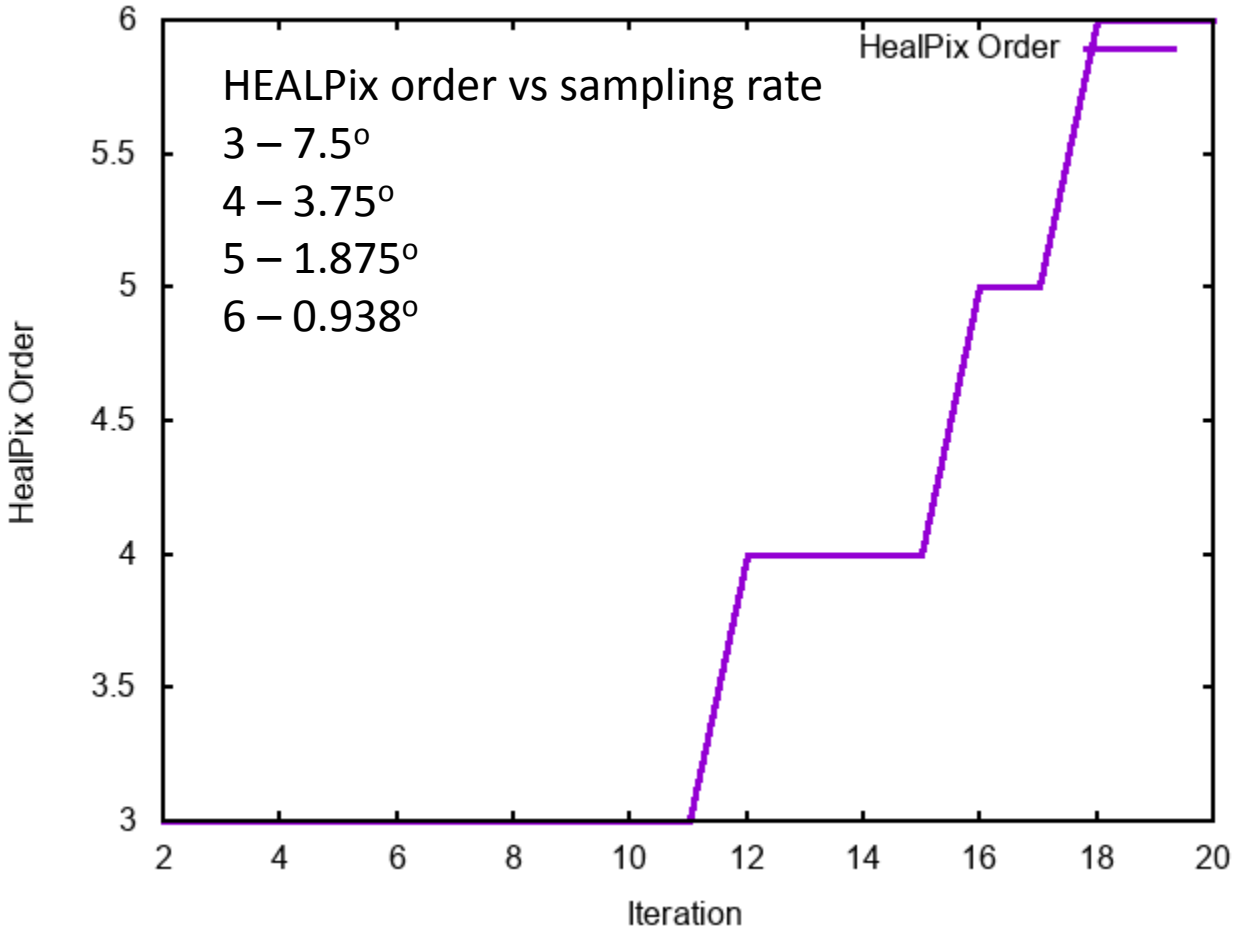




Change of resolution, sampling rate

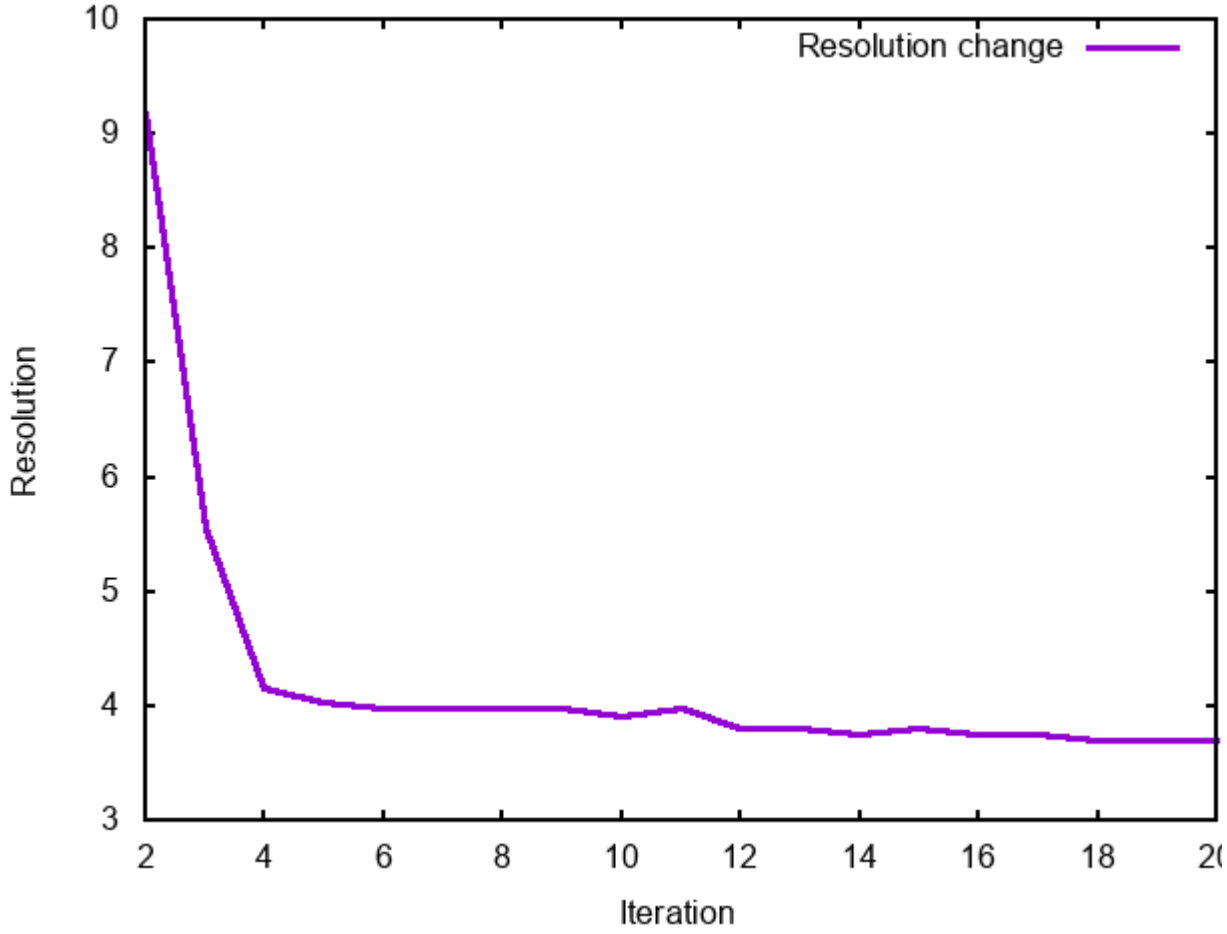
The higher the resolution the higher sampling rate is needed to align the partices to improve the resolution

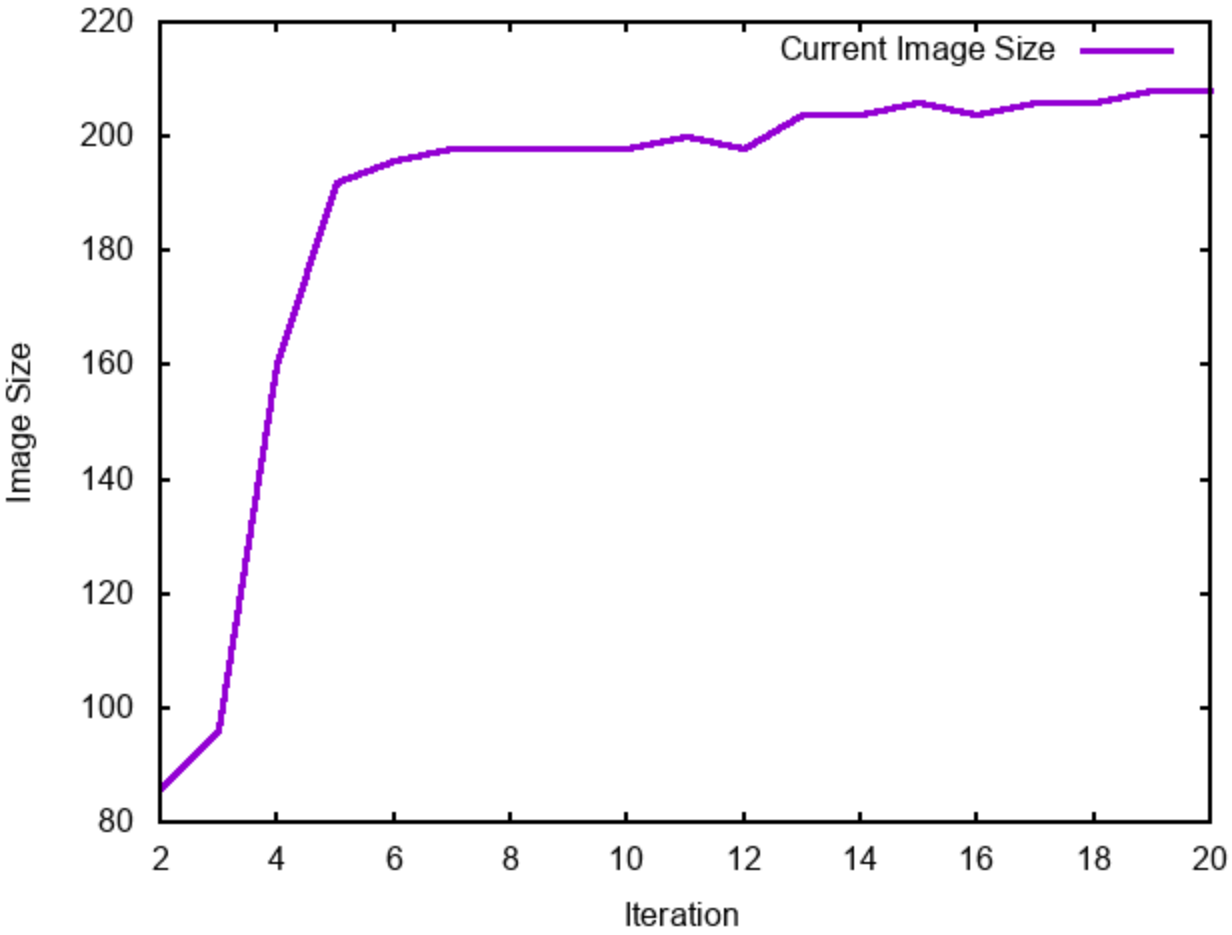




Refinement process in plots

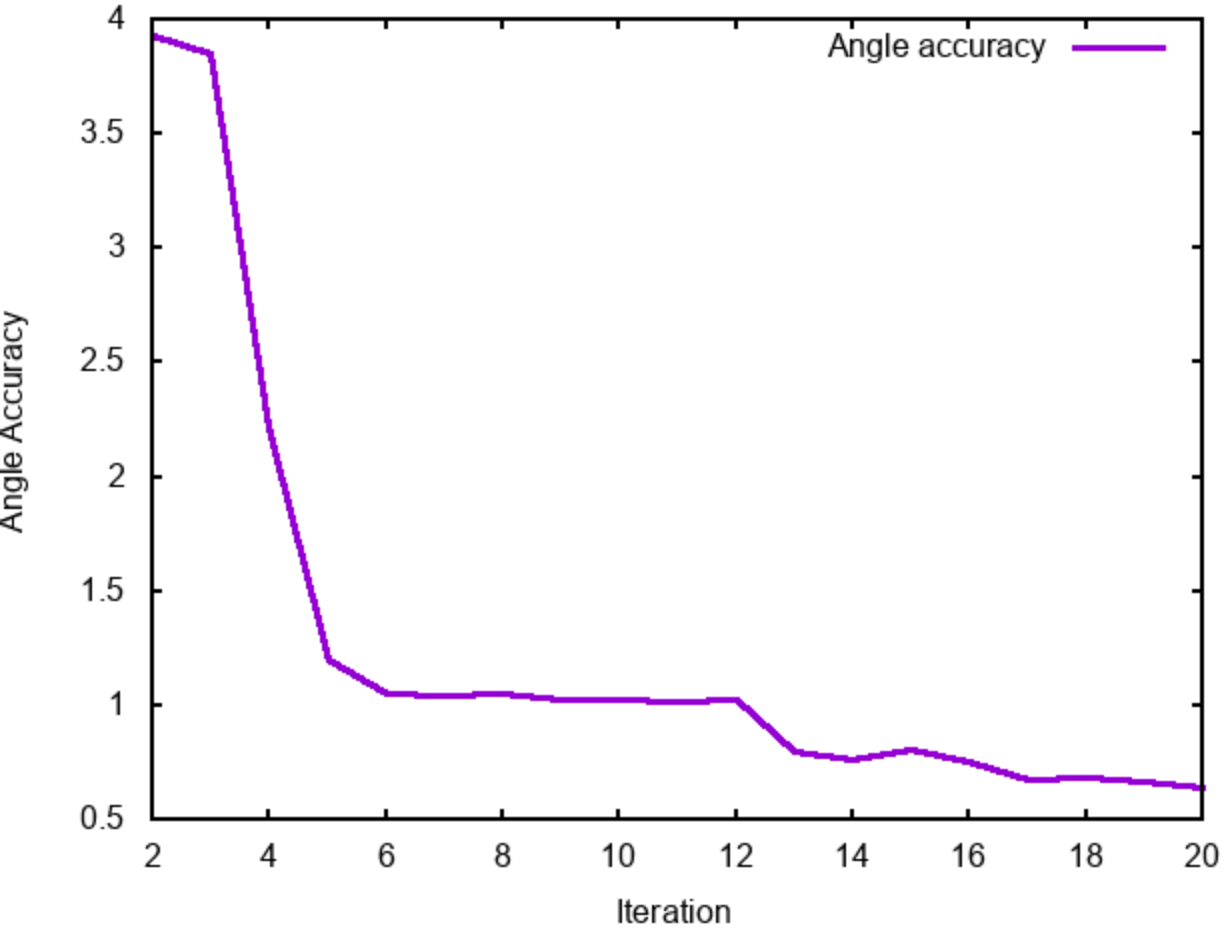
For low resolution alignment we do not need the high frequency information, the particles are automatically binned

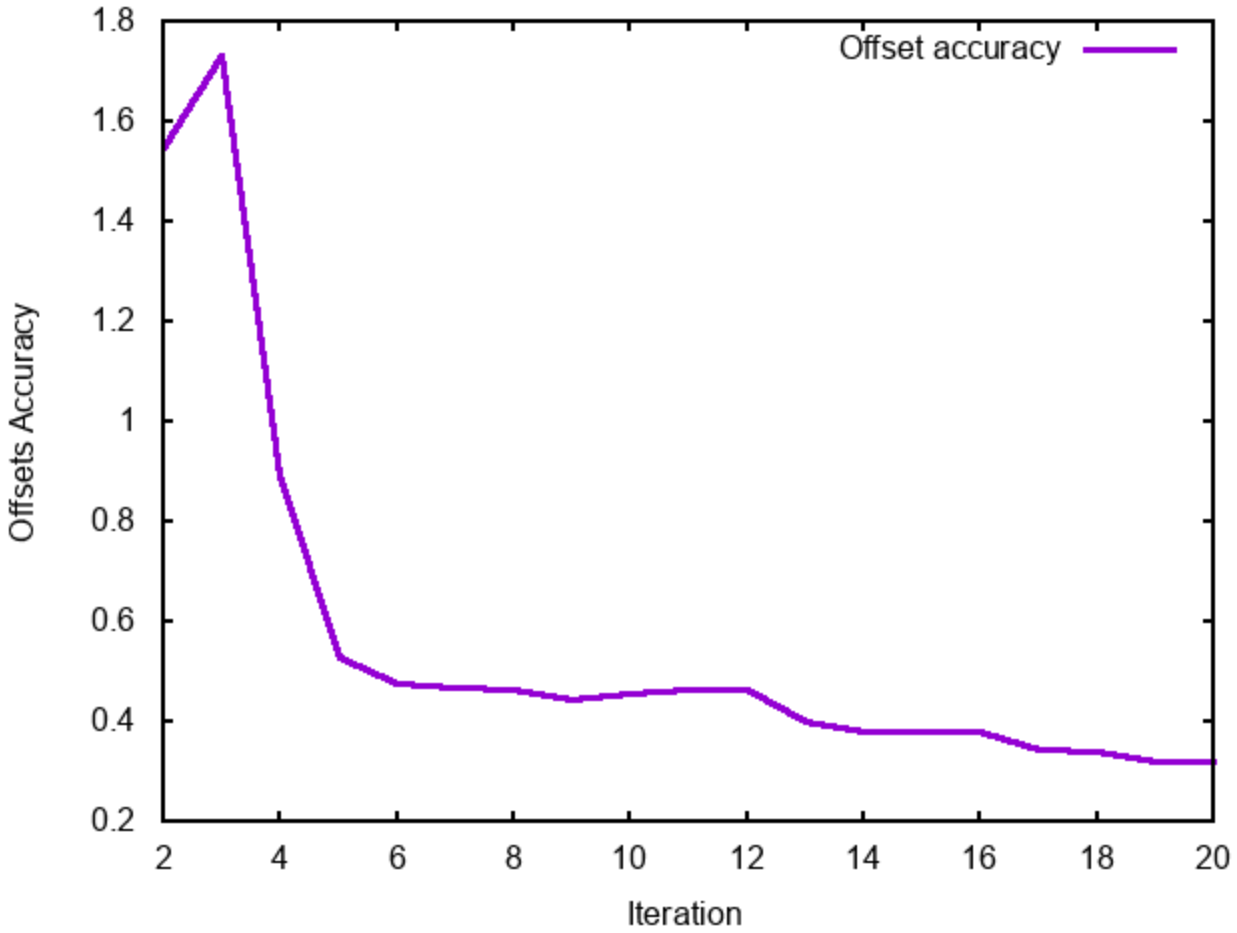




Refinement process in plots

Not only the resolution change defines the convergence of the refinement



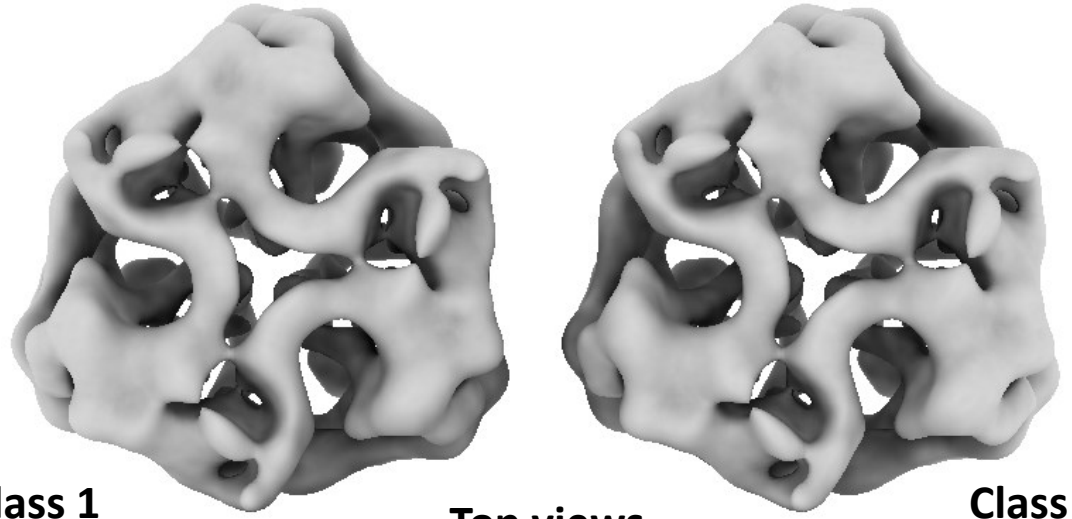


3D classification

- 3D reference based
  - Starting with low-pass filtered reference – gradually increasing resolution over iterations
- More sensitive than 2D classification
- Orientational search and classification
  - Rot, tilt, psi, shift X, shift Y, class
  - Similar to 2D classification approach in 3D
- Only classification
  - Reusing the rot/tilt/psi from previous a steps (e.g. 3D refinement)
  - Only sorting particles in between classes
- Restricted orientational search and classification
  - Restricting the orientational search range (local searches)
  - Need priors from previous a step (e.g. 3D refinement, 3D classification)
- Reported resolution is NOT FSC based resolution (missing the second independent half)
- Regularization factor
  - The higher value the more the classification is driven by data and less by smoothness

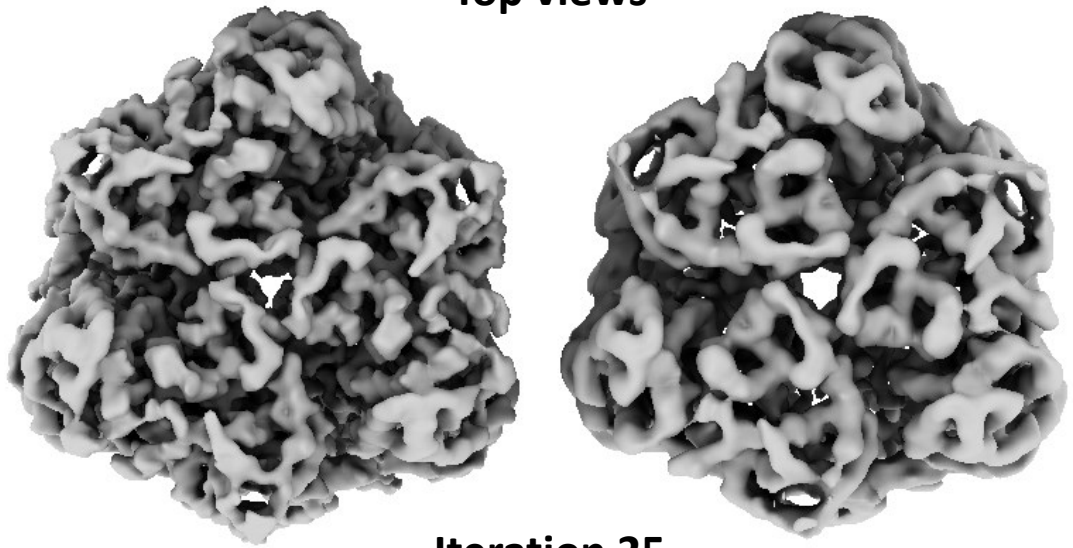
3D Classification

Iteration 0



Class 1 **Class 2**

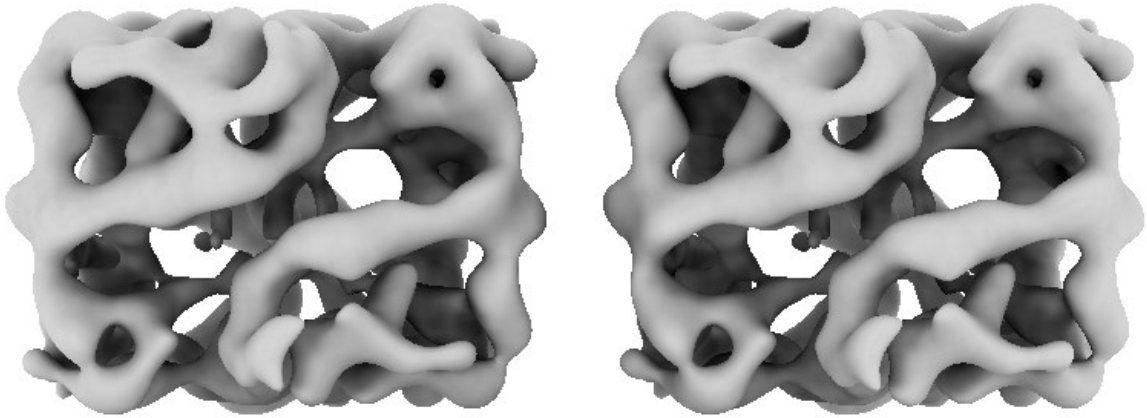
Top views



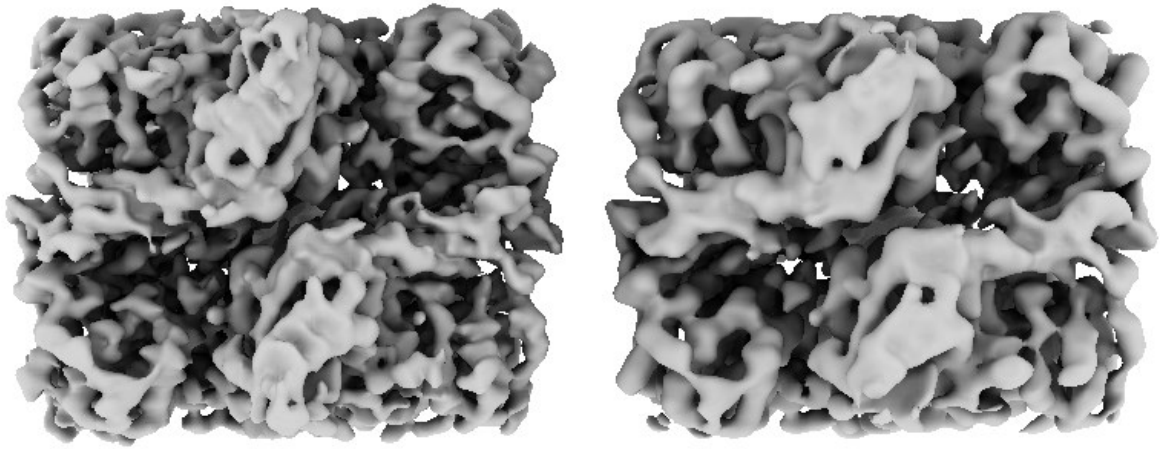
Iteration 25

Resolution: 4.22 Å Resolution: 4.75 Å

Iteration 0



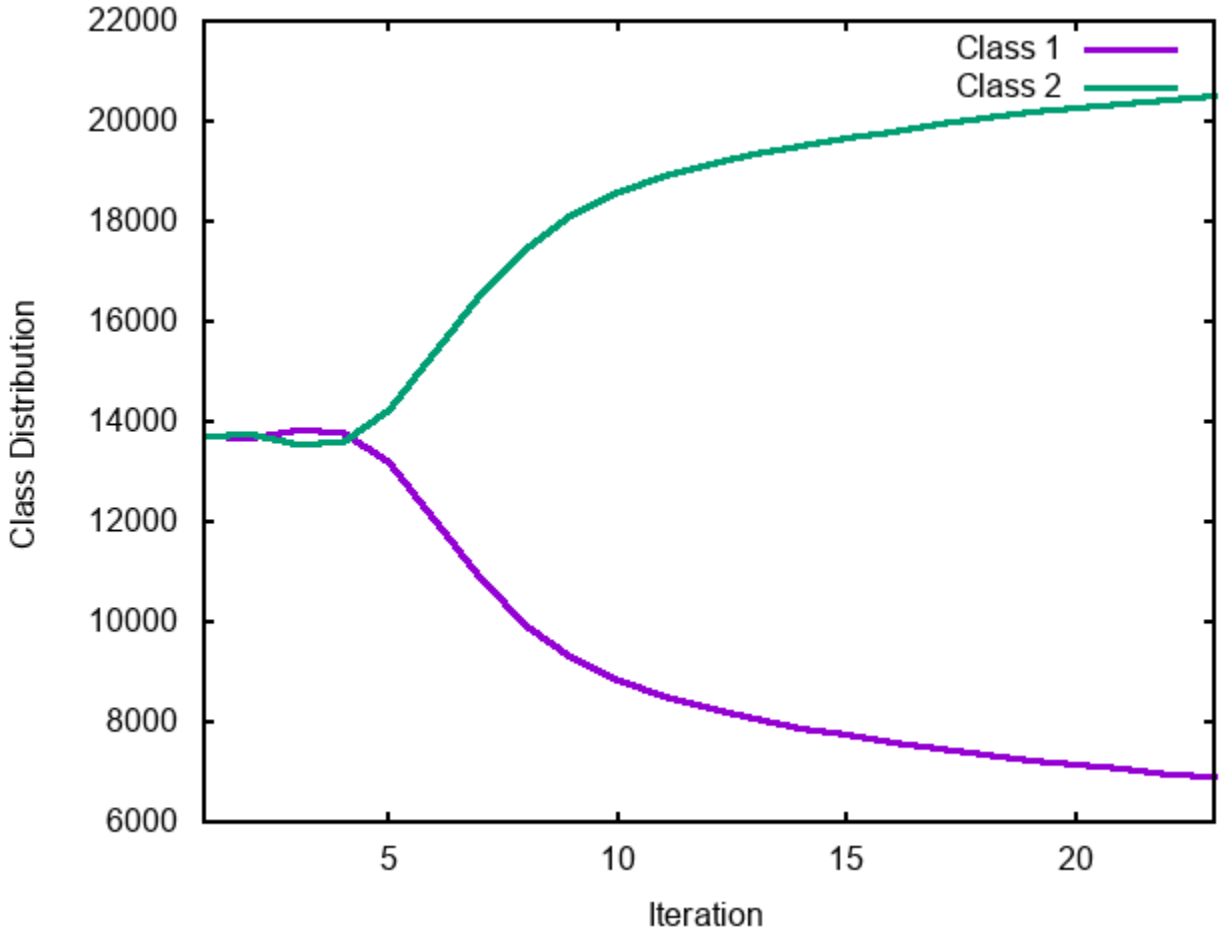
Class 1 **Side views** **Class 2**

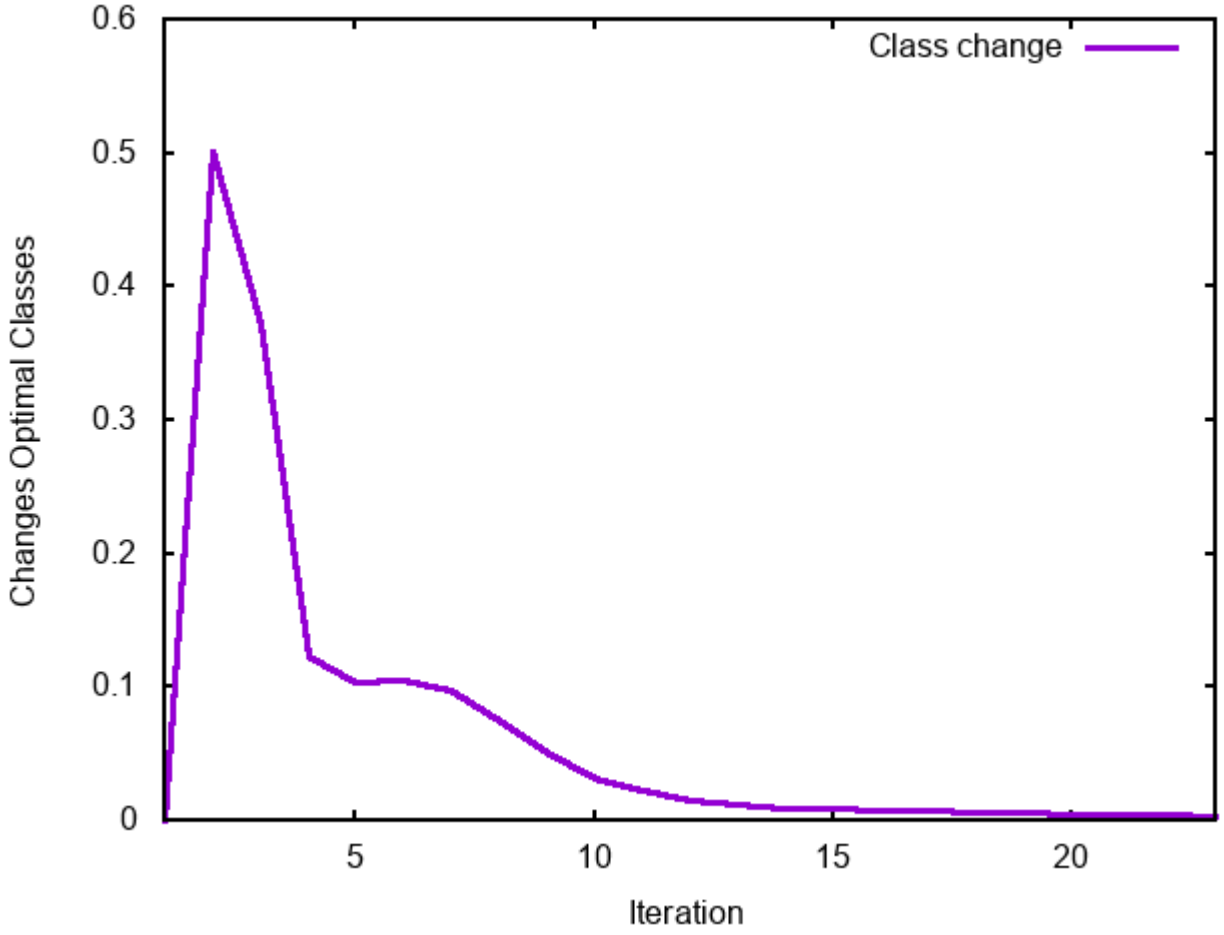


Iteration 25

3D Classification in plots

Convergence is need to be inspected





SPA limitations

- No computation can replace the perfect sample !

- Preferred orientation of particles – assumption of orientationally randomly distributed particles
- Large heterogeneity of the particles
- Flexible complexes
- Particle size limited by the visibility of the particles

- Computational demand:
  - GPU/CPU – could be done for reasonable boxes even on desktop workstations
  - Increasing box size -> cubic increase of RAM usage
  - Disk storage space

Thanks for your attention!

Next – in the advanced CryoEM methods

- Branch-and-bound refinement approach (cryosparc)
- Bayesian Polishing
- Ewald sphere correction
- Ctf refinement
- 3^{rd}, 4^{th} order aberration – estimation, correction
- Non-homogeneous refinement
- Multibody refinement
- Localized reconstruction

Postprocessing

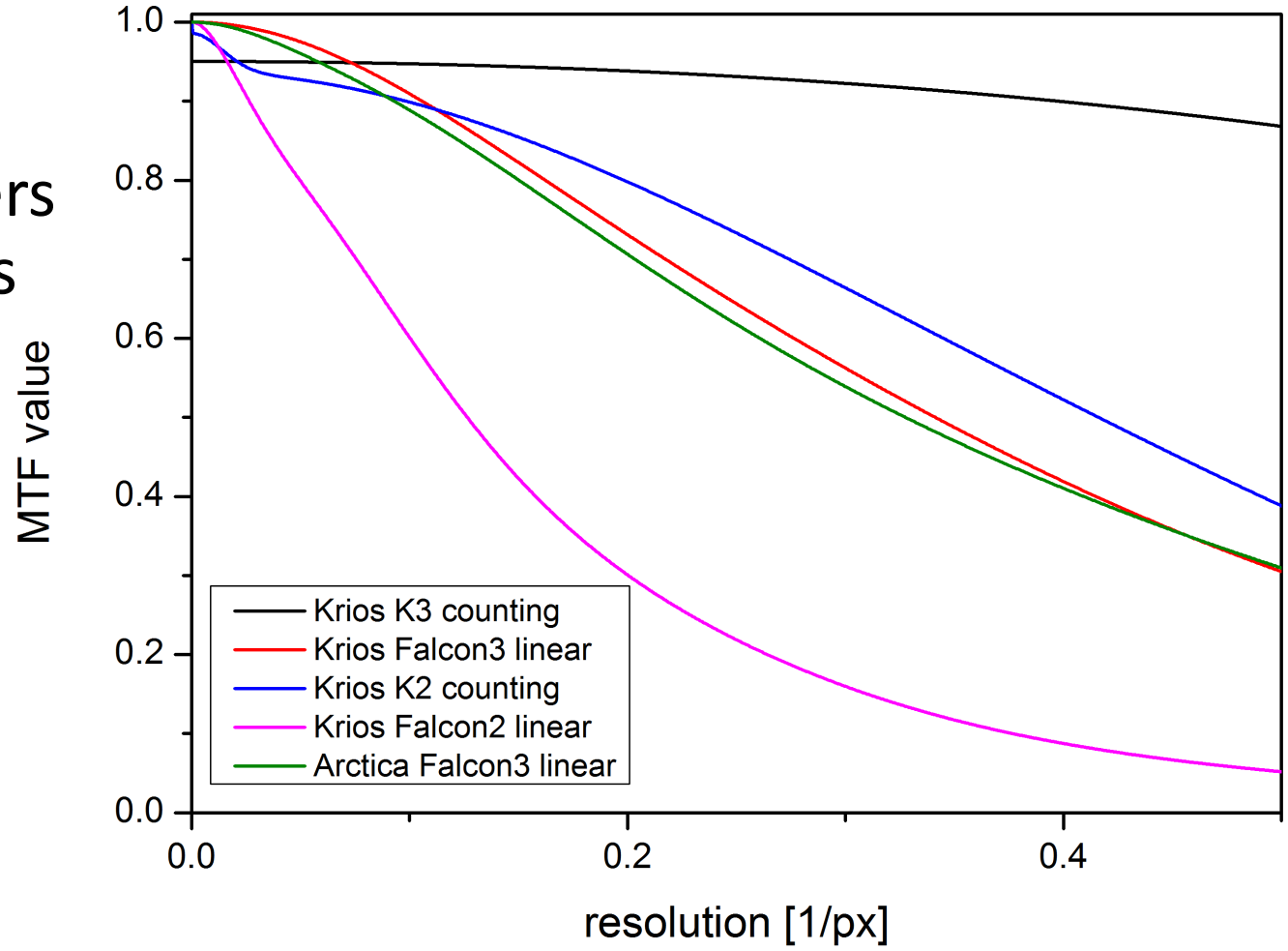
- SPA maps after refinement suffer from
  - Overrepresentation of the low frequencies
  - Attenuated hi frequencies
  - Unfiltered maps need to be properly lowpass filtered
  - Solvent filled areas are masked to avoid resolution underestimation
- Post-processing steps include
  - Masking
  - Resolution estimation
  - MTF correction – dividing
  - B-factor sharpening
  - FSC weighting
  - Optional: Local resolution estimation

Masking caveats

- Low FSC of solvent around the structure underestimates the global resolution (unstructured solvent acts as pure noise)
- Masking is a real space multiplication of map with a mask (<0,1> map)
  - Multiplication in Fourier space is convolution in real space
  - Multiplication in real space is convolution in Fourier space
- Avoid these mask properties:
  - Sharp edges (falling from 1 to 0 in a single step) – sharp edges need high-freq components in Fourier space to describe the change – introducing false correlation in high frequencies (FSC)
  - Avoid too tight mask - introducing false correlation in high frequencies (FSC)

MTF – Modulation Transfer Function

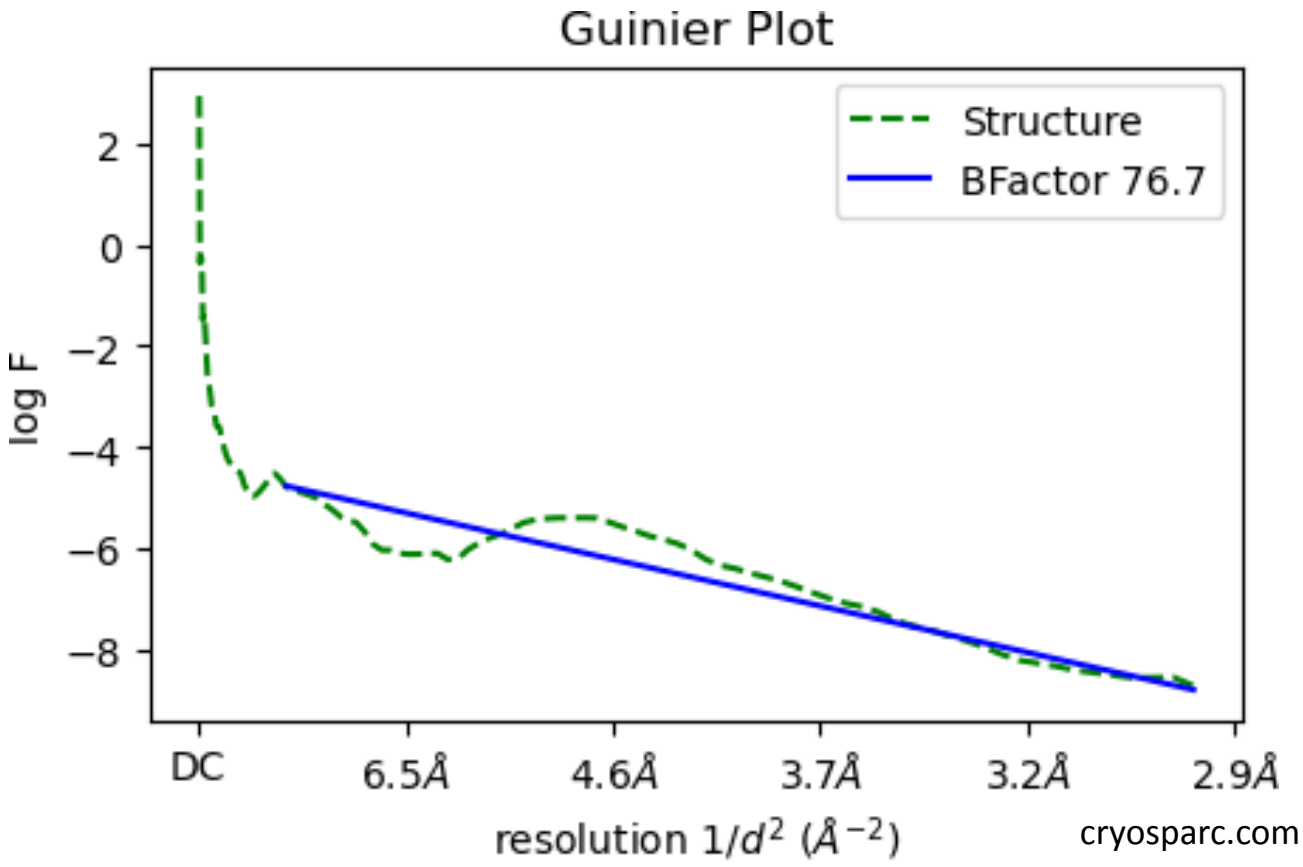
- Property of the detector
- Defines how well the detector transfers the information at certain frequencies
- DQE is proportional to MTF
- Dividing by MTF in Fourier space enhance hi-resolution information visibility



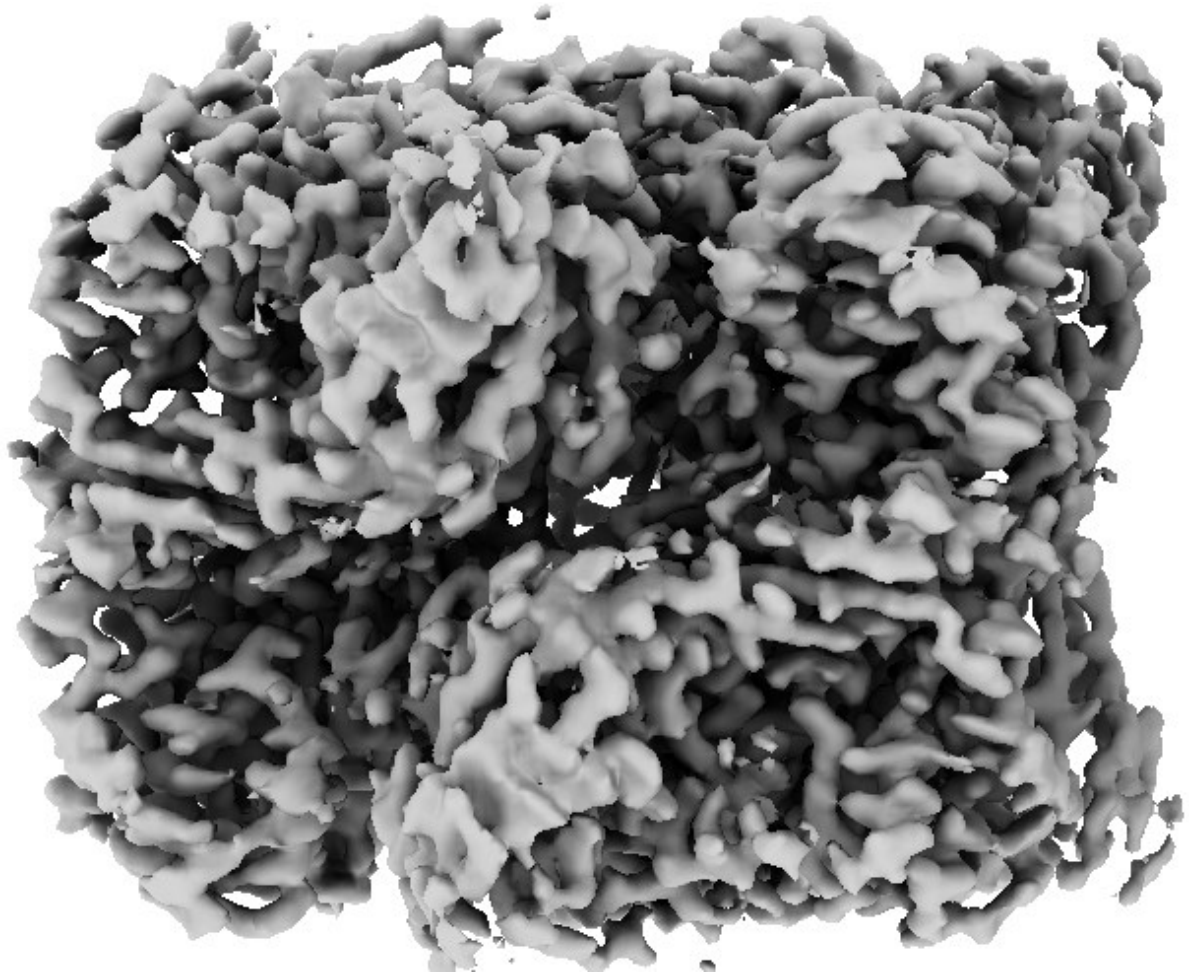
B-factor sharpening

$$F_{sharp}(k) = F_{map}(k) \cdot e^{-B(1/k)2}$$

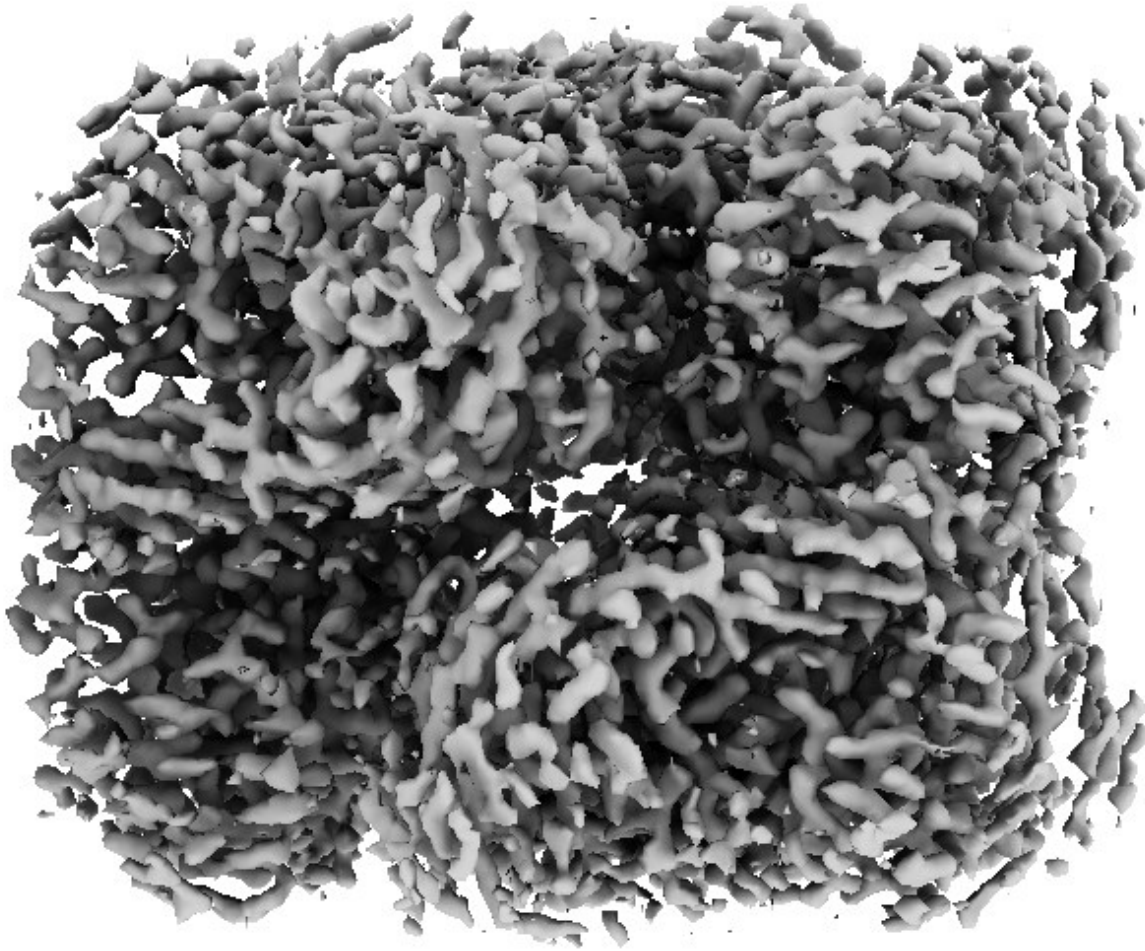
- Enhance the map interpretability
- Improves high frequency information visibility
- Does not improve map resolution
- Negative B-factor – sharpening
- Positive B-factor – blurring
- Map B-factor estimation
  - Automatic – form maps < 10 Å
  - User defined



Postprocessing



Map before postprocessing



Map after postprocessing