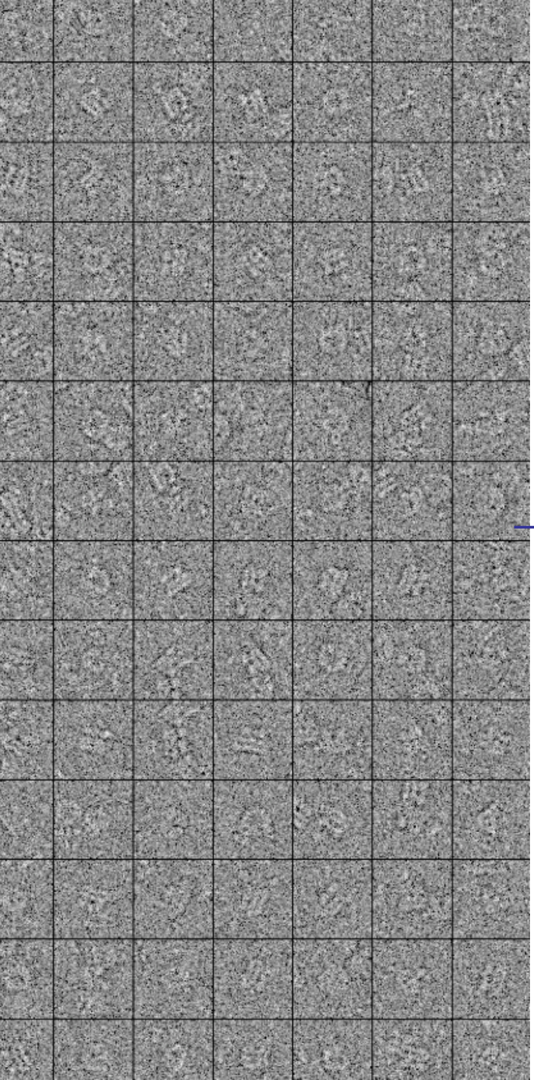
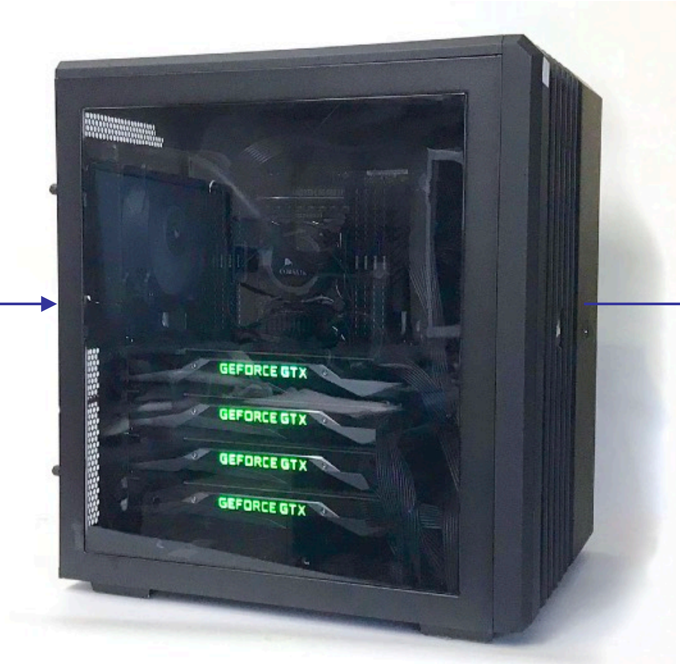
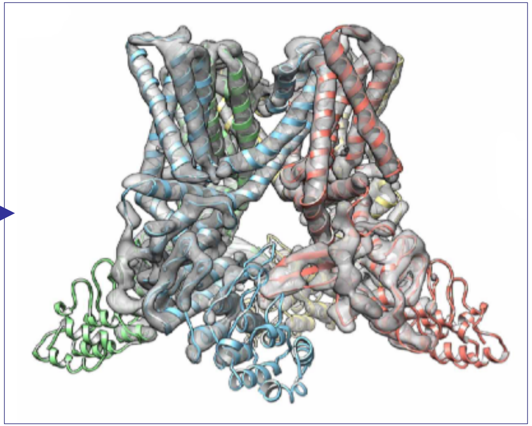
EM Image Formation and Single-Particle Reconstruction

Fred Sigworth

Yale University







Any sufficiently advanced technology is indistinguishable from magic.
-Arthur C. Clarke

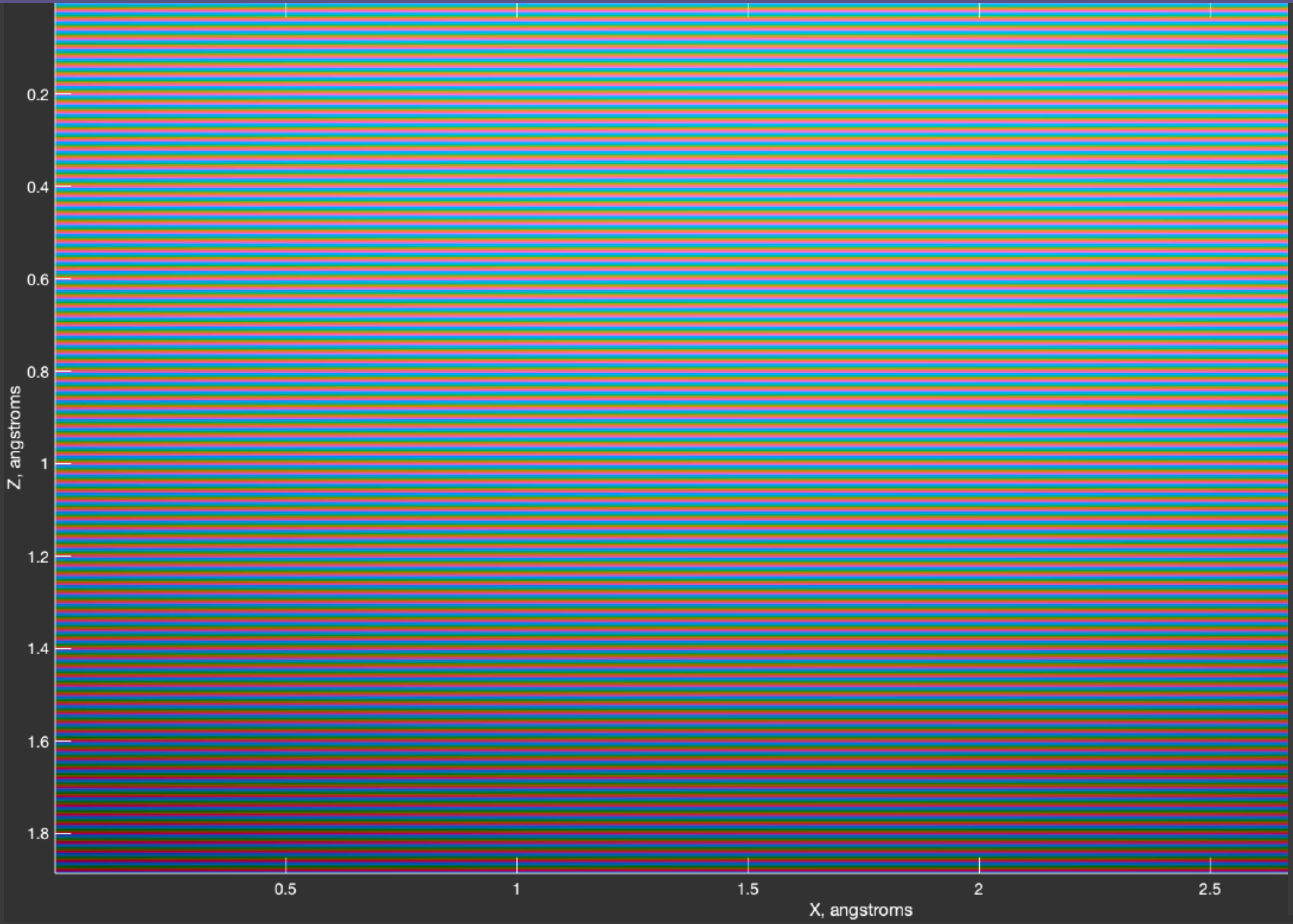
Defocus contrast and the CTF

Correlation and particle picking

Single-particle reconstruction

Maximum-likelihood methods

A snapshot of an electron wave



$\Psi_0 = e^{ikz}$

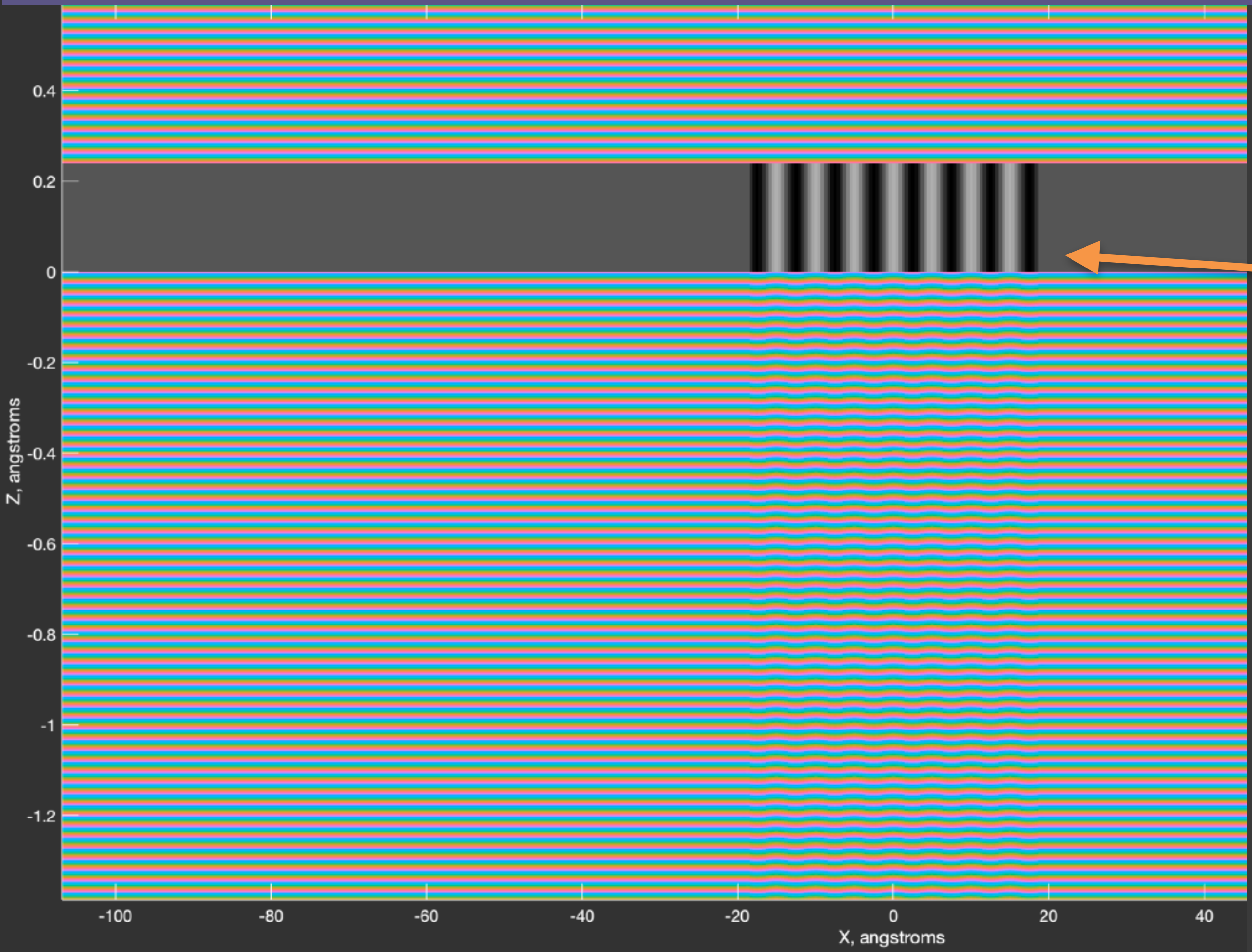
where

$k = 2\pi/\lambda$.

For 300keV electrons,

$\lambda = .02$ Å

Insert a phase-shifting object that perturbs the electron wave function



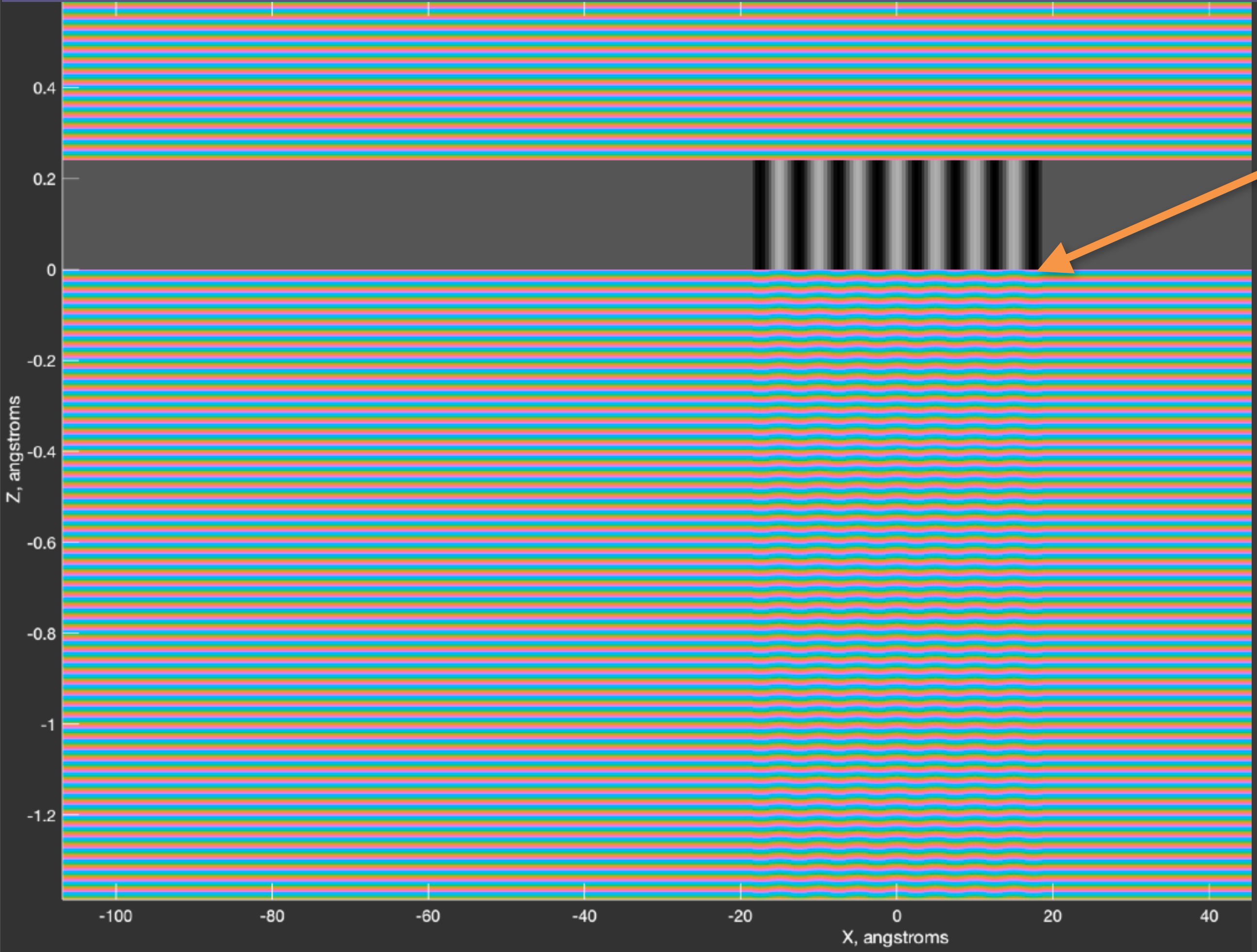
The object is a grating,

$\epsilon\phi(x) = \epsilon \cos(2\pi x/d)$.

In our example,

$d = 5\text{Å}$ and $\epsilon \ll 1$.

Insert a phase-shifting object that perturbs the electron wave function



At $z = 0$,

$$\Psi = e^{-i\epsilon\phi(x)}.$$

But the weak phase approximation* allows us to decompose Ψ into undiffracted and diffracted waves:

at $z = 0,$

$$\Psi \approx 1 - i\epsilon\phi(x).$$

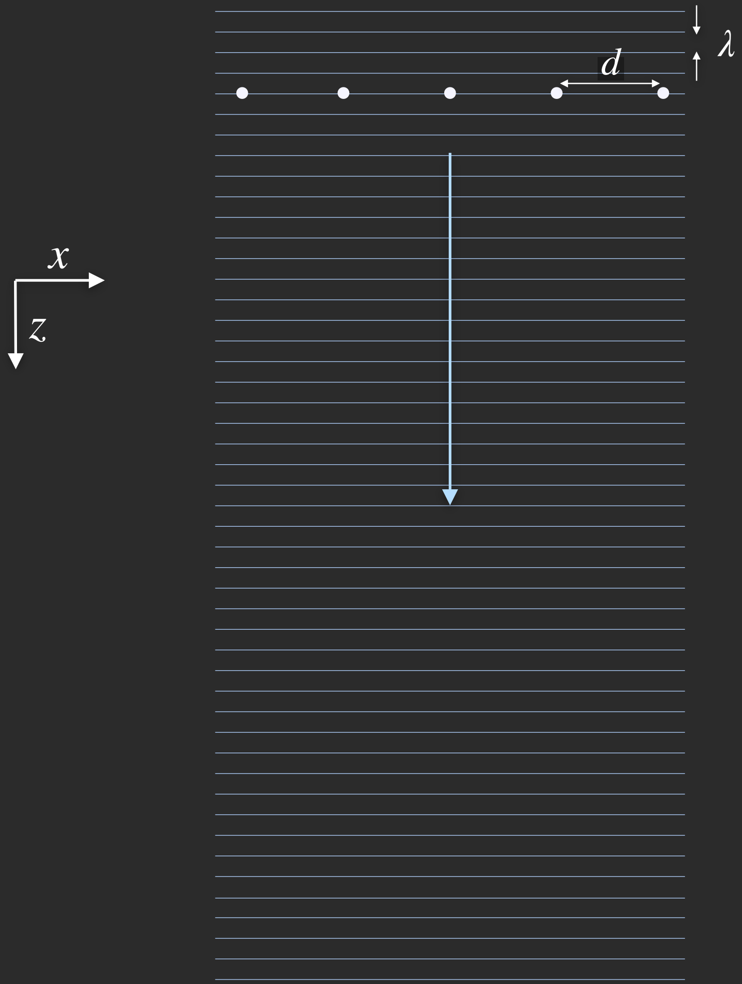
---

*This comes from the expansion

$$e^{iy} = 1 + iy - \frac{y^2}{2} + \dots$$

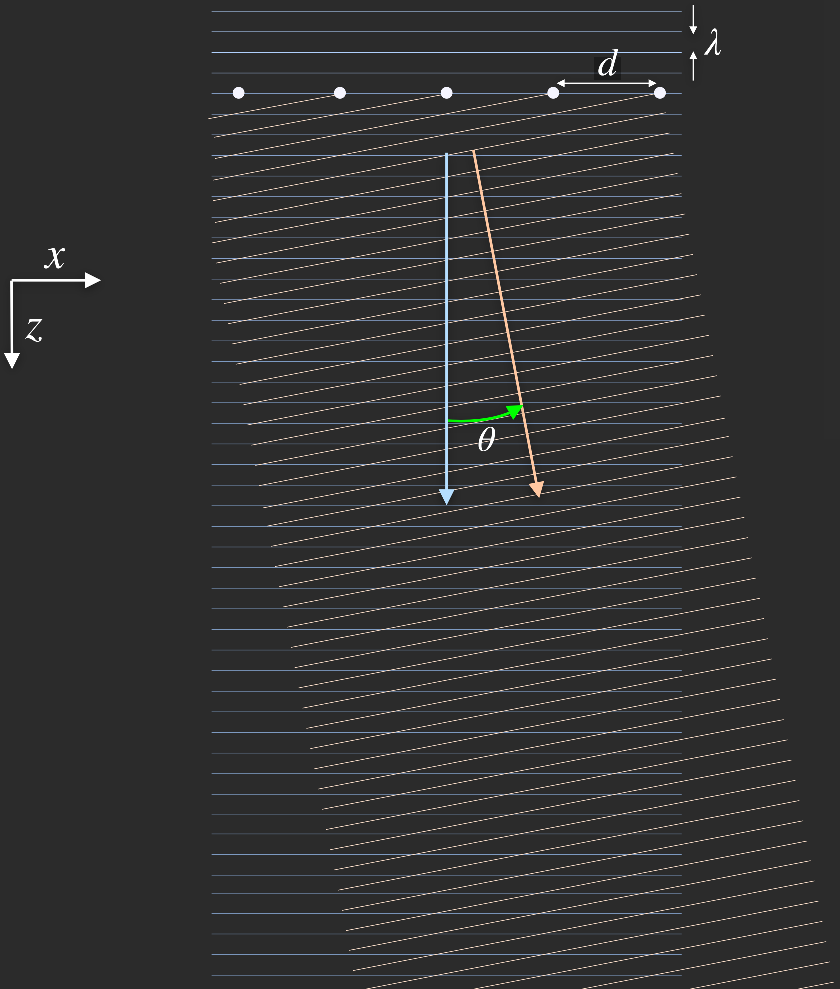
where terms y^2 and higher will be tiny, of order ϵ^2 or smaller.

Modeling the components of Ψ: first, the undiffracted wave

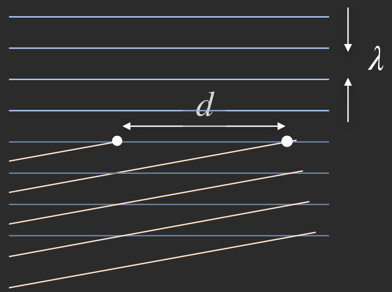


$$\Psi_0 = e^{ikz}$$

Classical diffraction yields a diffracted wave…



$$\sin\theta = \frac{\lambda}{d}$$



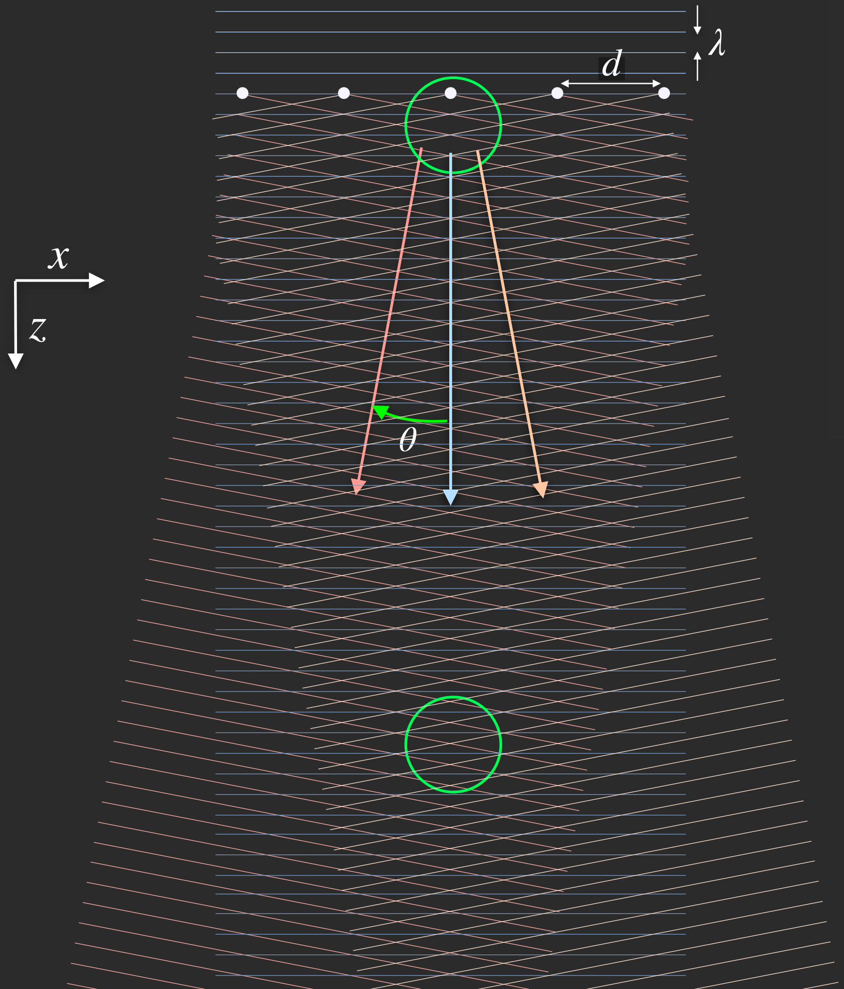
$$\Psi_0 = e^{ikz}$$

$$\Psi_+ = \frac{-i\epsilon}{2} e^{ik(z\cos\theta + x\sin\theta)}$$

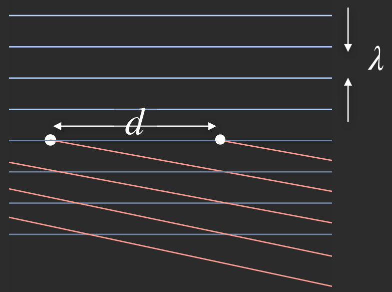
With $\lambda = .02\text{Å}$ and $d = 5\text{Å}$,

θ is actually only 4 milliradians, or 0.23°!

…and the other diffracted wave



$$\sin\theta = \frac{\lambda}{d}$$



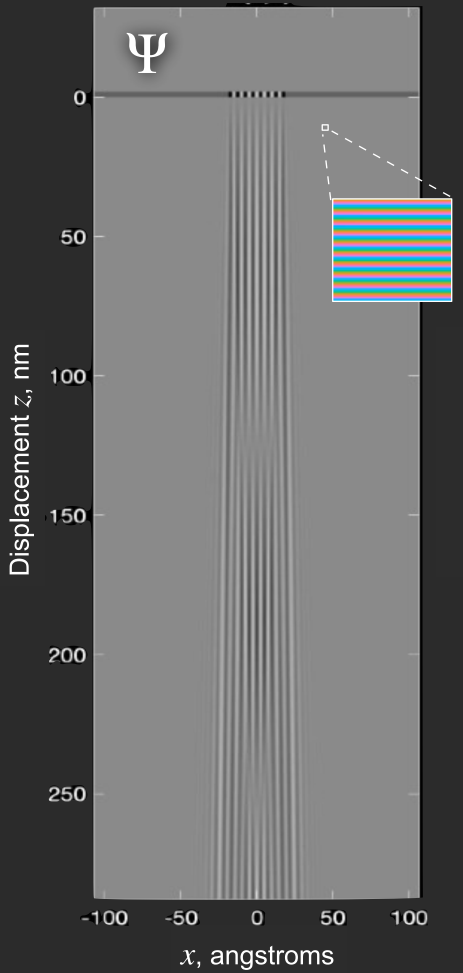
$$\Psi_0 = e^{ikz}$$

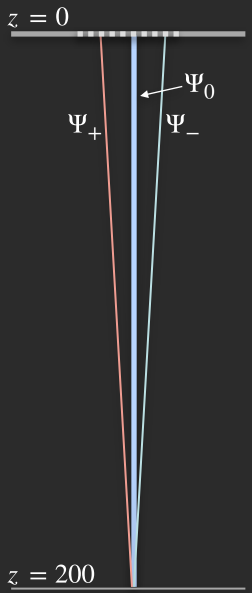
$$\Psi_+ = \frac{-i\epsilon}{2} e^{ik(z\cos\theta + x\sin\theta)}$$

$$\Psi_- = \frac{-i\epsilon}{2} e^{ik(z\cos\theta - x\sin\theta)}$$

Note there's a tiny shift of wavefronts, because the diffracted waves follow slightly longer paths.

The three waves interfere to make contrast





The complete wave function is

$$\Psi = \Psi_0 + \Psi_+ + \Psi_-$$

The relative phases change with z because Ψ_+ and Ψ_- have a different path length than Ψ_0 to arrive at a given point.

Suppose the Ψ_0 path length is

$$z_0 = 200\,\text{nm}.$$

The Ψ_- and Ψ_+ path lengths are then

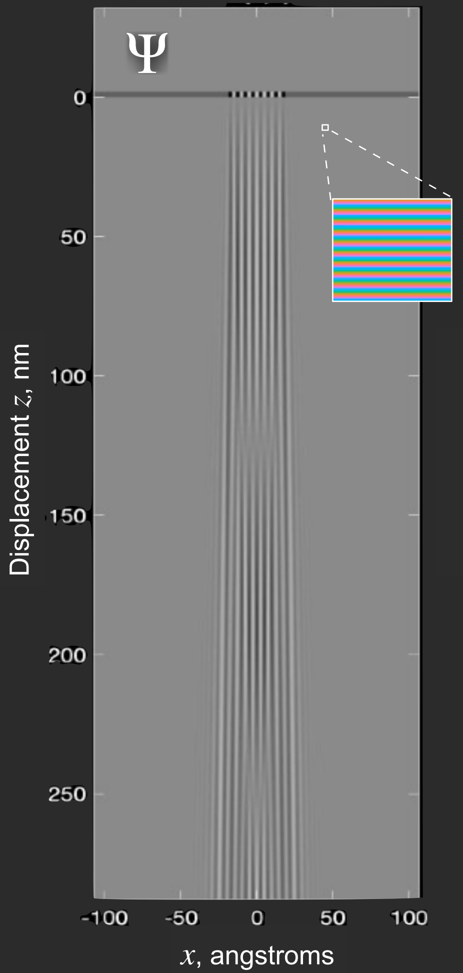
$$z_{\pm} = 200\,\text{nm}/\cos\theta$$

$$= 200.0016\,\text{nm}$$

This is a significant difference, since

$$\lambda = .002\,\text{nm}!$$

The diffracted waves have a slightly larger path length



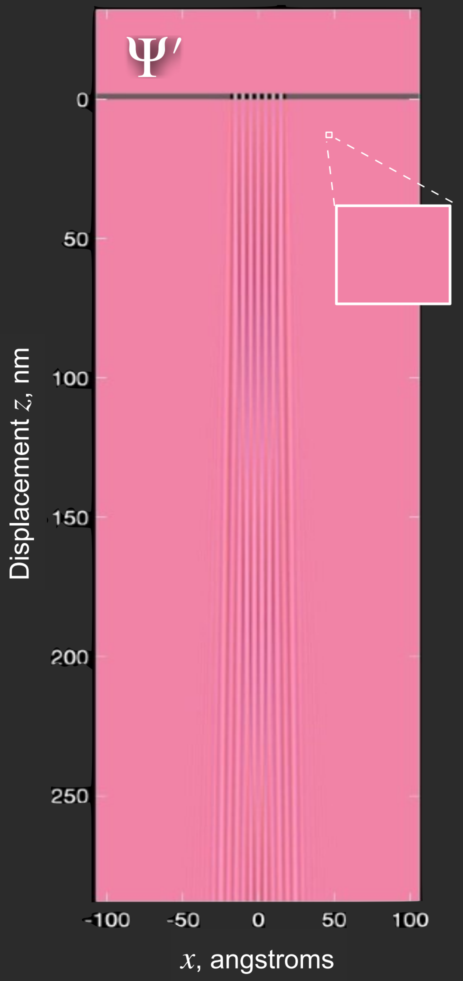
The path length difference is

$$\zeta = z_{\pm} - z_0 = \left(1 - \frac{1}{\cos\theta}\right) z$$

For $z \geq 0$,

$$\Psi = \Psi_0 + \Psi_+ + \Psi_- = e^{ikz} - i\epsilon\phi(x)e^{ik(z+\zeta)}.$$

Removing the fast oscillations to visualize relative phases



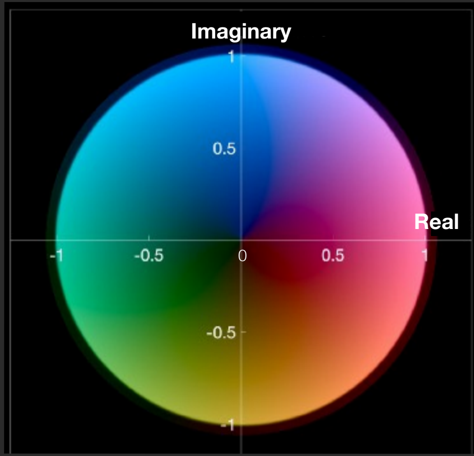
$$\Psi = \Psi_0 + \Psi_+ + \Psi_-$$
$$= e^{ikz} - i\epsilon\phi(x)e^{ik(z+\zeta)}$$

To remove the fast oscillations, Let's cancel them! Define

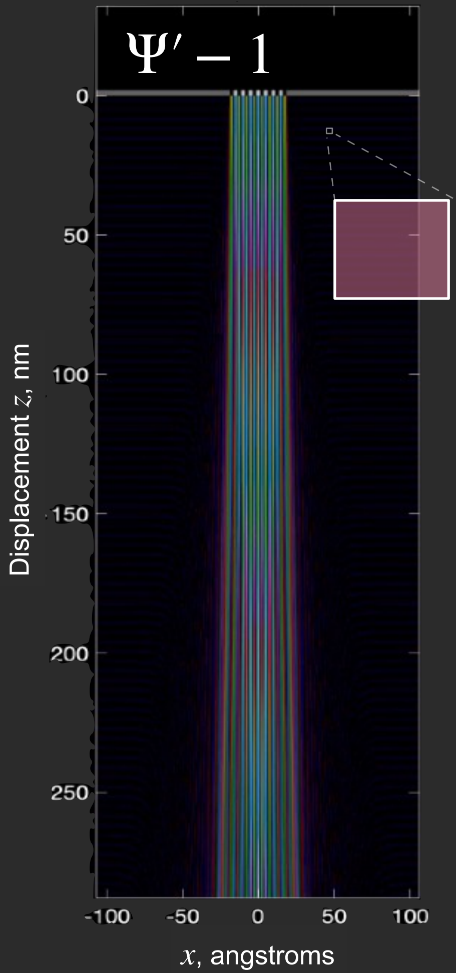
$$\Psi' = \Psi e^{-ikz}$$
$$= 1 - i\epsilon\phi(x)e^{ik\zeta}$$

The intensity of the electron waves is unchanged,

$$I = |\Psi|^2 = |\Psi'|^2$$



The diffracted waves alone

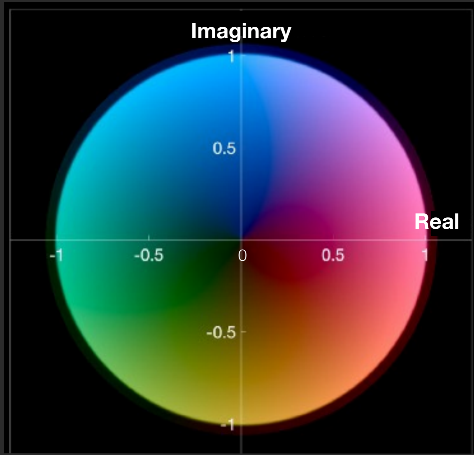


$$\Psi' = 1 - i\epsilon\phi(x)e^{ik\zeta}$$

Then, subtracting away 1 which represents the unscattered wave, you can observe

- the variation in x, because our original grating signal was $\epsilon\phi(x) = \epsilon\cos(2\pi x/d)$
- the phase variation along z from the complex exponential $e^{ik\zeta}$.

Next we'll see that we get contrast at z values where the diffracted waves have real values (red and green), not imaginary (blue and yellow).



The contrast comes from interference in the real part of Ψ'

$$\Psi' = 1 - i\epsilon\phi(x)e^{ik\zeta}$$
$$= 1 - i\epsilon\phi(x)[\cos(k\zeta) + i\sin(k\zeta)]$$

The measured intensity is

$$|\Psi'|^2 = (\text{real part})^2 + (\text{imag part})^2$$
$$= \left[1 + \epsilon\phi(x)\sin(k\zeta)\right]^2 - \left[\epsilon\phi(x)\cos(k\zeta)\right]^2$$
$$= \left[\boxed{1 + 2\sin(k\zeta)\,\epsilon\phi(x)} + \mathcal{O}\epsilon^2\right] + \left[\mathcal{O}\epsilon^2\right].$$

So, ignoring a factor of 2, we say the contrast transfer from phase shift to intensity change is

$$\boxed{\text{Contrast} = \frac{\text{intensity change}}{\text{phase shift}} = \sin(k\zeta)}$$

The contrast transfer function for defocus

$$\text{Contrast} = \frac{\text{intensity change}}{\text{phase shift}} = \sin(k\zeta)$$

$$k = 2\pi/\lambda$$

$$\zeta = (1 - \frac{1}{\cos\theta})\, z,$$

but a very useful approximation is

$$\zeta = \frac{\lambda^2 z}{2d^2}$$

so

$$k\zeta = \pi\lambda z/d^2$$

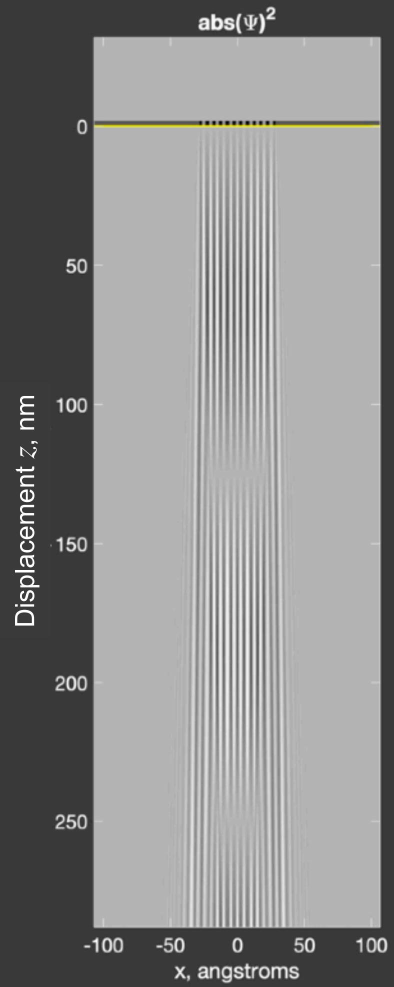
and

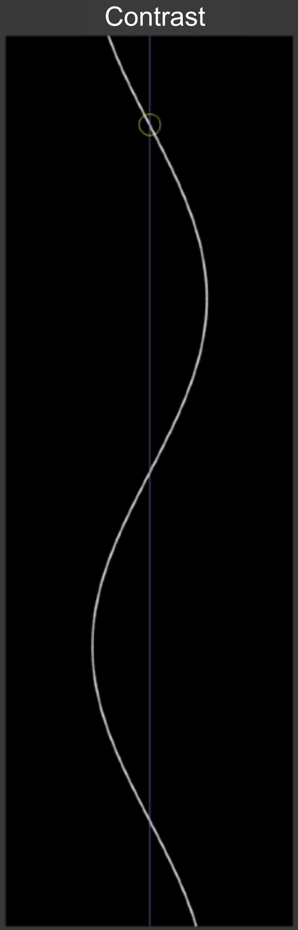
$$\text{CTF} = \sin(\pi\lambda z/d^2)$$ the defocus contrast transfer function

Contrast varies with the amount of defocus

Intensity at z

The original grating with $d = 5\text{Å}$

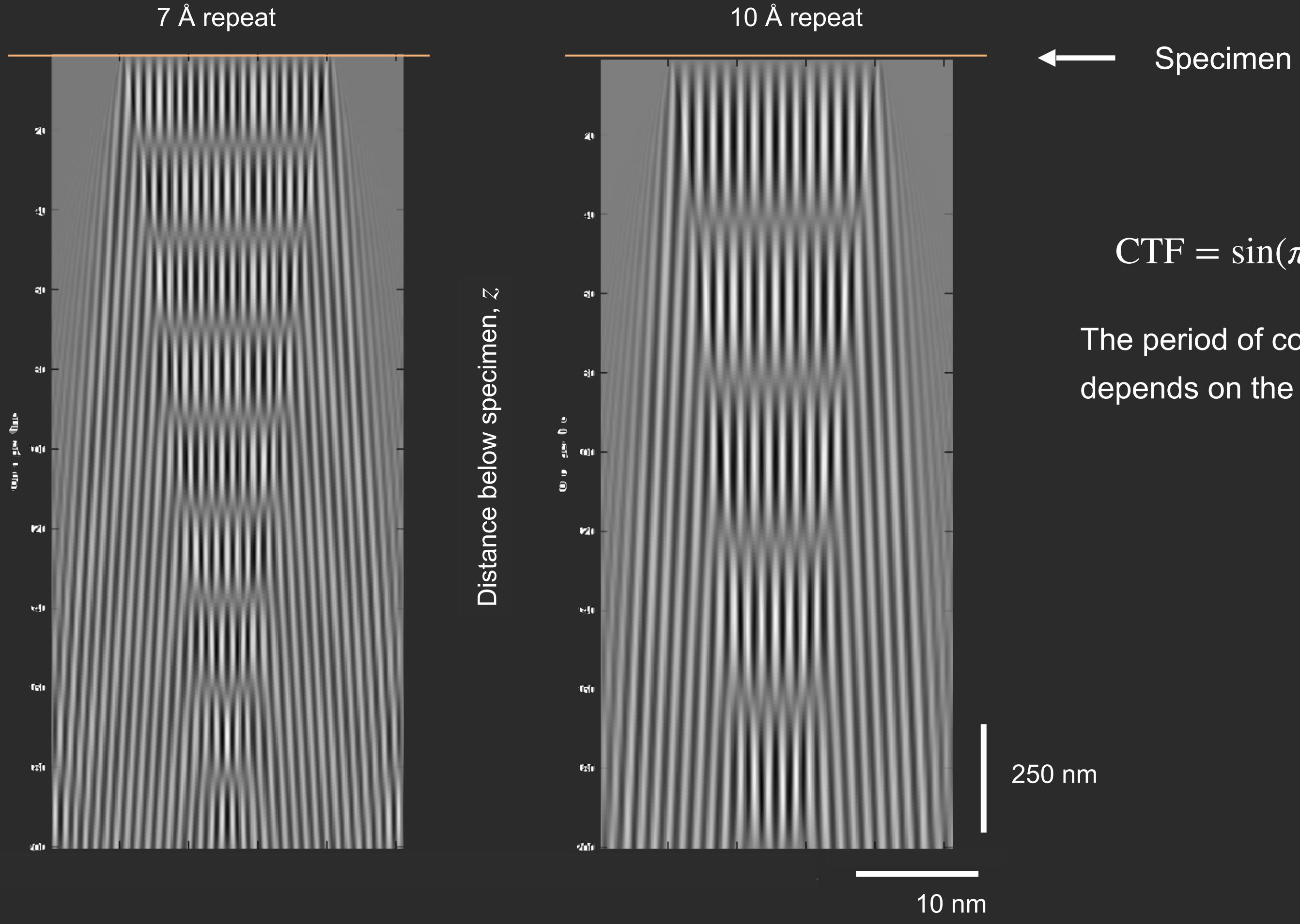




Interference between the unscattered wave and the diffracted waves produces contrast.

CTF = $\sin(\pi \lambda z / d^2)$

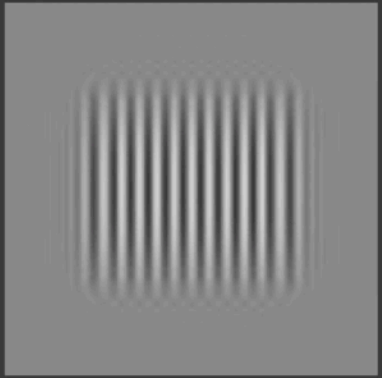
Periodicity of contrast depends on the grating spacing d



$$\mathrm{CTF} = \sin(\pi \lambda z / d^2)$$

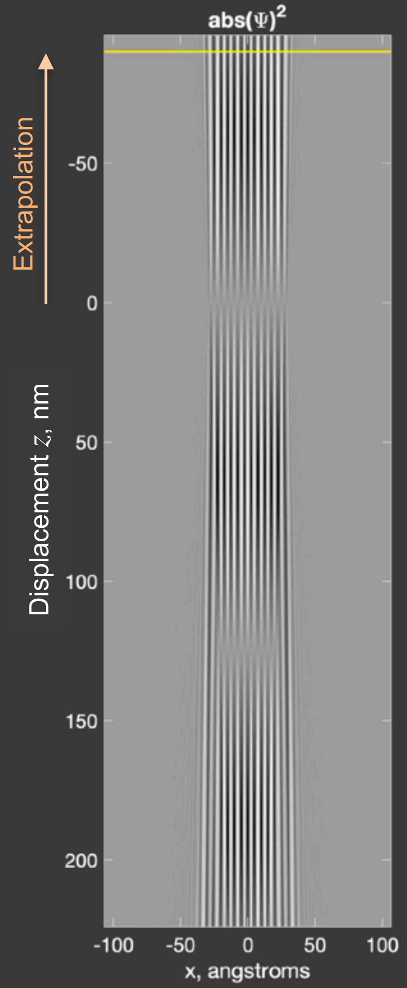
The period of contrast variations depends on the grating period d.

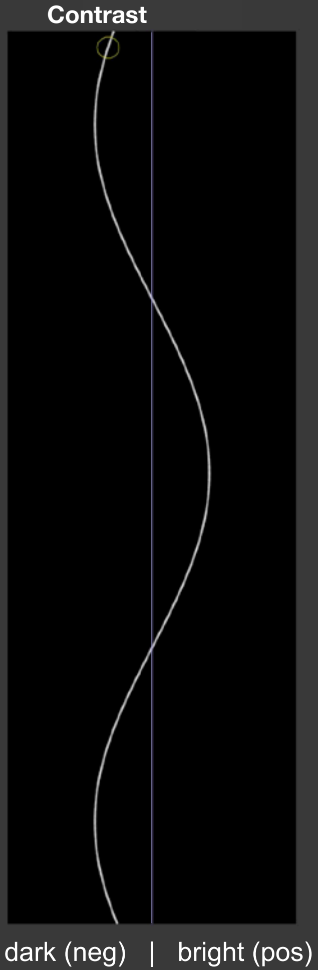
What happens when the objective lens is focused *above* the specimen?



Intensity at z

The grating $\phi(x)$





Underfocus is focusing the objective lens above the specimen.

Standard terminology

- Defocus values δ are positive for underfocus,

$$\delta = -z$$

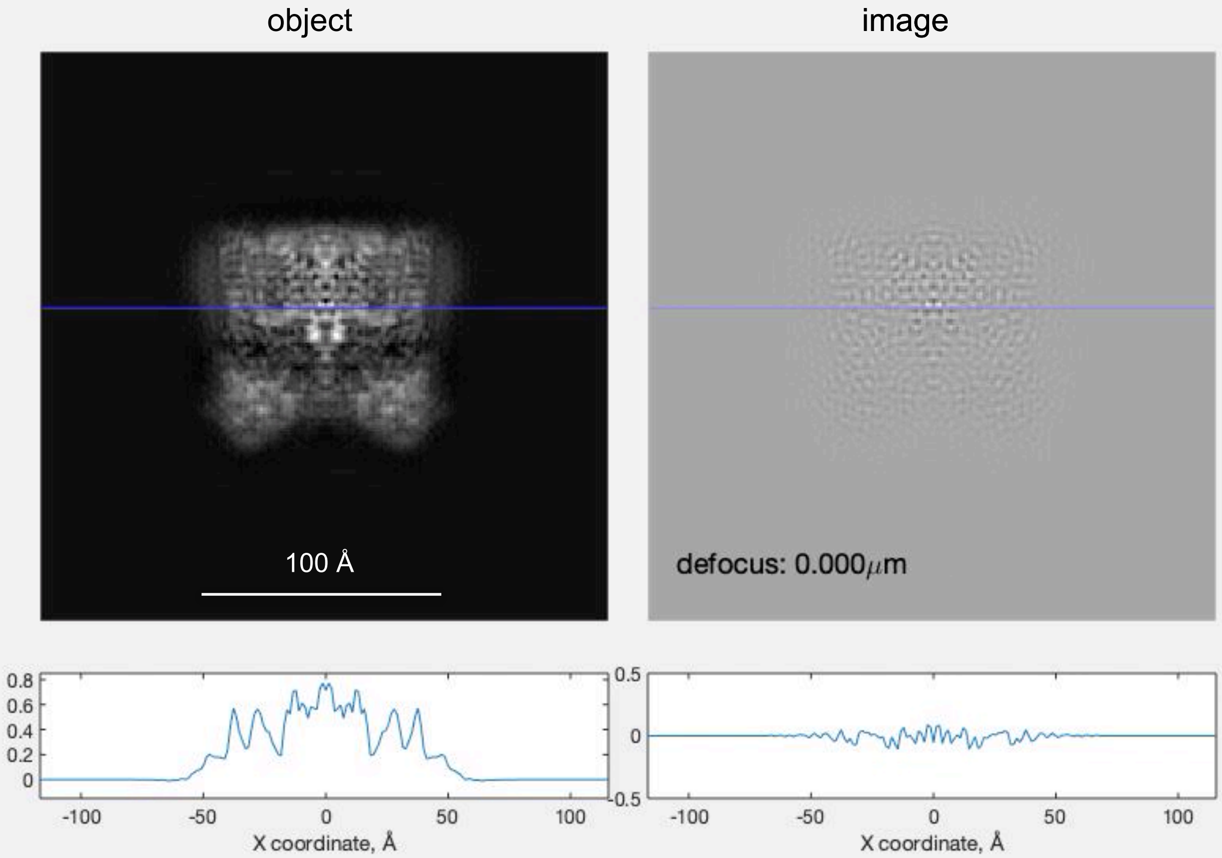
- Spatial frequency is

$$s = 1/d$$

- So we can write the defocus phase contrast as:

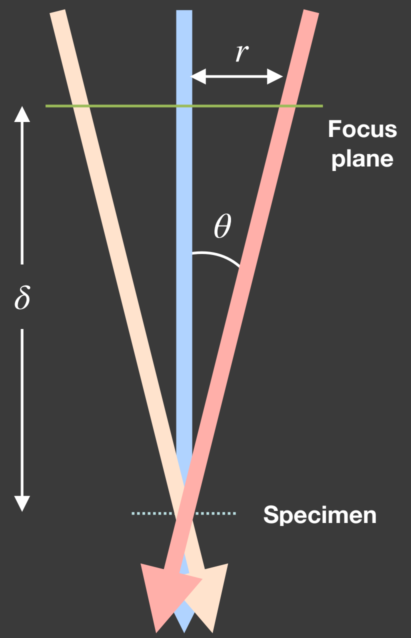
$$\mathrm{CTF} = \sin(-\pi\lambda\delta s^2)$$

Most cryo-EM data are acquired using defocus contrast



- People always use "underfocus". This means decreasing the strength of the objective lens, effectively focusing above the specimen.
- At high defocus, high-resolution information in the image is strongly delocalized.
- If the delocalized information is not lost by cropping, image processing can recover it.

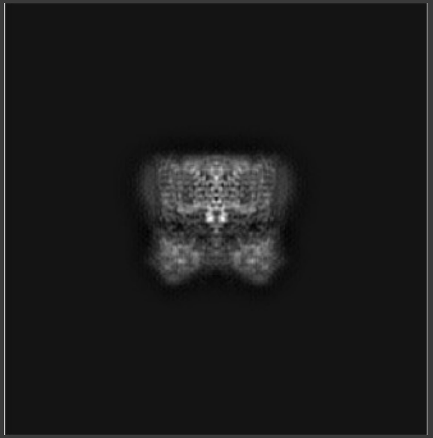
With large defocus, how bad is the image delocalization?



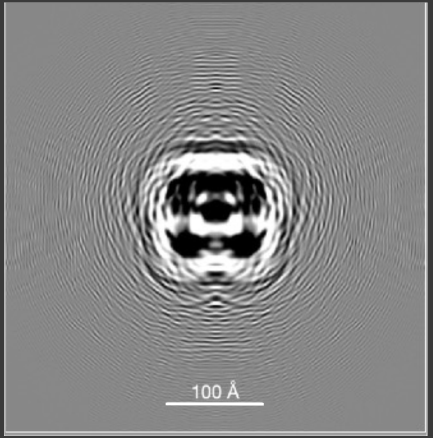
The dispersion radius is given by

$$r = \delta \tan\theta$$
$$= \delta\lambda s \quad \text{(small angle approx.)}$$

- How big a box do I need around my 120 Å-diameter particle to include all the information up to 3Å, if I use 3µm of defocus and $\lambda = .02\text{Å}$
- In this case $r = 200\text{Å}$, so to capture all the information the box should be 520Å on a side.



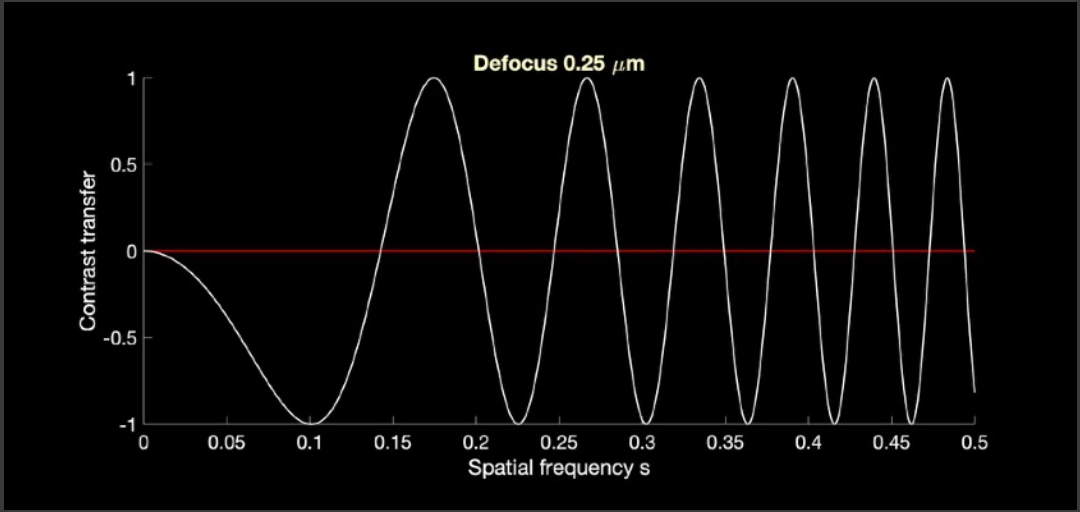
Object



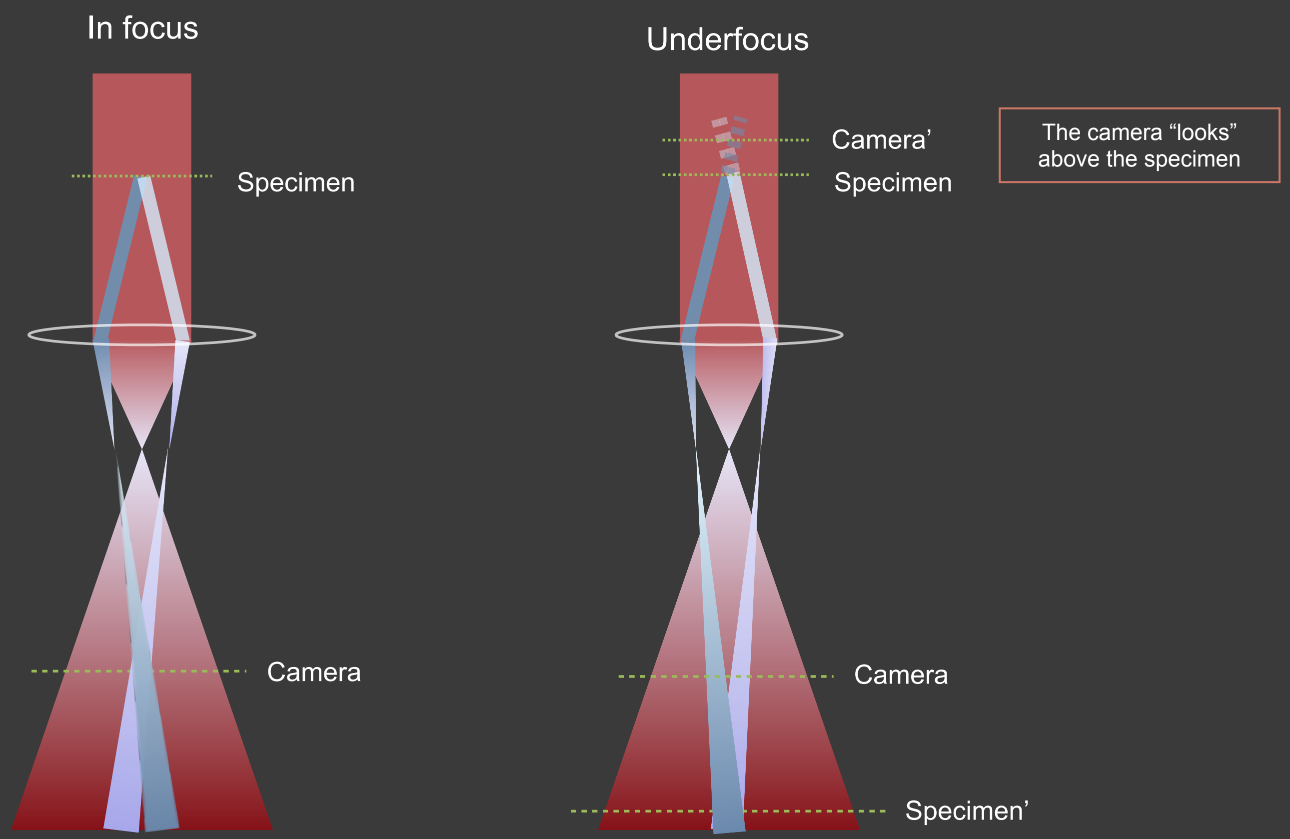
3 µm defocus

Contrast transfer function as a function of frequency $s = 1/d$

$$\text{CTF} = \sin(-\pi\lambda\delta s^2)$$



An objective lens reproduces interference patterns at the focus



With spherical aberration a lens bends high-angle rays more strongly

Spherical aberration changes the defocus by

$$\delta' = -C_s \lambda^2 s^2/2.$$

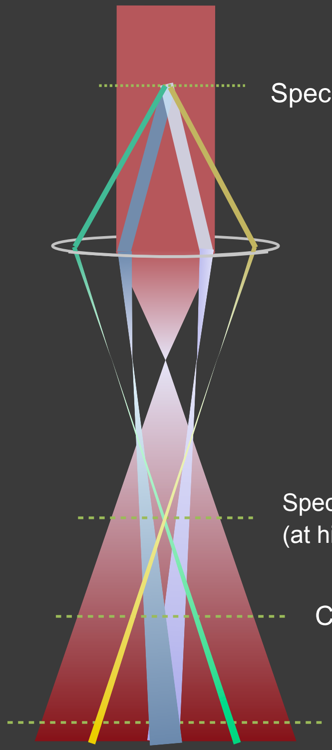
The contrast transfer function has a new term,

$$\mathrm{CTF} = \sin(-\pi\lambda\,(\delta + \delta')\,s^2)$$

or, expanded,

$$\mathrm{CTF} = \sin(-\pi\lambda\delta s^2 + \frac{\pi}{2} C_s \lambda^3 s^4)$$

The coefficient C_s is typically ~2mm. This makes spherical aberration important only for $s \gtrsim 0.25\text{Å}^{-1}$, or about 4Å resolution.



Very high-angle scattering yields some contrast

Electrons that pass very close to an atomic nucleus are scattered at high angles, and are caught by the objective aperture.

The loss of these electrons results in a small amount of negative amplitude contrast.

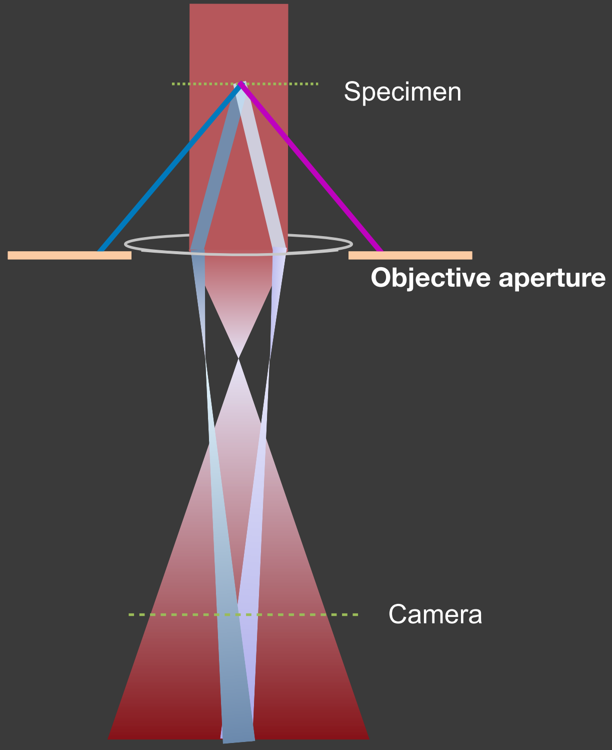
Its small magnitude, $\sin(-\alpha)$, is typically around -0.05.

The amplitude contrast term allows the CTF to have a small negative value even at zero spatial frequency.

Combining all these terms, the contrast transfer function is given by

$$\mathrm{CTF} = \sin\left(-\pi\lambda\delta s^2 + \frac{\pi}{2}C_s\lambda^3 s^4 - \alpha\right)$$

defocus, sphere abb., amplitude

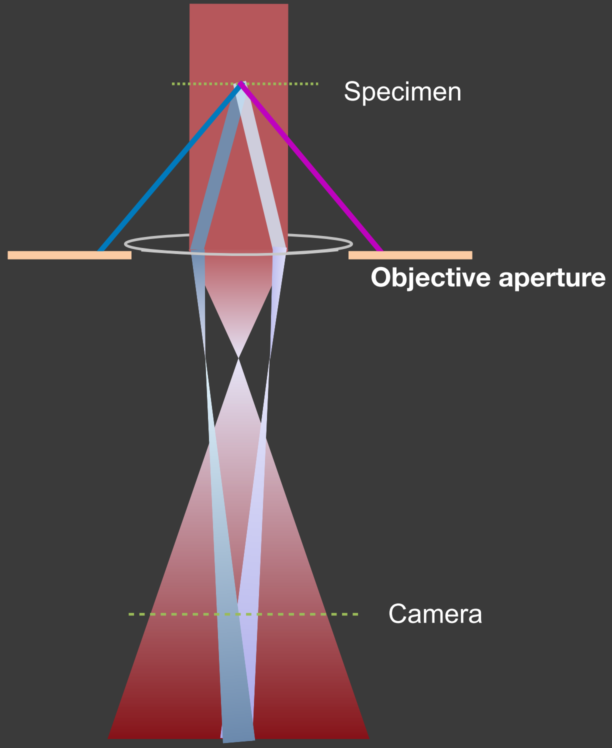


Parameters of the CTF

$$\mathrm{CTF} = \sin\left(-\pi\lambda\delta s^2 + \frac{\pi}{2}C_s\lambda^3 s^4 - \alpha\right)$$

defocus sphere abb. amplitude

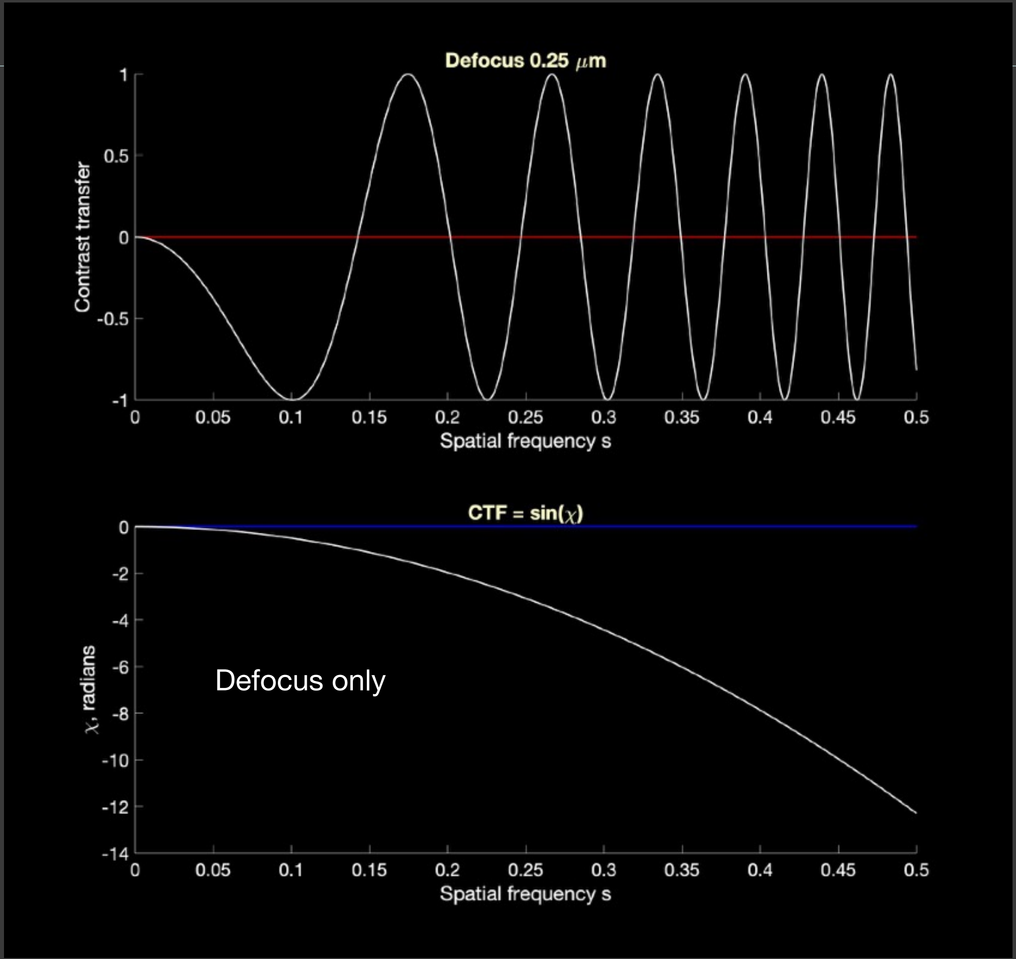
- C_s is a property of the microscope objective lens; you can ask the manufacturer for the value.
- α is a property of the atoms in the specimen. For proteins, its value can be assumed to lie between .05 and .1
- δ is the only parameter that must be estimated to high precision for each micrograph. If there is astigmatism, δ is a function of an in-plane angle θ, so one needs to estimate δ_{min}, δ_{max} and θ.



This is our simple defocus-contrast CTF

Simple defocus contrast

$$\mathrm{CTF} = \sin(-\pi\lambda\delta f^2)$$

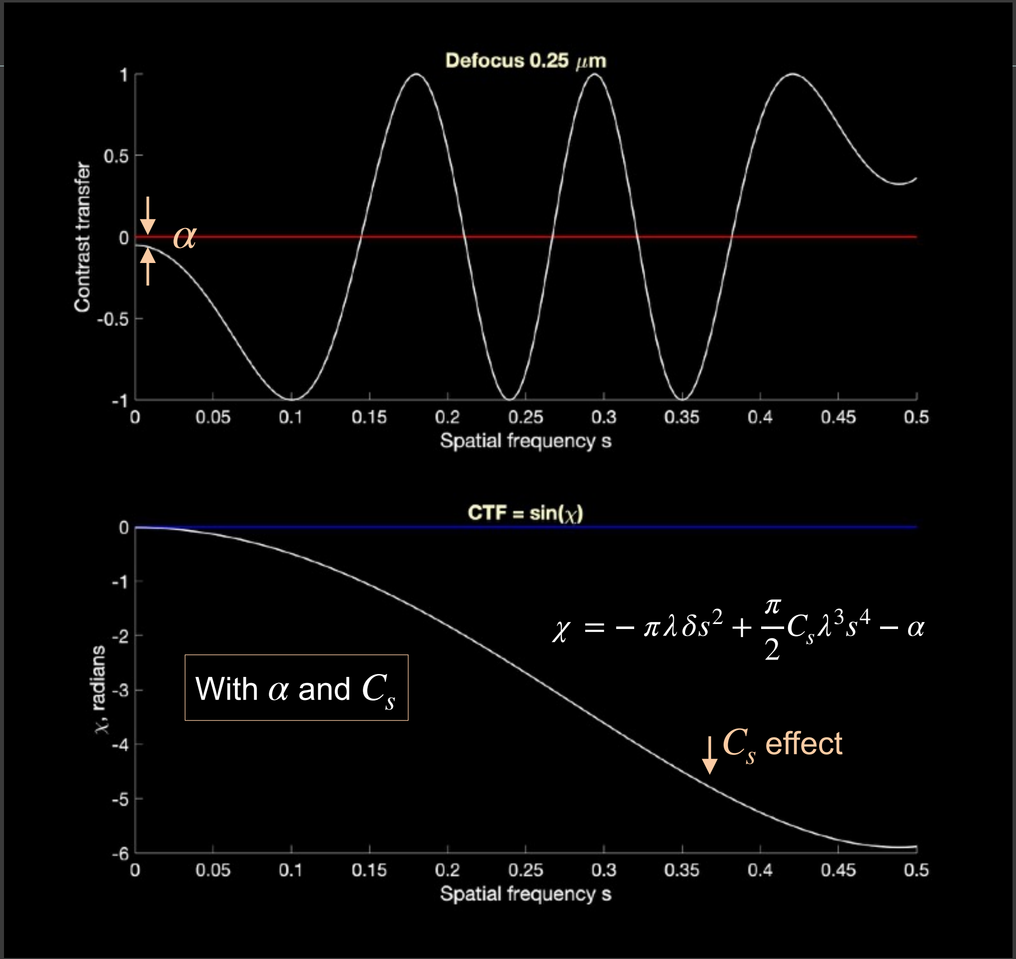


Up to a resolution of ~2Å three terms of the CTF suffice

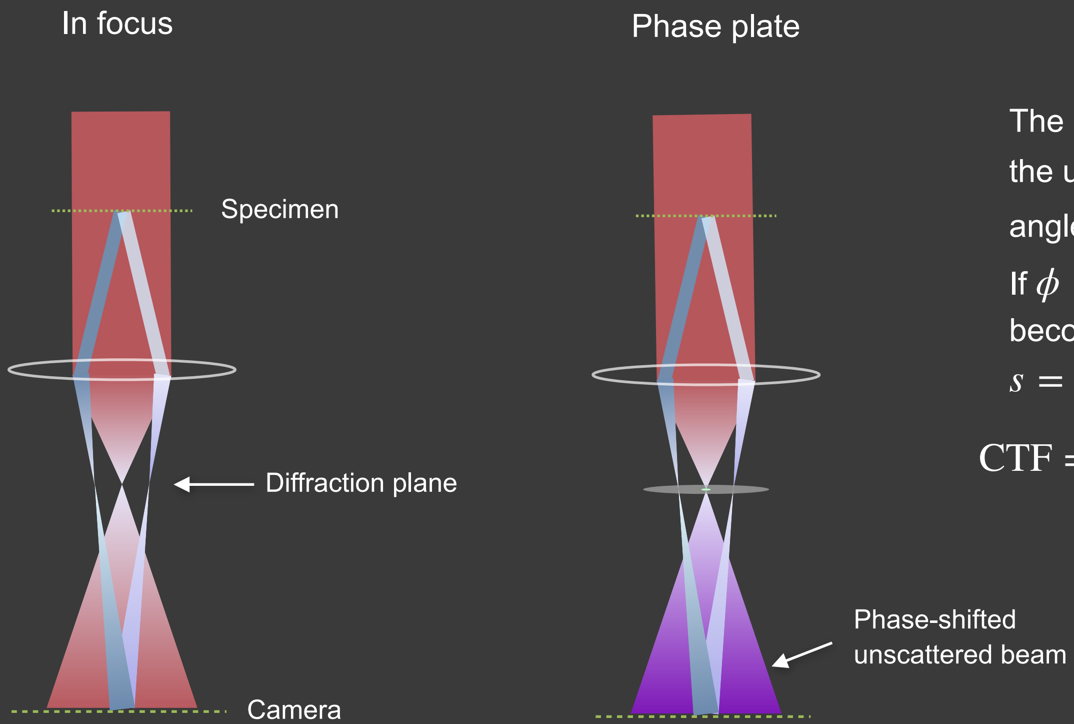
Combining all these terms, the contrast transfer function is given by

$$\mathrm{CTF} = \sin\left(-\pi\lambda\delta f^2 + \frac{\pi}{2}C_s\lambda^3 f^4 - \alpha\right)$$

Here you can see why everyone uses underfocus: the amplitude contrast and defocus contrast are additive at low *s*.



A phase plate modifies the interference of electron waves at the camera

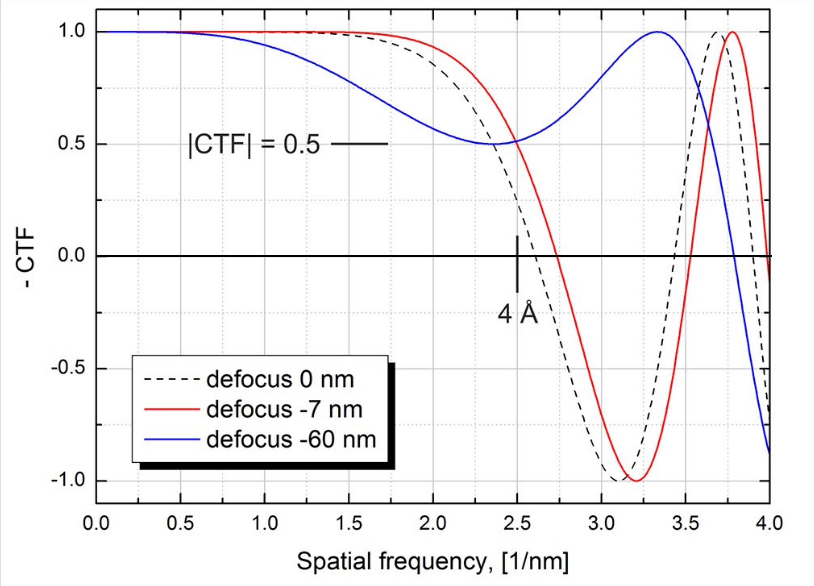


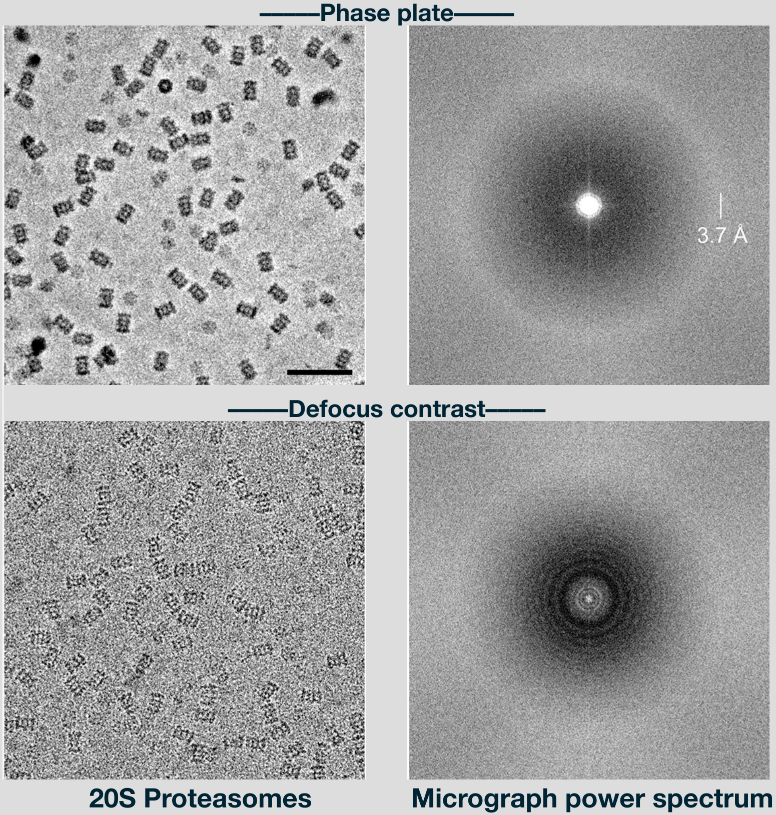
The phase plate shifts the phase of the undiffracted beam Ψ_0 by some angle ϕ.

If $\phi = 90^{\circ}$ then the sine function becomes a cosine, and the CTF at $s = 0$ becomes 1.

$$\mathrm{CTF} = \sin(\phi - \pi\lambda\delta f^2 + \frac{\pi}{2}C_s\lambda^3 f^4 - \alpha)$$

The phase plate allows in-focus imaging, but precise focusing is necessary.





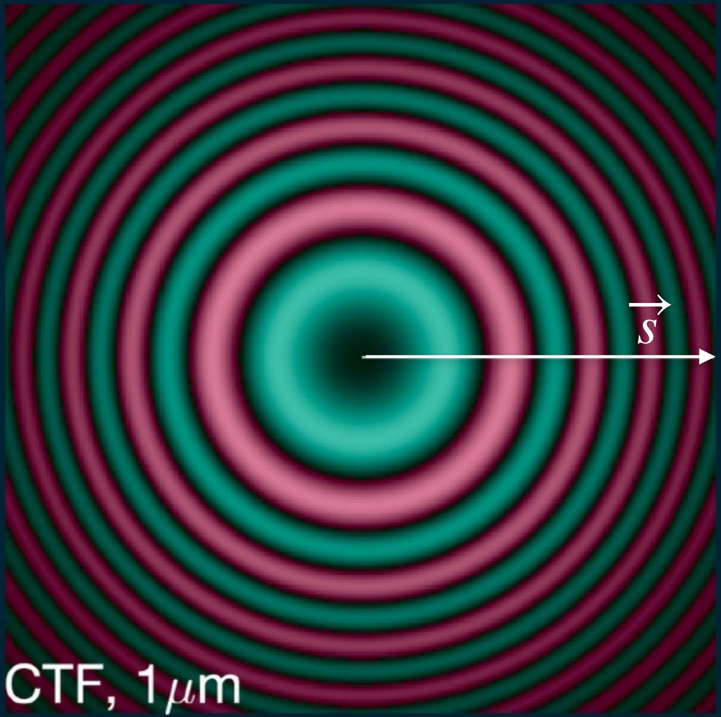
Cryo-EM single particle analysis with the Volta phase plate

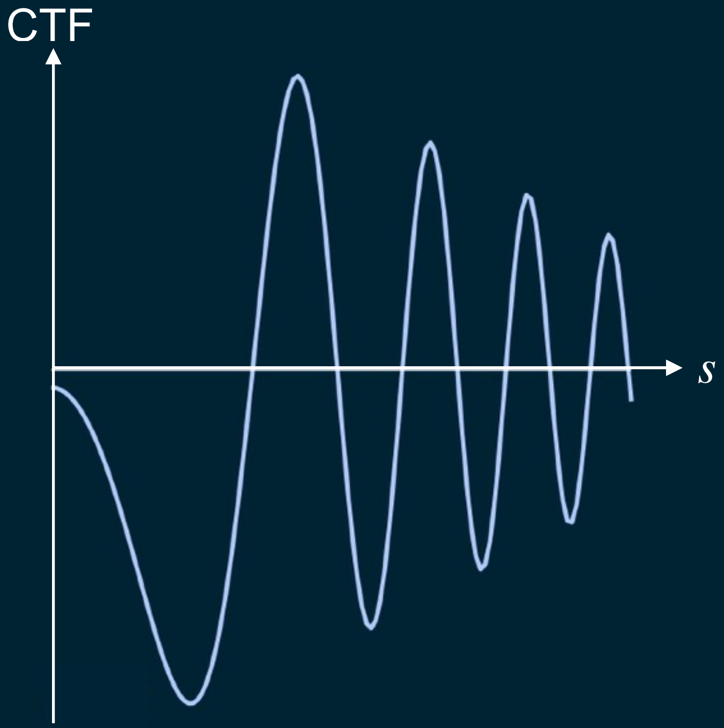
Radostin Danev*, Wolfgang Baumeister

Department of Molecular Structural Biology, Max Planck Institute of Biochemistry, Martinsried, Germany

eLife 2016

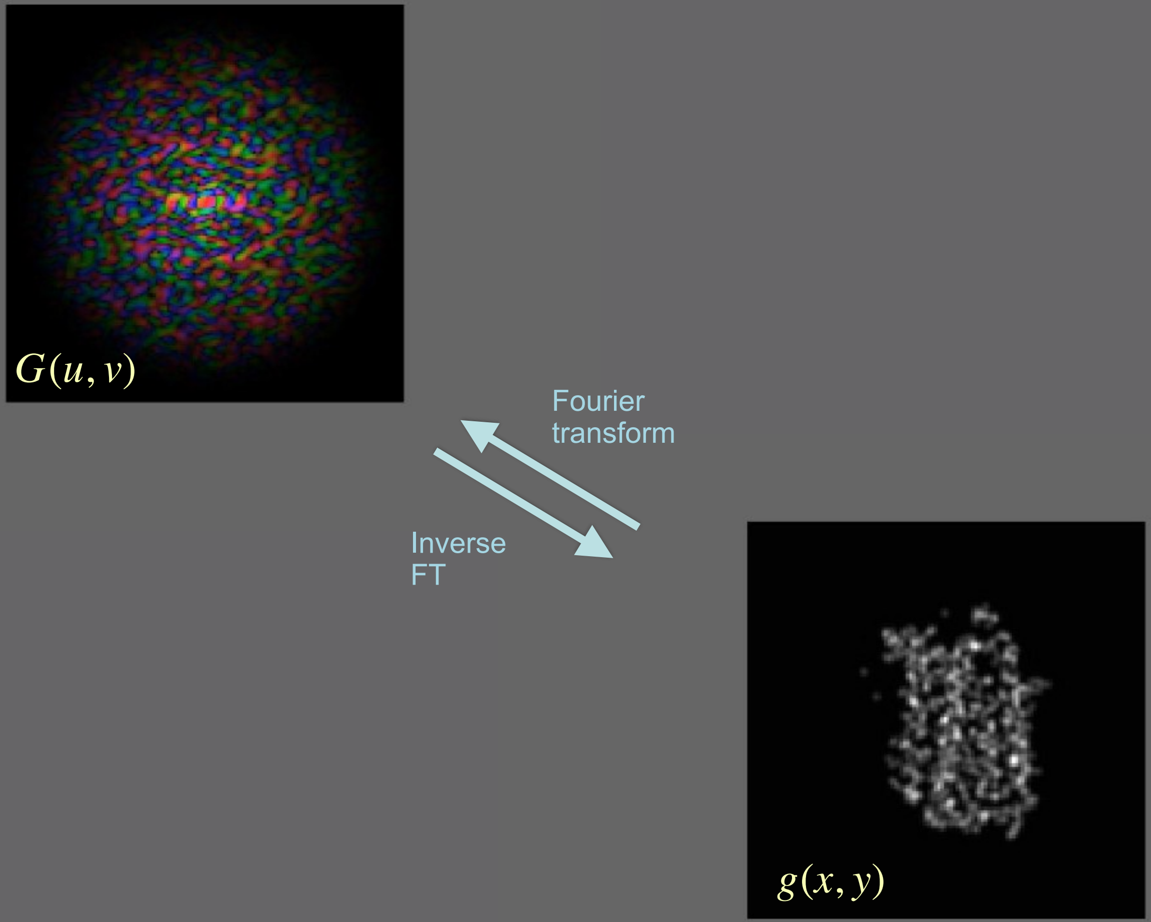
The CTF representation in 2D Fourier space





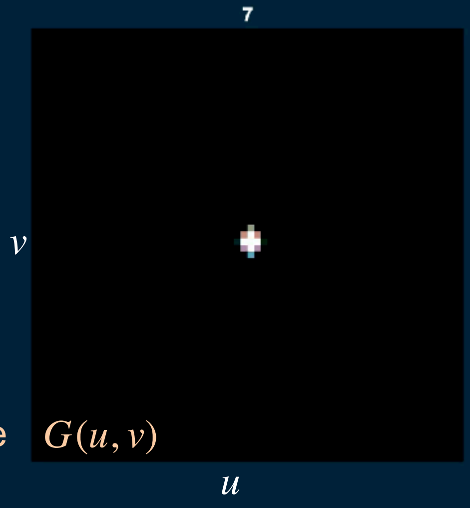
If astigmatism were present, the rings would be ellipses.

Every image has a 2D Fourier transform



Each point in the Fourier plane represents a grating “frequency”

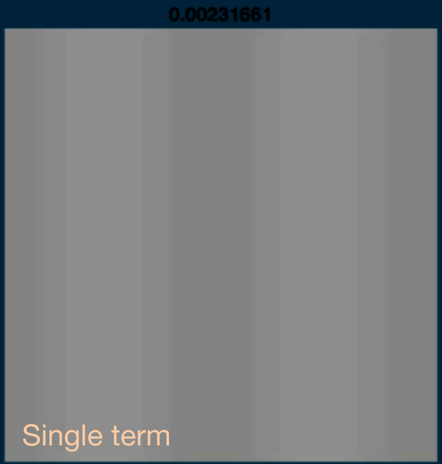
Fourier space



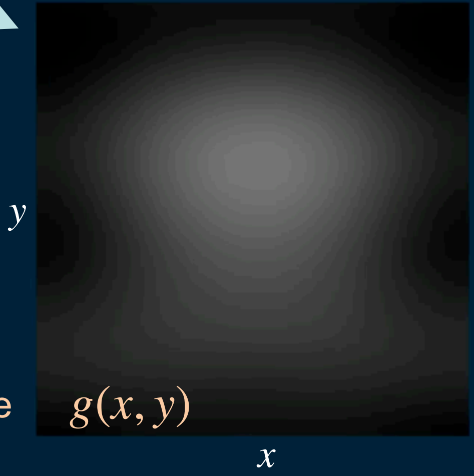
Amplitude: weight of the grating

Phase (hue): shift from the center

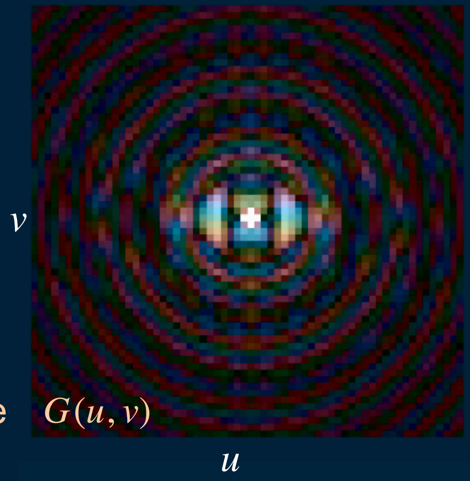
IFT



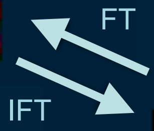
Real space



Each point in the Fourier plane represents a grating “frequency”

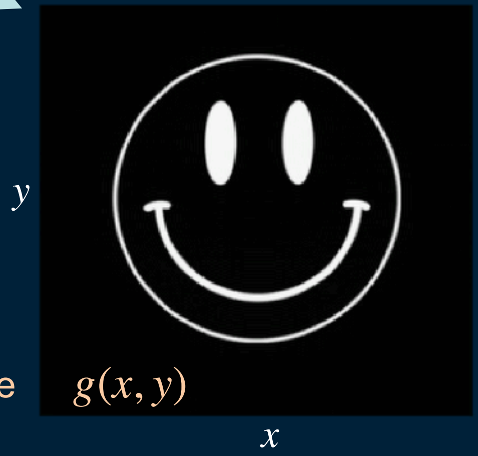


Fourier space $G(u,v)$



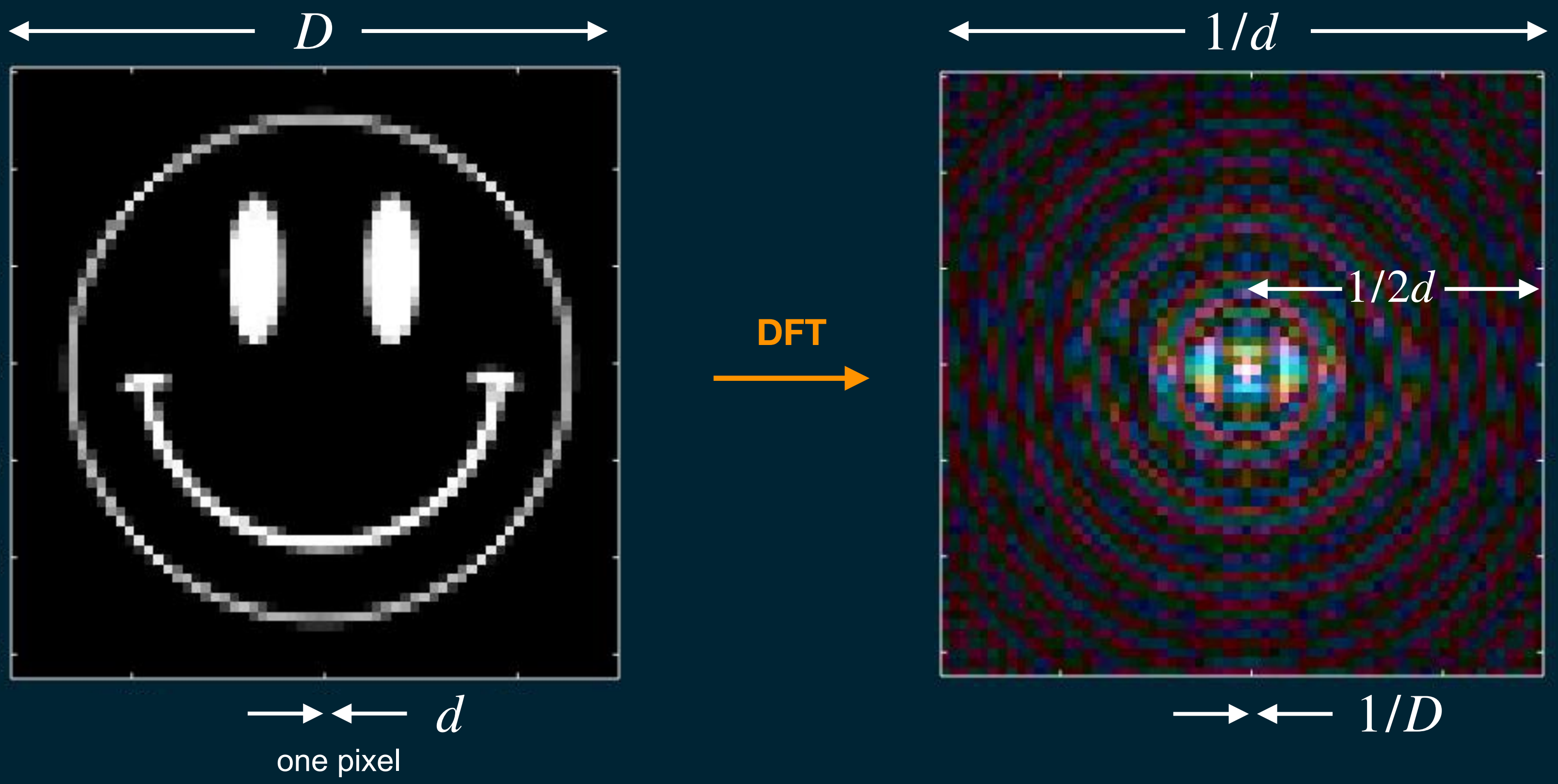
Amplitude: weight of the grating

Phase (hue): shift from the center



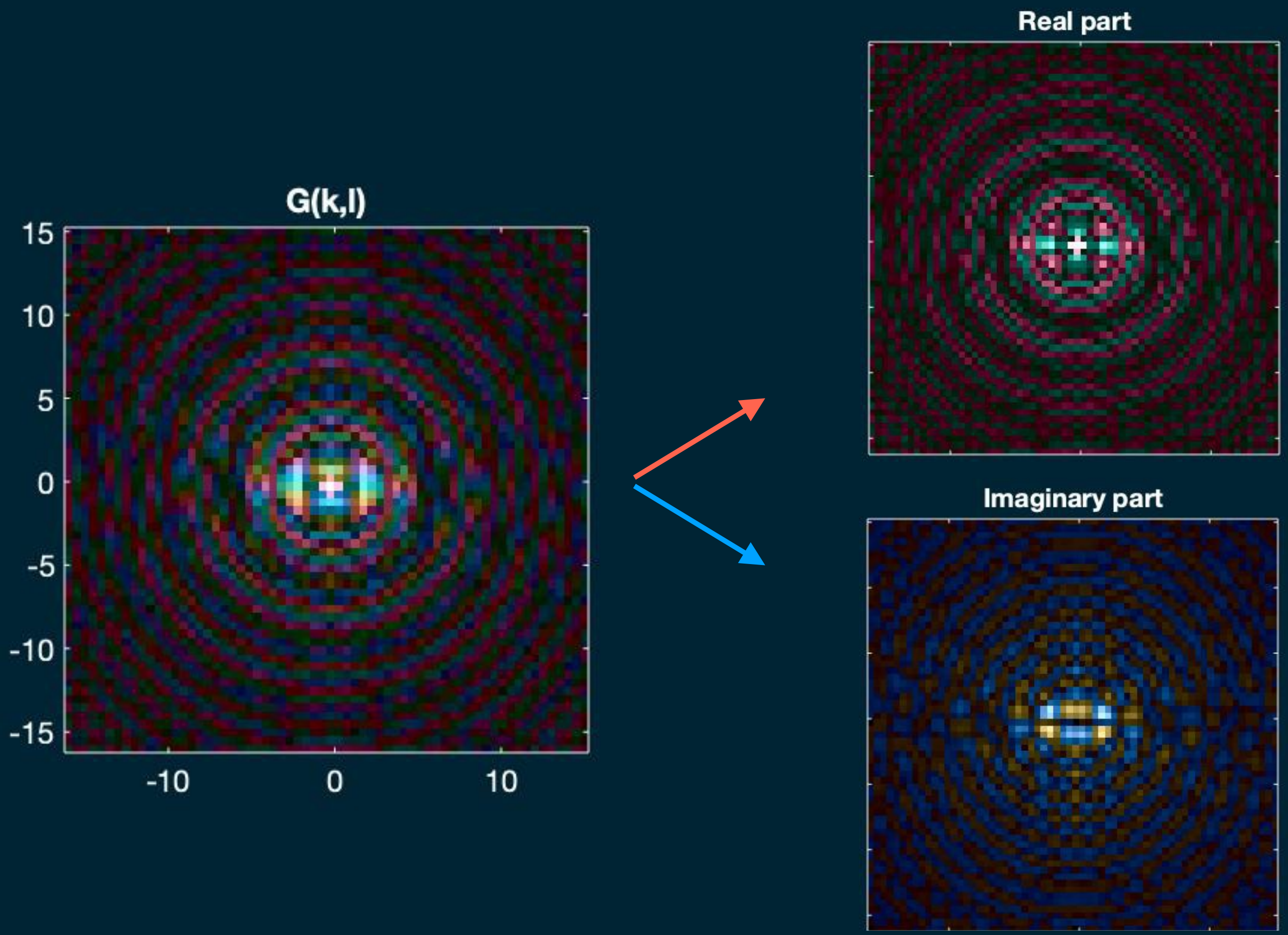
Real space $g(x,y)$

Scaling of an image and its discrete FT



$1/2d$ is the *Nyquist frequency*

The Fourier representation of an image has the same information content



2D Fourier transform properties

$$ab\, g(ax, by) \rightarrow G(u/a, v/b)$$ Scale

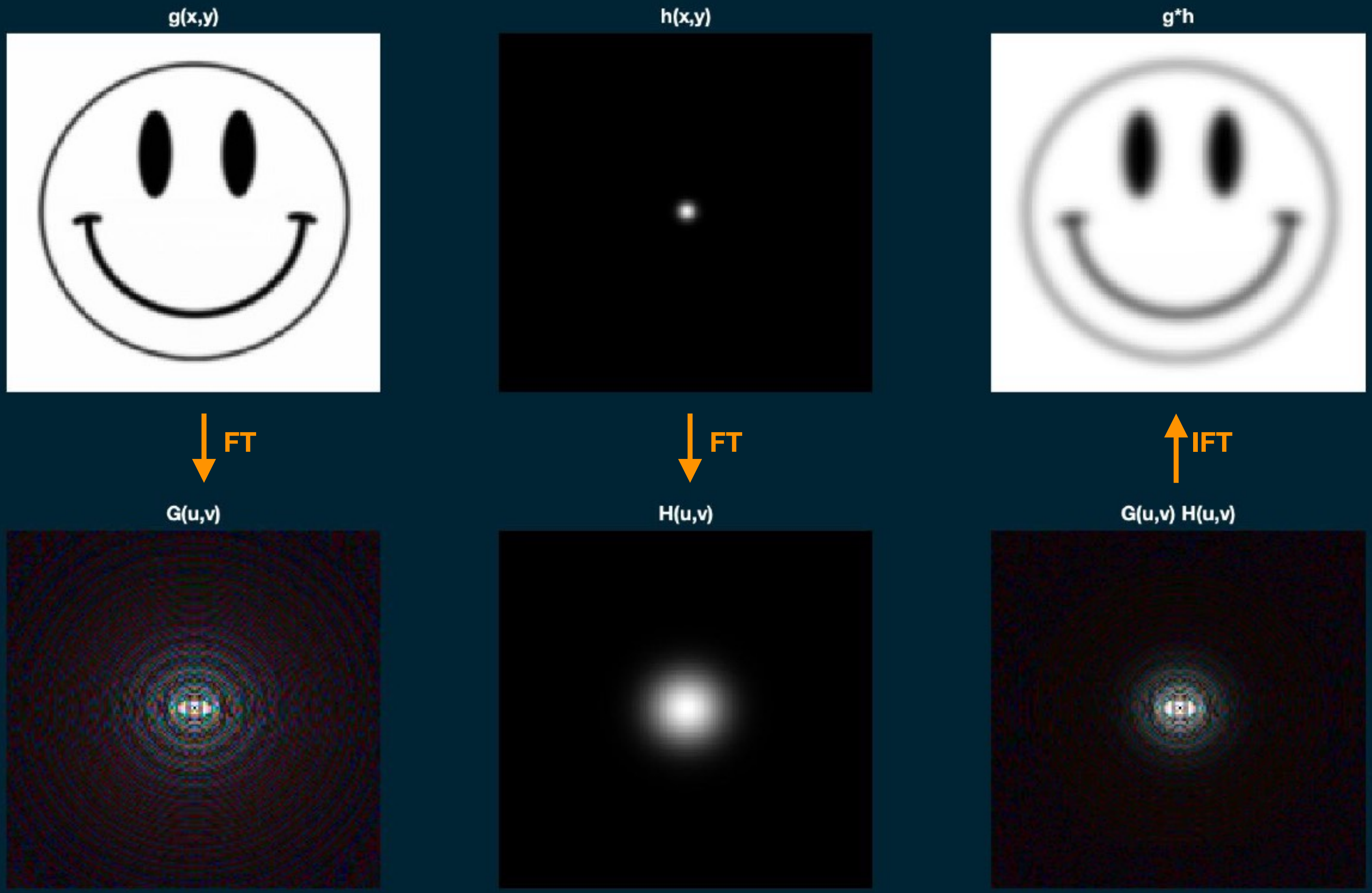
$$g(x-a, y-b) \rightarrow G(u,v)e^{-i2\pi(au+bv)}$$ Shift

$$g * h \rightarrow GH$$ Convolution

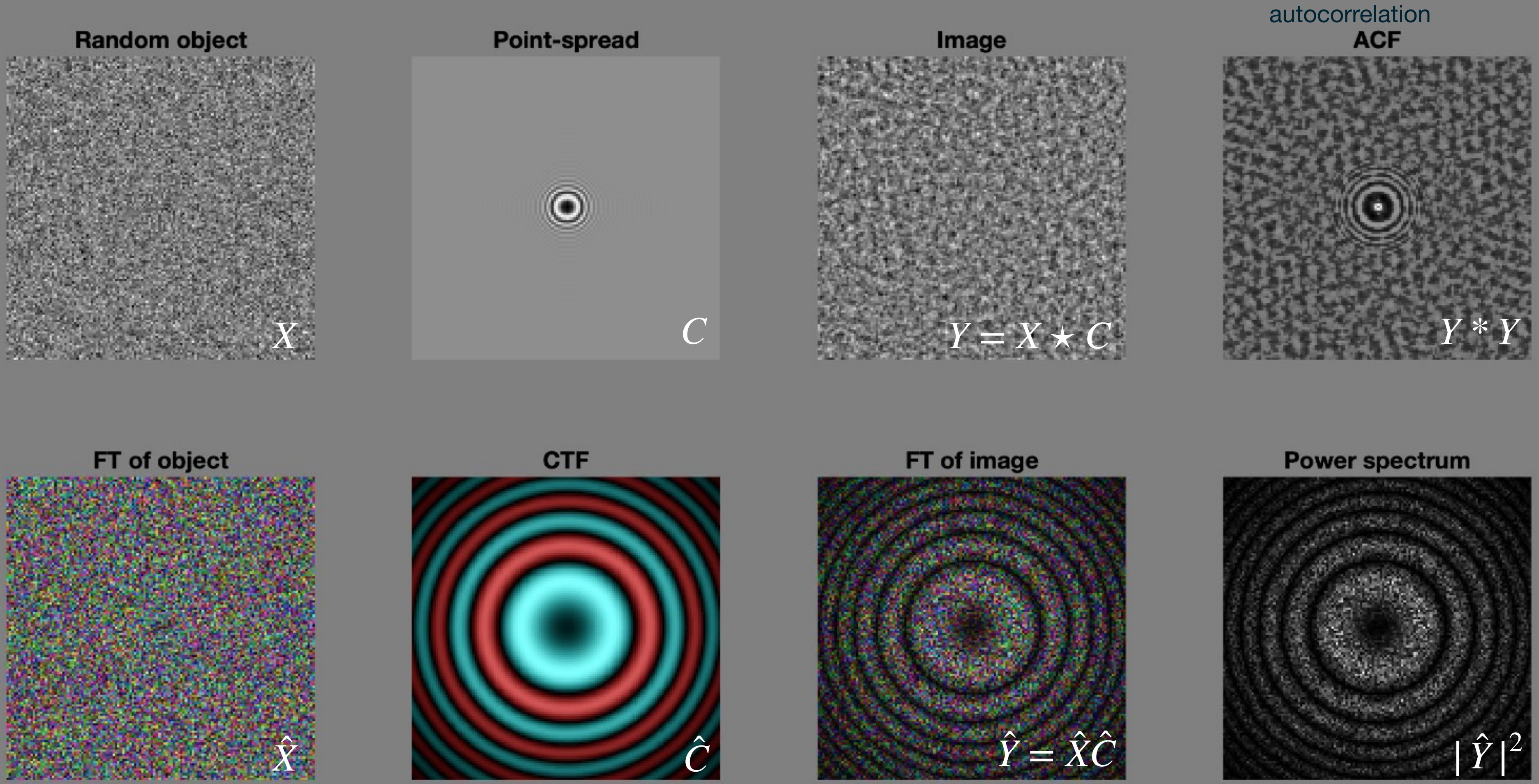
$$g(x', y') \rightarrow G(u', v')$$ Rotation

$$P_y\, g(x, y) \rightarrow G(u, 0)$$ Projection

Convolution with a Gaussian

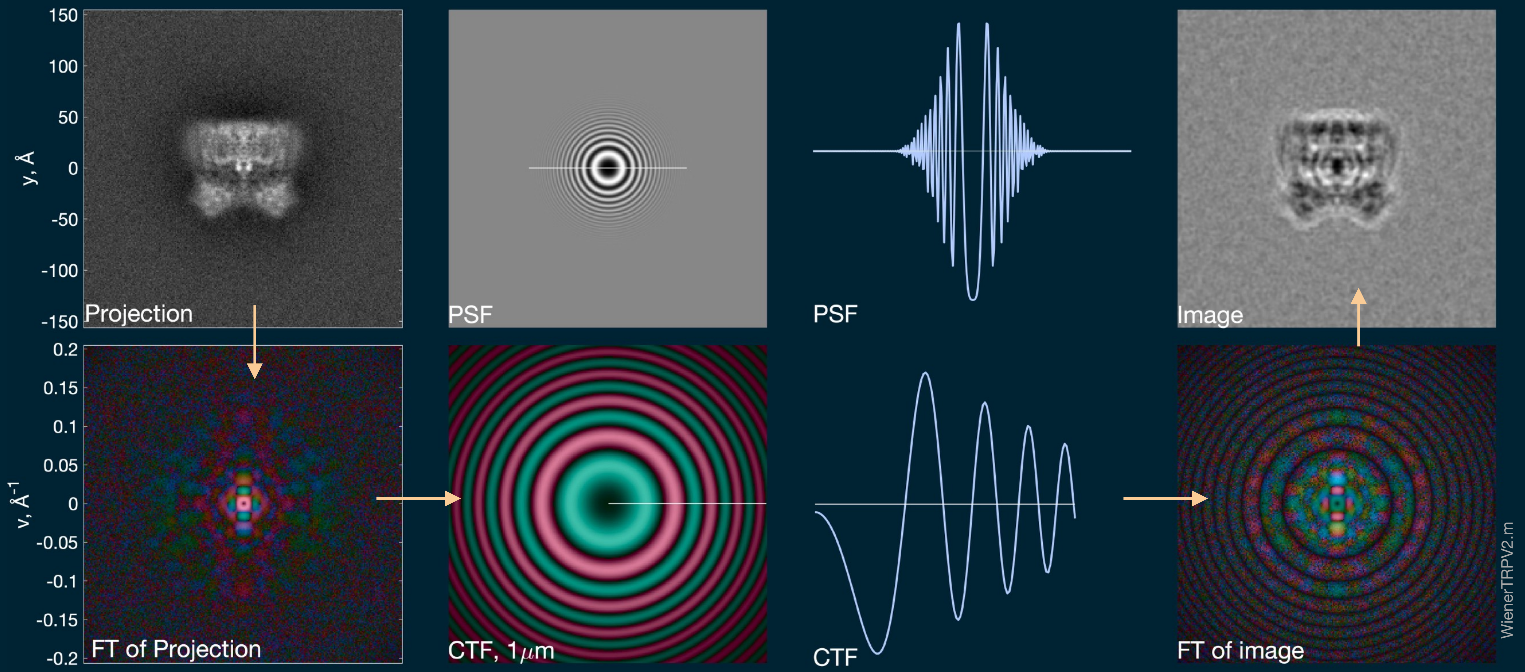


Visualizing the contrast transfer function



Modeling the CTF effect on an image

Model of an image
$X = CA + N$

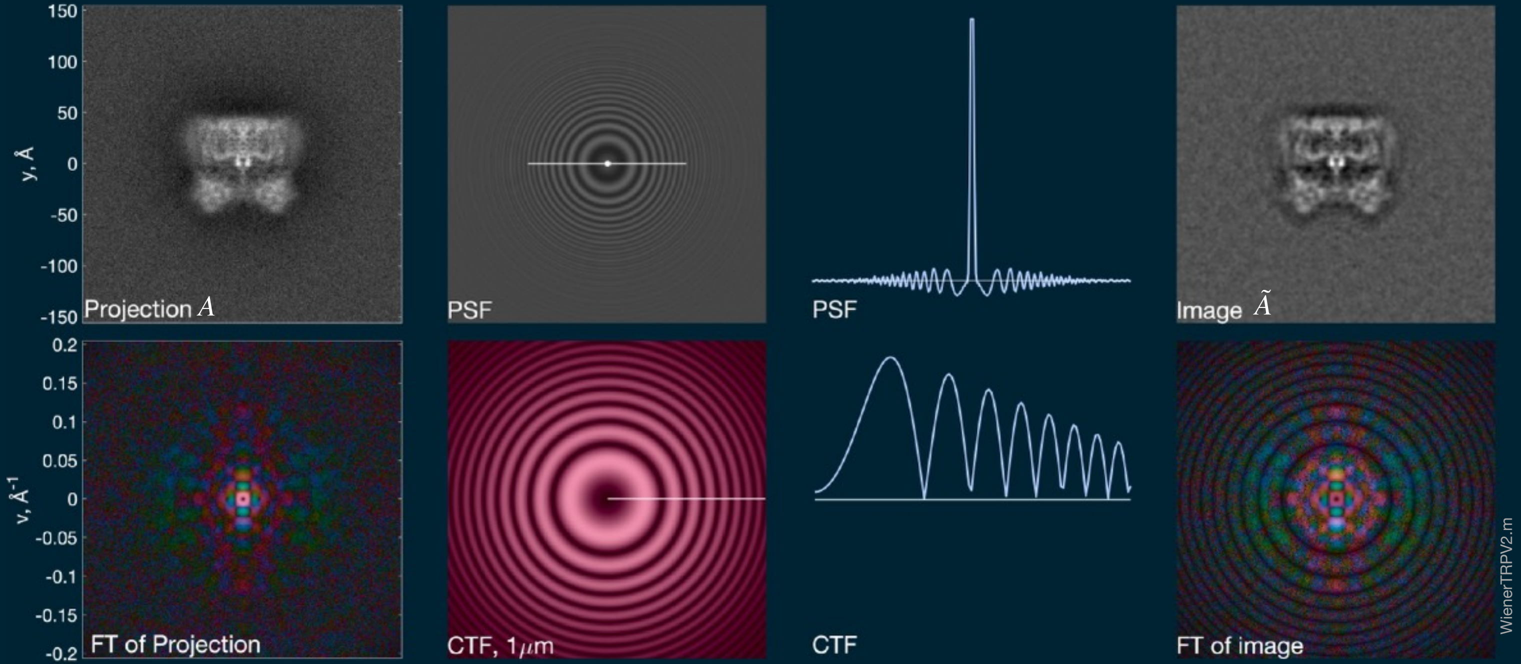


Can we do the deconvolution
$\tilde{A} = X/C$??

How to undo the CTF effects?

1. **Phase flipping**

$$\tilde{A} = \mathrm{sgn}(C)X$$



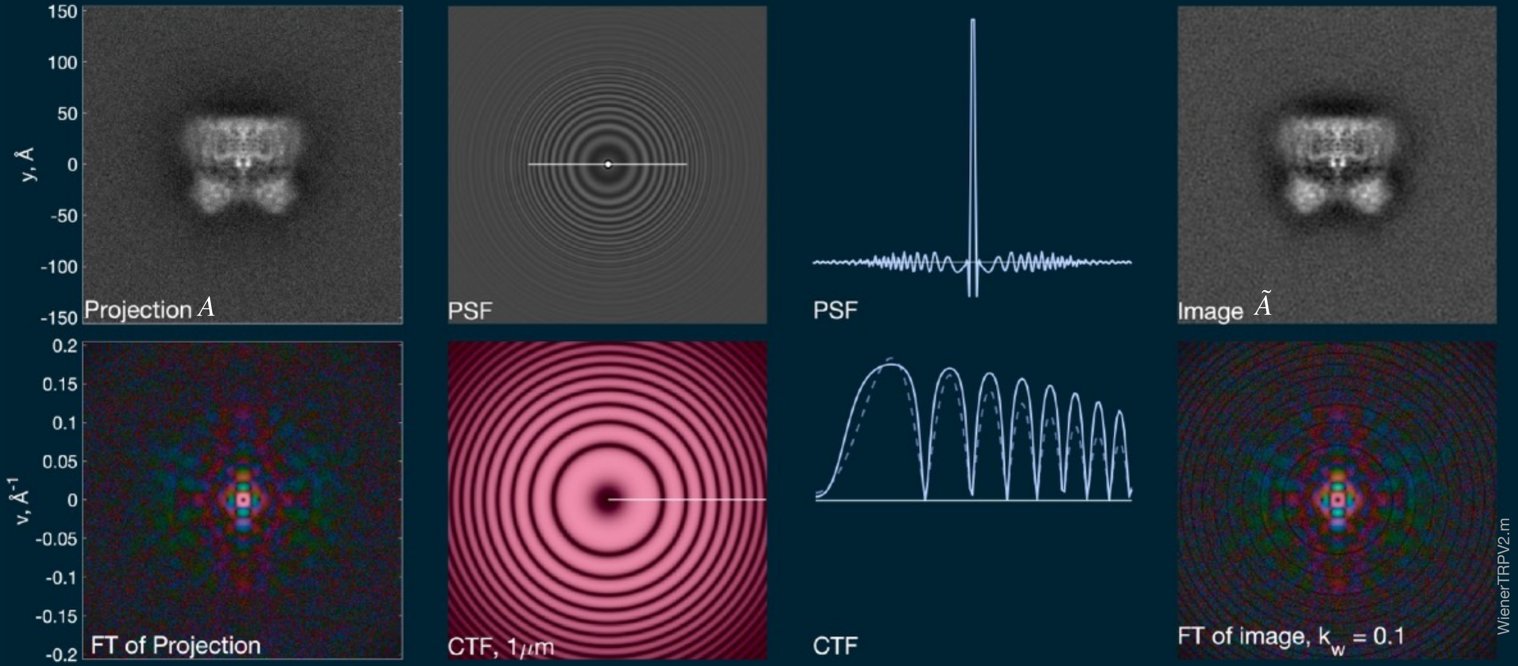
How to undo the CTF effects?

1. **Phase flipping**

$$\tilde{A} = \text{sgn}(C)X$$

2. **Wiener filter**

$$\tilde{A} = \frac{CX}{C^2 + k}$$



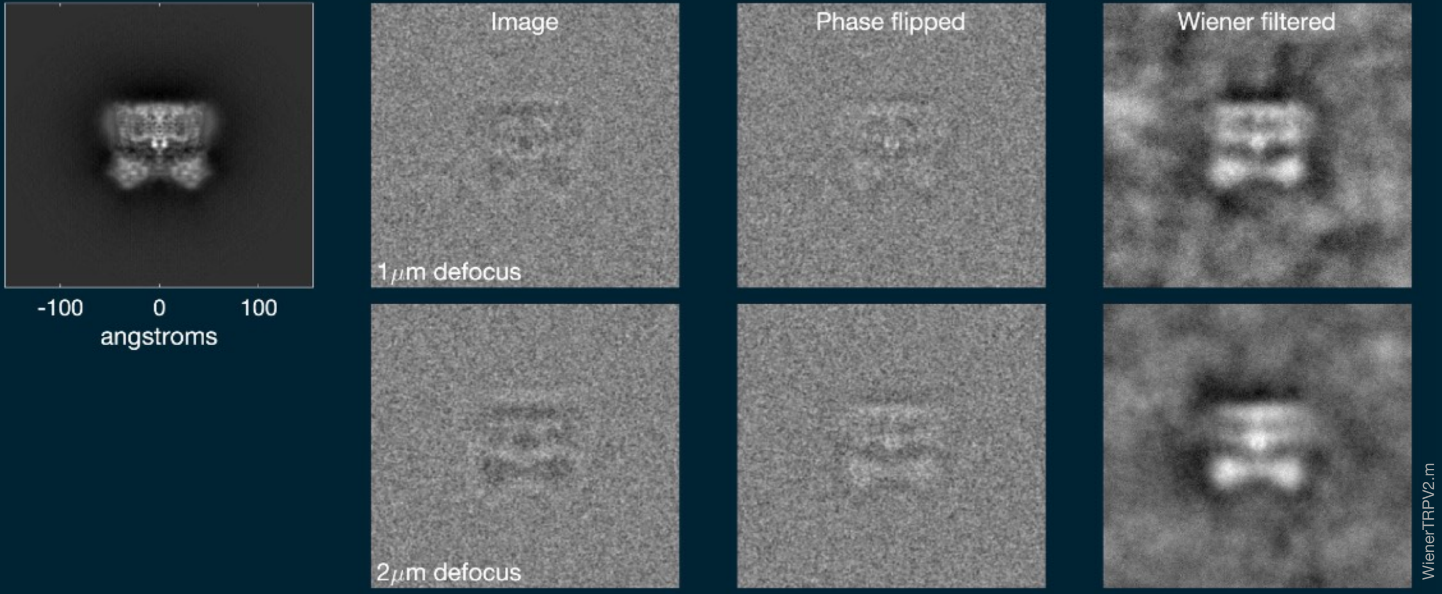
How to undo the CTF effects in noisy images?

1. Phase flipping

$$\tilde{A} = \mathrm{sgn}(C)X$$

2. Wiener filter

$$\tilde{A} = \frac{CX}{C^2 + k}$$



Modeling the CTF effect on an image

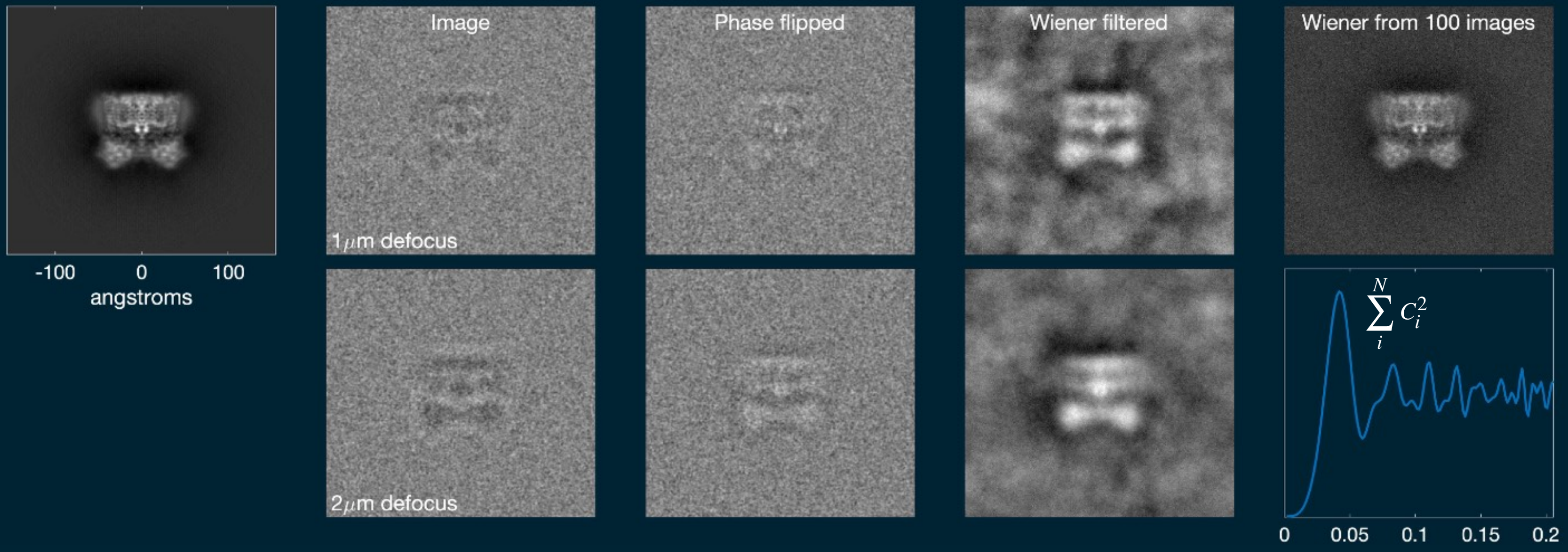
Model of an image

$$X = CA + N$$

A	"true" image
C	contrast-transfer function
N	noise image

We can interpret *C* as either the CTF operator (*x*,*y* space), or just the multiplicative CTF factor (*u*,*v* space)

How to undo the CTF effects in noisy images?



3. Wiener from multiple images

$$\tilde{A} = \frac{\sum_i^N C_i X_i}{k + \sum_i^N C_i^2}$$

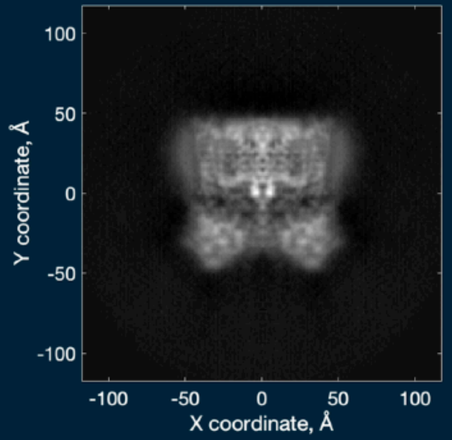
$$k(s) = 1/\mathrm{SNR} = \frac{||N||^2}{||A||^2}$$

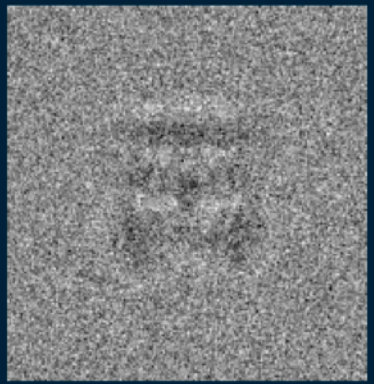
Image restoration when spectral SNR is known

Restoration from multiple images

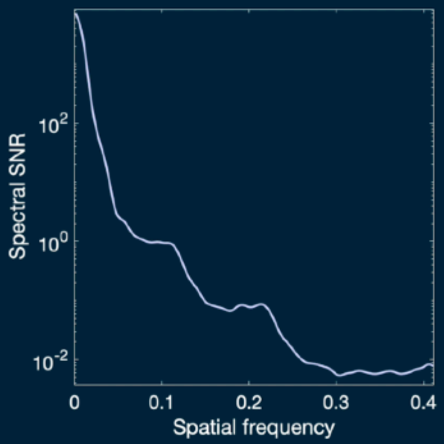
$$\tilde{A} = \frac{\sum_i^N C_i X_i}{k_w(s) + \sum_i^N C_i^2}$$

The defocus varies to fill in CTF zeros





N= 1 images



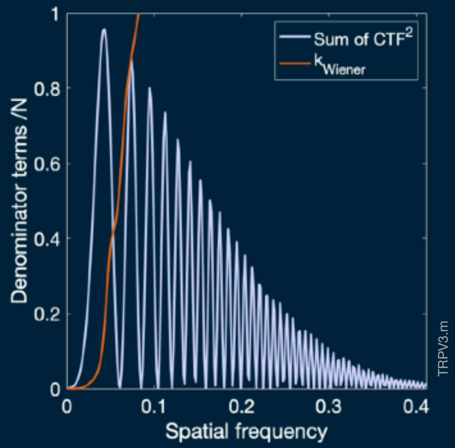
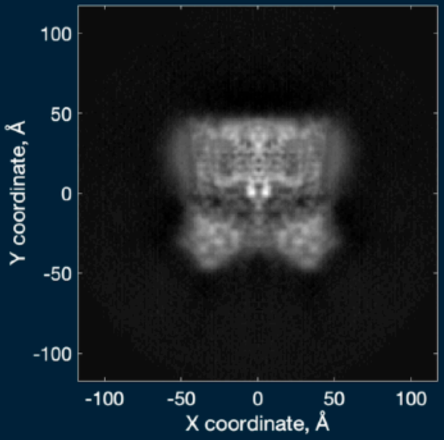


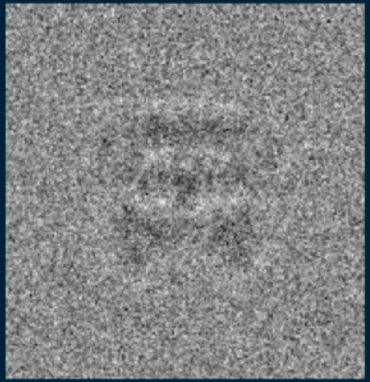
Image restoration when spectral SNR is known

Restoration from multiple images

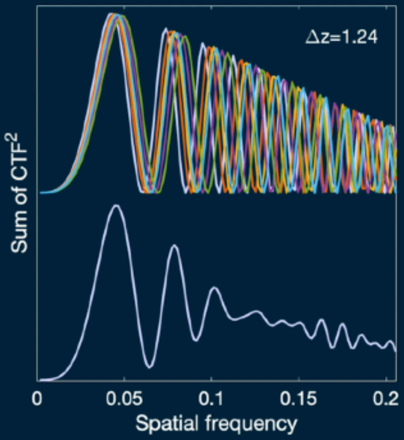
$$\tilde{A} = \frac{\sum_i^N C_i X_i}{k_w(s) + \sum_i^N C_i^2}$$

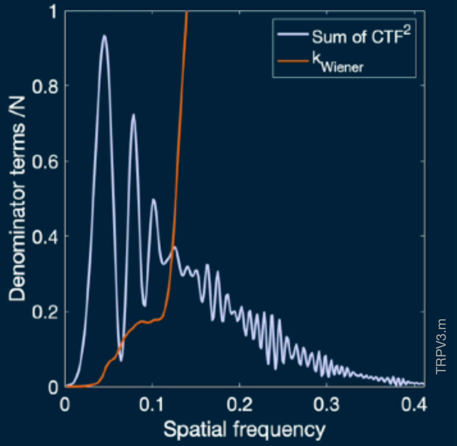
The defocus varies to fill in CTF zeros





N= 6 images





Even the small defocus range 1–1.5 μm is sufficient.

Defocus contrast and the CTF

Correlation and particle picking

Single-particle reconstruction

Maximum-likelihood methods

2D Fourier transform properties

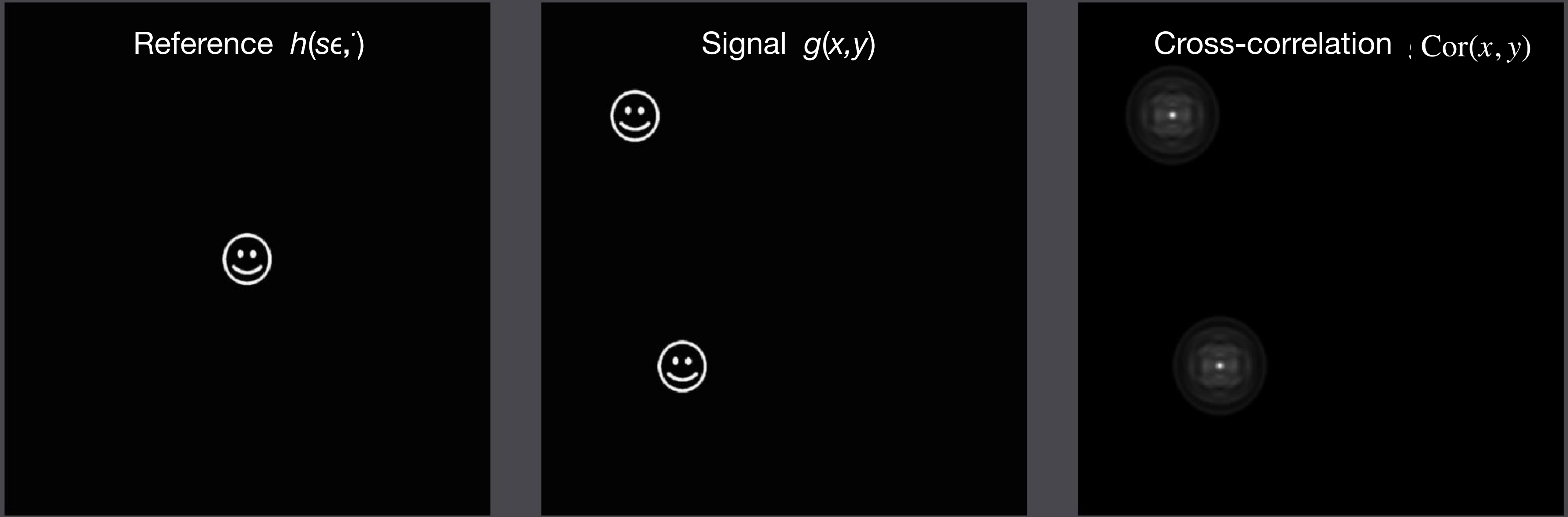
$ab\, g(ax, by) \rightarrow G(u/a, v/b)$	Scale
$g(x-a, y-b) \rightarrow G(u,v)e^{-i2\pi(au+bv)}$	Shift
$g * h \rightarrow GH$	Convolution
$g(x', y') \rightarrow G(u', v')$	Rotation
$P_y\, g(x,y) \rightarrow G(u,0)$	Projection

Correlation locates motifs in images

Translational cross-correlation function

$$\mathrm{Cor}(x,y) = X \otimes R$$
$$= \sum_{s,t} h(s,t)\, g(x+s, y+t)$$

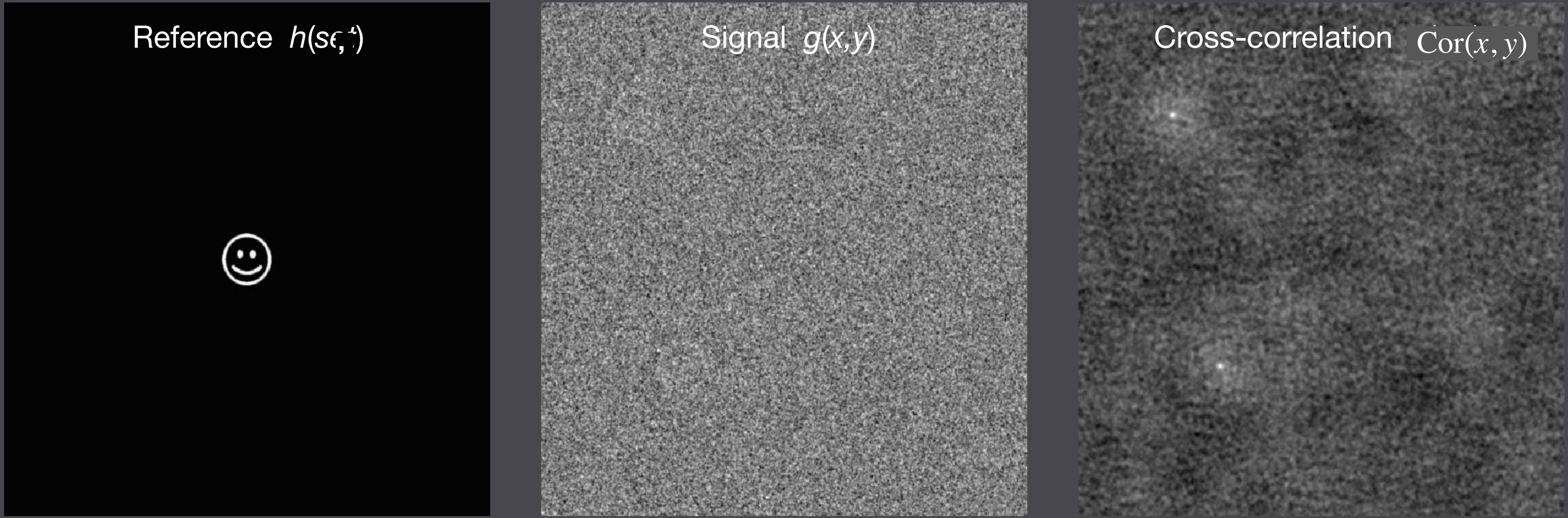
Correlation is like convolution.
The FT pair is: $g \otimes h \rightarrow GH^*$



Correlation locates motifs in images

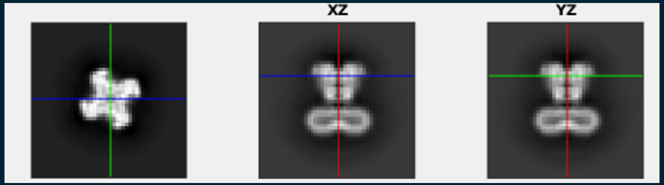
Translational cross-correlation function

$$\mathrm{Cor}(x,y) = X \otimes R$$
$$= \sum_{s,t} h(s,t)\, g(x+s, y+t)$$

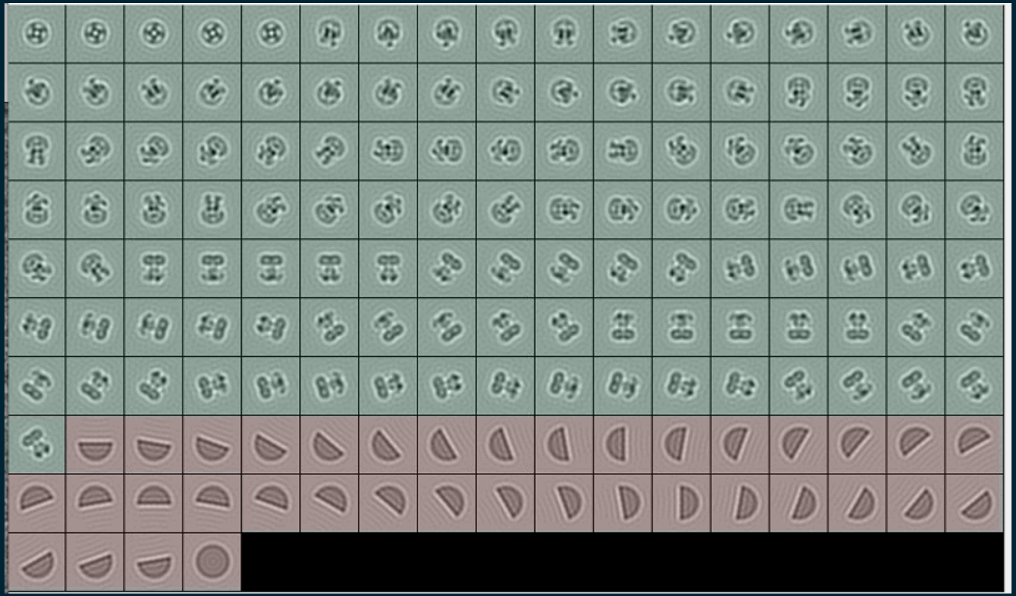


A correlation-based particle picker

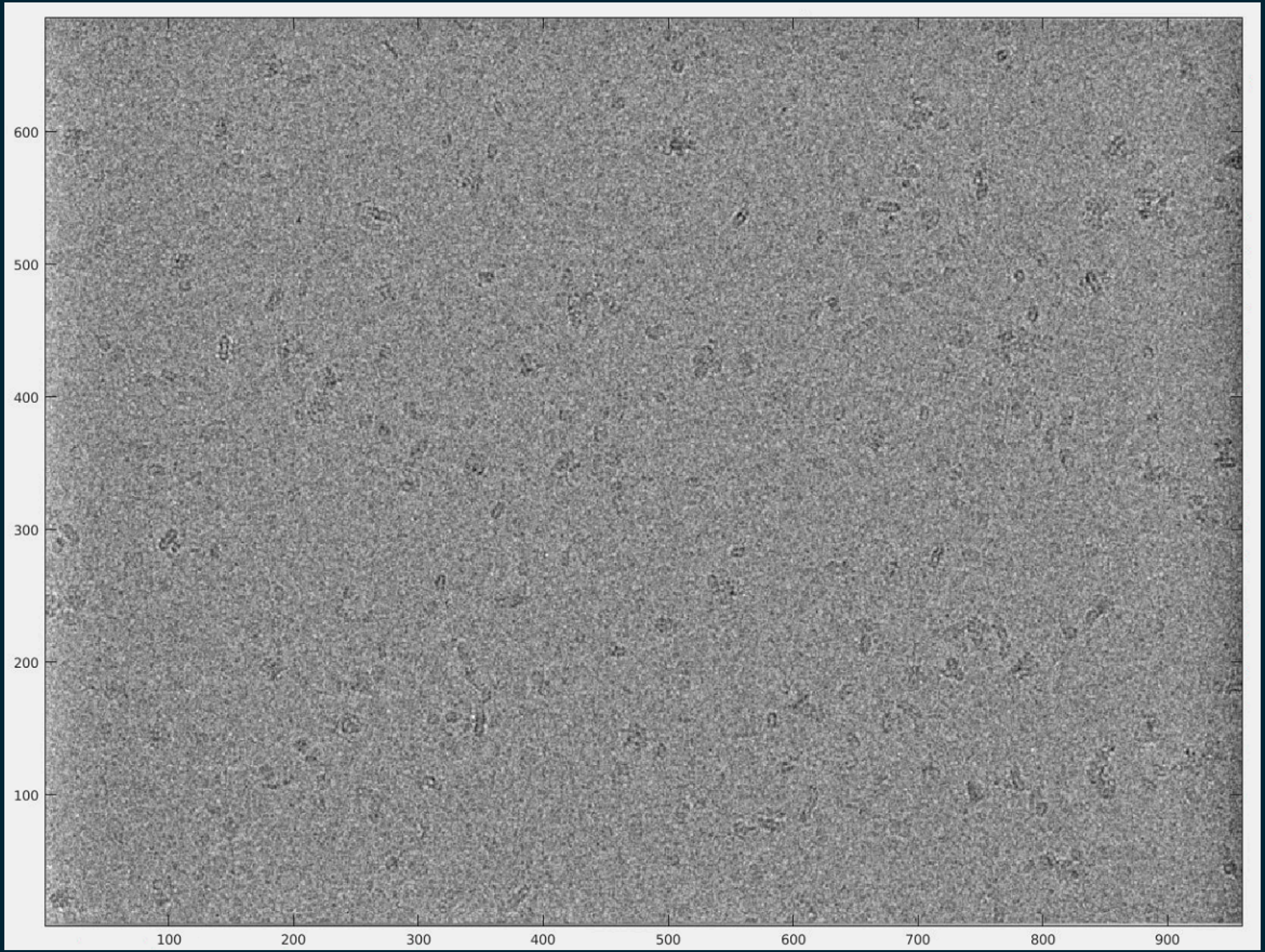
3D Reference



CTF-filtered projections and decoys

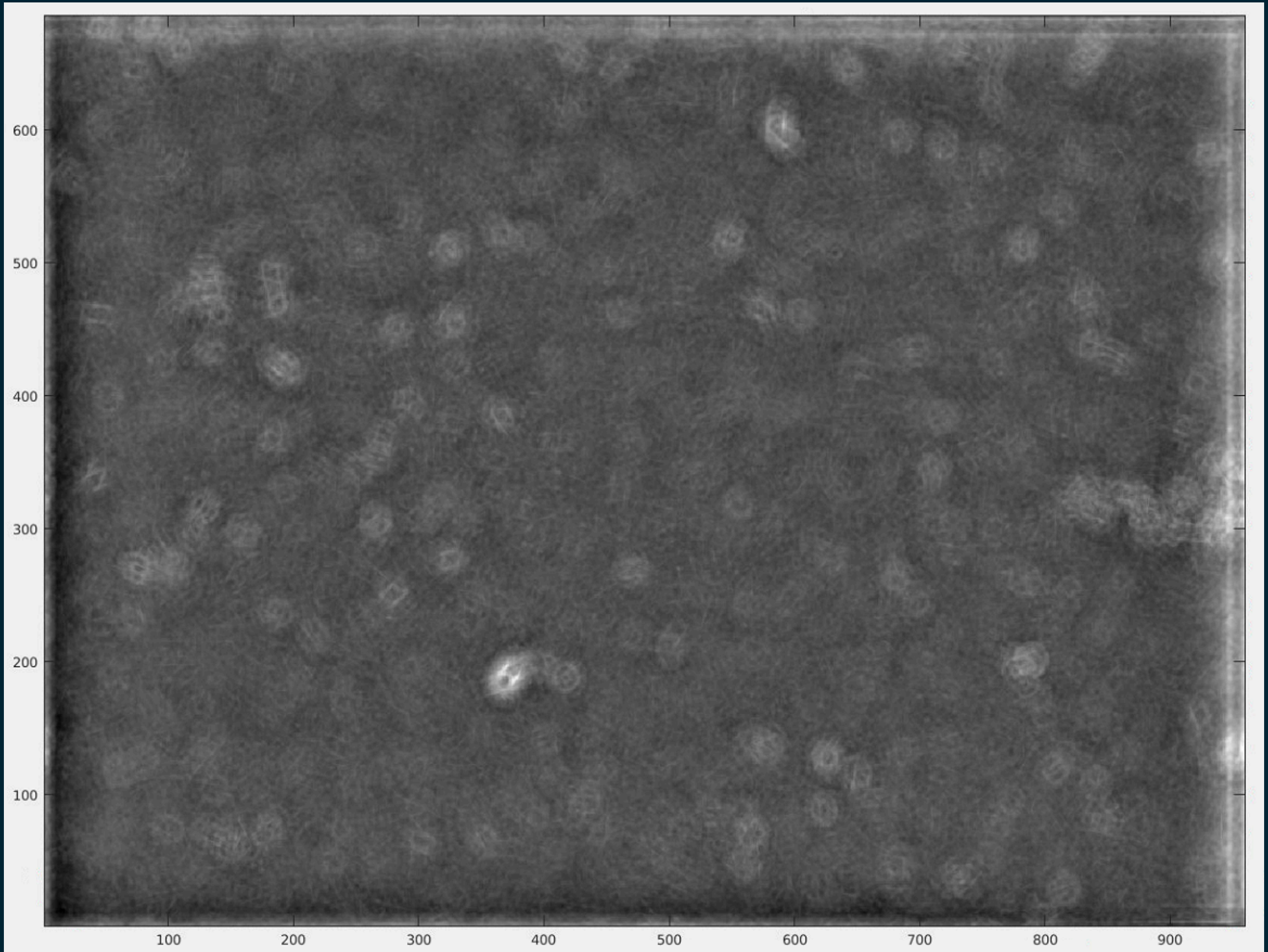


A correlation-based particle picker



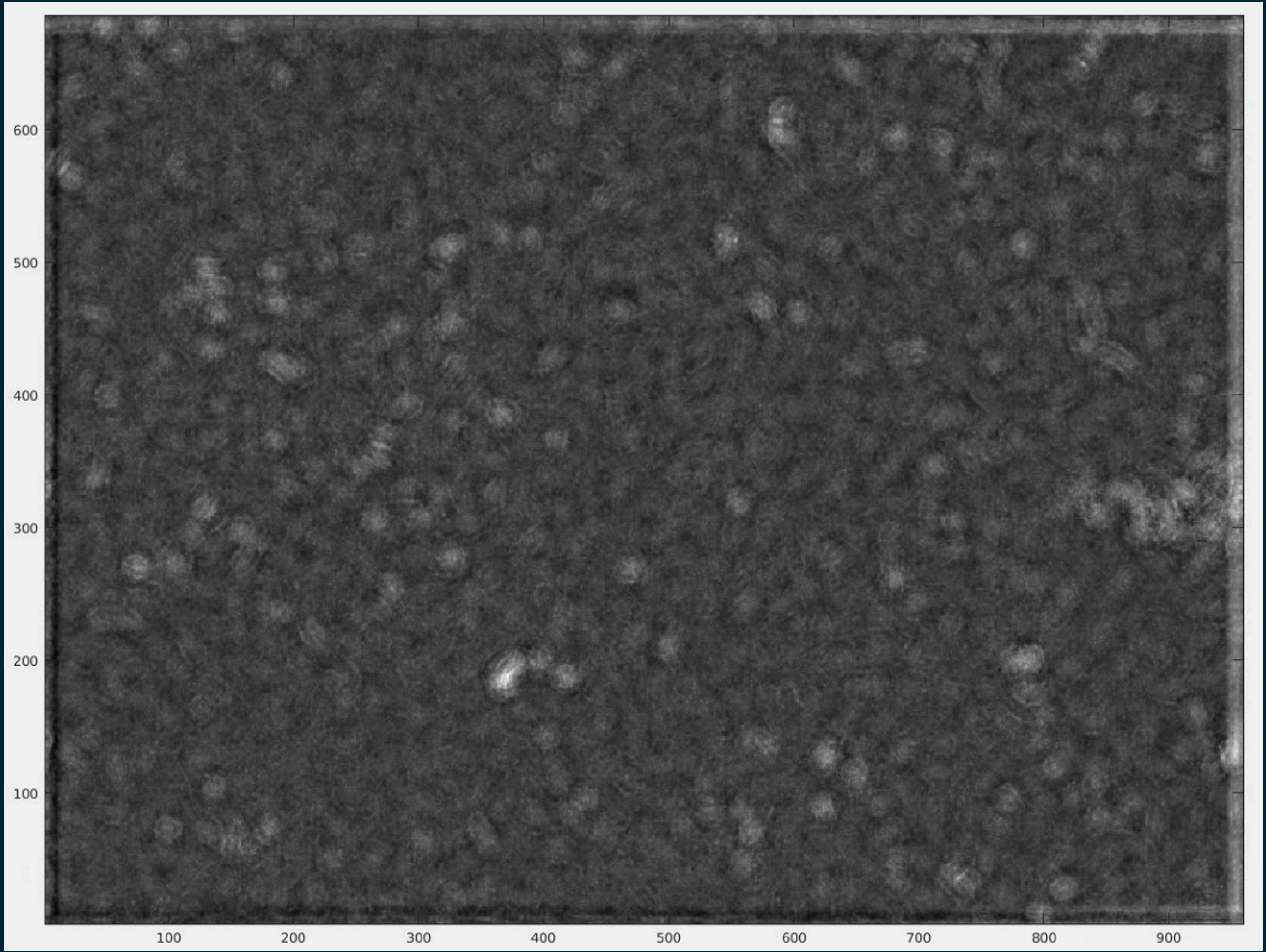
A micrograph

A correlation-based particle picker



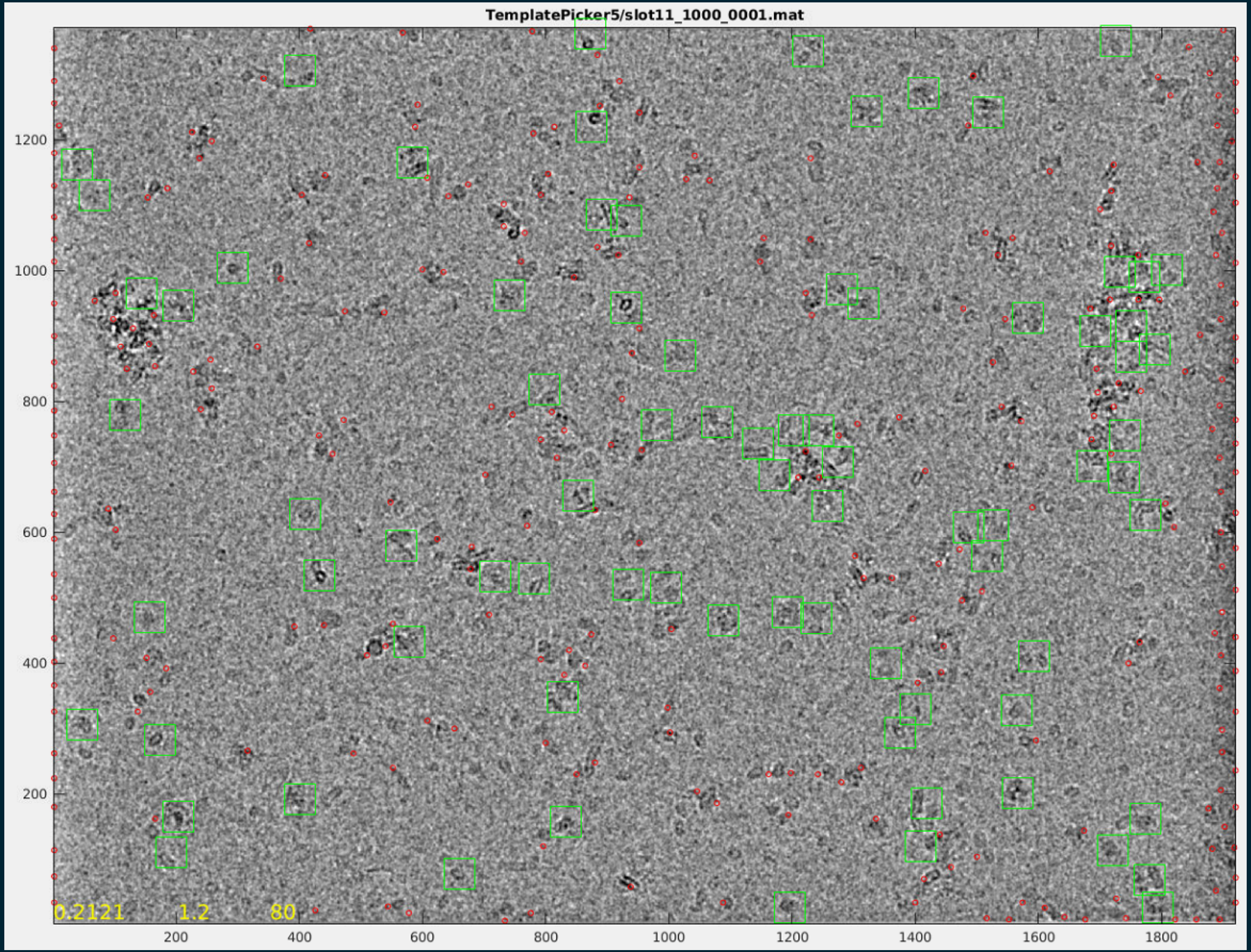
Max of correlations with decoy references

A correlation-based particle picker



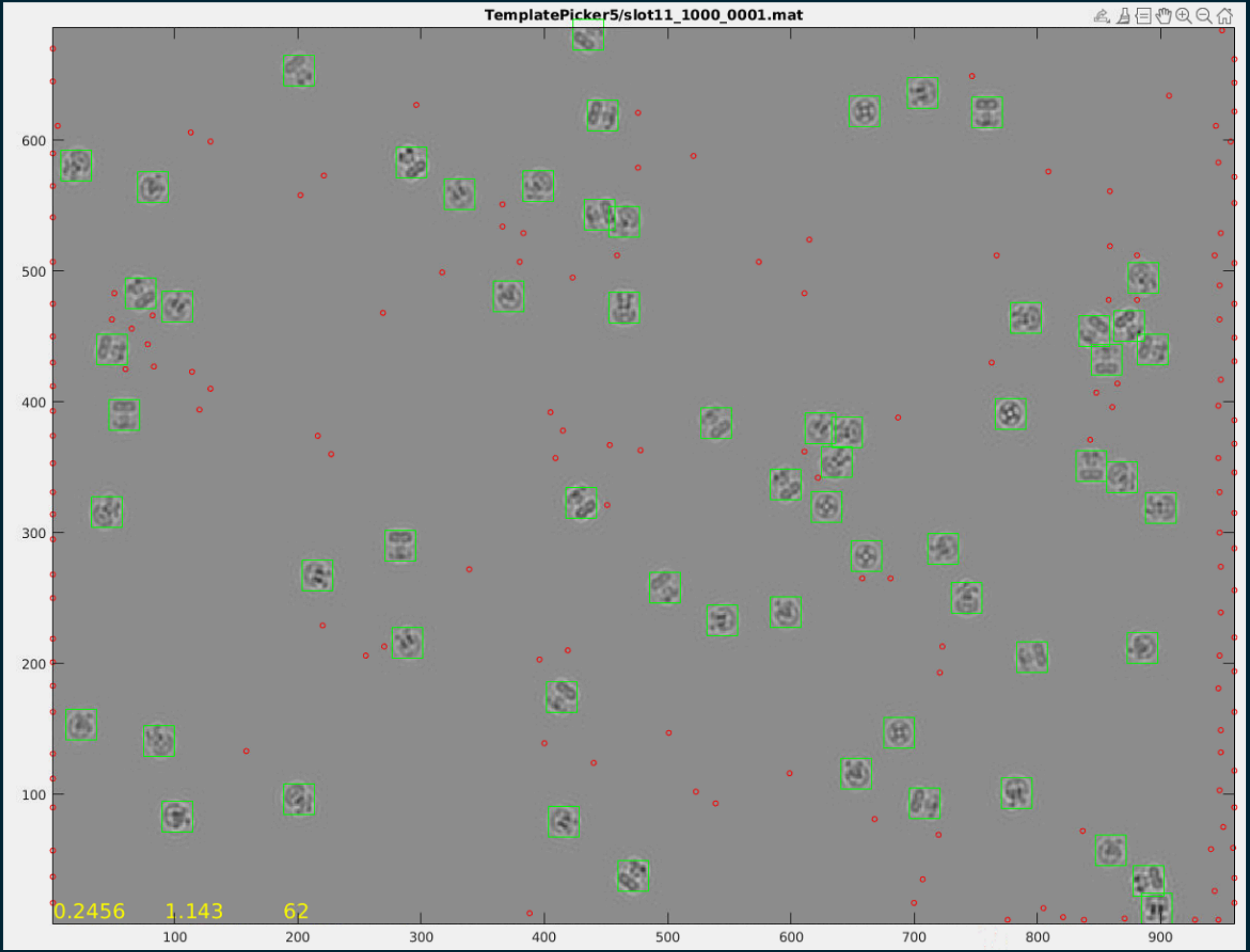
Max of correlations with particle references

A correlation-based particle picker



Green: found particles
Red dots: decoys

A correlation-based particle picker



Best-matching references

Defocus contrast and the CTF

Correlation and particle picking

Single-particle reconstruction

Maximum-likelihood methods

2D Fourier transform properties

$$ab\, g(ax, by) \rightarrow G(u/a, v/b)$$ Scale

$$g(x-a, y-b) \rightarrow G(u,v)e^{-i2\pi(au+bv)}$$ Shift

$$g * h \rightarrow GH$$ Convolution

$$g(x', y') \rightarrow G(u', v')$$ Rotation

$$P_y\, g(x,y) \rightarrow G(u,0)$$ Projection

The rotation property

2D Fourier Transform

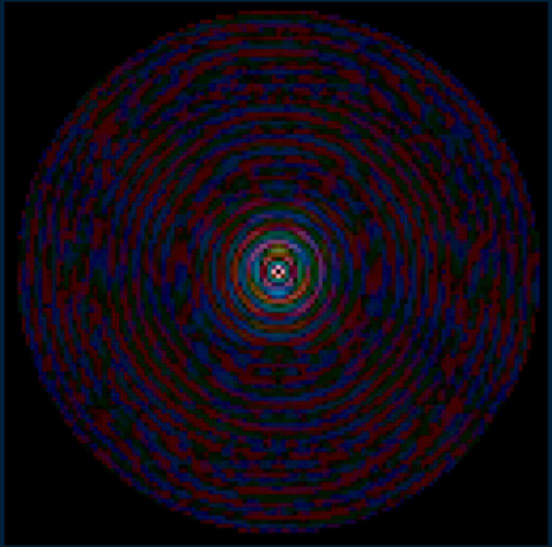
$$G(u, v) = \iint g(x, y) e^{-i2\pi(ux+vy)} dxdy$$

FT using 2D vectors

$$G(\mathbf{u}) = \int\int g(\mathbf{x}) e^{-i2\pi(\mathbf{u}\cdot\mathbf{x})} d^2\mathbf{x}$$

The dot-product is invariant under rotations!

FT →



Let R_θ signify a rotation, and

$$(x', y') = R_\theta (x, y)$$

$$(u', v') = R_\theta (u, v)$$

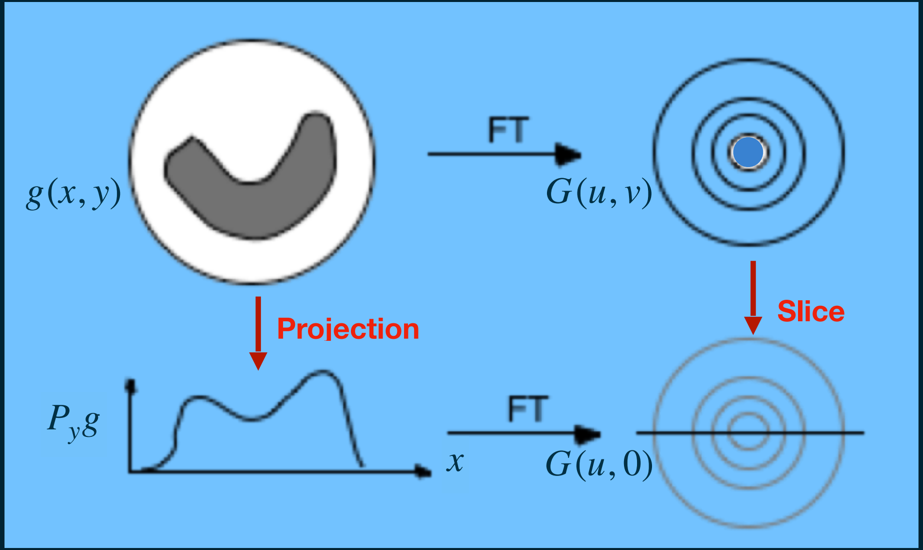
then

$$g(x', y') \to G(u', v')$$

or alternatively,

$$g(R_\theta \mathbf{x}) \to G(R_\theta \mathbf{u})$$

The Projection Theorem



2D Fourier Transform

$$G(u, v) = \iint g(x, y) e^{-i2\pi(ux+vy)} dx dy$$

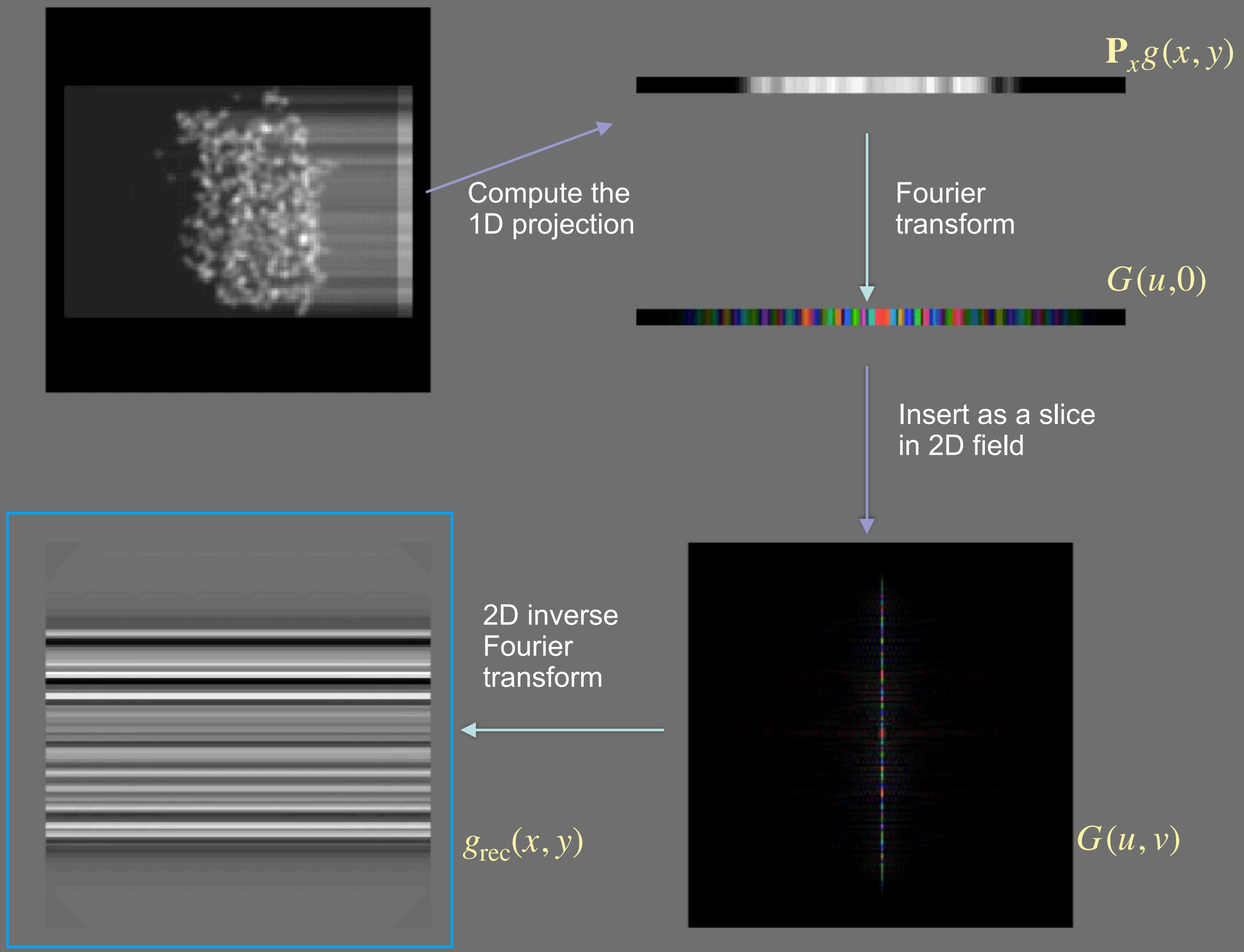
Projection along y

$$P_y g(x, y) = \int g(x, y) dy$$

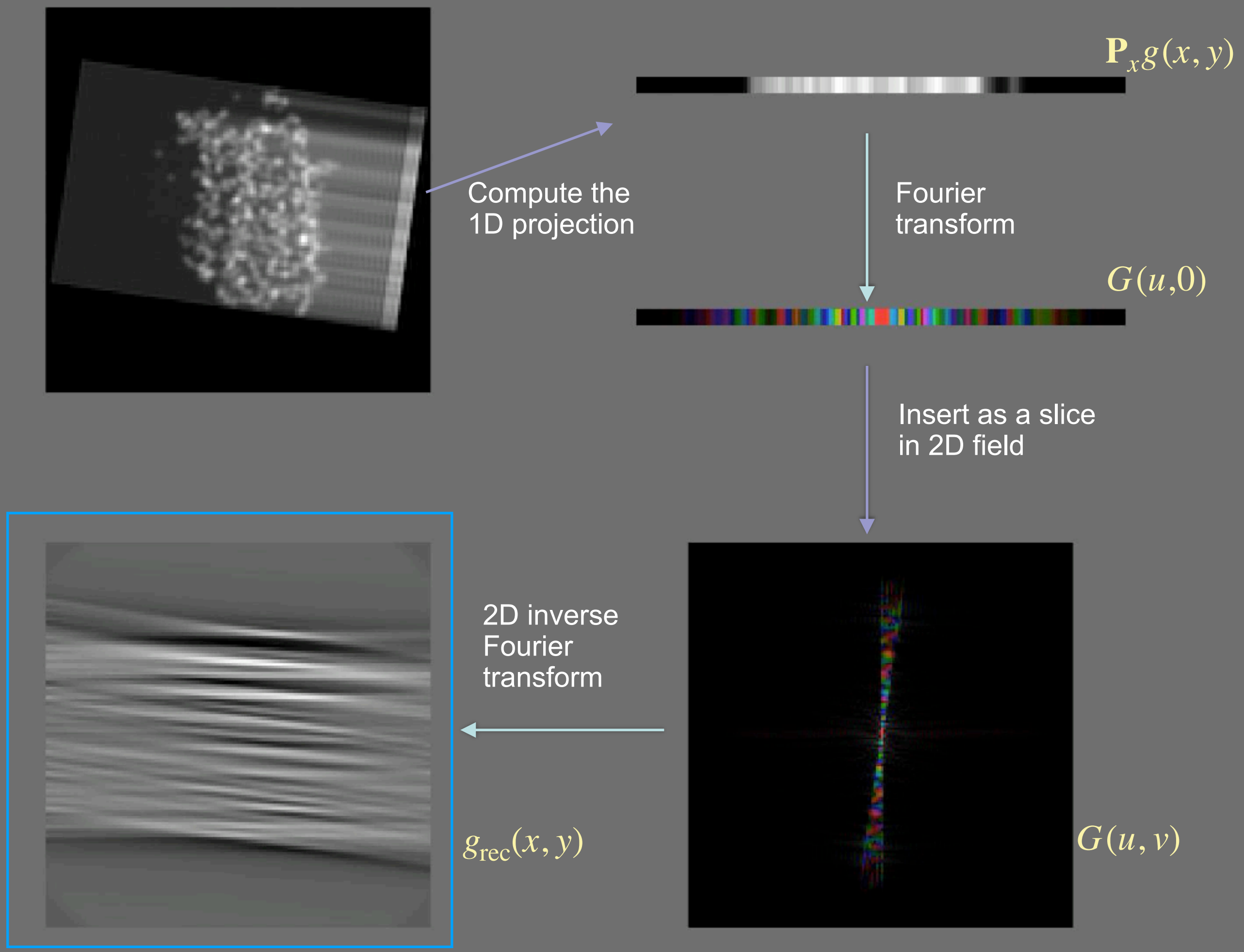
Values along the u axis

$$G(u, 0) = \int \left(\int g(x, y) dy \right) e^{-i2\pi(ux)} dx$$
$$= \mathcal{F}\{P_y g\}$$

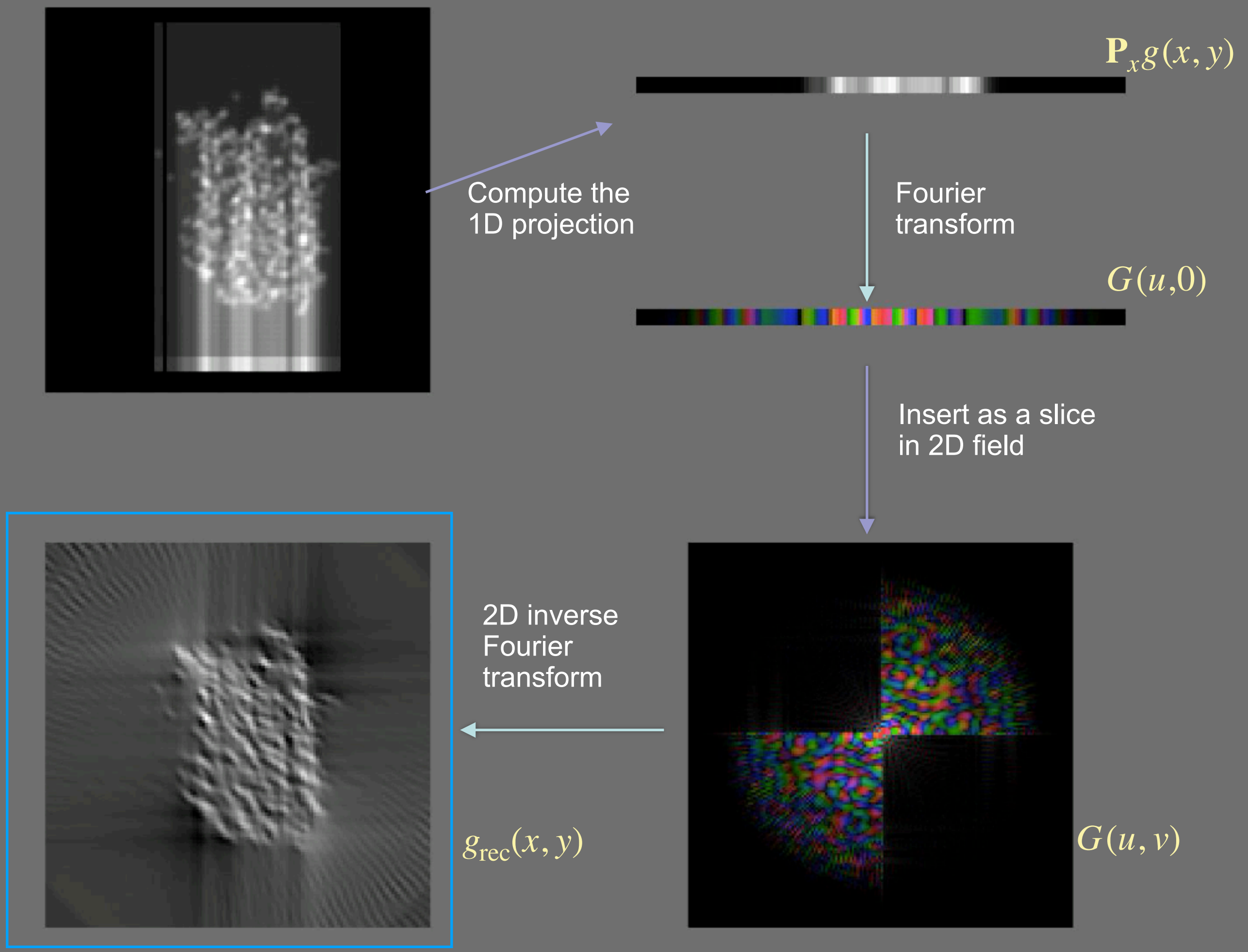
Reconstruction using the Fourier Slice Theorem



Reconstruction using the Fourier Slice Theorem



Reconstruction using the Fourier Slice Theorem

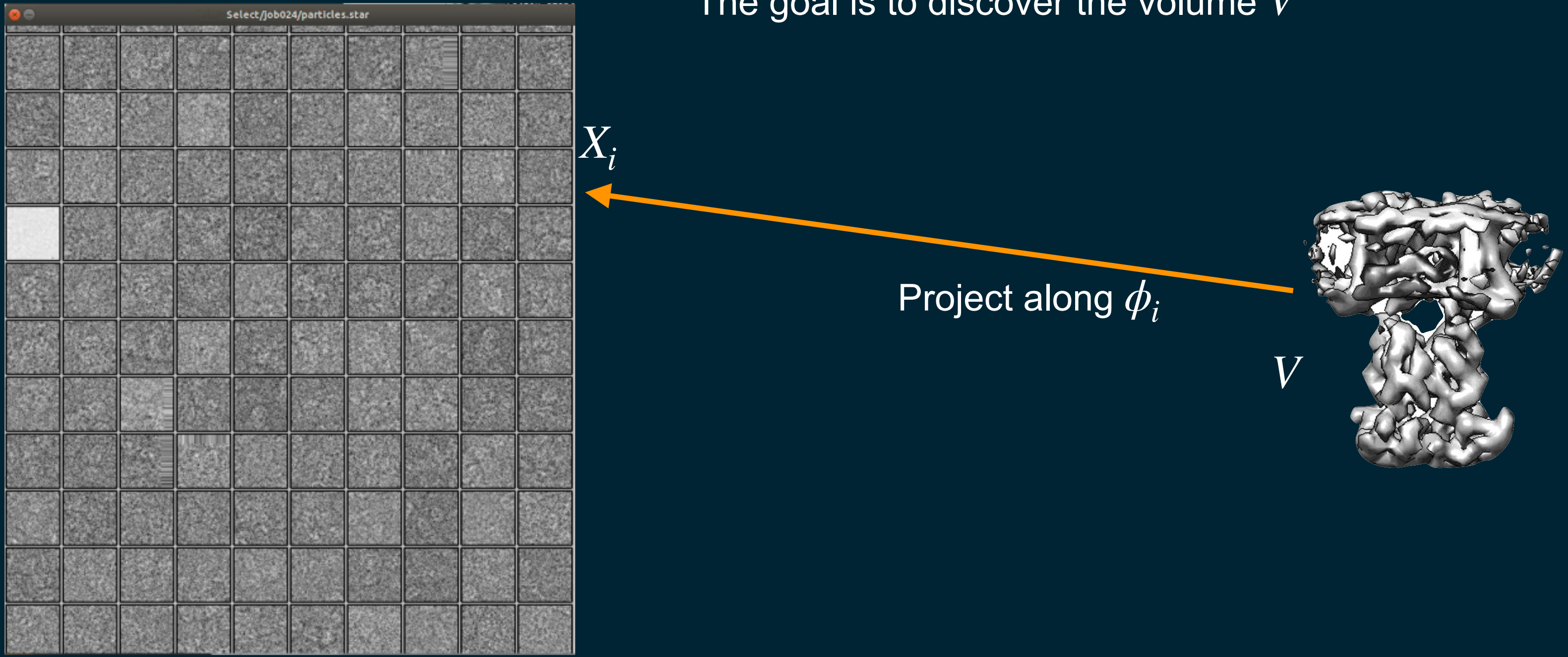


Single-particle reconstruction

We assume that image X_i comes from a projection in direction ϕ_i of volume V according to

$$X_i = C_i \mathbf{P}_{\phi_i} V + N_i$$

The goal is to discover the volume V



There are various ways to compare images

Define the "reference" as the true image A modified by the CTF C:

$$R = CA$$

We wish to compare a data image X with it.

Squared difference

$$\|X - R\|^2 = \sum_j (X_j - R_j)^2$$
$$= \|X\|^2 - 2X \cdot R + \|R\|^2$$

Correlation

$$\mathrm{Cor} = X \cdot R$$
$$= \sum_j X_j R_j$$

Correlation coefficient

$$\mathrm{CC} = \frac{X \cdot R}{|X||R|}$$

Notation used here:

A single pixel in the image X:

X_j —the j^{th} pixel (out of J pixels total)

The i^{th} image in the dataset $\mathbf{X}$:

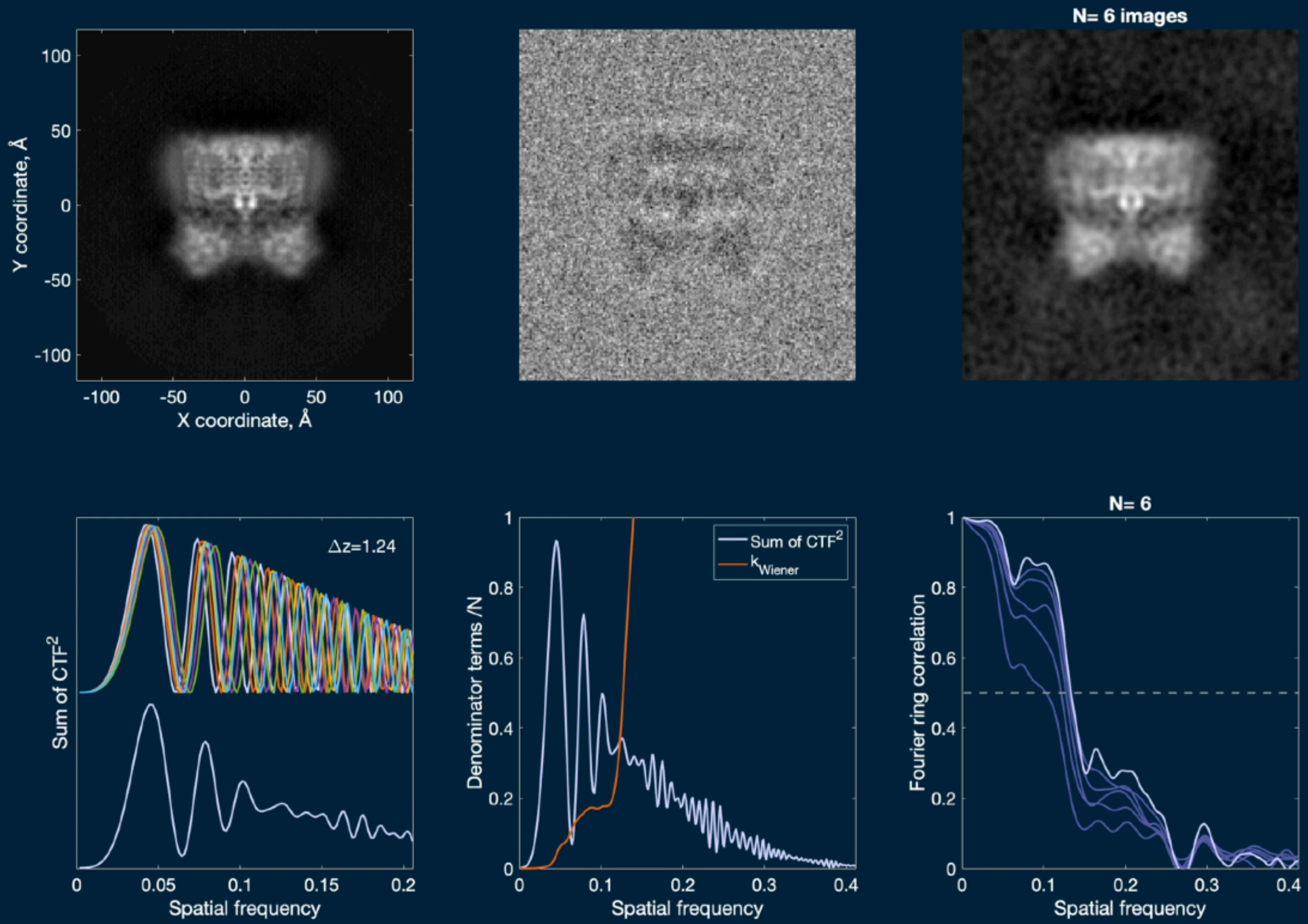
X_i

The Wiener filter applied to images

Restoration from multiple images

$$\tilde{A} = \frac{\sum_i^N C_i X_i}{k_w(s) + \sum_i^N C_i^2}$$

The defocus varies to fill in CTF zeros



FREALIGN combines correlation with Wiener filtering

A Frealign iteration, refining $V^{(n)}$ to $V^{(n+1)}$, consists of two steps:

1. Vary the projection direction ϕ_i to find the projection image $R_i = C_i \mathbf{P}_{\phi_i} V^{(n)}$ that maximizes the correlation coefficient for each image X_i,

$$\mathrm{CC} = \frac{X_i \cdot R_i}{|X_i||R_i|}.$$

2. Knowing the best projection direction ϕ_i for each image X_i, update the volume according to

$$V^{(n+1)} = \frac{\sum_i^N \mathbf{P}_{\phi_i}^{\mathbf{T}} C_i X_i}{k + \sum_i^N \mathbf{P}_{\phi_i}^{\mathbf{T}} C_i^2}$$

Notes

1. C_i is the CTF corresponding to the image X_i.
2. The projection operator $\mathbf{P}_\phi$ also includes translations. So ϕ consists of five variables: $\phi = \{\alpha, \beta, \gamma, t_x, t_y\}$.
3. $\mathbf{P}_{\phi_i}^{\mathbf{T}}$ is the corresponding back projection operator. In Fourier space it yields a volume that is all zeros except for values along a slice.
4. The sum

$$\sum_i^N \mathbf{P}_{\phi_i}^{\mathbf{T}} C_i X_i$$

is therefore the insertion of *N* slices.

3D reconstruction in FREALIGN—iterations

1. Start with a preliminary structure $V^{(n)}, n = 1$

2. For each particle image X_i find the projection angles ϕ_i that gives the best match, so $X_i \approx C_i \mathbf{P}_{\phi_i} V^{(n)}$

Iterate

3. Use the Frealign iteration to produce a new 3D volume $V^{(n+1)}$

3D Classification in FREALIGN

Suppose our model is that an image X can come from any of K different particle types $V_1, V_2, .. V_K$ and our images are selected randomly from these volumes, projected with noise added.

1. The references are

$$R_{ik} = C_i \mathbf{P}_{\phi_i} V_k \, .$$

We assign k_i such that V_{k_i} yields the projection (with direction ϕ_i) that gives the highest correlation coefficient with X_i.

2. Update the volume according to

$$V_k^{(n+1)} = \frac{\sum_{i \in k} \mathbf{P}_{\phi_i}^{\mathbf{T}} C_i X_i}{k_w + \sum_{i \in k} \mathbf{P}_{\phi_i}^{\mathbf{T}} C_i^2}$$

Defocus contrast and the CTF

Correlation and particle picking

Single-particle reconstruction

Maximum-likelihood methods

Probabilities, another way to compare images

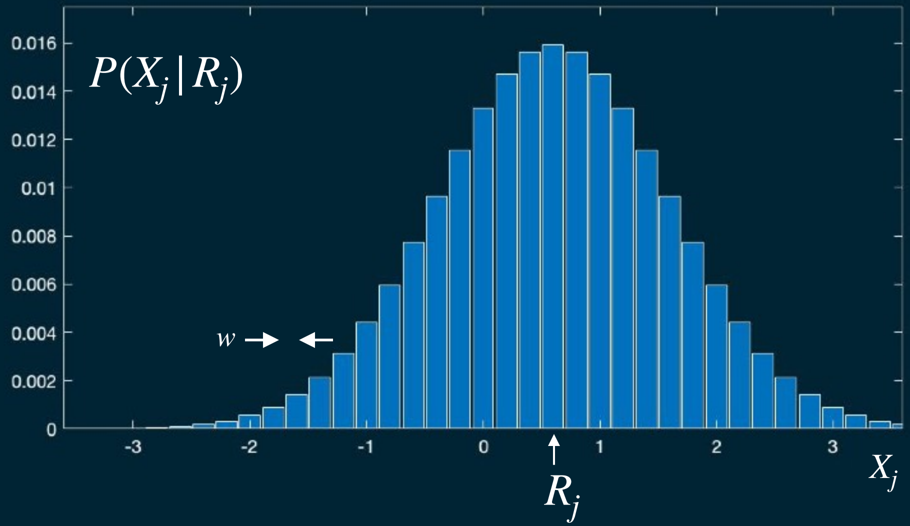
$$X = R + N$$

Probability of a pixel value:

$$P(X_j | R_j) = \frac{\cancel{w}\ 1}{\sqrt{2\pi\sigma^2}} e^{-(X_j - R_j)^2/2\sigma^2}$$

Probability of observing an image that comes from R:

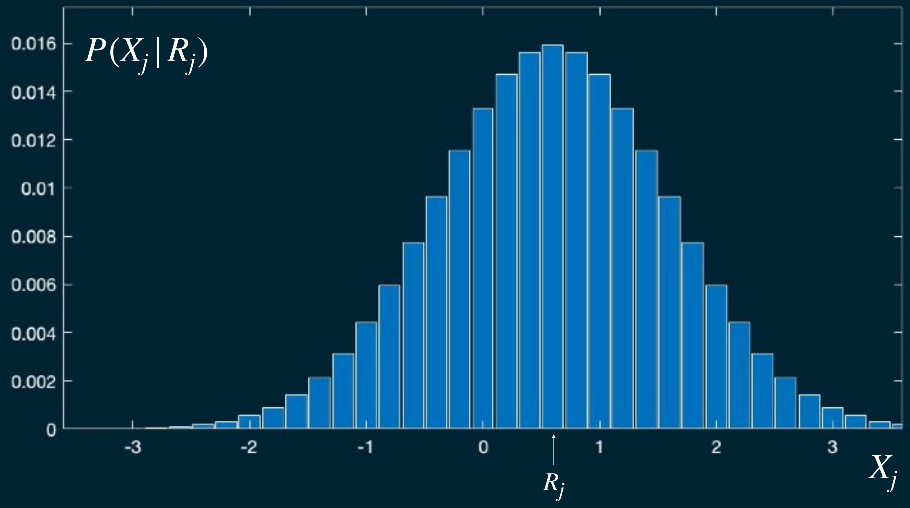
$$P(X | R) = \frac{\cancel{w^J}\ 1}{(2\pi\sigma^2)^{J/2}} e^{-||X-R||^2/2\sigma^2}$$



w is the finesse of the pixel intensity measurements. We'll ignore it (set it to 1).

Probabilities, another way to compare images

$$X = R + N$$



Probability of observing an image that comes from R:

$$P(X \mid R) = c\, e^{-||X-R||^2/2\sigma^2}$$

(The normalization factor c we'll treat as a constant and ignore it.)

The Likelihood

Let $\mathbf{X} = \{X_1 \ldots X_N\}$ be our "stack" of particle images. We'd like to find the best 3D volume consistent with these data, say maximizing

$$P(V \mid \mathbf{X}).$$

According to Bayes' theorem,

$$P(V \mid \mathbf{X}) = P(\mathbf{X} \mid V)\,\frac{P(V)}{P(\mathbf{X})}.$$

1. $P(\mathbf{X})$ doesn't depend on V so we can ignore it.
2. $P(V)$ is called the prior probability. It reflects any knowledge about V that we have before considering the data set.
3. $P(\mathbf{X} \mid V)$ is something we can calculate. It's called the likelihood of V.

$$\boxed{\mathrm{Lik}(V) = P(\mathbf{X} \mid V)}$$

Integrate over the projection directions to get the likelihood

We already know that

$$P(X \mid V, \phi) = ce^{-\|X - \mathbf{CP}_\phi V\|/2\sigma^2}$$

To get the likelihood for one image we just integrate over all the ϕ's:

$$P(X \mid V) = \int P(X \mid V, \phi)\, P(\phi)\, d\phi$$

To get the likelihood for the whole dataset we compute the product over all the images,

$$P(\mathbf{X} \mid V) = \prod_i^N \int P(X_i \mid V, \phi)\, P(\phi)\, d\phi,$$

or for numerical sanity, we compute the log likelihood,

$$L = \sum_i^N \ln \left(\int P(X_i \mid V, \phi)\, P(\phi)\, d\phi \right).$$

Maximum-likelihood estimation is asymptotically unbiased

If the size of the dataset grows without bounds
(and the number of parameters to be estimated do not)
ML converges to the right answer.

$$L = \sum_{i}^{N} \ln \left(\int P(X_i \,|\, V, \phi)\, d\phi \right).$$

To maximize the likelihood, we'll need a probability function $\Gamma(\phi)$

A projection

$$A = \mathbf{P}_\phi V$$

Probability of observing an image X_i

$$P(X_i \,|\, V, \phi) = c\, e^{-||X_i - \mathbf{C}\mathbf{P}_\phi V||^2/2\sigma^2}$$

Probability of a projection direction

$$\Gamma_i(\phi) = P(\phi \,|\, X_i, V) = \frac{P(X_i \,|\, V, \phi)}{\int P(X_i \,|\, V, \phi) d\phi}$$

The E-M algorithm finds a local maximum of the likelihood

The Expectation-Maximization (E-M) algorithm has this iteration, guaranteed to increase the likelihood:

$$V^{(n+1)} = \frac{\sum_i \int \Gamma_i^{(n)}(\phi)\mathbf{P}_\phi^\mathbf{T} C_i X_i \, d\phi}{\frac{\sigma^2}{T\tau^2} + \sum_i \int \Gamma_i^{(n)}(\phi)\mathbf{P}_\phi^\mathbf{T} C_i^2 \, d\phi}$$

For comparison, here is Frealign's iteration:

1. Find the best orientation ϕ_i for each particle image X_i
2. Update the volume according to

$$V^{(n+1)} = \frac{\sum_i \mathbf{P}_{\phi_i}^\mathbf{T} C_i X_i}{k + \sum_i \mathbf{P}_{\phi_i}^\mathbf{T} C_i^2}$$

…Relion's compute-intensive "Expectation" step is basically the evaluation of $\Gamma_i(\phi)$ for each image X_i

3D reconstruction in FREALIGN—iterations

1. Start with a preliminary structure $V^{(n)}, n = 1$

Iterate

2. For each particle image X_i find $\Gamma_i(\phi)$, the probability of projection angles ϕ.

3. Use the E-M iteration to produce a new 3D volume $V^{(n+1)}$

3D Classification

We can use Expectation-Maximization to optimize K different volumes $V_1, V_2, .. V_K$ simultaneously. The formula is essential the same except that the function Γ depends also on k:

$\Gamma^{(n)}_{\phi_i,k}$his iteration, guaranteed to increase the likelihood:

$$V_k^{(n+1)} = \frac{\sum_i \int \Gamma_{i,k}^{(n)}(\phi) \mathbf{P}_\phi^\mathbf{T} C_i X_i \, d\phi}{\frac{\sigma^2}{T\tau^2} + \sum_i \int \Gamma_{i,k}^{(n)}(\phi) \mathbf{P}_\phi^\mathbf{T} C_i^2 \, d\phi}$$

For comparison, here is Frealign's iteration:

1. Find the best orientation ϕ_i for each particle image X_i
2. Update the volume according to

$$V^{(n+1)} = \frac{\sum_i \mathbf{P}_{\phi_i}^\mathbf{T} C_i X_i}{k + \sum_i \mathbf{P}_{\phi_i}^\mathbf{T} C_i^2}$$

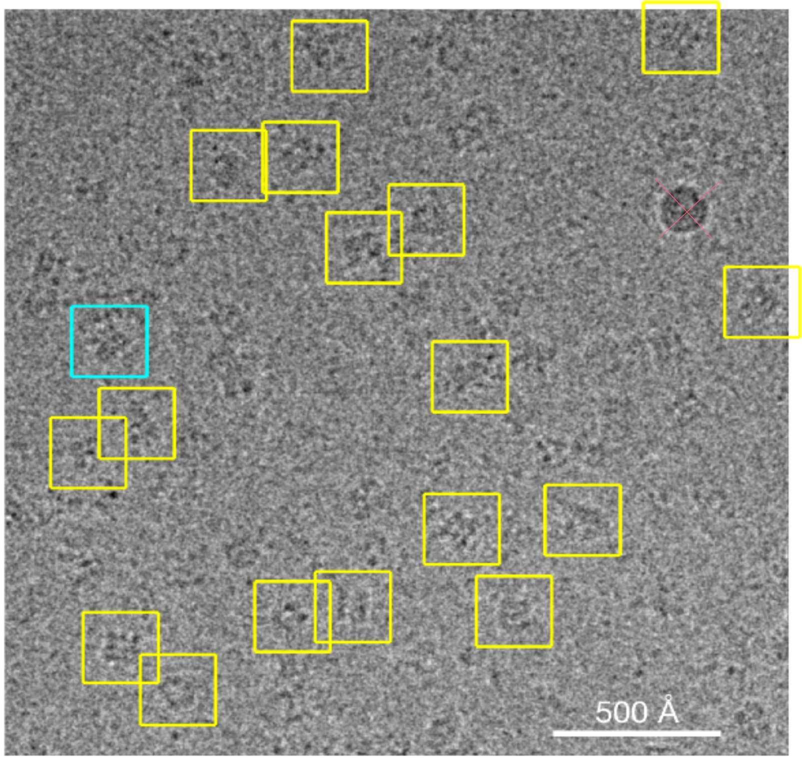
…Relion's compute-intensive "Expectation" step is basically the evaluation of $\Gamma_{i,k}(\phi)$ for each image X_i and volume V_k

Determining the orientation angles: example from the TRPV1 dataset

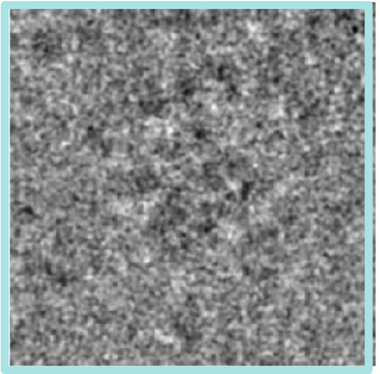
Structure of the TRPV1 ion channel determined by electron cryo-microscopy

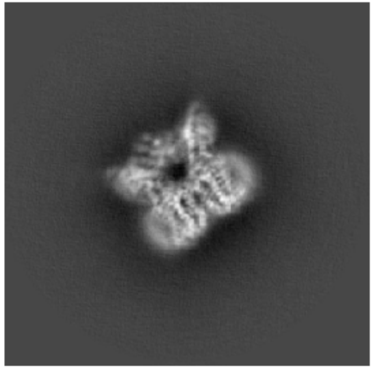
Maofu Liao[1]*, Erhu Cao[2]*, David Julius[2] & Yifan Cheng[1]

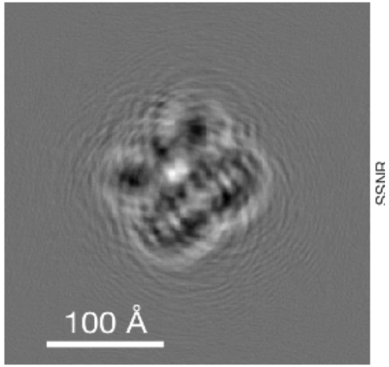
1/4 of a micrograph - empiar.org/10005

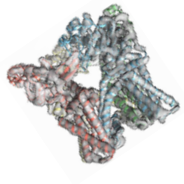


One particle image

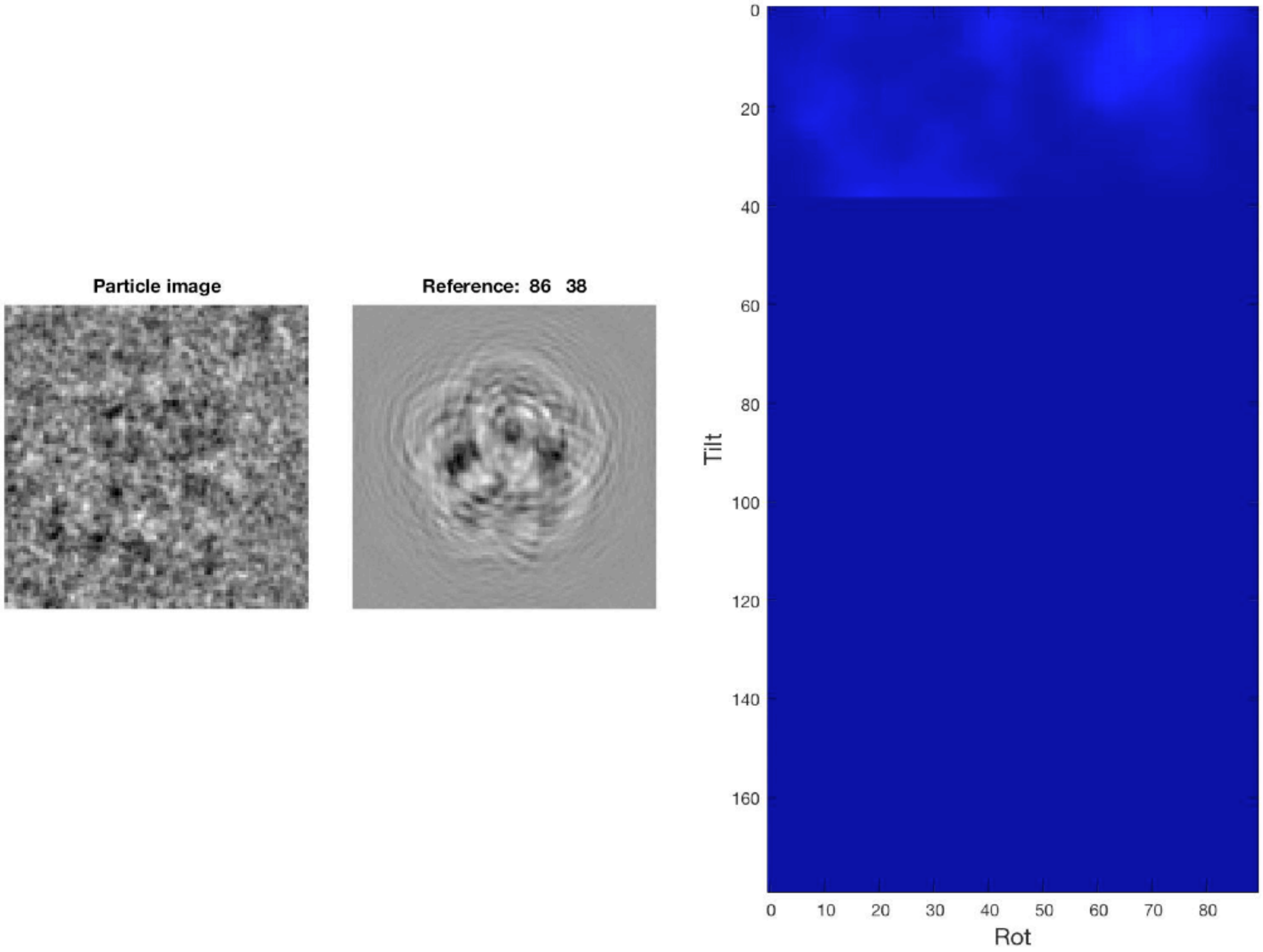




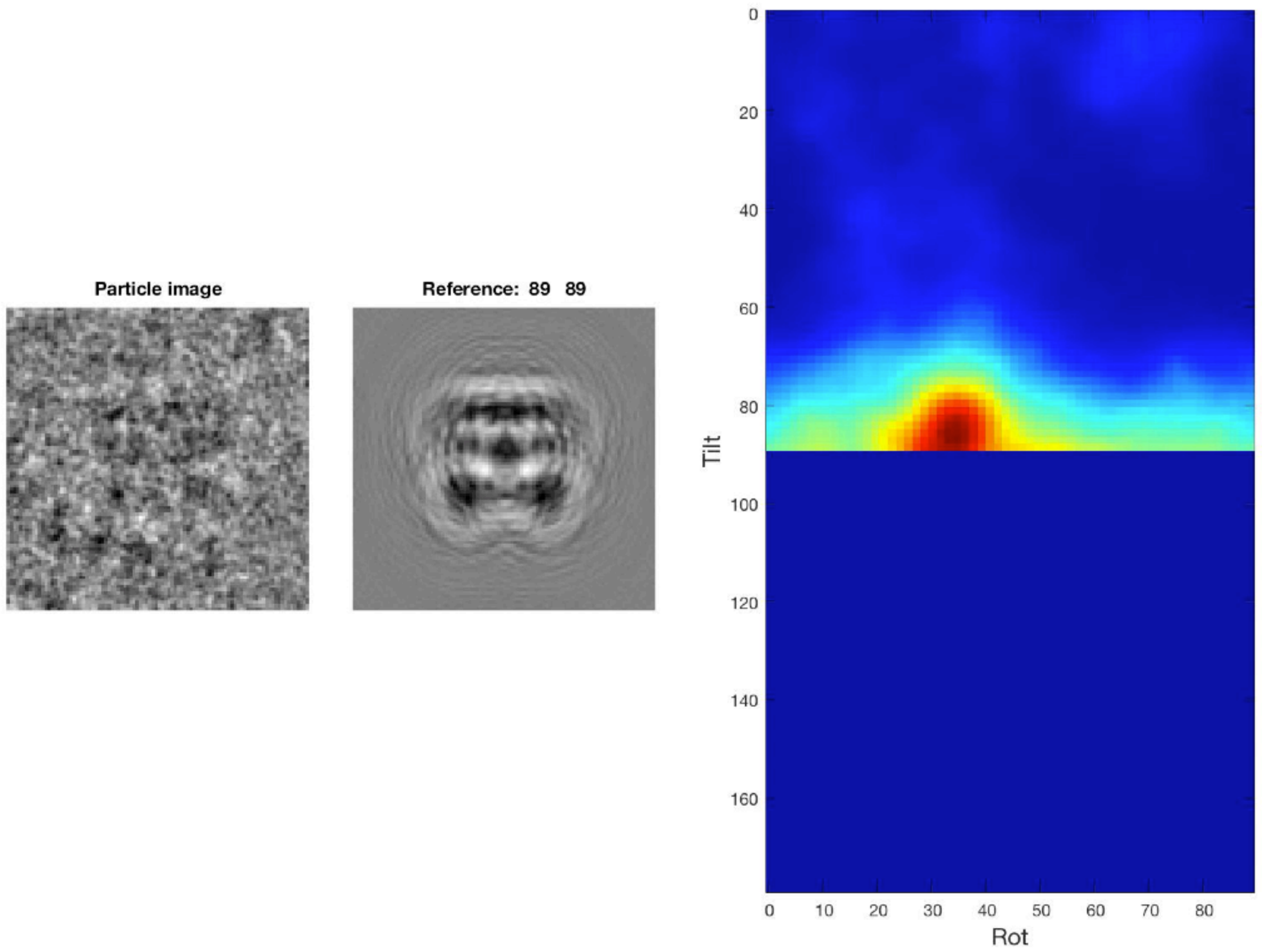




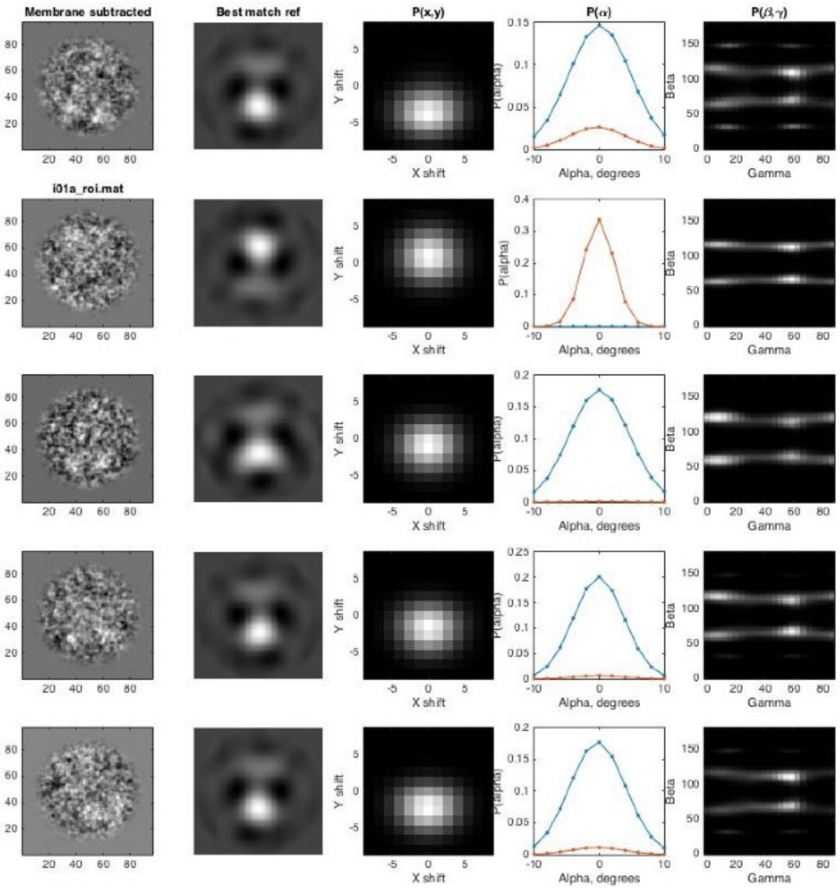
The probability of orientations $P(\phi \mid X, V)$ is remarkably sharp



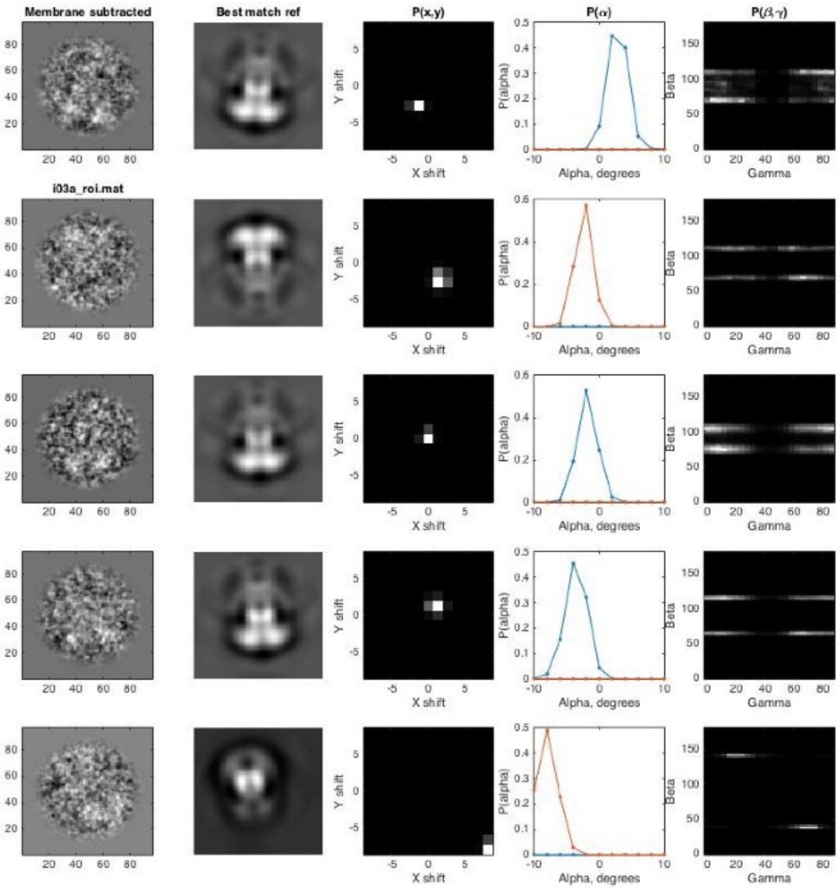
The probability of orientations $P(\phi \mid X, V)$ is remarkably sharp



Reconstruction: on the first EM iteration, angle assignments are not sharp

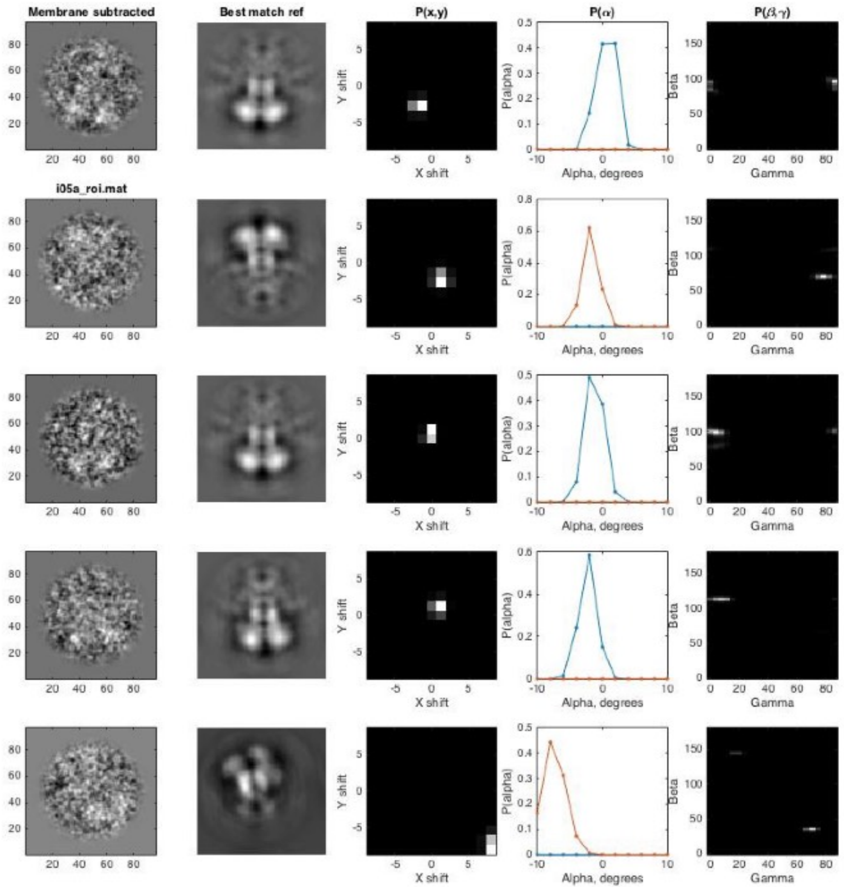


Iteration 3



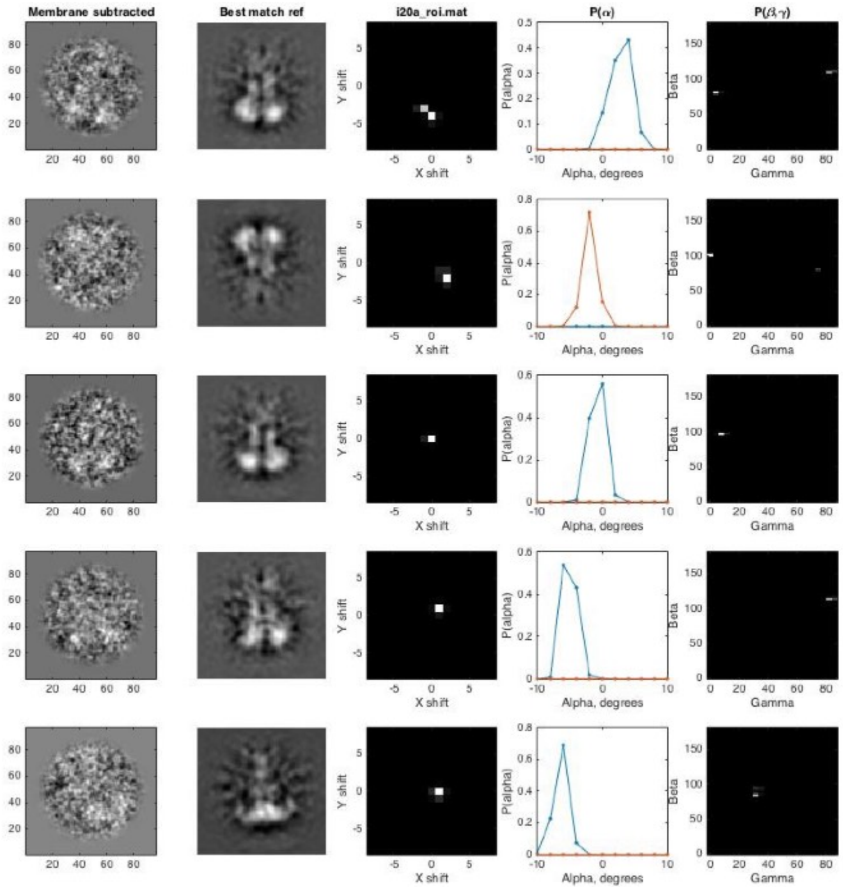
Membrane subtracted
Best match ref
P(x,y)
P(α)
P(β,γ)
i03a_roi.mat
X shift
Y shift
Alpha, degrees
P(alpha)
Gamma
Beta

Iteration 5



Membrane subtracted
Best match ref
P(x,y)
P(α)
P(β,γ)
i05a_roi.mat
X shift
Y shift
Alpha, degrees
P(alpha)
Gamma
Beta

Iteration 14, near convergence: distributions are becoming sharp



The orientation determination is the most expensive step

$$\text{No. operations} \approx \underbrace{\frac{\pi^3}{8} t^2 n^5 N}_{\text{finding orientations}} + \underbrace{\pi n^4 + N n^2}_{\text{3D reconstruction}}$$

The orientation determination is the most expensive step

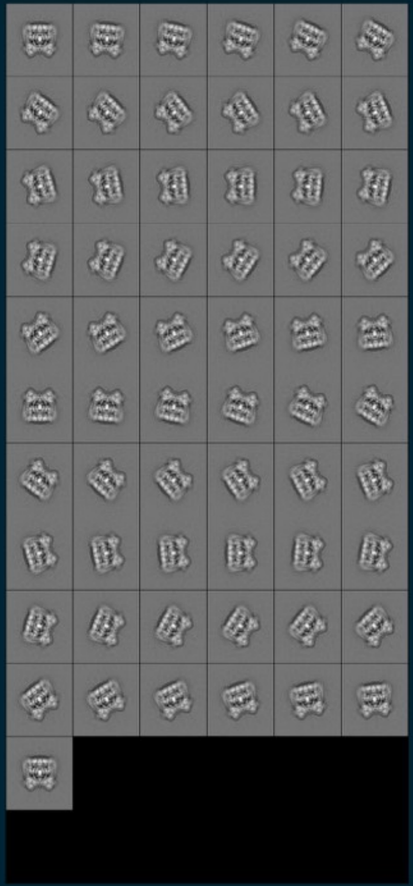
No. operations $\approx \underbrace{\frac{\pi^3}{8} t^2 n^5 N}_{\text{finding orientations}} + \underbrace{\pi n^4 + N n^2}_{\text{3D reconstruction}}$

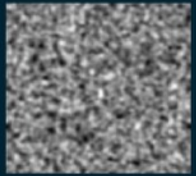
e.g. $N=10^5$, $n=128$, $t=7$

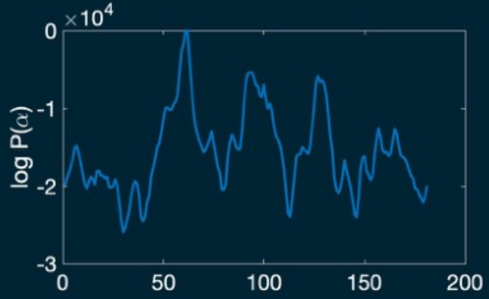
No. operations ≈ 6 x 10^{17} ≈ 19 CPU-years

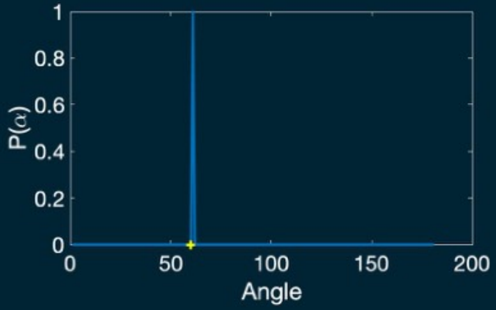
With efficient programs, ~ 1 CPU-month

Evaluating Γ_ϕ is expensive: one of 5 parameters

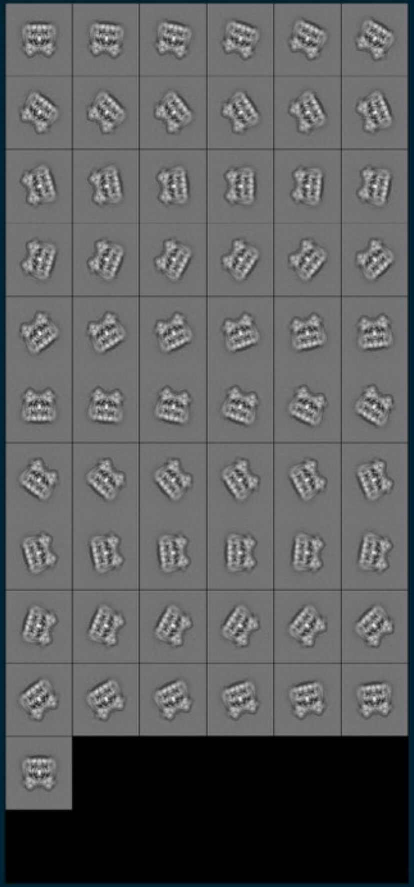


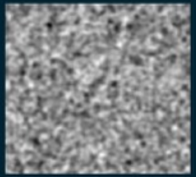


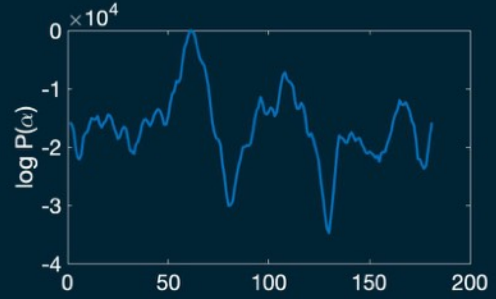


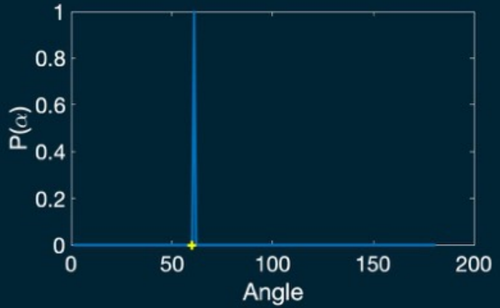


Evaluating Γ_ϕ is expensive: one of 5 parameters

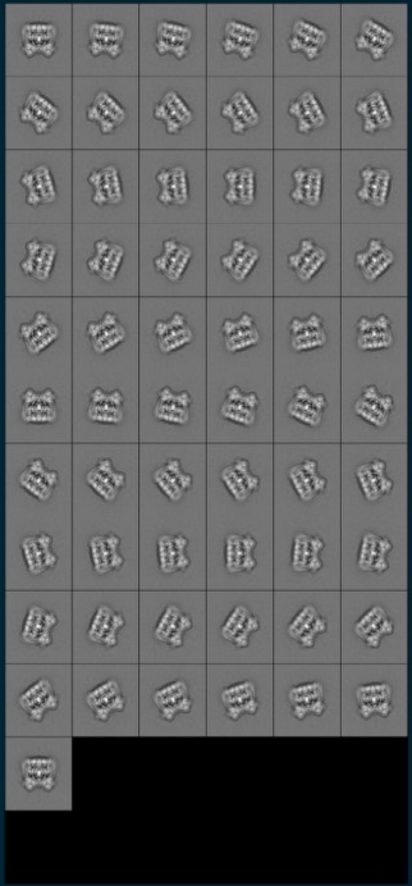


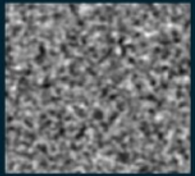


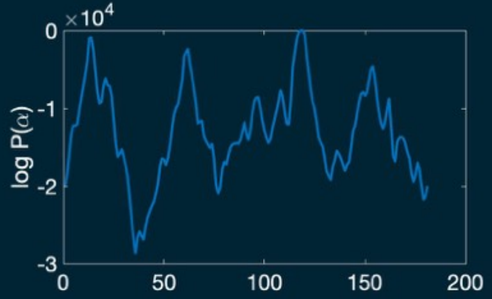


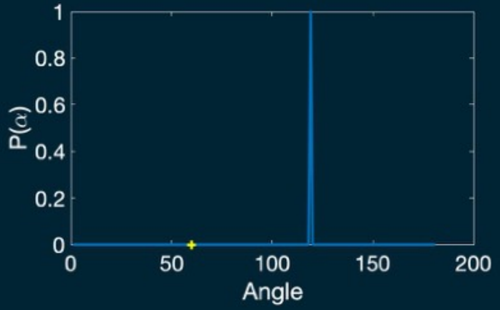


Evaluating Γ_ϕ is expensive: one of 5 parameters

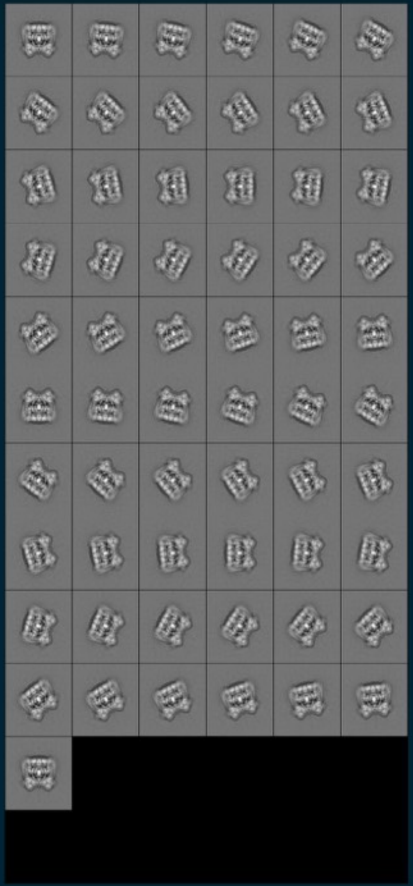


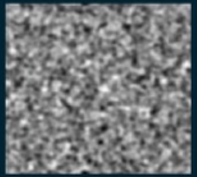


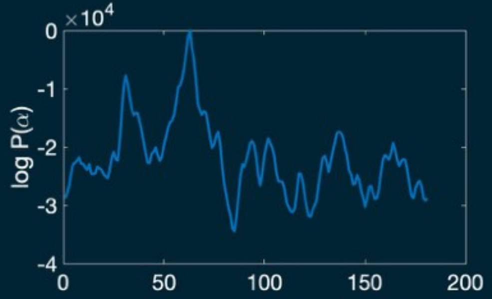


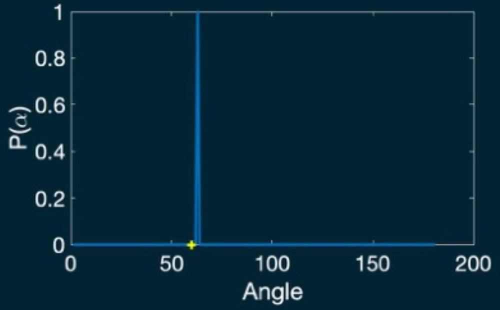


Evaluating Γ_ϕ is expensive: one of 5 parameters

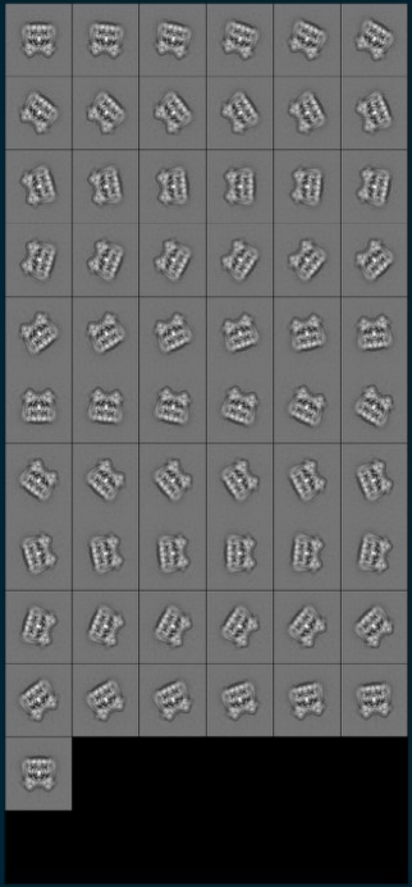


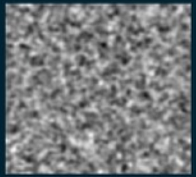


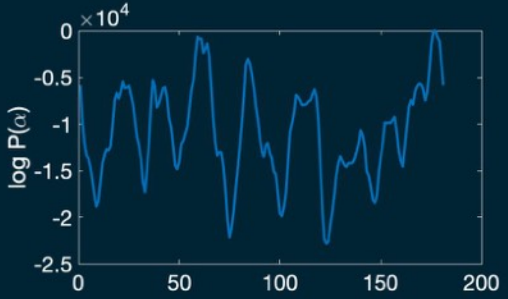


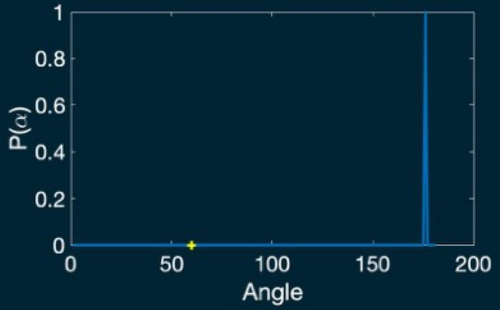


Evaluating Γ_ϕ is expensive: one of 5 parameters



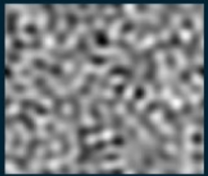




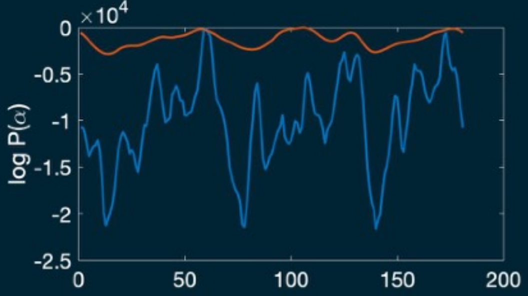


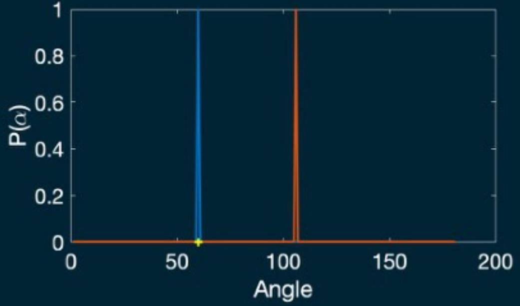
How to decrease the effort?

Domain reduction: branch and bound, illustrated for 1D



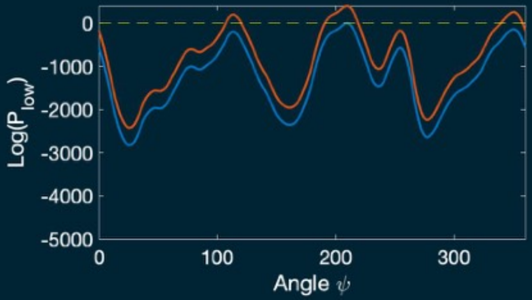
1. To save time, we compute probabilities of orientations at low resolution.

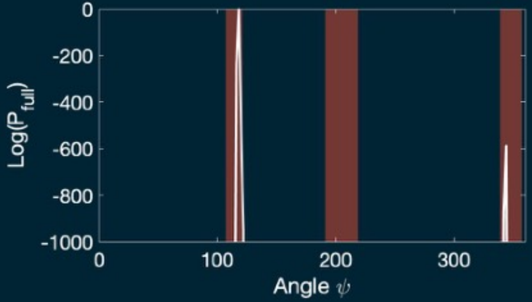


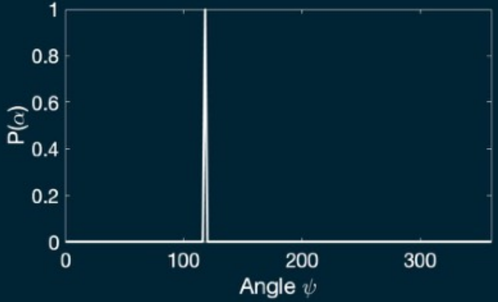


2. We place bounds on how much higher the probabilities could be at full resolution.

Given a cutoff value, we evaluate over a fraction of the domain.







In Relion, 2D and 3D classification and refinement use the same algorithm

Quantity	Meaning in 3D classification	Meaning in 2D classification
V_k	Class volume	Class average image
ϕ	3 Euler angles of orientation + 2 translations	1 angle of rotation + 2 translations
$\mathbf{P}_\phi$	Projection operator 3D $\longrightarrow$ 2D	Image rotation and shift
$\mathbf{P}_\phi^{\mathbf{T}}$	Back-projection operator 2D $\longrightarrow$ 3D	Reverse shift and rotation

data_particles

Interesting information in Relion's particle file

```
loop_
_rlnMicrographName #0
_rlnCoordinateX #1
_rlnCoordinateY #2
_rlnGroupName #3
_rlnDefocusU #4
_rlnDefocusV #5
_rlnDefocusAngle #6
_rlnOpticsGroup #7
_rlnClassNumber #8
_rlnAnglePsi #9
_rlnAutopickFigureOfMerit #10
_rlnImageName #11
_rlnAngleRot #12
_rlnAngleTilt #13
_rlnOriginXAngst #14
_rlnOriginYAngst #15
_rlnNormCorrection #16
_rlnLogLikeliContribution #17
_rlnMaxValueProbDistribution #18
_rlnNrOfSignificantSamples #19
_rlnGroupNumber #20
_rlnRandomSubset #21
Merged/2020-12-30_19_50_41_025035_1_240-1_0000_X-1Y-1-1_v.mrc  2927.910000  3601.070000   group_14 20129.600000 19778.200000
22.059200            1            1    10.822758    -999.00000 000003@Extract/job049/Merged/2020-12-30_19_50_41_025035_1_240-1_0000_X-1Y-1-1_v.mrcs
-24.83200     7.681132     4.095966     -3.44403     0.590449 54185.894732     0.264938           17             1            2

Merged/2020-12-30_19_50_41_025035_1_240-1_0000_X-1Y-1-1_v.mrc  1028.820000  3304.810000   group_14 20129.600000 19778.200000
22.059200            1            1    -76.56167    -999.00000 000006@Extract/job049/Merged/2020-12-30_19_50_41_025035_1_240-1_0000_X-1Y-1-1_v.mrcs
38.965843    12.031761     -2.68403     -3.44403     0.618694 54275.743065     0.086554           45             1            1

Merged/2020-12-30_19_50_54_025035_1_240-1_0001_X-1Y-1-2_v.mrc  2431.500000  3073.850000   group_14 19509.000000 18928.900000
10.654800            1            1   143.393906    -999.00000 000008@Extract/job049/Merged2020-12-30_19_50_54_025035_1_240-1_0001_X-1Y-1-2_v.mrcs
-4.83292   121.793929     -3.44403     1.835966     0.605107 54180.246800     0.093273           63             1            2
```