Towards Understanding and Optimising an ePIE for Electron Microscopy

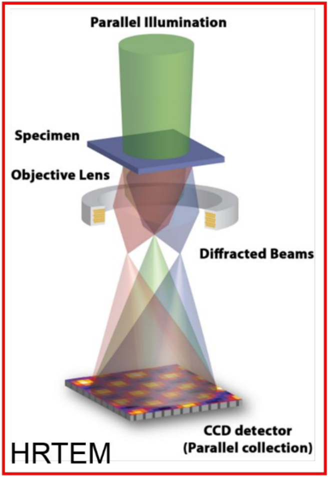
LU Shan

Supervisor: Prof. ZHU Ye,
Dr. CHEN Changsheng

Midterm Interview Presentation

Outline

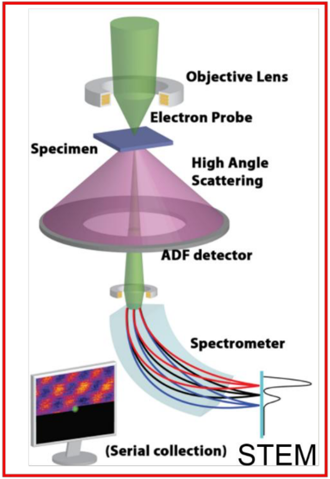
TEM vs. STEM Image Formation (1)



TEM (Transmission EM):

- Nearly parallel electron beam floods the sample.
- The **objective lens** after the specimen focuses the transmitted electrons to form a magnified image on a camera (detector).

TEM vs. STEM Image Formation (2)



STEM (Scanning TEM):

- A finely focused **electron probe** (from condenser lenses) scans across the sample.
- At each probe position, scattered electrons are collected by detectors (annular for dark-field), building up the image pixel by pixel.

TEM / STEM as a Forward Imaging Problem

In electron microscopy, the sample is described by a **complex transmission function**

$$O(\mathbf{r}) = \exp[i\sigma V(\mathbf{r})]\,,$$

where $V(\mathbf{r})$ is the projected electrostatic potential.

The exit wave after the specimen is

$$\psi_{\text{exit}}(\mathbf{r}) = P(\mathbf{r})O(\mathbf{r}),$$

with $P(\mathbf{r})$ denoting the incident probe or illumination.

What Is Actually Measured

In conventional TEM / STEM, the detector records only **intensity**:

$$I(\mathbf{k}) = |\mathcal{F}\{\psi_{\text{exit}}(\mathbf{r})\}|^2$$

- Phase information of ψ_{exit} is **lost**.
- Multiple distinct wavefunctions can produce the same intensity.
- The inverse problem becomes **ill-posed**.

Key limitation:

$$\psi(\mathbf{r}) \;\; \begin{array}{c}\longrightarrow \\ \not\longleftarrow\end{array} \;\; |\psi(\mathbf{r})|^2$$

Outline

1. Background and Motivation
2. Project Introduction
3. Results Obtained
   - Physics & Pipeline of PtyRAD
   - PtyRAD Reconstruction on $SrTiO_3$
4. Future Plan

From Phase Loss to Phase Retrieval

Key idea

Introduce **measurement diversity** to compensate for lost phase information.

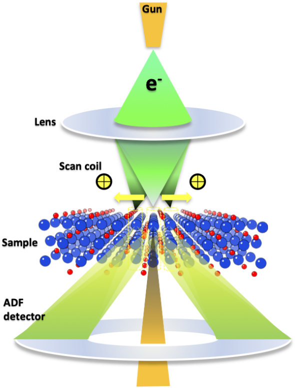
$$\text{Single measurement:} \quad I(\mathbf{k}) = \left|\mathcal{F}\{\psi(\mathbf{r})\}\right|^2 \qquad \Rightarrow \text{ill-posed}$$

$$\text{Multiple measurements:} \quad I_j(\mathbf{k}) = \left|\mathcal{F}\{\psi_j(\mathbf{r})\}\right|^2, \quad j = 1, \ldots, N$$

- Each measurement alone is insufficient.
- Redundancy across measurements provides additional constraints.

Question: how should these measurements be generated?

Electron Ptychography: Forward Model



Probe at position $\mathbf{R}_j$: $P_j(\mathbf{r}) = P(\mathbf{r} - \mathbf{R}_j)$

Exit wave: $\psi_j(\mathbf{r}) = P(\mathbf{r} - \mathbf{R}_j)\, O(\mathbf{r})$

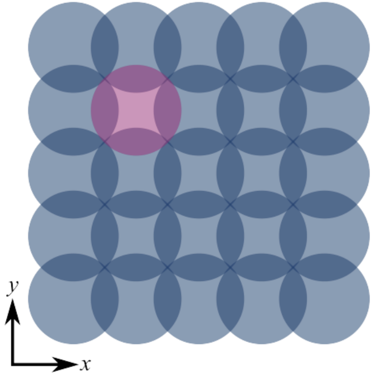
Measured intensity: $I_j(\mathbf{k}) = \left|\mathcal{F}\{\psi_j(\mathbf{r})\}\right|^2$

- Unknowns: object $O(\mathbf{r})$, probe $P(\mathbf{r})$.
- Overlap between adjacent probes introduces strong constraints.

Ptychographic Iterative Engine (PIE) (1)

Ptychographic measurement

A focused probe scans the specimen with overlapping illumination.



Assume Known Probe function!

- **Make an initial guess:** $O_{\text{guess}}(\mathbf{R_j})$
- **Known Probe:** $P(\mathbf{r} - \mathbf{R_j})$
- **Guess Exit Wavefuntion:** $\psi_{\text{guess}}(\mathbf{R_j}) = O_{\text{guess}}(\mathbf{R_j}) \times P(\mathbf{r} - \mathbf{R_j})$
- Note that we have the experimental measured result: $\sqrt{I(\mathbf{k_j})} = \left|\Psi_{\text{exp}}(\mathbf{k_j})\right|$

Ptychographic Iterative Engine (PIE) (2)

Iterative loop (conceptual):

$$\psi^{(n)}_{\text{guess}}(\mathbf{R_j}) = O^{(n)}_{\text{guess}}(\mathbf{R_j}) \times P(\mathbf{r} - \mathbf{R_j})$$

$$\Psi^{(n)}_{\text{guess}}(\mathbf{k_j}) = \mathcal{F}\left\{\psi^{(n)}_{\text{guess}}(\mathbf{R_j})\right\} = A^{(n)}_{\text{guess}}(\mathbf{k_j}) \exp\left(i\phi_{\text{guess}}(\mathbf{k}_j)\right)$$

$$\Psi^{(n)}_{\text{updated}}(\mathbf{k_j}) = |\Psi_{\text{exp}}(\mathbf{k_j})| \exp\left(i\phi_{\text{guess}}(\mathbf{k}_j)\right)$$

$$\psi^{(n)}_{\text{updated}}(\mathbf{R_j}) = \mathcal{F}^{-1}\left\{\Psi^{(n)}_{\text{updated}}(\mathbf{k_j})\right\}$$

$$O^{(n+1)}_{\text{guess}}(\mathbf{R_j}) = O^{(n)}_{\text{guess}}(\mathbf{R_j}) + P_{\text{norm}} \times P_{\text{filter}} \times \Delta\psi^{(n)}(\mathbf{R_j})$$

Stop until convergence!

- $\Delta\psi^{(n)}(\mathbf{R_j}) = \psi^{(n)}_{\text{updated}}(\mathbf{R_j}) - \psi^{(n)}_{\text{guess}}(\mathbf{R_j})$
- $P_{\text{norm}}, P_{\text{filter}}$ are carefully chosen "normalization" related to probe function.

Limitations of PIE

- Original PIE assumes a **known probe**.
- Sensitive to experimental imperfections.
- Convergence may depend strongly on initialization.

Motivation

Can we recover *both* the object and the probe simultaneously?

Extended PIE (ePIE)

Key extension

Treat both object and probe as unknowns and update them jointly.

Coupled updates (schematic):

$$O_{\text{guess}}^{(n+1)}(\mathbf{R_j}) \leftarrow O_{\text{guess}}^{(n)}(\mathbf{R_j}) + P_{\text{norm}} \times P_{\text{filter}} \times \Delta\psi^{(n)}(\mathbf{R_j})$$
$$P_{\text{guess}}^{(n+1)}(\mathbf{R_j}) \leftarrow P_{\text{guess}}^{(n)}(\mathbf{R_j}) + O_{\text{norm}} \times O_{\text{filter}} \times \Delta\psi^{(n)}(\mathbf{R_j})$$

- $O_{\text{norm}}, O_{\text{filter}}$ are carefully chosen "normalization" related to object function.
- Probe redundancy stabilizes reconstruction.
- More robust for experimental data.

Objective of this Project

Inspired by above reasoning, hopefully, we could complete followings:

1. Understanding the latest ePIE algorithm,
2. Trying to implement in our own group data (Make it works),
3. Optimising some functions or customizing some features to better suit our data.

Outline

1. Background and Motivation
2. Project Introduction
3. Results Obtained
   - Physics & Pipeline of PtyRAD
   - PtyRAD Reconstruction on $SrTiO_3$
4. Future Plan

Current Progress

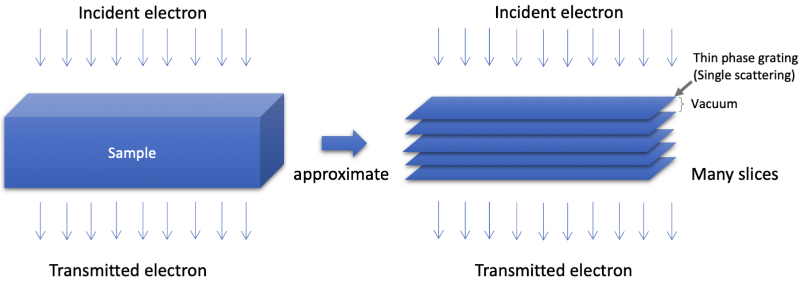
Since last semester, I was in exchange programme, then I mainly did some basic preparation works:

- Getting familiar with **ptychography**'s origins (from X-ray),
- Studying the PIE, and its variants, like ePIE, rPIE, mPIE etc,
- Delve into Ptychographic Reconstruction Framework with Automatic Differentiation (**PtyRAD**),
- Implemented the **PtyRAD** with our sample data on $SrTiO_3$.

Why **PtyRAD**?

- **Latest ePIE-based approach:** PtyRAD is a new iterative ptychography algorithm in the extended PIE family, designed to handle advanced physical models with high efficiency.
- **Optimization flexibility:** Traditional algorithms required manual update derivations for each new parameter. PtyRAD leverages automatic differentiation (AD) to seamlessly optimize many parameters (probe positions, sample tilt, slice spacing, mixed states) via gradient descent.
- **Computational efficiency:** Implemented in PyTorch, PtyRAD uses GPU-accelerated tensor operations and AD to achieve up to $\sim 17\times$ faster reconstructions than existing packages. It provides an open-source, modern replacement for slower, inflexible or proprietary ptychography codes.

Multislice Ptychography Model



- The object $O(\mathbf{r})$ is divided into N thin slices , $\{O_i(\mathbf{r})\}_{i=1}^{N}$, along the beam direction. At position $\mathbf{r_j}$, probe $P(\mathbf{r_j})$ propagates through slices sequentially, capturing multiple scattering.
- And last layers' exit wave is treated as next layers' incoming wave.

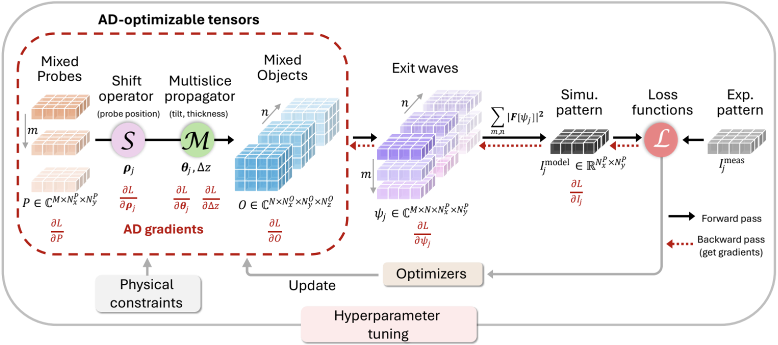
Mixed-State Probe & Object Modes

- PtyRAD uses a **mixed-state** formalism: represent probe as M incoherent modes $P^{(m)}(\mathbf{r})$ and object as N modes $O^{(n)}(\mathbf{r})$. Each mode pair produces a separate exit wave $\psi^{(m,n)}(\mathbf{r_j})$.
- The diffraction intensity at detector for position j is an incoherent sum over all mode combinations:

$$I(\mathbf{k_j}) = \sum_{m=1}^{M} \sum_{n=1}^{N} \left| \mathcal{F}\{\psi^{(m,n)}(\mathbf{r_j})\} \right|^2$$

The mixed-state model allows retrieval of multiple probe modes (partial coherence or mode instabilities) and multiple object modes (e.g. multi-layer specimens).

Schematic of the PtyRAD framework



Resulting Exit Wave by multislice algorithm

Using the standard multislice algorithm, for $P^{(m)}, O^{(n)}$ spliced into N_z, where z is the thickness, at $\mathbf{r_j}$:

$$\psi^{(m)(n)}(\mathbf{r_j}) = O^{(n)}_{N_z}(\mathbf{r_j}) \cdots \mathcal{F}^{-1}\left\{\mathcal{M}_{\theta_\mathbf{j},\Delta z} \cdot \mathcal{F}\left\{O^{(n)}_1(\mathbf{r_j}) \times P^{(m)}(\mathbf{r_j})\right\}\right\}$$

- θ_j is the local tilt at $\mathbf{r_j}$.
- Δz is the splice thickness.
- $\mathcal{M}_{\theta_\mathbf{j},\Delta z}$ is splice propagator related to $\theta_j, \Delta z$.

Then after calculating each one, the model intensity at $\mathbf{k_j}$ ($\mathbf{r_j}$):

$$I(\mathbf{k_j}) = \sum_{m=1}^{M} \sum_{n=1}^{N} \left| \mathcal{F}\{\psi^{(m,n)}(\mathbf{r_j})\} \right|^2$$

Loss Function and Optimization (1)

The total Loss Function is decomposed into four parts:

$$\mathcal{L}_{\text{total}} = \omega_1 \mathcal{L}_{\text{Gaussian}} + \omega_2 \mathcal{L}_{\text{Poisson}} + \omega_3 \mathcal{L}_{\text{PACBED}} + \omega_4 \mathcal{L}_{\text{sparse}}$$

With user-controlled weights: $\{\omega_i\}_{i=1}^{4}$, and loss functions defined below, where $\mathcal{D}, \mathcal{B}, \mathcal{R}$ refer to detector, batch, spatial dimension:

- **Gaussian Loss:** (with default $p = 0.5$)

$$\mathcal{L}_{\text{Gaussian}} = \frac{\sqrt{\left\langle \left(I^{p}_{\text{model}} - I^{p}_{\text{meas}} \right)^2 \right\rangle_{\mathcal{D},\mathcal{B}}}}{\langle I^{p}_{\text{meas}} \rangle_{\mathcal{D},\mathcal{B}}}$$

Loss Function and Optimizatio (2)

- **Poisson Loss:** (with default $p = 1$, $\varepsilon = 10^{-6}$)

$$\mathcal{L}_{\text{Poisson}} = -\frac{\left\langle I_{\text{meas}}^{p} \log\left(I_{\text{model}}^{p} + \varepsilon\right) - I_{\text{model}}^{p}\right\rangle_{\mathcal{D},\mathcal{B}}}{\left\langle I_{\text{meas}}^{p}\right\rangle_{\mathcal{D},\mathcal{B}}}$$

- **Position-Averaged Convergent Beam Electron Diffraction:** (with default $p = 1$)

$$\mathcal{L}_{\text{PACBED}} = \frac{\sqrt{\left\langle \left(\langle I_{\text{model}}\rangle_{\mathcal{B}}^{p} - \langle I_{\text{meas}}\rangle_{\mathcal{B}}^{p}\right)^{2}\right\rangle_{\mathcal{D}}}}{\left\langle I_{\text{meas}}^{p}\right\rangle_{\mathcal{D},\mathcal{B}}}$$

- **Sparse**: (with default $p = 1$)

$$\mathcal{L}_{\text{sparse}} = \langle |O_p|^{p} \rangle_{\mathcal{R},\mathcal{B}}^{\frac{1}{p}}$$

Updating Object & Probe

In PtyRAD, we minimize a total loss to update key variables:

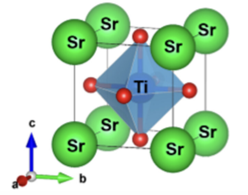
- **Object Transmission** $O^{(n)}(\mathbf{r_j})$,
- **Probe Wavefunction** $P^{(m)}(\mathbf{r_j})$,
- **Slice Spacing** Δz,
- **Local Tilt** θ_j (**Unique improvement compared with py4DSTEM**).

All updates are performed via **automatic differentiation (AD)** using PyTorch.

Hyperparameters Tuning

- Automatic parameter selection by **Bayesian Optimization**: Convergence in iterative optimization problems is highly sensitive to algorithmic parameters such as batch sizes, learning rates, and other configurational settings which are generally referred to as hyperparameters.

Experimental Setup

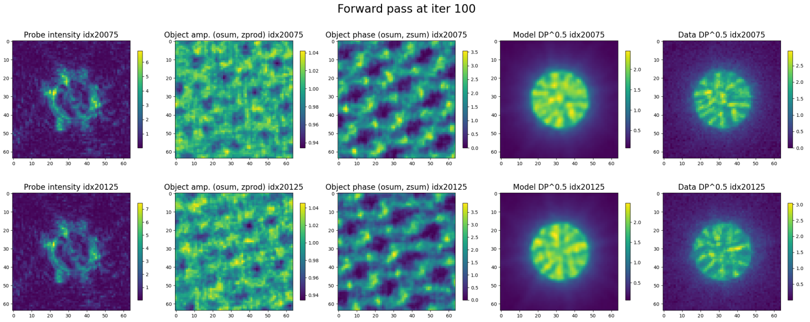


We are considering $SrTiO_3$:

- $N = 40000 = 200 \times 200$,
- Probe: 300 KeV electron at 28.9 mRad,
- Total Iteration: 100,
- For loss function we only consider:
  1. Gaussian ($\omega_1 = 1, p = 0.5$),
  2. sparse loss ($\omega_4 = 0.1, p = 1$).

Note here we used centered mode, which focus on $[50, 150] \times [50, 150]$.

Forward Consistency(1)

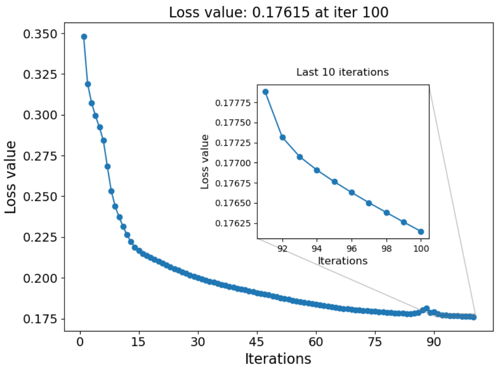


Forward Consistency (2)

What we could improve:

- Add a **residual map**: $\left|\sqrt{I_{\mathrm{sim}}} - \sqrt{I_{\mathrm{meas}}}\right|$ (iter 0/50/100/final).
- Refine detector center/background; enable PACBED loss if needed.
- Continue iterations and then switch `INDICES_MODE: center` $\rightarrow$ `full`.

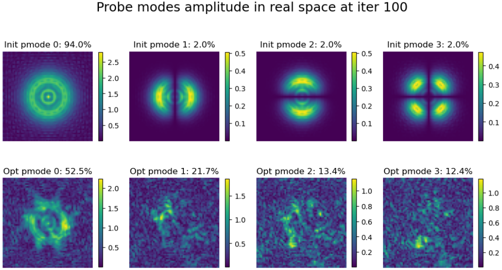
Convergence (Loss vs. Iterations)



What to improve next:

- Learning-rate schedule (warm-up $\rightarrow$ decay) to reduce late-stage plateau.
- Stage-wise optimization: freeze position/probe early, then unfreeze.

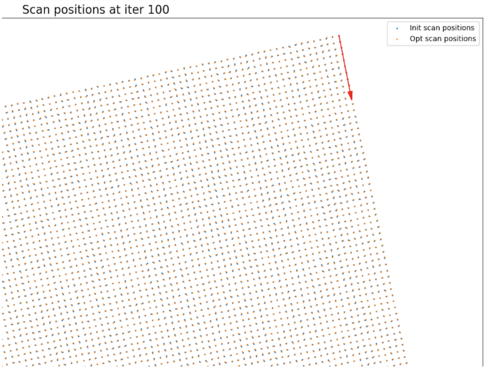
Mixed-State Probe Modes (Partial Coherence)



What to improve next:

- Check probe–object leakage: add stronger constraints / orthogonalization frequency.
- Try fewer/more modes as an ablation, then increase if needed.

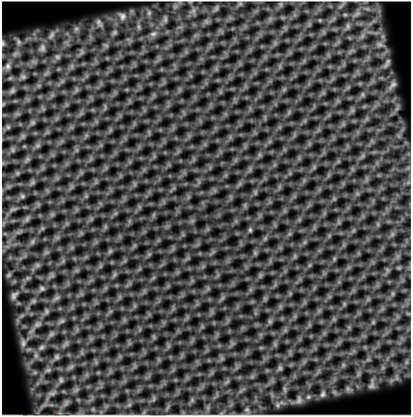
Scan Position Refinement (Drift/Distortion Correction)



What to improve next:

- Report mean displacement statistics as a quantitative figure.
- After center convergence, re-run full field for final refined positions.

Reconstructed $SrTiO_3$ Object (Multislice z-sum)



Sliced into 32 layers, each thinkness 10 $\AA$

What to improve next:

- Increase iterations / add final L-BFGS refinement after Adam.
- Validate with a baseline: virtual ADF image from raw 4D data (side-by-side).

Outline

Make it Works → Make it Fast & Robust

- **Phase A: Make it works (integration)**: reproduce PtyRAD-style AD reconstruction on *our group data* with a clean, modular pipeline.
- **Phase B: Optimize & customize (engineering)**: profile bottlenecks, tune hyperparameters, and add features that better match our experimental conditions.

Timeline and Expected Outcomes (1)

(1) **January – Understand & Replicate:**
- Read PtyRAD structure ⇒ code map + parameter schema.
- Reproduce official demo(s) and confirm numerical sanity.
- Forward DP + loss curve reproducible on reference dataset.

(2) **February – Test on Own Data:**
- Run on a controlled subset (center / smaller batches).
- Produce **standard figure set**: forward (with residual), loss, probe modes, scan positions, z-sum.
- End-to-end pipeline works on group dataset with stable convergence.

Timeline and Expected Outcomes (2)

(3) **March – Integrate & Adapt:**

- Integrate modules into group codebase (clean interfaces).
- Run ablations: slice/mode/loss/INDICES mode; document stable defaults.
- Selected physics options justified for our experiments.

(4) **April – Optimize & Finalize:**

- Profiling-driven speedups; finalize runtime benchmark.
- Final reconstructions on target dataset; prepare report-ready tables/figures.
- Deliverables: updated group code + reproducible scripts + final FYP report.

Q&A

Thank you for your attention!

(Questions and discussion)