# An improved workflow for reproducible processing and analysis of polycrystalline electron diffraction patterns

John R. Minter, Analytical Sciences, Eastman Kodak Co., Rochester, NY

Kodak

## Objective: Efficient, reproducible analyses

Microscopists rarely record and analyze a single image or diffraction patterns. . .

**Goal: eliminate barriers to reproducibility**

- Repetitive manual instrument adjustments
- Interaction with (multiple) software GUIs
- "Black box" software
- Copy/paste report generation

**Plan:** Develop a workflow that maximizes scripting of the analysis and implements "literate programming" [1] – where code chunks are embedded in the report, generating the results during compilation.

[1] D. Knuth, The Computer Journal **27** (1984) p. 97–111.

## Step 1: Reproducible pattern acquisition

Do the best we can with a 20 yr old CM20 with an early, limited RS-232 interface and a single grid specimen holder. . .

- Select conditions for unknown. Use "normalization" to correct for hysteresis.
- Record unknown pattern first & save lens conditions
- Restore lens conditions with standard and focus with stage control

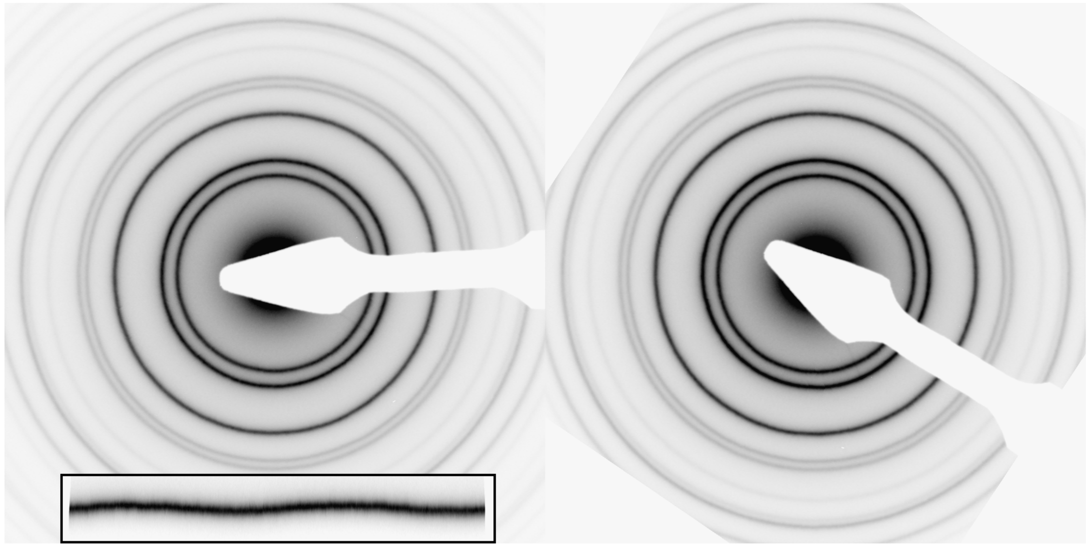
## Step 2: Reproducible distortion correction

Extended Hou and Li's [2] "DiffractionDistortion" plug-in for *DigitalMicrograph* to correct the elliptical distortion and perform the radial average.

- Still requires an "initial guess" for center.
- Uses the approach of Capitani *et al.* [3] to correct distortion (rotate to major axis and scale to a circle.)
- Performs the radial average, writes a '.csv' file for analysis.
- New plug-in "edp" (Open Source, version control) designed to save choices & maximize user efficiency.

[2] V Hou and D Li, Microscopy Today **16** (May 2008) p. 36-41.
[3] G Capitani *et al.*, Ultramicroscopy **106** (2006) p. 66-74.

## Processed NiO Diffraction Pattern

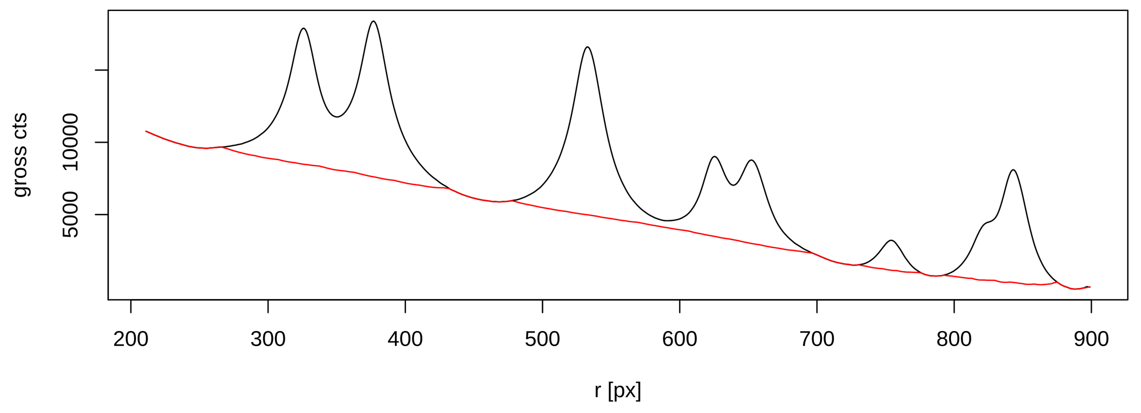


**Left:** Inverted diffraction pattern from NiO thin film. Bottom inset: "Unwound" angular image exhibiting the sinusoidal pattern characteristic of elliptical distortion produced by spherical aberration. **Right:** Corrected pattern after rotation and bi-axial scaling.

## Step 3: Background correction

We use the Open Source R statistical programming language for our post-processing. This is easily scripted, and frequently used operations assembled in packages maintained under version control, so an individual analysis needs few lines of code. Ours is encapsulated in the "edp" package.

The function, `tune.edp.baseline.dm()`, permits the analyst to find the best spline parameters (size of the averaging window and polynomial order) for a given material. This function uses algorithms developed by Morháč [4] and supplied in the "Peaks" package.

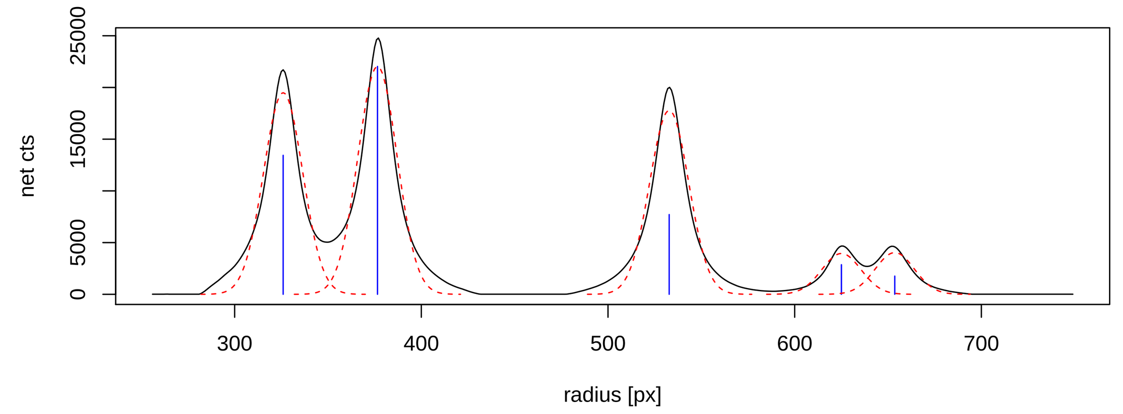



These parameters typically only need to be tuned once for an new material.

[4] M. Morháč *et al.*, Nuclear Instruments and Methods in Physics Research A **401** (1997) p. 113-132.
*N.B.*: The Peaks package on CRAN has not been properly updated to accommodate the changes in loading shared C libraries in R $\geq$ 3.0.0. We have a patched version in our github repository.

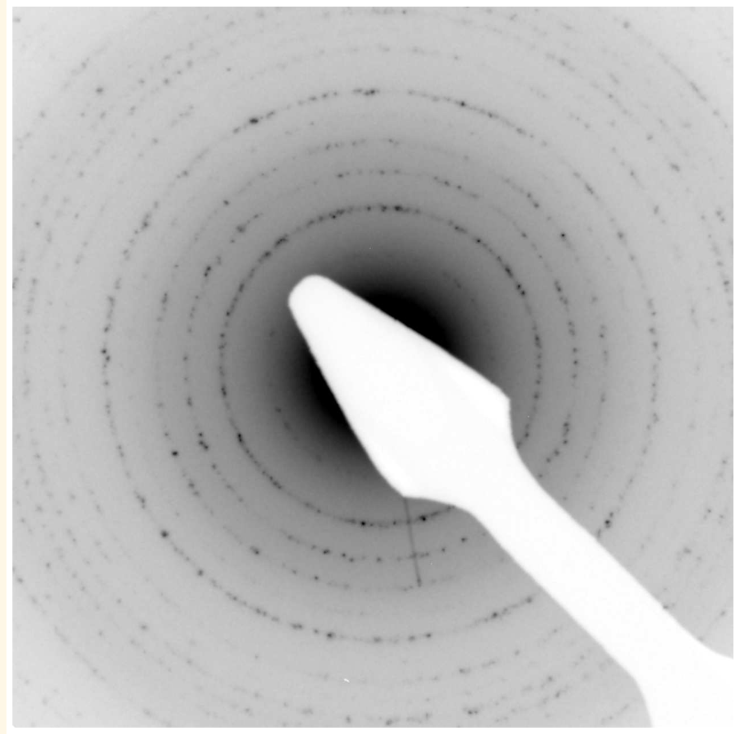
## Step 4: Peak detection & calibration

The "Peaks" package provides a robust peak detection algorithm implemented as the `SpectrumSearch()` function, which is easily tuned to detect diffraction peaks. Typically, we use the first 5 peaks for calibration. The `edp` package makes an initial guess of the camera constant from the highest peak and then performs a Gaussian fit to the background subtracted spectrum.

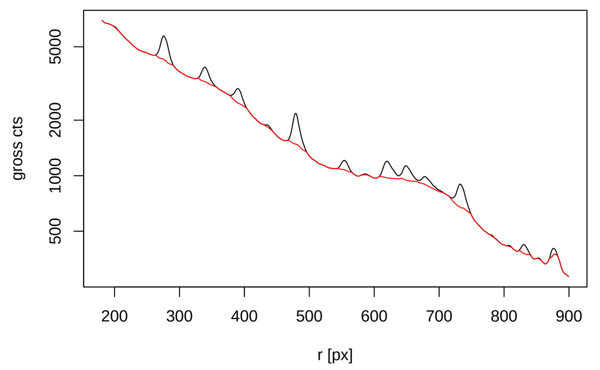



The background-subtracted spectrum is plotted in black. The pattern was then modeled as the sum of five Gaussian peaks (red dashes) with an initial guess of the uniform half-width at half height of 10 px. The fit peak positions are shown in blue. The camera constant estimated from the fit was 786.89 $\pm$ 0.23 px-Å. The camera constant from an Al thin film measured under the same conditions was 787.84 $\pm$ 0.18 px-Å, a difference of 0.03% – a respectable precision for this instrument.

## A test: $BaTiO_3$ nanoparticles

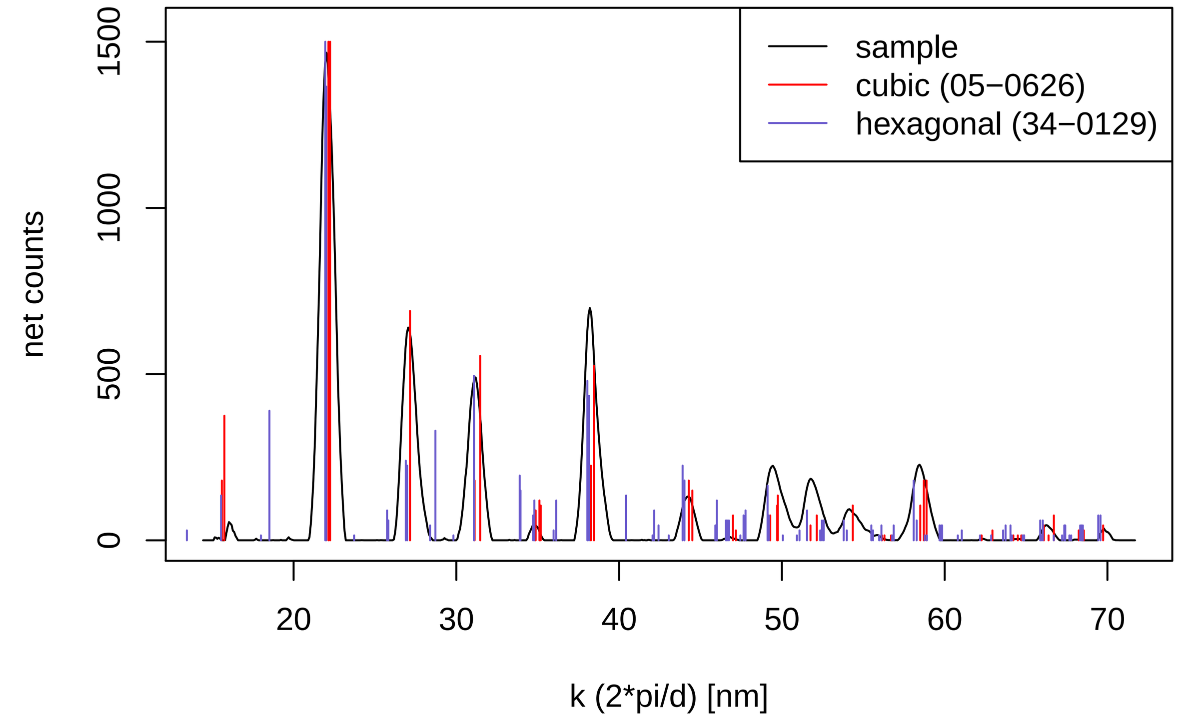


Distortion-corrected EDP




This pattern has more structure. Background tuning requires a more complicated spline, but does well except at the highest radius.

## Step 5: Comparison with standards




Note that the agreement with the standard cubic phase is better than the hexagonal phase.

## Key benefits

- Scripted analyses – Using R/Sweave/LaTeX for analysis and report generation permits "code-chunks" in the report to generate statistical and analysis and graphics with minimal "cut & paste." *N.B.* - results from computationally-intensive steps may be cached. The entire analysis (for example, this poster) may be replicated with a single command. All the documents are text based and work efficiently with version control. RStudio provides a comfortable development environment.
- Reuse – Packaging common code into packages minimizes duplication and is aided by version control.
- Easy extension – An analyst can extend a project with much less effort. Preparing a compressed compendium (7-zip) for a project permits an analyst to replicate a result in less than 15 minutes.

## Acknowledgements

The author is grateful to V. Hou for his willingness to both share the source code to his `DiffractionDistortion` plug-in and to grant permission to extend it and distribute the derivative work. A former colleague, M. Dirmeyer, supplied the dispersion of $BaTiO_3$ nanoparticles.

More information – email: john.r.minter@kodak.com, github: https://github.com/jrminter/edp.git